Daily rainfall-runoff forecasting using GA-BP neural network

A. Sedki* — D. Ouazar* — E. El Mazoudi**

* Department of Civil Engineering
Ecole Mohammadia d'Ingénieurs,Université Mohammed V-Agdal
765 Agdal Rabat, Maroc
MOROCCO
asedki@emi.ac.ma
ouazar@emi.ac.ma

** Department of Electrical Engineering Faculty of Sciences and Technique
Fés-Sais Faculté des Sciences et Technique, Fés, Maroc
MOROCCO
h_mazoudi@yahoo.fr

RÉSUMÉ. Un réseau de neurones artificiel (RNA) trainé par un hybride algorithme, appelé AG-RP, combinant l'algorithme génétique (GA) par l'algorithme de rétropropagation(RP), est proposé pour la prévision du débit d'un bassin versant situé en zone semi aride du Maroc. Pour prédire le débit journalier, on présente à l'entrée du réseau des valeurs de pluie et de débits observés à des instants précédents. On a adopté un AG en codage réel hybridé par l'algorithme de RP. Les opérateurs génétiques sont conçus soigneusement pour optimiser le réseau neuronal, évitant des problèmes prématurés de convergence et de permutation. Pour évaluer la performance du modèle hybride AG-RP, le modèle neuronal de RP a été également développé pour un but de comparaison. Les résultats ont prouvé que le modèle hybride donne des prévisions supérieures.

ABSTRACT. An artificial neural network (ANN) based on hybrid algorithm combining genetic algorithm (GA) with back-propagation (BP) algorithm, also referred to as GA-BP algorithm, is proposed to forecast the runoff in a catchment located in a semiarid climate in Morocco. To predict daily flow, the input variables are the rainfall and the runoff values observed on the previous time period. Our methodology adopts a real coded GA strategy and hybrid with a back propagation (BP) algorithm. The genetic operators are carefully designed to optimize the neural network, avoiding premature convergence and permutation problems. To evaluate the performance of the genetic algorithm-based neural network, BP neural network is also involved for a comparison purpose. The results showed that the GA-based neural network model gives superior predictions.

MOTS-CLÉS : Réseau de neurones; Algorithme génétique; Rétropropagation; Débit; Pluie; Bassin versant; Semi aride;

KEYWORDS : Neural network; Genetic algorithm; Back propagation; Runoff; Rainfall; Catchment; Semiarid climate.

1. Introduction

The rainfall-runoff relationship is one of the most complex hydrologic phenomena to comprehend due to the tremendous spatial and temporal variability of watershed characteristics and precipitation patterns, and the number of variables involved in the modeling of the physical processes. Moreover these processes, even when simplified, are generally nonlinear. Using models with a smaller number of parameters, in order to cope with nonlinearity, is therefore necessary.

In recent years, artificial neural networks (ANN) have gained more and more popularity for hydrological forecasting (Maier and Dandy, 2000 ; Dawson and Wilby, 2001). Mainly because of ANNs' wide range of applicability and their capability to treat complicated and non-linear problems. In the hydrological forecasting context, recent experiments have reported that ANNs may offer a promising alternative for rainfall-runoff modeling (Kang et al., 1993 ; Shamseldin, 1997 ; Sajikumar and Thandaveswara, 1999 ; Tokar and Johnson, 1999 ; Dawson and Wilby, 2000 ; Kin et al., 2001, Riad et al., 2004), streamflow prediction (Karunanithi et al., 1994 ; Thirumalaiah and Deo, 1998 ; Zealand et al., 1999 ; Campolo et al., 1999 ; Sivakumar et al.,2002 ;), and reservoir inflow forecasting (Saad et al., 1996 ; Jain et al., 1999 ; Coulibaly et al., 2000).

