Short-term flood forecasting with a neurofuzzy model

P. C. Nayak,[1] K. P. Sudheer,[2] D. M. Rangan,[1] and K. S. Ramasastri[3]

Received 11 August 2004; revised 30 December 2004; accepted 13 January 2005; published 5 April 2005.

[1] This study explores the potential of the neurofuzzy computing paradigm to model the rainfall-runoff process for forecasting the river flow of Kolar basin in India. The neurofuzzy computing technique is a combination of a fuzzy computing approach and an artificial neural network technique. Parameter optimization in the model was performed by a combination of backpropagation and least squares error methods. Performance of the neurofuzzy model was comprehensively evaluated with that of independent fuzzy and neural network models developed for the same basin. The values of three performance evaluation criteria, namely, the coefficient of efficiency, the root-mean-square error, and the coefficient of correlation, were found to be very good and consistent for flows forecasted 1 hour in advance by the neurofuzzy model. The value of the relative error in peak flow prediction was within reasonable limits for the neurofuzzy model. The neurofuzzy model forecasted 47.95% of the total number of flow values 1 hour in advance with less than 1% relative error, while for the neural network and fuzzy models the corresponding values were 36.96 and 18.89%, respectively. The forecasts by the neurofuzzy model at higher lead times (up to 6 hours) are found to be better than those from the neural network model or the fuzzy model, implying that the neurofuzzy model seems to be well suited to exploit the information to model the nonlinear dynamics of the rainfall-runoff process.

Citation: Nayak, P. C., K. P. Sudheer, D. M. Rangan, and K. S. Ramasastri (2005), Short-term flood forecasting with a neurofuzzy model, *Water Resour. Res.*, *41*, W04004, doi:10.1029/2004WR003562.

1. Introduction

[2] River flow forecasting has always been one of the most important issues in hydrology and it is an essential measure in water resource development and planning. Forecasting a river flow provides a warning of impending stages during floods and assists in regulating reservoir outflows during low river flows for water resource management. To date, a plethora of rainfall-runoff models belonging to different categories is available for flow forecasting purposes. Most of the rainfall-runoff models have been developed based either on physical considerations of the process or on a systems theoretic approach. In the physical approach, the primary motivation is the study of physical phenomena and their understanding, while in the system theoretic approach the concern is with the system operation, not the nature of the system by itself or the physical laws governing its operation.

[3] Although the conceptual (physics based) models provide reasonable accuracy, the implementation and calibration of conceptual models can typically present various difficulties [*Duan et al.*, 1992], requiring sophisticated mathematical tools, a significant amount of calibration data, and some degree of expertise and experience with the model. The need for a system theoretic approach arises in many instances, primarily from the complexities inherent in the physical approach and a belief that the system behavior may be approximated by major physical factors, thus permitting the neglect of minor features and exact spatial features of the physical parameters [*Sudheer*, 2000]. In this context data-driven models, which can discover relationships from input-output data without having the complete physical understanding of the system, may be preferable. While such models do not provide any physics of the hydrologic processes, they are in particular, very useful for river flow forecasting where the main concern is with making accurate predictions of flow at specific watershed locations.

[4] Traditionally, autoregressive moving average (ARMA) models have been used for modeling and forecasting water resource time series because such models are accepted as a standard representation of a stochastic time series [*Maier and Dandy*, 1997]. The method that is based on a statistical approach makes use of classical statistics to analyze the historical data with an objective to develop methods for the formulation of flood forecasts [e.g., *Box and Jenkins*, 1976; *Salas and Obeysekera*, 1982; *Sharma*, 1985]. However, such models do not attempt to represent the nonlinear dynamics inherent in the transformation of rainfall to runoff and therefore may not always perform well [*Hsu et al.*, 1995]. Owing to the difficulties associated with nonlinear model structure identification and parameter estimation, very few truly nonlinear system theoretic watershed models have been reported [*Jacoby*, 1966; *Amorocho and Brandstetter*, 1971; *Ikeda et al.*, 1976]. In most cases, linearity or piecewise linearity has been assumed [*Hsu et al.*, 1995]. Therefore it seems necessary that the conven-

[1] National Institute of Hydrology, Deltaic Regional Centre, Siddartha Nagar, Kakinada, India.
[2] Department of Civil Engineering, Indian Institute of Technology Madras, Chennai, India.
[3] National Institute of Hydrology, Roorkee, India.

0043-1397/05/2004WR003562$09.00

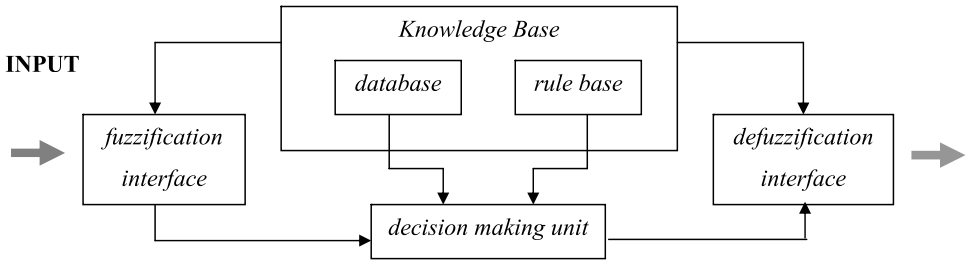

Figure 1. Fuzzy inference system.

tionally applied modeling techniques be refined or complemented to achieve improved performance by implementing new or different technologies.

[5] During the last two decades, the tools that engineers and scientists work with have improved significantly. The rapid growth of computing power has enabled the researchers to develop effective modeling tools. One of the most exciting ideas that emerged from the vast pool of computer-based research is the thought of emulating the low-level mechanism of the human brain through artificial neural networks (ANN). Already, useful applications have been designed, built and commercialized, and much research continues in the hope of extending this success [*Haykin*, 1994]. Similarly, the fuzzy rule based approach in modeling, introduced by *Zadeh* [1965], is also being widely used in various fields of science and technology [*Chang and Chang*, 2001; *Nayak et al.*, 2004]. The reason for such an increasing interest resides in their intrinsic generality, flexibility, and global performance in most applications where other models either tend to fail or become cumbersome [*Shamseldin et al.*, 2002]. Both these intelligent computing methods have far-reaching potential as building blocks in today's computational world.

[6] Artificial neural networks (ANN) are essentially semiparametric regression estimators and are well suited for hydrologic modeling [*Connor et al.*, 1994; *Atiya et al.*, 1999; *Babovic and Keijzer*, 1999], as they can approximate virtually any (measurable) function up to an arbitrary degree of accuracy [*Hornik et al.*, 1989]. The emergence of neural network technology has provided many promising results in the field of hydrology and water resource simulation [*ASCE Task Committee on Application of Artificial Neural Networks in Hydrology*, 2000a, 2000b; *Dawson and Wilby*, 2001]. Fuzzy rule based modeling is a qualitative modeling scheme where the system behavior is described using a natural language [*Sugeno and Yasukawa*, 1993]. The last decade has witnessed a few applications of fuzzy logic in water resource forecasting [*Fujita et al.*, 1992; *Zhu and Fujita*, 1994; *Zhu et al.*, 1994; *Stuber et al.*, 2000; *See and Openshaw*, 2000; *Hundecha et al.*, 2001; *Xiong et al.*, 2001]. A number of research papers in the past few years [*Minns and Hall*, 1996; *Khondker et al.*, 1998; *Solomatine et al.*, 2000; *Sudheer and Jain*, 2004] have shown that using these data-driven techniques to model hydrologic processes, such as rainfall-runoff forecasting, flash flood forecasting and prediction of surge water levels, are promising.

[7] These intelligent computational methods offer real advantage over conventional modeling, including the ability to handle large amounts of noisy data from dynamic and nonlinear systems, especially when the underlying physical relationships are not fully understood. While both these techniques are proven to be effective when used on their own, the individual strengths of each approach can be exploited in a synergistic manner for the construction of powerful intelligent systems by effectively combining the techniques. In recent years, the integration of neural networks and fuzzy logic has given birth to a new research field called neurofuzzy systems. Neurofuzzy systems have the potential to capture the benefits of each individual field into a single framework. Neurofuzzy systems eliminate the basic problem in fuzzy system design (i.e., obtaining a set of fuzzy if-then rules) by effectively using the learning capability of an ANN for automatic fuzzy if-then rule generation. As a result, these systems can utilize linguistic information from a human expert as well as measured data during modeling. In contrast to pure neural or pure fuzzy methods, the neurofuzzy hybrid method possesses (1) the advantages of both the fields (2) the learning and adaptation capabilities of a neural network, and (3) the inference approach of a fuzzy reasoning mechanism that enables approximate human reasoning capabilities. Neurofuzzy applications have been developed for signal processing, automatic control, information retrieval, database management, etc. (e.g., *Jang*, 1993]. However, there is little discussion, in the literature, on more pragmatic hydrologic application of this hybrid computing system.

[8] This paper demonstrates the applicability of neurofuzzy systems in developing effective nonlinear models of the rainfall-runoff process for short-term flood forecasting. A neurofuzzy model, that forecasts hourly flood discharge at a given streamflow gauge station at different lead times, has been developed for the river Kolar (a tributary of Narmada) in India. The paper also aims at an extensive evaluation of neurofuzzy model with a pure ANN and a pure fuzzy rule based model for rainfall-runoff modeling, and critically comment on their relative merits and limitations.

2. Neurofuzzy Model

[9] The basic concepts that comprise the neural network approach or fuzzy theory, such as weights, learning algorithm, fuzzy set, membership functions, the domain partitions, and fuzzy if-then inference rules are not reproduced in the body of this paper as that have been introduced in numerous hydrological papers and text books [*Haykin*, 1994; *Hundecha et al.*, 2001; *Xiong et al.*, 2001]. However, as the integration of both techniques is a relatively new concept, brief details of the method are presented in the following sections.

[10] Neurofuzzy modeling refers to the way of applying various learning techniques developed in the neural network literature to fuzzy modeling or a fuzzy inference system (FIS). The basic structure of a FIS (Figure 1) consists of

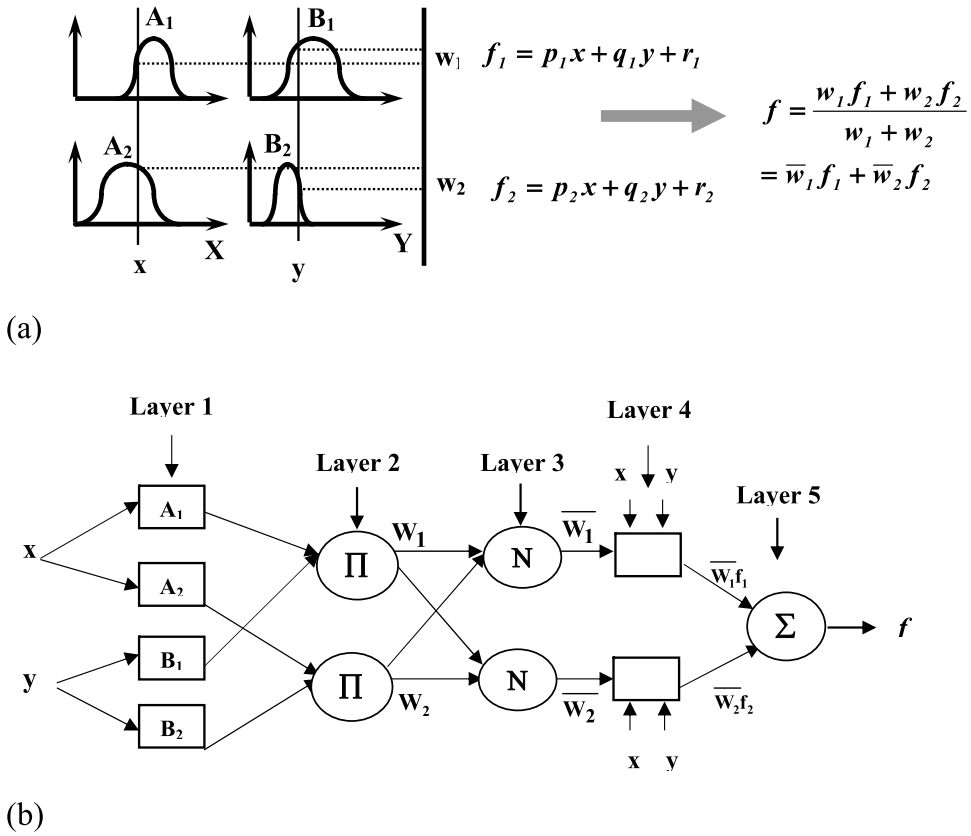

Figure 2. Schematic of fuzzy and neurofuzzy paradigm: (a) Fuzzy inference system and (b) equivalent ANFIS architecture.

