

ANN and Fuzzy Logic Models for Simulating Event-Based Rainfall-Runoff

Gokmen Tayfur¹ and Vijay P. Singh, F.ASCE²

Abstract: This study presents the development of artificial neural network (ANN) and fuzzy logic (FL) models for predicting event-based rainfall runoff and tests these models against the kinematic wave approximation (KWA). A three-layer feed-forward ANN was developed using the sigmoid function and the backpropagation algorithm. The FL model was developed employing the triangular fuzzy membership functions for the input and output variables. The fuzzy rules were inferred from the measured data. The measured event based rainfall-runoff peak discharge data from laboratory flume and experimental plots were satisfactorily predicted by the ANN, FL, and KWA models. Similarly, all the three models satisfactorily simulated event-based rainfall-runoff hydrographs from experimental plots with comparable error measures. ANN and FL models also satisfactorily simulated a measured hydrograph from a small watershed 8.44 km² in area. The results provide insights into the adequacy of ANN and FL methods as well as their competitiveness against the KWA for simulating event-based rainfall-runoff processes.

DOI: 10.1061/(ASCE)0733-9429(2006)132:12(1321)

CE Database subject headings: Rainfall; Runoff; Neural networks; Fuzzy sets; Kinematic wave theory; Simulation.

Introduction

Fundamental to flood prediction, sediment transport, migration of contaminants, design of agricultural irrigation systems, solute uptake from soil, infiltration into soil, and water budgeting is the flow of water over land surfaces. The St. Venant equations and their simplified forms have long been employed to model flow over such surfaces. The models using these equations or their simplified versions, such as the diffusion wave approximation or the kinematic wave approximation (KWA), are generally designated as physically-based (PB) models. PB models permit calculations of temporal and spatial variations of flow variables, such as flow depth and velocity, over land surfaces. The knowledge of these variables is important in the study of erosion, sediment transport and solute transport processes. However, for these PB models to be effective, a detailed description of the overland flow plane at the grid scale of the numerical model to be used is needed. Such a detailed description is rarely available, primarily due to time and budgetary constraints. Therefore, for many practical purposes, numerical models of overland flow amount to an overkill due to the lack of data. Further, the numerical solution of these highly nonlinear flow equations is prone to problems of

instability and lack of convergence. Thus, simpler solution methodologies may be needed for prediction of discharge, especially for design purposes (Govindaraju et al. 1992).

This study presents the development of artificial neural network (ANN) and fuzzy logic (FL) models for prediction of event-based overland flow discharge due to rainfall. The ANN and FL methodologies have been commonly employed, especially in this last decade, for solution of a wide range of problems ranging from finance to environment to water resources to engineering. The ASCE Task Committee (2000b) summarized applications of ANN to solution of many hydrologic problems. Dawson and Wilby (1998) discussed the application of ANN to flow forecasting in two flood-prone catchments in England using hourly hydrometric data. They were able to construct ANN models to predict flow with 6 h lead times for Amber and Mole Rivers. Rajurkar et al. (2002) applied ANN for modeling daily flows during monsoon flood events for a large size catchment in India using daily rainfall data in the input vector of the network model. Tokar and Johnson (1999) developed ANN to forecast daily runoff as a function of daily precipitation, temperature, and snowmelt for a watershed in Maryland. Tokar and Markus (2000) applied ANN to predict monthly streamflow for the Fraser River Watershed in Colorado. ANN has also been applied to sediment transport (Tayfur 2002b); streamflow forecasting (Wu et al. 2005; Ancil and Rat 2005); rainfall forecasting (Ramírez et al. 2005; Olsson et al. 2004); seepage (Tayfur et al. 2005); and dispersion (Tayfur and Singh 2005).

Most of the previous studies employed only daily or monthly hydrometric data in the input vectors of neural networks to make long-term predictions for various watersheds and tested ANNs against the regression-based, simple conceptual black box, or stochastic models (ASCE Task Committee 2000b). For prediction of runoff, any model that does not, in addition to hydrometric data, consider the physical parameters that influence runoff and infiltration may not lead to adequate results. Further, not only long term, but also short term, predictions of runoff can be vital for flood forecasting. Realistic short term prediction requires event-

¹Professor, Dept. of Civil Engineering, Izmir Institute of Technology, Gulbahcekoyu, Urla, Izmir 35340, Turkey. E-mail: gokmentayfur@iyte.edu.tr

²Carolyn and William N. Lehrer Distinguished Chair in Water Engineering, Dept. of Biological and Agricultural Engineering, Texas A&M Univ., Scoates Hall, 2117 TAMU, College Station, TX 77843-2117. E-mail: vsingh@tamu.edu

Note. Discussion open until May 1, 2007. Separate discussions must be submitted for individual papers. To extend the closing date by one month, a written request must be filed with the ASCE Managing Editor. The manuscript for this paper was submitted for review and possible publication on October 19, 2004; approved on February 7, 2006. This paper is part of the *Journal of Hydraulic Engineering*, Vol. 132, No. 12, December 1, 2006. ©ASCE, ISSN 0733-9429/2006/12-1321-1330/\$25.00.

based rainfall-runoff modeling. In the event-based runoff hydrograph prediction, the time variable should also be considered in the input vector of the model. This study develops an ANN model to predict event-based runoff. The model considers not only the rainfall rate but also surface gradient, time, and infiltration rate in the input vector of the network. The model is trained and tested using measured data. Also, the model is tested against a PB model that is based on the KWA.

A number of investigators have also investigated the use of FL methodology in rainfall-runoff studies. Yu and Yang (2000) presented a fuzzy multiobjective function (FMOF) to improve the performance of conventional objective functions of root-mean-square error (RMSE) and mean percentage error (MPE) that are used in calibrating rainfall-runoff conceptual models. Using daily rainfall and flow discharge measurements as well as monthly evaporation estimates for calibrating and verifying the rainfall-runoff model, they showed that the FMOF led to improved simulation of a wide range of flow stages as it was capable of combining various objective functions with different acceptable levels. Ozelkan and Duckstein (2001) proposed a fuzzy conceptual rainfall-runoff framework to deal with parameter uncertainties of conceptual rainfall-runoff models. They concluded that the fuzzy logic framework enabled a decision-maker to gain insight into the model sensitivity and the uncertainty stemming from the elements of the conceptual rainfall-runoff model. See and Openshaw (2000) presented artificial intelligence methodology based forecasting models to provide a solution to a river level and flood prediction problem. They developed the models based on historical time series flow data from the River Ouse in England. The fuzzy logic approach has also been applied to flood forecasting (Chang et al. 2005); precipitation (Maskey et al. 2004); sediment transport (Tayfur et al. 2003), reservoir operation (Tilmant et al. 2002), and storm water infiltration (Hong et al. 2002).

It seems that there is no study on the application of FL in predicting event-based runoff hydrograph using rainfall, time, and infiltration in the input vector. Also, there seems to be no study on testing a FL algorithm against a PB model in simulating event-based rainfall-runoff. This study develops a FL model to simulate event-based runoff. The model employs rainfall rate, infiltration rate, and time in the input vector of the algorithm. The model is calibrated and tested using measured data and then it is compared with the KWA model.

