An Artificial Neural Network Approach for Flood Forecasting

S Kumar, *Non-member*

R Kumar, *Non-member*

B Chakravorty, *Associate Member*

C Chatterjee, *Non-member*

N G Pandey, *Non-member*

Short term flood forecasts are used to issue flood warning to people residing in flood plains and act as an important non- structural solution for reducing flood damages. Flood forecasts are generally subject to errors, which may lead to false warning, thus affecting the evacuation and other relief operations. As flood forecasting has been an area of extensive research, new techniques have been introduced to minimize the forecast errors and to issue more accurate forecasts. Recently artificial neural network (ANN) has been introduced in modelling real time problems. The use of ANN in real time flood forecasting is new and still in the evolution stage. This study explains the ANN approach for flood forecastings and its working. The advantages of using ANN approach in modelling the rainfall runoff relationship over conventional techniques and a brief review of other methods and their limitations are also explained.

Keywords : Flood forecasting; Artificial neural network; Rainfall run off; Conventional techniques.

INTRODUCTION

Floods are natural disasters and are frequent in the Ganga and Brahamputra basins in the eastern and north-eastern part of the country. Flood brings devastation and wide spread miseries to life. Though complete immunity from floods is not possible, yet the damage potential due to floods can be reduced through various structural and non-structural measures. The structural measures are capital intensive compared to non-structural measures, which are less expensive. Non-structural measures are basically used to provide flood forecasts and warning to people who reside in different flood zones.

Since the eastern parts of the country are densely populated and there is a large habitation especially in flood prone areas, proper forecasting and warning can alleviate a lot of distress and damage. The development in information and communication technology can help in promptly communicating the forecast and warning to appropriate authorities so that effective steps can be undertaken. Authorities can warn the dewellers in the flood risked zones, based on the available forecast. However, this warning may be severely affected if the forecasts are subject to large errors. This may result in false warnings ie a forecast of high flood may follow a low flood or vice-versa, thus further increasing the sufferings of the flood plain dewellers.

Floods are natural phenomena and inherently complex to model. The quantity of runoff resulting from a given rainfall event depends on a number of factors. Different techniques have been used to generate the stream flows. Statistical correlation techniques have been employed[1] for direct prediction of water levels. The routing techniques[2] are more useful, when the travel time is longer and downstream flow is low or controlled. A range of conceptual models is available for computation of stream runoff based on rainfall inputs. The most complex of rainfall- runoff models currently being used are conceptual storage models. The US National Weather Service River Forecast System (NWSRFS) is a representative example. Relatively simple conceptual models for modelling rainfall-runoff process, are based on convoluting the response of the catchment to the effective rainfall. The instantaneous unit hydrograph is an example of this approach. Time series models can be used to describe the stochastic structure of the time sequences of stream flows and precipitation[3] and are most suitable for on line forecasting of floods. Time series modelling techniques have been shown to provide systematic empirical methods for simulating and forecasting the runoff resulting from a given rainfall event. However a major limitation of this model is that the structure of the model is generally taken as linear and is apriori chosen. Realistically the rainfall-runoff process is highly non-linear and is dependent on many interrelated physiographical and hydrological processes. This shortcoming can be eliminated if the non-linear structure of the model is identified by the modelling process itself.

Yet another class of Black-Box models, in the form of artificial neural network (ANN) are used for on-line forecasting. These methods are capable of adopting the non-linear relationship between rainfall and runoff as compared to conventional techniques, which assumes a linear relationship between rainfall and runoff. The applications of ANN in simulation and forecasting problems of water resources are few and relatively recent[6 – 10]. ANN has strong generalisation ability, which means that once properly trained, they can provide accurate results even for cases they have never seen before[11, 12]. The ANN are particularly useful in flood forecasting when the time required to generate a forecast is very short. These factors combine to make ANN a powerful tool for modelling problems in which functional relationship between

---

S Kumar, R Kumar, B Chakravorty, C Chatterjee and N G Pandey are with National Institute of Hydrology, Patna 801 505.

