Neural network models for river flow forecasting

Nguyen Tan Danh, Huynh Ngoc Phien* and Ashim Das Gupta
Asian Institute of Technology, PO Box 4, KlongLuang 12120, Pathumthani, Thailand

Abstract

In this study, back propagation neural network (BPNN) models were used to forecast daily river flows in two basins, namely the Da Nhim and La Nga basins, in the Central Highlands of Vietnam for comparison with the Tank Model. It was found that the developed BPNN models provided satisfactory forecast discharges for both basins. Moreover, the discharges were also forecast from individual data of different stations within the La Nga Basin which were input directly and separately to the considered model. In this case, however, the model took a longer time to run and the corresponding forecast discharges were not as accurate as those obtained when mean areal values of rainfall and evaporation were used instead.

Introduction

Forecasting of daily discharges has been one of the important problems for hydrologists, reservoir operators and for flood protection engineers. In this connection, the relationship between rainfall and runoff has been widely exploited in many conceptual rainfall-runoff models, of which the Tank Model of Sugawara (1961; 1979) appears to be a well-known example (Phien and Pradhan, 1983; Phien and Danh, 1997). Besides these conceptual models, several black-box models (having little or no physical considerations) have also been used. These models include the time-series approach using Box-Jenkins ARIMA models, multiple regression models (Phien et al., 1990) etc. Recently, back propagation neural networks (BPNNs), a particular type of neural network, have been developed and successfully used in many fields (Gorr et al., 1994; Lachtermacher and Fuller, 1995, Maier and Dandy; 1996).

In this study, BPNNs were used to develop models for forecasting daily discharges (with lead time equal to one day) in two catchments, namely the Da Nhim and La Nga Basins in the Central Highlands of Vietnam (Phien and Danh, 1997). Besides daily rainfall and evaporation data, the input to these models may include past values of the discharge itself, thanks to the structure of BPNNs. As a consequence of this flexibility in input data, the effect of the different combinations of input data was also investigated.

Back propagation neural networks

Neural networks are mathematical models of theorised mind and brain activities which attempt to exploit the massively parallel local processing and distributed storage architecture of the human brain. The basic building block of the brain and nervous system is the neuron, that sends/receives information to/from various parts of the body. Each neuron collects inputs from a single or multiple sources and produces a single output in accordance with a certain predetermined non-linear function. A neural network model is created by interconnecting many of these simple neuron models in a known configuration.

Among the different neural network structures, BPNNs introduced by Rumelhart et al. (1986) are most popular because of their applicability in many different areas (see, for example, Wasserman, 1989; Gershenfield and Weigend, 1993; Phien and Siang, 1993; Shamseldin, 1997). For forecasting purposes, a typical BPNN model consists of an input layer, one or two hidden layers and an output layer that has only one node. Shown in Fig. 1 is a typical simple structure which is most commonly used in forecasting. In this case, the input layer has several nodes, each representing an input variable. The hidden layer also has several nodes and represents the non-linearity of the network system. The output layer has only one node which represents the forecast value corresponding to each set of input values. In principle, a BPNN may have several hidden layers, but in practice, only one or two layers are used. The number of nodes in the hidden layer is determined mainly by trial and error. Several attempts have been made to arrive at some kind of optimal structure of a BPNN model (Fahlman and Lebiere, 1990; Lim and Hong, 1993).

The back propagation method

Back propagation is a systematic method for training (calibrating) multilayer neural networks. It uses a set of pairs of input and output values (called patterns). An input pattern is fed into the network to produce an output, which is then compared with the actual output pattern. If there is no difference between the network output and the actual output, then no learning is needed. Otherwise, the weights - which express the contribution of the input nodes to the hidden nodes, and the hidden nodes to the output - are changed (backward from the output layer through the hidden layer(s) to the input layer). Since the training makes use of the actual output, the back propagation method is referred to as a ***supervised training*** method.

Notation

The following notation system is used:

$W_{ji,m}(n)$ weight of the effect received by j^{th} unit in layer m caused by i^{th} unit in layer (m-1) at n^{th} iteration.