The objective of this study is to develop ANN and FL models for the prediction of event-based discharge due to rainfall, evaluate the performance of the models using event-based rainfall-runoff experimental data, and compare the models with the KWA model in order to investigate the appropriateness of using nondeterministic ANN and FL models versus a deterministic KWA model for predicting runoff.

Experimental Data

In this study, 36 event-based data sets—24 laboratory flume data sets from Kilinc and Richardson (1973) and 12 experimental plot data sets from Barfield et al. (1983) (Table 1)—were employed to train and test the models to predict peak discharge from rainfall events. Also, employed were 6 event-based runoff hydrographs from Barfield et al. (1983) for training and testing the models to simulate the runoff hydrographs.

Kilinc and Richardson (1973) performed experimental runs by using a 1.5 m wide \times 4.6 m long flume with an adjustable slope. The flume was filled with compacted sandy soil and the major

Table 1. Experimental Data

Run number	Slope (%)	Rainfall rate (mm/h)	Peak runoff rate (mm/h)
Kilinc and Richardson (1973)			
1	5.7	32	19.1
2	5.7	57	49.1
3 ^a	5.7	93	87.3
4	5.7	117	111.6
5	10	32	21.6
6	10	57	49.8
7	10	93	88.3
8 ^a	10	117	113.4
9	15	32	22.7
10 ^b	15	57	50.7
11	15	93	89.2
12	15	117	115.3
13 ^a	20	32	20.9
14 ^a	20	57	50.7
15 ^a	20	93	89.4
16 ^a	20	117	116.2
17	30	32	25.6
18	30	57	51.0
19 ^a	30	93	91.0
20	30	117	116.4
21 ^a	40	32	25.1
22	40	57	51.2
23	40	93	91.4
24	40	117	116.6
Barfield et al. (1983)			
S1R2	9.3	62	51.0
S2R2 ^b	9.2	97	76.0
S3R2 ^a	8.6	78	57.0
S1S2	9.1	78	64.0
S2S2 ^a	9.2	78	64.0
S3S2	8.6	78	67.0
T1R2	9.1	78	64.0
T2R2	9.0	78	48.0
T3R2	9.3	78	48.0
T1V2	9.3	78	60.0
T2V2 ^a	9.2	79	61.0
T3V2	9.1	79	55.0

^aUsed for calibration of the physically based model.

^bUsed for testing the models.

control variables were rainfall intensity and soil surface slope. Six bare slopes of 5.7, 10, 15, 20, 30, and 40% were tested with four different rainfall intensities of 32, 57, 93, and 117 mm/h. On average, the constant infiltration rate for each run was about 5.43 mm/h. Table 1 summarizes the data of the 24 runs including peak discharge data from each run measured at the downstream end of the plot.

Barfield et al. (1983) ran experiments on several plots, 4.2 m wide \times 22 m long, constructed over a base of a shale spoil. 12 runs from Barfield et al. (1983) presented in Table 1 had, on average, a constant infiltration rate of 18.8 mm/h. Table 1 summarizes the data on these 12 runs including the peak discharge from each run measured at the downstream end of the plot (Barfield et al. 1983).

Of the total 36 runs presented in Table 1, 25 and 11 runs were

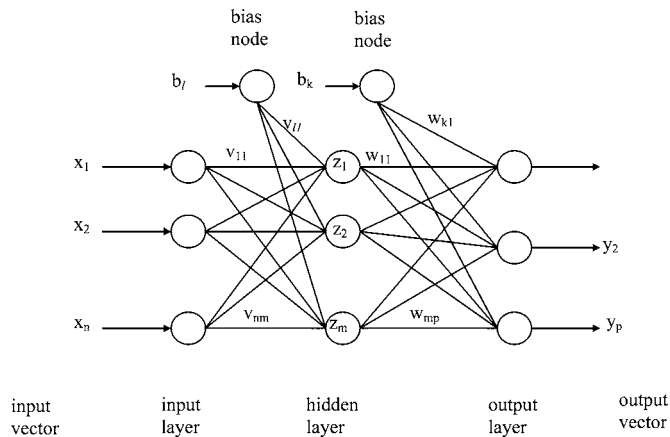


Fig. 1. Schematic representation of ANN architecture

used for training and testing of ANN and FL models, respectively. The data sets used for testing of ANN and FL models are marked by footnote “a” in Table 1 and they were also used for testing the KWA model. Data sets marked by footnote “b” in Table 1 were used for calibration of the KWA model. In addition, 6 runoff hydrographs of plots labeled as S1R2, S2R2, T1R2, S3S2, S3R2, and T3V2 (Barfield et al. 1983) were employed for runoff hydrograph simulation. Of the 6 runoff hydrographs, 4 (S1R2, S2R2, T1R2, and S3S2) and 2 (S3R2 and T3V2) were employed for training and testing ANN and FL models, respectively. The runoff hydrographs used for testing of ANN and FL models were also used for testing KWA model.

Rainfall-Runoff Models

Artificial Neural Network

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

$$E = \sum_P \sum_p (y_i - t_i)^2 \quad (1)$$

where y_i = component of a network output vector \mathbf{Y} ; t_i = component of a target output vector \mathbf{T} ; p = number of output neurons; and P = number of training patterns.

In the backpropagation algorithm, the optimal weights would generate an output vector $\mathbf{Y} = (y_1, y_2, \dots, y_p)$ as close as possible to the target values of the output vector $\mathbf{T} = (t_1, t_2, \dots, t_p)$ with a

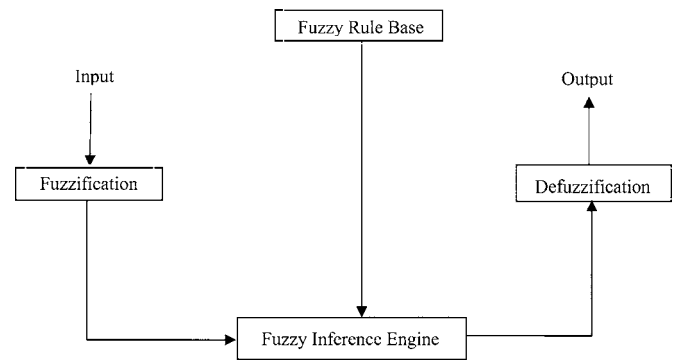


Fig. 2. Schematic representation of fuzzy system

selected accuracy. The back propagation algorithm employs the gradient-descent method, along with the chain rule of differentiation, to modify the network weights as (ASCE Task Committee 2000a):

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

where v_{ij} = weight from i th neuron in the previous layer to the j th neuron in the current layer and δ = learning rate.

The network learns by adjusting the biases and weights that link its neurons. Before training begins, a network's weights and biases are set equal to small random values. Also, due to the nature of the sigmoid function used in the backpropagation algorithm, all external input and output values before passing them into a neural network are standardized. Without standardization, large values input into an ANN would require extremely small weighting factors to be applied and this could cause a number of problems (Dawson and Wilby 1998). The details of ANNs are available in the literature (Somez 1998; ASCE Task Committee 2000a; Tayfur 2002b).

