Int. J. Advance. Soft Comput. Appl., Vol. 3, No. 1, March 2011
ISSN 2074-8523; Copyright © ICSRS Publication, 2011
www.i-csrs.org

Hourly Runoff Forecast at Different Lead-time for a Small Watershed using Artificial Neural Networks

Kuok King Kuok[1*], Sobri Harun[2] and Chiu Po Chan[3]

[1] Lecturer, School of Engineering, Computing and Science, Swinburne University of Technology Sarawak Campus, Jln Simpang Tiga, 93350 Kuching, Sarawak, Malaysia
e-mail:kelvinkuok100@gmail.com
[2] Associate Professor, Department of Hydraulics and Hydrology, Faculty of Civil Engineering, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia
e-mail: sobriharun@gmail.com
[3] Lecturer, Department of Information System, Faculty of Computer Science and Information Technology, University Malaysia Sarawak, 94300 Samarahan, Sarawak Malaysia
chiupochan@yahoo.com

Abstract

Rainfall-runoff relationships are among the most complex hydrologic phenomena. The conceptual models developed by Hydrologists for simulating runoff composed of a large number of parameters and the interactions are highly complicated. The accuracy of conceptual model simulation is very subjective and highly depends on the modeler's ability and understanding of the model. Therefore, ANNs is applied to model rainfall-runoff. ANN is an information-processing system composed of many nonlinear and densely interconnected neurons. It is able to extract the relation between the inputs and outputs of a process without the physics being provided to them. Natural behavior of hydrological processes is appropriate for the application ANN in hydrology. In this paper, MLP and REC networks are adopted to forecast the hourly runoff of Sungai Bedup Basin, Sarawak. Inputs data used are antecedent rainfall, antecedent runoff and current rainfall while output is the current runoff. ANNs were trained with different training algorithms, learning rates, number of hidden neurons and antecedent hours. Results are evaluated using Coefficient of Correlation (R), Nash-Sutcliffe Coefficient (E^2) and peak error. To ensure the reliability and robustness of the optimal configuration

obtained, MLP and REC networks will be further validated with six separate storm events at different lead time. Results show the performance of REC is slightly better than MLP. However, both networks are able to simulate hourly runoff with high accuracy. Therefore, both networks can be utilized as early warning flow forecaster to take necessary flood protection measures before a severe flood occurs.

Keywords: *Artificial Neural Networks, Multilayer Perceptron Neural Networks, Recurrent Neural Networks, Flood Forecasting.*

1 Introduction

Rainfall-runoff is the most complex hydrologic phenomena to comprehend due to the tremendous spatial and temporal variability of watershed characteristics and precipitation patterns. In the past, conceptual models are adopted to formulate the physical process of rainfall-runoff. However, these conceptual models composed of a large number of parameters and the interactions among these parameters are highly complicated. Thus, the accuracy of conceptual model simulation is very subjective and dependent on the user's ability and understanding of the model.

In recent years, artificial neural networks (ANNs) have been discovered to be a powerful tool for solving different problems in variety of applications. The nature behavior of complex hydrologic phenomena is appropriate for the application of ANNs. In particular, a well known method of supervised learning of neural networks called backpropagation neural network (BPNN), is useful for handling large volume of real-time, non-stationary and non-linear natural phenomena (Nishimura and Kojiri, 1996). The most recent applications of neural networks are Nurmaini et al. (2009) applied weightless neural network (WNN), also called n-tuple networks or RAM based networks, for recognizing and classifying the environment in mobile robot through a simple microprocessor system. Idris et al. (2009) employed self organizing map (SOM) and backpropagation (BP) algorithm to discover the connection between the domain concepts contained in the learning object and the learner's learning need. Sivasankari and Thanushkodi (2009) utilized backpropagation neural network for detecting the presence of epileptic seizures in EEG signals automatically. Hemanth et al. (2010) proposed modified counter propagation neural network (CPN) for classifying the abnormal magnetic resonance (MR) brain image. Osman et al. (2010) developed neural network enzyme classification (NNEC) to classify an enzyme found in protein data bank (PDB) to a given family of enzymes.Nowadays, ANNs are widely used as an efficient tool in different areas of water engineering. These include modeling of rainfall-runoff relationship (Elshorbagy *et al.*, 2000); inflow estimation (Harun *et al.*, 1996); runoff analysis in humid forest catchment (Gautam *et al.*, 2000); river flow prediction (Imrie *et al.*, 2000, Dastorani and Wright 2001); setting up stage-discharge relations (Jain and Chalisgaonkar 2000); ungauged catchment flood prediction (Wright and Dastorani 2001) and short term river flood forecasting (Garcia-Bartual 2002).

The current study presents the development of real-time flood forecasting models using artificial neural networks (ANNs) with the input of rainfall and runoff data only. Two types of neural networks, named as Multilayer Perceptron (MLP) network and Recurrent (REC) network are applied to simulate hourly runoff. The ANNs were trained and tested with different training algorithm, different number of hidden neurons, different learning rate and different number of antecedent hours in order to select the best performance ANNs. Subsequently, the optimal configuration of ANNs model obtained will be adopted to forecast the runoff at 3, 6, 12 and 18 hours ahead for Bedup basin.

2 Neural Network Structure

Hydrological models were divided into three categories based on the consideration of the hydrological processes. These three categories are physically based distributed models, lumped conceptual models and black box models (Dooge, 1974). Neural Network model, which inherently involves mapping of input and output vectors, is classified as a black box model. Such black box model has little significance in enhancing the understanding of hydrological and hydraulic processes. Nevertheless, their usefulness can be paramount in operational hydrology.