$O_{j,m}$ output of the jth element in layer m (m = 1, 2, ..., L)

I_i i^{th} element of the input.

t_j j^{th} element of the desired output (target)

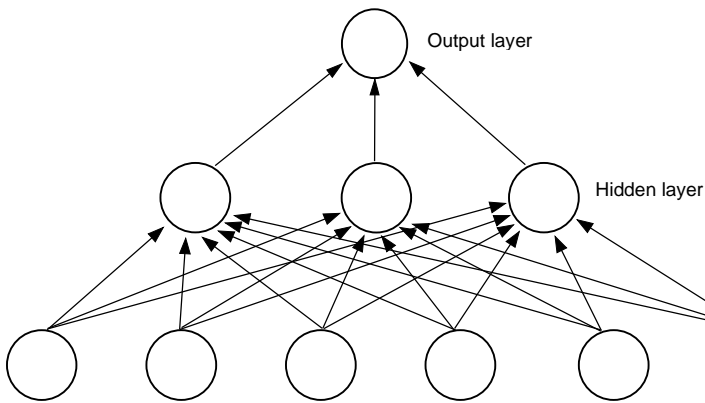n_m number of units in the mth layer.

Figure 1
A simple back propagation neural network for forecasting

The output of a node is obtained as a non-decreasing and differentiable function (called an activation or transfer function) of a linear combination (plus a bias term) of the node inputs. In practice, the *sigmoid* activation function is usually used for back propagation:

$$O_{j,m} = f(N_{j,m}) = \frac{1}{1 + e^{-N_{j,m}}} \qquad (1)$$

It has a simple derivative:

$$f'(N_{jj,n}) = O_{j,n}(1 - O_{jj,m}) \qquad (2)$$

where:

$$N_{j,m} = \sum_{i=1}^{n_{m-1}} W_{ji,m}.O_{i,m-1} + \theta_{j,m} \quad (m = 1, 2, \ldots, L) \qquad (3)$$

$O_{i,o} = I_i$: input unit

$\theta_{j,m}$: a bias (similar to the constant term in a regression model).

The output from the last layer (the network output) is compared with the actual (desired) output and the error for pattern p is calculated as:

$$E_p = \frac{1}{2} \sum_{j=1}^{n_L} (t_j - O_{j,l})^2 \qquad (4)$$

and the total error is:

$$E = \sum_p E_p \qquad (5)$$

(which is equal to half of the sum of squared errors). It should be noted that for the case of Fig.1, the number of layers L is 3 and n_L is 1.

Training procedure

During the training (calibration) stage, the weights and bias factors are estimated. The standard training procedure is summarised as follows with an explanation of the notation following Eq. (11):

1. Initialise all weights and bias factors to small random values.

Forward pass:

2. Present input vector $(I_1, I_2, \ldots, I_{no})$ and specify the desired output $(t_1, t_2, \ldots t_{nL})$

For layer m = 1, 2, ..., L:
a. Compute:

$$N_{j,m} = \sum_{i=1}^{n_{m-1}} W_{ji,m}.O_{i,m-1} + \theta_{j,m} \qquad (6)$$

with: $O_{i,o} = I_i$.

b. Compute the output of the jth unit in layer m:

$$O_{j,m} = \frac{1}{1 + e^{-N_{j,m}}} ; j = 1, 2, \ldots, n_m. \qquad (7)$$

3. Compare the final output $(O_{1,1}, O_{2,1}, \ldots, O_{nL,L})$ with the desired output $(t_1, t_2, \ldots, t_{nL})$ by computing:

$$E_p = \frac{1}{2} \sum_{j=1}^{n_L} (t_j - O_{j,L})^2$$

so that the total error is:

$$E = \sum_p E_p$$

where the sum extends all over data used in the training (calibration) stage. If the value of E reaches a minimum, the process is terminated and the system has learned. Otherwise, continue to the next step.

Backward pass:

4. For layer m = L, L-1,..., 1:

a. For j = 1, 2, ...,n_m ; compute the weight errors:

$$\delta_{j,m} = O_{j,m}(1 - O_{j,m}).(t_j - O_{j,m})$$

when m = L (output layer) $\qquad (8)$

$$\delta_{j,m} = O_{j,m}(1 - O_{j,m}) \sum_{k=1}^{n_{m+1}} W_{kj,m+1}.\delta_{k,m+1}$$

when m = hidden layers $\qquad (9)$

b. Compute the weight increments:

$$\Delta W_{ji,m}(n+1) = \eta.\delta_{j,m} O_{i,m-1} + \alpha \Delta W_{ji,m}(n) \qquad (10)$$

c. Compute the new values of the weights:

$$W_{ji,m}(n+1) = W_{ji,m}(n) + \Delta W_{ji,m}(n+1) \qquad (11)$$

5. Go to step 2.

For an explanation of the above procedure, it should be noted that the back propagation method tries to minimise the error E (or equivalently, the sum of squared errors) by adjusting the weights. Here:

$$\frac{\partial E_p}{\partial W_{ji,m}} = \frac{\partial E_p}{\partial N_{j,m}} \cdot \frac{\partial N_{j,m}}{\partial W_{ji,m}} \quad \text{by the chain rule,}$$

or, in view of Eq. (6):

$$\frac{\partial E_p}{\partial W_{ji,m}} = -\delta_{j,m} \cdot O_{i,m-1}$$

where:

$$\delta_{j,m} = \frac{\partial E_p}{\partial N_{j,m}} \tag{12}$$

For the nodes in the output layer (m = L):

$$\partial_{j,L} = -\frac{\partial E_p}{\partial O_{j,L}} \cdot \frac{\partial O_{j,L}}{\partial N_{j,L}}, \text{ again by the chain rule,}$$

or:

$$\partial_{j,L} = (t_j - O_{j,L}) \cdot f'(N_{j,L}) \text{ from Eqs. (2) and (4).}$$

Similarly, for the nodes in the hidden layer(s):

$$\delta_{j,m} = -\frac{\partial E_p}{\partial O_{j,m}} \cdot \frac{\partial O_{j,m}}{\partial N_{j,m}} \text{ (by the chain rule)}$$

By the chain rule, the partial derivative of E_p with respect to $O_{j,m}$ can be expressed as:

$$\frac{\partial E_p}{\partial O_{j,m}} = \sum_{k=1}^{n_{m+1}} \frac{\partial E_p}{\partial N_{k,m+1}} \frac{\partial N_{k,m+1}}{\partial O_{j,m}}$$

Using Eqs. (2), (6) and (12), we obtain:

$$\delta_{j,m} = (\sum_{k=1}^{n_{m+1}} \delta_{k,m+1} \cdot W_{kj,m+1}) f'(N_{j,m})$$

So the weight changes are:

$$\Delta W_{ji,m}(n+1) = \eta \cdot \delta_{j,m} \cdot O_{i,m-1} \tag{13}$$

where:

$$\delta_{j,m} = \begin{cases} f'(N_{j,L}) \cdot (t_j - O_{j,L}) \ , \ m = L \\ f'(N_{j,m}) \cdot \sum_{k=1}^{n_{m+1}} W_{kj,m+1} \cdot \delta_{k,m+1} \ , \ m < L \end{cases} \tag{14}$$

Here η is a constant which represents the **learning rate**. The larger η, the larger the changes in the weight, thus the faster the desired weight is found. But if η is too large, it may cause oscillations.

Rumelhart et al. (1986) proposed an additional term called the **momentum** which helps to increase the learning rate without leading to oscillations. With the addition of the momentum term, the weight changes become:

$$\Delta W_{ji,m}(n+1) = \eta \cdot \delta_{j,m} \cdot O_{i,m-1} + \alpha \Delta W_{ji,m}(n)$$

which is exactly Eq. (10). Clearly, α *(the momentum)* is a constant which determines the effect of the past weight changes

on the current direction of the movement.

There are several unresolved issues with BPNN models. Firstly, there is no automatic method to arrive at an optimum architecture (represented by the number of nodes in the input layer, the number of hidden layers and the number of nodes in hidden layers) for a BPNN model. Trial and error procedures have been used to arrive an acceptable structure. Secondly, the training procedure converges very slowly. It may converge to a local minimum of the total error (or the sum of squared errors) rather than the global minimum or it may not converge at all! The present study still follows the standard procedure introduced by Rumelhart et al. (1986), but limited to the BPNN models in which there is only one hidden layer. It should also be noted that instead of trying to obtain the (global) minimum value of the total error, E, the present study adopted a more relaxed stopping rule as follows:

$$\frac{|E(Old) - E(New)|}{E(Old)} \leq 0.001 \tag{15}$$

where E(Old) and E(New) denote the values of E obtained in one iteration and the next iteration, respectively. This stopping rule worked well in this study.

