Rainfall Runoff Modeling
using Radial Basis Function Neural Network for Sungai Tinjar Catchment, Miri, Sarawak

Suhaimi, S[1]., Rosmina A. Bustami[2]

Abstract

Artificial Neural Network (ANN) is a very useful data modelling tool that is able to capture and represent complex input and output relationships. The advantage of ANN lies in its ability to represent both linear and non-linear relationships and in its ability to learn these relationships directly from the data being modelled. Modeling of rainfall runoff relationship is important in view of the many uses of water resources such as hydropower generation, irrigation, water supply and flood control.

This study is to purposefully develop a rainfall runoff model for Sg. Tinjar with outlet at Long Jegan using Radial Basis Function (RBF) Neural Network. Training and simulation was done using Matlab 6.5.1 software with varying parameters to obtain the optimum result. Further, the results were compared to simulation done with Multilayer Percepteron model. The RBF network developed in this study has successfully modelled rainfall runoff relationship in Sungai Tinjar Catchment in Miri, Sarawak with an accuracy of about 98.3%.

Keywords: Neural Network, Radial Basis Function, Rainfall Runoff Relationship

I. INTRODUCTION

RAPID population growth, urbanization and industrialization have increased the demand for water and altered the watersheds and river systems. This will cause greater damages to the properties and resulted in losses of life if flooding occurred. Therefore, it is becoming intensely critical to plan, design and manage water resources systems carefully and intelligently.

For many years, hydrologists are trying to determine the relationship of transformation of precipitation to runoff. Forecasted runoff can be used in streamflow measurement and planning for water supply, flood control, irrigation, drainage, power generation, water quality, recreation, and fish and wildlife propagation.

A rainfall runoff model is required to obtain the relationship between rainfall and runoff. This model is capable of forecasting future river discharge values that are needed for hydrologic and hydraulic engineering design and water management purposes. Hence, it is important to hydrologist in determining these two relationships. However, this relationship is known to be highly non-linear and complex due to large spatial and temporal variability of watershed characteristics and precipitation patterns, and the number of variables involved in the modeling of the physical process [1].

Rainfall runoff modeling can be carried out based on observations of the inputs and outputs to a catchments area. However, the relationships between the rainfall over a catchment's area and the resulting flow in a river are the problems that are commonly confronted by most hydrologists. Some data were often missing despite the availability of plenty rainfall records in the study area. Therefore, simulation models are commonly used to overcome these problems.

Artificial Neural Networks (ANN) modeling has gained significant attention in recent years due to its ability to provide better solutions when applied to complex systems that have been poorly described or understood, problems that deal with noise or involve in pattern recognition, diagnosis, abstraction and generalization, and where input is incomplete or ambiguous by nature.

1 Student, Department of Civil Engineering, Faculty of Engineering, Universiti Malaysia Sarawak, Sarawak, Malaysia,
2 Lecturer, Department of Civil Engineering, Faculty of Engineering, , Universiti Malaysia Sarawak, Sarawak, Malaysia ,abrosmina@feng.unimas.my

In hydrological modeling, the ANN method has been widely proven to be a very potentially useful tool such as to modeling rainfall runoff processes [1] – [3], streamflow prediction [4], [5], water level prediction [6], [7], operation of reservoir system [8] and ground water reclamation systems [9].

In this study, Radial Basis Function (RBF) Neural Networks is used to construct a rainfall runoff model. To do the training, Matlab 6.5.1 computer software was used. Further, the result is compared to that obtained by rainfall runoff model designed with Multilayer Percepteron model.
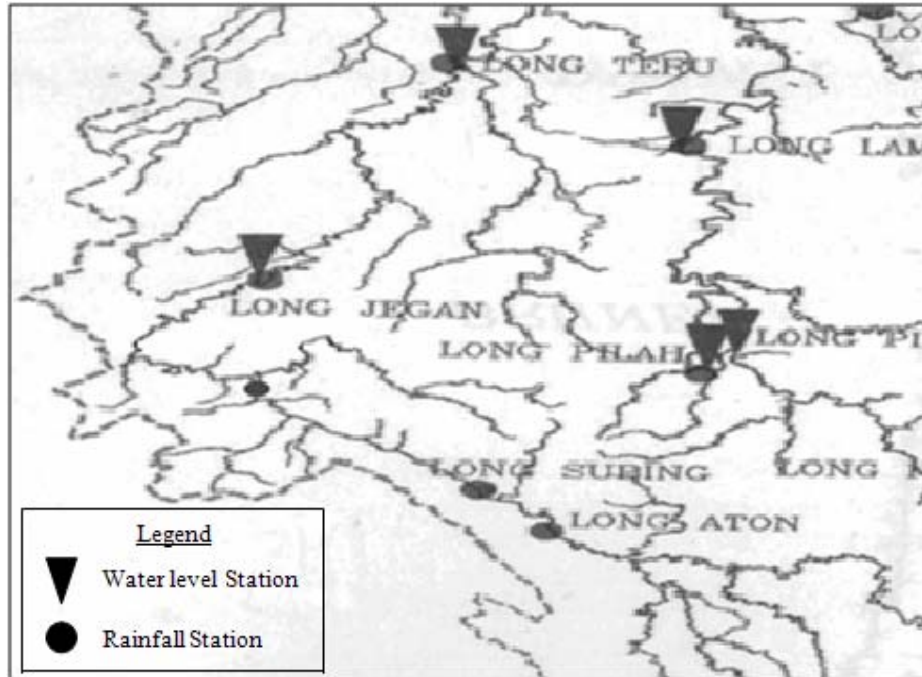
2. STUDY AREA