three conceptual components: A rule base, which contains a selection of fuzzy rules; a database which defines the membership function (MF) used in the fuzzy rules; and a reasoning mechanism, which performs the inference procedure upon the rules and a given condition to derive a reasonable output conclusion. A FIS implements a nonlinear mapping from its input space to an output space. A FIS can utilize human expertise by storing its essential components in a rule base and database, and perform fuzzy reasoning to infer the overall output value. The derivation of if-then rules and corresponding membership functions depends heavily on the a priori knowledge about the system under consideration. However there is no systematic way to transform experience of knowledge of human experts to the knowledge base of a FIS. On the other hand, ANN learning mechanisms do not rely on human expertise. Because of the highly parallel structure of an ANN it is hard to extract structured knowledge from either the weights or the configuration of the ANN. The weights of the ANN represent the coefficients of the hyperplane that partition the input space into two regions with different output values. If one can visualize the hyperplane structure from the training data then the subsequent learning procedures in an ANN can be reduced. On the contrary, a priori knowledge is usually obtained from the human experts and it is most appropriate to express the knowledge as a set of fuzzy if-then rules.

[11] To a large extent, the drawbacks of these two approaches seem to be complementary. Therefore it is possible to consider building an integrated system combining the concept of FIS and ANN modeling. A common way to apply a learning algorithm to a FIS is to represent it in a special ANN architecture. However the conventional ANN learning algorithms (e.g., gradient descent) cannot be applied directly to such a system as the function of the inference system is usually non differentiable. This problem can be tackled by using differentiable functions in the inference system or by not using the standard neural learning algorithm. In the present study, a class of adaptive networks [*Brown and Harris*, 1994] that act as a fundamental framework for adaptive fuzzy inference systems is employed. The procedure of developing a FIS using the framework of adaptive neural networks is called an adaptive neurofuzzy inference system (ANFIS).

3. ANFIS

[12] The selection of a FIS is the major concern while designing an ANFIS to model a specific target system. Various types of FIS are reported in the literature [e.g., *Mamdani and Assilian*, 1975; *Tsukamoto*, 1979; *Takagi and Sugeno*, 1985] and each is characterized by their consequent parameters only. The current study used a Takagi-Sugeno-Kang (TSK) fuzzy inference system [*Takagi and Sugeno*, 1985] since the conclusion of a fuzzy rule is constituted by a weighted linear combination, and the parameters can be estimated by a simple least squares error method.

[13] Consider that the FIS has two inputs x, y and one output z. Figures 2a and 2b illustrate a TSK fuzzy inference system and its corresponding ANFIS architecture respec-

tively when each input (x and y) is assigned two membership functions. For a first-order TSK model, a common rule set with two fuzzy if-then rules can be written as follows: rule 1, if x is A_1 and y is B_1, then $f_1 = p_1x + q_1y + r_1$, and rule 2, if x is A_2 and y is B_2, then $f_2 = p_2x + q_2y + r_2$, where the "if" statement is the antecedent, the "then" statement is the consequent, x and y are linguistic variables and A_1, A_2, B_1, B_2 are corresponding fuzzy sets, and p_1, q_1, r_1 and p_2, q_2, r_2 are linear parameters. The functionality of each layer in the ANFIS is as follows [*Jang*, 1993].

3.1. Layer 1

[14] Each node in this layer generates membership grades of an input variable. The node output OP_i^1 is defined by

$$OP_i^1 = \mu_{Ai}(x) \quad \text{for} \quad i = 1, 2 \tag{1}$$

or

$$OP_i^1 = \mu_{Bi-2}(y) \quad \text{for} \quad i = 3, 4 \tag{2}$$

where x (or y) is the input to the node; A_i (or B_{i-2}) is a fuzzy set associated with this node, characterized by the shape of the MFs in this node and can be any appropriate functions that are continuous and piecewise differentiable such as Gaussian, generalized bell shaped, trapezoidal shaped and triangular shaped functions. Assuming a Gaussian function as the MF, the output OP_i^1 can be computed as

$$OP_i^1 = \mu_{Ai}(x) = e^{-\frac{1}{2}\left(\frac{x-c_i}{\sigma_i}\right)^2} \tag{3}$$

where $\{c_i, \sigma_i\}$ is the parameter set that changes the shapes of the membership function with maximum equal to 1 and minimum equal to 0. These parameters are called premise parameters or antecedent parameters.

3.2. Layer 2

[15] Every node in this layer multiplies the incoming signals, denoted as Π, and the output OP_i^2 that represents the firing strength of a rule is computed as,

$$OP_i^2 = w_i = \mu_{Ai}(x)\mu_{Bi}(y), i = 1, 2. \tag{4}$$

3.3. Layer 3

[16] The *i*th node of this layer, labeled as N, computes the normalized firing strengths as

$$OP_i^3 = \overline{w_i} = \frac{w_i}{w_1 + w_2}, i = 1, 2. \tag{5}$$

3.4. Layer 4

[17] Node *i* in this layer computes the contribution of the *i*th rule toward the model output, with the following node function:

$$OP_i^4 = \overline{w_i} f_i = \overline{w_i}(p_i x + q_i y + r_i) \tag{6}$$

where $\overline{w}$ is the output of layer 3 and $\{p_i, q_i, r_i\}$ is the parameter set. Parameters in this layer will be referred to as consequent parameters.

3.5. Layer 5

[18] The single node in this layer computes the overall output of the ANFIS as:

$$OP_1^5 = \text{Overall output} = \sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{7}$$

The parameters for optimization in an ANFIS are the premise parameters $\{c_i, \sigma_i\}$, which describe the shape of the MFs, and the consequent parameters $\{p_i, q_i, r_i\}$, which describe the overall output of the system. ANFIS makes use of a mixture of backpropagation (to learn the premise parameters) and least mean square estimation (to determine the consequent parameters). A step in the learning procedure has two parts: in the first part the input patterns are propagated forward assuming random initial values for the premise parameters, and the optimal consequent parameters are estimated by an iterative least mean square procedure, while the premise parameters are assumed to be fixed for the current cycle through the training set; in the second part the patterns are propagated back to modify the premise parameters using the backpropagation algorithm, while the consequent parameters remain fixed. This procedure is then iterated till a predefined error goal is reached.

4. Rainfall-Runoff Model Development

[19] Consider modeling a river flow time series, where it is required to forecast the value of flow (y_{t+i}) at time $t + i$, where i is the lead time. The inputs to the ANFIS are typically chosen as the values of the time series up to time t and the output will be the forecast value. It may be noted that if the aim of the model is prediction, the output vector must be mapped against available input information only. In other words, any model that forecasts a flow y_{t+i} at time $t + i$ can use input information up to time t only. In addition to previous values of the time series, one can utilize the values or forecasts of other time series (or external variables) as inputs that have a correlated or causal relationship with the series to be forecasted. For a river flow forecasting problem such exogenous time series could be the rainfall, evaporation and/or temperature over the basin. The functional form of this type of model is:

$$y_{t+i} = f\left(y_t, y_{t-1}, \ldots, y_{t-j}, z_t, z_{t-1}, \ldots, z_{t-k}\right)$$

where f is the unknown function mapped by the model, i is an index representing lead time, and j and k are the maximum number of time steps in the past considered important in modeling y_{t+i} and z_{t-k} is the exogenous variable considered as an input. The appropriate values of j and k are not known a priori, and are usually determined by a trial and error procedure [*Maier and Dandy*, 2000]. Instead, a data driven approach proposed by *Sudheer et al.* [2002] has been explored in the current study for general guidance in selecting the inputs. Separate models result

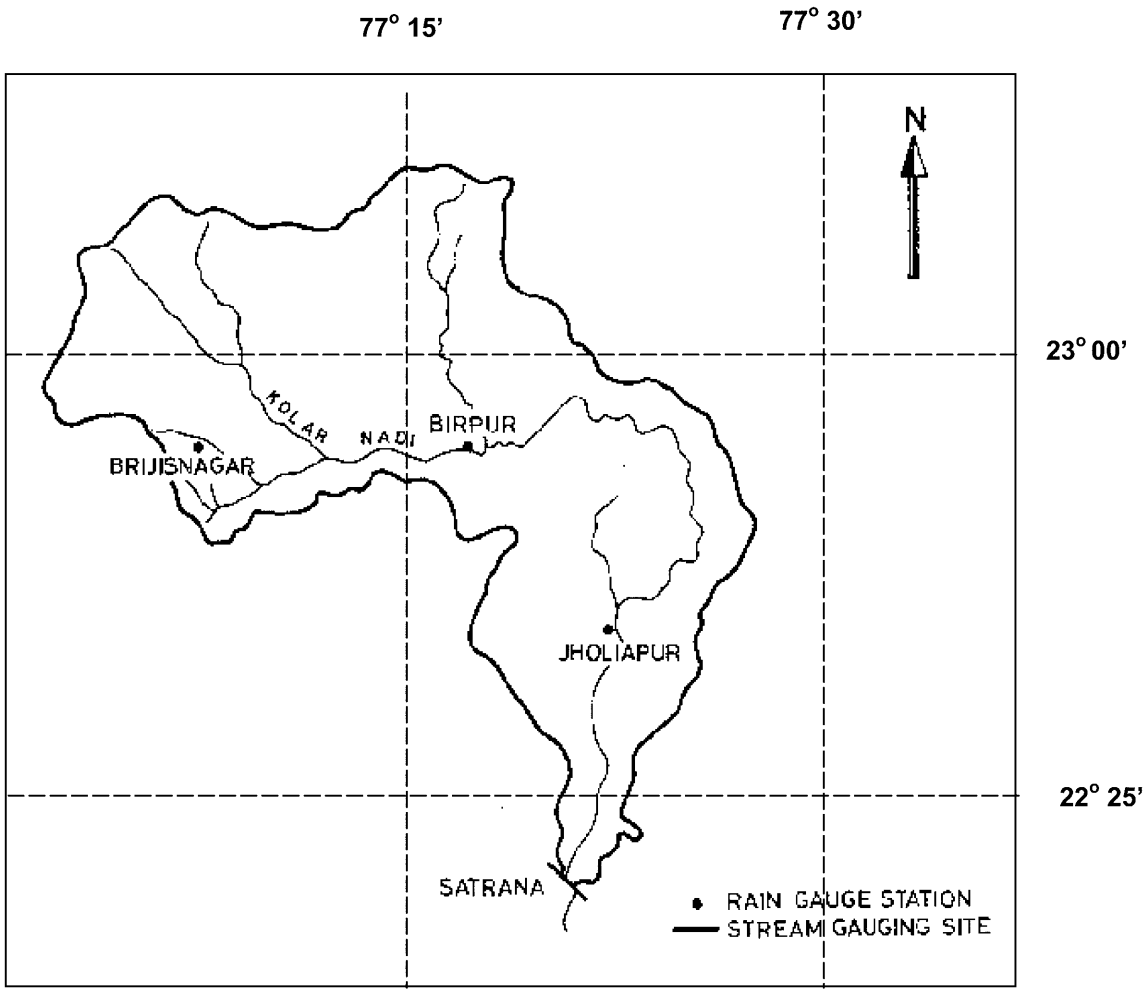

Figure 3. Map of the Kolar basin, India.

when these models are developed for different lead times i = 1, 2, 3, ., etc. A suitable transformation of the historic runoff series has been performed prior to model building as it aids in improving the model performance [*Sudheer et al.*, 2003]. In the current study, a logarithmic transformation is applied to the data. The deterministic component in the runoff and rainfall series is also removed and both the variables are scaled between zero and one, prior to developing each of the models.