This paper was presented at the 16th National Convention of Computer Engineers held at Patna on December 15-16, 2001.

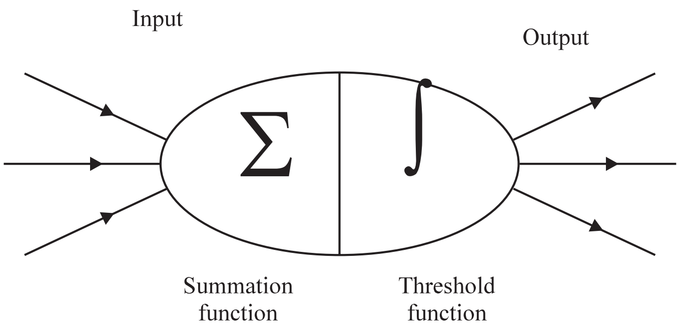

Figure 1 Simulated neuron

dependent (runoff) and independent variables (rainfall) are subject to uncertainty.

ARTIFICIAL NEURAL NETWORK (ANN) METHODOLOGY

Preliminaries

The neural-network approach, also referred to as connectionism or paralleled distributed processing, adopts a "Brain metaphor" of information processing. Information processing in a neural network occurs through interactions involving large number of simulated neurons, such as the one depicted in Figure 1. The simulated neuron, or unit, has four important components:

(i) Input connections, *ie*, synapses, through which the unit receives activation from other units.

(ii) Summation function that combines the various input activations into a single activation.

(iii) Threshold function that converts summation of input activation into output activation.

(iv) Output connections (axonal paths) by which a unit's output activation arrives as input activation at other units in the system.

Artificial neural-networks (ANNs) are massively parallel systems composed of many processing elements connected by links of variables weights. The back propagation network is the most popular[4] among ANN paradigms. The network consists of layers of neurons, with each layer being fully connected to the proceeding layer by inter connection strengths or weights *W*. Figure 2, illustrates a three-layer neural network consisting of input layer (L_i), hidden layer (L_H) and the output layer (L_o) with the inter-connection weights W_{ih} and W_{ho} between layers of neurons. Initial estimated weight values are progressively corrected during a training process that compares predicted outputs to known outputs, and back propagates any errors to determine the appropriate weight adjustments necessary to minimise the errors. The methodology used here for adjusting the weights is called "back algorithm" and is based on the "generalised delta rule"[5].

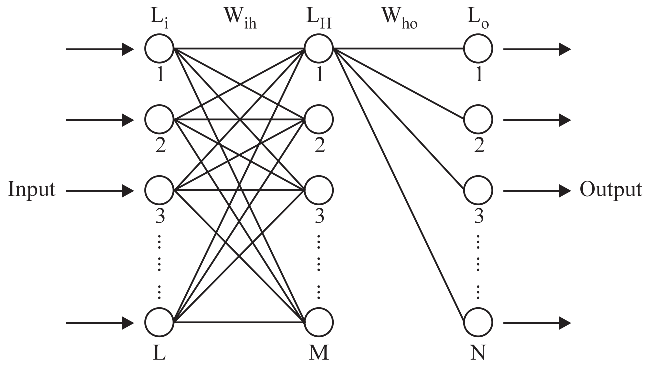

Figure 2 Configuration of three-layer neural network

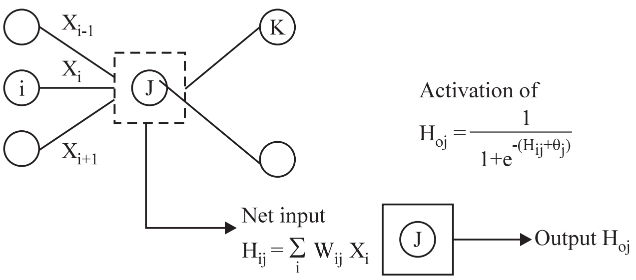

Figure 3 Neuron and its function

BACK PROPAGATION ALGORITHM

The generalised delta rule, which determines the approximate weight adjustments necessary to minimise the errors can be explained through Figure 3 which shows a neuron (*j*) and its functions. The total input H_{ij} to hidden units *j* is a linear function of outputs x_i of the units that are connected to *j* and of the weights w_{ij} on these connections *ie*.

$$H_{ij} = \sum_{i} x_i w_{ij} \tag{1}$$

units can be given biases (θ_j) by introducing extra input to each unit which always has a value of 1.