Fuzzy Logic

A general fuzzy system has basically four components—fuzzification, fuzzy rule base, fuzzy output engine, and defuzzification (Fig. 2). *Fuzzification* converts each piece of input data to degrees of membership by a look-up in one or more several membership functions. The key idea in fuzzy logic, in fact, is the allowance of partial belongings of any object to different subsets of the universal set instead of belonging to a single set completely. Partial belonging to a set can be described numerically by a membership function, which takes on values between 0 and 1 inclusive. Intuition, inference, rank ordering, angular fuzzy sets, neural networks, genetic algorithms, and inductive reasoning can be among many ways to assign membership values or functions to fuzzy variables. This intuitive approach is used rather commonly because it is simple and derived from the innate intelligence and understanding of human beings. Fuzzy membership functions may take on many forms, but in practical applications simple linear functions, such as triangular ones, are preferable.

The fuzzy rule base contains rules that include all possible fuzzy relations between inputs and outputs. These rules are expressed in the IF–THEN format. In the fuzzy approach, there are no mathematical equations and model parameters. All the uncertainties, nonlinear relationships, or model complications are included in the descriptive fuzzy inference procedure in the form of IF–THEN statements. There are basically two types of rule sys-

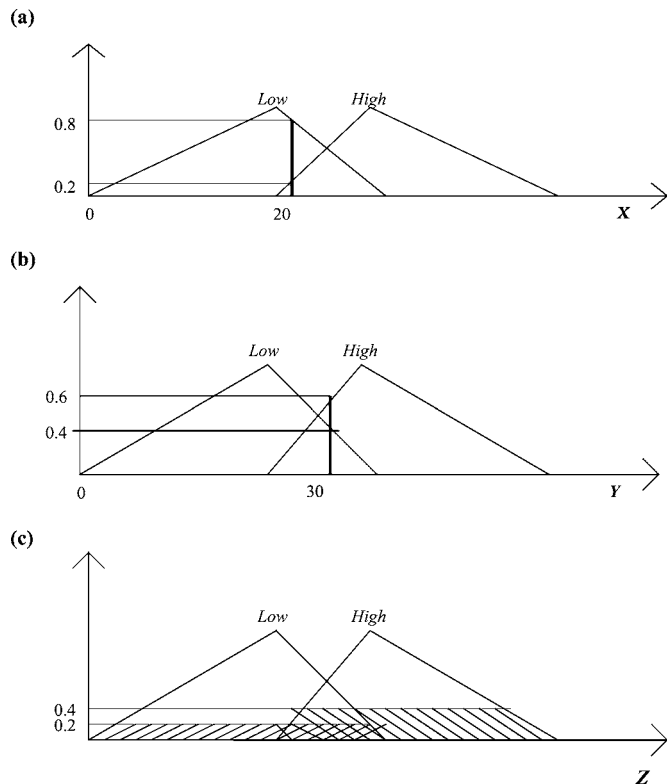


Fig. 3. Schematic representation of fuzzy inferencing: (a) $X=20$; (b) $Y=30$; and (c) fuzzy output sets for Z

tems, namely, Mamdani and Sugeno (Jantzen 1999). Depending upon a problem under consideration, a user can choose the appropriate rule system. According to the Sugeno rule system, the consequent part of the fuzzy rule is expressed as a mathematical function of the input variable and such a system is more appropriate for neurofuzzy systems (Sen 1998, 1999; Jantzen 1999). In the Mamdani rule system, however, the consequent part of the fuzzy rule is also expressed as verbally.

The fuzzy inference engine takes into account all the fuzzy rules in the fuzzy rule base and learns how to transform a set of inputs to corresponding outputs. To do so, it uses either **min** or **prod** activation operators. The activation of a rule is the deduction of the conclusion, possibly reduced by its firing strength. The **prod** activation (multiplication) scales the membership curves, thus preserving the initial shape, rather than clipping them as the **min** activation does. Jantzen (1999) points out that both methods, in general, work well. In order to illustrate the inferencing methodology, let us consider a simple case presented in Fig. 3, where there are two input variables X and Y [Figs. 3(a and b)] and one output variable Z [Fig. 3(c)]. For this simple system, let us also assume the following fuzzy rules:

IF X is low and Y is low **THEN** Z is high

IF X is high and Y is high **THEN** Z is low

Now, let us see how the inference engine would produce fuzzy outputs for a given input vector of X as 20 and Y as 30. As seen in Fig. 3(a), $X=20$ is part of 'low' and 'high' subsets with different degrees of membership of 0.8 and 0.2, respectively. Similarly, $Y=30$ is part of low and high subsets with 0.4 and 0.6 degrees of membership, respectively [Fig. 3(b)]. The fuzzy inference engine will consider the previous rules and by **min** activation find the

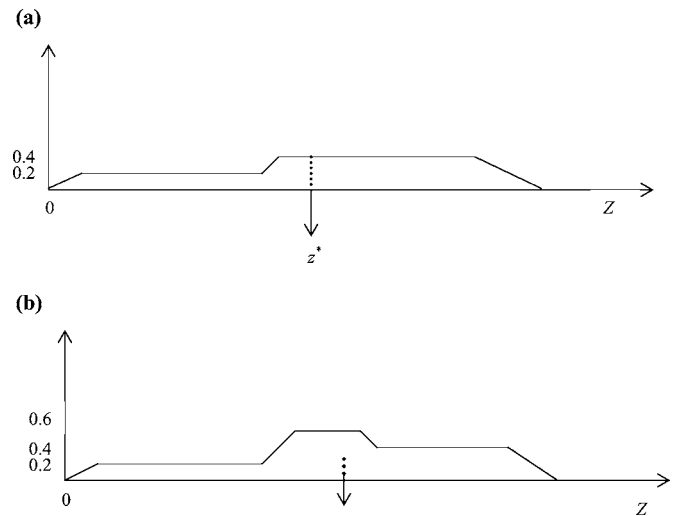


Fig. 4. Schematic representation of (a) **max** composition and **BOA** defuzzification (z^* halves the whole set); (b) **sum** composition and **COG** defuzzification (z^* is the abscissa under the center of gravity of the whole set)

fuzzy output set of high from the first rule with 0.4 firing strength (this value would be 0.32 by **prod** activation) and output set of low from the second rule with 0.2 firing strength (this value would be 0.12 by **prod** activation) [see Fig. 3(c)]. Note that inferencing produces not a crisp output value but assigns whole fuzzy output subsets to the output variable [see, shaded areas in Fig. 3(c)]. The next subprocess in inferencing is the composition subprocess where all of the fuzzy subsets assigned to the output variable are combined together to form a single subset for the output variable. For this purpose, there are basically two composition methods—maximization (**max**) and summation (**sum**). In **max** composition, the combined output fuzzy subset is constructed by taking the pointwise maximum over all of the fuzzy subsets assigned to the output variable by the inference rule. In **sum** composition, the combined output fuzzy subset is constructed by taking the pointwise sum over all of the fuzzy subsets. Consequently, in **sum** composition it is sometimes possible to obtain truth values greater than one. Note that **sum** composition must be followed by the center of gravity (**COG**) defuzzification method (Jantzen 1999). Figs. 4(a and b) present combined fuzzy output subsets derived by the **max** and **sum** compositions for the previous example, respectively.

