Computer Assisted Mechanics and Engineering Sciences, 14: 331–343, 2007.

The prediction of damage condition in regards to damage factor influence of light structures on expansive soils in Victoria, Australia

Norhaslinda Y. Osman, Kerry J. McManus, Huu D. Tran
Faculty of Engineering and Industrial Sciences
Swinburne University of Technology, Melbourne, Australia

Zbigniew A. Krezel
School of Architecture and Building, Deakin University
Geelong, Victoria 3217, Australia

(Received in the final form April 23, 2007)

This paper proposes a neural network model using genetic algorithm for a model for the prediction of the damage condition of existing light structures founded in expansive soils in Victoria, Australia. It also accounts for both individual effects and interactive effects of the damage factors influencing the deterioration of light structures. A Neural Network Model was chosen because it can deal with 'noisy' data while a Genetic Algorithm was chosen because it does not get 'trapped' in local optimum like other gradient descent methods. The results obtained were promising and indicate that a Neural Network Model trained using a Genetic Algorithm has the ability to develop an interactive relationship and a Predicted Damage Conditions Model.

1. Introduction

Damage to light structures founded in and on expansive soils occurs throughout the world. There is a number of studies performed on damage prediction for light structures [4, 7], but no Predicted Damage Conditions (PDC) Model has been developed for Victoria, Australia. Victoria is 228 000 km^2 in area and is the smallest and most densely populated Australian mainland state. It is bounded on the South and East by the Indian Ocean, Bass Strait, and the Tasman Sea. Approximately 50% of the surface area in Victoria is covered by moderate to highly expansive soils; mostly derived from Tertiary, Quaternary and Volcanic deposits [14].

Volume change of the expansive soil with change in moisture content can cause damage to light structures founded in and on the soil, such as houses and shallow pipelines. For example, a decade ago, more than 50% of the 20 000 houses owned by the Building Housing Commission (BHC) required extensive repairs due to expansive soil movement. It is predicted that approximately 30 000 new dwellings will be affected annually by expansive soil movement, increasing the building costs in Victoria alone, by AU$ 60M–90M [15].

The aim of this paper is to investigate the Interactive Relationship (IR) of the damage factors influencing the behaviour of light structures on expansive soils using a Neural Network (NN) model trained with weights generated initially by a Genetic Algorithm (GA) and to apply them to predict the future damage condition in existing light structures in Victoria, Australia. Approximately 400 cases of damage to light structures were considered in this study. A hybrid training strategy using Genetic Algorithm and back-propagation was used to train Neural Network where Genetic Algorithm generates the initial weights for Backpropagation (FB) to complete the training process. The training and learning process in Neural Network used the 'initial' weights obtained from Genetic

Algorithm to determine the interactive relationship between the Damage Factor Influence (DFI) such as Region (R), Construction Wall type (CW), Construction Foundation type (CF), Thornthwaite Moisture Index old and new (TMIO, TMIN), Geology (G), Vegetation covers (V) and Age of building when first inspected (A) was investigated. The performance errors were noted. Then the interactive relationship was used to develop the Predictive Damage Condition model. To estimate the accuracy of the model, the predictive error was calculated and compared to the original model where only the individual Damage Factor Influence (DFI) was used.

Neural Network was chosen because it is fault tolerant [6] while Genetic Algorithm was chosen because it improves the learning process in Neural Network where it generates the initial weights for Neural Network. Genetic Algorithm can be used even though the error gradient information is unavailable or when the transfer function of the neurons is not discontinuous Genetic Algorithm has the ability to solve large, non-linear, and complex problems more accurately than other methods [26]. Genetic Algorithm is presumed to perform a global search of the weight space and should be less likely to become stuck in a local minimum than feedforward backpropagation, which pursues only a single route along the weight spaces [26]. However, it should not be regarded as a replacement for other existing methods but rather as another optimisation approach.

2. Literature review

Artificial intelligence techniques such as Neural Network and Genetic Algorithm are still new to engineers in predicting the behaviour of expansive soils and the prediction of the location and time of damage to light structures founded on expansive soils. Most of the studies involved the investigation and prediction of movement of individual structure in a building for example beams and footings caused by seismic movement and founded on non-expansive soils such as sand. The techniques are however seldom used to predict settlement of foundation on cohesive soils such as clay.