[20] For the bulk of the study, rainfall and runoff data on an hourly interval for Kolar basin (Figure 3) in India during the monsoon season (July, August, and September) for three years (1987–1989) are used (Figure 4). The Kolar river is a tributary of the river Narmada that drains an area of about 1350 sq km before its confluence with Narmada near Neelkanth. In the present study the catchment area up to the Satrana gauging site is considered, which constitutes an area of 903.87 sq km. The 75.3 km long river course lies between north latitude 21°09′ to 23°17′ and east longitude 77°01′ to 77°29′. Topographically, the Kolar subbasin can be divided into two zones. The upper four fifths having elevations ranging from 300 to 600 m, is predominately covered by deciduous forests. Soils are skeletal to shallow except near canals where they are relatively deep. In this area the rocks are weathered and deep fissures are visible. The channel beds are rocky or graveled. General response of this upper part of the basin to rains is quick. The lower one fifth of the basin consists of a flat-bottomed valley narrowing toward the outlet and having elevations ranging from 300 to 350 m. The area is predominately cultivable and soils are deep and have flat slopes and as such response of this area to rainfall is likely to be quite slow. The rainfall data available were in the form of areal average values in the basin. The total available data has been divided into two sets, calibration set (data during the years 1987–1988) and validation set (data during the year 1989). Different models for lead times of up to 6 hours have been developed in the study. The parameters of the model are identified using the calibration data set, and the model is tested for its performance on the validation data set. The resulting hydrographs from the model is analyzed statistically using various evaluation measures.

4.1. Selection of Inputs to the Model

[21] One of the most important steps in the model development process is the determination of significant input variables. Usually, not all of the potential input variables will be equally informative since some may be correlated, noisy or have no significant relationship with the output variable being modeled [*Maier and Dandy*, 2000]. Generally, some degree of a priori knowledge is used to specify the initial set of candidate inputs [e.g., *Campolo et*

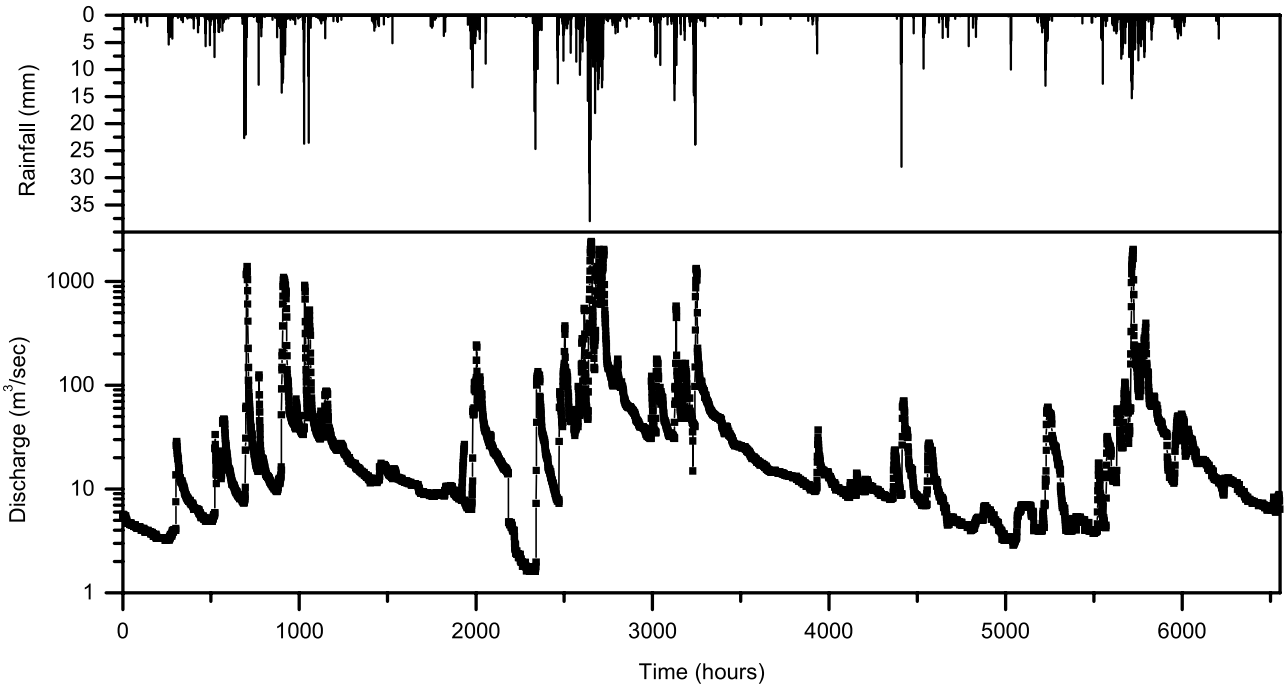

Figure 4. Rainfall and runoff series of the Kolar basin.

al., 1999; *Thirumalaiah and Deo*, 2000]. Although a priori identification is widely used in many applications and is necessary to define a candidate set of inputs, it is dependent on an expert's knowledge, and hence is very subjective and case-dependent. Intuitively, the preferred approach for determining appropriate inputs and lags of inputs, involves a combination of a priori knowledge and analytical approaches [*Maier and Dandy*, 1997]. When the relationship to be modeled is not well understood, then an analytical technique, such as cross correlation, is often employed [e.g., *Sajikumar and Thandaveswara*, 1999; *Luk et al.*, 2000; *Silverman and Dracup*, 2000; *Coulibaly et al.*, 2000, 2001; *Sudheer et al.*, 2002]. The major disadvantage associated with using cross correlation is that it is only able to detect linear dependence between two variables. Therefore cross correlation is unable to capture any nonlinear dependence that may exist between the inputs and the output, and may possibly result in the omission of important inputs that are related to the output in a nonlinear fashion. *Bowden et al.* [2004], while reviewing the current state of input selection procedures in water resources applications, report that the cross-correlation methods represent the most popular analytical techniques for selecting appropriate inputs. It follows that there is good scope for addressing this issue in future studies.

[22] The current study employed a statistical approach suggested by *Sudheer et al.* [2002] to identify the appropriate input vector. The method is based on the heuristic that the potential influencing variables corresponding to different time lags can be identified through statistical analysis of the data series that uses cross correlation, autocorrelation, and partial autocorrelation between the variables in question. The cross correlation between the runoff and rainfall series at various lags (Figure 5) showed significant correlation at 7, 8 and 9 hours of rainfall lag on the flow at any time. The autocorrelation and partial autocorrelation function (Figure 5) suggests a significant correlation, at 95% confidence level up to 1 hour of runoff lag. Note that this correlogram is for the stochastic component of the runoff series, i.e., after removing the deterministic components from the original series. Therefore a total number of 4 variables (3 rainfall and 1 runoff) are identified as inputs according to *Sudheer et al.* [2002]. As stated earlier, three kinds of models were developed in this study, namely, ANN, FIS and ANFIS, all of them built using the same input variables.

4.2. Model Structure Identification

[23] It may be noted that in theory, all these models (ANFIS, ANN, and FIS) in comparison are capable of fitting a nonlinear function to any arbitrary accuracy. Selection of model architecture is one decisive factor in the simulation and comparison. The identification of the optimal network geometry is one of the major tasks in developing an ANN model. While the number of input-output nodes is problem dependant, there is no direct and precise way of determining the optimal number of hidden nodes. The model architecture of an ANN is generally selected through a trial-and-error procedure [*Sudheer and Jain*, 2004], as currently there is no established methodology for selecting the appropriate network architecture prior to training [*Coulibaly et al.*, 2001]. On the other hand, a fuzzy inference system can be viewed as a partition in the multidimensional feature space, where the number of partitions in each dimension corresponds to the number of fuzzy sets and the corresponding membership function that are defined in that dimension. Consequently, the input space partitioning plays a major role in the optimal architecture of the model. Input space partitioning is carried out in different ways: grid and scattering partitioning. A clear drawback of grid partitioning is that the number of rules grows exponentially. Moreover, the optimization of antecedent param-

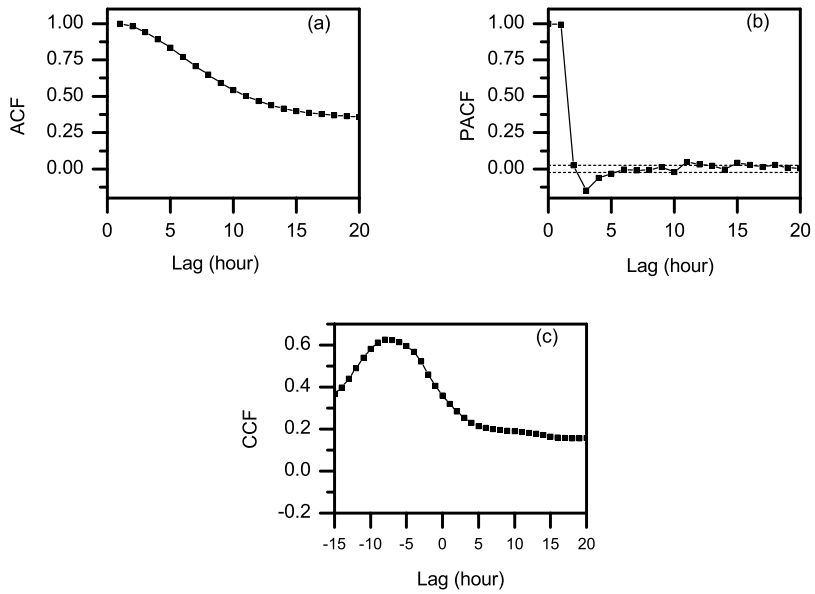

Figure 5. Correlogram plots of the data series: (a) Autocorrelation function (ACF), (b) partial autocorrelation function (PACF), and (c) cross-correlation function (CCF) between runoff at different lags of rainfall.

eter becomes complex if grid partitioning is employed. Consequently, while developing an FIS, scattering partitioning is commonly employed [*Setnes*, 2000] for FIS structure identification as the antecedent parameters are obtained directly from the fuzzy clusters. A limitation of this clustering based FIS models is that if any data point falls away from the cluster centre or outside the clusters the model performance may not be satisfactory. However, grid partitioning method represents the whole range of the input space, and if the complex optimization of antecedent parameters can be properly addressed, the resulting FIS may perform better than clustering based FIS. This is precisely one of the advantages of the ANFIS. In the current investigation, the optimization of antecedent parameters is addressed through the backpropagation algorithm. However, it is not advisable to use grid partitioning in ANFIS when the input dimension is more than six [*Nayak et al.*, 2004] due to curse of dimensionality.