A hidden unit has a real-value output H_{oj}, which is a non-linear function of the total input.

$$H_{oj} = \frac{1}{1 + e^{-(H_i j + \theta_j)}} \tag{2}$$

The aim is to find a set of weights that ensure that for each input vector, the output vector produced by the network is the same or sufficiently close to the desired output vector. If there is a fixed, finite set of input-output cases, the total error in the performance of the network with a particular set of weights can be computed by comparing the actual and desired output vectors for every case. The total error *E*, is defined as:

$$E = \frac{1}{2} \sum_{C} \sum_{f} (O_{j,c} - T_{j,c})^2 \tag{3}$$

where '*c*' is an index over cases, *ie*, input-output pairs; *j* the index over output units; '*O*', actual state of an output unit; and T, is its targeted stated state. To minimise *E* by gradient descent, it is necessary to compute the partial derivative of *E* with respect to each weight in the network, *ie*, $\partial E/\partial W_{ji}$. This can be successively computed as follows:

Differentiating equation (3) for a particular case, *c*,

$$\frac{\partial E}{\partial O_j} = (O_j - T_j) \tag{4}$$

Next $\partial E/\partial x_j$ is computed using chain rule, *ie*

$$\frac{\partial E}{\partial x_j} = \frac{\partial E}{\partial O_j}\frac{\partial O_j}{\partial x_j} \qquad (5)$$

Differentiating equation (2) to get the value of $\partial O j/\partial x j$ and substituting in (5)

$$\frac{\partial E_j}{\partial x_j} = \frac{\partial E}{\partial O_j} O_j(1 - O_j) \qquad (6)$$

equation (5) shows how the change in the total input '*x*' to an output unit, will affect the error *E*. The total input is a linear function, of the states of the lower level units and also linear function of the weights on the connections. It is, therefore, easy to compute how the error will be affected by changing these states and weights. For a weight w_{ij}, from *i* to *j* the derivative is

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial x_j}\frac{\partial x_j}{\partial w_{ji}} = \frac{\partial E}{\partial x_j} O_i \qquad (7)$$

and for the output of the *i*th unit the contribution to $\partial E/\partial O_i$ resulting from the effect of *i* on *j* is simply

$$\frac{\partial E}{\partial x_j}\frac{\partial x_j}{\partial O_j} = \frac{\partial E}{\partial x_j} w_{ji} \qquad (8)$$

Taking into account all the connections emanating from unit *i*

$$\frac{\partial E}{\partial O_i} = \sum_j \frac{\partial E}{\partial x_j} w_{ij} \qquad (9)$$

Given $\partial E/\partial O$ for all units *j*, in the previous layer, the $\partial E/\partial O_i$ in the penultimate layer can be computed using equation (9). The same procedure can be repeated for the successive layers.

The simplest version of gradient descent is to change each weight by an amount proportional to the accumulated $\partial E/\partial W$.

$$\Delta w = - \in \partial E/\partial w \qquad (10)$$

where $\in$ is the learning rate. The convergence of equation (10) can be significantly improved, by acceleration method wherein the incremental weights at *t* can be related to the previous incremental weights using equation (11).

$$\Delta w\,(t) = \in \frac{\partial E}{\partial w\,(t)} + \alpha\,\Delta\,w\,(t-1) \qquad (11)$$

where α is an exponential decay factor between '0' and '1' that determines the relative contribution of the current gradient and earlier gradients to the weight change.