The first Neural Network which was inspired by biological nervous systems was designed by McCulloch and Pitts (1943) cited in [5]. As in nature, the network function is determined largely by the connections between elements. The goal of the network is to learn or to discover some association between input and output patterns, or to analyse, or to find the structure of the input patterns [1]. The learning or training process is achieved through the modification of the connection weights between units. Given a sample of input and output vectors, Neural Network "learn" this relationship, and store this learning into their parameters [13]. Neural Network can be used to determine which variables or parameters are important and to build a model relating to those parameters [3].

Genetic Algorithm is based on a Darwinian-type survival of the fittest strategy was invented by [16] in the 1960's. Most organisms evolve by means of two primary processes: natural selection and sexual reproduction [10]. The first determines which members of population survive and reproduce, and the second ensures mixing and recombination among the genes of their offspring. Potential solutions to a problem compete and mate with each other (cross-over) in order to produce increasingly stronger individuals. This mixing allows creatures to evolve much more rapidly than they would if each offspring simply contained a copy of the genes of a single parent, modified occasionally by mutation [10]. Genetic Algorithm works on a random set of points in the population by using a set of operators that are applied to the population. The different operators are scaling, selection, reproduction, crossover, migration and mutation.

Examples of Artificial intelligence techniques using Neural Network and Genetic Algorithm include work by [8, 9, 12, 17–24]. [12] presented a model to locate and assess the damage occurring at any position in a cantilever beam by backpropagation neural network considering displacement and strain as input parameter to the network. From the review done by [23], it is evident that Neural Network has been applied successfully to many geotechnical engineering areas which includes pile capacity prediction, settlement of foundations, soil properties and behaviour, liquefaction, site characterisation, earth retaining structures, slope stability and the design of tunnels and underground openings. [17, 19–21] have adopted a Neural Network trained with Genetic Algorithm. The performance of the model was better compared to just using an individual Artificial Intelligence technique. The results indicated that Neural Network and Genetic Algorithm have the ability to predict the potential damage to light structure on expansive soils when presented with any kind of data be it with missing or with complete parameters. There is no doubt that Neural Network can adapt to changes in the dataset as it can always be updated to obtain better results by preventing new training examples as new data becomes available [19]. Genetic Algorithm on the other hand is very useful in improving the learning process of Neural Network [21].

Neural Network trained with Genetic Algorithm was adopted in the development of the predictive damage condition model and the interactive relationship between factors which influence damage to light structures on expansive soils in this paper.

3. APPROACHES AND PROCESSES

Approximately 400 sets of data from Building Housing Commission database were used in this paper. All the data was coded into numeric form. The data set was divided into nine categories where eight categories are the input and one category is the output.

Input

There were eight categories for the input.

Regions (R)

The state of Victoria was divided administratively into six geographical regions, viz. Melbourne, South East Victoria, North East Victoria, North West Victoria, West Victoria and South West Victoria. Since most of the data from Building Housing Commission focused on Melbourne region, the Melbourne region was divided again into nine smaller regions; inner, inner eastern, outer eastern, western, southern, south east, north, north east and Mornington to give more precision to the analysis. Table 1 shows the coding for numeric data.

Table 1. Numeric data of R

Input category for R	Numeric data
Inner Melbourne	1
Inner Eastern Melbourne	2
Outer Eastern Melbourne	3
Western Melbourne	4
South Melbourne	5
South East Melbourne	6
North Melbourne	7
North East Melbourne	8
Mornington	9
South East Victoria	10
North East Victoria	11
West Victoria	12
North West Victoria	13
South West Victoria	14

Construction Wall type (CW)

This category refers to the type of wall used for the light structure. Table 2 shows the coding for numeric data.

Table 2. Numeric data of CW

Input category for CW	Numeric data
No data	0
Brick veneer	1
Cavity wall	2
Blocks	3
Clad frame	4
Solid brick/full masonry	5
Reinforced concrete	6
Precast concrete	7
Double brick	8

Construction Footing type (CF)

This category refers to the type of footing used for the light structure. Table 3 represents the coding for numeric data.