[24] The trail and error procedure of determining the optimal number of hidden nodes in ANN requires huge amount of time as each time a new hidden node is added (or deleted) the model requires being retrained and evaluated. The major parameter that requires trial and error evaluation in FIS development is the cluster radius that specifies the range of influence of the cluster centre of each input and output dimension. As the subtractive clustering algorithm [*Chiu*, 1994] that is used in the current work is a non-iterative procedure, the time taken to reach the final FIS architecture by trail and error is less compared to ANN. In the case of ANFIS, the number of MF associated with each input variable is to be fixed by trail and error. A major drawback of the ANFIS is that the number of parameters grows exponentially with the number of MF resulting in large training time. Nonetheless, ANFIS converges faster than the typical feed forward ANN as the first three layers of the ANFIS are not fully connected (see Figure 2) and the backpropagation is limited to the layer 3 alone. The hybrid algorithm that is used for optimizing the parameters of the ANFIS in the current study, which is a combination of backpropagation and least squares error method, allows a significant reduction in the number of training epochs. The computational burden for developing these models with respect to the current study is discussed in section 6.

4.3. ANFIS Model Development

[25] The ANFIS rainfall-runoff model is developed following the procedure described in section 2. As the choice of membership function of the ANFIS architecture is not momentous to the model performance [*Nayak et al.*, 2004], Gaussian membership functions (equation (3)) are selected for the model. Note that in the grid partitioning method, ideally for n domains (rules) and p input variables there could be n^p different if-then rules. Consequently, increasing the number of membership functions on the input variables will increase the number of fuzzy if-then rules; simultaneously it increases the model complexity and hence affects the model parsimony. Moreover, a fuzzy model with a large number of rules often encounters the risk of over fitting the data [*Babuska*, 1998; *Yen and Wang*, 1999]. In the current study, the number of MFs assigned to each input variable has been varied from 2 (16 rules) to 4 (256 rules). No significant improvement in model performance is observed with respect to the change in number of MFs. However, as the number of MFs increases, the time taken for model training also increases considerably. Consequently, by the principle of parsimony, the model with 2 MFs was selected for further analysis.

4.4. ANN Model Development

[26] To have a true comparison between models, the input and the output vector to the ANN model are kept as the same as that of the ANFIS. Single hidden layer with sigmoid function nodes is used in the ANN. The sigmoid activation function is considered in the output layer also. As the sigmoid transfer function has been used in the model, the input-output data have been scaled appropriately to fall

within the function limits. A standard back propagation algorithm with adaptive learning rate and momentum factor has been employed to estimate the network parameters. The number of hidden neurons in the network, which is responsible for capturing the dynamic and complex relationship between various input and output variables was identified by various trials [*Eberhart and Dobbins*, 1990; *Maier and Dandy*, 2000]. The trial and error procedure started with two hidden neurons initially, and the number of hidden neurons was increased up to 10 with a step size of 1 in each trial. For each set of hidden neurons, the network was trained in batch mode to minimize the mean square error at the output layer. In order to check any overfitting during training, a cross validation was performed by keeping track of the efficiency of the fitted model. The training was stopped when there was no significant improvement in the efficiency, and the model was then tested for its generalization properties. The parsimonious structure that resulted in minimum error and maximum efficiency during training as well as testing was selected as the final form of the ANN model. The final structure of the ANN model is: 4 input neurons, 3 hidden neurons and 1 output neuron.

4.5. FIS Model Development

[27] A Takagi-Sugeno fuzzy inference system has been developed using the subtractive clustering [*Chiu*, 1994] algorithm integrated with a linear least squares estimate algorithm for the rainfall-runoff model. The FIS model has been developed based on the assumption that the cluster estimation method when applied to a collection of input and output data produces cluster centers where, each cluster center is in essence a prototypical data point that represents a characteristic behavior of the system. Hence each cluster center can be used as the basis of a rule that illustrates the system behavior.

[28] Consider a set of c cluster centers $\{x_1^*, x_2^*, \ldots\ldots, x_c^*\}$ in an M dimensional space. Let the first N dimensions corresponds to input variables and the last M-N dimensions correspond to output variables. Each vector x_i^* may be decomposed into two component vectors y_i^* and z_i^*, where y_i^* contains the first N elements of x_i^* (i.e., the coordinates of the cluster center in input space) and z_i^* contains the last M-N elements (i.e., the coordinates of the cluster center in output space).

[29] Each cluster center x_i^* may be considered as a fuzzy rule that describes the system behavior. Given an input vector y, the degree to which rule i is fulfilled is defined as

$$\mu_i = e^{-\alpha\left\|y-y_i^*\right\|^2} \tag{8}$$

where α is a function of cluster radius (i.e., a constant for a specific FIS). The output vector z may be computed as

$$z = \frac{\sum_{i=1}^{c} \mu_i z_i^*}{\sum_{i-1}^{c} \mu_i} \tag{9}$$

Equations (8) and (9) provide a simple and direct way to translate a set of cluster center into a FIS. This computational scheme may be viewed in terms of a FIS using traditional fuzzy if-then rules. Each rule has the following form: If y_1 is A_1 and y_2 is A_2 and ... then z_1 is B_1 and z_2 is B_2..., where y_i is the ith input variable and z_j is the jth output variable; A_i is a fuzzy set defined by an exponential membership function and B_j is a singleton. This computational scheme is equivalent to an inference method that uses multiplication as the AND operator, weights the output of each rule by the rule's firing strength and the overall output is determined as the weighted average of each rule output. For more details on the FIS model development, readers are referred to *Nayak et al.* [2005].

[30] In this case also, the input-output vector has been fixed as the same as that of ANFIS and ANN models. The major parameter that needs to be identified in FIS model is the clustering radius. Please note that the radius specifies the range of influence of the cluster centre of the each input and output dimension. Assuming that the cluster radius falls within the hyper box of unit dimension, a smaller cluster radius will yield more cluster in a data and hence a greater number of rules. Simultaneously it increases the model complexity and decreases parsimony. The clustering radius is identified through a trial-and-error procedure by varying the clustering radius from 0.05 to 0.5 with an increment of 0.05. For each radius, the variance explained by the FIS model was computed. The explained variance varied from 25.77% for a radius of 0.05 to 94.63% for a radius of 0.2. With further increase in radius the explained variance was found to deteriorate (93.23% corresponding to a radius of 0.3 and 92.34% corresponding to a radius of 0.5). The model that explained maximum variance was selected for further analysis. The selected FIS model had 3 clusters with a cluster radius of 0.2 (implying 3 rules).

5. Model Evaluation

[31] Quantitative assessments of the degree to which the model simulations match the observations are used to provide an evaluation of the model's predictive abilities. Frequently, evaluations of model performance utilize a number of statistics and techniques, usually referred to as "goodness of fit" statistics. Many of the principal measurements that are used in the hydrological literature have been critically reviewed by *Legates and McCabe* [1999]. Still, there is diversity in the use of global goodness of fit statistics to determine how well models forecast flood hydrographs. As a single evaluation measure is not available [*Sudheer and Jain*, 2003], a multicriteria assessment was performed in the current study with various goodness of fit statistics. These measures can be grouped into two types: relative and absolute. Relative goodness of fit measures are nondimensional indices, which provide a relative comparison of the performance of one model against another. In contrast, absolute goodness of fit statistics are measured in the units of the flow measurement. The criteria that are employed are the root-mean-square error (RMSE) between the observed and forecasted values, the coefficient of efficiency [*Nash and Sutcliffe*, 1970], the standard error of estimate (SEE), the noise to signal ratio and the mean absolute error (MAE). The definition of these evaluation criteria is provided in Table 1.

6. Results and Discussions

6.1. Forecasts at 1 Hour Lead Time

[32] The values of the performance indices for the 1-hour ahead forecast of all the models are presented in Table 2.

Table 1. Performance Evaluation Criteria[a]

Evaluation Criteria	Definition
Coefficient of correlation (R)	$R = \left[\frac{\sum_{i=1}^{n}(y_i^o - \overline{y^o})(y_i^c - \overline{y^c})}{\sqrt{\sum_{i=1}^{n}(y_i^o - \overline{y^o})^2}\sqrt{\sum_{i=1}^{n}(y_i^c - \overline{y^c})^2}}\right]$
Coefficient of efficiency (E)	$E = 1 - \frac{\sum_{i=1}^{n}(y_i^o - y_i^c)^2}{\sum_{i=1}^{n}(y_i^o - \overline{y^o})^2}$
Root-mean-square error (RMSE)	$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i^o - y_i^c)^2}{n}}$
Standard error of estimate (SEE)	$SEE = \sqrt{\frac{\sum_{i=1}^{n}(y_i^o - y_i^c)^2}{\nu}}$
Mean bias error (MBE)	$MBE = \frac{1}{n}\sum_{i=1}^{n}(y_i^c - y_i^0)$
Noise to signal ratio	$NS = \frac{SEE}{\sigma_y}$

[a]Note y_i^o and y_i^c are the observed and computed flow values at time t, respectively, $\overline{y^o}$ and $\overline{y^c}$ are the mean of the observed and computed flow values corresponding to n patterns, ν is the number of degrees of freedom, and σ_y is the standard deviation of the observed flow. Normalized values of these statistics are obtained by dividing the value by the observed mean.

The correlation statistics, which evaluate the linear correlation between the observed and computed runoff, is consistent for all models during calibration as well as the validation period. The model efficiency that evaluates the capability of the model in predicting runoff values away from the mean is found to be more than 93% during the calibration and validation periods for all models, which according to *Shamseldin* [1997], is very satisfactory. The RMSE statistic, which indicates a quantitative measure of the model error in units of the variable, was good for all models as is evidenced by the low values. Overall, for a 1-hour ahead forecast, the performance of all the models was comparable. However, it is worth mentioning that the ANFIS model has the lowest noise to signal ratio and SEE.

6.2. Forecasting at Longer Lead Times (>1 Hour)

[33] The variation of RMSE with different lead times is presented in Figure 6. As was expected, all the models show

Table 2. Performance Statistics of All Models at 1-Hour Lead Forecasting

	Calibration			Validation		
Performance Index	ANFIS	ANN	FIS	ANFIS	ANN	FIS
R	0.9812	0.9775	0.98	0.9824	0.9687	0.9837
E	0.9628	0.9554	0.9591	0.9651	0.935	0.9661
RMSE	40.13	43.9	42.06	24.64	33.65	24.31
Normalized RMSE	0.62	0.68	0.65	0.77	1.05	0.76
SEE	40.21	43.94	42.1	24.73	33.71	24.36
Normalized SEE	0.623	0.681	0.651	0.749	1.053	0.758
MBE	−1.18	−1.39	−1.83	0.04	−0.6	−0.49
Normalized MBE	−0.02	−0.02	−0.03	0	−0.02	−0.02
NS	0.312	0.366	0.337	0.121	0.167	0.124
MAE	7.21	7.69	7.4	3.44	5.18	3.33

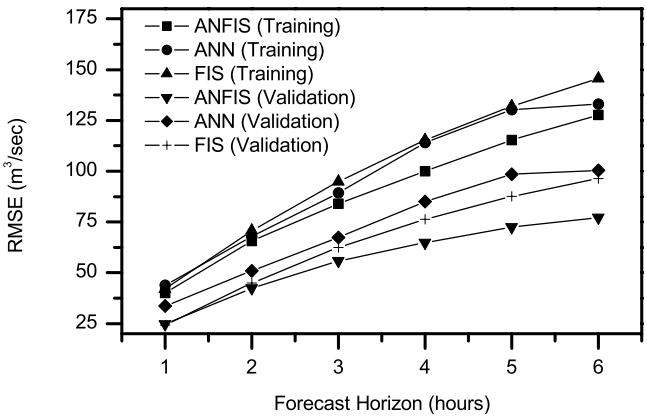

Figure 6. Variation of RMSE along the forecast time horizon.

an increase in RMSE with lead time. However, it is evident that the slope of the RMSE vs. prediction time horizon is the minimum for the ANFIS model during calibration as well as validation. While the ANFIS model forecasted the flows with a RMSE of 77.52 m^3/s at 6 hours, the ANN and FIS models forecasted the flows with an RMSE of 100.38 and 96.48 m^3/s respectively.