Figure 1: Sg. Tinjar Catchment [10]

The study area selected for this study is Sg. Tinjar catchment, as shown in figure 1. This catchment is located in Baram basin, Miri, Sarawak. Sg. Tinjar catchment has four rainfall gauging stations which are Long Jegan, Long Bantan, Long Subing and Long Aton. Long Jegan also functions as the outlet of the catchments. The runoff was converted from water level data through a rating curve given by (1).

$$Q = 6.64 \ (H\text{-}2.13)^{1.82} \tag{1}$$

Where:
  Q : discharge in m^3/s, and
  H : stage discharge in m.

Daily hydrologic data for rainfall and runoff were obtained from Department of Irrigation and Drainage, Kota Samarahan, Sarawak.

3. METHODOLOGY

Data used in this study are rainfall and runoff data for selected rainfall gauges and one water level station. For this purpose, data was collected from year 2005 to 2007. The data obtained will be divided into two sets, for calibration (training) and for verification (testing). Three sets of different models have been developed as presented in table 1. The first set is built using three months data for a particular year, and tested with a month of data, a second set use a six months of data for training and it is tested against three months of output data, and the third set used two years of data for training and one year data for testing.

Table 1: Training and testing period of built networks.

Set	Training Period		Testing Period
	Length of Data	Description of Data	
1	Three month	January to March 2005	April 2005
2	Six month	January to June 2006	July to September 2006
3	Two years	2005 & 2006	2007

Input data used for daily rainfall runoff models are antecedents total daily precipitation {P(t-1), P(t-2),.......P(t-n)}, total rainfall of the current day from each rain gauge {P(t)}, antecedent daily mean discharge from outlet gauge for {Q(t-1), Q(t-2)....Q(t-n)} and the equation of this nonlinear model is given by (2). Output data is the simulated mean daily runoff {Q(t)}. Network designed here is tested with 3 different antecedent days; namely 3 days, 4 days, and 5 days to foresee their effect on rainfall runoff relationship.

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

    Where,
        P : Precipitation
        Q : Discharge
        t : Number of days

The results obtained from training and testing are evaluated to determine the difference between observed and predicted values. Performances of the RBF model is evaluated using the coefficient of correlation (R) and coefficient of efficiency (E^2). The formulas of these two coefficients are given in (3) and (4) respectively.

$$R = \frac{\sum_{i=1}^{i=n}[Q_m(t_i) - \overline{Q_m(t_i)}][Q_s(t_i) - \overline{Q_s(t_i)}])}{\sqrt{\sum_{i=1}^{i=n}[Q_m(t_i) - \overline{Q_m(t_i)}]^2 \sum[Q_s(t_i) - \overline{Q_s(t_i)}]^2}} \tag{3}$$

$$E^2 = 1 - \frac{\sum_{i=1}^{i=n}[Q_m(t_i) - Q_s(t_i)]^2}{\sum_{i=1}^{i=n}[Q_m(t_i) - \overline{Q_m}]^2} \tag{4}$$

    Where:
        n: number of data
        Q_m: *measured value of discharge*
        Q_s: *simulated value of discharge*
        $\overline{Q_m}$: *average of measured discharge value*
        $\overline{Q_s}$: *average of simulated discharge value*

ANN develop for the prediction of rainfall runoff is trained with different length of training data, number of neurons in its hidden layer, spread value and antecedent days. The aim is to create a network which will give an optimum result.