Applications

For comparison with the results obtained in the previous study (Phien and Danh, 1997), the back propagation method is applied to the forecasting of daily inflows to the Da Nhim Reservoir in the Da Nhim Basin (see Fig. 1, Phien and Danh, 1997) and of the daily discharge at Phu Dien in the La Nga Basin (see Fig. 2, Phien and Danh, 1997), both in the Central Highlands of Vietnam. The forecasting lead time is equal to one day. The performance indices introduced in the previous study (Phien and Danh, 1997), namely the efficiency index (EI), the root mean squared error (RMSE), the root mean squared error with respect to the mean (RMSEM) and the standard deviation (RMSES), and the mean absolute deviation (MAD), are used again in this work.

As BPNN models use the sigmoid function of which the output values lie in the interval [0,1], all the input values are transformed into the interval [0.05, 0.95] (instead of [0,1] because the logistic activation function approaches 0 and 1 asymptotically when the variable approaches negative infinity and positive infinity, respectively). This is done for any (input or output) variable X by using the following equation:

$$X' = 0.05 + 0.90*(X - X_{max}) / (X_{max} - X_{min}) \tag{16}$$

where X' is the transformed variable, X_{max} and X_{min} are its maximum and minimum values in the observed data.

Once the output values are obtained from a BPNN model, the daily discharge values are obtained by transforming them back to the original scale using the inverse transformation of Eq.(16):

$$X = X_{min} + (X_{max} - X_{min})* (X' - 0.05)/0.9 \tag{17}$$

There is obviously a constraint imposed on the output by Eq. (17): as X' $\leq$ 0.95, the values obtained from a BPNN model for the discharge are always less than or equal to X_{max}: $X \leq X_{max}$. Similarly, as X' $\geq$ 0.05, the values obtained from a BPNN model are always greater than or equal to X_{min}, i.e. $X \geq X_{min}$. In other words the forecast values of the discharge will always fall in the range of its past observed values. To avoid this constraint, the values of X may be assumed to lie in the interval [0.8 X_{min}, 1.2 X_{max}], or in a wider interval.

Da Nhim Basin

Data application

- For training (calibration): data in 7 years (1982 - 1988)
- For testing (validation) : data in 4 years (1989 - 1992)

The streamflow station under consideration is located at the Da Nhim Dam (see Fig.1, Phien and Danh, 1997).

Calculation combinations

Case 1. Discharge at time t+1 is a function of rainfall and evaporation at time t
$$Q_{t+1} = f(R_t, E_t).$$
Case 2. Discharge at time t+1 is a function of rainfall and evaporation at time t, as well as of the past value of discharge at time t:
$$Q_{t+1} = f(R_t, E_t, Q_t)$$
Case 3. Discharge at time t+1 is a function of rainfalls and evaporation at times t and t-1, and of past values of discharge at times t, t-1, and t-2:
$$Q_{t+1} = f(R_t, E_t, R_{t-1}, E_{t-1}, Q_t, Q_{t-1}, Q_{t-2})$$

It should be noted that Case 1 was used in making the Tank Model applicable to daily flow forecasting by Phien and Danh (1997). Case 2 is similar to Case 1, except that the value of the discharge on the immediately preceding day is also included among the input factors. This is an obvious extension of Case 1, thanks to the flexibility of the BPNN models as compared to conceptual models such as the Tank Model. Case 3 corresponds to an extreme situation for a catchment area as small (about 775 km²) as that of the Da Nhim Basin: the effect of rainfall and evaporation on the discharge should not last longer than 2 d, while the persistence in the discharge itself can be adequately represented by its values in the last 3 d.

Discussions

- The architecture for each case is shown in Table 1. In fact, for all the cases, trial and error was used to determine the number of nodes in the hidden layer because the number of nodes in the input layer has been specified in the Cases considered : 2 for Case 1, 3 for Case 2, and 7 for Case 3. An appropriate architecture of a network was found when satisfactory values of the performance indices were obtained.
- During the training process, the learning rate (η) and the momentum (α) were varied between 0.1 to 1.0. It was found that the most appropriate values for both of them are 0.5.