Basically, ANNs is an information-processing system composed of many nonlinear and densely interconnected processing elements or neurons. Each neuron is linked with its neighbors with an associated weight that represent information used by the net to solve a problem. Neurons arranged in groups called layers and operated in logical parallelism. Information is transmitted from one layer to others in serial operations. Three basic layers of ANNs are input layer, hidden layer and output layer. A three layer feedforward neural network model is shown in Fig. 1.

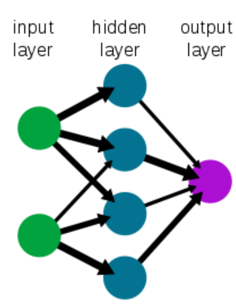


Fig.1: A Simple Feedforward Neural Network Model.

In recent years, several neural network algorithms had been developed. Backpropagation algorithm that performs a gradient descent search in weights space using generalized delta rule is often used in applications (Minn and Halls, 1996). This gradient descent technique will minimize the network error function and it is embedded in MLP and REC networks. BPNN were developed by Rumelhart and McClelland (1988) for learning associations between input and output patterns using more than a single layer perception, which overcomes some limitations of a single-layer perception (no hidden layer). The learning process of BPNN is comparing the actual output with the target output and then readjusts the weights in the backward direction. In the next iteration, the same input is presented to the network so the actual output will be closer to the target output.

2.1 Multilayer Perceptron Neural Network (MLP)

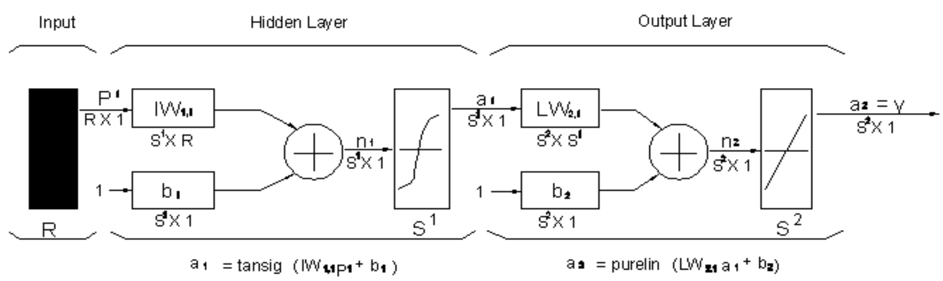


Fig. 2: MLP Network Architecture.

MLP network is a two-layer feedforward network trained with backpropagation learning algorithm (Fig. 2). The transfer function used in hidden layer is tan-sigmoid (*tansig*) and linear transfer function (*purelin*) at the output layer. Number of hidden neurons was determined through training. A preliminary study was conducted to select the suitable training algorithms to apply in this research. After trying various types of training algorithm, three different variants of backpropagation algorithms were identified for further investigation. These three variants of backpropagation algorithms are:

a) Scaled Conjugate Gradient (TRAINSCG). TRAINSCG was designed to avoid the time consuming line search, which produces generally faster convergence than the steepest descent directions used by the basic backpropagation.

b) Variable Learning Rate Backpropagation (TRAINGDX). TRAINGDX allows the learning rate to change during training process and attempt to keep the learning step size as large as possible while keeping learning stable. This increases the learning rate without increases of large error.

c) Powell-Beale Restarts (TRAINCGB). TRAINCGB will restart if there is very little orthogonality left between the current gradient and the previous gradient and the search direction is reset to the negative of the gradient. At each iteration, the step size is adjusted.

The neurons in each layer are connected to the neurons in the subsequent layer by a weight, which were adjusted during training. Fig. 2 shows the input vector p_1 was propagated towards hidden layer by multiplying with input weight matrices $IW_{1,1}$ to form $IW_{1,1}p_1$. Then the sum of bias b_1 and the product of $IW_{1,1}p_1$ propagated forward to *tansig* transfer function. This sum was passed through *tansig* transfer function to get the hidden neurons's output a_1, where $a_1 = tansig(IW_{1,1}p_1+b_1)$.

Similarly, the output a_1 from hidden layer is propagated forward through the network towards output layer. At output layer, a_1 is then multiplied with weight matrices in layer outputs that called layer weights, $LW_{2,1}$ to form $LW_{2,1}\ a_1$. Then, the sum of bias b_2 and product $LW_{2,1}\ a_1$ will transform through *purelin* transfer function to get the neurons's output a_2, where $a_2 = purelin(IW_{2,1}a_1+b_2)$.

Thereafter, feedback iteration calculated error signals that propagated backwards through the network and used to adjust the weights. The weights of the output layer are adjusted first. Then, adjustments are made for interconnection weights between each layer, based on the computer error and learning rate parameter.

2.2 Recurrent (REC) Network

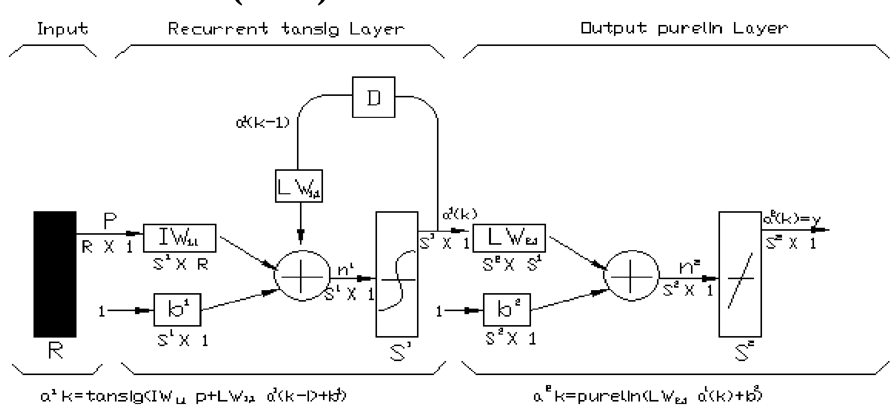


Fig. 3: REC Network Architecture.