[34] Figure 7 depicts the noise to signal ratio of all models at different lead times. It is evident that while all the models have comparable value of this performance index at 1 hour lead time, the performance of all the models was found to deteriorate at higher lead times. Note that the ANFIS model also has a minimum rate of change for this parameter. It is worth mentioning that the value of the noise to signal ratio did not exceed unity for the ANFIS model up to a 5 hour lead time, while the other models exceeded the limiting value after a 3 hour lead time.

[35] The variation of efficiency along the forecast horizon is presented in Figure 8, from which it is clear that the performance of ANFIS is far superior to the two other models at all lead times. It is to be noted that the efficiency statistic is consistent during training and validation for the ANFIS model, which indicates a good generalization property for the model.

[36] The performance of the models in terms of correlation between the forecasted and observed values of flows is

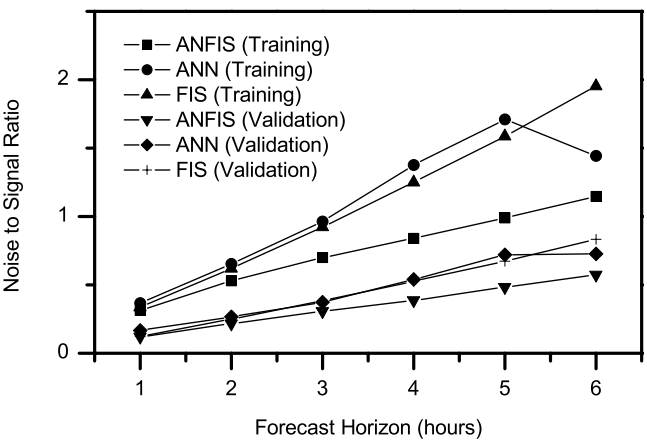

Figure 7. Variation of noise to signal ratio along the forecast time horizon.

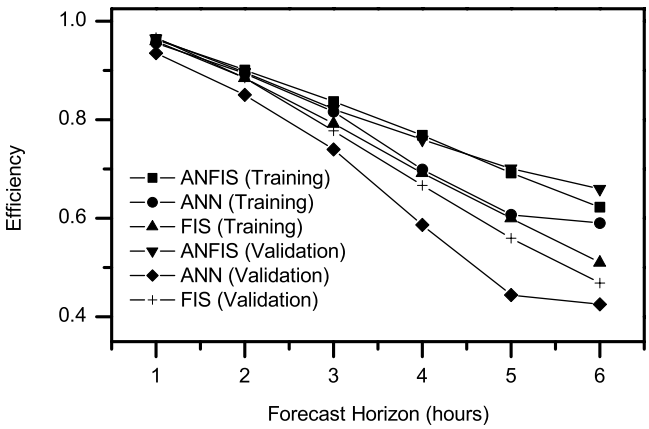

Figure 8. Variation of efficiency along the forecast time horizon.

similar to that in terms of the efficiency statistic (Figure 9). It is to be noted that even at a 6 hour lead time, the forecasted values have good correlation with the observed values in the case of the ANFIS model.

[37] An analysis to assess the potential of each of the models to preserve the statistical properties of the historic flow series reveals that the flow series computed by the ANFIS model reproduces the first three statistical moments (i.e., mean, standard deviation and skewness) better than that computed by the other two models. The values of the first three moments for the historic and computed flow series are presented in Table 3 for comparison. It is to be noted that the ANFIS model preserved the skewness at all lead times compared to the ANN and the fuzzy models. This observation is significant as the skewness of the data plays an important role in the overall model performance [*Sudheer et al.*, 2003].

[38] The above discussed error statistics provide relevant information on overall performance of the models, but do not provide specific information about model performance at high flow, which is of critical importance in a flood-forecasting context. Hence two additional storm-specific evaluation measures were also considered: percentage error in predicted peak flow and percent error in total runoff volume. Both these evaluation measures are computed as

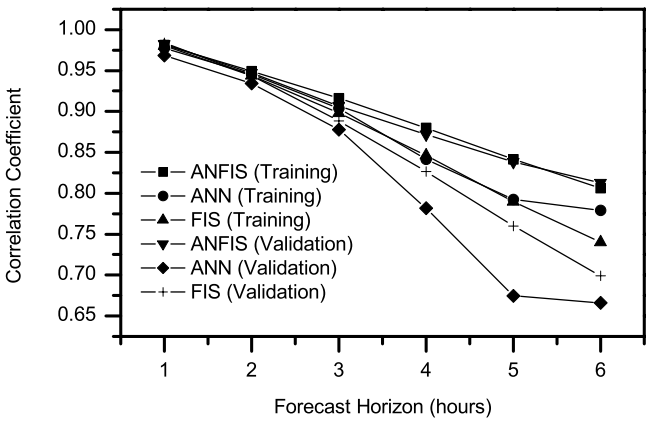

Figure 9. Variation of coefficient of correlation along the forecast time horizon.

Table 3. Summary Statistics of Forecasts up to 6 Hours by All the Models

	Calibration			Forecast Lead Time	Validation		
	Average	STD	Skew		Average	STD	Skew
Observed	64.69	207.98	6.80		32.04	132.01	11.74
ANFIS	63.51	203.99	7.04	1	32.08	128.87	11.76
ANFIS	61.63	195.93	7.37	2	31.96	123.78	11.55
ANFIS	59.31	182.93	7.42	3	31.95	120.08	11.33
ANFIS	56.92	168.79	7.43	4	32.02	119.49	11.32
ANFIS	54.27	150.77	7.11	5	31.76	116.79	11.35
ANFIS	51.80	135.18	7.04	6	31.25	111.72	11.49
ANN	63.30	201.98	7.03	1	31.45	120.19	11.47
ANN	61.33	192.27	7.26	2	30.48	103.75	10.46
ANN	59.05	181.38	7.73	3	29.99	92.91	9.63
ANN	55.53	157.58	7.86	4	29.02	82.95	9.15
ANN	52.82	137.02	6.66	5	28.42	76.35	8.64
ANN	52.13	138.14	7.45	6	29.20	92.26	10.14
FIS	62.86	196.80	6.70	1	31.55	124.73	11.51
FIS	60.19	180.21	6.54	2	30.82	114.09	11.16
FIS	57.28	162.36	6.35	3	29.94	102.92	10.72
FIS	54.44	145.25	6.14	4	29.05	92.39	10.24
FIS	51.85	130.14	5.93	5	28.14	83.20	9.79
FIS	49.41	116.11	5.70	6	27.23	74.66	9.30

the ratio of predicted and observed values expressed as a percentage. The results suggest that the value of the percent error in peak flow prediction, which is a useful index in simulating events such as floods, is within reasonable limits for the ANFIS model. The forecast error on a few typical peak flows during the period of the analysis is presented in Table 4 during the validation period. Note that these peak flows were observed at different periods of time, and do not correspond to the same flood event. It is observed that the ANFIS model was able to forecast most of the peaks with reasonable accuracy even up to 6 hours in advance, while

Table 4. Comparison of Error in Model Forecasted Peak Flows at Different Forecast Lead Times During Validation Period

Forecast Lead Time	Observed Peak Flow	Percent error in Forecasted Peak Flow		
		ANFIS	ANN	FIS
1 hour	61.11	9.31	6.37	5.32
1 hour	235.28	−2.23	5.18	−8.89
1 hour	393.67	−0.50	2.02	−7.97
1 hour	2028.98	−1.85	−9.10	−10.29
2 hours	61.11	13.87	21.45	10.57
2 hours	235.28	1.30	2.88	−17.81
2 hours	393.67	−10.10	−17.81	−17.45
2 hours	2028.98	−2.46	−12.35	−19.18
3 hours	61.11	10.76	66.31	14.71
3 hours	235.28	−22.48	−22.23	−26.87
3 hours	393.67	−24.22	−32.21	−27.27
3 hours	2028.98	−13.16	−30.91	−30.88
4 hours	61.11	6.59	8.31	19.26
4 hours	235.28	−29.20	−29.78	−37.46
4 hours	393.67	−19.51	−38.12	−46.38
4 hours	2028.98	−21.70	−44.72	−42.58
5 hours	61.11	9.52	19.36	17.35
5 hours	235.28	−17.02	−40.91	−38.98
5 hours	393.67	−24.00	−44.80	−46.69
5 hours	2028.98	−24.43	−56.98	−52.70
6 hours	61.11	5.94	9.12	16.27
6 hours	235.28	−18.21	−36.79	−40.54
6 hours	393.67	−21.16	−31.18	−48.82
6 hours	2028.98	−13.09	−34.48	−60.42

Table 5. Comparison of Model Estimated Hydrograph Characteristics of Two Typical Flood Events During Validation Period

		Event 1[a]			Event 2[b]		
Model	Forecast Lead Time, hours	Percent EV[c]	Estimated Peak	Time Difference to Peak Flow	Percent EV[c]	Estimated Peak	Time Difference to Peak Flow
ANFIS	1	2.59	2063.23	1	−4.99	420.05	1
ANFIS	2	6.32	2043.47	2	−4.57	431.88	2
ANFIS	3	9.34	1981.49	1	−3.33	435.06	2
ANFIS	4	10.00	1945.98	2	4.17	434.60	3
ANFIS	5	13.86	1934.56	1	14.27	414.57	4
ANFIS	6	18.74	1981.49	1	23.63	391.36	5
ANN	1	10.94	1935.56	1	−14.66	467.22	2
ANN	2	22.18	1663.95	2	−17.56	485.59	2
ANN	3	32.07	1334.21	3	−21.62	481.55	3
ANN	4	38.80	1235.63	4	−7.47	481.99	2
ANN	5	43.78	1128.85	5	14.29	427.14	5
ANN	6	49.23	923.90	6	41.31	338.76	6
FIS	1	5.36	1847.83	2	−1.26	393.27	1
FIS	2	13.30	1726.85	2	1.24	384.92	2
FIS	3	21.85	1532.22	3	6.90	371.51	3
FIS	4	30.11	1348.40	4	15.84	353.85	4
FIS	5	37.73	1187.78	5	30.06	331.73	5
FIS	6	44.95	1039.88	6	47.25	307.26	6

[a]Observed peak flow: 2028.98 m^3/s.
[b]Observed peak flow: 393.67 m^3/s.
[c]Percentage error in volume under estimated hydrograph.

for both the ANN and FIS the error increased drastically with an increase in prediction lead time. It is worth mentioning that the ANFIS was able to forecast the peak flows with minimum relative error, irrespective of the magnitude of the peak flow. However, for the ANN and the FIS, as the magnitude of peak flow increases the forecast error also increases, which gives less confidence in the developed models.