Table 3. Numeric data of CF

Input category for CF	Numeric data
No data	0
Concrete slab footing	1
Strip footing	2
Raft slab footing	3
Bluestone	4
Stump	5
Stiffened slab	6

TMI Old (TMIO) and TMI New (TMIN)

TMI is the classification of climate based on potential evapotranspiration, areal evapotranspiration and rainfall [25]. TMIO was computed for 1940–1960 while TMIN was computed for 1961–1990. Table 4 shows the coding for numeric data for both TMIO and TMIN.

Table 4. Numeric data of TMIO and TMIN

Input category for TMIO and TMIN	Numeric data
−25	1
−20	2
−5	3
0	4
5	5
10	6
30	7

Geology (G)

This category is the geological classification of the site by rock type in accordance with Geological map of Victoria [14]. Table 5 shows the coding for numeric data.

Table 5. Numeric data of G

Input category for G	Numeric data
Quaternary	1
Tertiary	2
Volcanic	3
Silurian	4
Upper Devonian	5
Jurassic	6
Ordovician	7

Type of vegetation cover (V)

This category represents the type of vegetation currently existing on site such as built up area, native grassland, native forests, horticultural trees and shrubs. This data was categorised based on the Bureau of Rural Sciences map [11]. Table 6 shows the coding for numeric data.

Table 6. Numeric data of V

Input category for V	Numeric data
Built Up	1
Annual Crops and Highly modified pastures	2
Native grassland and minimally modified pastures	3
Native forests and woodlands	4
Horticultural trees and shrubs	5

Age (A)

This category refers to the age of structure when it was first inspected, following a report of damage. Table 7 shows the coding for numeric data.

Table 7. Numeric data of Age

Input category for A (Years)	Numeric data
1 to 10	1
11 to 20	2
21 to 30	3
31 to 40	4
41 to 50	5
51 to 60	6
61 to 70	7
71 to 80	8
81 to 90	9
91 to 100	10
101 to 110	11
111 to 120	12

Output

The Damage Classification (DC) was set as an output. Table 8 shows the numeric data of Damage Condition. The damage classification refers to the Australian Standard, AS2870 [2] guideline. The measurements were based on visual inspection. They were measured using simple instruments such as a ruler and a spirit level to check for size of cracks and amount of movement. The measurements were taken on interior and exterior walls, ceilings, floors and foundation of the light structure where the movement occurs.

Table 8. Numeric data of DC

Description	Damage classification
Hairline crack < 0.1 mm	0
Fine crack < 1 mm	1
Distinct crack < 5 mm with noticeably change in level	2
Wide crack > 5 mm & < 15 mm with obvious change in level	3
Severe crack < 15 mm with disturbing change in level	4

4. Development of Neural Network Model

The Genetic Algorithm and Neural Network toolboxes for MATLAB were used to support computational tasks. A feed forward neural network with eight input nodes, four neurons in one hidden layer and one output layer were selected among other topologies of the Neural Network by using trial and error search methods. The aim of the learning procedure in Neural Network is to update the weights of the links connecting the nodes and to minimize the average squared system error between the observed and the computed outputs [6]. Genetic Algorithm was used to initially generate the weights for Neural Network prior to the learning process. Figure 1 shows a model of Neural Network trained by weights generated by Genetic Algorithm.

