Applying Neural Network Model in Seismic Liquefaction Case Analysis: probabilistic neural network model v.s. multilayer perceptrons model

Chung-Jung LEE, Ta-Kang HSIUNG
Dept. of Civil Engineering, National Central University, TAIWAN
No.300, Jungda Rd, Chung-Li City, Taoyuan, TAIWAN 32054
Email: cjleeciv@cc.ncu.edu.tw ; s1342006@cc.ncu.edu.tw

Abstract— **In this paper the capabilities and powerfulness of neural network models to classify the seismic liquefaction potential of many field cases were presented. Two neural network models, the probabilistic neural network (PNN) model, which developed based upon the Bayesian strategy for pattern classification, and the well-known multilayer perceptron (MLP) model were used to implement the analysis to identify the liquefied and non-liquefied cases during earthquakes. Field case records derived from in-situ test measurement, CPT-q_c, were collected and compiled. Both the PNN and the MLP mdoel were created and trained by the same 75% of cases, which randomly selected from all the ones we gathered and then tested by the other 25% of cases. The major features and differences between using these two models in the liquefaction potential identification were also presented and discussed. Analysis results show that both models give nearly perfect performance on the classification of liquefaction potential, but the MLP model performs slightly higher rate of recognition. However, more searching time was required in MLP model to overcome the local minima problem that could interrupt the back-propagation error correction approach to find the optimal result.**

I. INTRODUCTION

Soil liquefaction is known as one of the most severe seismic hazards that can damage structures founded on both shallow and deep foundations and disrupt buried lifelines in the ground. It has been widely seen in loose sand deposits. Recently, the widespread soil liquefaction and the related damages occurred in Yaulin, Wufeng, and in the parts of Taichung Harbor and caused a lot of loss of property in the center of Taiwan during the Chi-Chi earthquake, 1999 [1,13].

In earthquake engineering practice, it is of substantial importance to identify the areas vulnerable to liquefaction and then to mitigate possible damages on them by taking appropriate measures in advance. Evaluation of liquefaction potential is a complicated multivariable problem and needs to find out key parameters, which control liquefaction occurrence, including earthquake parameters, in-situ soil properties, and stress conditions. Several methods have been proposed to evaluate liquefaction potential. These methods range from purely empirical to highly analytical and require various degrees of laboratory and/or in-situ testing. It is more common practice to employ in situ tests such as the Standard Penetration Test (SPT), the Cone Penetration Test (CPT), or the Shear Wave Velocity Test (V_s) to determine liquefaction resistance of saturated sandy soils. These simplified methods were generally presented in a chart that defines the boundary of liquefaction and non-liquefaction in a plot of cyclic resistance ratio (CRR) versus the corrected SPT-N values $(N1)_{60}$, the corrected CPT tip resistance (q_{c1}), and the corrected Vs, respectively [2, 3, 4, 5].

In recent years, a useful and powerful computation tool, Artificial Neural Networks (ANN), was introduced for solving the complicated multifactor problem. It is a model-free approach and does not require any prior knowledge to setup the probable model-type of the relationship indicated by original data. Many researchers have reported that liquefaction potential can be estimated more accurately than by the conventional simplified procedures calibrated by many seismic liquefaction case histories [6, 7, 8, 9]. Recently most of researchers used the Multi-Layer Perceptron (MLP) as the ANN model to analysis the liquefaction problem, while Goh used the Probabilistic Neural Network (PNN), originally proposed by Dr. Specht [10], to evaluate the liquefaction potential [11]. Although some studies have been done to recognize liquefaction potential by using the PNN model and MLP model trained with CPT-q_c cases, little effort was made on discussing the major difference of the performance between the two models.

In this study, a total of 315 CPT-based in-situ seismic cases of liquefied and non-liquefied site investigation, including the cases of Chi-Chi earthquake in Taiwan, were collected, complied, and used to train and test the PNN model and MLP model, respectively. The major features and differences between using these two models in the liquefaction potential evaluation were also presented and discussed in detail.

