Indian Geotechnical Journal, 38(4), 2008, 513-526

TECHNICAL NOTE

An Approach to Predict Ultimate Bearing Capacity of Surface Footings using Artificial Neural Network

J. Noorzaei[*] , S. J. S. Hakim and M. S. Jaafar

Introduction

The ultimate bearing capacity of soil is the maximum load per unit area, which the soil can sustain before failure and is determined through analytical approaches that can incorporate appropriate soil parameters and details about the size, shape and the depth of the footing (Das, 1987 and Cernica,1995).

There are methods, such as Terzaghi, Meyrhoff, Hason, Vesic, etc. to calculate the ultimate bearing capacity of footings. These methods have some limitations. Terzaghi's method considers general failure of a strip footing having rough foundation base while Meyerhof's bearing-capacity equation similar to that of Terzaghi but includes a shape factor and factors for depth and inclination. Hansen suggests factors to be used when the base is tilted or when the footing is on a slope while the basic approach is same as that suggested by Meyerhof (Bowles, 1988, Bolton, 1979 & Cernica, 1995). All these procedures assume a uniform soil below the foundation, while there can be several thin layers of different properties influencing the bearing capacity.

Artificial Neural Networks, ANNs, are very strong and accurate tools expected to overcome such limitations. However, the development of this tool or system require tremendous amount of data to be collected. Also, calculation of UBC in layered soils with different physical and geotechnical specification by empirical methods is very difficult and results by experimental are not exact. ANNs usually employed when the relationship between the input and output is complicated or application of another available method takes a large computational time and effort is very expensive. It requires suitable input parameters, good data selection for training and suitable computational algorithm, so that it is able to learn complicated relationship between inputs and outputs with high precision. (Noorzaei, et.al. 2005 and Hakim, 2006).

Many investigators in different field of the civil engineering have recently used ANNs. For example, Lee et.al (1996) used ANN for predicting the pile bearing capacity. Ghaboussi et.al (1994) showed that ANNs were powerful tools for the mathematical constitutive modeling of geomechanics. Backpropagation Multi-Layer Perceptrons (MLPs) have been applied successfully by Shahin et al. (2002) to settlement prediction of shallow foundations on granular soils. In this

---

* Civil Engineering Department, Faculty of Engineering, University Putra Malaysia, 43400 UPM, Serdang, Selangore, Malaysia E-mail: jamal@eng.upm.edu.my

paper, ANN is used to obtain more accurate settlement prediction. A large database of actual measured settlements is used to develop and verify the ANN model. The predicted settlements found by utilizing ANNs were compared with the values predicted by three of the most commonly used traditional methods. The results indicate that ANNs are a useful technique for predicting the settlement of shallow foundations on cohesionless soils, as they outperform the traditional methods.

Goh (1994) demonstrated that ANNs could model the complex relationship between seismic soil parameters and liquefaction potential using actual field records. Turk et.al (2001) applied the ANNs to predict soil behaviour in uniaxial strain condition. Kerh (2003) presented an ANN model to estimate consolidation settlement caused by groundwater drawdown. Baziar (2005) applied the ANN to estimate the displacement caused by the liquefactions during the earthquake. Employing the ANN method in settlement of the ground due to tunnelling was presented by Kim et.al (2001). Based on the literature review, it is seen that there is no enough work reported on development and application of ANN on bearing capacity in the continuous footing on multilayer soils. The present study deals with development of a neural network model for prediction of the ultimate bearing capacity of shallow foundation on the layered soils.

Artificial Neural Networks

Neural networks are data processing systems consisting of a large number of simple, highly interconnected processing elements (artificial neurons) in an architecture inspired by the structure of the central cortex of the brain. They operate as black box and powerful tools to capture and learn significant structures in data. Neural networks can provide meaningful answer even when the data to be processed include errors or are incomplete and can process information extremely rapidly when applied to solve real world problems.

