Flood forecasting for the upper reach of the Red River Basin, North Vietnam

Huynh Ngoc Phien* and Nguyen Duc Anh Kha
School of Advanced Technologies, Asian Institute of Technology, Bangkok, Thailand

Abstract

Flood forecasting remains a very important task. Good forecast values with sufficient lead times can help reduce flood damages significantly. This paper proposes two types of black-box model obtained by using multiple regression analysis and back-propagation neural networks in forecasting 6-h water levels at three important stations on the upstream section of the Red River basin, North Vietnam. The results obtained show that highly accurate forecast values can be obtained with lead times of up to 18 h by using two most recent past values of the water level at the station considered or two most recent past values at this station and two most recent values of an upstream station.

Keywords: water level forecasting, lead time, neural networks, back-propagation.

Introduction

Flood forecasting remains very important in any relief activities. In many cases, this task may rely on many different approaches. Due to the availability of data required, a daily basis has been widely adopted. This, in turn, leads to the forecasting of daily discharges at a number of selected stations - treated as reference stations - in the forecasting task, as well as in the alleviation of possible flood damages. In many cases, some conceptual rainfall-runoff models are used when these models are not really intended for forecasting purposes (Phien and Danh, 1997). These authors have made some slight changes to the way to treat input data to render these models suitable for forecasting. However, they may not perform as well as black-box models (Jain and Indurthy, 2003).

In the case of a number of stations in the Red River basin, North Vietnam (Fig.1), 6-h water level data are available during the flood season, from 1 June to 30 September. To deal with these cases, we propose the use of black-box models for the following reasons:

- Generally speaking, these models are data-driven: The model to be selected for use at a station should be based on the data available. For example, for conceptual models, data on evaporation (and several other factors) are required. Unfortunately, such data are not available (or at least unobtainable) to us. Therefor most conceptual models cannot be used.
- Most conceptual rainfall-runoff models normally produce values for discharge rather than for water levels. In order to obtain water-level values from such a model, the drainage area of, and the rating curve at the station concerned are required. Due to the inaccuracy in the rating curve to convert discharge values to water-level values (and vice versa), the forecast values obtained for the water level may be far from accurate.
- From the experience gained in previous studies (Phien et al., 1990; Danh et al., 1999; Phien and Sureerattanan, 1999), it was decided to make use of two general models, namely the multiple linear regression model and the back-propagation neural network model, for forecasting the water level at three stations, one on each main tributary of the Red River (Fig. 1):

- Ta Bu station on the Black (Da) River
- Yen Bai station on the Thao River, and
- Vu Quang station on the Lo River.

Located on the upper reach of the Red River basin, these stations do not have any tidal effect. As such, no tidal data are needed for forecasting purposes.

Multiple linear regression (MLR) model

The general forecasting equation based on MLR can be written as follows:

$$H_{t+\tau} = A + \sum_{j=1}^{m_1} a_j H_{t-j+1} + \sum_{j=1}^{m_2} b_j U_{t-j+1} + \sum_{j=1}^{m_3} c_j R_{t-j+1} + \dots \quad (1)$$

where:

H : water level at the station under consideration
U : water level at the upstream station(s)
R : rainfall
A, a_j, b_j, c_j : regression coefficients
τ : forecast lead time.

Using the least squares method, the regression coefficients can readily be obtained.

Back-propagation (BP) neural network model

This is a multilayer feed forward neural network with a back-propagation algorithm used for updating its weights. The BP neural network model has been extensively used in many applications, including forecasting problems.

