International Journal of Environment and Pollution, Vol. 28, No. 3-4, 2006, pp. 261-273

A flood forecasting neural network model with genetic algorithm

C.L. Wu[1] & K.W. Chau[2*]
[1]*Changjiang Water Resources Commission, 430010, Wuhan, HuBei, China*
[2]*Department of Civil and structural Engineering, Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China*
**Corresponding author*
Email: cekwchau@polyu.edu.hk

Abstract

It will be useful to attain a quick and accurate flood forecasting, particularly in a flood-prone region. The accomplishment of this objective can have far reaching significance by extending the lead time for issuing disaster warnings and furnishing ample time for citizens in vulnerable areas to take appropriate action, such as evacuation. In this paper, a novel hybrid model based on recent artificial intelligence technology, namely, a genetic algorithm (GA)-based artificial neural network (ANN), is employed for flood forecasting. As a case study, the model is applied to a prototype channel reach of the Yangtze River in China. Water levels at downstream station, Han-Kou, are forecasted on the basis of water levels with lead times at the upstream station, Luo-Shan. An empirical linear regression model, a conventional ANN model and a GA model are used as the benchmarks for comparison of performances. The results reveal that the hybrid GA-based ANN algorithm, under cautious treatment to avoid overfitting, is able to produce better accuracy performance, although in expense of additional modeling parameters and possibly slightly longer computation time.

Keywords: *Flood forecasting model; Hybrid algorithms; Artificial neural networks; Genetic algorithms*

Biographical notes

C.L. Wu is currently working on project planning, design and research in Changjiang Water Resources Commission of China. He received his Bachelor of Irrigation and Drainage Enginnering degree from Wuhan University in 1998. He then continued his studies in Wuhan University and obtained his Master of Water Resources Engineering in 2000. He previously worked as Research Assistant in Department of Civil and Structural Engineering of The Hong Kong Polytechnic University.

Dr. K.W. Chau is currently an Associate Professor at the Department of Civil and Structural Engineering of The Hong Kong Polytechnic University. He received both his BSc(Eng) and MPhil in Civil Engineering from the Department of Civil Engineering, The University of Hong Kong, Hong Kong. He received his PhD in Civil engineering from the Department of Civil Engineering, The University of Queensland, Australia. He is very active in undertaking research works and the scope of his research interest is very broad, covering numerical flow modeling, water quality modeling, hydrological modeling, knowledge-based system development and knowledge management.

1. Introduction

It will be useful to attain a quick and accurate flood forecasting, particularly in a flood-prone region, for the timely issue of disaster warnings well in advance in order to furnish ample time for the evacuation of populations endangered by imminent rising water levels. In general, there are two main categories of numerical models for flood propagation in a river reach, namely, conceptual or physically based models and empirical “black-box” models. In conceptual models, the flood propagation process is usually described by the de Saint Venant equations comprising two partial differential equations on continuity and momentum which are, however, not amenable to any analytical solutions.

Nowadays, many conceptually based numerical schemes (Chau and Lee 1991a & 1991b) are available which can represent the mechanisms of the hydrological process. Their usual drawbacks are their requirements on large amount of ambient data, such as characteristics of terrain and river networks, rainfall, runoff and so on, during the calibration of these hydrological model. Yet, these data are often unavailable or may not be easily gleaned. Thus, in real-time forecasting, sophisticated physical models may not be useful because of the demand on both exhaustive data and the excessive computation time.

In real practice, empirical models, based on an evidence of relationships manifested in historical records of input and output records without analyzing the internal structure of the physical process, might have their edges. In fact, the main emphasis of the planning authority and hydraulic engineers is often about making accurate and timely predictions at specific locations. A simple “black box” model might have higher preference and become more useful in identifying a direct correlation between inputs and outputs in flood forecasting.

Owing to the above reason and the advances in artificial intelligence technologies, different nonlinear approaches, such as, artificial neural network (ANN), and genetic algorithm (GA), have been used in solving flood forecasting problems in recent years. Smith and Eli (1995) applied a back-propagation ANN model to predict discharge and time to peak over a hypothetical watershed. Olivera and Loucks (1997) employed a GA to formulate operating rules for multireservoir systems. Wardlaw and Sharif (1999) evaluate a GA for optimal reservoir system operation. Tokar and Johnson (1999) compared ANN models with regression and simple conceptual models. Liong *et al.* (2000) employed an ANN approach for river stage forecasting in Bangladesh. The ASCE Task Committee (2000) summarized the state-of-the-art applications of ANN in hydrology and posed some future directions. Chau and Cheng (2002) performed a real-time prediction of water stage with ANN approach using an improved back propagation algorithm. Cheng *et al.* (2002) applied a GA to calibrate conceptual rainfall-runoff models. Chau (2002) calibrated flow and water quality modeling using a GA. Chau (2004a & b) employed particle swarm optimization in river stage forecasting and rainfall-runoff correlation. However, so far, ANN and GA have only been employed in flood forecasting problems individually. It appears that hybrid combinations of these algorithms have never been used in flood forecasting.