The type of REC networks used is Elman network (Fig. 3). Elman networks composed two-layer backpropagation networks with addition feedback connection from the output of the hidden layer to its input. This feedback path allows Elman networks learn to recognize, generate temporal patterns and spatial patterns, stores values from the previous time step and use them in the current time step. This makes REC networks useful in areas such prediction where time plays an important role.

Elman network used *tansig* transfer function in hidden (recurrent) layer and *purelin* neurons in output layer. Similar with MLP network, the backpropagation training functions investigated for Elman network are TRAINSCG, TRAINGDX and TRAINCGB.

Elman networks must have enough hidden neurons to divide the input space in a useful way. It will perform better when there are more hidden neurons than actually required. When fewer neurons, Elman network is less able to find the most appropriate weights for hidden neurons since the error gradient is approximated.

3 Study Area

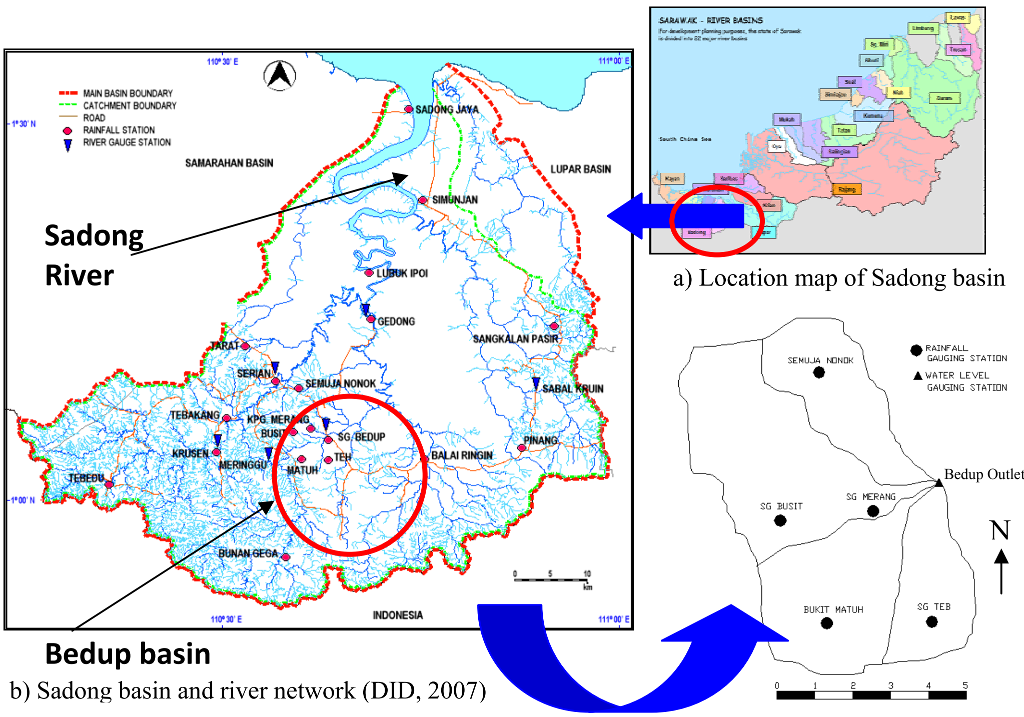
The selected study area is Bedup basin. The catchment area of Bedup basin, which is located approximately 80km from Kuching City, Sarawak, Malaysia is about 47.5km^2. Bedup basin is mainly covered with shrubs, low plant and forest. The elevation are varies from 8m to 686m above mean sea level (JUPEM, 1975). There is no significant land use change in the past 30 years. The length of Bedup river is approximately 10km. Main soil type of Bedup basin is clayey soils and part of it is covered with coarse loamy soil.

Bedup river is located at upper stream of Batang Sadong, where the tide is not reachable. Rating curve equation for Bedup basin is represented by Equation 1 (DID, 2007).

$$Q=9.19(H)^{1.9} \quad (1)$$

where Q is the discharge (m^3/s) and H is the stage discharge (m). These observed runoff data were used to compare the model runoff.

Fig.4 shows the location of Bedup basin. Locality of Sadong basin in Sarawak is presented in Fig. 4a. Fig. 4b shows the boundary of Sadong basin, location of rainfall and water level gauging stations that were installed by Department of Irrigation and Drainage (DID) Sarawak. Fig. 4c presents the 5 rainfall gauging stations installed within Bedup basin named as Bukit Matuh (BM), Semuja Nonok (SN), Sungai Busit (SB), Sungai Merang (SM) and Sungai Teb (ST), and one river stage gauging station at Sungai Bedup.



a) Location map of Sadong basin

b) Sadong basin and river network (DID, 2007)

c) Bedup basin

Fig. 4: Locality Map of Bedup basin, Sub-basin of Sadong basin, Sarawak

For calibration and validation purposes, input data fed into ANNs model are current hourly rainfall data, antecedent hourly rainfall data and antecedent hourly runoff data. The output of ANNs model is current hourly runoff. The calibrated ANNs models will then carry out computation to simulate the hydrograph at Bedup outlet at 3, 6, 12 and 18 ahead runoff forecast.