- Atiya et al. (1999) applied BP neural networks to forecast the flow of the Nile in Egypt with fairly good results.
- Maier and Dandy (2000), in their survey, found that out of 43 papers dealing with the use of neural networks for forecasting of water resources variables, 41 papers employed BP models.
- Hsieh et al. (2001) applied the BP neural network model for flood forecasting for two different scale watersheds, namely the Sala River in Croatia and a segment of the Mississippi River, USA. They found that good downstream river-flow forecasts could be obtained from upstream gauges for the Sala

* To whom all correspondence should be addressed.
☎ (66-2) 524-5731; fax: (66-2) 524-5745; e-mail: hnp@cs.ait.ac.th

Received 15 January 2003; accepted in revised form 27 March 2003.

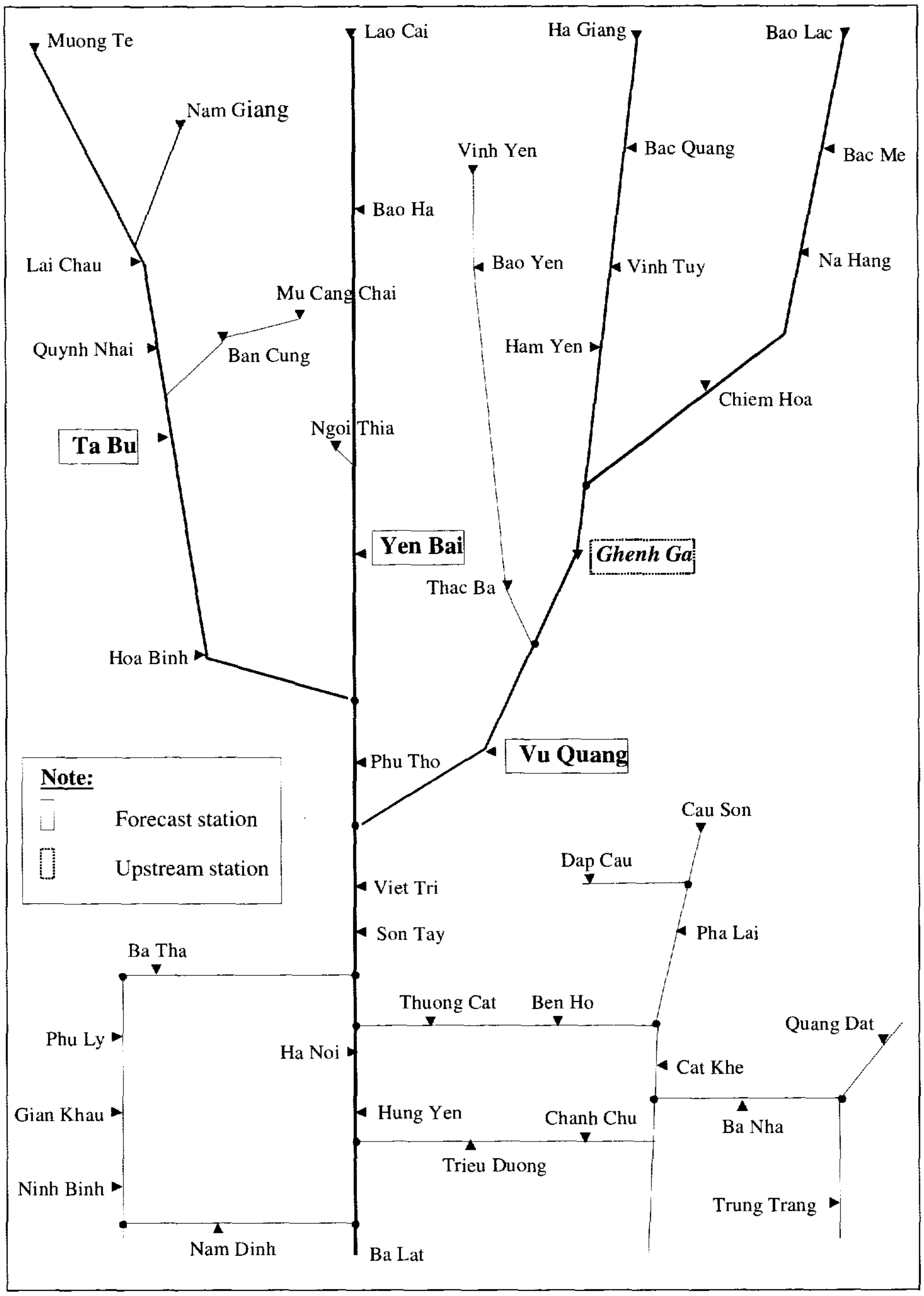


Figure 1 (left)
Map of stations on the Red River system

River, and very good river-flow forecasts from two upstream stations, without the need to use of rainfall data, for the Mississippi River.

Other applications of the BP neural network model can be found in Jain et al. (2001), and Koussis et al. (2003).

The structure of a BP model for the problem at hand is shown in Fig. 2 with the input layer, one hidden layer, and the output layer with only one node. The activation function adopted is the sigmoid function:

$$x_{jk} = f(y_{jk}, T_{jk}) = \frac{1}{1+\exp\left(-\frac{y_{jk}}{T_{jk}}\right)} \qquad (2)$$

$$y_{jk} = \sum_{i=1}^{N_{j-1}} w_{jki}\, x_{j-1,i} + \theta_{jk}$$

$$= \sum_{i=0}^{N_{j-1}} w_{jki}\, x_{j-1,i} \qquad (3)$$

where:

N_j : number of nodes in the jth layer