II. NEURAL NETWORK MODELS

A. Multilayer Perceptron (MLP) Model

The multilayer perceptron (MLP) model is one of the most fundamental and popular multilayer feedforward networks and has been successfully employed in many diverse applications. Figure 1 shows a typical architecture of MLP consisting of three layers of interconnected neurons. Each neuron in the layer is connected to all neurons in the next higher layer, and each connection has a weight (a scalar) associated with it. These weights determine the nature and strength of the influence between the interconnected neurons. The neurons in the hidden layer play the roles of nonlinear transformation that enable the MLP to simulate a more complex and nonlinear system. Sometimes the number of hidden layer may be more than one, however, one hidden layer is good enough to simulate a nonlinear problem in practice.

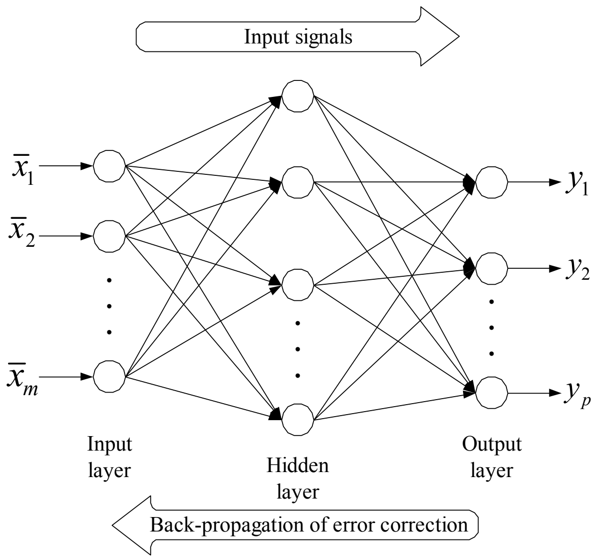

Fig. 1 Typical architecture of MLP model

The well-known back-propagation algorithm is used as a learning mechanism to correct the connection weights iteratively and to minimize the system error produced by each forward processing of input signal in the MLP. The system error in the nth training pattern, $E(n)$, is defined as

$$E(n) = \frac{1}{2}\sum_{j=1}^{p}\left(d_j(n) - y_j(n)\right)^2 \tag{1}$$

where $d_j(n)$ and $y_j(n)$ are the jth component of desired output and computed output, respectively; p is the number of neuron in the output layer. The incremental correction of each interconnection weight then can be computed by

$$\Delta w_{ji}(n) = -\eta\frac{\partial E(n)}{\partial w_{ji}(n)} + \alpha\cdot\Delta w_{ji}(n-1) \tag{2}$$

where $\Delta w_{ji}(n)$ is the incremental correction of the interconnection weight between neuron-i and neuron-j; $\Delta w_{ji}(n-1)$ is the incremental correction in the last iteration; η is the learning rate and its range is $0<\eta<1$; α is the momentum factor and its range is $0\le\alpha<1$.

The MLP has the generalized curve-fitting capability by using the incremental adaptation approach. However, this approach is time-consuming and is susceptible to falling in false local minima in practice. To improve those drawbacks, Specht [10] introduced the PNN model, a feedforward neural network that can classify patterns using the Bayesian strategy.

B. Probabilistic Neural Network (PNN) Model

The PNN is the Bayesian classifier technique that has been widely used in many classical pattern-recognition problems. It can quickly form nonlinear decision boundaries from the existing input examples and then use the boundaries to classify patterns into any number of classifications. Since the PNN is developed based upon the Bayesian strategy, it has been proved that the decision boundary implemented by the PNN can asymptotically approach the Bayesian optimal decision surface under the limit conditions. By using this optimal decision surface to perform a pattern classification, the "expected risk" resulted from the misclassification can be minimized.

Consider a two-category problem in which any pattern vector $\bar{x}$ with m dimensions that belongs to one of two classes (Class A or B) and the states of nature can be labeled as θ_A or θ_B. According to the Bayesian strategy for pattern classification, the classification rules can be expressed as follows:

$$\begin{aligned} d(\bar{x}) &= \theta_A \quad \text{if} \quad h_A I_A f_A(\bar{x}) > h_B I_B f_B(\bar{x}) \\ d(\bar{x}) &= \theta_B \quad \text{if} \quad h_A I_A f_A(\bar{x}) < h_B I_B f_B(\bar{x}) \\ \bar{x}^T &= \left[\bar{x}_1 \quad \bar{x}_2 \quad \bar{x}_3 \quad \cdots \quad \bar{x}_{m-1} \quad \bar{x}_m\right] \end{aligned} \tag{3}$$

where $f_A(\bar{x})$ and $f_B(\bar{x})$ are the probability density functions for Class A and B, respectively; I_A is the loss function associated with the decision $d(\bar{x})=\theta_B$ while $\bar{x}$ belongs to Class A; I_B is the loss function associated with the decision $d(\bar{x})=\theta_A$ while $\bar{x}$ belongs to Class B; h_A is the priori probability of occurrence of patterns from Class A; and h_B is the priori probability of occurrence of patterns from Class B ($h_A+h_B=1$).