This paper presents the application of a hybrid algorithm, namely, a genetic algorithm-based artificial neural network (ANN-GA), for flood forecasting in a real prototype river channel in China. An evaluation of its performance is made in comparison with several benchmark models, namely, a linear regression (LR) model, a conventional ANN model, and a conventional GA model. The organization of this paper is as follows: the algorithm of the ANN-GA model is introduced; the study area is depicted; the prediction result based on this hybrid algorithm is compared with those of the benchmarking models; and finally, conclusions are drawn.

2. Algorithm of ANN-GA Flood Forecasting Model

2.1 Artificial neural network (ANN)

An artificial neural network (ANN) has the ability to train and learn the outputs from the inputs by mimicking the function of the human brain and nervous system. This capability renders it possible to simulate large-scale arbitrarily complex non-linear problems. Its knowledge acquisition process is mainly through a learning process that aims to determine an optimal set of weights for the connections and threshold values for the neurons (Rumelhart *et al.* 1994). Currently, the most widely used ANN is the feed-forward back-propagation network (BPN). Figure 1 shows a typical architecture of an ANN model employed in this study. It comprises basically a forward pass and a reverse pass. The neuron response is computed from the weighed sum of its inputs and bias with a predetermined activation function in the forward pass whilst the weights are adjusted based on the error between the computed and target outputs in the reverse pass. The error is then distributed to neurons in each layer by the derivatives of the objective function with respect to the weights, which can be moved in the direction in which the error declines most quickly by using a gradient descent method. The termination criterion is reached when the error is smaller that a preset value. It should be noted that the initialization of weights and biases may also have some effects on the network performance and hence improper assigned values can result in local convergence. The major drawback of the conventional BPN with gradient descent learning algorithm is the slow convergence rate.

2.2 Genetic algorithm (GA)

A genetic algorithm (GA) applies biological principles into computational algorithm to obtain the optimum solutions and is a robust method for searching the optimum solution to a complex problem. Although it may not necessarily lead to the best possible solution, but usually the requisite precision can be attained (Goldberg and Kuo 1987). In order to compare the performance with the linear model, the linear model with GAs for optimizing parameters is described as follows:

$$x_{t+1} = ax_t + bx_{t-1} + cx_{t-2} - d \tag{1}$$

where, a , b , c ,and d are parameters.

The goal of the problem is to ascertain optimal parameters so that accumulative errors between measured data and simulated data are minimal. Therefore, the fitness function is developed as follows:

$$f(a,b,c,d) = \sum_{i=1}^{p} \sum \left| (X_m)_i - (X_s)_i \right| \tag{2}$$

The ranking selection method (Baker 1985) is adopted in the present study and the probability $prob(rank)$ is defined as follows:

$$prob(rank) = q(1-q)^{rank-1} \tag{3}$$

where q is a parameter defined by a user, $rank$ is the position of an individual ranked in a descending or ascending order. The goal of the ranking selection is to ensure that a good chromosome has a higher chance of being selected for the next generation. After the determination of the $prob(rank)$, roulette wheel selection (Goldberg and Deb 1989), which is based on cumulative $prob(rank)$, is adopted here.

Although a GA holds the ability of searching the global optimum solution to a complex problem, the drawback is that it may not necessarily lead to the best possible solution owing to limitation on local searching capability.

2.3 A hybrid algorithm ANN-GA

A genetic algorithm-based artificial neural network (ANN-GA) model is developed here since it is possible that a hybrid integration of ANN and GA algorithms may have better performance by taking advantages of the characteristics of both of them. It may speed up the convergence of an ANN model and enhance the local searching capability of a GA model. In this algorithm, a GA is employed to optimize initial parameters of ANN as a first step, which is then followed by training with a conventional ANN. The objective of the GA sub-model is to determine optimal parameters in order to attain minimal cumulative errors between the measurement and computation. The following equation represents the fitness function of the

GA sub-model used for initializing weights and biases:

$$\min J(W,\theta) = \sum_{i=1}^{p} \left| Y_i - f(X_i, W, \theta) \right| \tag{4}$$

where W is the weight, θ is the bias or threshold value, i is the data sequence, p is the total number of training data pairs, X_i is the ith input data, Y_i is the ith measured data, and $f(X_i, W, \theta)$ represents computed output. Figure 2 shows the overall flow chart of the ANN-GA model, where p_c is the crossover probability, p_m is the mutation probability, $G_{\max}$ is the maximum number of generation, and $N_{\max}$ is the population size.

3. Application Results

3.1 Yangtze River

This model is applied to a reach in the middle section of the Yangtze River (Figure 3). Being the largest river in China, it passes through Wuhan city, the capital of Hubei province. One of the characteristics of the Yangtze River is its intrinsic unsteady but roughly seasonal flow behavior. In general, the peak flow and the dry weather flow occur during the summer and winter months, respectively. Thus, a hydrological year can be divided into a flooding season and a non-flooding season, basically from June to October and from November to next May, respectively. For instances, the variation of water level at Luo-Shan station can be from 17.3m during the non-flooding season to 31.0m during the flooding season. If the mean water levels are considered, the values are 20.8m and 27.1m during the non-flooding and flooding seasons, respectively.