w_{jki} : weight from node k of layer j and node i of layer (j-1)

θ_{jk} : bias at node k of layer j

T_{jk} : temperature at node k of layer j of the sigmoid function

It should be noted that the temperature term is present in Eq. (2) because of the involvement of the training algorithm developed by Phien and Sureerattanan (1999), which was found to perform very satisfactorily. This algorithm can be summarised as follows:

1. Randomise all weights, biases and temperatures of the sigmoid function and set the initial value to the inverse matrix R^{-1} where R is the correlation matrix of the training set.
2. Present a training pattern pair p (x_{p0}, o_p) to the network.
3. For each layer from the first hidden layer to the output layer:
   (a) Calculate the model output x_{jk} for every node k in layer j, following Eqs. (2) and (3).

   (b) Calculate the Kalman gain k_j and update the inverse matrix R_j^{-1} for each layer j from the first hidden layer through output layer L by the following expressions:

$$k_j = \frac{R_j^{-1}\, x_{j-1}}{b_j + x_{j-1}^T\, R_j^{-1}\, x_{j-1}} \qquad (4)$$

$$R_j^{-1} = \left(R_j^{-1} - k_j\, x_{j-1}^T\, R_j^{-1}\right) b_j^{-1} \qquad (5)$$

where:

b_j is the forgetting factor with values between the [0,1] interval.

(c) Calculate the error signal e_{Lk} for the weights of the output layer L and e_{jk} for that of the hidden layers:

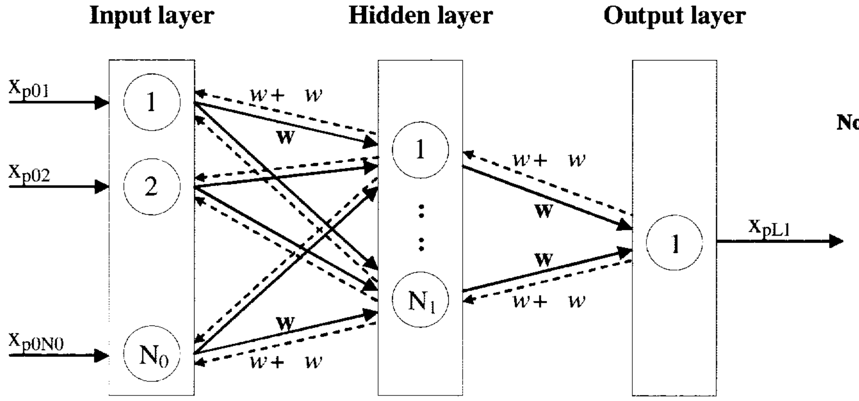


Figure 2
Sketch of the back-propagation model

$$e_{Lk} = x_{Lk}(1 - x_{Lk})(o_k - x_{Lk}) \tag{6}$$

$$e_{jk} = x_{jk}(1 - x_{jk})\sum_{l} e_{j+1,l}\, w_{j+1,l,k} \tag{7}$$

(d) Calculate the actual pre-image output at the output layer:

$$d_k = f^{-1}(o_k) = T_k \ln\left(\frac{o_k}{1 - o_k}\right) \tag{8}$$

(e) Calculate the weight change Δw_{jk} for each layer j from 1 to L:

$$\Delta w_{jk}(t+1) = \begin{cases} k_L\,(d_k - y_{Lk})\lambda_j + \alpha\Delta w_{Lk}(t) & \text{when } j = L \\ k_j\, e_{jk}\, \lambda_j + \alpha\Delta w_{jk}(t) & \text{when } j \neq L \end{cases} \tag{9}$$

where:

λ_j is the learning rate in the j^th layer
α is the momentum coefficient
t is the present iteration number.

(f) Calculate the error signal δ_{Lk} for the temperature of the output layer and δ_{jk} for that of the hidden layers:

$$\delta_{jk} = \begin{cases} o_k - x_{Lk} & \textit{when } j = L \\ \sum_{l} \delta_{j+1,l}\, \dfrac{1}{T_{j+1,l}}\, \dfrac{e\left(-\dfrac{y_{j+1,l}}{T_{j+1,l}}\right)}{\left[1 + e\left(-\dfrac{y_{j+1,l}}{T_{j+1,l}}\right)\right]^2}\, w_{j+1,l,k} & \textit{when } j \neq L \end{cases} \tag{10}$$

$$\Delta T_{jk}(t+1) = -\mu\, \delta_{jk}\, \frac{y_{jk}}{T_{jk}^2}\, \frac{e\left(-\dfrac{y_{jk}}{T_{jk}}\right)}{\left[1 + e\left(-\dfrac{y_{jk}}{T_{jk}}\right)\right]^2} + \gamma\, \Delta T_{jk}(t) \tag{11}$$

(g) Update the weights w_{jk} and the temperature of the sigmoid function T_{jk}

$$w_{jk}(t+1) = w_{jk}(t) + \Delta w_{jk}(t+1) \tag{12}$$

$$T_{jk}(t+1) = T_{jk}(t) + \Delta T_{jk}(t+1) \tag{13}$$

4. Repeat from Steps 2 to 3 for all training patterns until the system error (sum of squared errors) reaches its minimum.

The number of hidden nodes of the required back-propagation model for each case was determined by using the Bayesian information criteria (BIC) as proposed by Phien and Sureerattanan (1999).

Performance statistics

The performance of a model can be measured by the root mean square error (RMSE), efficiency index (EI), mean absolute deviation (MAD) and maximum relative error (Remax).

- **Root mean square error (RMSE)**

$$\text{RMSE} = \sqrt{\frac{\text{SSE}}{\text{N}}} \tag{14}$$

where:
SSE is the sum of squared errors and N is the number of data points used:

$$\text{SSE} = \sum_{i=1}^{N}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{N} e_i^2 \tag{15}$$

where:
y_i and $\hat{y}_i$ are respectively the observed and computed (from the model under consideration) values of y, and e_i is the error, being the difference between the observed and computed values.

- **Efficiency index (EI)**

This index, introduced by Nash and Sutcliffee (1970), is defined as:

$$\text{EI} = \frac{\text{ST} - \text{SSE}}{\text{ST}} \tag{16}$$

$$\text{ST} = \sum_{i=1}^{N}(y_i - \bar{y})^2 \tag{17}$$

$$\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i \tag{18}$$

where:
ST is the total variation and $\bar{y}$ is the mean value taken over N values of y.