The most widely used neural network model is the multilayer percepton (MLP), in which the connection weight training is normally completed by a back propagation (BP) learning algorithm (Rumelhart et al., 1986). The idea of weight training in MLPs is usually formulated as minimization of an error function, such as mean square error (MSE) between the network output and the target output over all examples, by iteratively adjusting connection weights. The back propagation algorithm is gradient descent in essence. Despite its popularity as an optimization tool for neural network training, the BP algorithm also has several drawbacks. For instance, the performance of the network learning is strictly dependent on the shape of the error surface, values of the initial connection weights, and the convergence to the global optimum is not guaranteed. These limitations cause the Back-propagation neural network (BPN) to become inconsistent and unpredictable on different applications (Maniezzo, 1994 ; Yasumasa et al., 1994 ; Sexton et al., 1998). Therefore, it is necessary to develop an effective technique for optimizing BPN in order to improve its forecasting performance.

Recently some investigations into neural network training using genetic algorithms have been successfully employed to overcome the inherent limitations of the BP (Sexton et al., 1998 ; Li et al., 2003). Designing neural networks through genetic algorithms has been investigated for many years and comprehensive reviews can be found in (Yao, 1999 ;Castillo et al.,2000). Genetic algorithms have been used with neural network to search for input variables (Hansen, et al., 1999 ; Potter, et al., 1999) or to determine the number of nodes or connections in the network (Dybowski, et al., 1997 ; Sexton et al., 1998 ; Heckerling, et al., 2004). However, There is still few article concerning evolutionary neural network in hydrological forecasting that has been published in the literature. Thus, this paper demonstrates a method that uses GA to train the neural network to forecast runoff at a Semiarid catchment in Morocco.

The outline of this paper is as follows. First, in Section 2, we introduce the multilayer feed-forward neural network model, the genetic algorithm, and methodology to hybrid real coded GA with a back propagation algorithm for neural network training. Next in Section 3, the application the GA-BP neural network for daily flow forecasting at the Ourika catchment is presented and compared with BP neural network using the same

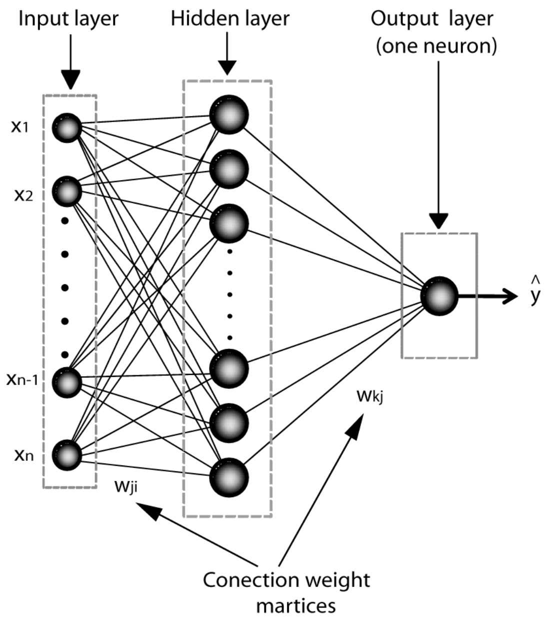

Figure 1. *Architecture of the neural network model used in this study.*

observed data. Finally in Section 4, some conclusions are drawn and future work is also proposed.

2. Methodology

2.1. ANN forecasting model

The network used in this study consists of an input layer, one hidden layer, and an output layer as shown in Fig.1. MLP can have more than one hidden layer, however theoretical works have shown that a single hidden layer is sufficient for ANNs to approximate any complex nonlinear function (Cybenko, 1989 ; Hornik et al., 1989). The input vectors to the selected ANN model and the number of hidden nodes were determined by the trial-and-error method commonly used for network design (Shamseldin, 1997 ; Zealand et al., 1999 ; Abrahart and See, 2000 ; Pan and Wang, 2004).
Referring to Fig.1, each neuron in the network operates by taking the sum of its weighted inputs and passing the result through a nonlinear activation function (transfer function). In this study the tangent function is used to transfer the values of the input layer nodes to the hidden layer nodes, whereas the linear transfer function is adopted to transfer the values from the hidden layer to the output layer. Each hidden neuron's output is calculated using Eq.(1), while the output neuron's output is calculated using Eq.(2),

