Lecture Notes in Artificial Intelligence, Vol. 3029, 2004, pp. 1166-1173

River Stage Forecasting with Particle Swarm Optimization

Kwokwing Chau

Department of Civil and Structural Engineering, Hong Kong Polytechnic University, Hunghom, Kowloon, Hong Kong
cekwchau@polyu.edu.hk

Abstract. 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 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 easily getting stuck in a local minimum. In this paper, a particle swarm optimization model is adopted to train perceptrons. The approach is demonstrated to be feasible and effective by predicting real-time 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 from the verification simulations that faster and more accurate results can be acquired.

1 Introduction

Flooding is a type of natural disaster that has been occurring for centuries, 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, 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 which demands huge computing cost and time. 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, the artificial neural networks (ANN), and in particular, the feed forward backward propagation perceptrons, are widely applied in different fields. It is claimed that the multi-layer perceptrons can be trained to approximate and accurately generalize virtually any smooth, measurable function whilst taking no prior assumptions concerning the data distribution. Characteristics, including built-in dynamism in forecasting, data-error tolerance, and lack of requirements of any exogenous input, render it attractive for use in river stage prediction in hydrologic engineering. Thirumalaiah and Deo [1] depict the use of a conjugate gradient ANN in real-time forecasting of water levels, with verification of untrained data. Liong et al. [2] demonstrate that a feed forward ANN is a highly suitable flow prediction tool yielding a very high degree of water level prediction accuracy in Bangladesh. Chau and Cheng [3] describe the sensitivity of various network characteristics for real-time prediction of water stage with the ANN approach in a river in Hong Kong. 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; it is easily to get stuck in a local minimum. Different algorithms have been proposed in order to resolve these drawbacks, yet the results are still not fully satisfactory [4].

Particle swarm optimization (PSO) is a method for optimizing hard numerical functions based on metaphor of human social interaction [5-6]. Although it is initially developed as a tool for modeling social behavior, the PSO algorithm has been recognized as a computational intelligence technique intimately related to evolutionary algorithms [7-8]. 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 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.

2 Multi-layer Feed-forward Perceptron

A multi-layer feed-forward perceptron represents a nonlinear mapping between input vector and output vector through a system of simple interconnected neurons. It is fully connected to every node in the next and previous layer. The output of a neuron is scaled by the connecting weight and fed forward to become an input through a nonlinear 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 presenting with an input vector not belonging to the training pairs. The training processes of ANN are usually complex and high dimensional problems. The commonly used gradient-based BP algorithm is a local search method, which easily falls into local optimum point during training.

3 Particle Swarm Optimization (PSO)

Particle swarm optimization (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. The key concept of PSO is that potential solutions are flown through hyperspace and are accelerated towards better or more optimum solutions.

3.1 PSO Algorithm

PSO is a populated search method for optimization of continuous nonlinear functions resembling the movement of organisms in a bird flock or fish school. Its paradigm can be implemented in a few lines of computer code and is computationally inexpensive in terms of both memory requirements and speed. It lies somewhere between evolutionary programming and genetic algorithms. As in evolutionary computation paradigms, the concept of fitness is employed and candidate solutions to the problem are termed particles or sometimes individuals. A similarity between PSO and a genetic algorithm is the initialization of the system with a population of random solutions. 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. Moreover, a randomized velocity is assigned to each potential solution or particle so that it is flown through hyperspace. The adjustment by the particle swarm optimizer is ideally similar to the crossover operation in genetic algorithms whilst the stochastic processes are close to evolutionary programming. The stochastic factors allow thorough search of spaces between regions that are spotted to be relatively good whilst the momentum effect of modifications of the existing velocities leads to exploration of potential regions of the problem domain.

There are five basic principles of swarm intelligence: (1) proximity; (2) quality; (3) diverse response; (4) stability; and, (5) adaptability. The n-dimensional space calculations of the PSO concept are performed over a series of time steps. 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. The principle of stability is adhered to since the population changes its state if and only if the best group value changes. It is adaptive corresponding to the change of the best group value.

In essence, each particle adjusts its flying based on the flying experiences of both itself and its companions. 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. Vectors are taken as presentation of particles since most optimization problems are convenient for such variable presentations. The stochastic PSO algorithm has been found to be able to find the global optimum with a large probability and high convergence rate. Hence, it is adopted to train the multi-layer perceptrons, within which matrices learning problems are dealt with.

3.2 Adaptation to Network Training

A three-layered preceptron 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 preceptrons, the i-th 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 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

$$\begin{aligned} V_i^{[j]}(m,n) = V_i^{[j]}(m,n) + r\alpha[P_i^{[j]}(m,n) - W_i^{[j]}(m,n)] \\ + s\beta[P_b^{[j]}(m,n) - W_i^{[j]}(m,n)] \end{aligned} \tag{5}$$

and

$$W_i^{[j]} = W_i^{[j]} + V_i^{[j]} \tag{6}$$

where j = 1, 2; m = 1, …, M_j; n= 1, …, 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. Equation (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 social behavior, the cognition part $r\alpha[P_i^{[j]}(m,n) - W_i^{[j]}(m,n)]$ represents the private thinking of the particle itself whilst the social part $s\beta[P_b^{[j]}(m,n) - W_i^{[j]}(m,n)]$ denotes the collaboration among the particles as a group. Equation (6) then determines the new position according to the new velocity.