In this study, the water levels of the downstream station, Han-Kou, is forecasted on the basis of the known water levels of the upstream station, Luo-Shan, at different lead times. Owing to the relatively small value of the lateral inflow in comparison with the discharge of the main stream, it is largely neglected. According to the observation data together with the application of the Muskingum method, the travel time of flood between Luo-Shan and Han-Kou is found to be at the order of 24 hours or so. As such, the phase difference between the flood at Han-Kou and that at Luo-Shan is about one day. Moreover, it is expected that water stages during the previous few days at Lou-Shan might contribute to the water level at Han-Kou. Based on the above, it is possible to determine the correlation function between a time series having D points of spacing Δ apart, $x(t-(D-1)\Delta), \cdots, x(t-\Delta), x(t))$ and a forecasted value $x(t+p)$ at certain time in future. Owing to the data availability and the phase lag between the two locations, the parameters are chosen to have the following values: $p=1$ day and $\Delta = 1$ day. Since it is anticipated that the choice of D will have significant effect on the results, numerical experiments are performed to determine the optimal value of D. An integer value between 1 and 4 is tried in this case. The data used for modeling are daily averages for water levels of Luo-Shan and Han-Kou stations in 1984, 1985, 1986, and 1987.

One of the challenging difficulties in pattern recognition is overfitting problem, which usually occurs when the output fits the training data too well. In such cases, in addition to the underlying mapping, the noise is simulated as well so that the model, on contrary, does not fit well to other new data. Smith (1993) suggested several methods to address this problem: limitation of the number of hidden nodes; avoidance from large weights; and, limitation of the number of training epochs. Shahin et al. (2002) suggested dividing the data into three sets. In this study, their recommendations are strictly adhered to. The data are randomly divided into three independent sets, namely, training, testing, and validation sets, with proportion of 50%, 25% and 25%, respectively. As a result, 1456 input-output data pairs under the following format are extracted from the entire data record:

$$[x(t-(D-1)),\ldots x(t-2), x(t-1), x(t);\ x(t+1)] \quad (5)$$

which represents the correlation among water levels at Luo-Shan during the past few days and the water level at Han-Kou at the following day. Efforts have been used to ensure that data used for training, testing and validation represent the same population and that there is no need to extrapolate beyond the range of their training data. Table 1 shows the statistical parameters, including the mean, standard deviation, minimum, maximum, and range, for the training, testing, and validation sets, respectively. It can be observed that the distribution of different data set basically fulfill the abovementioned criterion.

Two conventional goodness-of-fit measures are employed to gauge the performances of the predictions resulting from training, testing, and validation, namely, the: root mean square error (RMSE) and coefficient of correlation (CC):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}[(X_m)_i - (X_c)_i]^2}{n}} \quad (6)$$

$$CC = \frac{\sum_{i=1}^{n}[(X_m)_i - (\overline{X}_m)_i][(X_c)_i - (\overline{X}_c)_i]}{\sqrt{\sum_{i=1}^{n}[(X_m)_i - (\overline{X}_m)_i]^2[(X_c)_i - (\overline{X}_c)_i]^2}} \quad (7)$$

where n = total number of data pairs considered; subscripts m and c = the measured and computed data set, respectively; $\overline{X}_m$ and $\overline{X}_c$ = mean value of the measured and computed water levels, respectively. The two goodness-of-fit measures have different characteristics so that the use of both of them together will complement one another. It should be noted that RMSE and the coefficient of correlation provide a quantitative indicator of the model error in units of the variable and qualitative indicator between the measured and computed data, respectively. Another feature under the RMSE measure is that larger errors will incur greater effect than smaller errors.

3.2 Application of linear regression (LR) model

There have been no significant changes to the basic characteristics of the Yangtze River such as bathymetric, topographic, and climatic conditions. It is believed that some correlation might be found between the water levels at Luo-Shan in the upstream and Han-Kou in the downstream locations of the river, respectively. Being the simplest and well-developed representation of a causal, time-invariant, relationship between an input function of time and the corresponding output function, the LR model is an ideal benchmark for comparison with all other sophisticated models in flood forecasting. Table 2 shows the results of four regression models initially developed to determine the optimal D value between 1 and 4. It is found that strong correlations, which might easily lead to multi-collinearity problem, exist amongst the input variables. From Table 2, it is observed that, when D = 2 or 1, the input variables are not sufficient. However, with D = 4, the surplus variables might lead to overfitting of the training set whilst the performance of the validation set is lowered. Thus, the linear regression prediction model with D = 3 is adopted as follows:

$$X_{t+1} = 1.441X_t - 0.823X_{t-1} + 0.393X_{t-2} - 5.073 \quad (8)$$

3.3 Application of ANN model

The input and output data are normalized to a range from 0 to 1, corresponding to the minimum and the maximum water levels, respectively. In this study, a three-layer network is used. In order to determine the optimal network geometry that holds for good generalization, ANN models with different number of nodes ranging from 1 to 7 in the hidden layer are trained under a trial and error procedure. The performances for training set and testing set at different training epochs are recorded. Training is stopped when the error learning curve of the testing set starts to increase and that of the training set is still decreasing. The optimal ANN architecture is determined to be 3-3-1. The RMSE for the training set, the RMSE for the validation set, and the optimal training time are 0.268, 0.272 and 4,096 s, respectively.