$$X_j = \tanh(\sum_{i=1}^{n} x_i w_{ji} + w_{j0}) \quad [1]$$

$$\hat{y} = \sum_{j=1}^{m} X_j w_{kj} + w_{k0} \quad [2]$$

where x_i is the value of the input variable, w_{ji} and w_{kj} are connection weights between the input and hidden neuron, and between the hidden neuron and output neuron, w_{j0} and w_{k0} are the threshold (or bias) for the ith and kth neuron, respectively, and i, j and k are the number of neurons for the layers, respectively.
The value of $\hat{y}$ is compared with the desired output y for training data and the output differences are summed to generate an error function E defined as

$$E = \frac{1}{2} \sum_{p=1}^{N} (y_p - \hat{y}_p)^2 \qquad [3]$$

where n is the number of data in the training data set. Through training, it is hoped that the network learns, or generalizes, the nonlinear relationships that map inputs to outputs so that it can make reasonable estimates for data to which it was not exposed during the training process.

2.2. Methods and steps combining GA with BP

The combination of genetic algorithm and neural network for weight training consists of three major phases. The first phase is to decide the representation of connection weights, i.e., whether we use a binary strings form or directly use a real number form to represent the connection weights. Since this paper uses a real code genetic algorithm , what we have to do is just to set each neuron's connection weights and bias to its correspondent gene segments. However, it is difficult to reach convergence using the binary-encoded simple genetic algorithms (SGA) to solve the optimization problems that have too much design variables. So, a real code genetic algorithm was used to overcome the disadvantages of SGA. The second step is the evaluation on the fitness of these connection weights by constructing the corresponding neural network. The objective function (shown in Eq.3) is selected as the fitness function directly. Because of the generalization of ANN, its model can be used as the knowledge source for the optimization algorithm. This method can compute the objective function in real time. The evaluation criterion of the individuals was "LOWS- BEST". The third one is applying the evolutionary process such as selection, crossover, and mutation operations by a genetic algorithm according to its fitness. The evolution stops when the fitness is smaller than a predefined value.

The hybrid network learning process consists of two stages : firstly employing GA to search for optimal or approximate optimal connection weights and thresholds for the network, then using the BP to adjust the final weights. At first, the populations initialization is done ; then The fitness of every chromosome is evaluated by measuring the value of the total mean square error, see Eqs.(1)-(3). After evaluating all chromosomes, an intermediate population is created by extracting chromosomes from the current population using the reproduction (selection) operator. In this study, the roulette wheel selection based on ranking algorithm was applied for the reproduction operator. Chromosomes were selected in quantities according to their relative fitness after ranking in the roulette wheel operator and placement into the intermediate population. Finally, the population of the next generation is formed by applying the crossover and mutation operator to the chromosomes of the intermediate population. Then the new chromosomes reproduced by selection, crossover, and mutation operators are evaluated, and this procedure for evaluation and reproduction of all chromosomes was repeated until the stopping criterion is satisfied. The basis of GA is the continual improvement of the fitness of the population by means of genetic operators, as individuals are passed from one generation to the next. In this way, the ANN weights and thresholds are initialized as chromosome of best fitness population member.

This procedure is completed by applying a BP algorithm on the GA established initial connection weights and thresholds. If the BP stopping condition is false, the weights and thresholds are updated ; otherwise, they are saved and provided for future prediction of the flow.

3. Application Case

Data used in this paper are from the Ourika basin, located in semi-arid region of Marrakech and is the most important sub-catchment of Tensift basin drainage. The basin area is 503 km^2, and its mean annual rainfall is approximately 800 mm. The Rainfall and Runoff daily data at the average of Aghbalou, Oukaimeden and Arghbar stations were used for model investigation. The data contains information for a period of four years (2000 to 2003). Furthermore, data from 2000 to 2002 constitute the training set and the 365 remaining data is used in the testing phase.