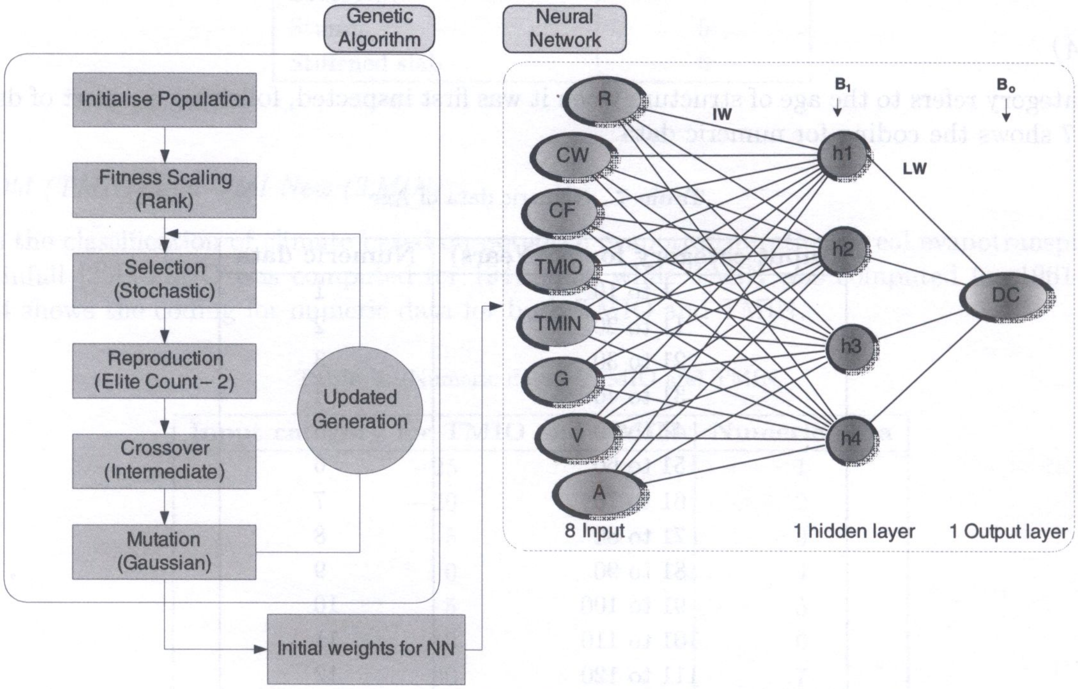

Fig. 1. A model of Neural Network trained using weights generated by Genetic Algorithm

Since the data from the Building Housing Commission database is 'noisy' because of incomplete data, Neural Network was chosen as it can handle noisy data. Genetic Algorithm on the other hand was chosen because it cannot be easily 'trapped' in local optimum like Feedforward Backpropagation and other gradient descent methods and also it strives to optimise the fitness value.

Genetic Algorithm

The parameters used in Genetic Algorithm toolbox for MATLAB were set as Table 9 [26].

Table 9. Parameters of Genetic Algorithm

Parameters	Description
Population	The population size = 80, Initial range = between −1 and +1. The population size is used as a trade-off between the time of convergence and the fitness value of the final result [26].
Fitness scaling	A ranking mechanism is used in order to maintain constant selective pressure as the algorithm progresses and the fitness values are closer together.
Selection	A stochastic uniform selection mechanism is used.
Reproduction	Every newly generated chromosome is unconditionally inserted into the population of the next generation [26]. The reproduction type is children replacing their parent. Elite children are the individuals in the current generation with the best fitness value which automatically survive to the next generation. The default value equals to 2 of elite count with a crossover function of 0.8 was chosen.
Crossover	Combining the vectors of a pair of parents creates crossover children. Here, an intermediate function with a ratio of 1 was chosen.
Stopping criteria	The following are set in Genetic Algorithm as the stopping criteria. Generations = 3000, stall generation = 50, time limit = ∞, fitness limit = $-\infty$ and stall time limit = ∞.
Mutation	Introducing random mutations to a single parent creates mutation children. Gaussian addition was chosen because the data uses real numbers. The shrink and scale controls were set to the default value of 1.
Variables	For a two layer Neural Network model, the number of variables (V) for Genetic Algorithm to calculate the weights is calculated using Eq. (1). $V = (m \times n) + 2n + 1$ (1) where m is number of input and n is number of neurons in the hidden layer.

Neural Network

A feed forward Neural Network with eight input nodes, one hidden layer with four neurons and one output layer was adopted. A supervised feedforward backpropagation was used where the output corresponding to the input data is immediately available. The weights generated in Genetic Algorithm were set in Neural Network. During the training phase of the Neural Network, the training process continuously modifies the values of the weights until some previously agreed criteria are met. An Activation Function is included in the Neural Network to allow for varying input conditions and their effect on the output. The operation of the Neural Network comes from the processing of a series of signals from input to hidden layer and from hidden layer to output. The following were used in Neural Network:

1. Input to hidden layer

$$\text{Sum input,} \quad X = \sum IW + B_I \tag{2}$$

where IW is input multiplied by weights from input to hidden layer and B_I is the bias weight for hidden layer neuron. Logsigmoid activation function (T_L) was used to transfer the sum input to hidden layer. Logsigmoid was used to ensure that the network is limited to a small range and it can account for non-linear relationship in the network. Equation (3) shows the Logsigmoid function,

$$T_L = 1/(1+\exp(-X)). \tag{3}$$

2. Hidden to output layer.

   The sum of all the neurons in the processing from input to hidden layer was then fed in as a ramp function (T_R) which acts as the activation function for the hidden to output layer. Equation (4) shows the sum of signal from the input to hidden layer,

$$\text{Sum input to hidden,} \quad Y = \sum T_L LW + B_o \tag{4}$$

where B_o is the bias weight for the output layer. Then T_R was used to transfer the sum into the output. It was used because the output has fixed values from 0 to 4. In this model, the maximum and minimum of ramp function are 4 and 0 respectively,

$$T_R = \begin{cases} \text{Max} & \text{when } Y > \text{upper limit}, \\ Y & \text{when } Y < \text{upper limit and } Y > \text{lower limit}, \\ \text{Min} & \text{when } Y < \text{lower limit}. \end{cases} \tag{5}$$

The performance error (Ep) of the Neural Network was then calculated using Eq. (6),

$$\text{Ep} = \frac{\left[\sum_{k=1}^{K}(d_k - O_k)^2\right]}{K}. \tag{6}$$

where d_{pk} is the target, O_{pk} is the output and K is the number of data points.

4.1. Interactive relationship (IR) model development

From the training process, an interactive relationship model was developed. New dataset matrices were developed in regards to the Interactive relationship of individual damage factor influence which are Region (R), Construction Wall (CW), Construction footing (CF), TMIO, TMIN, Geology (G), Vegetation (V) and Age (A). This was done to determine different combination of the interaction and also its effect on the model. It also improves the performance of the predictive model. The performance error for each of the matrix was noted. Table 10 shows the performance error for Genetic Algorithm and Neural Network. The following illustrate the interactive relationship with each damage factor influence.

$$\begin{aligned}
\text{Matrix } \mathrm{R_{IR}} &= [\mathrm{R(CW, CF, TMIO, TMIN, G, V, A)}] \\
\text{Matrix } \mathrm{CW_{IR}} &= [\mathrm{CW(R, CF, TMIO, TMIN, G, V, A)}] \\
\text{Matrix } \mathrm{CF_{IR}} &= [\mathrm{CF(R, CW, TMIO, TMIN, G, V, A)}] \\
\text{Matrix } \mathrm{TMIO_{IR}} &= [\mathrm{TMIO(R, CW, CF, TMIN, G, V, A)}] \\
\text{Matrix } \mathrm{TMIN_{IR}} &= [\mathrm{TMIN(R, CW, CF, TMIO, G, V, A)}] \\
\text{Matrix } \mathrm{G_{IR}} &= [\mathrm{G(R, CW, CF, TMIO, TMIN, V, A)}] \\
\text{Matrix } \mathrm{V_{IR}} &= [\mathrm{V(R, CW, CF, TMIO, TMIN, G, A)}] \\
\text{Matrix } \mathrm{A_{IR}} &= [\mathrm{A(R, CW, CF, TMIO, TMIN, G, V)}]
\end{aligned}$$

Table 10. Performance error for the matrix

Matrix	Performance error for Genetic Algorithm	Performance error for Neural Network
R_{IR}	0.260	0.222
CW_{IR}	0.225	0.149
CF_{IR}	0.249	0.159
$TMIO_{IR}$	0.266	0.229
$TMIN_{IR}$	0.237	0.180
G_{IR}	0.249	0.208
V_{IR}	0.223	0.178
A_{IR}	0.230	0.198

Table 11. Best results for interactive relationship in each matrix interaction

Interactive Relationship	Connection weight
R*(TMIN)	11.33
CW*(TMIN)	51.10
CF*(V)	72.73
TMIO*(CF)	4.32
TMIN*(A)	21.52
G*(TMIO)	453.11
V*(TMIN)	31.81
A*(V)	32.21

The connection weights for each combination were calculated using Eq. (7).

Connection weight =

$$= \sum \left\{ \sum (\text{connection of input–hidden})_{IR} \times \sum (\text{connection of hidden–input})_{IR} \right\}. \quad (7)$$

The best 8 results from the 56 possible combinations selected on the basis of their connection weights are summarised in Table 11.