4 Model Development

MLP and REC networks are applied to train and test the hourly rainfall-runoff model. The performance of MLP and REC are evaluated from four main perspectives including:

a) Different types of training algorithms
b) Different number of antecedent hours.
c) Different number of hidden neurons
d) Different learning rate values

Five models were developed for investigating the effect of the number of antecedent hours on the performance of MLP and REC. The input data for this particular hourly rainfall-runoff model are antecedent hour precipitation *{P(t-1)………P(t-n)},* antecedent hour discharges *{Q(t-1)………Q(t-n)}* and the rainfall of the current hour*{P(t)}*. The output data is the hourly runoff *{Q(t)}*, as given by Equation 2..

$$Q(t)=f\{P(t),P(t-1),P(t-2), P(t-3)\ldots P(t-n),Q(t-1),Q(t-2),Q(t-3)\ldots Q(t-n)\} \quad (2)$$

The sequences of input arrangement are arranged in order since the time is important factor in the model development. Therefore, the configurations of five MLP and REC models with different number of antecedent hours are listed below:

a) $Q(t)=f\{P(t), P(t-1), Q(t-1)\}.$ (H1)

b) $Q(t)=f\{P(t), P(t-1), P(t-2), Q(t-1), Q(t-2)\}$ (H2)

c) $Q(t)=f\{P(t), P(t-1), P(t-2), P(t-3), Q(t-1), Q(t-2), Q(t-3)\}$ (H3)

d) $Q(t)=f\{P(t), P(t-1), P(t-2), P(t-3), P(t-4), Q(t-1), Q(t-2), Q(t-3), Q(t-4)\}$ (H4)

e) $Q(t)=f\{P(t), P(t-1), P(t-2), P(t-3), P(t-4), P(t-5), Q(t-1), Q(t-2), Q(t-3), Q(t-4), Q(t-5)\}$ (H5)

where t = time (hours), P = precipitation (mm), Q = discharge (m^3/s). Equations (H1), (H2), (H3), (H4) and (H5) represent operations to forecast discharge at current hour with 1, 2, 3, 4, and 5 hours of antecedent data, respectively.

A single objective function named as Mean Squared Error (MSE) is used to calibrate the model. The accuracy of the simulation results are measured with coefficient of correlation (R) and Nash-Sutcliffe coefficient (E^2). The closer R and E^2 values to 1.0 indicate better results obtained. The formulas of these two coefficients are given in Table 1.

Table 1: Statistics for Model Comparison.

Concept	Name	Formula
Coefficient of Correlation	R	$\dfrac{\sum(obs-\overline{obs})(pred-\overline{pred})}{\sqrt{\sum(obs-\overline{obs})^2\sum(pred-\overline{pred})^2}}$
Nash-Sutcliffe Coefficient	E^2	$E^2=1-\dfrac{\sum_i^j(obs-pred)^2}{\sum_i^j(obs-\overline{obs})^2}$

Note : obs = observed value, pred = predicted value, $\overline{obs}$ = mean observed values, $\overline{pred}$ = mean predicted values and N = number of values.

The simulated peak for each storm was compared with observed peak at different lead-time. Error between observed peak and simulated peak is calculated using Equation 3.

Error = {(Simulated peak-observed peak)/observed peak} X 100% (3)

5 Learning Mechanism

Three years of rainfall and runoff data starting from 1998 to 2000 are collected from Hydrology and Water Resources Section, Department of Irrigation and Drainage (DID), Sarawak for calibrating the rainfall-runoff model. Three separate storm hydrographs that happened from 8 to 12 August 1998, 5 to 8 April 2000, 18 to 21 January 2000, respectively, are used for model calibration for both MLP and REC networks. Once the optimal configuration was found, the optimal model will be validated with a single storm dated 26 to 31 January 2000 based on R and E^2 performance criteria. The learning mechanism for MLP and REC models are trained with:

a) TRAINSCG, TRAINGDX and TRAINCGB training algorithms.
b) 1, 2, 3, 4 and 5 number of antecedent hours.
c) 100, 125, 150, 175 and 200 number of neurons in the hidden layer
d) Learning rate value of 0.2, 0.4, 0.6 and 0.8 respectively.

Thereafter, the optimal configuration of MLP and REC models obtained will be applied to forecast 6 separate storm events with 3, 6, 12 and 18 hours ahead. The details of 6 separate storm events are presented in Table 2.

Table 2: Validation datasets

Item	Period
1	1-7 Jan 1999
2	5-8 Feb 1999
3	8-12 Aug 1998
4	9-12 Sep 1998
5	15-18 Mac 1999
6	20-24 Jan 1999

The framework of calibration, testing and forecasting procedures are illustrated in Fig. 5.

