Geosynthetics - 7th ICG - Delmas, Gourc & Girard (eds) © 2002 Swets & Zeitlinger, Lisse ISBN 90 5809 523 1

Artificial neural networks-based model for GRW design height forecasting

J. P. ZHOU & S. W.YAN, School of civil engineering, Tianjin University, Tianjin, China

ABSTRACT: Design height of geosynthetics-reinforced retaining wall (GRW) is a key factor for design. This paper presents an artificial neural networks-based approach for predicting the design height of GRW. Seven major affecting factors have been identified from analyzing the general failure cause. A radial basis function neural network (RBFN), as well as a back propagation neural network (BPN) for comparison, is trained and tested using 23 series of centrifuge model test data, 2 full-scale test data, and 1 actual project. The modeling results indicated that the RBFN is much better than the BPN on learning speed, prediction accuracy and generalization ability. The paper provides a novel research area for GRW design.

1 INTRODUCTION

Forecasting the design height plays an important role in GRW designing. It is well known that most walls collapsed for they are higher than their allowable height. Current design methods for GRW are mainly semi-empirical which is based on the so called 'apparent cohesion theory'. Due to the complexity of soil property and reinforcement mechanism, many simplified assumptions have to be made in developing a calculation model.

An artificial neural network (ANN) model is fundamentally different from the conventional calculation model. One of its distinctive features is that it is based on experimental data rather than on assumptions made in developing a mathematic model. These features ascertain the ANN model to be an objective model that can truly represent the natural neural connections among variables, rather than a subjective model which assumes the variables obey a set of predefined relations. The ANN model learns from experimental data and forms neural connection stimuli from the learning process functioning somewhat like a human brain. Because of its unique learning, training, and prediction characteristics, the ANN model has great potential in soil engineering application, particularly for situations where good experimental data are available and where conventional constitutive modeling may be difficult and time consuming.

There are no references with regard to the neural network model of GRW. Xu L.R. et al. (1999) evaluate the performance of the geogrid-reinforced steep slope using fuzzy cluster method. The research work is illuminated by Xu's work.

This paper presents a RBFN model for predicting the design height of GRW. For comparison, a BPN is also built. Using gradient-based learning algorithm, BPN is very popular for its conceptive clarity, simple structure and effective network training. However, they have several important drawbacks, such as:
- They are often slow to reach a satisfactory error level;
- The training performance is sensitive to the choice of the learning rate and initial values of the weight parameters;
- They are prone to getting trapped in local minima.

Making use of both supervised and unsupervised learning algorithms, RBF neural networks have been proven to have the ability to approximate arbitrarily well any multivariate continuous function in a compact domain, provided a sufficient number of RBF units are given. Also, its simple network structure and efficient training procedures are making it increasingly popular. In this paper, to reach the same satisfactory error level 0.004, the RBFN need six hundreds iteration steps while BPN have to iterate 130,000 steps. A computer program has been developed. Excellent agreement between modeling results and centrifuge test data are deserved, which demonstrated high effectiveness and efficiency of the RBFN approach in the design of GRW.

2 RBF NETWORKS VERSUS BP NETWORKS

2.1 *Radial bases function neural networks*

The RBF networks, firstly proposed by Moody J in 1989, is quite different from other feed forward networks in that the hidden layer is restricted to radially symmetric function, which produces a local response to the input data. The typical architecture for a neural network is shown in figure 1. The network consists of three layers, i.e. an input layer, a hidden layer, and an output layer. The input vectors to the network are passed to the neurons in the hidden layer via the connection weights. The hidden layer consists of a set of RBF or pattern units, which are radially symmetric. Functionally, there are three elements in a pattern unit :
- A center, which is an input vector in the training set, and after training which will be stored in the weight vector from the input vector to the hidden layer.
- A distance measure, to determine how far an input vector is to a center. In this study the Euclidean distance measure was used.
- A RBF, which is a function of single variable and used to determine the output of a pattern unit .