3.4 Application of GA model

According to floating point coding, every chromosome is composed of a , b , c , and d . Ranges of a , b , c are all from -2.0 to 2.0 and the range of d is from -10.0 to 10.0, which are determined by referring to coefficients of the LR model. The size of populations pop_{size} is set to be 300, which is determined through numerous experiments. The initiation process can then be begun. Prior to the commencement of the genetic operations, several parameters have to be assigned. The crossover probability p_c , mutation probability p_m and q are set to be 0.9, 0.1 and 0.08, respectively Thus, in each generation, we randomly select $p_c \cdot pop_{size}$ chromosomes for crossover operation and $p_m \cdot pop_{size}$ chromosomes for mutation operation and the selection operation is based on the cumulative $prop(rank)$. The smaller is the value of the fitness function, the higher its rank because the goal of the solution is to minimize the cumulative errors. Finally, the optimal parameters for a , b , c , and d are determined to be 1.620, -1.005,0.395, and -5.073, respectively. The performance of the model based on GAs is shown in Table 3.

Similar to the LR model, the GA model attains the best performance when D equals 3. The advantage of GAs over LR might not be conceivable due to high linearization of the studied problem. As a matter of fact, one of the merits of a GA is its robustness in searching the optimum solution for a complex problem. In this case, its characteristic to perform optimization, but not to ensure the accomplishment of the optimal result, is again shown. In essence, this result is quite natural and predictable because many random operations are involved in GAs, such as selection, initialization, and crossover and mutation operations. It is not easy to search the optimal solution in random whilst it is more realistic to obtain a comparatively near-to-global optimal solution.

3.5 Application of ANN-GA model

In order to compare under the same basis, three inputs and one output are applied to ANN-GA model. Similar to the treatment to ANN models, ANN-GA models with different number of nodes ranging from 1 to 7 in the hidden layer are trained under a trial and error procedure. After the same trial and error process as for the ANN model, the optimal ANN-GA architecture is also found to be 3-3-1. The performances for training and testing sets with different numbers of hidden nodes are shown in Table 4. RMSE_tra and RMSE_tst represent performance of training set and testing set, respectively, and the epochs corresponding to values identified by bold & italic are the stopping epochs for ANN-GA with different hidden layers nodes. Figure 4 shows the prediction results and absolute errors for the validation data set with the ANN-GA model. Table 5 shows the comparison of performance between the ANN and ANN-GA models. The results show that the integration of GA can accelerate the convergence of the traditional ANN model. When attaining the same performance of RMSE_vali, the hybrid algorithms only consumes 135s, the ANN model, however, requires 4,096s, which are over 30 times than the former.

3.6 Analysis and performance comparisons

Table 6 represents the comparison of the performance of LR, ANN, GA and ANN-GA models, using indicators including RMSE_tra RMSE_vali, training time, and number of parameters. In terms of RMSE_vali and RMSE_tra, the ANN-GA model is the best in accuracy amongst various algorithms. It should be observed that the ANN-GA model has the advantage to be able to contort itself into a complex form to accommodate the temporal changes of the input-output data pairs whilst the LR model can only fit a linear function to input-output data pairs. It is logical that an ANN-GA model with 16 parameters is more flexible than LR model with 4 parameters, which can be considered an analogy to the comparison of performance between a power or polynomial function and a simple linear function. Regarding the coupling of ANN and GA, it is reasonable to attain better performance by taking advantage of the local optimization of ANN and the global optimization of GA. Results demonstrate that the integration of GA results in much less training time than that of the ANN model whilst the incorporation of ANN enhances the local searching capability and hence the accuracy of a GA model. Hence, it is believed that the ANN-GA algorithm will have strong potential for further developments and applications in hydrological problems.

4. Conclusions

It is shown in this paper that, when cautious treatment is addressed to avoid overfitting problems, the hybrid ANN-GA model produces accurate flood predictions of the channel reach between Luo-Shan and Han-Kou stations in the Yangtze River. It is demonstrated that this model is able to avoid the complication of traditional physical model, in particular the necessity to glean enormous amount of site-specific parameters. It adequately combines the advantage of ANN for fast convergence and local optimization with the advantage of GA for global searching ability. Of course, the better accuracy performance may be in expense of additional modeling parameters and possibly relatively longer computation time when compared with the empirical LR and GA models. Nevertheless, hybrid models, such as ANN-GA model developed here, could be considered as feasible alternatives to conventional models. It is worth exploring into different types of hybrid techniques since new solution approach with better accuracy performance might be found.