The calculated performance statistics are shown in Table 2. In three cases, it was clear that Case 3 gives the best results. By examining the hydrographs showing the observed discharges and their forecast values, it was found that in Case 1, baseflow in the

TABLE 1
ARCHITECTURE OF BPNN MODELS FOR THE DA NHIM BASIN

Case	No. of nodes in input layer	No. of nodes in hidden layer	No. of nodes in output layer
1	2	2	1
2	3	2	1
3	7	4	1

TABLE 2
PERFORMANCE STATISTICS OF BPNN MODELS FOR THE DA NHIM BASIN

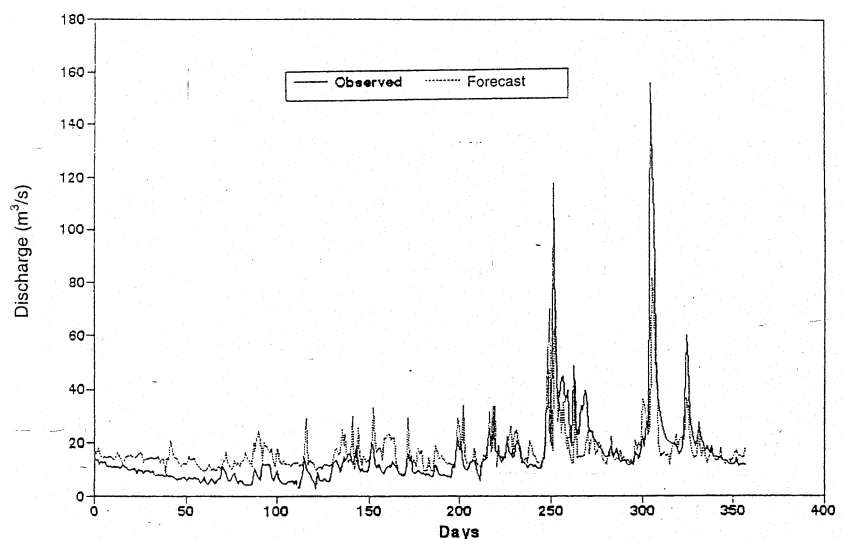| Case	Stage	EI	RMSE	RMSEM	MAD	RMSES
1	Calibration	0.67	14.72	0.80	8.78	0.59
	Validation	0.60	13.26	0.75	7.65	0.69
2	Calibration	0.74	13.07	0.71	4.20	0.53
	Validation	0.68	11.86	0.67	5.35	0.62
3	Calibration	0.83	10.56	0.57	2.83	0.43
	Validation	0.77	7.60	0.39	2.14	0.39

Figure 2
Comparison between observed and forecast discharges for Da Nhim, 1987 (Case 1)

low-flow period was over-estimated while flow during the high-flow period was under-estimated (Fig.2 for the year 1987, calibration). In Cases 2 and 3, an improvement of the forecast discharges has been achieved: values of both low-flow and peak periods are well matched, especially in Case 3 (Fig. 3 for Case 2 and Fig. 4 for Case 3, both for the year 1987).

As seen in Table 2, the efficiency index for Case 1 is equal to 0.67 for the calibration stage and 0.60 for the validation stage. Both are relatively smaller than the values obtained by the Tank Model (0.69 and 0.65, respectively), indicating that the perform-

ance of the BPNN model is a little worse than that of the Tank Model (see Table 4, Phien and Danh, 1997). Other performance indices indicate the same thing. However, BPNN models can readily be expanded to take into account other influencing factors as shown in Cases 2 and 3. Table 2 clearly shows that the resulting models are far better than the Tank Model. In fact, by removing both E_t and E_{t-1}, the efficiency index value reduces very slightly. These results show important contributions of past values of discharge toward its future values.

La Nga Basin (see Fig. 2, Phien and Danh, 1997)

The data on rainfall, evaporation and discharge are available from 1987 to 1994. The data in the first three years (1987-1989) were used for calibration and those in the last five years (1990-1994) were used for validation. It should be noted that the discharge station under consideration is Phu Dien with a catchment area of 3 060 km², which is much larger than that of the dam site station in the Da Nhim Basin.

Calculation combinations

Case 1. $Q_{t+1} = f([R_t]_{mean\ areal,}\ [E_t]_{mean\ areal})$.

Case 2. $Q_{t+1} = f([R_t]_{mean\ areal},\ [R_{t-1}]_{mean\ areal},\ [E_t]_{mean\ areal},\ [E_{t-1}]_{mean\ areal},\ Q_t,\ Q_{t-1},\ Q_{t-2})$.

Case 3. $Q_{t+1} = f([R_t]_{all\ stations},\ [E_t]_{mean\ areal},\ Q_t,\ Q_{t-1},\ Q_{t-2})$.