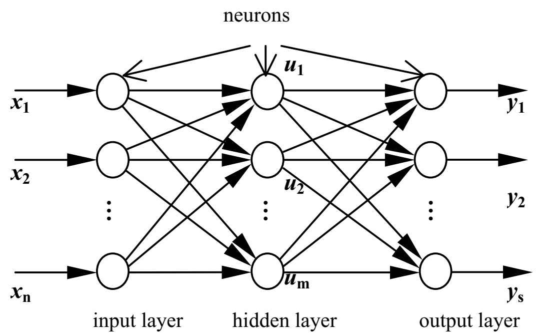

Figure 1. Typical architecture of neural networks

For the network in figure 1, a pattern unit performs a nonlinear functional map, which is based on the current value of the n-dimensioned input vector, $X=(x_1, x_2, ...x_n)$; the center, c_i ; and width parameter, σ_i, to produce the corresponding output:

$$u_i = \phi(L_i) = \exp\left(\frac{-L_i^2}{2\sigma_i^2}\right) \qquad \text{i=1,2,…m} \qquad (1)$$

Where the Euclidean distance is

$$L_i = \left\|X - c_i\right\| = \sqrt{\sum_{j=1}^{n}\left(x_j - c_{ij}\right)^2}$$

j=1,2,…n; and i=1,2,…m (2)

The response of the output layer neuron may be considered as a map y, which is

$$y_k = \sum_{i=1}^{m} w_{ki}\phi(L_i) = \sum_{i=1}^{m} w_{ki}u_i - \theta_k \qquad k\text{=1,2,…s} \qquad (3)$$

where W_k is a weight vector, θ_k is the bias term.

The training process in RBF networks consists of an unsupervised learning and a supervised learning. The learning algorithm is summarized as follows:

- Start training the hidden layer with the unsupervised learning algorithm (*k*-means clustering algorithm) to determine centers, c_i and the radial basis function width parameter, σ_i, which is obtained using a *P* nearest neighbors heuristic. Given a cluster c_i let k_1, k_2, …k_p be indices of the *P* nearest neighboring cluster centers. The width can be calculated as:

$$\sigma_i = \sqrt{\frac{1}{P}\sum_{p=1}^{P}\left\|c_i - c_{ip}\right\|^2} \qquad (4)$$

- Continue training the output layer with the supervised learning algorithm (standard delta rule algorithm) to determine the weights from the hidden layer to the output layer. Interestingly, this is a linear optimization problem that can be solved by ordinary least squares. This avoids the problems of gradient descent methods and local minima characteristics of BPN.
- Simultaneously apply the supervised learning algorithm to the hidden and output layers to fine-tune the network.

2.2 *Back propagation neural networks*

Among various network architectures, back propagation (BP) networks learn from correct patterns (e.g. past experience), and have gained wide application in engineering. A typical BP network has an input layer, an output layer, and a hidden layer, as shown in figure 1.

BP network is feed forward network trained using error back propagation with a gradient descent algorithm with momentum term. The learning algorithm processes the patterns in two stages. In the first stage, the input pattern generates a forward flow of signals from the input layer to the output layer. The error of each output neurode is then computed from the difference between the computed and the desired output. For the hidden layer, the output is:

$$u_i = \sum_{i=1}^{m} w_{ji}x_j - \theta_i \qquad \text{i=1,2,…m} \qquad (5)$$

where w_{ji} is the weight vector associated with x_j and u_i . θ_i is the bias term.

For the output layer:

$$y_k = \sum_{i=1}^{m} w_{ki}u_i - \theta_k \qquad k\text{=1,2,…s} \qquad (6)$$

Where w_{ki} is the weight vector associated with y_k and u_i. θ_k is the bias term.

The second stage involves the readjustment of the weights in the hidden and output layers to reduce the difference between the actual and desired output. The modification of the weighs is carried out using a "generalized delta rule" (Rumelhart et al. 1986) through the gradient descent on the error. The updating of the weights is controlled through a learning rate η and a momentum term α to improve the efficiency of the learning process. Training is carried out iteratively until the average sum squared errors *E* over all training patterns are minimized. *E* is defined as

$$E = \frac{1}{2Q}\left\{\sum_{i=1}^{Q}\ \sum_{k=1}^{s}\left(y_{ik} - y_{ik}'\right)^2\right\} \qquad (7)$$

Where y_k and y'_k is desired and actual output respectively; *Q* is number of patterns; and *s* is number of output neurodes. Further details of the back propagation algorithm can be found in a number of publications including Rumelhart et al. (1986) and Ghaboussi etal. (1991).

3 NEURAL NETWORK MEDELING FOR GRW DESIGN HEIGHT FORCASTING

3.1 *Input data selection and output of neural network model*

For the application of ANN, it is important to properly select input data (i.e. the training data), which is required to represent the characteristics of the studied system. The major factor affecting the design height of GRW can be classified to: backfill soil; reinforcement; foundation; figures of wall and facing rigidity. For lack of data, facing rigidity is not considered here. Backfill soil factor can be represented by three parameters: cohesion *c*, inner friction angle ϕ and unit weight γ. Wall height *H* and slope angle α can mainly describe the wall figure factor. The foundation type can be classified to rigid or firm. While the reinforcement placement and reinforcement strength can be described by its limit strength σ_j, reinforcement space h_j and reinforcement length *L*. The reinforcement place information can be described by a parameter: length ratio *L*/*H*. and the reinforcement strength can be represented by the apparent strength: $\sigma_o = \sigma_j / h_j$.