4.2. Predictive Damage Condition (PDC) model

Another dataset was developed for the development of the Predictive Damage Condition model. This is the combination of the original dataset with damage factor influence which are Region (R), Construction Wall (CW), Construction Footing (CF), TMIO, TMIN, Geology (G), Vegetation (V) and Age (A); and the dataset for the best results for interactive relationship (Table 11). A two layer Neural Network was used with sixteen input nodes, one hidden layer with six neurons and one output layer. Here, the same procedure was used to generate the weights for Neural Network. Figure 2 shows the architecture of Neural Network for Predictive Damage Condition.

The results obtained after training the Neural Network with weights generated in Genetic Algorithm are as below:

Performance error for Genetic Algorithm = 0.257,

Performance error for Neural Network = 0.215.

A predictive damage condition model was developed in accordance to the following equation,

$$\text{PDC} = f[\text{R}, \text{CW}, \text{CF}, \text{TMIO}, \text{TMIN}, \text{G}, \text{V}, \text{A}, \text{TMIN}(\text{R} + \text{CW} + \text{V}), \text{TMIO}(\text{G}), \text{V}(\text{CF} + \text{A})]. \quad (8)$$

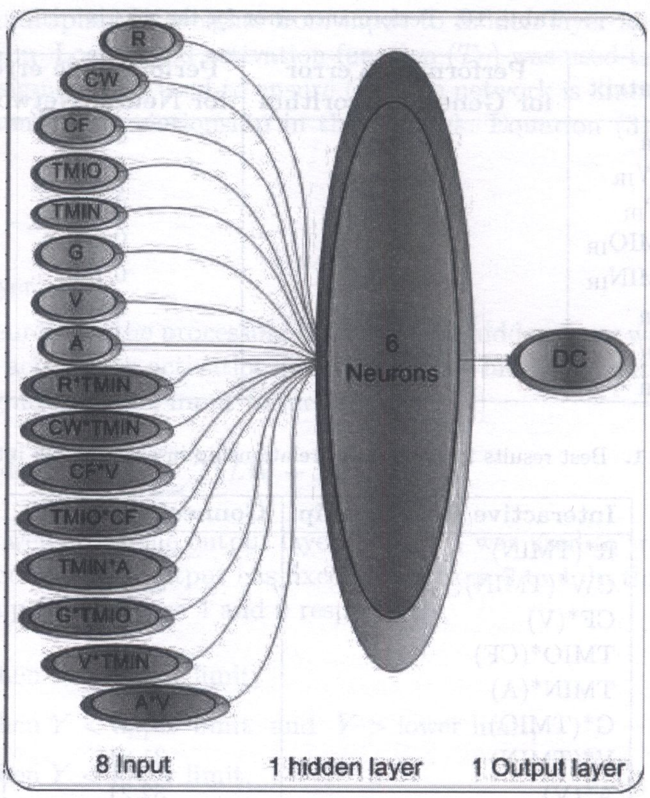

Fig. 2. Architecture of Neural Network for Predictive Damage Condition

Table 12. Property information

Damage Factor Influence	Description of input category	Numeric data
Region (R)	Inner Melbourne	1
Construction Wall (CW)	Brick Veneer	1
Construction Footing (CF)	Concrete slab footing	1
TMIO	−20	2
TMIN	10	6
Geology (G)	Volcanic	3
Vegetation (V)	Built up area	1
Age (A)	5 years old	1

Equation (8) is rewritten as

$$\mathrm{PDC} = f[\mathrm{R}, \mathrm{CW}, \mathrm{CF}, \mathrm{TMIO}, \mathrm{TMIN}, \mathrm{G}, \mathrm{V}, \mathrm{A}, X, Y, Z] \tag{9}$$

where

$$X = f[\mathrm{TMIN} * (\mathrm{R} + \mathrm{CW} + \mathrm{V})],$$
$$Y = f[\mathrm{TMIO} * \mathrm{G}],$$
$$Z = f[\mathrm{V} * (\mathrm{CF} + \mathrm{A})].$$

From Eq. (9), the damage condition can be predicted for an existing light structure. For example, if a property has the following information as in Table 12, then the damage condition of the property can be predicted using Eq. (9).

$$\mathrm{PDC} = f(\mathrm{R}, \mathrm{CW}, \mathrm{CF}, \mathrm{TMIO}, \mathrm{TMIN}, \mathrm{G}, \mathrm{V}, \mathrm{A}, X, Y, Z) = f(1, 1, 1, 2, 6, 3, 1, 1) = 0.$$

PDC = 0 means the property has a minor damage with hairline crack < 0.1 mm (ref. Table 8).