Network Training and Identification of ANN Model

The data are divided into two statistically similar parts. (i) For training and (ii) Testing the performance of the ANN. The training of the neural network is accomplished by adjusting the interconnecting weights till such time that the root mean square error (rmse.) between the observed and the predicted set of values is minimised. The adjustment of inter-connecting weights is accomplished using the back propagation algorithm. It may however be highlighted here that the accuracy of prediction and the network's learning ability can be severely affected if the architecture is not suitable. The standard back propagation algorithm can train only on a network of predetermined size. Several network architectures with different number of input neuron in the input layer, and different number of hidden layers with varying number of hidden neurons have to be considered to select the optimal architecture of the network. A trial and error procedure based on the minimum r m s e. crite- rion is used to select the best network architecture.

Validation

Before a developed neural network model can be used for flood forecasting with certain degree of confidence, there is a need to establish the validity of the forecasts it generates. A model could provide almost perfect answers to the set of problems with which it was trained, but fails to produce meaningful answers to other patterns. Validation involves evaluating the network performance on a set of test problems that were not used for training, but for which solutions are available for comparison. The correct solutions and those produced by the network may be compared in qualitative manner (plotted points) or in a quantitative manner using various statistical tests such as the correlation coefficient. The evaluation of the network on the test problems should be undertaken after training has been completed.

CONCLUSION

Neural networks offer several advantages over the conventional approaches to computing. The ability of ANN to develop a generalized solution to a problem from a set of examples, allows them to be applied to problems other than that used for training and to produce valid solutions even when there are errors in the training data. These factors combine to make neural network a powerful tool for modelling problems in which functional relationship are subject to uncertainty, or likely to vary with the passage of time. Such problems are common in hydrologic processes.

Another area that can be benefited from neural network approach is where the time required to generate solution is critical, such as real time applications that require many solutions in quick succession. The ability of neural networks to produce quick solutions irrespective of the complexity of the problem, make them valuable even when alternative techniques are available that can produce more accurate solutions.

This paper has concentrated on the most popular class of neural network system, *ie*, feed forward backpropagation networks and their working. However, there are many alternative forms of neural network systems and many different ways in which they may be applied to a given problem. The success of neural networks in model development depends not only on factors mentioned in the study but also on the selection of an appropriate training algorithm and strategy for application.

REFERENCES

1. K N Mutreia, A Yin and I Martino. 'Flood Forecasting Model for Citandy River.' Flood Hydrology, Eds V P Singh and Reidel Dordrecht, Netherlands, 1987, pp 211-220.

2. V P Singh. 'Hydrologic Systems — Rainfall-Runoff Modelling.' vol 1, *Prentice Hall Inc., Englewood Cliffs*, New Jersey, USA, 1988.

3. A Kumar. 'Prediction and Real Time Hydrological Forecasting.' Ph D Thesis, *Indian Institute of Technology*, Delhi, 1980.

4. R P Lippman. 'An Introduction to Computing with Neural Networks.' *IEEE ASSP Magazine*, 1987, pp 4-22.

5. D E Rumelhart, G E Hinton and R J Williams. 'Learning Internal Representation by Error Propagation.' In: Parallel Distributed Processing: Explorations in the Microstructures of Cognition, *MIT Press*, Cambridge, 1986.

6. M N French, W F Krajewski and R R Cuykendall. 'Rainfall Forecasting in Space and Time using a Neural Network. *Journal of Hydrology*, vol 137, 1992, pp 1-31.

7. K W Kang. 'Evaluation of Hydrologic Forecasting System based on Neural Network Model.' *Proceedings of IAHS Floods and Droughts*, Japan, 1993.

8. A M Buch, H S Mazumdar and R C Pandey. 'A Case Study of Runoff Simulation of Himalayan Glacier Basin.' *Proceedings of International Joint Conference on Neural Networks*, vol 1, 1993, pp 971-974.

9. N Karunanithi, W J Greeney, D Whitley and K Boree. 'Neural Network for River Flow Prediction. *Journal of Computing in Civil Engineering, ASCE*, vol 80, 1994, pp 201-219.

10. J Smith and R N Eli. 'Neural Network Models of Rainfall- Runoff Process.' *Journal of Water Resource Planning and Management*, vol 121, no 6, 1995, pp 499-507.