- **Mean absolute deviation (MAD)**

MAD is the average of the absolute values of the differences between model output and observed values of the output:

$$\text{MAD} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{19}$$

To give a better indication of the magnitude of the error, the ratio between MAD and the mean $\bar{y}$ should be used. It is denoted as RAD:

$$RAD = \frac{MAD}{\bar{y}} \qquad (20)$$

- **Maximum relative error (Remax)**

$$Remax = Max\left(\left|\frac{y_i - \hat{y}_i}{y_i}\right|\right) \text{ where } i = 1,2,\ldots, N \qquad (21)$$

As there is a close relationship between the root mean squared error and the efficiency index, the latter is used in summarising the results because it can indicate the performance of the model being attempted, along with RAD and Remax.

Data employed

In this study, the 6-h data on the rainfall and water level are used. After several runs, it was found that incorporation of rainfall data did not improve any models concerned. So finally, the data employed at the selected stations are as follows:

- Ta Bu Station: 6-h data on water level (1989-1996)
- Yen Bai Station: 6-h data on water level (1989-1996)
- Vu Quang Station: 6-h data on water level for 1987-1996 at Vu Quang and Ghenh Ga, an upstream station of Vu Quang.

Each set of data was split into two parts: one for model calibration (training) and the other for model validation (testing), as shown in Table 1.

Results

Ta Bu Station

The results for both models are collected in Tables 2a and 2b for the training and testing stages, respectively. The values of the efficiency index are quite high (more than 0.90) showing that the models obtained by multiple linear regression (MLR) and back-propagation networks (BP) perform very satisfactorily for lead times of up to 3 time units, i.e. 18 h.

By inspecting the values of RAD and Remax, it is clear that the models are good:

- the average absolute deviation is about 0.5% of the mean value of the water level;
- the maximum relative error is within 20% (even for the testing stage).

Figure 3 shows that there is a good match between the observed and forecast water level for the case with forecast lead times of 3 units (or 18 h).

Yen Bai Station

The results are shown in Tables 3a and 3b for the training and testing stages, respectively. They are not as good as those for Ta Bu. However, all the performance statistics, as well as Fig.4, indicate that the models are very good.

Vu Quang Station

The results of this station are shown in Tables 4a and 4b, for the training and testing stages respectively. A graphical comparison between the forecast and observed values of the water level is shown in Fig. 5 for lead time equal to 18 h. Again the resulting models performed excellently.

TABLE 1
Data for training and testing stage

Station	Training	Testing
Ta Bu	1989-1993	1994-1996
Yen Bai	1989-1993	1994-1996
Vu Quang	1987-1992	1994-1996
Ghenh Ga	1987-1992	1994-1996

TABLE 2a
Results at Ta Bu (training stage)

τ	MLR			BP			
	EI	RAD	Remax	EI	RAD	Remax	Structure (*)
1	0.99	0.002	0.037	0.98	0.002	0.037	2-1-1
2	0.96	0.004	0.042	0.95	0.004	0.043	2-1-1
3	0.92	0.005	0.052	0.92	0.005	0.053	2-1-1