4.3. Model without interactive relationship (Original model)

This model takes into account only individual effect of damage factor influence. This model was developed to compare with the Predictive Damage Condition model. A dataset with damage factor influence was trained using the same procedure. The results obtained after training the Neural Network with weights generated in Genetic Algorithm are as follows,

Performance error for Genetic Algorithm = 0.281,

Performance error for Neural Network = 0.220.

From the performance error point of view, Predictive Damage Condition model was much better than the original model as it has low performance error. Thus, the Predictive Damage Condition model will give a better prediction than a model without interactive relationship.

4.4. Test of model

The Predictive Damage Condition model with interactive relationship was tested using a dataset which was different with the learning dataset. The tested result of Predictive Damage Condition model was also compared with similar tested result of original model. The aim was to see the accuracy of the proposed model. It was found that the prediction error was 46% compared to the original model with predictive error of 49% therefore interaction model was better.

5. Findings

The following findings were determined for the prediction of damage condition in regards to damage factor influence of light structures on expansive soils in Victoria, Australia:

1. *Neural Network trained using hybrid Genetic Algorithm and Backpropagation.* By generating weights in Genetic Algorithm, it can be seen that it improves the performance of the model when training Neural Network. From Table 10, all the performance error improved in average of 30% after training the Neural Network with weights generated from Genetic Algorithm.
2. *Interactive Relationship model.* From the results, it was obvious that there was an interactive relationship between all the damage factor influence especially TMIN, TMIO and Vegetation. This means that one way or another, the individual damage factor influence rely on each other. As an example, the designs of walls and footings were established with respect to TMIO.

   Codes such as Australian Standard, AS2870 [2] did not accommodate change in TMI over the life of the structure. Therefore, in Table 11, Construction Wall relies on TMIN, as the design should incorporate TMIN. Here, the performance error was also the lowest among all the interactive relationship. This shows that the interactive relationship between Construction Wall and TMIN was the most significant.

3. *Predictive Damage Condition model.* A new model for the prediction of damage, (PDC model), was developed. It can be seen that the Predictive Damage Condition model was better than the original model as it gives a more accurate result. Although the error difference between the two was small, it is still suggestive that interaction between inputs should be seriously considered.

6. Conclusion

From the analysis, a model of predicted damage condition for light structures founded in and on expansive soil was developed. The preliminary results are promising and an improvement on existing predictive systems. The prediction model based on Neural Network and Genetic Algorithm proved to be useful in estimating potential damage to light structures founded in and on expansive soils in Victoria, Australia.

All the damage factors were interrelated in one way or another. The damage factor influence; Region, Construction Wall and V showed that they were very dependent on TMIN while Geology was dependent on TMIO and Construction Footing and Age was dependent on Vegetation. The analysis results show that TMIN, TMIO and Vegetation were dominant and important in relation to their effect on damage condition. There is scope for developing a set of independent damage factors to improve the model.

The results indicate that Neural Network and Genetic Algorithm have the ability to develop an interactive relationship and to predict the potential damage to light structure when presented with "noisy" data. The advantage of Neural Network was that it can always be updated to obtain better results by presenting new training examples as new data becomes available. Genetic Algorithm on the other hand was very useful in improving the learning process of Neural Network.

References

[1] H. Abdi. Neural Networks. *Encyclopedia of Social Sciences Research Methods: Quantitative Applications in the Social Sciences*, vol. **124**, 2003.

[2] AS2870-1996 *Residential slabs and footings-construction*. Standards Australia International, Australia, 1996.

[3] J.P. Bigus. *Data Mining with Neural Networks — Solving Business Problems from Application Development to Decision Support*. McGraw-Hill, New York, 1996.

[4] J.H. Chou, J. Ghaboussi. Structural damage detection and identification using Genetic Algorithm. *Computers and Structures*, **79**: 1335–1353, 2001.

[5] H. Demuth, M. Beale. *Neural network toolbox for use with Matlab: User's Guide - Version 4*. The MathWorks, 2001.