Acknowledgement

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

References

Baker, J.E. (1985). "Adaptive selection methods for genetic algorithms." *Proceeding of the 1st International Conference on Genetic Algorithms and Their Applications*, Grefenstette, J.J., eds. Lawrence Erlbaum Associates, Hillsdale, NJ, 101-111.

Chau, K.W. (2002). "Calibration of flow and water quality modeling using genetic algorithm." *Lecture Notes in Artificial Intelligence*, Vol. 2557, 720-720.

Chau, K.W. (2004a). "River stage forecasting with particle swarm optimization." *Lecture Notes in Computer Science*, Vol. 3029, 1166-1173.

Chau, K.W. (2004b). "Rainfall-runoff correlation with particle swarm optimization algorithm." *Lecture Notes in Computer Science*, Vol. 3174, 970-975.

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

Chau, K.W., and Lee, J.H.W. (1991a). "Mathematical modelling of Shing Mun River network." *Advances in Water Resources*, Vol. 14, No. 3, 106-112.

Chau, K.W., and Lee, J.H.W. (1991b) "A microcomputer model for flood prediction with applications." *Microcomputers in Civil Engineering*, Vol. 6, No. 2, 109-121.

Cheng, C.T., Ou, C.P., and Chau, K.W. (2002). "Combining a fuzzy optimal model with a genetic algorithm to solve multi-objective rainfall-runoff model calibration." *Journal of Hydrology*, Vol. 268, No. 3, 72-86.

Goldberg, D.E., and Kuo, C.H. (1987). "Genetic algorithms in pipeline optimization." *Journal of Computing in Civil Engineering, ASCE*, Vol. 1, No. 2, 128-141.

Goldberg, D.E., and Deb, K. (1991). "A comparative analysis of selection schemes used in genetic algorithms." *Foundations of Genetic Algorithms*, Morgan Kaufmann, San Mateo, Calif., 69-93.

Govindaraju, R.S. (2000). "Artificial neural networks in hydrology. II: hydrologic application." By the ASCE Task committee on Application of ANNs in Hydrology. *Journal of Hydrologic Engineering, ASCE*, Vol. 5, No. 2, 124-137.

Liong, S.Y., Lim, W.H., and Paudyal, G.N. (2000). "River stage forecasting in Bangladesh: neural network approach." *Journal of Computing in Civil Engineering, ASC*E, Vol. 14, No. 1, 1-8.

Olivera, R., and Loucks, D.P. (1997). "Operating rules for multireservoir systems." *Water Resources Research*, Vol. 33, No. 4, 839-852.

Rumelhart, D.E., Widrow, B., and Lehr, M.A. (1994). "The basic ideas in neural networks." *Communications of the ACM*, Vol. 37, No. 3, 87-92.

Shahin, M.A., Maier, H.R., and Jaksa, M.B. (2002). "Predicting settlement of shallow foundations using neural networks." *Journal of Geotechnical and Geoenviromental Engineering, ASCE*, Vol. 128, No. 9, 785-793.

Smith, M. (1993). *Neural Networks for Statistical Modeling*, Van Nostrand Reinhold, New York.

Smith, J., and Eli, R.N. (1995). "Neural-network models of rainfall-runoff process." *Journal of Water Resources Planning and Management, ASCE*, Vol. 121, No. 6, 499-508.

Tokar, A.S. and Johnson, P.A. (1999). "Rainfall-runoff modeling using artificial neural networks." *Journal of Hydrologic Engineering, ASCE*, Vol. 4, No. 3, 232-239.

Wardlaw, R., and Sharif, M. (1999). "Evaluation of genetic algorithms for optimal reservoir system operation." *Journal of Water Resources Planning and Management, ASCE*, Vol. 125, No. 1, 25-33.

Table 1. Statistical parameters for training, testing, and validation sets

Data sets	Statistical parameters		
	Mean	Standard Deviation	Range
Training set	23.44	3.71	17.35 - 31.04
Testing set	23.44	3.71	17.39 - 30.96
Validation set	23.44	3.71	17.37 - 30.93

Table 2. Performance comparison for different values of D in LR model

	Training set		Validation set	
D	RMSE	CC	RMSE	CC
4	0.235	0.9880	0.238	0.9960
3	0.238	0.9880	0.237	0.9960
2	0.241	0.9880	0.243	0.9958
1	0.242	0.9880	0.244	0.9958

Table 3. Comparison of different values for D in GA model

	Training set		Validation set	
D	RMSE	CC	RMSE	CC
4	0.235	0.9958	0.245	0.9957
3	0.240	0.9959	0.238	0.9960
2	0.2417	0.9958	0.243	0.9958
1	0.2423	0.9958	0.244	0.9958

Table 4. Performance of training and testing sets with different numbers of hidden nodes in ANN-GA model

