Zeitschrift: IABSE reports = Rapports AIPC = IVBH Berichte
Band: 68 (1993)

Artikel: Damage detection using artificial neural system

Autor: Hsu, De-Shiu

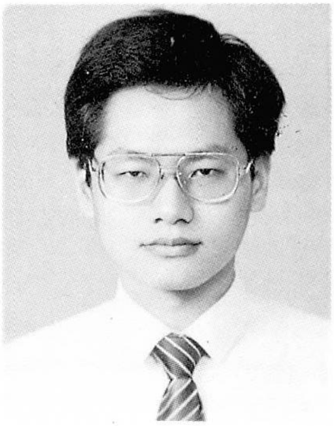
DOI: https://doi.org/10.5169/seals-51862

Damage Detection Using Artificial Neural System ·
Détection des dommages au moyen d'un système neuronal
Auffinden von Schäden mittels künstlichem neuronalen System

De-Shiu HSU
Professor
Nat. Cheng-Kung Univ.
Tainan, China

Deh-Shiu Hsu, born 1947, obtained his Civil Eng. degree at the Nat. Cheng Kung Univ. China, in 1970 and received his Master degree and PhD in 1974 and 1977 at Purdue Univ., USA.

I-Cheng YEH
Assoc. Professor
Chung-Hua Polytechnic Inst.
Hsin Chu, China



I-Cheng Yeh, born 1963, got his PhD at the Nat. Cheng Kung Univ. Tainan, China. His current research interests include expert systems, artificial neural networks, and their application in civil engineering.

SUMMARY
Artificial neural system is for inspecting the stiffness loss of each storey in multi-storey buildings with the data of the eigen value change rates. Each mode eigen value change rate is represented by an input unit and each story stiffness change rate is represented by an output unit. Training examples are randomly generated. The net is trained to achieve the machine learning purpose. Once the net is trained, the stiffness degradation of each story can be deduced by input of the measured eigen values of the damaged building. The learning result with various factors of inspected building and various factors of training examples are compared. The results show that the error rates of inspection of stiffness loss are dependent on these factors. Therefore, selection of these factors is important for minimizing the error rate.

RÉSUMÉ
Un système neuronal artificiel permet de déterminer la réduction de la rigidité de chaque étage de bâtiments à étages multiples, par suite du taux de changement de la fréquence propre. Le taux de changement de fréquence de chaque valeur propre correspond à une unité d'entrée du réseau, tandis que le taux de changement de rigidité de chaque étage équivaut à une unité de sortie. L'entraînement résulte d'exemples générés de manière aléatoire. Après avoir entré les valeurs propres mesurées sur le bâtiment endommagé, le réseau est apte à fournir la modification de rigidité. La comparaison des résultats d'apprentissage est faite à partir de divers facteurs des bâtiments inspectés et d'exemples résultant de l'entraînement. Les taux d'erreur du changement de rigidité sont fonction de ces facteurs, dont la sélection joue un rôle important pour minimiser le taux d'erreur.

ZUSAMMENFASSUNG
Es wird gezeigt, wie mit einem künstlichen neuronalen System in mehrgeschossigen Bauten der Steifigkeitsabfall eines jeden Stockwerks durch die Rate der Eigenfrequenzänderung bestimmt werden kann. Bei diesem Vorgehen entspricht die Frequenzänderungsrate eines jeden Eigenwerts einer Inputeinheit des Netzes und jede Steifigkeitsänderungsrate einer Outputeinheit. Das Trainieren des Netzes erfolgt durch zufällig generierte Beispiele. Die Lernergebnisse werden mit verschiedenen Faktoren inspizierter Gebäude und von Trainingsbeispielen verglichen. Die Fehlerrate hängt von diesen Faktoren ab. Ihrer Auswahl kommt grosse Bedeutung zu.