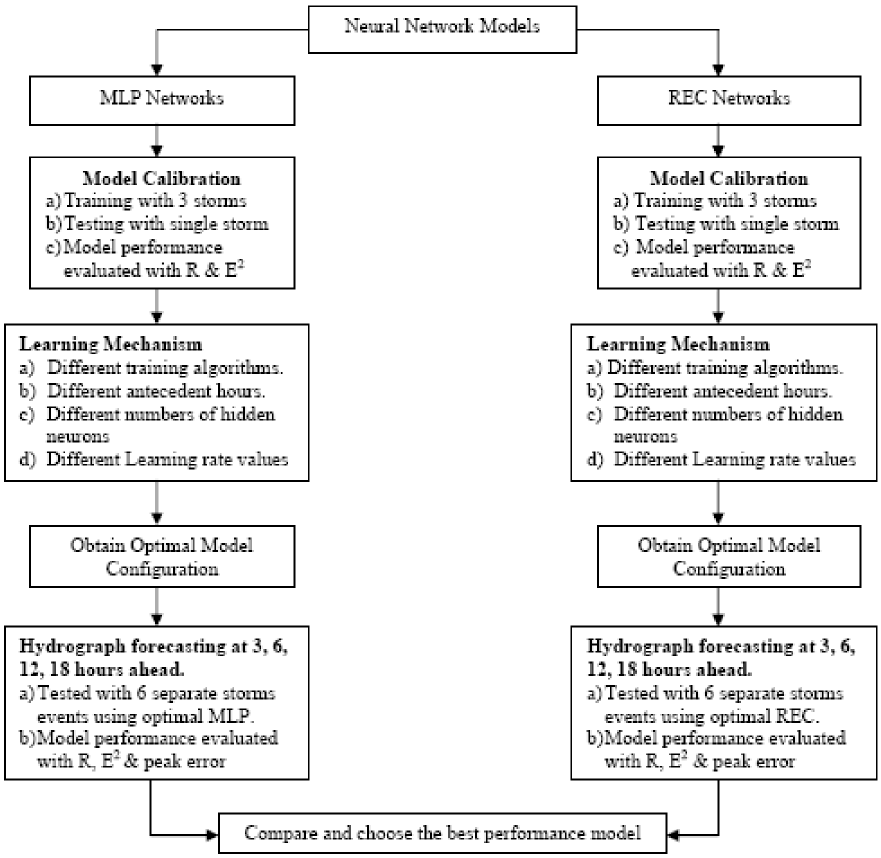


Fig. 5: Framework for Training, Testing and Forecasting Using ANNs.

6 Results and Discussion

The models developed were trained and tested using different learning algorithm, different length of training data, different number of neurons in the hidden layer and different learning rate value. The effect of each parameter to MLP and REC testing data sets are presented below.

6.1 Result and discussion for MLP Network

6.1.1 Effect of Different Types of Training Algorithm

Reasonable results were obtained by MLP network using three different training algorithms namely TRAINSCG, TRAINGDX and TRAINCGB. However, the best result was obtained using TRAINSCG as shown in Table 3. MLP with TRAINGDX is performing slightly lower than TRAINSCG but better than TRAINCGB.

Table 3: MLP Network with Different Training Algorithm.

No. of Antecedent Hours	TRAINSCG		TRAINGDX		TRAINCGB	
	R	E^2	R	E^2	R	E^2
H1	0.826	0.793	0.805	0.775	0.792	0.762
H2	0.858	0.856	0.846	0.834	0.845	0.822
H3	0.880	0.875	0.864	0.867	0.861	0.853
H4	0.985	0.972	0.973	0.947	0.945	0.932
H5	0.960	0.949	0.933	0.936	0.913	0.905

Note: MLP trained with 150 hidden neurons and LR of 0.8.

6.1.2 Effect of Number of Hidden Neurons

The performance of MLP network at 4 antecedent hours increased with the increase of number of hidden neurons from 100 to 150 (refer to Table 4). However, model performance is slightly decreased at 175 and 200 hidden nodes. Therefore, the optimum number of hidden nodes for this particular MLP network is 150.

Table 4: MLP Network at Different Number of Hidden Nodes.

No. of Hidden Neurons	R (Testing)	E^2 (Testing)
100	0.881	0.896
125	0.918	0.903
150	0.985	0.972
175	0.970	0.962
200	0.967	0.956

Note: MLP trained with TRAINSCG, 4 antecedent hours and LR of 0.8.

6.1.3 Effect of Learning Rate Value

The performance of MLP network is not clearly affected by learning rate. Table 5 presents the results of MLP when trained with different learning rate ranging from 0.2 to 0.8. Experiments show that the required training period is shortening as the learning rate increased. Hence, the best learning rate selected for MLP network is 0.8.

Table 5: MLP Network at Different Learning Rate Value.

Learning Rate	R (Testing)	E^2 (Testing)
0.2	0.985	0.972
0.4	0.985	0.969
0.6	0.984	0.971
0.8	0.985	0.972

Note: MLP trained with TRAINSCG, 150 hidden neurons and 4 antecedent hours.

6.1.4 Effect of Antecedent Hours

From H1 to H4, the performance of MLP network kept on increasing with the increase of antecedent hours. For H4, the R and E^2 values are 0.985 and 0.972 respectively. However, the R and E^2 decreased to 0.960 and 0.949 respectively for H5 (refer to Table 6). Thus, the optimum number of antecedent hours found in this study is 4.

Table 6: MLP Network at Different Number of Antecedent Hours.

No. of Antecedent Data	**R (Testing)**	**E^2 (Testing)**
H1	0.826	0.793
H2	0.858	0.856
H3	0.880	0.875
H4	0.985	0.972
H5	0.960	0.949

Note: MLP trained with TRAINSCG, 150 hidden neurons and LR of 0.8.

6.2 Results and Discussion of REC Network

6.2.1 Effect of Different Types of Training Algorithm

REC network performed the best with TRAINSCG, followed second best with TRAINGDX. When trained with TRAINSCG, the accuracies of the result obtained are slightly lower than TRAINSCG and TRAINGDX algorithms (refer to Table 7).

Table 7: REC Network with Different Training Algorithm.

No. of Antecedent Hours	**TRAINSCG**		**TRAINGDX**		**TRAINCGB**	
	R	E^2	R	E^2	R	E^2
H1	0.864	0.877	0.842	0.864	0.832	0.852
H2	0.938	0.887	0.898	0.884	0.855	0.871
H3	0.948	0.925	0.944	0.921	0.893	0.913
H4	0.990	0.983	0.978	0.947	0.944	0.930
H5	0.966	0.954	0.949	0.946	0.927	0.930

Note: REC trained with 150 hidden neurons and LR of 0.8.