Epochs	Nodes	1	2	3	4	5	6	7
1	RMSE_tra	3.3313	3.5354	3.3792	4.0290	3.1916	4.5184	5.1050
	RMSE_tes	3.3212	3.5325	3.3295	4.0870	3.2609	4.4393	4.9882
50	RMSE_tra	0.2272	0.2914	0.2526	0.2244	0.2384	0.2521	0.2557
	RMSE_tes	0.2880	0.2997	0.2912	0.2838	0.2513	0.2632	0.2607
100	RMSE_tra	0.2192	***0.2197***	0.2234	0.2183	***0.2179***	***0.2189***	***0.2322***
	RMSE_tes	0.2473	***0.2440***	0.2555	0.2751	***0.2483***	***0.2575***	***0.2432***
200	RMSE_tra	0.2184	0.2185	***0.2156***	***0.2150***	0.2146	0.2144	0.2234
	RMSE_tes	0.2458	0.2465	***0.2360***	***0.2525***	0.2604	0.2622	0.2471
300	RMSE_tra	0.2183	0.2183	0.2152	0.2129	0.2137	0.2137	0.2202
	RMSE_tes	0.2458	0.2459	0.2491	0.2754	0.2691	0.2846	0.2465
500	RMSE_tra	0.2182	0.2182	0.2131	0.2121	0.2120	0.2132	0.2125
	RMSE_tes	0.2457	0.2457	0.2727	0.2949	0.2960	0.2965	0.2747
750	RMSE_tra	***0.2182***	0.2182	0.2121	0.2113	0.2106	0.2121	0.2094
	RMSE_tes	***0.2456***	0.2456	0.2987	0.3174	0.2974	0.3032	0.2939
1000	RMSE_tra	0.2182	0.2182	0.2118	0.2108	0.2098	0.2116	0.2089
	RMSE_tes	0.2456	0.2456	0.2963	0.3242	0.2923	0.3139	0.3008

Table 5. Comparison of performance between ANN and ANN-GA

Epochs	ANN			ANN-GA		
	RMSE_tra(m)	RMSE_vali(m)	Training time(s)	RMSE_tra(m)	RMSE_vali(m)	Training time(s)
1	19.610	19.732	0.5	3.379	3.367	113
50	6.174	6.286	4	0.258	0.262	125
100	4.026	4.102	8	0.235	0.241	131
200	2.523	2.532	21	0.213	0.226	135
500	2.013	2.042	52			
1,000	1.814	1.833	89			
1,500	1.691	1.705	139			
2,500	1.516	1.524	199			
5,000	1.184	1.185	445			
10,000	0.664	0.673	600			
20,000	0.401	0.402	1044			
50,000	0.309	0.311	1732			
100,000	0.268	0.272	4096			

Table 6. Performance comparison for different models

Model	RMSE_tra(m)	RMSE_vali(m)	Training time(s)	Number of parameters
LR	0.238	0.237		4
ANN	0.268	0.272	4096	16
GA	0.240	0.238	65	4
ANN-GA	0.213	0.226	135	16