1. INTRODUCTION

Both of damage detection and damage assessment of existing structures become essential topics in structural safety. Many of papers [1-5] dealing with the damage assessment of existing structures by various methods such as energy absorbing potential by histeristic response loop of structure, seismic behavior affection, static or dynamic testing to identify the strength of the structures ...etc. End up to a quantified value called damage index or safety index. According to this index, more or less, information is provided to assist people to make some kind of decision such as the structural should be rebuilt, should be repaired or the structure can still be used as usual. However, once a structure has been assessed to be repaired. What should we do? First of all, detect the weak point, or the damaging region. Only after the damage point can be found, then the repairing process can be performed. Otherwise nothing we can do with the damaged structure.

How can the weak points be detected? People try to solve this problem by several points of aspects. For example, check the crack position by visualizing technique, detect the length and depth of the crack and assess the damage of structure subjectively; perform the dynamic tests to the existing structures and then calculate the stiffness of the structure globaly and locally through the structural identification techniques. Because of the issue dealing with highly nonlinear problems, it is very difficult to find the one to one function relationship between the structural behavior and the testing output such as the natural frequencies of the structure. Turns out the issue of the structural safety assessment usually become the collection of some subjective opinions. Where damage location is? still left unsolved.

In this decade, expert system become more popular to solve problems with complex interrelations between factors. Comments collected from experts are transfered to quantified index provide us relatively objective solution of the problems. When dealing with the problems such as damage assessment of existing structures, one of the primary obstacle is the set up of the knowledge base. Data acquisition or knowledge collection usually is an uneasy task. In order to overcome the bottle neck of knowledge acquisition, an artificial neural network (ANN) is employed as the learning mechanism to transfer engineering experience into understandable knowledge. In this paper, the effect of diagnosing damaged structures by artificial neural network is described.

For an existing structure, have been constructed and used for years, more or less damage would be existed to a certain level. Damage usually caused by some of abnormed forces such as earthquake, strong wind load ...etc. Dynamic tests provide data such as the natural frequencies in various vibration modes. An expert system software is employed to find the stiffness of the structure. However, whether the performance efficiency is high or not of the software rely on whether knowledge base is sound or not. To set up a strong knowledge base become an essential task.

Artificial neural system is a teachable system that consists of many simple units in a highly inter-connected network. Information is stored in the strength of the connections between units. ANS is modeled to simulate the gross structure of the brain: a collection of nerve cells, or neurons, where each of them is connected to as many as 10,000 others from which it receives stimuli-inputs and feedback, and to which it sends stimuli. The most famous artificial neural system model includes Back Propagation, Hopfield Net, Boltzman Machine, and so on [6-10].

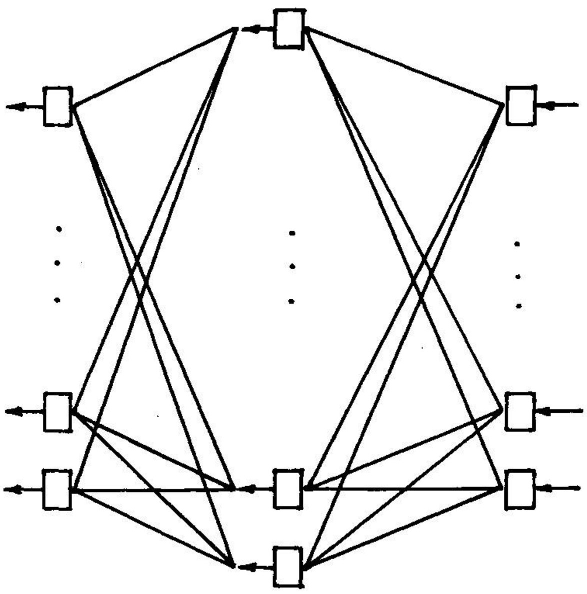
This paper describes an effort that applies ANS to inspect the location and degree of damage of the building. In this paper. Section 2 introduces artificial neural system model and the back propagation learning algorithm. Section 3 describes the method that employs artificial neural system to inspect the stiffness degradation of each story of a multistory shear building. In Section 4, several numerical examples are employed to illustrate and several remarks are concluded. Finally, Section 5 gives conclusion.