It should be noted that $[R_t]_{mean\ areal}$ denotes the mean areal rainfall calculated in Phien and Danh (1997) with the weights obtained by the method suggested by Sugawara (1961), while $[E_t]_{mean\ areal}$ is the arithmetic mean taken all over the five stations located within the La Nga Basin. In Case 3, $[R_t]_{all\ stations}$ means individual values of rainfall at each of the five stations, namely Bao Loc, Di Linh, Dai Nga, Phu Dien and Ta Pao (see Fig. 2, Phien and Danh, 1997).

As for the Da Nhim Basin, Case 1 is the same for the Tank Model. In Case 2, even though for this medium size catchment, the effect of rainfall and evaporation on discharge may last longer than two days, but it is taken as 2 d (to reduce the number of nodes in the input layer) while any remaining effect is assumed to be taken care of by the past values of the discharge itself in the last 3 d. In Case 3, due to a large number of input variables (being equal to 9), only the values at time t for rainfall and evaporation were used.

Discussion

For Case 1, the results obtained from the BPNN model are satisfactory (see Table 3) but a little

TABLE 3
PERFORMANCE STATISTICS OF THE BPNN MODEL FOR LA NGA BASIN

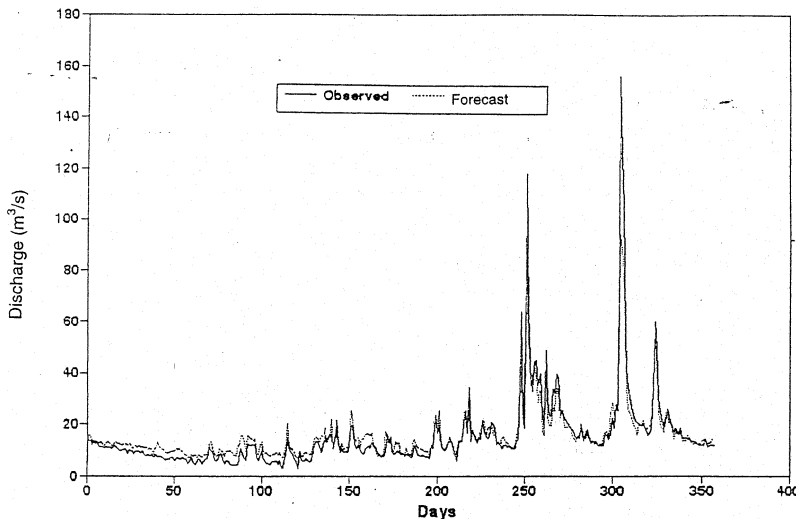| Case		EI	RMSE	RMSEM	MAD	RMSES
1	Calibration	0.91	43.13	0.34	26.42	0.29
	Verification	0.92	45.14	0.28	30.15	0.25
2	Calibration	0.99	16.28	0.10	9.63	0.09
	Verification	0.96	17.35	0.11	10.24	1.00
3	Calibration	0.97	32.89	0.20	22.30	0.18
	Verification	0.93	35.67	0.22	24.60	0.20

Figure 3
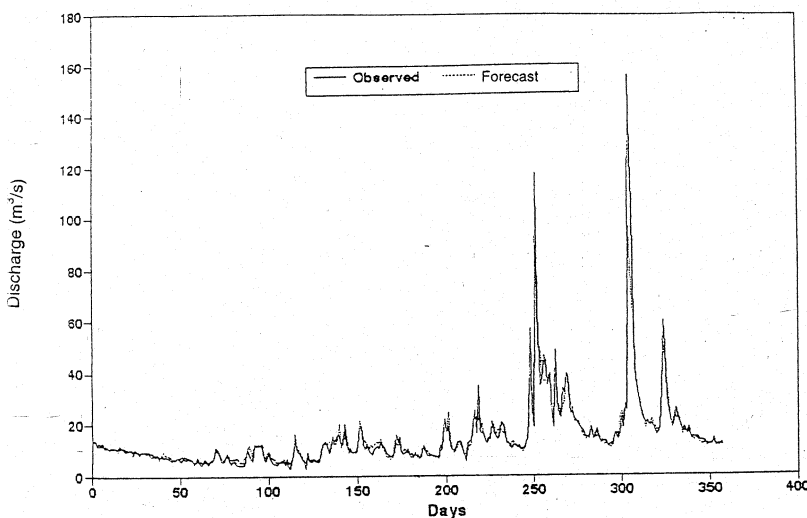*Comparison between observed and forecast discharges for Da Nhim, 1987 (Case 2)*