Defuzzification converts the resulting fuzzy outputs from the fuzzy inference engine to a number. There are many defuzzification methods, such as **COG**, bisector of area (**BOA**), mean of maxima (**MOM**), left-most maximum (**LM**), and right-most maximum (**RM**), etc. (Jantzen 1999; Sen 1999). The **MOM**, **LM**, and **RM** methods disregard the shape of the fuzzy set and hence they are employed in particular problems (Jantzen 1999). The **BOA** method picks the abscissa of the vertical line that divides the area of the combined output fuzzy subset in two equal halves [Fig. 4(a)]. In the **COG** method the crisp output value is the abscissa under the centre of gravity of the combined output fuzzy subset [Fig. 4(b)]. The **COG** method is the most commonly used defuzzification method and for the discrete case it is expressed as (Jantzen 1999):

$$z^* = \frac{\sum_i \mu(z_i)z_i}{\sum_i \mu(z_i)} \quad (3)$$

where z^* =defuzzified output value; z_i =output value in the i th subset; and $\mu(z_i)$ =membership value of the output value in the i th subset. For the continuous case, the summations in Eq. (1) are replaced by the integrals.

The details of FL algorithm are available in the literature (McNeill and Thro 1994; Jantzen 1999; Sen 1998; Sen 1999; Tayfur et al. 2003).

Kinematic Wave Approximation

This study employed the experimental data from plots whose gradient varied from 5.7 to 40%. KWA reasonably approximates the full St. Venant equations in predicting runoff over plots whose gradient is greater than 3% (Woolhiser 1974; Tayfur et al. 1993). Hence, this study employed KWA which, in one dimension, can be expressed as (Tayfur 2002a):

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x} \left(\frac{\sqrt{S}}{n} h^{5/3} \right) = (r - i) \quad (4)$$

where h =flow depth [L]; r =rainfall intensity [L/T]; i =infiltration rate [L/T]; S =bed slope; and n =Manning's roughness coefficient [L^{1/3}/T].

Eq. (4) was solved numerically using an implicit centered finite-difference method. The multivariate Newton-Raphson iterative method was used for the solution of the set of nonlinear difference equations resulting from the implicit procedure. Initially, flow starts on a dry surface and hence zero flow depth was assumed over the whole surface as an initial condition. The zero flow depth was assumed as the upstream boundary condition. The details of the numerical solution can be found in Tayfur et al. (1993).

Evaluation of Models

The developed ANN, FL, and KWA models were calibrated and tested with peak discharge and runoff hydrograph data from the event-based rainfall-runoff experiments and comparison of the models is presented here.

Peak Discharge Prediction

ANN Model Training and Testing

For predicting peak discharge, the three-layer feedforward ANN model had 3 neurons in the input layer, 8 neurons in the hidden layer, and 1 neuron in the output layer. The rainfall intensity, infiltration rate, and slope were inputs and runoff rate was the target output. For the number of neurons in the hidden layer, a trial-and-error procedure was used. The sigmoid function was employed as an activation function and the supervised training algorithm of backpropagation was employed for training the network. Before starting the training process, random values of 0.2–0.4, and –1.0 were assigned for the network weights and biases, respectively. Also, before training and testing, all the external input and output data were standardized by (Tayfur 2002b).

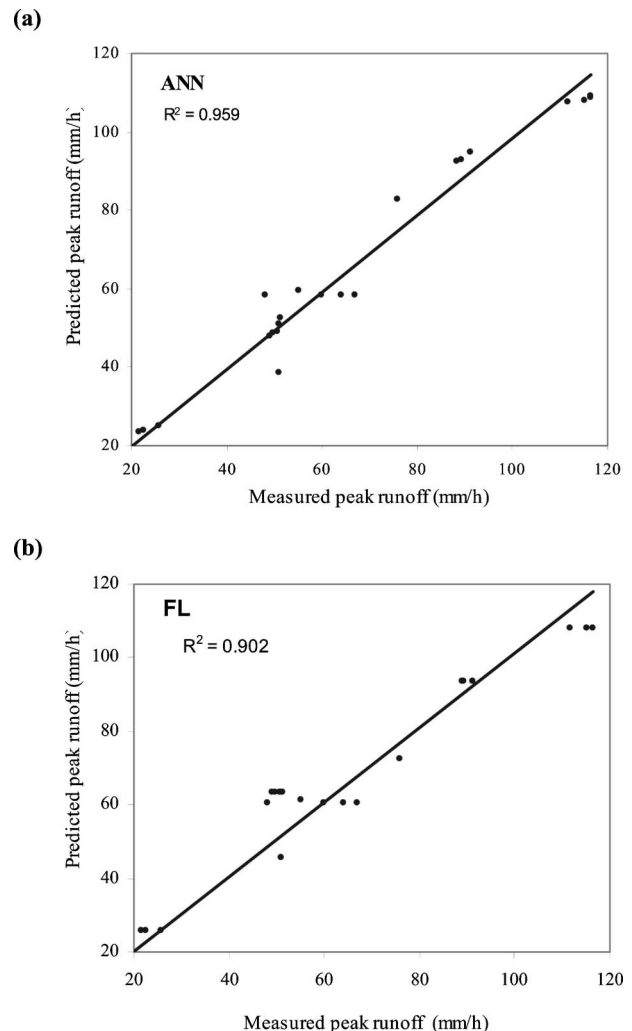


Fig. 5. Measured versus predicted peak runoff data by (a) ANN model; (b) FL model at training stage

$$z_i = \frac{x_i}{x_{\max} + 1} \quad (5)$$

where z_i =standardized value calculated for neuron i ; x_i =real input or output value applied to neuron i ; and x_{\max} =maximum input or output value of all values applied to neuron i .

For training the network, 25 sets of data (Table 1) were used. The training was accomplished with a 0.01 learning rate and 20,000 iterations were needed. Fig. 5(a) shows the measured runoff data versus the corresponding ANN-predicted output data at the end of training. As seen, the training of the ANN model was successfully accomplished with a $R^2=0.96$. Note that the coefficient of determination (R^2) can be calculated as

$$R^2 = \frac{\sum_{i=1}^N (Q_{mi} - \bar{Q}_m)(Q_{pi} - \bar{Q}_p)}{\left(\sum_{i=1}^N (Q_{mi} - \bar{Q}_m)^2 \right)^{0.5} \left(\sum_{i=1}^N (Q_{pi} - \bar{Q}_p)^2 \right)^{0.5}} \quad (6)$$

where Q_m =measured runoff rate [L/T]; \bar{Q}_m =measured mean runoff rate [L/T]; Q_p =predicted runoff rate [L/T]; \bar{Q}_p =predicted mean runoff rate [L/T]; and N =number of observations.