6.2.2 Effect of Number of Hidden Neurons

Table 8 shows that REC network is able to predict accurately ($R \geq 0.9$ and $E^2 \geq 0.85$) with different number of hidden neurons ranging from 100 to 200. The performance of REC network kept on improving from 100 to 150 hidden neurons. However, R and E^2 are decreased to 0.987 and 0.951 respectively at 175 hidden neurons, 0.978 and 0.948 respectively at 200 hidden neurons. Similar with MLP network, REC also performs best with 150 hidden neurons.

Table 8: REC Network at Different Number of Hidden Nodes

No. of Hidden Neurons	**R (Testing)**	**E^2 (Testing)**
100	0.901	0.860
125	0.962	0.924
150	0.990	0.983
175	0.987	0.951
200	0.978	0.948

Note: REC trained with TRAINSCG, 4 antecedent hours and LR of 0.8.

6.2.3 Effect of Learning Rate Values

The performance of REC network is not affected by the learning rate as shown by the R and E^2 obtained (refer to Table 9). However, when analysing E^2 values, it can be adopted that LR=0.8 as an optimum value (LR=0.6 looks quite good as well). Moreover, the training period required for learning rate of 0.8 is the shortest among the 4 learning rates investigated.

Table 9: REC Network at Different Learning Rate Value.

Learning Rate	**R (Testing)**	**E^2 (Testing)**
0.2	0.989	0.982
0.4	0.985	0.980
0.6	0.990	0.979
0.8	0.990	0.983

Note: REC trained with TRAINSCG, 150 hidden neurons and 4 antecedent hours.

6.2.4 Effect of Antecedent Data

The performance of REC network is increased from H1 to H4. R and E^2 for H4 are found to be 0.990 and 0.983 respectively. The R and E^2 values have decreased to 0.966 and 0.954 respectively for H5 (refer Table 10). Similar to MLP network, the optimum results for REC network was obtained using 4 antecedent hours.

Table 10: REC Network at Different Number of Antecedent Hours.

No. of Antecedent Data	R (Testing)	E^2 (Testing)
H1	0.864	0.877
H2	0.938	0.887
H3	0.948	0.925
H4	0.990	0.983
H5	0.966	0.954

Note: REC trained with TRAINSCG, 150 hidden neurons and LR of 0.8.

6.3 Comparison of the Two ANNs for Hourly Runoff

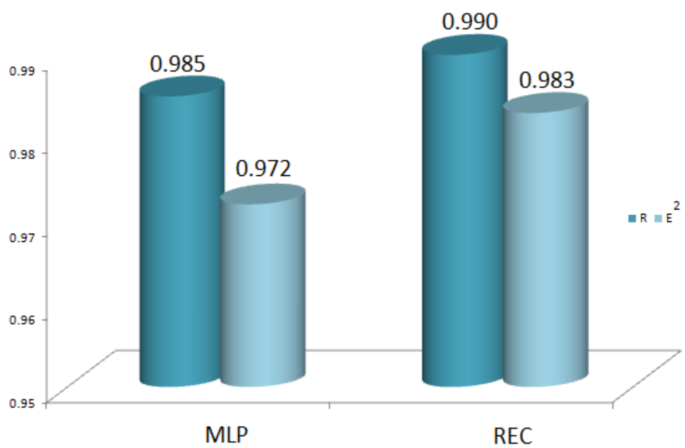


Fig. 6: Comparison of Optimal MLP and REC networks for Hourly Runoff.

Fig. 6 shows the best results produced by the optimal configuration of MLP and REC networks for hourly runoff. The optimum R and E^2 for MLP are found to be 0.985 and 0.972 respectively with the configuration of TRAINSCG training algorithm, 150 hidden neurons, learning rate of 0.8 and 4 antecedent hours. With the same configuration, the optimal results are obtained for REC networks, where optimal R and E^2 are yielded to 0.990 and 0.983 respectively. The results indicate that both networks are able to simulate the hourly runoff acutely. However, the performance of REC is slightly better than MLP.

6.4 Runoff Simulation with Lead-time

The optimal MLP and REC networks determined previously will be utilized to generate storm hydrograph for 6 selected storms. These 6 storm hydrographs will be forecasted with a lead-time of 3, 6, 12 and 18 hours ahead. The average R and E^2 for foresting the 6 storms events at different lead time using optimal MLP and REC networks are presented in Fig. 7 and Fig. 8 respectively. In general, the performance for both optimal MLP and REC networks are decreasing with the increase of lead-time

Average R and E^2 produced by optimal MLP for 6 storms with 1 hour ahead is 0.983 and 0.970 respectively. Results revealed that the average R and E^2 are decreased with the increased of lead time from 3, 6, 12 to 18-hour ahead runoff forecast. At 3 hours ahead runoff forecast, the average R and E^2 decreased to 0.963 and 0.945 respectively. Thereafter, the average R and E^2 continuously decreased to 0.946 and 0.914 respectively at 6 hours ahead lead time, 0.930 and 0.867 respectively at 12 hours ahead lead time, 0.889 and 0.758 respectively at 18 hours ahead lead time.

Table 11: Comparison between Observed and Simulated Peak Flow for Optimal MLP at Different Lead Time.