Figure 4
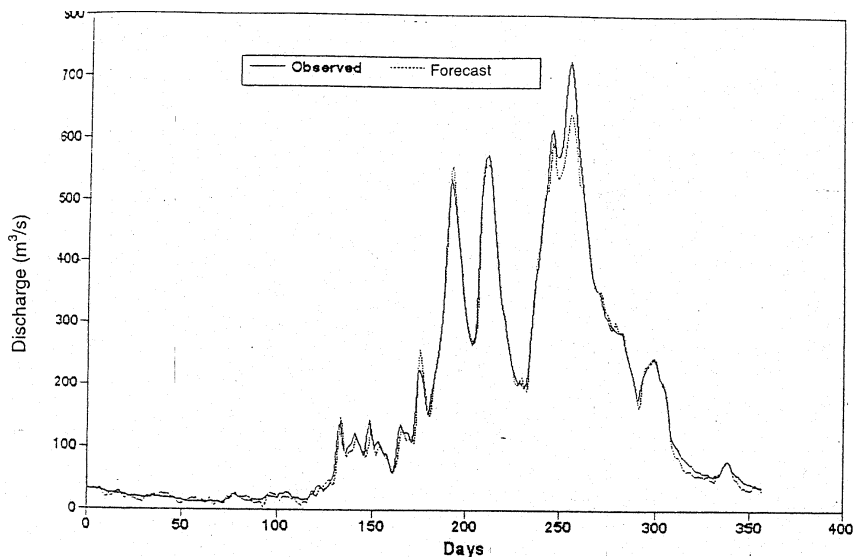*Comparison between observed and forecast discharges for Da Nhim, 1987 (Case 3)*

Figure 5
*Comparison between
observed and forecast
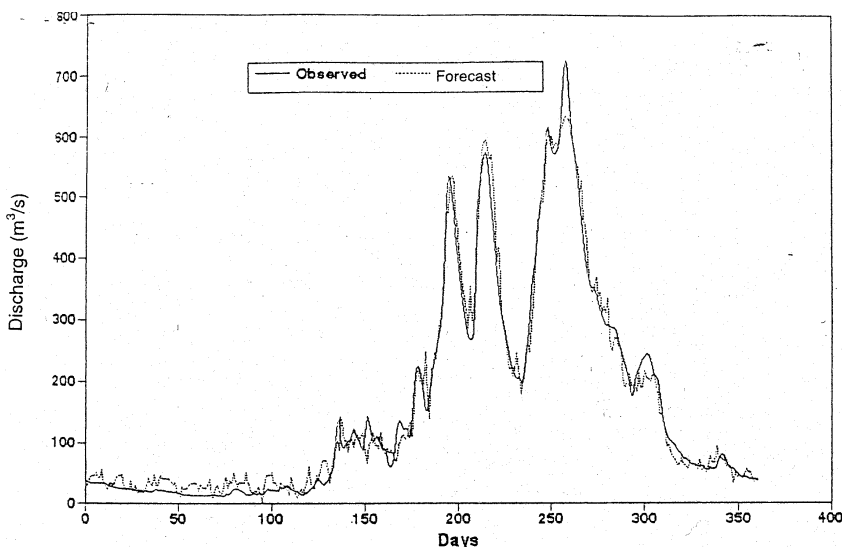discharges for
La Nga, 1994 (Case 2)*

Figure 5
*Comparison between
observed and forecast
discharges for
La Nga, 1994 (Case 3)*

worse than those obtained from the Tank Model. From these results as well as from those obtained for the Da Nhim Basin, it appears that as long as the same input variables were used, the BPNN models can perform almost as well as the Tank Model.

An important aspect related to BPNN models is that they can use as input data any meaningful combinations among the input variables, including their lagged variables. As such, Cases 2 and 3 deserve more attention.