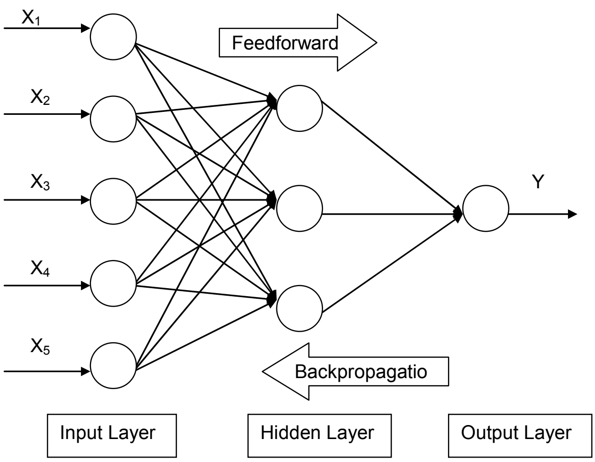

Fig. 1 Architecture of a Typical Multilayer Feed forward Neural Network

As shown in Figure 1 a typical neural network has three layers: the input layer, the hidden layer and the output layer. Each neuron in the input layer represents the value of one independent variable. The neurons in the hidden layer are only for computation purposes. Each of the output neurons computes one dependent variable. Signals are received at the input layer, pass through the hidden layer, and reach the output layer.

An error function in the form of the sum of the squares of the errors between the actual outputs from the training set and the computed outputs is minimized iteratively. In this study, the error incurred during the learning can be expressed as least Mean Squared Error (MSE) and is calculated Eqn (1). (MATLAB 6.5., 2003).

$$\text{MSE}=\frac{1}{Q}\sum_{K=1}^{Q}e(k)^{2}=\frac{1}{Q}\sum_{K=1}^{Q}(t(k)-a(k))^{2} \quad (1)$$

In this equation, e(k) is calculated error in k_{th} neuron, t_k is exact output in k_{th} neuron, a_k is network output in k_{th} neuron and Q is number of training patterns . The least mean square error algorithm adjusts the weights and biases of the network so as to minimize this mean square error.

Model Inputs and Output

The selection of the property-related parameters, or input parameters, is based on the physical background of how the target property is determined. In this present work, the inputs to the network include the width of foundation (B), friction angle in each layer (ϕ_1, ϕ_2, ϕ_3), cohesion of the layers (C_1, C_2, C_3) and depth of first and second layers (H_1, H_2). Depth in third layer is considered infinite and the output of the network is the ultimate bearing capacity of soil. Figure 2 illustrates the geometrical and geotechnical specifications of the soil and foundation, which are selected as input parameters. The input variations are limited as indicated in Table 1.

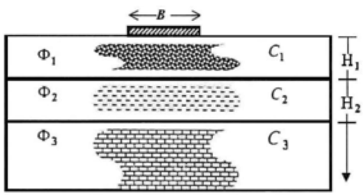

Fig. 2 Geometrical and Geotechnical Specifications of Soil and Foundation

Table 1 Variation of Inputs Parameters

No.	*Input parameters*	*Input variations*
1	B	2 - 10 (m)
2	H1 , H2	0 - 10 (m)
3	Φ1 , Φ2 , Φ3	0 – 40 (deg)
4	C1 , C2 , C3	0 - 0.1 (KN/m^2) *10^3

Data Selection

One of the major issues in development of any ANN model is the collection of the data set related to the problem under investigation. The data are separated into three sets, namely:

> Training sets: knowledge about the learning task is given in the form of examples called training sets. The training set is used to gradually reduce the ANN error.

> Testing sets: The testing set is used to visually inspect performance after training

> Validation sets: The validation set is used to as a further check for the generalization of the neural network, but do not have any effect on the training.

A total of 1660 data were collected from different geotechnical reports (Technical Reports on civil engineering projects, ICTM, Malaysia, 1991, 2001, 2003 & 2004) and were divided for training, testing and validation sets.

Training data sets comprises 1180 data entries, and the remaining data (480) are divided between the testing and validation sets. In these technical reports, for calculation of UBC, in general Hansen's method was used. Also a typical dataset used for training of ANN, with actual numbers is tabulated in Table 2.

Table 2 The List of Typical Data Sets for Training of ANN

No	B (m)	C_1 $(KN/m^2)*10^3$	C_2 $(KN/m^2)*10^3$	C_3 $(KN/m^2)*10^3$	Φ_1 (deg)	Φ_2 (deg)	Φ_3 (deg)	H_1 (m)	H_2 (m)	U.B.C $(KN/m^2)*10^2$
1	10	40	40	60	30	8	25	6	8	684
2	8	35	40	60	24	5	28	8	8	409
3	9	30	40	50	20	5	30	7	7	231
4	7	10	30	10	10	24	38	4	1	47
5	10	15	30	10	20	22	40	4	2	677
6	8	90	100	100	30	30	5	4	7	1121
7	5	10	20	20	10	14	22	1	5	31
8	4	60	70	65	20	11	36	8	5	102
9	9	10	20	10	16	28	40	5	1	207
10	10	15	30	10	18	26	34	4	2	482
11	10	15	30	10	20	28	40	5	1	762
12	9	30	50	60	26	8	25	7	8	613
13	7	10	20	10	10	20	36	5	3	23
14	2	65	80	55	15	11	37	9	4	20
15	9	40	40	50	22	5	30	7	9	327
16	4	65	80	55	15	10	39	9	4	40