2. ARTIFICIAL NEURAL SYSTEM

An ANS is basically a system that uses simple processing units connected in a highly parallel manner. Some terminologies are introduced as follows:

Processing unit: Processing unit is an artificial neuron in a neural network. The output from one processing unit is fanned out and becomes the input to many other units.

Layered neural network: A layered neural network consists of several distinct layers of neurons (refer to Fig 1), including an input layer that employs sensor neurons to monitor external signals and an output layer that transmits signals to the external world. In addition, a layered network may contain one or more hidden layers. Hidden units are necessary to represent interaction of units and internal structures of the domain.
Connections: Connection is a signal transmission pathway between processing units. Each connection has a numerical weight that roughly corresponds to the influence between units. In general, all units in a layer are fully interconnected to the units in adjacent layers. Information flows unidirectionally from input layer through hidden layers to output layer. However, it flows in the reverse direction during learning.
Transfer function: Transfer function is a mathematical formula that determines a processing unit's output value as a function of the input signals and weights.
Learning algorithm: Learning algorithm is an algorithm that modifies the weights in connections according to the information it has learned.



$$U_j = \frac{1}{1+\exp(-\sum_k W_{kj}H_k)}$$

output layer units

links with weights W_{kj}

$$H_k = \frac{1}{1+\exp(-\sum_i V_{ik}S_i)}$$

hidden layer units

links with weights V_{ik}

input layer units

input value

Fig.1 A three-layer neural network.

In this paper, the type of network is a layered neural network using back-propagation algorithm as described by Rumelhart, Hiton and Williams [10]. The neuron cell receives a net signal of the linear weighted sum of all its inputs. A transfer function $1/(1+e^{-x})$ converts the net signal to output. The output, H_k, of the kth hidden unit is given by

$$H_k = \frac{1}{1+\exp(-\sum_i V_{ik}S_i)} \quad \text{.......................................(1)}$$

Where (a) V_{ik} is the weight on the link between the ith input unit and the kth hidden unit.
(b) S_i is the input value of the ith input unit.
Similarly, the output, U_j, of the jth output unit is given by

$$U_j = \frac{1}{1+\exp(-\sum_k W_{kj}H_k)} \quad \text{.......................................(2)}$$

where W_{kj} is the weight on the link between the kth hidden unit and the jth output unit.

The network learns by comparing its output of each input pattern with a target output T_j of that pattern, then calculating the error and propagating an error function backward through the net. The error function is defined as

$$E = \frac{1}{2}\sum_j (T_j - U_j)^2 \qquad (3)$$

Error function is the function of weights. For minimizing the error function, the gradient steepest descent method (Rumelhart et al. 1986) is employed. For a weight W_{kj}, the partial derivative of the error function is

$$\frac{\partial E}{\partial W_{kj}} = -U_j \cdot (1 - U_j) \cdot (T_j - U_j) \cdot H_k \qquad (4)$$

Then, the weight W_{kj} is modified by the incremental amount according to

$$\triangle W_{kj} = \eta \cdot (-\frac{\partial E}{\partial W_{kj}}) \qquad (5)$$

where η is called "learning rate" that gives the step size to minimize the error function. Let d_j be defined as

$$d_j = U_j \cdot (1 - U_j) \cdot (T_j - U_j) \qquad (6)$$

Then, the increment amount can be written as follows

$$\triangle W_{kj} = \eta \cdot d_j \cdot H_k \qquad (7)$$

In addition to the increment indicated in Eq. (7), a momentum term is often included in the formula to make the learning process more efficient. Consequently, the nth increment of weight W_{kj}, $\triangle W_{kj}^n$, is

$$\triangle W_{kj}^n = \eta \cdot d_j \cdot H_k + \alpha \cdot \triangle W_{kj}^{n-1} \qquad (8)$$

where α is the momentum factor.

