Journal of Hydrology (2006) **329**, 363–367

available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/jhydrol

Particle swarm optimization training algorithm for ANNs in stage prediction of Shing Mun River

K.W. Chau *

Department of Civil and Structural Engineering, Hong Kong Polytechnic University, Hunghom, Kowloon, Hong Kong

Received 2 February 2005; received in revised form 9 January 2006; accepted 17 February 2006

KEYWORDS
Particle swarm optimization;
Artificial neural networks;
Shing Mun River

Summary An accurate water stage prediction allows the pertinent authority to issue a forewarning of the impending flood and to implement early evacuation measures when required. Existing methods including rainfall-runoff modeling or statistical techniques entail exogenous input together with a number of assumptions. The use of artificial neural networks (ANN) has been shown to be a cost-effective technique. But their training, usually with back-propagation algorithm or other gradient algorithms, is featured with certain drawbacks such as very slow convergence and easy entrapment in a local minimum. In this paper, a particle swarm optimization model is adopted to train perceptrons. The approach is applied to predict water levels in Shing Mun River of Hong Kong with different lead times on the basis of the upstream gauging stations or stage/time history at the specific station. It is shown that the PSO technique can act as an alternative training algorithm for ANNs.

Introduction

Flooding is a type of natural disaster that has been occurring, but can only be mitigated rather than completely solved. Prediction of river stages becomes an important research topic in hydrologic engineering. An accurate water stage prediction allows the pertinent authority to issue a forewarning of the impending flood and to implement early evacuation measures when required. Currently, environmental prediction and modeling includes a variety of approaches, such as rainfall-runoff modeling or statistical techniques such as autoregressive moving-average models (Box and Jenkins, 1976), which entail exogenous input together with a number of assumptions. Conventional numerical modeling addresses the physical problem by solving a highly coupled, non-linear, partial differential equation set. However, physical processes affecting flooding occurrence are highly complex and uncertain, and are difficult to be captured in some form of deterministic or statistical model.

During the past decade, artificial neural networks (ANNs), and in particular, feed forward backward propagation perceptrons, were widely applied in different fields (Chau and Cheng, 2002). It was claimed that the multi-layer perceptrons can be trained with non-linear transfers to

* Tel.: +852 27666014.
E-mail address: cekwchau@polyu.edu.hk.

0022-1694/$ - see front matter
doi:10.1016/j.jhydrol.2006.02.025

approximate and accurately generalize virtually any smooth, measurable function whilst taking no prior assumptions concerning the data distribution (Rumelhart et al., 1986). Several characteristics, including built-in dynamism in forecasting, data-error tolerance, and lack of requirements of any exogenous input, render ANNs attractive for use in river stage prediction in hydrologic engineering. Thirumalaiah and Deo (1998) depicted the use of a conjugate gradient ANN in real-time forecasting of water levels, with verification of untrained data. Liong et al. (2000) demonstrated that a feed forward ANN is a highly suitable flow prediction tool yielding a very high degree of water level prediction accuracy in Bangladesh. Luk et al. (2000) studied optimal model lag and spatial inputs to artificial neural network for rainfall forecasting. Lekkas et al. (2001) compared ANNs with transfer functions in a flow routing application. Balkhair (2002) determined aquifer parameters for large diameter wells using neural network approach. Bazartseren et al. (2003) showed that both ANN and neuro-fuzzy systems outperformed the linear statistical models for short-term water level predictions on two different river reaches in Germany. Riad et al. (2004) developed and used a multi-layer perceptron ANN to model the rainfall-runoff relationship, in a catchment located in a semiarid climate in Morocco. Sarangi and Bhattacharya (2005) compared several ANN and regression models for sediment loss prediction from Banha watershed in India. Although the back propagation (BP) algorithm is commonly used in recent years to perform the training task, some drawbacks are often encountered in the use of this gradient-based method. They include: the training convergence speed is very slow and easy entrapment in a local minimum. Haykin (1999) discussed several data-driven optimization training algorithms, such as Levenberg–Marquardt algorithm and scaled conjugate gradient algorithm, which may overcome these drawbacks. Rogers et al. (1995) used the genetic algorithm for optimal field-scale groundwater remediation together with ANN. Kumar et al. (2004) employed the Bayesian regularization for neural network training in order to improve the performance in pulse radar detection. The PSO technique can act as an alternative training algorithm for ANNs that can be used for hydrologic applications.