All of the factor affecting the design height of GRW can be induced to the following 8 parameters: cohesion *c*, unit weight γ, inner friction angle ϕ of backfill soil, foundation type, wall height *H*, wall slope angle α, reinforcement length ratio *L*/*H* and reinforcement apparent strength $\sigma_o = \sigma_j / h_j$.

Those 8 parameters are inter-inhibitive. Once 7 of them are determined, the last one will be determined. So the input data of the RBF neural network is 7 dimensions, and the output data is only 1 dimension. The input-output information of neural networks is shown in figure 2.

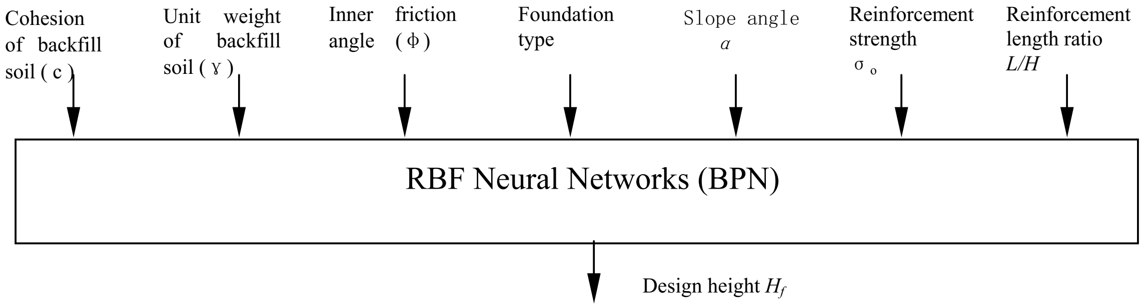

Figure 2. Input-output information of RBF (BP) neural network model

3.2 *Data for network training and testing*

Most of the network training data come from centrifuge model test of reference Porbaha, A. & Goodings, K.J. (1996). Table 1 shows data for network training.

Table 1. Data for network training.

Model	Wall figure		Foundation	Backfill soil			Reinforcement	
number	H_f (m)	α (°)	type	c(kPa)	ϕ(°)	γ	σ_o**(kPa)	L/H
M- 49	7.4	90	Firm	17.8	21.5	17.8	135.8	0.67
M- 45	5.5	90	Firm	18.4	18.2	17.8	101.0	0
M- 56	7.3	90	Firm	22.8	19.4	17.8	133.9	0.75
M- 14	5.3	90	Firm	17.8	20.7	17.8	97.4	0
M- 32	11.4	80.5	Firm	23.8	20.6	17.8	209.2	0.75
M- 35	11.1	80.5	Firm	22.7	21.3	17.8	203.6	0.75
M- 46	11.3	80.5	Firm	24.5	19.3	17.8	207.3	0.75
M- 58	6.95*	80.5	Firm	18.8	21.8	17.8	170.1	0.75
M- 52	7.34*	80.5	Firm	20.7	20.4	17.8	139.5	0.75
M- 53	11.4	80.5	Firm	23.7	18.6	17.8	209.2	0.75
M- 60	11.2	80.5	Firm	23.7	18.6	17.8	205.6	0.75
M- 37	10.6	80.5	Firm	19.3	21.4	17.8	194.5	0.67
M- 33	8.5	80.5	Firm	18.1	20.2	17.8	156.0	0.50
M- 29	7.2	80.5	Firm	17.8	21.7	17.8	132.0	0
M- 31	7.4	80.5	Firm	20.2	18.5	17.8	135.8	0
M- 57	8.8	80.5	Rigid	22.9	18.2	17.8	161.5	0.75
M- 27	6.1	80.5	Rigid	17.3	21.4	17.8	111.6	0
M- 11	9.2*	90	Firm	24.7	19.3	17.8	168.8	0.75
M- 12	6.8*	90	Rigid	16.5	20.4	17.8	124.8	0.75
M- 15	9.2*	80.5	Firm	23.3	21.7	17.8	168.8	0.75
Ref. ***	6.0	85	Firm	42.0	26.8	17.0	120.0	0.86
Ref. ****	6.0	87.2	Firm	0.0	45.0	20.0	800.0	0.33

* Data calculated by method from reference Urszula, B. (1998).
** Apparent strength calculated by N× σ_j/h_j, where N is centrifuge acceleration rate on limit state.
*** Data from reference Helwany, M.B. (1996).
**** Data from reference Lee, K. et al. (1994).