[39] Table 5, which depicts the relative error in the forecasted value of the total discharge from the basin for two typical hydrographs during the validation period, suggests that the ANFIS model was far superior to the other two models. However, the error is found to be increasing as the forecast horizon increases. It is evident from Table 5 that the FIS has failed to preserve the time to peak flow in both flood events. It can be seen that the ANFIS also failed to preserve the time to peak characteristics of a hydrograph, but note that the value of flow forecasted at the observed time of peak was with minimum error in the case of the ANFIS (Table 5). The results indicate that the ANFIS fails to identify the recession process accurately. One of the reasons for this appears to be related to the small number of training examples present in the subdomain represented by rules corresponding to a higher-flow range, from which a generalized nonlinear behavior of the process cannot be assessed. Note that the low- and medium-flow events are more influenced by local topography, soil moisture status, infiltration properties, and several other characteristics, often resulting in a complex rainfall-runoff relationship [*Zhang and Govindraju*, 2000]. During the high-flow events, rainfall dominates the discharge at the stream, and other factors tend to have a minor role in the collective response. In order to assess the transformation of rainfall into runoff in such cases, more example data are required. It may also be noted that the models used spatially averaged rainfall information to forecast the flows. The results also suggest that the global evaluation measures (e.g., RMSE, SEE, Efficiency etc.) are not good indicators for peak flow predictions. Shifting the focus to the peak prediction has also highlighted significant variation in the forecasting power of the three modeling approaches.

[40] It appears that while assessing the performance of a streamflow forecasting model for its applicability in forecasting streamflows at larger lead time, it is not only important to evaluate the average prediction error but also the distribution of prediction errors. It is important to know whether the model is predicting higher- or lower-magnitude flows poorly, which may help in further refining the model. The statistical performance evaluation criteria employed so far in this study are global statistics and do not provide any information on the distribution of errors. It is to be noted that the coefficient of efficiency can be high (80 or 90%) even for poor models, and the best models do not produce values which, on first examination, are impressively higher [*Garrick et al.*, 1978; *Legates and McCabe*, 1999]. The RMSE statistic indicates only the model's ability to predict a value away from mean [*Hsu et al.*, 1995]. Moreover, the models are trained by minimizing the sum squared error at the output layer that is similar to the RMSE. The correlation statistics provide information only on the strength of relationship between the observed and the computed values. Therefore, in order to test the effectiveness of the model developed, it is important to test the model using some other performance evaluation criteria such as average absolute relative error (AARE) and threshold statistics [*Jain et al.*, 2001; *Jain and Ormsbee*, 2002]. The AARE and threshold statistics (TS) not only give the performance index in terms of predicting flows but also the distribution of the prediction errors.

[41] These criteria can be computed as

$$AARE = \frac{1}{n}\sum_{i=1}^{n} |RE_t| \tag{10}$$

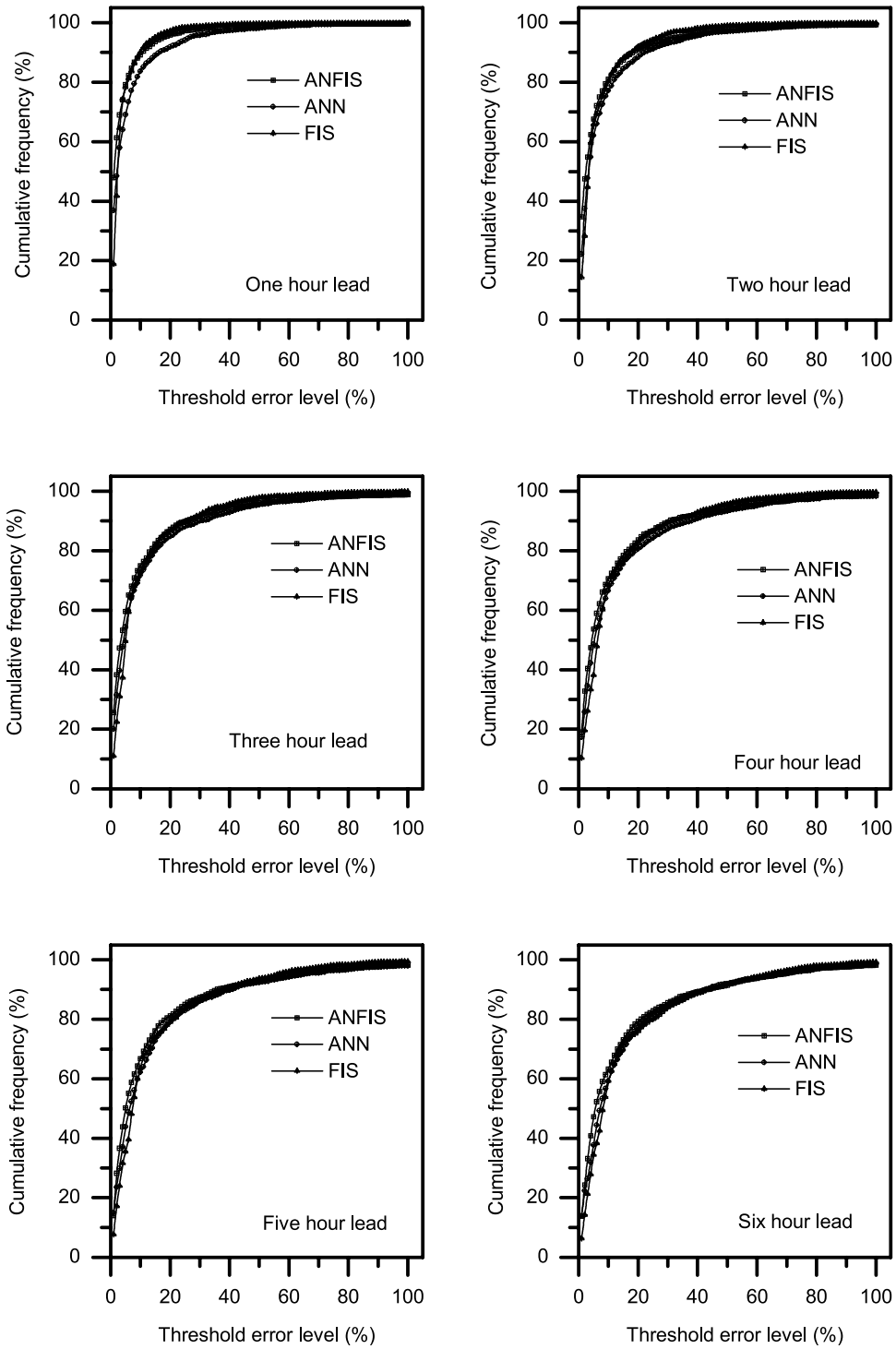

Figure 10. Distribution of forecast error across different error thresholds for all the models.

in which

$$RE_t = \frac{y_t^o - y_t^c}{y_t^o} * 100$$

where RE_t is the relative error in forecast at time t expressed as a percentage, y_t^o is the observed streamflow at time t, y_t^c is the computed streamflow at time t, and n is the total number of testing patterns. Clearly, the smaller the value of AARE the better is the performance.

[42] The threshold statistic for a level of x% is a measure of the consistency in forecasting errors from a particular model. The threshold statistics are represented as TS_x and expressed as a percentage. This criterion can be expressed for different levels of absolute relative error from the model. It is computed for x% level (TL) as

$$TS_x = \frac{Y_x}{n} * 100 \quad (11)$$

Table 6. Number of Data Points in Low-, Medium-, and High-Flow Categories (Validation Period)[a]

Category	Number of Points	Percentage of the Total Data
Low ($x < \mu$)	1783	81.97
Medium ($\mu \le x \le \mu + 2\sigma$)	369	16.97
High ($x > \mu + 2\sigma$)	23	1.06
Total	2175	100

[a]Note μ is the mean, and σ is the standard deviation.

where Y_x is the number of computed streamflows (out of n total computed) for which the absolute relative error is less than x% from the model.

[43] The distribution of errors is presented in Figure 10, which gives a clear indication of better performance by the ANFIS model. It can be observed from Figure 10 that the ANFIS and FIS perform almost similarly in terms of the distribution of the errors. It is worth noting that the ANFIS model forecasted 47.95% of the total number of flow values at 1 hour in advance with less than 1% relative error, while for ANN and FIS the corresponding values were 36.96% and 18.89% respectively. However, it is apparent that an evaluation based on relative error over the entire range of flow may cause significant weighting bias to the low-flow values. Hence, in order to assess the strengths and weaknesses of different models in predicting different magnitudes of flows, a "partitioning analysis" [*Jain and Srinivasulu*, 2004] is carried out by dividing the total flow range into low-, medium-, and high-magnitude flows, and then separately computing the AARE statistic on each of these clusters. The partitioning of the data was based on the relative spread of the flows from the mean, and a careful examination of the distribution of flows in terms of different statistical measures (Table 3). The criteria for partitioning and the data counts in each range are presented in Table 6. It is evident from the Table 6 that any evaluation based on the whole range of data will be biased toward low-flow values as about 82% of the data points fall in this range.

Table 7. Threshold and AARE Statistics for Different Models During Validation Period

	1-Hour Lead, %			3-Hour Lead, %			6-Hour Lead, %		
	ANFIS	ANN	FIS	ANFIS	ANN	FIS	ANFIS	ANN	FIS
				Low Flow					
AARE	0.07	0.07	0.07	0.08	0.08	0.08	0.12	0.12	0.13
TS1	54.68	37.27	19.45	30.05	29.57	11.94	15.60	17.75	7.59
TS5	85.19	84.63	84.20	66.10	65.15	55.68	53.54	49.94	39.38
TS10	93.21	92.51	93.03	81.44	81.10	79.79	70.70	68.74	66.39
TS15	96.13	95.72	96.40	87.72	87.46	88.40	80.54	78.25	79.34
TS20	97.59	97.52	98.31	92.25	92.11	92.08	85.38	82.54	82.46
				Medium Flow					
AARE	0.08	0.07	0.06	0.40	0.38	0.39	0.80	0.79	0.66
TS1	18.16	17.60	17.47	6.58	7.14	7.77	6.80	6.42	1.91
TS5	53.39	59.73	55.11	32.63	30.69	26.57	22.17	25.43	17.46
TS10	74.25	76.80	79.84	48.68	51.32	46.62	34.51	37.78	34.45
TS15	84.01	88.27	90.86	60.53	61.64	57.64	44.33	48.64	43.78
TS20	90.24	93.07	94.35	67.89	69.31	68.17	55.92	57.53	54.55
				High Flow					
AARE	0.24	0.29	0.22	0.72	0.66	0.58	1.21	0.91	0.64
TS1	4.35	0.00	0.00	0.00	0.00	0.00	3.85	0.00	0.00
TS5	26.09	16.67	4.17	4.55	3.45	0.00	7.69	3.23	2.33
TS10	26.09	23.33	29.17	9.09	10.34	3.57	7.69	6.45	18.60
TS15	52.17	33.33	54.17	22.73	10.34	21.43	19.23	6.45	27.91
TS20	56.52	41.67	58.33	22.73	24.14	25.00	26.92	12.90	32.56

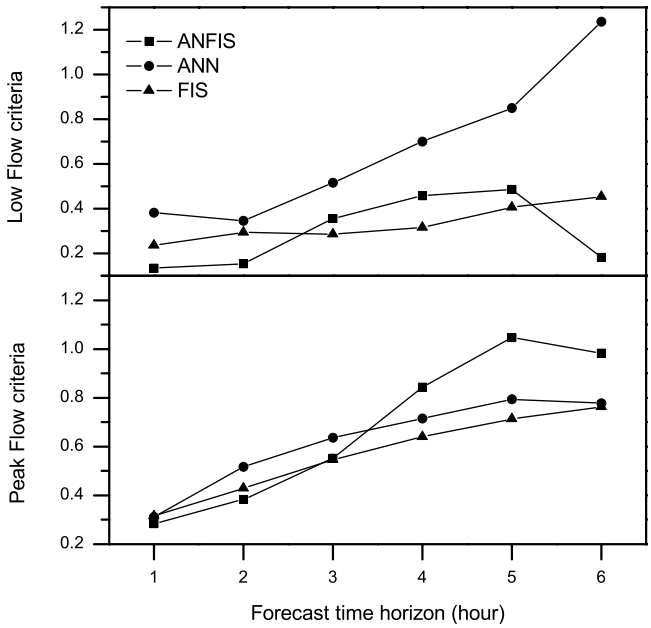

Figure 11. Peak and low-flow criteria for ANFIS, ANN, and FIS model during validation.