The fitness of the i-th 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.

4 The Study Area

The system is applied to study the tidal dynamics and potential flood hazards in the Shing Mun River network, Hong Kong. Details regarding the location of the Shing Mun River and its tributary nullahs can be found in [9-17]. The existing Shing Mun River has been trained for a length of about 2840m, from the bell-mouth outlet of Lower Shing Mun Dam to Sha Tin Tsuen. The three minor streams, i.e. the Tin Sam, Fo Tan and Siu Lek Yuen nullahs, form tributaries of the extended 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 flow at the river for a 200-year storm is about 1500 m^3/s.

In this study, water levels at Fo Tan is forecasted with a lead time of 1 and 2 days based on the measured daily levels there and at Tin Sam. The data available at these locations pertain to continuous stages from 1999 to 2002, in the form of daily water levels. In total, 1095 pairs of daily levels were available, of which 730 were used for training and 365 were used to validate the network results with the observations. 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.

The perceptron has an input layer with one neuron, a hidden layer with three neurons, and output layer with two neurons. The input neuron represents the water stage at the current day whilst the output nodes include the water stages after 1 day and 2 days, respectively. 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.

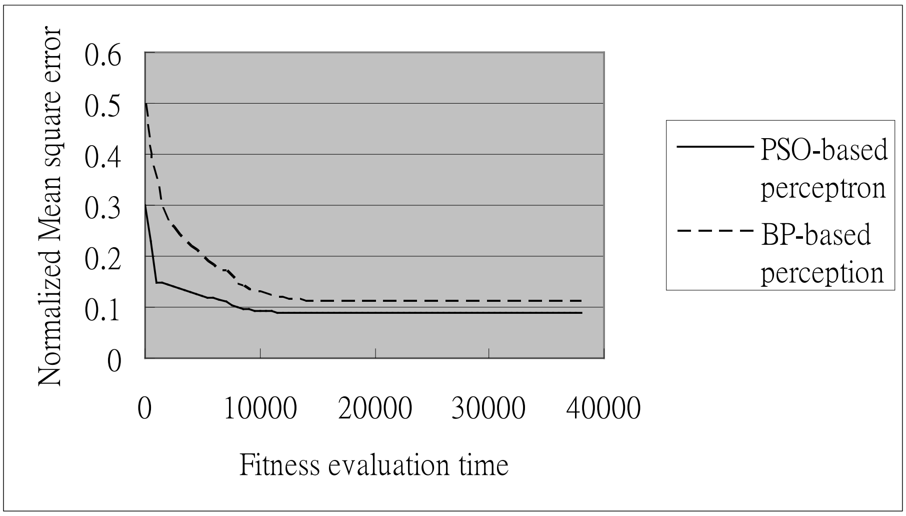

Fig. 1. Relationships between the normalized mean square error and fitness evaluation time during training for PSO-based and BP-based perceptrons

5 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. Figure 1 shows the relationships between the normalized mean square error and fitness evaluation time during training for PSO-based and BP-based perceptrons whilst Figure 2 shows the 2 day lead time normalized water level prediction by both perceptrons in the validation process. Table 1 and Table 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.

The fitness evaluation time here for the PSO-based perceptron is equal to the product of the population with the number of generations. It can be observed that the PSO-based perceptron exhibits much better and faster convergence performance in the training process as well as better prediction ability in the validation process than those by the BP-based perceptron. It can be concluded that the PSO-based perceptron performs better than 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.

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

Algorithm	Coefficient of correlation			
	Training		Validation	
	1 day ahead	2 days ahead	1 day ahead	2 days ahead
BP-based	0.945	0.913	0.934	0.889
PSO-based	0.974	0.965	0.956	0.944

Table 2. Results for forecasting at Fo Tan based on data at Tin Sam

Algorithm	Coefficient of correlation			
	Training		Validation	
	1 day ahead	2 days ahead	1 day ahead	2 days ahead
BP-based	0.967	0.934	0.954	0.894
PSO-based	0.989	0.982	0.983	0.974

6 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 demonstrated that the novel optimization algorithm, which is able to provide model-free estimates in deducing the output from the input, is an appropriate forewarning tool. It is shown from the training and verification simulation that the water stage prediction results are more accurate and are obtained in relatively short computational time, when compared with the commonly used BP-based perceptron. Both the above two factors are important in real-time water resources management. It can be concluded that the PSO-based perceptron performs better than the 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. Future research can be directed towards forecasting river stages by using the more empirical hydrological and rainfall data at the upstream catchment area in order to further extend the lead time of forewarning.