Table 2 shows the network testing data; the preceding three series of which are centrifuge model test results cited from reference Porbaha, A. & Goodings, K.J. (1996).

Table 2. Data for networks testing

Refe-	Wall figure		Foundation	Backfill soil			Reinforcement	
rence	H_f (m)	α (°)	type	c(kPa)	ϕ(°)	γ	σ_o**(kPa)	L/H
M- 28	8.2	90	Firm	20.0	20.8	17.8	150.5	0.75
M- 48	6.1	90	Firm	18.6	20.1	17.8	111.8	0.50
M- 34	5.3	90	Firm	16.3	21.3	17.8	97.4	0
Ref. *	6.1	90	Firm	42.0	26.8	19.3	222.2	0.88

* Data from reference Helwany, M.B. (1996).
** Apparent strength calculated by N× σ_j/h_j, where N is centrifuge acceleration rate on limit state.

3.3 *Results and Discussion*

The network training was based on data shown in table 1. It is necessary, from our experience, to normalize every set of data with respect to its maximum and minimum values before the training process starts. After normalization, each set of data value is presented within a range of (0,1), with their maximum and minimum values represented by 1 and 0, respectively. This preprocessing of data guarantees that the network operates in a more efficient and more reliable manner.

Figure 3 shows a typical variation of the magnitude of the errors at different stages of the BP neural network-training phase. While figure 4 shows typical convergence characteristics of RBF neural networks. Training was carried out until the average sum squared errors over all the training patterns were minimized.

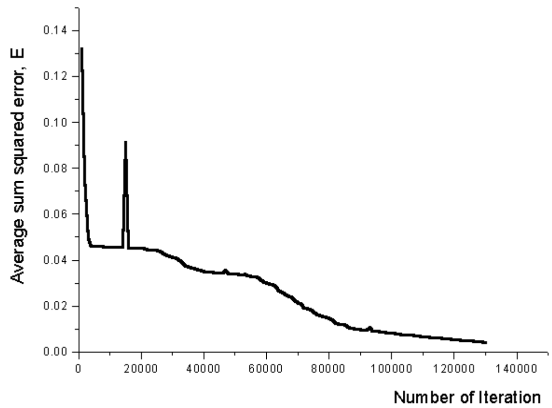

Figure 3. Typical convergence characteristics of BP neural network

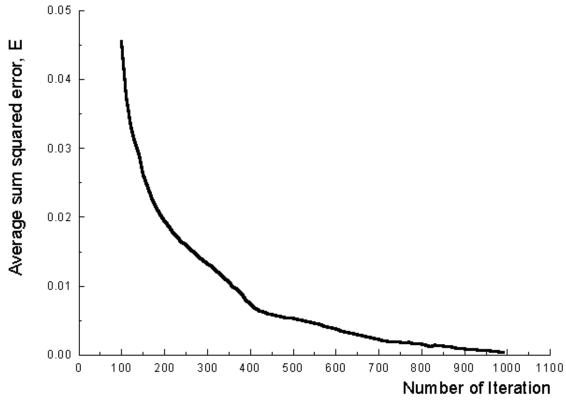

Figure 4. Typical convergence characteristics of RBF neural network

On the satisfactory completion of the training phase, verification of the performance of the neural network is then carried out using patterns that were not included in the training set. This determines the quality of the predictions in comparison to the desired outputs. This is often called the testing phase. No additional learning occurs during this phase.

From figure 3 and figure 4, the differences in the average sum squared errors of BPN were minimal after about 80,000 iterations when the differences of RBFN were minimal after about 400 iterations. To reach the same satisfactory error level 0.004, the RBFN need six hundreds iteration steps while BPN have to iterate about 130,000 steps, which indicates that RBFN takes much shorter training time and performs better in classified ability and learning speed.

Table 2 shows the data for testing networks, and table 3 shows the modeling results. The forecasting results of RBFN are excellent agreement with the test data, even for which they have not been trained. The maximum prediction error ratio of RBFN for the same data is reduced to 5.64%, which shows a significant improvement over the performance of BPN. The successes of artificial neural networks-based models indicate that the neural networks are capable of generalization.

Table 3. Modeling results of RBFN & BPN (Unit: m).

Reference	Test results (m)	Forecasting results			
		BPN	error ratio	RBFN	error ratio
Model no. 28*	8.2	8.339	1.70%	8.289	1.09%
Model no. 48*	6.1	6.521	6.90%	6.402	4.95 %
Model no. 34*	5.3	5.451	2.85%	5.184	2.19%
Reference **	6.1	6.757	10.8%	6.444	5.64%

* Data from reference Porbaha, A. & Goodings, K.J. (1996).
** Data from reference Helwany, M.B.(1996).