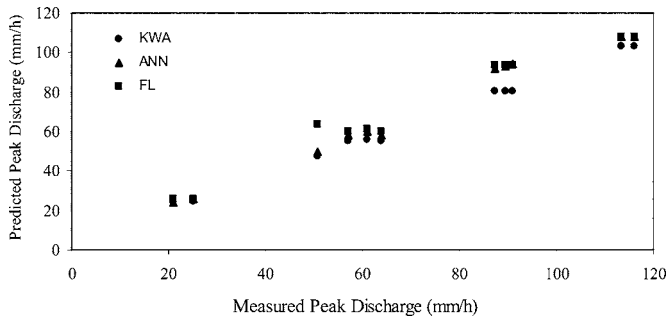


Fig. 6. Prediction of measured peak discharge data

The coefficient of efficiency (E) was also computed for Fig. 5(a). As this coefficient accounts for model errors in estimating the mean or variance of the observed data sets, it is commonly employed in hydrology. E ranges from minus infinity (poor model) to 1.0 (perfect model), and can be calculated as

$$E = 1.0 - \frac{\sum_{i=1}^N (Q_{mi} - Q_{pi})^2}{\sum_{i=1}^N (Q_{mi} - \bar{Q}_m)^2} \quad (7)$$

For Fig. 5(a), E is computed as equal to 0.96 which is very close to 1.0, implying the successful training of the ANN model.

The trained ANN model was then applied to predict peak discharge of 11 other sets of data (Table 1). The model-predicted peak runoff data are given in Fig. 6, from which it is seen that the ANN model satisfactorily predicted the measured data. In order to objectively evaluate the model performance, the most commonly employed error measures, such as the root-mean-square error (RMSE) and the mean absolute error (MAE) were computed for Fig. 6 and are summarized in Table 2. The RMSE and MAE are defined as

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (Q_{mi} - Q_{pi})^2}{N}} \quad (8)$$

$$MAE = \frac{\sum_{i=1}^N |Q_{mi} - Q_{pi}|}{N} \quad (9)$$

The values of $RMSE=4.11$ mm/h, $MAE=3.46$ mm/h, and $E=0.98$, as shown in Table 2, imply a satisfactory performance of the developed ANN model in predicting the peak discharge data.

Table 2. Calculated Error Measures

	ANN			FL			PB		
	RMSE (mm/h)	MAE (mm/h)	E	RMSE (mm/h)	MAE (mm/h)	E	RMSE (mm/h)	MAE (mm/h)	E
Peak									
Q	4.11	3.46	0.98	5.81	4.76	0.96	7.69	6.65	0.94
S3R2	8.79	5.74	0.95	11.62	7.44	0.91	9.86	6.75	0.94
T3V2	8.73	7.15	0.93	10.68	8.69	0.90	9.48	6.03	0.92

The experimental results of Barfield et al. (1983) and Kilinc and Richardson (1973) show that the gradient of a plot or a flume has minor effect on the peak runoff rate (see Table 1). For example, under 57 mm/h rainfall rate, the range of runoff rates from 5.7 to 40% slopes was quite narrow: 49.1–51.2 mm/h (Table 1). Similarly, under 78 mm/h rainfall rate, from slopes having almost the same gradients of 9.1–9.3%, the runoff rates ranged considerably higher as 48–64 mm/h (Table 1). These results imply that rainfall and infiltration rates exercise a greater influence on the rate of peak discharge than does the gradient of a plot.

In order to examine the effect of gradient on peak discharge, the ANN model was reconstructed by utilizing only the rainfall and infiltration data as input. The purpose here was to see whether utilizing only the rainfall intensity and infiltration rate data in the ANN model would be sufficient to satisfactorily predict the peak runoff rate. The reconstructed ANN model, in this case, had 2 input neurons, 5 hidden neurons, and 1 output neuron. The network was retrained for the same 25 data sets and tested against the same 11 data sets (Table 1). The results revealed that the exclusion of slope data caused an insignificant effect on the prediction of the peak runoff rate by the ANN model with a R^2 of 0.94. However, it should be noted that the slope of a plot would, no doubt, affect the rising and recession parts of a runoff hydrograph. Depending upon the steepness of a slope, the runoff hydrograph might reach equilibrium faster or slower. Similarly, it can recede faster or slower depending upon the mildness of the slope. However, under a uniform constant infiltration rate, the magnitude of the equilibrium discharge of the runoff hydrograph would almost be the same, regardless of the degree of the gradient of the plot.

FL Model Calibration and Validation

In the developed FL model, rainfall intensity and infiltration rate were input variables and runoff rate was the output variable. As the slope has an insignificant effect on peak discharge, it was not considered in the input vector of the FL model. The input and output variables, taking into account the measured data, were intuitively fuzzified and a triangular fuzzy membership function was used for each fuzzy subset of each variable in the constructed FL model [Figs. 7(a–c)]. The data used for the ANN training were employed in the calibration of the FL model where the Mamdani fuzzy rules relating rainfall rate and infiltration rate to peak discharge were constructed (Table 3). The FL model prediction of the measured 25 data sets at the end of calibration is shown in Fig. 5(b). As seen, the FL model was successfully calibrated with a $R^2=0.90$ and $E=0.92$.

The calibrated FL model was then applied to predict the peak discharge of other 11 sets of data (Table 1). The model-predicted peak runoff rate data is given in Fig. 6, from which it is seen that the FL model satisfactorily predicted the measured data. The

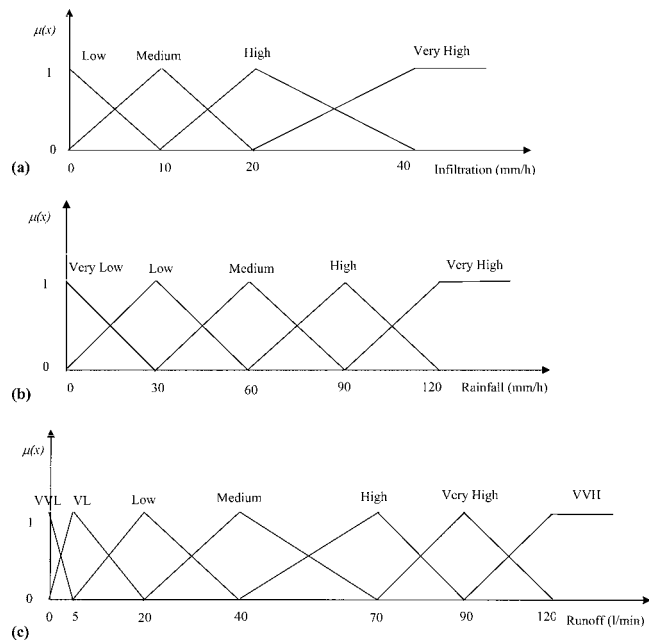


Fig. 7. Fuzzy membership functions for (a) infiltration; (b) rainfall; and (c) runoff (VVL=very very low; VL=very low; and VVH=very very high)

computed error measures for Fig. 6 is given in Table 2, where RMSE=5.81 mm/h, MAE=4.76 mm/h, and $E=0.96$.

KWA Model Calibration and Validation

The KWA model was calibrated for two different data sets in Table 1. The runs in Kilinc and Richardson (1973) had an average infiltration rate of 5.43 mm/h and the runs in Barfield et al. (1983) had an average infiltration rate of 18.8 mm/h. When the KWA model was calibrated for the data of Kilinc and Richardson (1973) it used a constant infiltration rate of 5.43 mm/h. Similarly, when it was calibrated for the data of Barfield et al. (1983) it used a constant infiltration rate 18.8 mm/h. For calibration runs, the experiment over a 15% slope under 57 mm/h rainfall intensity from Kilinc and Richardson (1973) and the experiment over a 9.2% slope under 97 mm/h rainfall intensity from Barfield et al. (1983) were employed (Table 1). The model calibration runs predicted 47.8 and 72.7 mm/h of peak runoff rates, respectively. The actual measured peak runoff rates are 50.7 and 76 mm/h, respectively (Table 1) and the difference in each case was less than 6% which is an acceptable error percentage. The calibrated model parameter of Manning's roughness coefficient [Eq. (1)] which resulted in these best fits is 0.016, which is in agreement with Woolhiser (1974).