Note that in equation (3) the priori probabilities h_A and h_B are known and can be estimated accurately. Determination of the loss functions I_A and I_B requires the subjective evaluation but the loss functions usually are set to be equal (i.e. $I_A=I_B$). Thus the key to using equation (3) is the ability to estimate the probability density functions $f_A(\bar{x})$ and $f_B(\bar{x})$ based on those discrete patterns from Class A and B. There are so many well-developed probability density function estimators (PDF estimators) that can be used. In this study, the simple summation of the multivariate Gaussian distributions centered at each training sample is used as a PDF estimator and it can be expressed as:

$$f_A(\bar{x}) = \frac{1}{(2\pi)^{m/2}\sigma^m}\frac{1}{N_A}\sum_{i=1}^{N_A} exp\left[-\frac{\left(\bar{x}-X_{Ai}\right)^T\left(\bar{x}-X_{Ai}\right)}{2\sigma^2}\right] \tag{4}$$

where m is dimensionality of input patterns; N_A is the total numbers of training patterns from Class A; X_{Ai} is the i^{th} training pattern from Class A; σ is a smoothing parameter.

A four-layer feedforward architecture is used to construct the data processing of pattern classification with the Bayesian strategy. Figure 2 shows a neural network organization for the classification of input patterns $\bar{x}$ into two categories. In fig. 2, the input units, just like the neurons in the input layer of MLP, merely receive data from the input patterns and supply the same input data to all of the pattern units. Each pattern unit then performs the nonlinear calculation with input pattern vector $\bar{x}$ and interconnection weights X_{Ai} (or X_{Bi}). That is:

$$exp\left[-\frac{(\bar{x}-X_{Ai})^T(\bar{x}-X_{Ai})}{2\sigma^2}\right] \tag{5}$$

And then the summation units simply sum the inputs up from the pattern units corresponding to the same categories, respectively. Finally, the output unit makes a comparison of the values of probability density functions determined by the summation units and produces a binary output depending on the highest probability density.

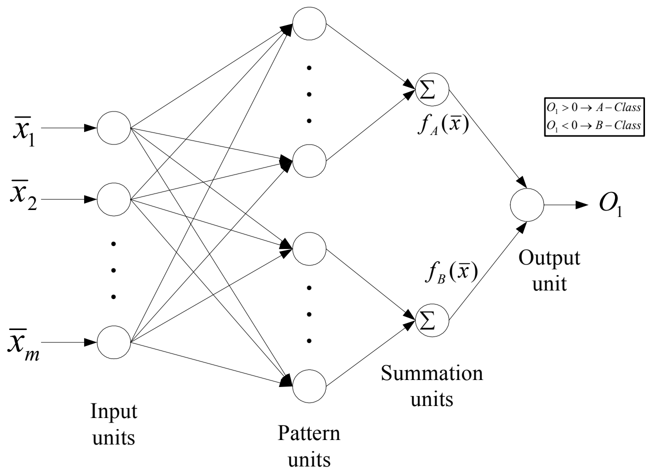

Fig. 2 Probabilistic neutral network architecture

Instead of the application of the back-propagation algorithm and two distinct passes of computation in the MLP, there is only one pass of computation, referred to as the forward pass, in the PNN and no error correction is required. Training in the PNN is simply setting the interconnection weights in one of the pattern units equal to each of the input training patterns X_{Ai} (or X_{Bi}) and then connecting the pattern unit's output to the appropriate summation unit. Training of the PNN can be regarded as searching the optimal value of the smoothing parameter σ since it is the key parameter that dominates the whole decision boundaries. The PNN and the MLP have the similar architectures but the distinct activation functions, key parameters and data processing skills. The major characteristics and differences of the PNN and the MLP were summarized in Table 1.