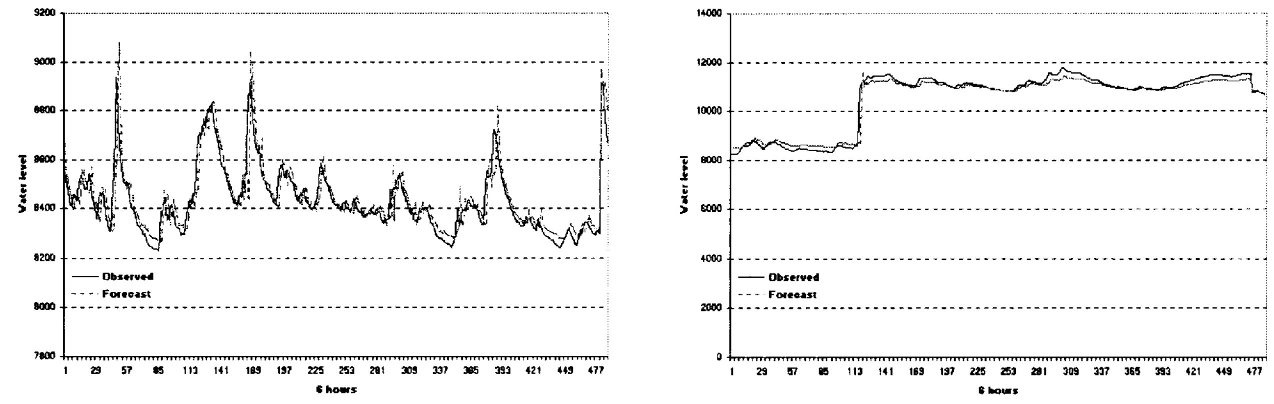
(*) 2-1-1 denotes a network with 2 input nodes, 1 hidden node and 1 output node.

TABLE 2b
Results at Ta Bu (testing stage)

τ	MLR			BP		
	EI	RAD	Remax	EI	RAD	Remax
1	0.99	0.003	0.120	0.99	0.010	0.111
2	0.99	0.007	0.217	0.99	0.010	0.207
3	0.98	0.011	0.221	0.98	0.013	0.209

Discussion

- For two stations, namely Ta Bu and Yen Bai, only the past values of the water level are used in forecasting future values. This leads to a very simple structure of the back-propagation neural network required: two input nodes and one hidden node. The fact that there are only two input nodes shows that only the two most recent values can be used to forecast the following three values. Even with this simple structure, the BP models obtained perform very well.
- For Vu Quang, due to the involvement of the upstream station (Ghenh Ga), the BP model appears a bit more complicated. We need four input nodes and three hidden nodes. Future water-level values can be forecast quite accurately with the use of only the two most recent values.
- All the MLR models obtained require the same two most recent values, as do the BP models. Inspecting all the results shown in the above tables reveals that the MLR models perform slightly better than the BP models. As the MLR models are much simpler than the BP models, this means that a simpler model may perform better than a more sophisticated model.

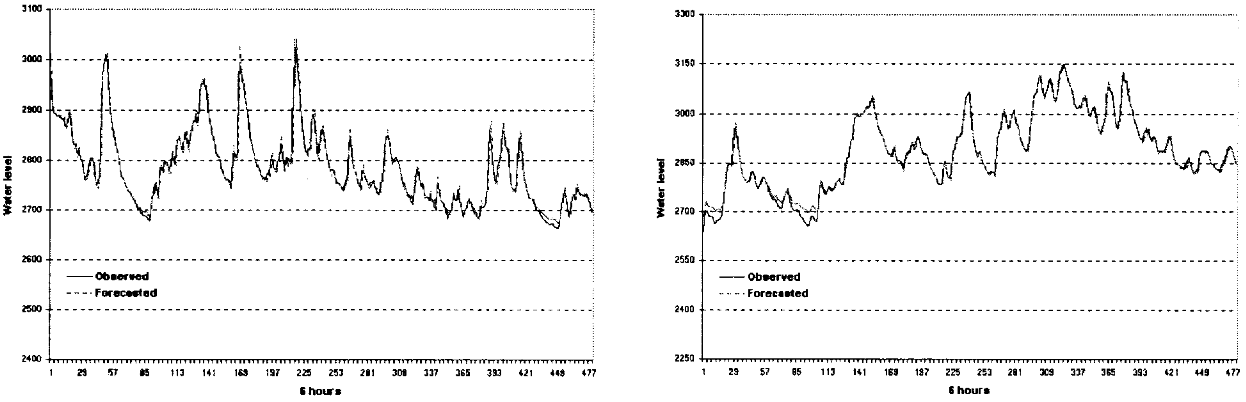


Calibration period

Validation period

Figure 3

Comparison between forecast and observed water level at Ta Bu Station (lead time = 18 h)