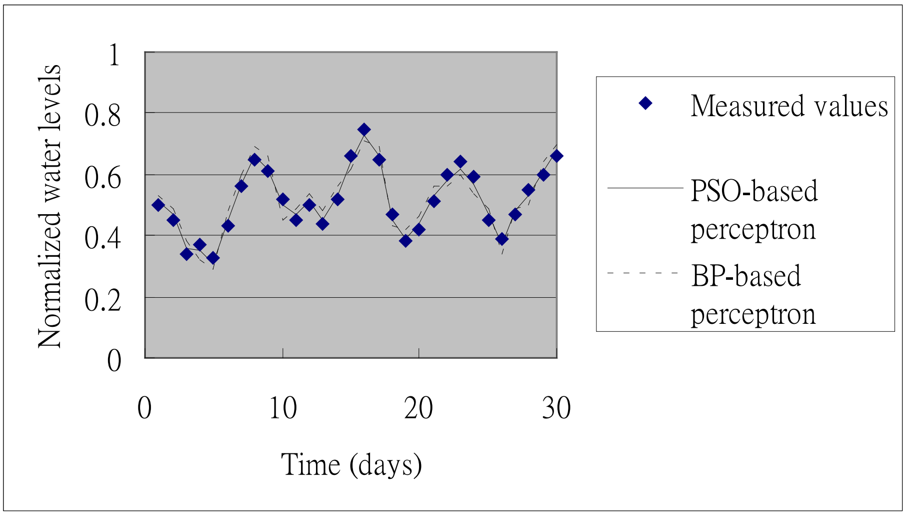

Fig. 2. 2 day lead time water level prediction by both perceptrons in the validation process

7 Acknowledgement

This research was supported by the Central Research Grant of Hong Kong Polytechnic University (G-T592) and the Internal Competitive Research Grant of Hong Kong Polytechnic University (A-PE26).

References

1. Thirumalaiah, K., Deo, M.C.: River Stage Forecasting Using Artificial Neural Networks. Journal of Hydrologic Engineering, ASCE **3(1)** (1998) 26-32
2. Liong, S.Y., Lim, W.H., Paudyal, G.N.: River Stage Forecasting in Bangladesh: Neural Network Approach. Journal of Computing in Civil Engineering, ASCE **14(1)** (2000) 1-8
3. Chau, K.W., Cheng, C.T.: Real-time Prediction of Water Stage with Artificial Neural Network Approach. Lecture Notes in Artificial Intelligence, **2557** (2002) 715-715

4. Govindaraju, R., Rao, A. (Ed.): Artificial Neural Networks in Hydrology. Kluwer Academic Publishers, Dordrecht (2000)
5. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. Proceedings of the 1995 IEEE International Conference on Neural Networks. Perth (1995) 1942-1948
6. Kennedy, J.: The Particle Swarm: Social Adaptation of Knowledge. Proceedings of the 1997 International Conference on Evolutionary Computation. Indianapolis (1997) 303-308
7. Clerc, M., Kennedy, J.: The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional Complex Space. IEEE Transactions on Evolutionary Computation **6(1)** (2002) 58-73
8. Kennedy, J., Eberhart, R., Shi, Y.: Swarm Intelligence. Morgan Kaufmann Publishers, San Francisco (2001)
9. Chau, K.W.: Manipulation of Numerical Coastal Flow and Water Quality Models. Environmental Modelling & Software **18(2)** (2003) 99-108
10. Chau, K.W.: Intelligent Manipulation of Calibration Parameters in Numerical Modeling. Advances in Environmental Research **8(3-4)** (2004) 467-476
11. Chau, K.W.: A Prototype Knowledge-Based System on Unsteady Open Channel Flow in Water Resources Management. Water International **29(1)** (2004) (in press)
12. Chau, K.W., Chen, W.: A Fifth Generation Numerical Modelling System in Coastal Zone. Applied Mathematical Modelling **25(10)** (2001) 887-900
13. Chau, K.W., Chen, W.: An Example of Expert System on Numerical Modelling System in Coastal Processes. Advances in Engineering Software **32(9)** (2001) 695-703
14. Chau, K.W., Cheng, C., Li, C.W.: Knowledge Management System on Flow and Water Quality Modeling. Expert Systems with Applications **22(4)** (2002) 321-330
15. Chau, K.W., Lee, J.H.W.: Mathematical Modelling of Shing Mun River Network. Advances in Water Resources **14(3)** (1991) 106-112
16. Chau, K.W., Lee, J.H.W.: A Microcomputer Model for Flood Prediction with Applications. Microcomputers in Civil Engineering **6(2)** (1991) 109-121
17. Chau, K.W., Yang, W.W.: Development of an Integrated Expert System for Fluvial Hydrodynamics. Advances in Engineering Software **17(3)** (1993) 165-172