Table 2 Contd. The List of Typical Data Sets for Training of ANN

No	*B* *(m)*	C_1 $(KN/m^2)*10^3$	C_2 $(KN/m^2)*10^3$	C_3 $(KN/m^2)*10^3$	Φ_1 (deg)	Φ_2 (deg)	Φ_3 (deg)	H_1 (m)	H_2 (m)	U.B.C $(KN/m^2)*10^2$
17	3	0	10	20	0	10	24	1	3	0
18	6	60	70	65	20	11	36	8	5	154
19	9	15	20	10	10	20	30	3	1	196
20	5	10	20	30	10	10	30	3	4	16
21	7	15	20	10	10	20	40	3	1	87
22	8	15	20	10	10	20	40	3	1	258
23	5	15	20	10	20	28	30	3	2	167
24	6	60	70	45	15	14	38	7	4	60
25	8	90	100	95	35	40	10	4	7	4264
26	5	15	20	10	20	28	32	3	2	167
27	5	0	10	20	0	10	24	1	3	5
28	6	10	30	10	14	20	36	3	3	66
29	6	85	90	100	30	35	10	5	5	872
30	8	35	40	60	24	5	30	8	8	409
31	9	35	40	60	24	5	30	8	8	460
32	10	60	70	45	15	14	38	7	4	101
33	7	15	30	10	18	26	30	4	2	167
34	6	15	30	10	18	26	32	4	2	113
35	8	65	70	55	30	12	40	6	5	868

Normalization of Inputs and Output

For a better network performance, the input–output data pairs are subjected to a scaling process before being used in the network operation. This is because the compiled raw training data for different parameters can vary significantly in their actual values [Demuth, et.al.1996]. Also, the use of the higher number is not desirable as the networks are generally simulated on the computer and this can create floating-point problems.

In this study the most common expression suggested for normalization purpose is expressed as Eqn (2) [Demuth, et.al.2005 and Ince, 2004]. In this study work, all inputs and output has been normalized between “0” and “1”.

$$X_i^n = \frac{x_i^a - x_i^{\min}}{x_i^{\max} - x_i^{\min}} \quad (2)$$

Where x_i^a and x_i^n are the i_{th} components of the input vector before and after normalization, respectively, and $x_i^{\min}$ and $x_i^{\max}$ are the minimum and maximum values of all the components of the input vector before the normalization [Ince, 2004]. Any new proposed ANN should go through the training, testing and validation phase. These three phases are explained in the following sections.

Neural Network Training

A neural computing system can modify its behaviour in response to its environment. When sets of inputs are shown to the network, it will self-adjust to produce consistent responses through a process called training. Training is the process of changing the weights systematically in order to achieve some desired results for a given set of inputs. The aim of training is to find a set of connection weights that will minimize the MSE forecasting error in the shortest possible training time [Demuth, et.al. 1996 and 2005].

In this study, the effects of the neuron number in a hidden layer, numbers of hidden layers, activation function, learning parameters, including momentum coefficient and learning rates on the convergence of the learning algorithm are investigated. These parameters are changed in certain ranges step by step to try to find a better combination of these parameters to attain convergence faster. The above missions of training neural network are presented in the subsequent paragraphs.

Numbers of Hidden Layers

The selection of the number of hidden layer(s) is the most challenging part in the total network development process. Unfortunately, there are no fixed guidelines available for this purpose and hence this has to be done by the trial-and-error method [Kartalopoulos, 2002]. To determine the optimum number of hidden layer, four networks with one, two, three and four hidden layers are trained. It is seen from Figure 3(a) that network with one hidden layer, lead to minimum MSE value in comparison with two, three and four hidden layers.

In the present case, it is seen that after a few trials with the network with two, three and four hidden layers, good convergence could not be achieved. Based on this observation, it was decided to select the number of hidden layer equal to one. However, in a network with one hidden layer, good convergence has been achieved.