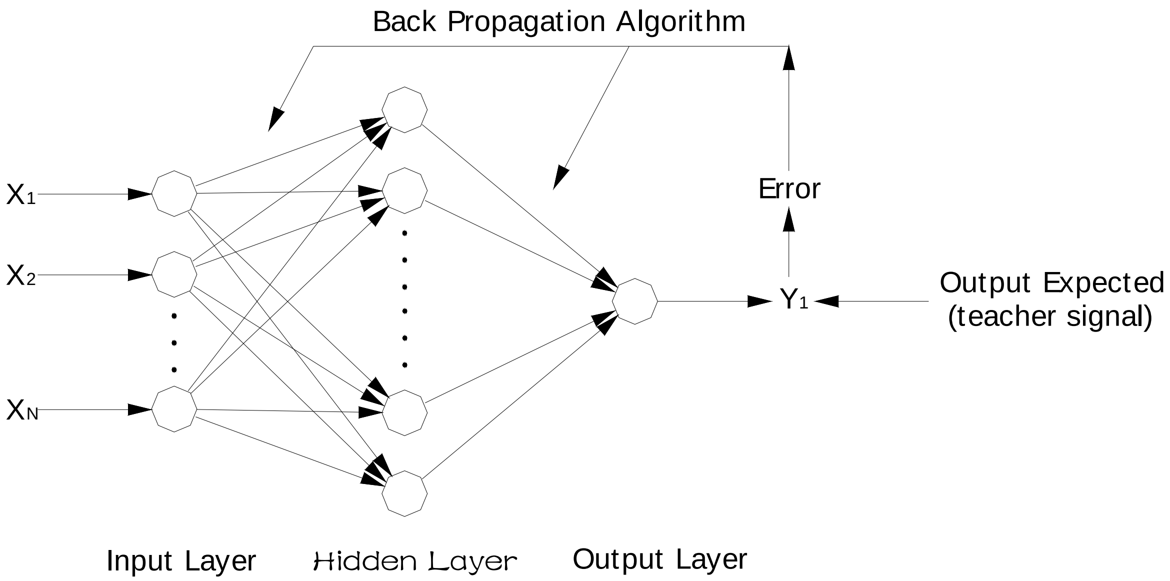

Figure 1. A typical ANN architecture

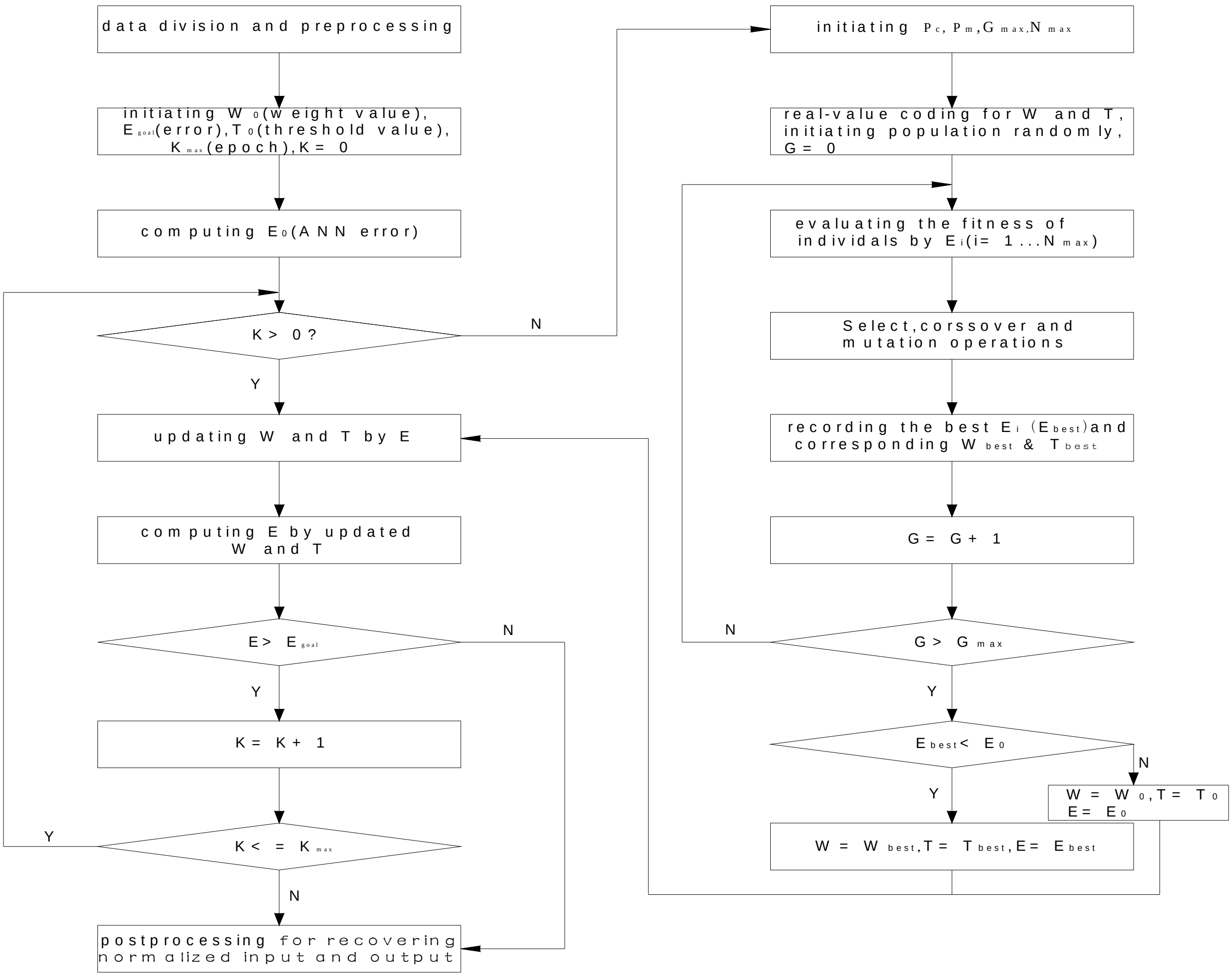

Figure 2. Flow chart for the ANN-GA model

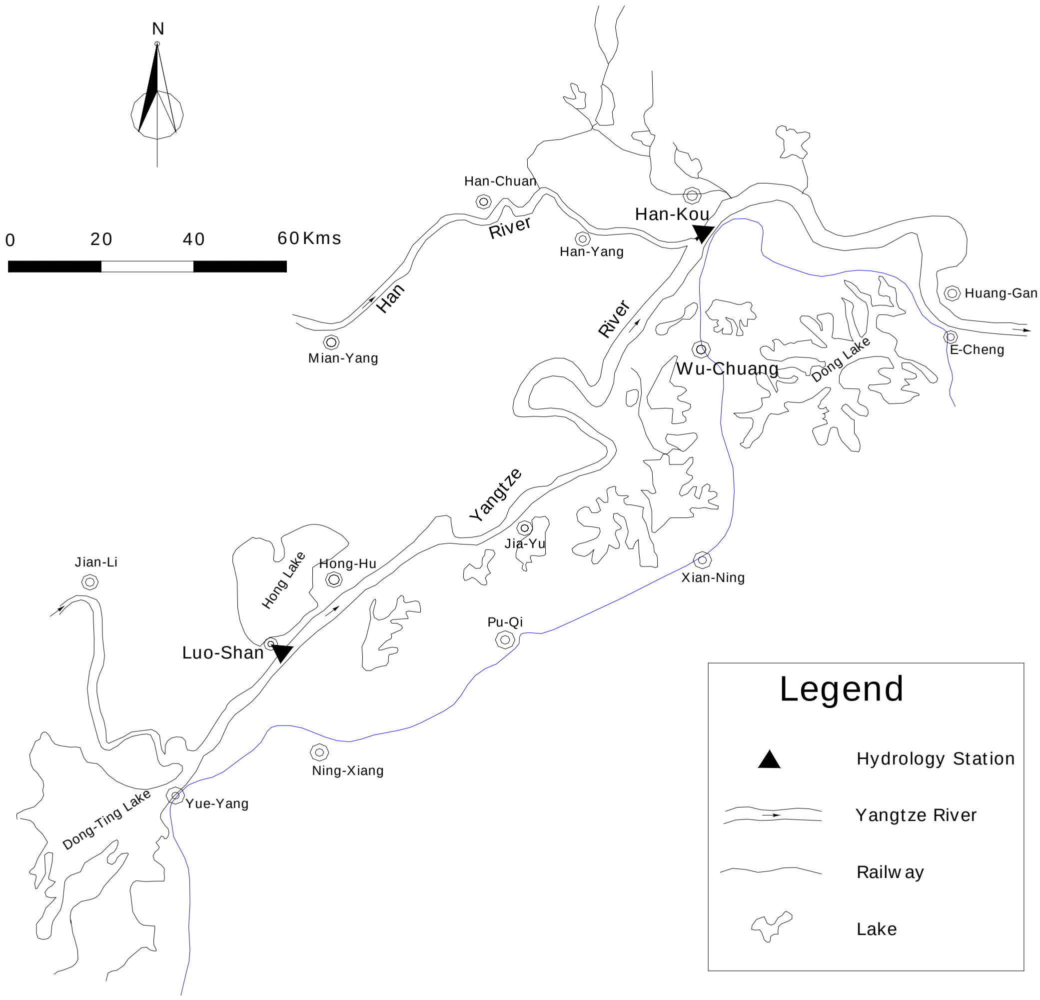