Particle swarm optimization (PSO) algorithm, with capability to optimize complex numerical functions, is initially developed as a tool for modeling social behavior (Kennedy and Eberhart, 1995 and Kennedy, 1997). Moreover, it is recognized as an evolutionary technique under the domain of computational intelligence (Clerc and Kennedy, 2002). In this paper, a PSO-based neural network approach for river stage prediction is developed by adopting PSO to train multi-layer perceptrons. It is then used to predict real-time water levels in the Shing Mun River of Hong Kong with different lead times on the basis of the upstream gauging stations or stage/time history at the specific station.

Multi-layer feed-forward perceptron

A multi-layer feed-forward perceptron represents a non-linear mapping between input vector and output vector through a system of simple interconnected neurons to every node in the next and previous layer (Rumelhart et al., 1986). The output of a neuron is scaled by the connecting weight and fed forward to become an input through a non-linear activation function to the neurons in the next layer of network. In the course of training, the perceptron is repeatedly presented with the training data. The weights in the network are then adjusted until the errors between the target and the predicted outputs are small enough, or a pre-determined number of epochs is passed. The perceptron is then validated by an input vector not belonging to the training pairs. The training processes of ANN are usually complex and high dimensional problems.

Particle swarm optimization (PSO)

Lying somewhere between evolutionary programming and genetic algorithms, PSO is an optimization paradigm that mimics the ability of human societies to process knowledge. It has roots in two main component methodologies: artificial life (such as bird flocking, fish schooling and swarming); and, evolutionary computation (Clerc and Kennedy, 2002).

PSO algorithm

The principle of PSO algorithm is founded on the assumption that potential solutions will be flown through hyperspace with acceleration towards more optimum solutions. It is a populated search method for optimization of non-linear functions resembling the movement of organisms in a bird flock or fish school. Candidate solutions to the problem are termed particles or individuals. Instead of employing genetic operators, the evolution of generations of a population of these individuals in such a system is by cooperation and competition among the individuals themselves. In essence, each particle adjusts its flying based on the flying experiences of both itself and its companions. During the process, it keeps track of its coordinates in hyperspace which are associated with its previous best fitness solution, and also of its counterpart corresponding to the overall best value acquired thus far by any other particle in the population.

In the algorithm, vectors are taken as representation of particles since most optimization problems are convenient for such variable presentations. The population is responding to the quality factors of the previous best individual values and the previous best group values. The allocation of responses between the individual and group values ensures a diversity of response. Its major advantages are the relatively simple and computationally inexpensive coding and its adaptability corresponding to the change of the best group value. The stochastic PSO algorithm has been found to be able to find the global optimum with a large probability and high convergence rate (Clerc and Kennedy, 2002). Hence, it is adopted to train the multi-layer perceptrons, within which matrices learning problems are dealt with.

Adaptation to network training

A three-layered perceptron is chosen for this application case. Here, $W^{[1]}$ and $W^{[2]}$ represent the connection weight matrix between the input layer and the hidden layer, and that between the hidden layer and the output layer, respectively. When a PSO is employed to train the multi-layer perceptrons, the ith particle is denoted by

$$W_i = \{W_i^{[1]}, W_i^{[2]}\}. \tag{1}$$

The position representing the previous best fitness value of any particle is recorded and denoted by

$$P_i = \{P_i^{[1]}, P_i^{[2]}\}. \tag{2}$$

If, among all the particles in the current population, the index of the best particle is represented by the symbol b, then the best matrix is denoted by

$$P_b = \{P_b^{[1]}, P_b^{[2]}\}. \tag{3}$$

The velocity of particle i is denoted by

$$V_i = \{V_i^{[1]}, V_i^{[2]}\}. \tag{4}$$

If m and n represent the index of matrix row and column, respectively, the manipulation of the particles are as follows

$$V_i''^{[j]}(m,n) = V_i^{[j]}(m,n) + \{r\alpha[P_i^{[j]}(m,n) - W_i^{[j]}(m,n)] + s\beta[P_b^{[j]}(m,n) - W_i^{[j]}(m,n)]\}/t \tag{5}$$

and

$$W_i''^{[j]} = W_i^{[j]} + V_i^{[j]}t, \tag{6}$$

where $j = 1, 2$; $m = 1, \ldots, M_j$; $n = 1, \ldots, N_j$; M_j and N_j are the row and column sizes of the matrices W, P, and V; r and s are positive constants; α and β are random numbers in the range from 0 to 1; t is the time step between observations and is often taken as unity; V'' and W'' represent the new values. Eq. (5) is employed to compute the new velocity of the particle based on its previous velocity and the distances of its current position from the best experiences both in its own and as a group. In the context of the social behavior, the cognition part, i.e., the second element on the right hand side of Eq. (5), represents the private thinking of the particle itself whilst the social part, i.e., the third element on the right hand side of Eq. (5), denotes the collaboration among the particles as a group. Eq. (6) then determines the new position according to the new velocity.