Numbers of Hidden Neurons in Hidden Layer

The decision on how many hidden neurons should be used in a layer is rather arbitrary, and has been usually decided by trial and error [Yeh, et.al. 1992]. Figure 3(b) shows the effect of different number of hidden neurons on MSE is investigated. Based on the evaluations summarized by Figure 3(b), it can be seen, that with increasing of hidden neurons, training error is reduced, but there is a number of hidden units existing for minimizing error rate.

The reason is that, with too many hidden units, a network can simply memorize the correct response to each pattern in its training set instead of learning a general solution. It can be seen that with increasing number of hidden neurons, MSE is decreased, but variations in MSE values for more than 15 neurons are insignificant. On the other hand, utilizing more than 15 neurons in network makes the computation process complicated and expensive in terms of time. In summary, in order to have minimum compatibility cost and high accuracy, the number of hidden neurons is fixed to 15.

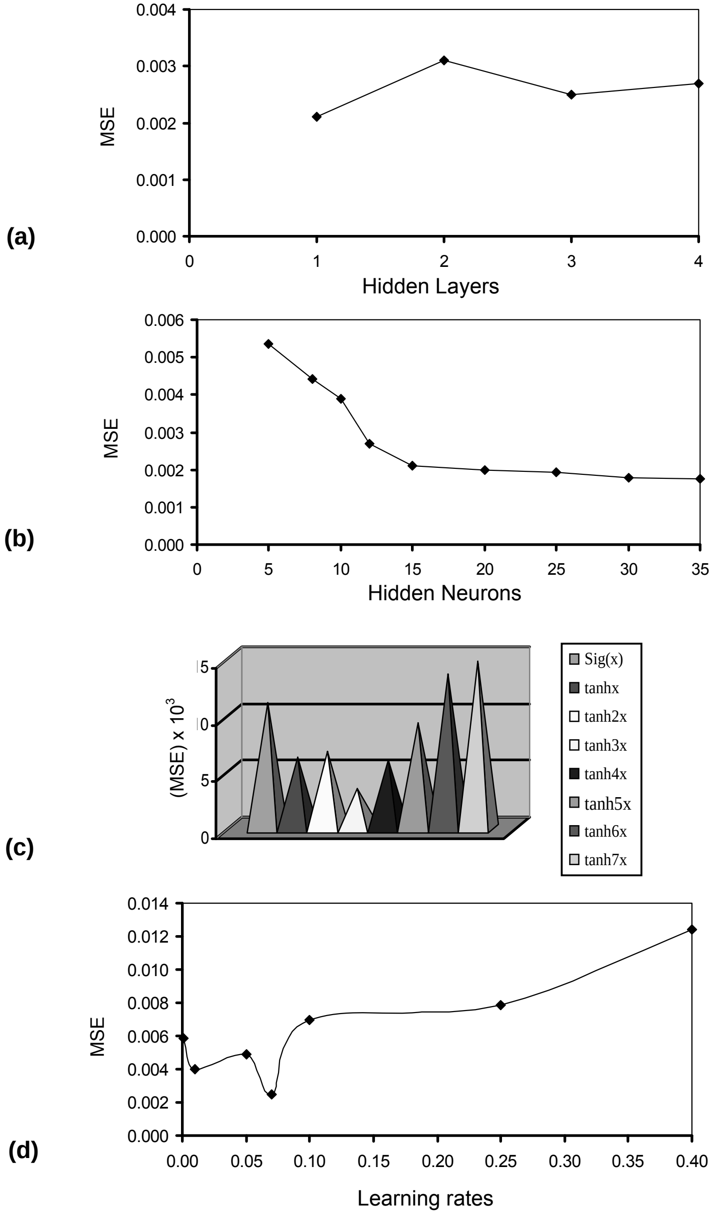

Fig. 3 Determine the Optimum Value of Each Parameter of Neural Network

(a). Comparisons of the Learning Results with Various Numbers of Hidden Layers

(b). Comparison of the learning results with various numbers of hidden neurons

(c). Influence type of activation function on value of network error

(d). Comparison of the learning rate values against MSE

Selection of Activation Function in the Hidden Layer and Output Layer

An activation function is used for producing the neuron output and limiting the amplitude of the output of a neuron. It determines the relationship between inputs and outputs of a neuron and a network. [Yeh, et.al.1992 and Can, 2002]. Decision on activation function for layers is another important parameter.