Figure 3. Studied channel section in Yangtze River

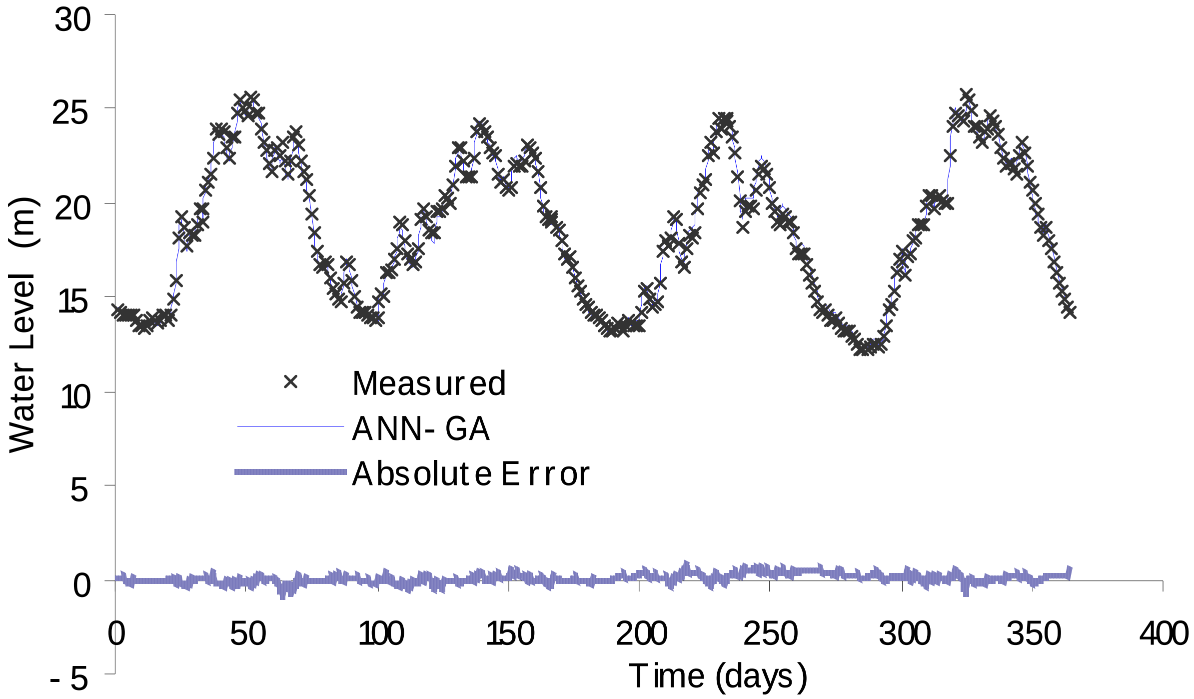

Figure 4. Prediction results and absolute errors for validation data set with the ANN-GA model