The fitness of the ith particle is expressed in term of an output mean squared error of the neural networks as follows

$$f(W_i) = \frac{1}{S}\sum_{k=1}^{S}\left[\sum_{l=1}^{O}\{t_{kl} - p_{kl}(W_i)\}^2\right], \tag{7}$$

where f is the fitness value, t_{kl} is the target output; p_{kl} is the predicted output based on W_i; S is the number of training set samples; and, O is the number of output neurons.

The study area

The model is applied to study the potential flood hazards in the Shing Mun River network, Hong Kong. Details regarding the location map of the Shing Mun River and its tributary nullahs can be found in Chau and Lee (1991a,b) and Chau and Chen (2001). The main conveyance channel is of trapezoidal shape with side slope of 1 in 1.5 along most length. The three minor streams, i.e., the Tin Sam, Fo Tan and Siu Lek Yuen nullahs, form tributaries of the river. Surface water from an extensive catchment with an area of approximately 5200 ha flows into Sha Tin Hoi via the Shing Mun River. The maximum daily runoff as a percentage of the annual flow is typically less than 5% (Chau and Lee, 1991a,b).

In this study, water levels at Fo Tan are forecasted with a lead time of 1 and 2 days based on the measured daily levels there and at the upstream station (Tin Sam) with a distance about 2 km apart. The data available at these locations pertain to continuous stages from 1999 to 2002, in the form of daily water levels. The first two years' data are used for training whilst the final year data are used to validate the network results. It is ensured that the data series chosen for training and validation comprised both high and low discharge periods of the year and also rapid changes in water stages.

Two separate models are developed. The perceptron has an input layer with one neuron, a hidden layer with three neurons, and output layer with one neuron. Similar to Thirumalaiah and Deo (1998), the input neuron represents the water stage at the current day whilst the output node denotes the water stage after 1 day or 2 days. This approach is found to improve the results than its counterpart when the output layer has two neurons with both 1-day and 2-days ahead forecast. During the training stage, the single input neuron represents time series information of water stages. The number of nodes in the hidden layer is set by trial and error during the course of training to whatever size leads to the most accurate predictions.

Twenty thousand training epochs are adopted as the stopping criteria. The sigmoid function is adopted at the hidden and output nodes. All source data are normalized into the range between 0 and 1, by using the maximum and minimum values of the variable over the whole data sets. In the PSO-based perceptron, the number of population is set to be 40 whilst the maximum and minimum velocity values are 0.25 and -0.25, respectively. These values are obtained by trial and error. In order to evaluate the performance of the model in longer-term forecast, a third model with 7-days ahead forecast is also tried.

Results and discussions

The PSO-based multi-layer ANN is evaluated along with a commonly used standard BP-based network. In order to furnish a comparable initial state, the training process of the BP-based perceptron commences from the best initial population of the corresponding PSO-based perceptron. Three goodness-of-fit measures, namely, the coefficient of efficiency (R^2), which is 1 $-$ the sum of squared errors divided by the total sum of squares, root mean squared error (RMSE) and mean relative error (MRE) are adopted to evaluate the model performance. Tables 1 and 2 show comparisons of the results of network for the two different perceptrons based on data at the same station and at different station, respectively. It can be observed that the PSO-based perceptron exhibits better performance in the training process as well as better prediction ability in the validation process than those by the BP-based perceptron. Moreover, forecasting at Fo Tan made by using the data collected at the upstream station (Tin Sam) is generally better compared to the data collected at the same location. This can possibly be explained by the lead time required for the flow to travel

Table 1 Results for forecasting at Fo Tan based on data at the same station

Algorithm	Lead time (days)	Goodness-of-fit measure					
		Training			Validation		
		R^2	RMSE	MRE	R^2	RMSE	MRE
BP-based	1	0.96	0.16	0.09	0.96	0.21	0.12
	2	0.93	0.24	0.15	0.92	0.29	0.24
	7	0.89	0.35	0.27	0.88	0.43	0.38
PSO-based	1	0.99	0.08	0.04	0.99	0.12	0.06
	2	0.99	0.14	0.07	0.98	0.16	0.09
	7	0.95	0.25	0.18	0.92	0.32	0.21

Table 2 Results for forecasting at Fo Tan based on data at Tin Sam (upstream of Fo Tan)

Algorithm	Lead time (days)	Goodness-of-fit measure					
		Training			Validation		
		R^2	RMSE	MRE	R^2	RMSE	MRE
BP-based	1	0.97	0.14	0.07	0.96	0.16	0.10
	2	0.94	0.21	0.12	0.93	0.24	0.20
	7	0.91	0.30	0.22	0.89	0.41	0.32
PSO-based	1	0.99	0.07	0.04	0.99	0.09	0.05
	2	0.99	0.11	0.06	0.98	0.14	0.08
	7	0.96	0.22	0.16	0.93	0.29	0.18

from upstream section to downstream section and the correlation between the water stages at the two locations.