	Observed peak	1-hour Ahead		3-hour Ahead		6-hour Ahead		12-hour Ahead		18-hour Ahead	
		Simulated peak	Peak error	Simulated peak	Peak error	Simulated peak	Peak error	Simulated peak	Peak error	Simulated peak	Peak error
Storm Hydrograph 1	7.8800	8.2091	4.20%	8.7406	10.90%	9.0621	15.00%	9.5375	21.00%	10.7178	36.00%
Storm Hydrograph 2	18.8600	20.6393	9.40%	21.7964	15.60%	22.3439	18.50%	20.6365	9.40%	22.4600	19.10%
Storm Hydrograph 3	17.8900	18.7054	4.60%	19.3026	7.90%	19.5590	9.30%	20.2345	13.10%	20.3288	13.60%
Storm Hydrograph 4	4.1000	4.1588	1.40%	4.3899	7.10%	4.7809	16.60%	4.7131	15.00%	5.0873	24.10%
Storm Hydrograph 5	9.7200	10.0175	3.10%	10.0449	3.30%	10.3291	6.30%	11.7375	20.80%	9.9010	19.00%
Storm Hydrograph 6	17.8900	18.9074	5.70%	19.9495	11.50%	20.7119	15.80%	21.7049	21.30%	20.4680	14.40%
Average peak error			4.73%		9.38%		13.58%		16.77%		21.03%

Table 12: Comparison Between Observed and Simulated Peak Flow for Optimal REC at Different Lead Time.

	Observed peak	1-hour Ahead		3-hour Ahead		6-hour Ahead		12-hour Ahead		18-hour Ahead	
		Simulated peak	Peak error	Simulated peak	Peak error	Simulated peak	Peak error	Simulated peak	Peak error	Simulated peak	Peak error
Storm Hydrograph 1	7.88	8.0588	2.30%	8.3167	5.50%	9.4428	19.80%	9.6811	22.90%	10.5699	34.10%
Storm Hydrograph 2	18.86	20.3754	8.00%	21.8097	15.60%	22.1529	17.50%	21.4443	13.70%	21.9223	16.20%
Storm Hydrograph 3	17.89	18.4777	3.30%	19.7792	10.60%	19.4313	8.60%	20.2044	12.90%	19.9821	11.70%
Storm Hydrograph 4	4.1	4.1554	1.40%	4.2611	3.90%	4.4549	8.70%	4.7906	16.80%	5.1328	25.20%
Storm Hydrograph 5	9.72	9.8884	1.70%	10.3428	6.40%	10.703	10.10%	10.3447	6.40%	10.0289	3.20%
Storm Hydrograph 6	17.89	18.6003	4.00%	19.9495	11.50%	20.6352	15.30%	21.4383	19.80%	20.468	14.40%
Average peak error			3.45%		8.92%		13.33%		15.43%		17.47%

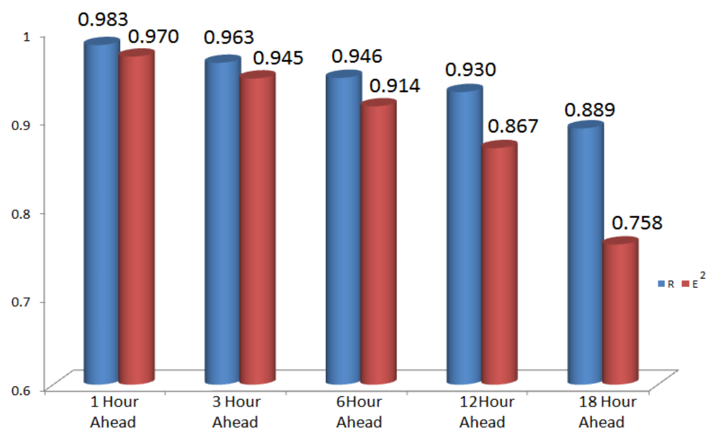


Fig. 7: Average R and E^2 for Optimal MLP with the Increase of Lead-time.

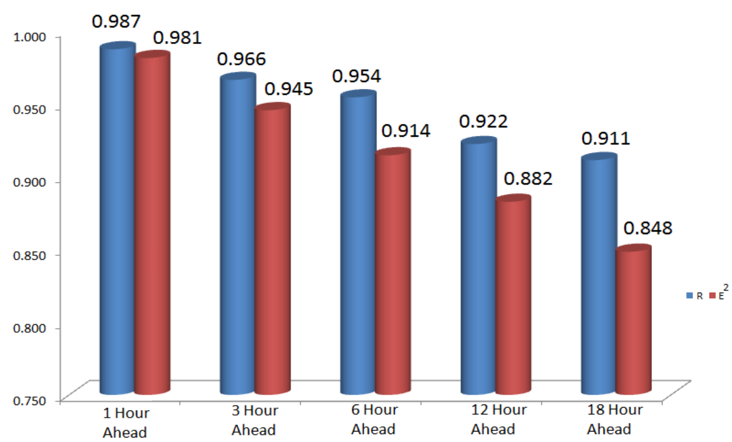


Fig. 8: Average R and E^2 for Optimal REC with the Increase of Lead-time.

Similarly for REC network, the accuracies of the results obtained are decreased with the increased of lead time. Average R and E^2 produced by optimal REC at 1 hour ahead runoff forecast is 0.987 and 0.981 respectively. The results are decreasing with the increased of lead time where average R and E^2 decreased to 0.966 and 0.945 respectively at 3 hours ahead runoff forecast, decreased to 0.954 and 0.914 respectively at 6 hours ahead lead time, 0.922 and 0.882 respectively at 12 hours ahead lead time, 0.911 and 0.848 respectively at 18 hours ahead lead time.

6.5 Comparison between Observed Peak and Simulated Peak

Comparison between observed and simulated peak flow for optimal MLP and REC are presented in Table 11 and Table 12 respectively.

In general, the results indicate that peak error is increasing with the increased of the lead-time. Both MLP and REC are found to be able to simulate peak flow accurately for 1, 3 and 6 hours ahead with average peak error less than 15%. Thereafter, the average peak error for optimal MLP at 12 and 18 hours ahead lead time increased to 16.77% and 21.03% respectively. Similarly for optimal REC network, the average peak error also increased to 15.43% and 17.47% for 12 and 18 hours lead time respectively. The results confirmed that these MLP and REC networks can be used as early warning flow forecaster to take necessary flood protection measures before a severe flood occurs.