In this study, to determine the type of activation function by keeping the sigmoid function in output layer and assume different activation functions for hidden layer and in each case the MSE is calculated, eventually the activation function with minimum error will be the best possibly activation function for hidden layer. However, in this investigation, for training of the network, the sigmoid function has been used in the output layer and by trial and error in the hidden layer a suitable activation function, has been found.

In this study, the following activation functions were tried: sig(x), tanh(x), tanh (2x), tanh (3x), tanh (4x), tanh (5x), tanh (6x), tanh (7x), respectively. Each of these activation functions has been given to the network and MSE value was determined. The activation functions and their behaviour against MSE are illustrated in Figure 3(c). It is clear in this figure that for tanh (3x) activation function, network is trained slowly and do not have any oscillation and conversion is very good. However, it can be concluded that, for tanh (3x) the training error is minimum when compared to the other functions. Therefore, tanh (3x) is selected as an activation function in the hidden layer.

Determination of Learning Rate

The learning rate is a parameter that determines the size of the weights adjustment each time the weights are changed during training process. The value of learning rate ranges is between 0 and 1. To obtain the learning rate, trial and error method is to be used [Kim, et.al.2001 and Okine, 1999].

In this study present, it is obvious from Figure 3(d) that there is a critical range, (about 0.05-0.1) for the learning rate to minimize the error. The reason is that the change in weights is proportional to the learning rate, a lower learning rate results in less weights change during each learning cycle, while a higher learning rate may result in weights over-change and oscillation. In summary, after several trials and errors, the learning rate of 0.07 was found to yield minimum error as shown in Figure 3(d). A summary of the final achieved parameters is shown in Figure 3.

Determination of Momentum

To prevent unstable and oscillation network in backpropagation algorithm, there is a value that is called momentum. Momentum term adds a proportion of the previous weights changes to the current weight changes. It provides a momentum in weight space to prevent the movement of weight from oscillations (Okine, 1999 and Yeh, et.al.1992). Based on the experience of the earlier investigations, the value of momentum is suggested to be between “0” and “1” (Can, 2002). To obtain the value of momentum, trial and error method is used. The results show that there is a critical range (about 0.4-0.7) for the momentum factor to minimize the error.

In this investigation after several trials and errors, and comparison of the results, the best value for momentum is selected to be 0.6. It is considered that, the contribution of the learning rate and the momentum in backpropagation neural network algorithm is very significant. In this investigation it appeared that a very small learning rate (roughly 0.001) and relatively high momentum term (between 0.7–0.9) does not provide an appropriate combination for a three-layered network for predicting the ultimate bearing capacity of soil. It appears that a learning rate of around 0.05 to 0.2 and momentum term of around 0.5-0.7 provide the appropriate combination for the ultimate bearing capacity prediction. Table 3 shows the various learning rates and momentum terms, and the general learning behaviour for this study.

Table 3 General Learning Behaviours

Learning Rate	*Momentum Rate*	*Remarks*
0.001	0.9	Gross over learning from the beginning
0.01	0.8	Relatively few iterations need for learning
0.05	0.5	Relatively few iterations need for learning
0.07	0.6	Good learning form the beginning with about 20000 iterations
0.1	0.6	Relatively good learning with about 10000 iterations
0.2	0.7	Relatively good learning with about 7000 iterations
0.3	0.6	Learning is not good after 1000 iterations

Training Results

Based on the major aspects that are discussed in previous sections, an ANN for prediction of UBC of soil has been developed, as illustrated in Figure 4.

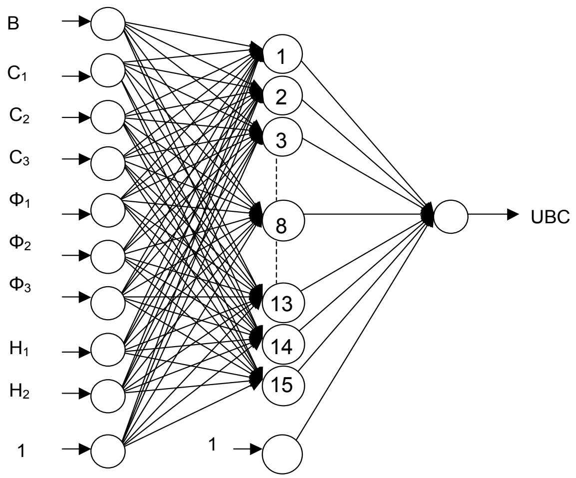

Fig. 4 The Architecture of the Developed Artificial Neural Network for Prediction of Ultimate Bearing Capacity