The calibrated KWA model was then applied to predict the peak discharge of 11 data sets used for testing of ANN and FL models (Table 1). The model-predicted peak runoff rate data is given in Fig. 6, from which it seen that the KWA model satisfac-

Table 3. Rules Relating Rainfall and Infiltration to Peak Discharge

Infiltration rate	Rainfall rate			
	L	M	H	VH
M	L	H	VH	VVH
H	L	M	H	VH

Note: L=low; M=medium; H=high; and V=very.

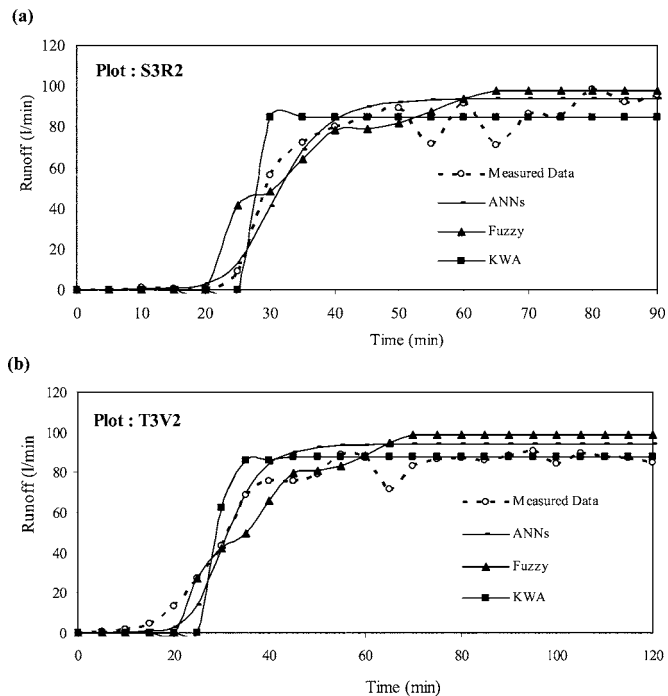


Fig. 8. Simulation of observed runoff hydrographs over experimental plots (a) S3R2; (b) T3V2

torily predicted the measured data. The computed error measures for Fig. 6 are given in Table 2, where RMSE=7.69 mm/h, MAE=6.65 mm/h, and $E=0.94$.

Runoff Hydrograph Simulations

ANN Model Training and Testing

The ANN model was trained by employing 4 experimental runoff hydrograph data from plots S1R2, S2R2, S3S2, and T1R2. Barfield et al. (1983) recorded the runoff data at every minute. However, in training of the network, the data measured at every 5 min were utilized in order to allow the network to recognize the pattern between the input and output variables. The gradients of the plots varied slightly from 8.6 to 9.3% and also the infiltration rate was almost uniform at 18.8 mm/h for each run. Hence, slope and infiltration rate were not considered in the network training. The input variables were only time and rainfall intensity and the target output variable was flow discharge. From the 4 runs, a total of 76 data sets were used in the training of the network. The network had 2 neurons in the input layer, 2 neurons in the hidden layer, and 1 neuron in the output layer. It was successfully trained with a 0.10 learning rate, 30,000 iterations, $R^2=0.85$, and $E=0.83$.

The trained ANN model was then applied to simulate runoff hydrographs from plots of S3R2 and T3V2 as shown in Figs. 8(a and b), respectively. As seen, the developed ANN model simulated the runoff hydrographs satisfactorily. The computed error measures for these simulations are given in Table 2.

FL Model Calibration and Validation

The FL model was constructed by considering rainfall intensity and time as input variables and flow discharge as output variable for the runoff hydrograph simulation. The reason that gradient and infiltration were not considered in the input vector is because,

Table 4. Mamdani Fuzzy Rules Relating Rainfall and Time to Discharge

Rainfall rate	Time						
	VVS	VS	S	Sf	Lg	VLg	VVLg
M	VVL	VVL	L	M	H	H	VH
H	VVL	VL	L	M	VH	VH	VVH
VH	VH	VL	H	VVH	VVH	VVH	VVH

Note: L=low; M=medium; H=high; S=short; Sf=sufficient; Lg=long; and V=very.

as explained earlier, the experiments have almost a constant gradient of 9.0% and an infiltration rate of 18.8 mm/h. The same data sets from the 4 runs [S1R2, S2R2, S3S2, and T1R2 (Barfield et al. 1983)] that were employed for the ANN model training were used for the FL model calibration where the Mamdani fuzzy rules relating rainfall intensity and time to runoff were derived (Table 4). The fuzzy membership functions for rainfall and runoff are presented in Figs. 7(b and c), respectively. The fuzzy membership functions for time variable are shown in Fig. 9. The FL model was successfully calibrated with a $R^2=0.84$ and $E=0.82$.

The calibrated FL model was then applied to simulate runoff hydrographs from plots of S3R2 and T3V2 as shown in Figs. 8(a and b), respectively. As seen, the developed FL model simulated the runoff hydrographs satisfactorily. The computed RMSE, MAE, and E values are given in Table 2.

KWA Model Calibration and Verification

The KWA model was calibrated for the experiment from S2R2 plot (Table 1). The calibration run is given elsewhere (Tayfur et al. 1993). The calibrated KW model was applied to simulate runoff hydrographs from S3R2 and T3V2 plots. Simulations are given in Figs. 8(a and b), respectively, where the KWA model satisfactorily simulated the hydrographs. The related error measures of RMSE, MAE, and E are given in Table 2.

Comparison of Modeling Results

As seen in Fig. 6, the ANN, FL, and KWA models satisfactorily predicted peak discharge data. Overall, higher values of the peak discharge data were better predicted by the ANN and FL models. The error measures for Fig. 6 are comparable for all the three models as presented in Table 2, whereas ANN had the minimum RMSE=4.11 mm/h and MAE=3.46 mm/h and maximum $E=0.98$. The average of the 11 peak discharge data sets used for testing the models is 70.5 mm/h. Hence, on a relative basis, on average, the errors are 5.4, 7.5, and 10.2% for ANN, FL, and KWA models, respectively, for the predicted peak discharge data.

With respect to runoff hydrograph simulation, all the three models performed satisfactorily as shown in Figs. 8(a and b). By comparison, as seen in Fig. 8(a), the S3R2 runoff hydrograph was better simulated by the ANN model. The ANN models simulated

both the rising and equilibrium parts of the hydrograph quite satisfactorily [Fig. 8(a)]. When the computed error measures for Fig. 8(a) in Table 2 were compared, it was seen that ANN had the minimum values of RMSE=8.79 L/min and MAE=5.74 L/min and the maximum value of $E=0.95$. The total measured runoff volume from the S3R2 experiment was 97.7 m³. The predicted runoff volumes for the S3R2 experiment by the three models were 102.9, 104.6, and 99.1 m³ for ANN, FL, and KWA models, respectively. On a relative basis, the errors were 5.1, 7.0, and 1.4% for ANN, FL, and KWA models, respectively.