Based on [11], Radial basis networks consist of two layers: a hidden radial basis layer of S^1 neurons and an output linear layer of S^2 neurons as presented in figure 2.
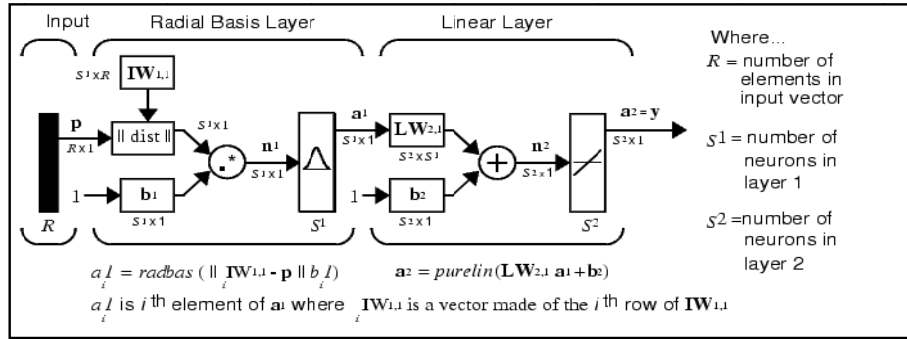
Figure 2: Radial Basis Function [11]

In order to design the network, a design function newrb will be the net code and their outputs can be obtained with sim. This network will behave by following an input vector $\mathbf{p}$ through the network to the output $\mathbf{a}^2$. When an input vector is present to the network, each neuron in the radial basis layer will output a value according to how close the input vector is to each neurons's weight vector. Typically several neurons are always firing, to varying degrees. The bias $\mathbf{b}$ allows the sensitivity of the radbas neuron to be adapted.

The function newrb creates a two layer network. The first layer has a radbas neurons and calculates its weighted inputs with dist, and its net input with netprod. Number of neurons in the hidden layer must be sufficient enough to withhold the mass of inputs. Number of hidden neurons in the hidden layer was investigated by trial and error method. The values investigated are 10, 20, 30, 40, 50, and 60. While the second layer has purelin neuron that calculates its weighted input with dotprod and its net input with netsum. Both of the layers have biases. The training process will keep generating until the neurons networks mean squared error goal is met or the maximum number of neuron is achieved.

A smoothing parameter known as spread, need to be set. Basically, the larger the value of spread, the smoother the function approximation will be. However, if the value of the spread is too large, many neurons will be required to fit a fast changing function. Meanwhile, if it is too small, many neurons will be required in order to fit a smooth function and the networks will not be generalized well. Therefore, in this project a set of trial and error of spread value will be tested to find the optimum value for determining the best network for radial basis function network. The spread value tested will be in the low range (0.01, 1, 10), intermediate range (100,200,300) and high range (500,800,1000).

The optimum result from this study will then be compared to an earlier study done in the same catchment [12], which made use of Multilayer Perceptron ANN to simulate rainfall runoff relationship in Sg. Tinjar Catchment.

4. RESULTS

Results of simulation for rainfall runoff modeling are discussed in the following subsections A-D:

A. Length of training data

Results for three sets (Set 1, Set 2 and Set 3) with 3 antecedent days show that RBF neural network gave a good fit in modeling the rainfall runoff relationship, as shown in table 2.

Table 2: Three sets of training and testing (with 20 neuron and 200 spread value) of "3 antecedent days".

Set	Training		Testing	
	R	E^2	R	E^2
1	0.9709	0.9426	0.9828	0.9533
2	0.8858	0.7847	0.9521	0.8473
3	0.9099	0.8278	0.8971	0.8009

The results indicate that Set 1 with 3 antecedent days give a better performance compared to the other sets. This shows that RBF neural network can achieve a good results even when the training data set (input) is small, in this case only three months as it achieved values for R(testing) = 0.9828 and E^2(testing) = 0.9533. For this, RBF neural network can be beneficial at a station which has limited number of training data for simulation.

B. Number of hidden neurons

The number of neurons in a hidden layer that were used for the training of the network is set to vary from 10 to 60. The performance of networks with 3 days antecedent data for Set 1 is shown in table 3. The results show the performance of the networks simulated with different number of neurons in its hidden layer.

Table 3: Set 1 training and testing for R and E^2

Neurons	Training		Testing	
	R	E^2	R	E^2
10	0.9511	0.9046	0.9750	0.9387
20	0.9709	0.9426	0.9828	0.9533
30	0.9862	0.9725	0.9779	0.9479
40	0.9912	0.9825	0.9791	0.9514
50	0.9949	0.9898	0.9791	0.9487
60	0.9976	0.9952	0.9591	0.8834

Thus, the optimum number of neurons with the best performance during training and testing is when the number of neurons is 20 which produced the results of R(testing) = 0.9828 and E^2(testing) = 0.9533 for Set 1.