The input vector is represented by rainfall and runoff values for the preceding 4 days, (i.e., $t-1, t-2, t-3, t-4$). Accordingly, the output vector represents the expected runoff value for day t ($\hat{Q}_t$). In this study data, the best ANN architecture was : 4-5-1 (4 input units, 5 hidden neurons, 1 output neuron). Before training and testing all source data are normalized into the range between -1 and 1, by using the maximum and minimum values of the variable over the whole data sets.

4. Results and discussion

Figure 3 shows the comparison between predicted and measured runoff values at training and testing phases by hybrid GA-ANN model using the daily data from the Ourika catchment. The GA-ANN algorithm was run with a population size of 100, uniform crossover probability was set to 0.9 and uniform mutation probability was set to 0.1. GA-ANN was trained by 80 generations, followed by a BP training procedure. The value of learning coefficient 0.01 and momentum correction factor 0.08 were used for the back propagation training algorithm.

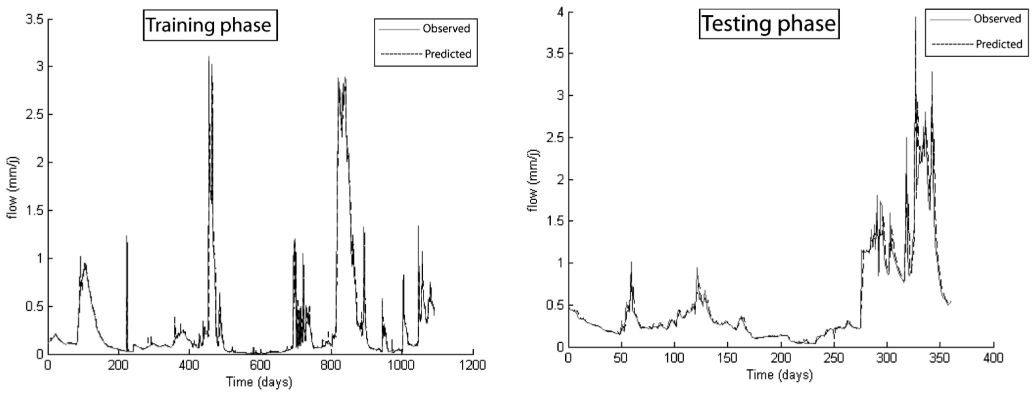

Figure 2. *Comparison between predicted and measured flow values at training and testing phases.*

In Fig.3 the output of the model, simulated with test data, shows a good agreement with the target. The simulation performance of the GA-ANN model was evaluated on the basis of Root Mean Square Error (RMSE) and efficiency coefficient R^2 (Nash and Sutcliffe, 1970). The parameters RMSE = 0.162 and $R^2 = 0.91$ suggest a very good performance. In general, a R^2 value greater than 0.9 indicates a very satisfactory model performance, while a R^2 value in the range 0.8-0.9 signifies a good performance and values less than 0.8 indicate an unsatisfactory model performance (Coulibaly and Baldwin, 2005).

In order to evaluate the performance of the genetic algorithm-based neural network, back propagation neural network was applied with the same data sets used in the GA-ANN model. Figures 4 show the extent of the match between the measured and predicted daily flow values by GA-BP and BP neural networks in term of a scatter diagram. Table 1 gives the RMSE and R^2 values for the two different models of the testing phases. It can be observed that the GA-BP exhibits better performance than those by the BP-ANN models.