Calibration period

Validation period

Figure 4

Comparison between forecast and observed water level at Yen Bai Station (lead time = 6 h)

TABLE 3a
Results at Yen Bai (training stage)

τ	MLR			BP			
	EI	RAD	Remax	EI	RAD	Remax	Structure
1	0.97	0.004	0.050	0.97	0.004	0.052	2-1-1
2	0.92	0.007	0.062	0.91	0.008	0.063	2-1-1
3	0.85	0.010	0.074	0.85	0.010	0.075	2-1-1

TABLE 3b
Results at Yen Bai (testing stage)

τ	MLR			BP		
	EI	RAD	Remax	EI	RAD	Remax
1	0.98	0.003	0.049	0.98	0.004	0.052
2	0.93	0.007	0.066	0.92	0.008	0.062
3	0.85	0.010	0.090	0.85	0.011	0.085

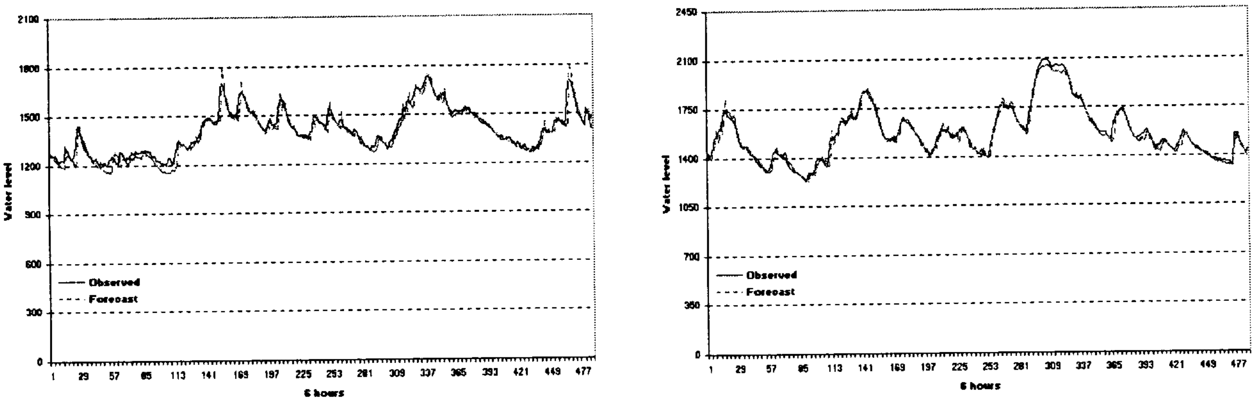
TABLE 4a
Results at Vu Quang (training stage)

τ	MLR			BP			
	EI	RAD	Remax	EI	RAD	Remax	Structure (*)
1	0.99	0.005	0.068	0.99	0.008	0.076	4-3-1
2	0.99	0.010	0.098	0.98	0.012	0.103	4-3-1
3	0.97	0.015	0.137	0.96	0.016	0.141	4-3-1

(*) 4-3-1 denotes a network with 4 input nodes, 3 hidden nodes and 1 output node

TABLE 4b
Results at Vu Quang (testing stage)

τ	MLR			BP		
	EI	RAD	Remax	EI	RAD	Remax
1	0.99	0.004	0.030	0.99	0.008	0.052
2	0.99	0.008	0.071	0.98	0.011	0.087
3	0.97	0.013	0.113	0.96	0.016	0.167



Calibration period

Validation period

Figure 5
Comparison between forecast and observed water level at Vu Quang Station (lead time = 18 h)