Conclusions

This paper presents a PSO-based perceptron approach for real-time prediction of water stage in a river with different lead times on the basis of the upstream gauging stations or stage/time history at the specific station. It is shown from the training and verification simulation that the water stage prediction results are more accurate when compared with the commonly used BP-based perceptron. Moreover, forecasting at Fo Tan made by using the data collected at the upstream station is generally better compared to the data collected at the same location. The initial result shows that the PSO technique can act as an alternative training algorithm for ANNs that can be used for hydrologic applications. Since it might not be able to draw concrete conclusions from this pilot study, more rigorous testing on more complex problems will be performed in future works.

Acknowledgement

This research was supported by the Central Research Grant of Hong Kong Polytechnic University (G-T592).

References

Balkhair, K.S., 2002. Aquifer parameters determination for large diameter wells using neural network approach. Journal of Hydrology 265 (1–4), 118–128.

Bazartseren, B., Hildebrandt, G., Holz, K.-P., 2003. Short-term water level prediction using neural networks and neuro-fuzzy approach. Neurocomputing 55 (3–4), 439–450.

Box, G.E.P., Jenkins, G.M., 1976. Time Series Analysis Forecasting and Control. Holden-Day, San Francisco.

Chau, K.W., Chen, W., 2001. A fifth generation numerical modelling system in coastal zone. Applied Mathematical Modelling 25 (10), 887–900.

Chau, K.W., Cheng, C.T., 2002. Real-time prediction of water stage with artificial neural network approach. Lecture Notes in Artificial Intelligence 2557, 715.

Chau, K.W., Lee, J.H.W., 1991a. Mathematical modelling of Shing Mun River network. Advances in Water Resources 14 (3), 106–112.

Chau, K.W., Lee, J.H.W., 1991b. A microcomputer model for flood prediction with applications. Microcomputers in Civil Engineering 6 (2), 109–121.

Clerc, M., Kennedy, J., 2002. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation 6 (1), 58–73.

Haykin, S., 1999. Neural Networks, A Comprehensive Foundation. Prentice Hall, Upper Saddle River.

Kennedy, J., 1997. The Particle swarm: Social adaptation of knowledge. In: Proceedings of the 1997 International Conference on Evolutionary Computation, Indianapolis, pp. 303–308.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of the 1995 IEEE International Conference on Neural Networks, Perth, pp. 1942–1948.

Kumar, P., Merchant, S.N., Desai, U.B., 2004. Improving performance in pulse radar detection using Bayesian regularization for neural network training. Digital Signal Processing 14 (5), 438–448.

Lekkas, D.F., Imrie, C.E., Lees, M.J., 2001. Improved non-linear transfer function and neural network methods of flow routing for real-time forecasting. Journal of Hydroinformatics 3 (3), 153–164.

Liong, S.Y., Lim, W.H., Paudyal, G.N., 2000. River stage forecasting in Bangladesh: Neural network approach. Journal of Computing in Civil Engineering, ASCE 14 (1), 1–8.

Luk, K.C., Ball, J.E., Sharma, A., 2000. A study of optimal model lag and spatial inputs to artificial neural network for rainfall forecasting. Journal of Hydrology 227 (1—4), 56—65.

Riad, S., Mania, J., Bouchaou, L., Najjar, Y., 2004. Rainfall-runoff model using an artificial neural network approach. Mathematical and Computer Modelling 40 (7—8), 839—846.

Rogers, L.L., Dowla, F.U., Johnson, V.M., 1995. Optimal field-scale groundwater remediation using neural networks and the genetic algorithm. Environmental Science and Technology 29 (5), 1145—1155.

Rumelhart, D.E., Hinton, E., Williams, J., 1986. Learning internal representation by error propagation. Parallel Distributed Processing 1, 318—362.

Sarangi, A., Bhattacharya, A.K., 2005. Comparison of artificial neural network and regression models for sediment loss prediction from Banha watershed in India. Agricultural Water Management 78 (3), 195—208.

Thirumalaiah, K., Deo, M.C., 1998. River stage forecasting using artificial neural networks. Journal of Hydrologic Engineering, ASCE 3 (1), 26—32.