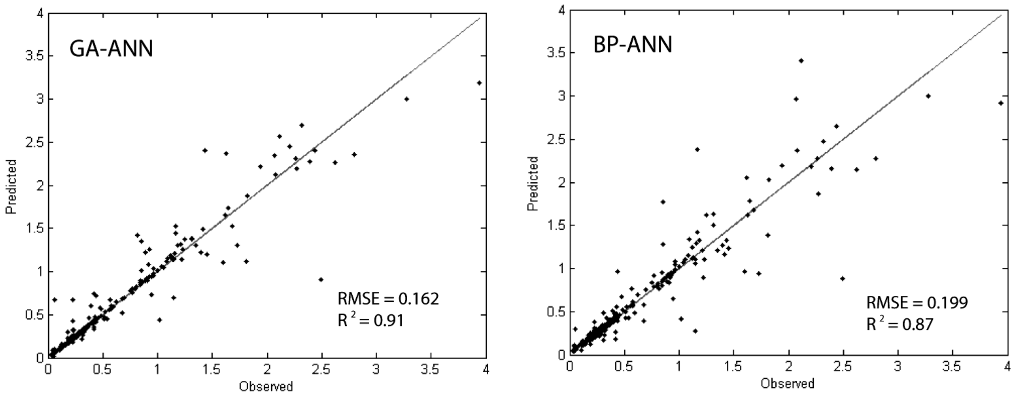

Figure 3. *The performance comparison of GA-BP and BP neural networks.*

GA-BP		BP	
RMSE	R^2	RMSE	R^2
0.162	0.91	0.199	0.87

Table 1: Comparison of GA-BP and BP neural networks

5. Conclusion

In this article, the advantages and the key issues of the genetic algorithm evolved neural network has been presented to model the rainfall-runoff relationship in Ourika catchment. Our methodology adopts a real coded GA strategy and hybrid with back propagation algorithm. The genetic operators are carefully designed to optimize the neural network, avoiding problems premature convergence and permutation. The experiment with real rainfall-runoff data have showed that the predictive performance of the proposed model is better than that of the traditional BP neural network. This has been supported by the analysis of the changes of connection weights and biases of the neural network.

6. Bibliographie

Abrahart, R.J., See, L., 2000. Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments. Hydrol. Process. 14, 2157-2172.

Campolo, M.,Andreussi, P., Soldati,A., 1999. River flood forecasting with a neural network model. Water Resour. Res. 35 (4), 1191-1197.

Castillo, P. A., Merelo, J. J., Prieto, A., Rivas, V., and Romero, G., 2000. G-prop : Global optimization of multilayer perceptrons using Gas. Neurocompution 35 :149-163.

Coulibaly, P., Baldwin, C. K., 2005. Nonstationary hydrological time series forecasting using nonlinear dynamic methods, J. Hydrol., pp. 307, 164-174.

Coulibaly, P., Anctilb, F., Bobeéc B., 2000. Daily reservoir inflow forecasting using artificial neural networks with stopped training approach. J. Hydrol. 230, 244-257.

Cybenko, G., 1989. Approximation by superposition of a sigmoidal function. Math. Control Signals Syst. 2, 303-314.

Dawson, C.W., Wilby, R.L., 2000. A comparison of artificial neural networks used for rainfall-runoff modelling, Hydrology and Earth Systems Sciences 3, 529-540.

Dawson, C.W., Wilby, R.L., 2001. Hydrological modeling using artificial neural networks. Progress in Physical Geography, 25(1), 80-108.

Dybowski, R., Weller, P., Chang, R., and Gant, V., 1997. Prediction of outcome in critically ill patients using artificial neural network synthesised by genetic algorithm. Lancet 347 :1146-1150.

Jain, S.K., Das, D., Srivastava, D.K., 1999. Application of ANN for reservoir inflow prediction and operation. J. Water Resour. Planning Mgmt ASCE 125 (5), 263-271.

Hansen, J. V., McDonald, J. B., and Nelson, R. D., 1999. Time Series Prediction with Genetic Algorithm Designed Neural Networks : An Experimental Comparison with Modern Statistical Models. Computational Intelligence, Vol. 15(3), pp. 171-184.

Heckerling, P. S., Gerber, B. S., Tape, T. G., and Wigton, R. S., 2004. Use of genetic algorithm for neural network to predict community-acquired pneumonia. Artif. Intell. Med. 30 :71-84.