Comparison of UBC predicted by ANN and experiments for 1180 training data set is shown in Figure 5. Calculation of mean percentage relative error for training set data show that ANN can predict the UBC with an error of 14.83%. In the training process, weights and biases are constantly adjusted to minimize the error between the actual and the desired outputs of the units in the output layer. It should be reported that at the beginning of the training, a set of inappropriate random initial values of weights and biases always resulted in the training divergence. This shows that the combination of weights and biases strongly influences the training process. When the training was over, the weights and biases were fixed.

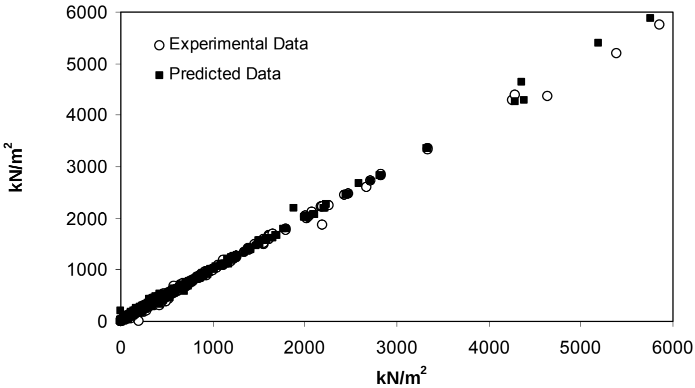

Fig. 5 Comparison of Predicted UBC and Experiments for Training Sets

After successful training has been reached, the network weights and biases are fixed and used for predicting the output corresponding to any set of values of the input parameters. In this network, with 9 input neurons, 15 hidden neurons and one output neuron, it obtained 166 connectivity weights of neural network neurons and biases.

Neural Network Testing

After the network was trained, to highlight the capability of the developed ANN model for UBC in soil, an attempt has been made to predict the UBC of soil for new sets of data. As already mentioned, for the testing purpose, 240 sets of data have been selected. A comparison between the UBC of soil predicted through the ANN and the experimental evidence for testing set is shown in Figure 6. An important observation in this figure is that the results of ANN were very close to the Experimental data. This figure illustrates the compression of the results predicted through ANN and that of actual values for the testing purpose. This indicates that ANN model is capable of predicting UBC of soil with acceptable accuracy. Results show that the ANN was successful in training the relationship between the input and output data with the MSE of 15.73%.

Neural Network Validation

Once the training phase of the model has been successfully accomplished, the performance of the trained model is validated using the validation data, which have not been used as part of the model building process.

The purpose of the model validation phase is to ensure that the model has the ability to generalize within the limits set by the training data, rather than simply having memorized the input–output relationships that are contained in the training data. Hence , in this study , validation set is used as the stopping criterion, as it is considered to be the most valuable tool to ensure that overfitting does not occur and as sufficient data are available to create training, testing, and validation sets.

This is worth mentioning that, stopping criteria are those used to decide when to stop the training process. They determine whether the model has been optimally or suboptimally trained. (Shahin et.al, 2002).

Comparison of value of UBC in validation sets (240 data sets) between experimental value (target value) and predicted value (network value) is shown in Figure 7. The results obtained from validation set showed that the ANN was successful in training with the MSE of 15.91 percent (MSE for validation set).

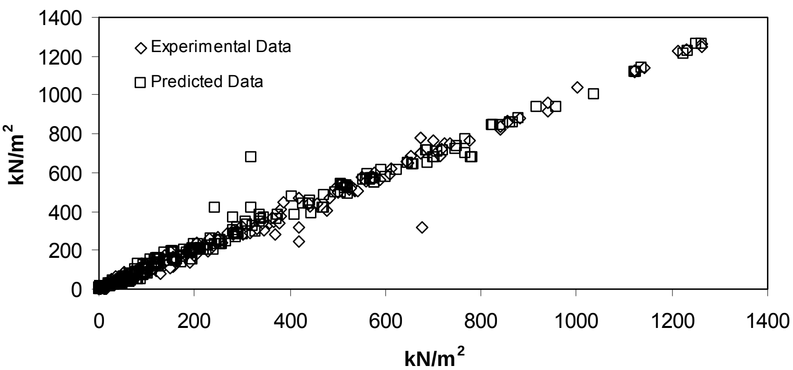

Fig. 6 Comparison of UBC (kN/m^2) Predicted by ANN and Experiments for Testing Sets