C. Spread value

A series of training is done to find the optimum value for the spread constant. The results obtained for Set 1 for "antecedent 3 days" and 20 neurons in hidden layer are presented in table 4.

Table 4: Set 1 training and testing

Range	SPREAD	Training		Testing	
		R	E^2	R	E^2
Low	0.01	0.8395	0.7048	-inf	-7.7305e-004
	1	0.8395	0.7048	-inf	-7.7305e-004
	10	0.8397	0.7051	0.1928	-7.7305e-005
Intermediate	100	0.9593	0.9202	0.9394	0.8596
	200	0.9709	0.9426	0.9828	0.9533
	300	0.9750	0.9507	0.9779	0.9398
High	500	0.9700	0.9409	0.9768	0.9503
	800	0.9712	0.9433	0.9772	0.9516
	1000	0.9687	0.9384	0.9765	0.9489

From table 4, the accuracy of the results for RBF neural network was strongly influenced by the optimal spread value. As noted, the optimal spread value can only be found through a series of trial and error process.

The optimal spread value should be large enough so that active input regions of the radbas neurons overlap enough and always have fairly large output at any given moment. In contrast, spread should not be too large so that each neuron is effectively responding in the same, large, area of input space. Accuracy of an RBF neural network decreased when the chosen spread value is above the optimal value. Hence, from trial and error procedure, spread value of 200 is the optimal value for this study.

D. Antecedent days

Three antecedent days are chosen in the study, which are 3 days, 4 days and 5 days. Figure 3 shows the performance of the RBF networks with comparison of different antecedent days when simulated with 20 hidden neurons and a spread value of 200.
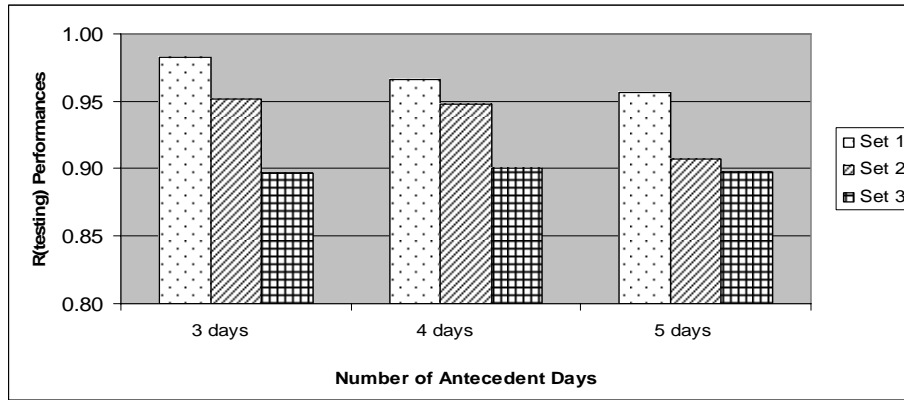
Figure 3: R(testing) for different antecedent days and Set.

From figure 3 the number of antecedent days that performs the best is antecedent 3 days. The value for R(testing) is 0.9828 for Set 1. Meanwhile, antecedent 4 days and antecedent 5 days give a lower performance in comparison with antecedent 3 days.

5. DISCUSSION

The accuracy of the RBF neural network was strongly influenced by its optimal spread value. Referring to Set 1 with 3 antecedent days, when optimum spread value is reached (spread value of 200) the network model become more accurate during testing as it gave the result of R = 0.9828 and E^2 = 0.9533. If the spread value used is less than its optimum value, the neural network will not generate good results, whilst, if it is above the optimal value, the accuracy of the RBF neural network will be decreased.

Number of neurons in the hidden layer also influences the performances of RBF neural network. Set 1 with 3 antecedent days which gave the optimum result as mentioned previously was fed with 20 input nodes. If the number of hidden layer neurons is small, the network model did not achieve a desired level of accuracy. Meanwhile, with too many nodes it will take a longer time to train and sometimes it will over fit the data.