Table 1 Characteristics and differences of MLP and PNN

Neural network	MLP model	PNN model
Type of network	Multi-layer feedforward network	Multi-layer feedforward network
Architecture	Three-layer structure (Input layer, hidden layer and the output layer)	Four-layer structure (Input units, pattern units, summation units, and output units)
Learning algorithm	Back-propagation algorithm	none
Control parameters	Learning rate, η Momentum factor, α	Smoothing parameter, σ
Activation function	Sigmoid function	Exponential function
Optimal searching skill	Steepest-descent method	Searching in all possible range
Computing time	Long	Very short
Drawbacks	Time consuming Sensitive to a local minima	Large memory required

III. CASE HISTORY

In the present study, the database of field liquefaction cases with the in-situ test measurements (CPT-q_c) was used on the training and testing the neural network models. These field cases were collected and complied from the worldwide field records of the liquefied and non-liquefied sites [11, 14, 15]. A total of 315 CPT-based cases are contained in the database. The cases of the Chi-Chi earthquake in Taiwan are included. The total number of field cases used in the analysis is more than that used in the previous studies. The complete database forms a basis to evaluate the feasibility and the performance on the evaluation of liquefaction potential by using both the PNN model and MLP model.

IV. DATA PROCESSING

Data processing can be simply divided into several procedures, such as selection of the input and output variables, data preprocessing of scaling of data range, construction of the neural network's architecture, preparation of training and testing dataset, and so on. The detailed procedures used in the study are described as follows.

A. Input Variables

Various combinations of the input parameters have been tried to assess liquefaction potential in previous study [6, 8, 9]. Table 2 summarizes the input variables and the number of case records that they concerned and used. Table 2 shows that the empirically calibrated parameters were used as the input variables in the ANN analysis by many of the researchers. In fact, the calibration on the input parameters was not necessitated because the ANN model considered the inner-relationships between the different input parameters inherently. Any calibration or empirical correction of input variables would be automatically taken into account in the ANN model. Hence, in this study the original measurements without any prior-calibration that significantly influenced liquefaction potential were selected as input variables. They are earthquake magnitude (M), peak ground acceleration (a), the total and effective overburden pressures (σ_o, σ'_o), and the cone resistance (CPT-q_c).

Table 2. Various combinations of input variables and number of cases used by previous researchers

Authors (year)	Selected input variables	No. of cases	No. of training cases	No. of testing cases
Goh (1996)	$M, q_{C1}, \tau/\sigma'_o, D_{50}$ $M, q_{C1}, \tau/\sigma'_o, a, \sigma'_o$ $M, q_{C1}, \tau/\sigma'_o, D_{50}, a, \sigma'_o$ $M, q_{C1}, \tau/\sigma'_o, D_{50}, a, \sigma'_o, \sigma_o$ $M, q_C, \sigma'_o, a, D_{50}$	109	74	35
Juang et al. (2000)	$q_{cN}, R_f, \sigma'_o, R_p, SL$	225	163	62
Baziar et al. (2003)	$D_{50}, a, \sigma'_o, q_c, M$ $D_{50}, a, \sigma'_o, q_c, M, \sigma_o$ $D_{50}, a, \sigma'_o, q_c, M, \sigma_o, Z$	170	134	36
	$D_{50}, a, \sigma'_o, q_c, M, FC$	80	70	10
This study	$a, M, \sigma'_o, \sigma_o, q_c$	315	236	79

B. Output Variables

The observation of liquefied or non-liquefied sites is the only output variable in the neural network modeling. The output unit gives a binary value of 1 for liquefied sites and of 0 for non-liquefied sites. During the testing phase, the output in the PNN model was designated as the value of 1 corresponding to the highest probability density function for the liquefaction category and the value of 0 corresponding to the highest probability density function for the non-liquefaction category. However, in the MLP model, the input pattern was considered either liquefied if the final output was larger than 0.5 or non-liquefied if it was less than 0.5 during the testing phase.

C. Data Preprocessing

Before entering the data to the ANN, the preprocessing on the input data are generally required because of the reasons of accuracy and numerical convergence. One of the most useful preprocessing is normalized the primary data into the range of 0-1. Eq. (6) shows the linearly scaling way considered in this study.

$$\bar{x}_i = \frac{x_i - MIN(x_i)}{MAX(x_i) - MIN(x_i)}, \; for \; i = 1 \sim m \qquad (6)$$

where $x_i =$ ith dimension of the input data; $\bar{x}_i =$ ith dimension of the scaled input data; $MAX(x_i) =$ maximum value in the ith dimension of input data; $MIN(x_i) =$ minimum value in the ith dimension of input data; $m =$ dimensionality of input data.