From the hydrographs of the observed and forecast discharges, it was found that the discharges were very well forecast in Case 2. Both the low flows and high flows were accurately forecast, and the forecast hydrograph follows the observed hydrograph very closely. Typical results are illustrated in Fig. 5 for the year 1994 of the validation stage. In Case 3, the forecast discharge hydrograph does not match the observed hydrograph as well as in Case 2. Discharges were overestimated during the low-flow period. Only the peaks were forecast satisfactorily (see Fig. 6 for the year 1994). From the performance statistics and the hydrographs, one can say that Case 2 is better than Case 3. Moreover, since fewer input units are introduced to the model in Case 2, the resulting model runs faster. It should be noted that the RMSE for Case 2 is about 10% of the mean value and about 9%

of the standard deviation of the observed discharge, indicating that a very satisfactory forecasting model is obtained.

It was also found that as for the Da Nhim Basin, the removal of evaporation from the cases considered only affects the performance statistics slightly, indicating that the contribution of evaporation to the daily discharge in the La Nga Basin is negligible. In fact the evaporation was considered in this study because the Tank Model and many conceptual rainfall-runoff models include this variable as one of the important factors that produce the daily discharge at any station.

Conclusions

The results obtained in this paper indicate the capability of BPNN models in forecasting of daily river flows using daily evaporation and rainfall data as inputs. It should be noted that the forecasting results were found to be better for the La Nga Basin which is the larger basin of the two. This confirms the common belief that forecasting of discharges for larger basins is often more accurate than for smaller basins.

It was noted that the contribution of past values of discharge was very important in BPNN models, just as in regression models

(Phien et al., 1990). Perhaps this fact is the main reason for the wide acceptance of Box-Jenkins' ARIMA models which are based upon the linear relationship between successive observations of the discharge itself. For BPNN models, again as for regression models, mean areal values or individual values at all rainfall and evaporation stations within the basin under consideration can be used as inputs in forecasting the required discharges. However, while individual inputs are useful in determining the contribution of each factor, the results obtained in this work for the case of the La Nga Basin indicate that use of appropriately calculated mean values leads to better forecasting performance. Moreover, it was also found that the contribution of the evaporation is not important.

References

FAHLMAN SE and LEBIERE C (1990) The Cascade-Correlation Learning Architecture. Technical Report, School of Computer Science, Carnegie Mellon University, CMU-CS-90-100.

GERSHENFIELD NA and WEIGEND AS (1993) The Future of Time Series. Technical Report. Palo Alto Research Center.

GORR WL, NAGIN D and SZCZYPULA J (1994) Comparative study of artificial neural networks and statistical models for predicting student grade point average. *Int. J. Forecasting* **10** 17-34.

LACHTERMACHER G and FULLER JD (1995) Backpropagation in time series forecasting. *J. Forecasting* **14** 381-393

LIM SF and HONG SB (1993) Dynamic Creation of Hidden Units with Selective Pruning in Backpropagation. Technical Report, Dept. of Information Systems and Computer Science, National University of Singapore, No. TRD4/93.

MAIER HR and DANDY GC (1996) The use of artificial neural networks for the prediction of water quality parameters. *Water Resour. Res.* **32** (4) 1013-1022.

PHIEN HN and DANH NT (1997) A hybrid model for daily flow forecasting. *Water SA* **23** (3) 201-108.

PHIEN HN and PRADHAN PSS (1983) The Tank Model in rainfall-runoff modelling. *Water SA* **9** (3) 93-102.

PHIEN HN and SIANG JJ (1993) Forecasting monthly flows of the Mekong River using back propagation. *Proc. IASTED Int. Conf.* 17-20.

PHIEN HN, HUONG BK and LOI PD (1990) Daily flow forecasting with regression analysis. *Water SA* **16** (3) 179-184.

RUMELHART DE, HINTON GE and McCLELLAND JL (1996) A general framwork for parallel distributed processing. In: RUMELHART DE and McCLELLAND (eds.) *Parallel Distributed Processing; Explorations in the Microstructure of Cognition,* Vol. **1**, MIT Press, Cambridge.

SHAMSELDIN AY (1997) Application of a neural network technique to rainfall-runoff modelling. *J. Hydrol.* **199** (3-4) 272-294.

SUGAWARA M (1961) On the analysis of runoff structure about several Japanese rivers. *Japanese J. Geophys.* **2** (4) 1-76.

SUGAWARA M (1979) Automatic calibration of the Tank Model. *Hydrol. Sci. Bull.* **24** (3) 375-388.

WASSERMAN PD (1989*) Neural Computing Theory and Practice*. Van Nostrand Reinhold, New York. 1-59.