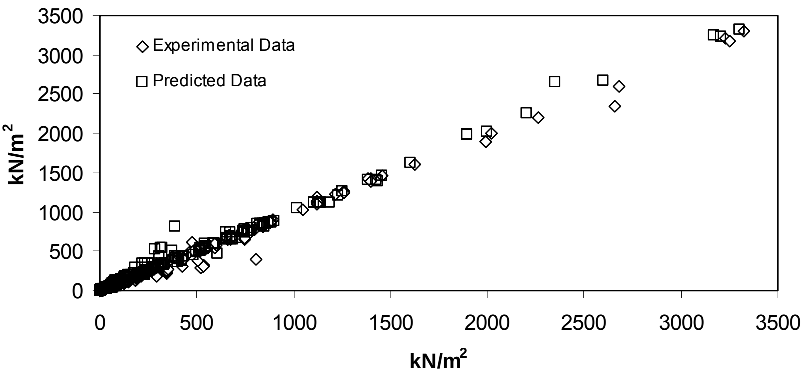

Fig. 7 Comparison of UBC (kN/m^2) Predicted by ANN and Experiments for Validation Sets

Comparison of Training, Testing and Validation sets

The progress of the training was examined by plotting the training, validation and testing Mean Square Error (MSE), versus the performed number of iterations as presented in Figure 8.

The test set error and the validation set error has very similar characteristics and there is no over fitting. The results show that the neural network model has a percentage relative error of about 14.83% for the training data and a percentage relative error of about 15.73% for the test data. Percentage relative error for the validation data was about 15.91%.

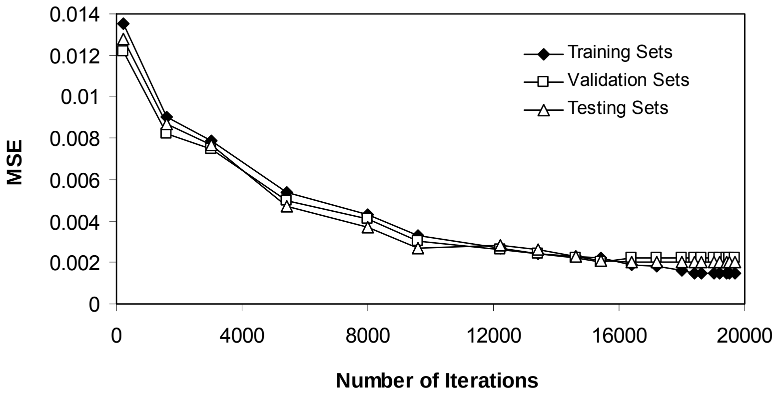

Fig. 8 Comparison of MSE of the Training, Testing and Validation Sets Versus the Number of Iterations

Conclusions

Based on the above study, the following conclusions can be drawn:

> The performance of the 9-15-1 architecture was found to have better than other architectures. That means one hidden layer with a set of 15 neurons has the most reasonable agreement architecture. This number is reached based on our extensive experience with ANNs modelling. Utilizing more than 15 neurons in network makes the computation process complicated and expensive in terms of time. So, in order to have minimum compatibility cost and high accuracy, the number of hidden neurons is fixed to 15.

> The values of 0.07 for learning rate and 0.6 for momentum were the best possible values for suitable learning and preventing oscillations. Tanh (3x) and sig (x) are selected as activation functions in hidden layer and output layer, respectively.

> Calculation of mean percentage relative error for training set data showed that ANN can predict the soil ultimate bearing capacity with an error of 14.83%. Also error for testing and validation set were 15.73% and 15.91%, respectively. This is acceptable in geotechnical engineering practice.

> (iv) The results prove that the ANNs have the advantage that once the model is trained, it can be used as an accurate and quick tool for predicting the soil ultimate bearing capacity with high precision.

Reference

Baziar, M.H. and Ghorbani, A (2005): 'Evaluation of Lateral Spreading Using Artificial Neural Networks', *Soil Dynamics and Earthquake Engineering*, Vol. 25, pp.1-9

Bolton, M. (1979): *A Guide of Soil Mechanics*, Macmillan Press., U.K.