The T3V2 runoff hydrograph was successfully simulated by all the models [Fig. 8(b)]. The ANN model had the minimum values of RMSE=8.73 L/min and the KW model had the minimum MAE=6.03 L/min for Fig. 8(b) (Table 2). The total runoff volume from the T3V2 experiment was 191.2 m³. The predicted runoff volumes for T3V2 were 205.2, 203.8, and 196.2 m³ for ANN, FL, and KWA models, respectively. On a relative basis, the errors were 7.3, 6.6, and 2.6% for ANN, FL, and KWA models.

Considering both the runoff hydrograph simulations, all three models, on average, had comparable values of RMSE, MAE and E (Table 2). Also, with respect to the total runoff volumes from both the experiments, all the models, on average, had comparable predictive performances, although KW had, on average, the minimum error of 2.0%, followed by ANN and FL models with 6.2 and 6.8% errors, respectively (Table 2).

Discussion

To summarize, the ANN and FL models developed in this study for simulating event-based runoff due to rainfall, considered not only rainfall but also gradient, infiltration, and time variables in the input vectors of the models. The performance of the ANN and FL models was comparable with that of the KWA model. The KWA model has well defined relations among input and output variables and allows one to obtain spatial and temporal variations of the state variables over the domain of interest for different values of model parameters. Such information is essential for the study of erosion, sediment transport, and solute transport processes. Furthermore, the KWA model can be more effective when extensive data in the domain of interest is available.

The FL and ANN models make no assumptions on the physics of the process and hence simplify the process. However, ANNs do not reveal any explicit mathematical relation between input and output variables of the process, and hence it is difficult to gain an insight into understanding the physics of the process. Also, ANNs are not good extrapolators for the cases where they are not trained for. On the other hand, in the FL model, all the model complications and uncertainties are included in the descriptive fuzzy inference procedure in the form of IF-THEN statements. The transparency of fuzzy rules provides explicit qualitative and quantitative insights into the physical behavior of the system. This study may be the first to address the use of ANN and FL models

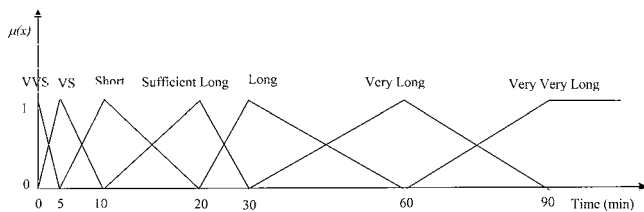


Fig. 9. Fuzzy membership functions for time variable (VVS=very very short and VS=very short)

Table 5. Characteristics of Storms Observed in Alazan Creek, San Antonio, Tex. (<http://cleveland1.cive.uh.edu>)

Event data	Average rainfall intensity (mm/h)	Peak discharge (m ³ /s)	Peak time (min)
May 15, 1969 ^a	6.66	25.04	50
May 8, 1975	31.8	124.1	35
June 10, 1975	7.05	29.75	35
April 4, 1976	3.04	22.75	45

^aUsed for model application.

for simulating event based rainfall runoff and compare the models with the KWA model. Hence, it might give an insight regarding the appropriateness of using nondeterministic ANN and FL models versus a deterministic model of KWA for predicting runoff.

ANN and FL Model Simulation of a Hydrograph at a Watershed Scale

ANN and FL models were applied to simulate a runoff hydrograph at a larger scale such as a small watershed. Data from laboratory and experimental plots can be obtained under relatively controlled conditions such that one can achieve a spatial and temporal uniform rainfall and infiltration resulting in an idealized situation. On the other hand, in real world problems, one can rarely encounter such idealized situations. Nevertheless, the developed artificial intelligence models presented in this study can be applied to the real world data as well. The following is an example applying the artificial intelligence models to real world data in order to show their usefulness for the practicing engineers.

Data from Alazan Creek Watershed, San Antonio, Tex., were obtained from the website (<http://cleveland1.cive.uh.edu> with) in public domain and employed. The watershed characteristics are as follows:

Station no:	08178300
Latitude:	29° 27' 29"
Longitude:	98° 32' 59"
Area:	8.44 km ²
Perimeter:	12,805 m
Main stream length:	4,953 m
Highest elevation:	305 m
Lowest elevation:	213 m

For calibrating and applying the models 4 events were employed. Table 5 summarizes the characteristics of the events. Of the 4 events, 3 were used for calibrating the models (Table 5). The calibrated ANN model had 2 neurons in the input layer, 12 neurons in the hidden layer, and 1 neuron in the output layer. Time and average rainfall intensity formed the input variables and discharge was the output variable. The model was trained with 49,000 iterations and a 0.0022 learning rate. The constructed FL model had a (0–120 mm/h) range with 5 fuzzy subsets for the average rainfall intensity. Time variable had a (0–180 min) range with 8 subsets and discharge variable had a (0–100 m³/s) range with 8 subsets. The fuzzy rules relating time and rainfall intensity to discharge were inferred from the measured data that were used for the FL model calibration.

Fig. 10 shows an application of the calibrated models for simulating the hydrograph measured on May 15, 1969 in Alazan Creek Watershed, Tex. As seen in Fig. 10, both the models performed satisfactorily. For Fig. 10, ANN model had MAE=1.98 m³/s; RMSE=2.41 m³/s; and $E=0.90$ whereas FL

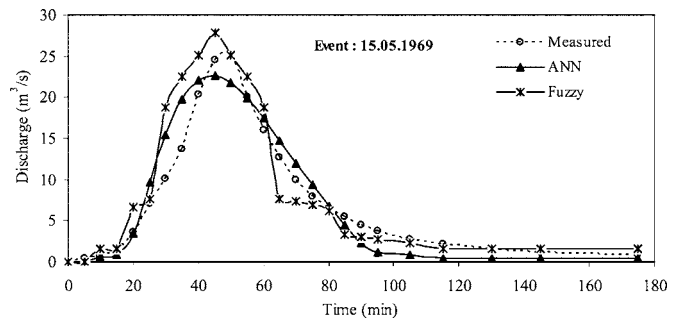


Fig. 10. Simulation of hydrograph measured in the Alazan Creek Watershed

model had MAE=1.64 m³/s; RMSE=2.52 m³/s, and $E=0.89$. Both the models had lower error measures (MAE; RMSE) and higher values of coefficient of efficiency (E) implying that these models can be also be applied to simulate runoff hydrographs at small watershed scales.

Concluding Remarks

The following conclusions are drawn from this study: (1) ANN and FL models were applied to predict runoff at a very small scale of a flume (6.9 m²); a larger scale of a plot (92.4 m²); and a small scale of a watershed (8.44 m²); (2) the models need to be recalibrated with sufficient site specific measured data when applied at larger scales such as large watersheds; (3) ANN and FL models are based on the observations of the physical system and consequently they require sufficiently long historical data for describing the process under consideration; (4) the KWA equation is valid across different scales. It only requires the recalibration of the model parameters pertinent to the specific area; and (4) it is easier to construct ANN and FL models than KWA model. The KWA model has many parameters to be estimated from the field and it requires complicated techniques for the solution of its non-linear equation.