D. Input Data

Input data in the training and testing phase was randomly selected from all the gathered cases in a specific proportion. In this study, 236 cases (about 75% of all the cases) were used as input data in the training phase while the other 79 cases (25% of all the cases) were implemented during the testing phase.

E. Architecture of Network

As described previously as section II, we knew that the architecture of the PNN model can be created completely according to the following factors, i.e., the number of the input variables, the number of the output variables, the number of the cases used in the training phase, and the number of category that needed to be classified. Once the four factors are known, the architecture of the PNN model can be determined easily. However, in the MLP model, the determination of its optimal network architecture would be depending on trial and error. The structures which of one to ten neurons in the hidden layers were used on data processing in MLP model and tried to searching out the optimal number of the hidden layer neurons with the highest performance on recognition. Table 3 shows the numbers of neurons used in the network architectures of PNN and MLP models.

Table 3. The numbers of neurons used in the network architectures of PNN and MLP models.

In PNN model				In MLP model		
No. of input units	No. of pattern units	No. of summation units	No. of output units	No. of neurons in input layer	No. of neurons in hidden layer	No. of neurons in output layer
5	236	2	1	5	1~10	1

F. Analysis and Results

Because of the sensitivity of false local minima in the MLP model analysis and to overcome this drawback, one thousand times of training processing were implemented for each MLP model of different number of hidden layer neurons with randomly initialized connection weights. And then the optimal result was picked up in each “one thousand processing”. Table 4 summarizes those results of different number of hidden layer neurons. It shows that the MLP model with 7 hidden-layer neurons has nearly perfect performance on identification of liquefaction potential of field cases. Only two cases were failed to be recognized in all 315 cases. And more than 99% of overall successful rate of recognition could be obtained by using this MLP model. Nevertheless, this kind of approach that needed perform very large number of trail and error processing was still very time-consuming.

Table 4 No. of successfully recognized cases by using MLP models with different hidden layer neurons

No. of hidden layer neurons	No. of cases recognized successfully		
	In training phase	In testing phase	Overall
1	223	71	294
2	231	73	304
3	235	76	311
4	235	76	311
5	235	77	312
6	235	77	312
7	235	78	313
8	236	77	313
9	236	76	312
10	236	77	313

Meanwhile, in the data analysis of using the PNN model, for giving the highest performance on pattern classification it is very essential to find out the proper smoothing parameter, σ because the value of σ can directly dominate the shape modes of PDF and the PDF is used to form the basis for determination of the decision boundary to classify input patterns correctly. The optimal value of σ can make PNN model to form the optimal decision boundary to separate the liquefied and non-liquefied cases with a maximum rate of recognition both in the training and testing phases. Experience shows that the optimal value of σ could be determined by searching through all the possible value in the range from 0.0005 to 1. Very little computing time was required on searching for the optimal value of smoothing parameter σ .

In this study, the optimal value of the smoothing parameter σ in PNN model was found to be 0.035. By using the PNN model with this optimal smoothing parameter, four cases were failed to be recognized in all 315 cases and the overall successful rate of recognition was 98.7%.

For comparison, both the optimal analysis results by using MLP model and PNN model were listed as table 5(a) and table 5(b). In the training dataset, all the cases could be recognized successfully by using PNN model while there was only one case failed to be identified by MLP model. On the other hand, in the testing dataset, the rate of recognition made by MLP model was better than that predicted by PNN model. There was only one case failed to be recognized by using MLP model in 79 testing cases, while there were four cases misclassified by using PNN model in the same dataset.

The overall rate of recognition made by using MLP model was over 99% and was slightly higher than that performed by using PNN model. Nevertheless, the difference between the overall rate of recognition in the PNN and the MLP is considerably minor. But more searching effort should be made in data analysis of MLP model to overcome the false local minima problem and to find out the optimal results.