Bowles, J. E. (1988): *Foundation Analysis and Design*, McGraw-Hill Publishing Company Ltd.

Can, I.A (2002): 'A New Improved Na/K Geothermometer by Artificial Neural Networks' ,*Geothermic*, Vol.31, pp.751–760

Cernica, J (1995): *Geotechnical Engineering: Foundation Design*, John Wiley and Sons, Publishing Company Ltd, USA.

Das, B.M (1987): *Theoretical Foundation Engineering*, Elsevier Science Publishers, Netherlands.

Demuth, H., Beale, M. and Hagan, M (2005): *Neural Network Toolbox User's Guide*, Version 4.0.6, The Math Works Inc.

Demuth, H.B. and Hagan, M.T (1996): *Neural Network Design*, PWS Publishing Company, Boston, USA.

Ghaboussi, J., Sidarta, D.E and Lade, P.V (1994): 'Neural Network Based Modeling in Geomechanics', *Computer Methods and Advances in Geomechanics*, Balkema, Rotterdam Publishing; pp. 153-164

Goh, A.T.C (1994): 'Seismic Liquefaction Potential Assessed by Neural Networks', *Geotechnique Engineering*, Vol. 120, No.9, pp.1467-1480

Hakim, S.J.S (2006): *Development and Applications of Artificial Neural Network for Prediction of Ultimate Bearing Capacity of Soil and Compressive Strength of Concrete*, Master thesis, Civil Engineering Department, University Putra Malaysia, Malaysia.

Ince, R (2004): 'Prediction of Fracture Parameters of Concrete by Artificial Neural Networks', *Engineering Fracture Mechanics*, Vol.71, pp. 2143-2159

Kartalopoulos, S.V (2002): 'Understanding Neural Networks and Fuzzy Logic Basic Concepts and Application', IEEE Press, pp.45-52

Kerh, T., Hu, Y.G and Wu, C.H (2003): 'Estimation of Consolidation Settlement Caused by Groundwater Drawdown Using Artificial Neural Networks', *Advances in Engineering,* Vol. 34, pp. 559–568

Kim, C.Y., Bae, G.J., Hong, S.W., Park, C.H., Moon, H.K. and Shin, H.S (2001): 'Neural Network Based Prediction of Ground Surface Settlements due to Tunnelling', Computers and *Geotechnique,* Vol.28, pp.517–547

Lee, M.I and Lee, H.J (1996): 'Prediction of Pile Bearing Capacity Using Neural Networks', Computers and *Geotechnique*, Vol.18, No.3, pp. 189-200

MATLAB 6.5., (2003): The Mathworks. Inc., Natick, Massachusetts

Noorzaei, J., Jaafar, M.S., Thanoon, W.A.M. and Hakim, S.J.S (2005): "Development of an Artificial Neural Network Model for Prediction of Ultimate Soil Bearing Capacity', Proc. 8th Int. Conf. Application of Artificial Intelligence to Civil, Structural and Environmental Engineering, Paper 55, Civil-Comp Press Scotland.

Okine, A.N.O (1999): 'Analysis of Learning Rate and Momentum Term in Backpropagation Neural Network Algorithm Trained to Predict Pavement Performance', Advance in Engineering Software, Vol.30, pp.291-302

Shahin, M.A, Maier, H.R and Jaksa, M.B (2002): 'Predicting settlement of shallow foundations using neural networks', Geotechnical and Geoenvironmental Engineering, Vol.128, No.9, pp.785–93

Technical Report' The Detailed Design, Construction Supervision and Contract Administration of the Immediate Action Plan Multimedia Super Corridor', Ikram College of Technology of Malaysia (ICTM): Malaysia, 1991

Technical Report 'Soil Investigation report', Ikram College of Technology of Malaysia (ICTM): Malaysia, 2001

Technical Report 'Soil Investigation report', Ikram College of Technology of Malaysia (ICTM): Malaysia, 2003.

Technical Report 'Soil Investigation report', Ikram College of Technology of Malaysia (ICTM): Malaysia, 2004

Turk, G., Logar, J. and Majes, B (2001): 'Modeling Soil Behaviour in Uniaxial Strain Conditions by Neural Networks', Advances in Engineering Software, Vol. 32, pp. 805-812

Yeh, Y.C., Kuo, Y.H and Hsu, D.S (1992): 'Building an Expert System for Debugging FEM Input Data With Artificial Neural Networks', Expert Systems with Applications, Vol.5, pp.59-70