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. Neural Networks 2(5), 359-366.

Kang, K.W., C.Y., Park and Kim, J.H., 1993. Neural network and its application to rainfall-runoff forecasting, Korean Journal of Hydroscienee 4, 1-9.

Karunanithi, N., Grenney, W.J., Whitley, D., Bovee, K., 1994. Neural networks for river flow prediction. J. Comp. Civil Engng ASCE 8 (2), 201-220.

Kin, C.L., Ball J.E., and Sharma, A., 2001. An application of artificial neural networks for rainfall forecasting, Mathl. Comput. Modelling 33 (6/7), 683-693.

Li, T. S., Su, C. T., and Chiang, T. L., 2003. Applying robust multiresponse quality engineering for parameter selection using a novel neural-genetic algorithm. Computers in Industry, 50, 113-122.

Maier, H.R., Dandy, G.C., 2000. Neural networks for the prediction and forecasting of water resources variables : a review of modeling issues and applications. Environmental Modelling and Software 15, 101-124.

Maniezzo, V., 1994. Genetic evolution of the topology and weight distribution of neural networks. IEEE Transaction on Neural Networks, 5, 39-53.

Nash, J.E., Sutcliffe, J. V., 1970. River flow forecasting through conceptual models I : A discussion of principles. J. Hydrol., 10, 282-290.

Pan, T.Y., Wang, R.Y., 2004. State space neural networks for short term rainfall-runoff forecasting. J. Hydrol., 297, 34-50.

Potter, S. R.,Miller, M. C., andMangold, L. A., 1999. Genetically engineered neural networks for predicting prostate cancer progression after radical prostatectomy., Urology. 54 :791-795.

Riad, S., Mania, J., Bouchaou, L. et Najjar, Y.,2004. Rainfall-Runoff Model Using an Artificial Neural Network Approach. Mathematical and Computer Modelling, 40, 839-846.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning internal representation by error propagation. In : Rumelhart, D.E., McClelland, J.L. (Eds.), Parallel Distributed Processing : Explorations in the Microstructure of Cognition, vol. 1. MIT Press, Cambridge, MA, pp. 318-362.

Saad, M., Bigras, P., Turgeon, A., Duquette, R., 1996. Fuzzy learning decomposition for the scheduling of hydroelectric power systems. Water Resour. Res. 32 (1), 179-186.

Sajikumar, N., Thandaveswara, B.S., 1999. A non-linear rainfall- runoff model using an artificial neural network. J. Hydrol. 216, 32-55.

Sexton, R. S., Dorsey, R. E., and Johnson, J. D., 1998. Toward global optimization of neural networks : A comparison of the genetic algorithm and back propagation. Decision Support Systems, 22, 171-185.

Shamseldin, A.Y., 1997. Application of a neural network technique to rainfall-runoff modelling. J. Hydrol. 199, 272-294.

Sivakumar, B., Jayawardena A.W. , and Fernando, T.M.K.G., 2002. River flow forecasting : Use of phase-space reconstruction and artificial neural networks approaches, Journal of Hydrology 265 (1-4), 225-245.

Thirumalaiah, K., Deo, M.C., 1998. Real-time flood forecasting using neural networks. Computer-Aided Civil Infrastruct. Engng 13 (2), 101-111.

Tokar, A.S., Johnson, P.A., 1999. Rainfall-runoff modeling using artificial neural networks. J.Hydrol. EngngASCE4 (3), 232-239.

Yasumasa, I., Hiroaki, K., Yoshiaki, S., Masaya, H., Toshikuni, N., and Yashio, I., 1994. Application of an improved genetic algorithm to the learning of neural networks. IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences, E77-A(4), 731-735.

Yao, X., 1999. Evolving artificial neural networks. IEEE Trans. Neural Netw. 87 :1423-1447.

Zealand, C.M., Burn, D.H., Simonovic, S.P., 1999. Short term streamflow forecasting using artificial neural networks. J. Hydrol. 214, 32-48.