[44] The threshold and AARE statistics for different ranges of flow computed from all the three models are presented in Table 7 for comparison. It can be observed from Table 7 that in low-flow range all the three models show comparable performance in terms of AARE, while in case of medium and high ranges of flow, FIS model is found to be efficient than the other two. However, the distribution of error is seen to be better for the ANFIS model in low-flow range as 54.68% of the low-flow counts are predicted by ANFIS within 1% error compared to 19.45% for FIS. Even though only ANFIS is able to predict some high-flow ranges within 1% error, the results in Table 7 do not give any conclusive evidence that any specific model is superior to the other, as the values of AARE are comparable for all models.

[45] As the forecast accuracy is the main concern in river flow forecasting context, two more event specific evaluation criteria, namely peak flow and low-flow criteria (PFC and LFC), have been employed further. These criteria provide a more accurate measure of the model performance on low- and high-flow ranges. They can be computed as follows:

$$PFC = \frac{\left(\sum_{t=1}^{T_p} \left(Q_t^o - Q_t^c\right)^2 Q_t^{o^2}\right)^{1/4}}{\left(\sum_{t=1}^{T_p} Q_t^{o^2}\right)^{1/2}} \quad (12)$$

$$LFC = \frac{\left(\sum_{t=1}^{T_l} \left(Q_t^o - Q_t^c\right)^2 Q_t^{o^2}\right)^{1/4}}{\left(\sum_{t=1}^{T_p} Q_t^{o^2}\right)^{1/2}} \quad (13)$$

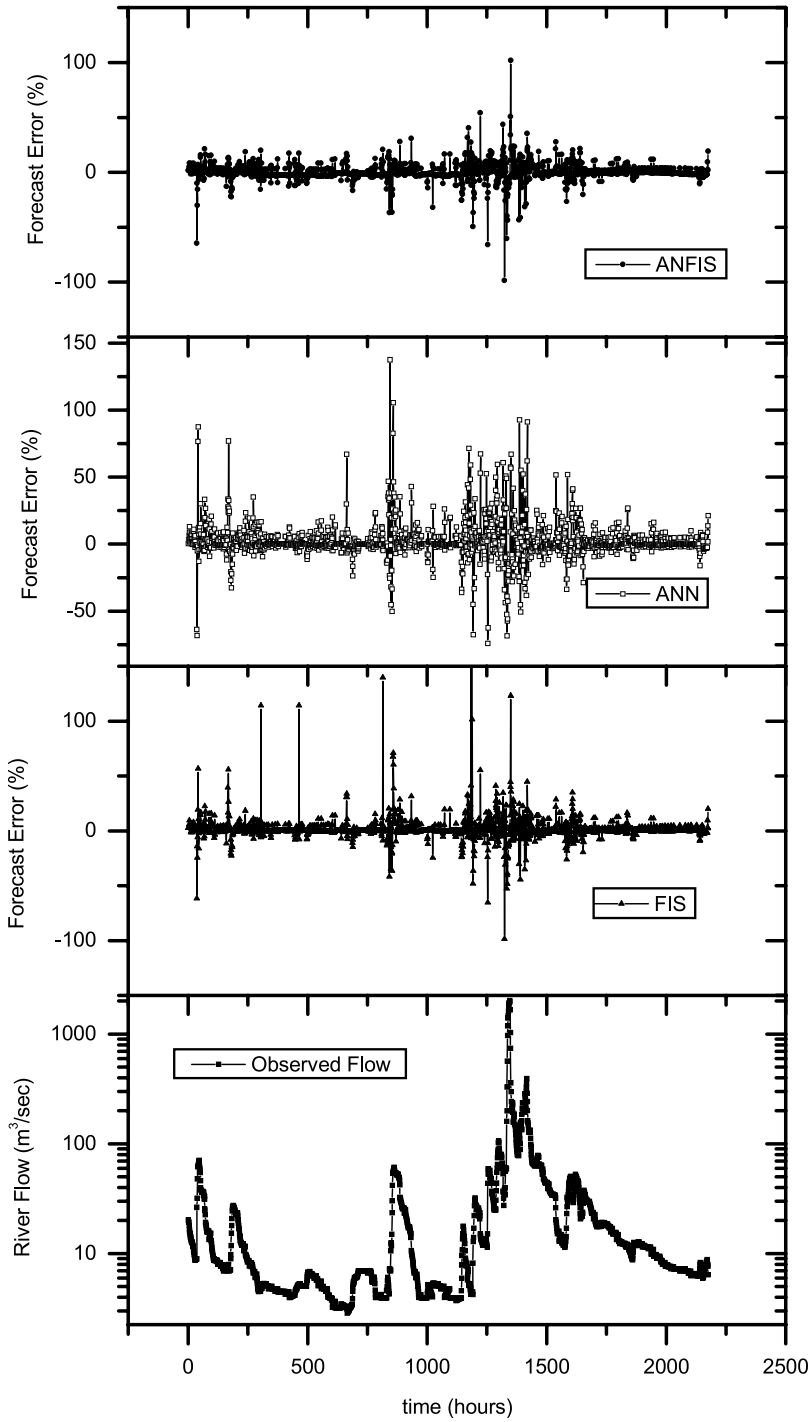

Figure 12. Distribution of forecast error across the full range of a typical flood hydrograph (1-hour lead forecast).

in which T_p is the number of peak flows greater than one third of the mean peak flow observed; T_l is the number of low flows lower than the one third of the mean low-flow observed; Q_t^o, Q_t^c are observed and computed flows for the time period t respectively.

[46] The PFC and LFC for different models along the forecast horizon are presented in Figure 11. It is observed from Figure 11 that the ANFIS performs better at lower forecast horizons. As the forecast horizon increases the FIS outperforms the other two. It is evident that the ANN model performance is not comparable to the other two in both the cases.

[47] The distribution of forecast errors by all models for a typical flood event is presented in Figure 12 from which it is evident that the ANFIS performs better than the other two models. It can be observed from Figure 12 that the forecast errors of the ANFIS model is clustered around the rapid rising limb near the peak flow, while for both the ANN and the FIS no specific clustering is observed. The reason for such a clustering by the ANFIS can be attributed to the

underestimation of the gradient of the rising limb and its rate of variation. Since only one antecedent runoff event is provided as input to the model, it appears that an inadequate representation of the state of saturation of the basin is given to the model. *Campolo et al.* [1999] point out that information about the changing state of saturation of a basin is vital in order to correctly predict the flow.

[48] The foregoing discussions clearly illustrate that the ANFIS model performs better than the ANN and FIS models in modeling the rainfall-runoff process. The performance of these models was comparable at a 1-hour lead time, but only the ANFIS tends to preserve the performance at higher lead times compared to the others. Although the preliminary concepts of the FIS and the ANN were developed on a different basis, they are essentially rooted in the same concepts of data driven modeling. In other words, the FIS as well as the ANN approach is based on the concept of reconstruction of a single-variable series in a multidimensional phase-space to represent the underlying dynamics, and a local approximation method is used for making predictions. The basic concept of a FIS resides in the idea that the combined (or averaged) estimators may be able to exceed the limitation of a single estimator, which is followed in an ANN. This heuristic is essentially found to be true in the current study, as the FIS is found to perform better than the ANN in terms of most of the performance statistics. It is evident from Table 4 (and Table 5) that the ANN is not able to preserve the nonlinear dynamics at different ranges of flow; the percent error in forecasted flood hydrograph is relatively higher for all the flood events analyzed in the case of ANN. The FIS, on the other hand, specializes for low-, medium- and higher-flow events by considering the data in three subdomains. The performance of the ANFIS as illustrated in earlier discussions confirms that it is able to preserve the advantages of the FIS. In a conventional FIS, the efficiency of the model largely depends on the antecedent parameters of the MFs, which is usually derived by using clustering techniques. However, when the learning principles of the ANN are employed to derive the antecedent parameters of the FIS, the resulting ANFIS model with 16 subdomains is able to represent the nonlinear behavior of the data series better than a conventional FIS. This is also one of the reasons for the better performance of the ANFIS model at higher lead times.

[49] ANFIS appears to be one of the best tradeoff between neural and fuzzy systems providing smoothness (due to fuzzy clustering interpolation) and adaptability (due to neural network backpropagation). From the numerical experiments presented in this paper, it is observed that ANFIS takes less computational time in optimizing the parameters (40 s for the finally selected architecture in a normal Pentium IV processor, while it was 55 s and 48 s for ANN and FIS respectively), implying that the amount of time required for the trial and error evaluation to reach the final architecture will be considerably less for the ANFIS. In general, ANFIS can be easily implemented by any flexible neural network simulator, and hence it is attractive for developing application. One of the major advantages of ANFIS (and FIS) over ANN is the transparency. ANN models receive the major criticisms that they do not consider/explain the underlying physical processes in a watershed, resulting in them being labeled as black box models. On the contrary, knowledge in terms of fuzzy if-then rules can be extracted from ANFIS (and FIS). One can interpret the rules of the ANFIS to infer the system dynamics represented by the model. This suggests new and promising research areas for future studies.

7. Summary and Conclusions

[50] The paper addresses the problem of forecasting the river flow on the basis of rainfall and runoff data. The objective of the paper was twofold: one was to demonstrate the potential of the neurofuzzy computing paradigm in modeling the rainfall-runoff process; and second was to evaluate the relative merits and demerits of this paradigm with reference to already popular ANN and fuzzy modeling approaches. The study suggests that the ANFIS model is able to capture the inherent nonlinearity in the rainfall-runoff process better than the other two, and is able to forecast flows satisfactorily up to 6 hours in advance. A very close fit was obtained between computed and observed flows up to 1 hour in advance for all models, but only the ANFIS model tends to preserve this performance at higher lead times. A comparative analysis of prediction accuracy of these models in different ranges of flow indicates that the FIS is better than the ANN. The very short computer time required for a single forecast (a fraction of a second when using a normal Pentium processor) does not lead to any constraints in the use of the method for real time flood forecasting. The results of the study are highly encouraging and suggest that an adaptive neurofuzzy approach is viable for developing short-term forecasts of river flow series.

References

Amorocho, J., and A. Brandstetter (1971), A critique of current methods of hydrologic systems investigations, *Eos Trans. AGU*, *45*, 307–321.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000a), Artificial neural networks in hydrology I: Preliminary concepts, *J. Hydrol. Eng.*, *5*(2), 115–123.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000b), Artificial neural networks in hydrology II: Hydrologic applications, *J. Hydrol. Eng.*, *5*(2), 124–137.

Atiya, A. F., S. El-Shoura, S. I. Shaheen, and M. El-Sherif (1999), A comparison between neural-network forecasting techniques—Case study: River flow forecasting, *IEEE Trans. Neural Networks*, *10*(2), 402–409.