Notation

The following symbols are used in this paper:

- E = predetermined error function;
- h = flow depth;
- i = infiltration rate;
- MAE = mean absolute error;
- N = number of observations;
- n = Manning's roughness coefficient;
- P = number of training patterns;
- p = number of output neurons;
- \bar{Q}_m = average measured runoff rate;
- Q_m = measured runoff rate;
- Q_p = predicted runoff rate;
- R^2 = coefficient of determination;
- RMSE = root-mean-square error;
- r = rainfall intensity;
- S = bed slope;
- t_i = component of a target output vector \mathbf{T} ;
- x_i = input value;
- x_{\max} = maximum input value;

- v_{ij} = weight from i th neuron in the previous layer to the j th neuron in the current layer;
 y_i = component of a network output vector \mathbf{Y} ;
 z_i = standardized value; and
 δ = learning rate.

References

- Ancil, F., and Rat, A. (2005). "Evaluation of neural network streamflow forecasting on 47 watersheds." *J. Hydrol. Eng.*, 10(1), 85–88.
- ASCE Task Committee. (2000a). "Artificial neural networks in hydrology. I: Preliminary concepts." *J. Hydrol. Eng.*, 5(2), 115–123.
- ASCE Task Committee. (2000b). "Artificial neural networks in hydrology. II: Hydrologic applications." *J. Hydrol. Eng.*, 5(2), 124–137.
- Barfield, B. J., Barnhisel, R. I., Powell, J. L., Hirschi, M. C., and Moore, I. D. (1983). "Erodibilities and eroded size distribution of western Kentucky mine spoil and reconstructed topsoil." *Institute for Mining and Minerals Research Final Rep.*, Univ. of Kentucky, Lexington, Ky.
- Chang, L. C., Chang, F. J., and Tsai, Y. H. (2005). "Fuzzy exemplar-based inference system for flood forecasting." *Water Resour. Res.*, 41.
- Dawson, W. C., and Wilby, R. (1998). "An artificial neural network approach to rainfall-runoff modeling." *Hydrol. Sci. J.*, 43(1), 47–66.
- Govindaraju, R. S., Kavvas, M. L., and Tayfur, G. (1992). "A simplified model for two dimensional overland flows." *Adv. Water Resour.*, 15, 133–141.
- Hong, Y. S., Rosen, M. R., and Reeves, R. R. (2002). "Dynamic fuzzy modeling of storm water infiltration in urban fractured aquifers." *J. Hydrol. Eng.*, 7(5), 380–391.
- Jantzen, J. (1999). "Design of fuzzy controllers." *Technical Rep. No. 98-E864*, Dept. of Automation, Technical Univ. of Denmark, Denmark.
- Kilinc, M., and Richardson, E. V. (1973). "Mechanics of soil erosion from overland flow generated by simulated rainfall." *Hydrology papers*, Colorado State Univ., Fort Collins, Colo., Paper 63.
- Maskey, S., Guinot, V., and Price, R. K. (2004). "Treatment of precipitation uncertainty in rainfall-runoff modeling: A fuzzy set approach." *Adv. Water Resour.*, 27(9), 889–898.
- McNeill, F. M., and Thro, E. (1994). *Fuzzy logic: A practical approach*, Hyperion, New York.
- Olsson, J., et al. (2004). "Neural networks for rainfall forecasting by atmospheric downscaling." *J. Hydrol. Eng.*, 9, 1–12.
- Ozelkan, E. C., and Duckstein, L. (2001). "Fuzzy conceptual rainfall-runoff models." *J. Hydrol.*, 253(1–4), 41–68.
- Rajurkar, M. P., Kothiyari, U. C., and Chaube, U. C. (2002). "Artificial neural networks for daily rainfall-runoff modeling." *Hydrol. Sci. J.*, 47(6), 865–877.
- Ramírez, M. C. V., Velho, H. F. C., and Ferreira, N. J. (2005). "Artificial neural network technique for rainfall forecasting applied to the São Paulo region." *J. Hydrol.*, 301(1–4), 146–162.
- See, L., and Openshaw, S. (2000). "A hybrid multi-model approach to river level forecasting." *Hydrol. Sci. J.*, 45(4), 523–536.
- Sen, Z. (1998). "Fuzzy algorithm for estimation of solar irradiation from sunshine duration." *Sol. Eng.*, 63(1), 39–49.
- Sen, Z. (1999). "Fuzzy modelling in engineering." *Class notes*, Civil Engineering Faculty, Istanbul Technical Univ., Istanbul, Turkey (in Turkish).
- Somez, I. (1998). "Meteorological applications of artificial neural networks." MSc thesis, Dept. of Meteorological Engineering, Istanbul Technical Univ., Istanbul, Turkey (in Turkish).
- Tayfur, G. (2002a). "Applicability of sediment transport capacity models for non-steady state erosion from steep slopes." *J. Hydrol. Eng.*, 7(3), 252–259.
- Tayfur, G. (2002b). "Artificial neural networks for sheet sediment transport." *Hydrol. Sci. J.*, 47(6), 879–892.
- Tayfur, G., Kavvas, M. L., Govindaraju, G. S., and Storm, D. E. (1993). "Applicability of St. Venant equations for two-dimensional overland flows over rough infiltrating surfaces." *J. Hydraul. Eng.*, 119(1), 51–63.
- Tayfur, G., Ozdemir, S., and Singh, V. P. (2003). "Fuzzy logic algorithm for runoff-induced sediment transport from bare soil surfaces." *Adv. Water Resour.*, 26, 1249–1256.
- Tayfur, G., and Singh, V. P. (2005). "Predicting longitudinal dispersion coefficient in natural streams by artificial neural network." *J. Hydraul. Eng.*, 131(11), 991–1000.
- Tayfur, G., Swiatek, D., Wita, A., and Singh, V. P. (2005). "Case study: Finite element method and artificial neural network models for flow through Jeziorsko earthfill dam in Poland." *J. Hydraul. Eng.*, 131(6), 431–440.
- Tilmant, A., Vanclooster, M., Duckstein, L., and Persoons, E. (2002). "Comparison of fuzzy and nonfuzzy optimal reservoir operating policies." *J. Water Resour. Plann. Manage.*, 128(6), 390–398.
- Tokar, A. S., and Johnson, P. A. (1999). "Rainfall-runoff modeling using artificial neural networks." *J. Hydrol. Eng.*, 4(3), 232–239.
- Tokar, S. A., and Markus, M. (2000). "Precipitation-runoff modeling using artificial neural networks and conceptual models." *J. Hydrol. Eng.*, 5(2), 156–161.
- Woolhiser, D. A. (1974). "Unsteady free-surface flow problems." *Proc., Inst. on Unsteady Flow in Open Channels*, Colorado State Univ., Fort Collins, Colo., 195–213.
- Wu, J. S., Han, J., Annambhotla, S., and Bryant, S. (2005). "Artificial neural networks for forecasting watershed runoff and stream flows." *J. Hydrol. Eng.*, 10(3), 216–222.
- Yu, P.-S., and Yang, T.-C. (2000). "Fuzzy multi-objective function for rainfall-runoff model calibration." *J. Hydrol.*, 238(1–2), 1–14.