Comparisons of RBF neural network with Multilayer Percepteron (MLP) architecture from rainfall runoff simulation carried out in [12] is as shown in figure 4.
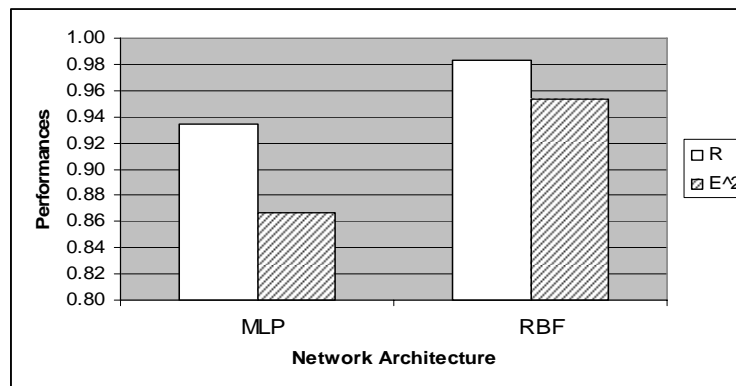

Figure 4: Comparison of MLP and RBF neural network performances.

From figure 4, resulting performances of RBF are R = 0.9828 and E^2 = 0.9533, meanwhile for MLP, they are R = 0.9344 and E^2 = 0.8668. Therefore, the RBF neural network model displays a better performance compared to the MLP neural network model.

The advantage of RBF neural network is that it can be trained much faster than the MLP neural network. It also can give a better performance with a minimum number of training data set (3 months) to perform a successful training and the performance is hardly influenced by the level of non-linearity of the data.

Therefore, RBF neural network developed in this study performed very well in modeling rainfall runoff relationship. Parameters of the optimum network for rainfall runoff modeling of the studied catchment are as shown in table 5.

Table 5: Optimal RBF neural network model.

Network	Set 1
Network Architecture	Feedforward Neural Network
Network Structure	Hidden Layer – Radial Basis Function Output Layer – Purelin Transfer Function
Number of Neurons	Hidden Layer – 20 Neurons Output Layer – 1 Neuron
Antecedent Days	3 Days
Optimal Spread Value	200
Coefficient of Correlation (R)	98.3%
Coefficient of Efficiency (E^2)	95.3%

Obviously, the application of RBF neural network in modeling the relationship between rainfall and runoff for Sg. Tinjar, Miri, Sarawak is appropriate. It also reflects that the performance of neural network model is satisfactory and it is feasible for use in rainfall-runoff modeling of Malaysian catchments.

6. CONCLUSIONS

The capability of artificial neural network models to predict runoff has evidently proved in this study. The non-linear nature of the relationship of rainfall-runoff processes is appropriate for the application of ANN methods. Moreover, the results reflect that the performance of the RBF neural network model is satisfactory and it is feasible for use in rainfall runoff modeling of Malaysian catchments

REFERENCES

[1] Tokar, A.S., and Johnson, P.A. (1999). Rainfall runoff modelling using artificial neural networks. *J. of Hydrologyc Eng. 4(3)* .232-239.
[2] Sobri, H., et al. Artificial neural network model for rainfall runoff relationship. *Jurnal Teknologi, 37(B) Dis. 2002: Universiti Teknologi Malaysia.* 1-12
[3] Kuok, K. K. (2004). Artificial neural network for rainfall runoff modeling with special reference to Sg. Bedup catchment area. Kota Samarahan: Universiti Malaysia Sarawak.
[4] O.R. Dolling, and E.A. Varas, "Artificial neural networks for streamflow prediction", *Journal of Hydraulic Research*. 40(5), 2002, pp. 547-554.
[5] N.G. Wright, and M.T. Dastorani, "Effects of river basin classification on Artificial neural networks based ungauged catchment flood prediction", *Proceedings of the International Symposium on Environmental Hydraulics,* Phoenix, 2001.
[6] Bessaih, N. et al. Artificial neural networks for daily runoff estimation. Engineering Technology Conference 2003, Sarawak, Malaysia
[7] Rosmina, B. et al. (2006). *Artificial neural network for precipitation and water level predictions of Bedup River*. IAENG International Journal of Computer Science. Vol. 34 (2), 228 – 233.
[8] Raman H. and Chandramouli V. (1996). "Deriving a General Operating Policy for Reservoirs Using Neural Network." *J. of Water Resources Planning and Management*. 122(5): 342-347.
[9] Ranjithan S. and Eheart J. W. (1993). "Neural Network-based Screening for Groundwater Reclamation Under Uncertainty." *Water Resources Research*. 29(3): 563-574.
[10] Department of Irrigation and Drainage. Retrieved September 20, 2008, from http://www.did.sarawak.gov.my.
[11] Demuth, H. and Beale, M. (2001). *Neural network toolbox in for use with MATLAB*. Natick, MA: The MathWorks, Inc.
[12] Hafiz, F. A. (2004). *Neural Network for Rainfall Runoff Modeling*. Kota Samarahan: Universiti Malaysia Sarawak.