Table 5(a). Number of successfully recognized cases by using PNN model and MLP model

No. of cases		No. of cases successfully recognized by PNN		No. of cases successfully recognized by MLP	
Training dataset	Testing dataset	Training dataset	Testing dataset	Training dataset	Testing dataset
236	79	236	75	235	78

Table 5(b). Rate of recognition by using PNN and MLP model

Rate of recognition by using PNN model (%)			Rate of recognition by using MLP model (%)		
Training dataset	Testing dataset	Overall	Training dataset	Testing dataset	Overall
100	94.9	98.7	99.6	98.7	99.4

IV. CONCLUSIONS

The powerfulness and high-performance of using the PNN model and MLP model in the evaluation of the liquefaction potential is presented. Very high rates of recognition on identifying the cases of liquefied and non-liquefied sites can be obtained by both using the PNN model and MLP model after examining the cases experienced during the past earthquakes. Various empirical calibrations on input data are not necessary because the complex inner-relationships between the input variables could be automatically considered both in the PNN model and the MLP model simulation.

Although the overall rate of recognition made by using MLP model was slightly higher than that performed by using PNN model, more searching efforts and computing time should be taken in the training phase of MLP model to overcome the interruption of false local minima and to find out the optimal results.

REFERENCES

[1] C.J.Lee, H.Y.Wen, D.G.Shoung. Investigation on damages of buildings caused by ground liquefaction and their remediation in Yuan-lin area after the Chi-Chi Earthquake, *Journal of Chinese Institute of Civil and Hydraulic Engineering*, vol.15, pp. 851-858, 2003.

[2] T.L.Youd, I.M.Idriss. (Editors). *Proceedings of the NCEER Workshop on Evaluation of Liquefaction Resistance of Soils*, NCEER Technical Report NCEER-97-0022, Buffalo, N.Y.,1997.

[3] K.Tokimatsu, Y.Yoshimi. Empirical correlation of soil liquefaction based on SPT-N value and fines content, *Soils and Foundations*, vol.23, pp. 56-74, 1983.

[4] H.B.Seed, I.M.Idriss, I.Arango. Evaluation of liquefaction potential using field performance data, *Journal of Geotechnical Engineering, ASCE*, vol.109, pp. 458-482, 1983.

[5] R.D.Andrus, K.H.Stokoe. Liquefaction resistance based on shear wave velocity, *Proceedings of the NCEER Workshop*

on Evaluation of Liquefaction Resistance of Soils, pp. 89-128, Buffalo, N.Y.,1997.

[6] C.H.Juang, C.J.Chen, W.H.Tang, D.V.Rosowsky. CPT-based liquefaction analysis, Part 1: Determination of limit state function. *Geotechnique*, vol.50, pp. 583-592, 2000.

[7] A.T.C.Goh. Seismic liquefaction potential assessed by neural networks, *Journal of Geotechnical Engineering, ASCE*, vol.120, pp. 1467-1480, 1994.

[8] A.T.C.Goh. Neural-network modeling of CPT seismic liquefaction data, *Journal of Geotechnical Engineering, ASCE* ,vol.122, pp. 70-73, 1996.

[9] M.H.Baziar, N.Nilipour. Evaluation of liquefaction potential using neural-networks and CPT results, *Soil Dynamics and Earthquake Engineering*, vol.23, pp. 631-636, 2003.

[10] D.F.Specht. Probabilistic neural networks, *Neural Networks*, vol.3, pp. 109-118, 1990.

[11] A.T.C.Goh. Probabilistic neural network for evaluating seismic liquefaction potential, *Canadian Geotehchnical Journal*, vol.39, pp. 219-232, 2002.

[12] S.Haykin. *Neural networks a comprehensive foundation*, Prentice Hall, New Jersey, 1999.

[13] J.H.Hwang, C.W.Yang. Verification of critical cyclic strength curve by Taiwan Chi-Chi Earthquake data, *Soil Dynamics and Earthquake Engineering*, vol.21, pp. 237-257, 2001.

[14] R.W.Boulanger, L.H.Mejia, I.M.Idriss. Liquefaction at Moss Landing during Loma Prieta Earthquake, *Journal of Geotechnical and Geoenvironmental Engineering, ASCE*, vol.123, pp. 453-467, 1997.

[15] D.H.Lee, C.H.Ku. A study on the CPT data at liquefied area, *Journal of Chinese Institute of Civil and Hydraulic Engineering*, vol.13, pp. 779-791, 2001.