[6] L. Fausett. *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*. Prentice Hall, 1994.

[7] M.I. Friswell, J.E.T. Penny, S.D. Garvey. A combined genetic algorithm and eigensensitivity algorithm for the location of damage in structures. *Computers and Structures*, **69**: 547–556, 1998.

[8] G. Habibagahi, A. Bamdad. A neural network framework for mechanical behaviour of unsaturated soils. *Canadian Geotechnical Journal*, **40**: 684–693, 2003.

[9] H. Hao, Y. Xia. Vibration-based damage detection of structures by genetic algorithm. *Journal of Computing in Civil Engineering*, **16**: 222–229, 2002.

[10] J.H. Holland. Genetic Algorithms: Computer programs that 'evolve' in ways that resemble natural selection can solve complex problems even their creators do not fully understand. *Scientific American*, 44–50, 1992.

[11] *Integrated Vegetation Map Online*. Bureau of Rural Sciences, Commonwealth of Australia, 2003.

[12] D. Maity, A. Saha. Damage assessment in structure from changes in static parameter using neural networks. *Sadhana – Academy Proceedings in Engineering Sciences*, **29**: 315–327, 2004.

[13] I. Maqsood, M.R. Khan, A. Abraham. Intelligent weather monitoring systems using connectionist models. *Neural, Parallel & Scientific Computations*, **10**: 157–178, 2002.

[14] J. McAndrew. The geological map of Victoria in geology of Australia ore deposits. Presented at *The Eighth Commonwealth Mining and Metallurgical Congress*, Australia and New Zealand, 1965.

[15] K.J. McManus, D. Lopes, N.Y. Osman. The effect of Thornthwaite Moisture Index changes in ground movement predictions in Australian soils. *Proceedings 9th Australia New Zealand Conference on Geo-mechanics*, Auckland, New Zealand, **2**: 555–561, 2004.

[16] M. Mitchell. *An Introduction to Genetic Algorithms*. Massachusetts Institute of Technology Press, 1996.

[17] N.Y. Osman, D. Lopes, K.J. McManus. An artificial intelligence examination of the influence of geological conditions and changes in climate on damage to light structures in Victoria. Presented at *7th Young Geotechnical Professionals Conference*, Adelaide, 2006.

[18] N.Y. Osman, K. McManus, A.W.M. Ng. Management and analysis of data for damage of light structures on expansive soils in Victoria, Australia. *Proceedings of the 1st International conference on Structural Condition Assessment, monitoring and Improvement*, Perth, Australia, 2005.

[19] N.Y. Osman, K.J. McManus. The ranking of factors influencing the behaviour of light structures on expansive soils in Victoria, Australia. *Proceedings of the Eighth International Conference on the Application of Artificial Intelligence to Civil, Structural and Environmental Engineering*, Rome, Italy, 2005.

[20] N.Y. Osman, K.M. McManus, H.D. Tran, Z.A. Krezel. The prediction of damage condition in regards to damage factor influence of light structures on expansive soils in Victoria, Australia. Presented at *International Symposium on Neural Networks and Soft Computing in Structural Engineering*, Cracow, Poland, 2005.

[21] N.Y. Osman, A.W.M. Ng, K.J. McManus. Selection of important input parameters using neural network trained with genetic algorithm for damage to light structures. *Proceedings of the Fifth International Conference on Engineering Computational Technology*, Las Palmas de Gran Canaria, Spain, 2006.

[22] C. Ratnam, D.V. Parameswara. Identification of damage in structures using genetic algorithms. *IE (I) Journal - MC*, **84**: 154–160, 2004.

[23] M.A. Shahin, M.B. Jaksa, H.R. Maier. Artificial neural network applications in geotechnical engineering. *Australian Geomechanics*, 49–62, March 2001.

[24] M.A. Shahin, H.R. Maier, M.B. Jaksa. Predicting settlement of shallow foundations using neural networks. *Journal of Geotechnical and Geoenvironmental Engineering*, **128**: 785–793, 2002.

[25] C.W. Thornthwaite. An approach towards a rational classification of climate. *Geographical Review*, **38**: 55–94, 1948.

[26] A.J.F. Van Rooij, L.C. Jain, R.P. Johnson. *Neural Network Training using Genetic Algorithm.* World Scientific Publishing, 1996.