- As expected, when the lead time increases (from one to three time units, i.e. from 6 h to 18 h), the performance of all the models decreases.
- It should be noted that the drainage area of Ta Bu or Yen Bai is much larger than that of Vu Quang. As such, the water level values at Ta Bu and Yen Bai do not fluctuate as much as those at Vu Quang. This may be the reason for the need to include the information contained in the water-level data at an upstream station like Ghenh Ga in order to arrive at a good model for forecasting the water level at Vu Quang.
- In view of the tabulated values of the performance statistics, a model may perform better in the testing stage than in the training stage. This happens purely by chance: The data set used for testing happens to be more appropriate for that model. In general, the performance in the testing stage should be less satisfactory than that in the training stage.

Acknowledgements

The paper is a result of the research project entitled FLOod COntrol Decision Support (FLOCODS) system sponsored by the Commission of the European Communities. The authors wish to thank the Commission for its financial support, and the Institute of Mechanics, Hanoi, Vietnam for the data used. Sincere appreciations are extended to the two reviewers whose comments have greatly helped in improving the presentation of the paper.

Conclusions

From the results obtained and the discussions presented above, the following conclusions can be drawn for the case of the three stations considered:

- For forecasting the water level at a station, only two most recent values of the water level at that station alone, or two most recent values at it and at an upstream station are required.
- For large catchment areas such as those of the stations considered in this study, a simple model obtained by multiple regression analysis can do an excellent forecasting task of up to 3 time units, which is the lead time needed for most flood mitigation activities.
- A more sophisticated model such as the back-propagation neural network can also perform very well. However, it does not necessarily follow that a more sophisticated model can do better. In fact, for all the three stations, the models based on multiple regression analysis perform slightly better than those based on neural networks.

References

ATIYA AF, EL-SHOURA SM, SHAHEEN SI and EL-SHERIF MS (1999) A comparison between neural-network forecasting techniques - Case study: River flow forecasting. *IEEE Trans. on Neural Networks* **10** (2) 436-461.

DANH NT, PHIEN HN and DAS GUPTA A (1999) Neural network models for river flow forecasting. *Water SA* **25** (1) 33-39.

HSIEH BB, BARTOS CL and ZHANG B (2001) Use of artificial neural networks in a streamflow prediction system. US Army Engineers Research and Development (USAE R&D) Center, Vicksburg, USA.

URL:http://www.nd.com/public/appsum/app-predict.doc

JAIN A and INDURTHY SKVP (2003) Comparative analysis of event-based rainfall-runoff modeling techniques - Deterministic, statistical, and artificial neural networks. *J. Hydrol. Eng. (*ASCE*)* **8** (2) 93-98.

JAIN A, VARSHNEY AK and JOSHI UC (2001) Short-term water demand forecast modeling at IIT Kanpur using artificial neural networks. *Water Resour. Manage.* **15** (5) 299-321.

KOUSSIS AD, LAGOUVARDOS K, MAZI K, KOLTRONI V, SITZMANN D, LANG J, ZAISS H, BUZZI A and MALGUZZI P (2003) Flood forecasts for urban basin with integrated hydro-meteorological model. *J. Hydrol. Eng. (ASCE)***8** (1) 1-11.

MAIER HR and DANDY GC (2000) Neural networks for the prediction and forecasting of water resources variables: A review of modeling issues and applications. *Environ. Model. & Software* **15** 101-124.

NASH JE and SUTCLIFFE JV (1970) River flow forecasting through conceptual models. *J. Hydrol.* **10** 282-290.

PHIEN HN and DANH NT (1997) A hybrid model for daily flow forecasting. *Water SA* **23** (3) 201-208.

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

PHIEN HN and SUREERATTANAN S (1999) Neural networks for filtering and forecasting of daily and monthly streamflows. In: Singh VP, Seo IW and Sonu JH (eds.) *Hydrologic Modeling.* Water Resources Publications LLC, Colorado, USA.