Babovic, V., and N. Keijzer (1999), Forecasting of river discharges in the presence of chaos and noise, in *Coping With Floods: Lessons From Recent Experiences*, edited by J. Marsalek, Springer, New York.

Babuska, R. (1998), *Fuzzy Modeling for Control*, Springer, New York.

Bowden, G. J., G. C. Dandy, and H. R. Maier (2004), Input determination for neural network models in water resources applications. Part 1—Background and methodology, *J. Hydrol.*, *301*(1–4), 75–92.

Box, G. E. P., and G. M. Jenkins (1976), *Time Series Analysis, Forecasting and Control*, Holden Day, Oakland, Calif.

Brown, M., and C. Harris (1994), *Neurofuzzy Adaptive Modelling and Control*, Prentice-Hall, Upper Saddle River, N. J.

Campolo, M., P. Andreussi, and A. Soldati (1999), River flood forecasting with neural network model, *Water Resour. Res.*, *35*(4), 1191–1197.

Chang, L.-C., and F.-J. Chang (2001), Intelligent control for modelling of real-time reservoir operation, *Hydrol. Processes*, *15*(9), 1621–1634.

Chiu, S. (1994), Fuzzy model identification based on cluster estimation, *J. Intell. Fuzzy Syst.*, *2*(3), 267–278.

Connor, J. T., R. D. Martin, and L. E. Atlas (1994), Recurrent neural networks and robust time series prediction, *IEEE Trans. Neural Networks*, *5*(2), 240–254.

Coulibaly, P., F. Anctil, and B. Bobee (2000), Daily reservoir inflow forecasting using artificial neural networks with stopped training approach, *J. Hydrol.*, *230*, 244–257.

Coulibaly, P., F. Anctil, and B. Bobee (2001), Multivariate reservoir inflow forecasting using temporal neural network, *J. Hydrol. Eng.*, *6*(5), 367–376.

Dawson, D. W., and R. Wilby (2001), Hydrological modeling using artificial neural networks, *Prog. Phys. Geogr.*, *25*(1), 80–108.

Duan, Q., S. Sorooshian, and V. K. Gupta (1992), Effective and efficient global optimization for conceptual rainfall runoff models, *Water Resour. Res.*, *28*(4), 1015–1031.

Eberhart, R. C., and R. W. Dobbins (1990), *Neural Network PC Tools: A Practical Guide*, Elsevier, New York.

Fujita, M., M.-L. Zhu, T. Nakoa, and C. Ishi (1992), An application of fuzzy set theory to runoff prediction, paper presented at Sixth IAHR International Symposium on Stochastic Hydraulics, Int. Assoc. for Hydraul. Res., Taipei, Taiwan.

Garrick, M., C. Cunnane, and J. E. Nash (1978), A criterion of efficiency for rainfall-runoff models, *J. Hydrol.*, *36*, 375–381.

Haykin, S. (1994), *Neural Networks: A Comprehensive Foundation*, MacMillan, New York.

Hornik, K., M. Stinchcombe, and H. White (1989), Multilayer feed forward networks are universal approximators, *Neural Networks*, *2*, 359–366.

Hsu, K., V. H. Gupta, and S. Sorroshian (1995), Artificial neural network modeling of the rainfall-runoff process, *Water Resour. Res.*, *31*(10), 2517–2530.

Hundecha, Y., A. Bardossy, and H.-W. Theisen (2001), Development of a fuzzy logic based rainfall-runoff model, *Hydrol. Sci. J.*, *46*(3), 363–377.

Ikeda, S., M. Ochiai, and Y. Sawaragi (1976), Sequential GMDH algorithm and its applications to river flow prediction, *IEEE Trans. Syst. Manage. Cybernetics*, *6*(7), 473–479.

Jacoby, S. L. S. (1966), A mathematical model for non-linear hydrologic systems, *J. Geophys. Res.*, *71*(20), 4811–4824.

Jain, A., and L. E. Ormsbee (2002), Evaluation of short-term water demand forecast modeling techniques: Conventional methods versus AI, *J. Am. Water Works Assoc.*, *94*(7), 64–72.

Jain, A., and S. Srinivasulu (2004), Development of effective and efficient rainfall-runoff models using integration of deterministic, real-coded genetic algorithms and artificial neural network techniques, *Water Resour. Res.*, *40*(4), W04302, doi:10.1029/2003WR002355.

Jain, A., A. K. Varshney, and U. C. Joshi (2001), Short-term water demand forecast modeling at IIT Kanpur using artificial neural networks, *Water Resour. Manage.*, *15*(5), 299–321.

Jang, J.-S. R. (1993), ANFIS: Adaptive network based fuzzy inference system, *IEEE Trans. Syst. Man Cybernetics*, *23*(3), 665–683.

Khondker, M., G. Wilson, and A. Klinting (1998), Application of neural networks in real time flash flood forecasting, in *Hydroinformatics '98: Proceedings of the Third International Conference on HydroInformatics*, edited by V. Babovic and C. L. Larsen, pp. 777–782, A. A. Balkema, Brookfield, Vt.

Legates, D. R., and G. J. McCabe, Jr. (1999), Evaluating the use of goodness-of-fit measures in hydrologic and hydroclimatic model validation, *Water Resour. Res.*, *35*(1), 233–241.

Luk, K. C., J. E. Ball, and A. Sharma (2000), A study of optimal model lag and spatial inputs to artificial neural network for rainfall forecasting, *J. Hydrol.*, *227*, 56–65.

Maier, H. R., and G. C. Dandy (1997), Determining inputs for neural network models of multivariate time series, *Microcomput. Civ. Eng.*, *12*, 353–368.

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 application, *Environ. Modell. Software*, *15*, 101–124.

Mamdani, E. H., and S. Assilian (1975), An experiment in linguistic synthesis with a fuzzy logic controller, *Int. J. Man Machine Stud.*, *7*(1), 1–13.

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

Nash, J. E., and J. V. Sutcliffe (1970), River flow forecasting through conceptual models: 1. A discussion of principles, *J. Hydrol.*, *10*, 282–290.

Nayak, P. C., K. P. Sudheer, D. M. Rangan, and K. S. Ramasastri (2004), A neuro-fuzzy computing technique for modeling hydrological time series, *J. Hydrol.*, *291*(1–2), 52–66.

Nayak, P. C., K. P. Sudheer, and K. S. Ramasastri (2005), Fuzzy computing based rainfall-runoff model for real time flood forecasting, *Hydrol. Processes*, doi:10.1002/hyp.5553, *19*, 955–968.

Sajikumar, N., and B. S. Thandaveswara (1999), A non-linear rainfall-runoff model using an artificial neural network, *J. Hydrol.*, *216*, 32–35.

Salas, J. D., and J. T. B. Obeysekera (1982), ARMA model identification of hydrologic time series, *Water Resour. Res.*, *18*(4), 1011–1021.

See, L., and S. Openshaw (2000), Applying soft computing approaches to river level forecasting, *Hydrol. Sci. J.*, *44*(5), 763–779.

Setnes, M. (2000), Supervised fuzzy clustering for rule extraction, *IEEE Trans. Fuzzy Syst.*, *8*(4), 416–424.

Shamseldin, A. Y. (1997), Application of a neural network technique to rainfall runoff modeling, *J. Hydrol.*, *199*, 272–294.

Shamseldin, A. Y., A. E. Nasr, and K. M. O'Connor (2002), Comparison of different forms of the multi-layer feed-forward neural network method used for river flow forecasting, *Hydrol. Earth Syst. Sci.*, *6*, 671–684.

Sharma, T. C. (1985), Stochastic characteristics of rainfall-runoff processes in Zambia, *Hydrol. Sci. J.*, *30*(4), 497–513.

Silverman, D., and J. A. Dracup (2000), Artificial neural networks and long-range precipitation in California, *J. Appl. Meteorol.*, *31*(1), 57–66.

Solomatine, D. P., C. J. Rojas, S. Velickov, and J. C. Wust (2000), Chaos theory in predicting surge water levels in the North Sea, in *Proceedings of the 4-th International Conference on Hydroinformatics, Iowa, USA*, pp. 1–8, A. A. Balkema, Brookfield, Vt.

Stuber, M., P. Gemmar, and M. Greving (2000), Machine supported development of fuzzy-flood forecast systems, in *Proceedings of European Conference on Advances in Flood Research, Potsdam*, *PIK Rep. 65*, edited by A. Bronstert, C. Bismuth, and L. Menzel, 504–515.

Sudheer, K. P. (2000), Modeling hydrological processes using neural computing technique, Ph.D. thesis, Indian Inst. of Technol., Delhi, India.

Sudheer, K. P., and A. Jain (2004), Explaining the internal behaviour of artificial neural network river flow models, *Hydrol. Processes*, *18*(4), 833–844.

Sudheer, K. P., and S. K. Jain (2003), Radial basis function neural networks for modeling stage discharge relationship, *J. Hydrol. Eng.*, *8*(3), 161–164.

Sudheer, K. P., A. K. Gosain, and K. S. Ramasastri (2002), A data-driven algorithm for constructing artificial neural network rainfall-runoff models, *Hydrol. Processes*, *16*, 1325–1330.

Sudheer, K. P., P. C. Nayak, and K. S. Ramasastri (2003), Improving peak flow estimates in artificial neural network river flow models, *Hydrol. Processes*, *17*(3), 677–686.

Sugeno, M., and T. Yasukawa (1993), A fuzzy-logic based approach to qualitative modeling, *IEEE Trans. Fuzzy Syst.*, *1*(1), 7–31.

Takagi, T., and M. Sugeno (1985), Fuzzy identification of systems and its application to modeling and control, *IEEE Trans. Syst. Man Cybernetics*, *15*(1), 116–132.

Thirumalaiah, K., and M. C. Deo (2000), Hydrological forecasting using neural networks, *J. Hydrol. Eng.*, *5*(2), 180–189.

Tsukamoto, Y. (1979), An approach to fuzzy reasoning method, in *Advances in Fuzzy Set Theory and Application*, edited by M. M. Gupta, R. K. Ragade, and R. R. Yager, pp. 137–149, North-Holland, New York.

Xiong, L. H., 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 system, *J. Hydrol.*, *245*(1–4), 196–217.

Yen, J., and L. Wang (1999), Constructing optimal fuzzy models using statistical information criteria, *J. Intelligent Fuzzy Syst.*, *7*, 185–201.

Zadeh, L. A. (1965), Fuzzy sets, *Inf. Control*, *8*(3), 338–353.

Zhang, B., and R. S. Govindraju (2000), Prediction of watershed runoff using Bayesian concepts and modular neural networks, *Water Resour. Res.*, *36*(3), 753–762.

Zhu, M.-L., and M. Fujita (1994), Comparison between fuzzy reasoning and neural network method to forecast runoff discharge, *J. Hydrosci. Hydraul. Eng.*, *12*(2), 131–141.

Zhu, M.-L., M. Fujita, N. Hashimoto, and M. Kudo (1994), Long lead time forecast of runoff using fuzzy reasoning method, *J. Jpn. Soc. Hydrol. Water Resour.*, *7*(2), 83–89.

P. C. Nayak and D. M. Rangan, National Institute of Hydrology, Deltaic Regional Centre, Siddartha Nagar, Kakinada, India 533 003. (nayakpc@yahoo.co.in; ranga_mohan@yahoo.co.in)

K. S. Ramasastri, National Institute of Hydrology, Roorkee, India 247 667. (ksr@nih.ernet.in)

K. P. Sudheer, Department of Civil Engineering, Indian Institute of Technology Madras, Chennai, India 600 036. (sudheer@iitm.ac.in)