7 Conclusion

In general, the results show that the performance of REC is slightly better than MLP network based on the R, E^2 and peak error analysis. The optimal configuration of both neural networks are found to be using TRAINSCG training algorithm, 150 hidden neurons, learning rate of 0.8 and 4 antecedent hours.

The results also revealed that the optimal configuration of MLP and REC networks are able to simulate peak flow accurately at different lead time. However, the models performance are decreasing with the increase of lead time. At the lead time of 1, 3 and 6 hours ahead runoff forecast, the average peak error obtained for both optimal MLP and REC networks are less than 15%. Thereafter, the average peak error increased to 21.03% and 17.47% for MLP and REC networks respectively at 18 hours ahead runoff forecast. This indicates both networks can be used as early warning flow forecaster to take necessary flood protection measures before a severe flood occurs.

This study also reveals that lag time can be excluded as input because ANNs is capable to adapt to the respective lag time of each gauge through training. For catchment in tropical region, rainfall and runoff as inputs are sufficient to develop an accurate rainfall-runoff model. Inclusion of more parameters such as temperature, moisture content, evaporation will make the ANNs model unnecessarily complex in nature without any significant improvement in performance.

8 References

[1] Dastorani, M.T., and Wright, N.G., "Artificial Neural Network Based Real-time River Flow Prediction", School of Civil Engineering, University of Nottingham, Nottingham NG7 2RD, UK, (2001).

[2] DID, *Hydrological Year Book Year 2007*, Department of Drainage and Irrigation Sarawak, Malaysia, (2007).

[3] Dooge, J.C.I., "Problems and Methods of Rainfall-Runoff Modeling, Mathematical Models for Surface Water Hydrology", *John Wiley and Sons*, (1974), pp.71-108.

[4] Elshorbagy, A., Simonovic , S.P., and Panu, U.S., "Performance Evaluation of Artificial Neural Networks for Runoff Prediction", *Journal of Hydrologic Engineering*, (2000), 5(4): 424-427.

[5] Garcia-Bartual, R., "Short Term River Flood Forecasting with Neural Networks", *Universidad Politecnica de Valencia, Spain,* (2002), pp.160-165.

[6] Gautam, M.R., Watanabe, K., and Saegusa, H., "Runoff Analysis in Humid Forest Catchment with Artificial Neural Networks", *Journal of Hydrology*, (2000), 235: 117-136.

[7] Harun, S., Kassim, A.H., and Van, T.N., "Inflow Estimation with Neural Networks", *10th Congress of The Asia and Pacific Division of the International Association for Hydraulic Research,* (1996), pp.150-155.

[8] Hemanth, D.J., Vijila, C.K.S., and Anitha, J., "Performance Improved PSO based Modified Counter Propagation Neural Network for Abnormal MR Brain Image Classification", *Int. J. Advance. Soft Comput. Appl.,* Vol. 2, No. 1, (2010), pp: 65-84.

[9] Idris, N., Yusof, N., and Saad, P., "Adaptive Course Sequencing for Personalization of Learning Path Using Neural Network", *Int. J. Advance. Soft Comput. Appl.,* Vol. 1, No. 1, (2009), pp:49-61.

[10] Imrie, C.E., Durucan, S. and Korre, A., "River Flow Prediction Using Artificial Neural Networks: Generalization Beyond the Calibration Range", *Journal of Hydrology*, (2000), 233: 138-153.

[11] Jain, S.K., and Chalisgaonkar, C., "Setting Up Stage-Discharge Relations Using ANN", *Journal of Hydrologic Engineering*, (2000), 5(4), 424-433.

[12] JUPEM, *Jabatan Ukur dan Pemetaan Malaysia*, Scale 1:50,000, (1975).

[13] Minn, A. W., and Halls, M. J., "Artificial neural networks as rainfall-runoff models", *Hydrological Science Journal*, 41(3), (1996), pp. 339-417.

[14] Nishimura, S., and Kojiri, T., "Real-time Rainfall Prediction Using Neural Network and Genetic Algorithm with Weather Radar Data", *10th Congress of the Asia and Pacific Division of the International Association for Hydraulic Research*, (1996), pp.204-211.

[15] Nurmaini, S., Hashim, S.Z.M., and Jawawi, D.N.A., " Modular Weightless Neural Network Architecture for Intelligent Navigation", *Int. J. Advance. Soft Comput. Appl.,* Vol. 1, No. 1, (2009), pp: 1-18.

[16] Osman, M.H., Liong, C.Y., and Hashim, I., "Hybrid Learning Algorithm in Neural Network System for Enzyme Classification", *Int. J. Advance. Soft Comput. Appl.,* Vol. 2, No. 2, (2010), pp: 209-220.

[17] Rumelhart, D. and McClelland, J. "Explorations in parallel distributed processing". *A handbook of models, Programs and exercises*, MIT Press, Cambridge, (1988).

[18] Sivasankari, N and Thanushkodi, K., "Automated Epileptic Seizure Detection in EEG Signals Using FastICA and Neural Network", *Int. J. Advance. Soft Comput. Appl.,* Vol. 1, No. 2, (2009), pp.91-104.

[19] Wright, N.G., and Dastorani, M.T., "Effects of River Basin Classification on Artificial Neural Networks Based Ungauged Catchment Flood Prediction", *Proceeding of the 2001 International Symposium on Environmental Hydraulics*, (2001).