The artificial neural network is an information processing system that has certain performance characteristics in common with biological neural networks. In situations in which input is noisy or the input-output relationship is illogical, the ANN can still produce reasonable results. As compared with the traditional design methods, the ANN-base methods appears to have several important advantages:

- The model is essentially based on experimental data only. No assumptions are made, which allows the model to become more objective than subjective.
- The ANN model is set up without any calculation of parameters required by a mathematical constitutive model. The implementation of the ANN models only involves a series of iteration calculations.
- The prediction accuracy of the neural networks will increase with the collection of training patterns.

4 CONCLUSION

Application of the neural network to forecasting the design height of GRW is a novel research area. Based on the modeling results obtained in this study, the following conclusions can be drawn:

1. This paper demonstrates the feasibility of using ANN to predict the design height of geosynthetis-reinforced retaining wall (GRW). A RBFN and a BPN were developed for forecasting the design height of GRW. Data from centrifuge test were utilized to construct the networks. Predictions made using RBFN were compared with that of BPN, and the RBFN models proved to provide the better predictions.
2. Many factors affecting the design height prediction are ignored, oversimplified or improperly introduced by the existing design methods. For ANN-based methods, however, the performance of the neural network models improved as more input variables are provided. In this paper, models consisting of seven variables: cohesion c, unit weight γ, inner friction angle ϕ of backfill soil, foundation type, wall slope angle α, reinforcement length ratio L/H and reinforcement apparent strength $\sigma_o = \sigma_j / h_j$.
3. The ANN simulated the training data well and effectively predicted the testing data. The efforts encourage further investigations of this paradigm for geomaterial–based modeling. Overall, the investigation revealed the potential to develop a comprehensive ANN-based design height forecasting method of GRW.
4. To improve the proposed prediction method, it is recommended that additional training data on failure height of GRW shall be collected which will improve the performance of the neural network models greatly. The main criticism of the neural network methodology is its inability at present to trace and explain the step-by-step logic it uses to arrive at the outputs from the inputs provided. This is expected to be a temporary drawback that will be overcame with further research.

ACKNOWLEDGMENTS

The authors would like to express their thanks to International Review Panel for reviewing the manuscript and providing valuable comments.

REFERENCES

Broomhead, D.S. & Lowe, K. 1988. Multivariable functional interpolation and adaptive networks. *Complex System* 2: 321-355.

Ghaboussi, J., Garrentt, J.H.Jr., & Wu, X. 1991. Knowledge-based modeling of material behavior with neural networks. *Journal of Engineering Mechanics,* ASCE. 117(1): 132-153.

Hartman, E.J et al. 1990. Layered neural networks with Gaussian hidden units as universal approximations. *Neural Computation* 2: 210-215.

Helwany, M.B. 1996. Effects of facing rigidity on the performance of geosynthetic reinforced soil retaining wall. *Soil and Foundations* 36(1): 27-38.

Lee, K. et al. 1994. Failure and deformation of four reinforced soil walls in eastern Tennessee. *Geotechnique* 44(3): 397-426.

Lee, S. & Kil, R.M. 1991. A Gaussian potential function network with hierarchically self-organizing learning. *Neural Networks* 4: 207-224.

Moody J, & Darken C.J. 1989. Fast learning in networks of locally tuned processing units. *Neural Computation* 1: 213-225

Rumelhart, K.E., Hinton, G.E., & Williams, R.J. 1986. Learning internal representation by error propagation. D.E. Rumelhart & McClelland, (eds), *Parallel distributed processing, foundations Vol. 1*, MIT Press, Cambridge, Mass.

Porbaha, A & Goodings, K.J. 1996. Centrifuge modeling of geotexile reinforced cohesive soil retaining walls. *Journal of Geotechnical Engineering,* ASCE. 122(10): 840-848.

Powell, M.J.D. 1985. Radial basis functions for multivariate interpolation: a review technical report. *Department of Application mathematics and Theory Physics.* Cambridge University. Cambridge. England.

Urszula, B. 1998. Discussion of 'centrifuge modeling of geotexile reinforced cohesive soil retaining walls'. *Journal of Geotechnical Engineering,* ASCE.124(8): 551.

Xu, L.R. et al. 1999. Investigation and analysis of performance of reinforced steep slope using fuzzy cluster. *Chinese Journal of Geotechnical Engineering* 21(4): 475-480.