For a hidden unit, d_k^* is defined as

$$d_k^* = H_k \cdot (1 - H_k) \cdot \sum_j (d_j \cdot W_{kj}) \qquad (9)$$

and the nth increament of the weight V_{ik}, $\triangle V_{ik}^n$, is

$$\triangle V_{ik}^n = \eta \cdot d_k^* \cdot S_i + \alpha \cdot \triangle V_{ik}^{n-1} \qquad (10)$$

According to Eq. (8) and Eq. (10), the strength of connections of newtwork would be modified iteratively to achieve convergence.

3. STRUCTURAL DAMAGE DETECTION

Artificial neural system is proposed to solve the problem of inspection of the stiffness loss of multistory shear building, i.e., to identify the damage location of the existing buildings. the central idea is as follows:

(1) Represent the eigen value chang rates of each vibration mode of the building by an input unit of ANS; represent the stiffness change rates of each story by an output unit of ANS.

(2) Generate training examples with simulation. Each example consists of a set of vibration mode eigen value change rates and a set of story stiffness change rates.

(3) Implement ANS learning to train the net by training examples.

(4) Measure the change rates of eigen values with instrument in real world, and input these value into the trained net, and then the change rates of story stiffness of the building can be deduced from the output of the net.

There are four phases for building an ANS mode, including:

(1) Identification phase: identifying the input and output of the net.

(2) Collection phase: collecting examples for training and testing.

(3) Implementation phase: implementing ANS learning to train the net by training examples.

(4) Verification phase: verifying the trained net by testing examples.

The detail procedure for using ANS to inspect structural damage is described as follows:

(1) Identification phase:
The definition of input of the ANS is defined as follows:

$$S_i = \frac{\lambda_i - \lambda_i'}{\lambda_i} \qquad (11)$$

where S_i = the actual input of the ith input unit.
λ_i = the ith mode eigen value of the original building.
λ_i' = the ith mode eigen value of the existing building.
The definition of output of the ANS is defined as follows:

$$T_j = \frac{R_j}{D_{max}} \qquad (12)$$

$$R_j = \frac{k_j - k_j'}{k_j} \qquad (13)$$

where T_j = the actual output of the jth output unit.
R_j = the change rate of the jth story stiffness.
k_j = the ith story stiffness of the original building.
k_j' = the ith story stiffness of the existing building.
D_{amx} = the maximum stiffness loss rate.
For example, $T_j = 1$, denotes $k_j' = (1 - D_{max}) \cdot k_j$
$T_j = 0$, denotes $k_j' = k_j$

(2) Collection phase:
Training examples are generated with the following simulation procedure:
Set the of damaged story of the building, Prob $(0 < Prob < 1)$
Set the maximum stiffness loss rate, $D_{max}(0 < D_{max} < 10$
Set the number of story of the building, N_{floor}
Set the number of desired examples, N_{exam}
Set the number of desired mode, N_{mode}
Let $I = 0$
Repeat until $I = N_{exam}$
  Let $I = I + 1$
  Let $J = 0$
  Repeat until $J = N_{floof}$
    Let $J = J + 1$
    Generate a uniform random number, Random, in [0,1]
      if Random > Prob
      then $k_j' = k_j$
      else $k_j' = k_j \cdot (1 - D_{max} \cdot Random/Prob)$
  Analyze the N_{mode} lowest mode eigen values, and calculate the input and output of ANS by Eq. (11) and Eq. (12).
Testing examples can be obtained from the same procedure.

(3) Implementation phase:
After collection phase, ANS learning can be employed to train the net by training examples. In this phase several parameters need to be assigned, including (1) learning rate; (2) momentum factor; (3) number of hidden layers; (4) number of hidden units. The initial net weights can be assigned with uniformly distributed random values as follows

initial net weight $= C_{weight} \cdot Random$ .... (14)

where C_{weight} = the multiplier factor of initial net weight,
Random = a uniform random number in the $[-1, +1]$ interval.

(4) Verification phase:
For evaluating the learning results, the error rate is defined as follows:

$$\text{error rate} = \sqrt{\frac{\sum (T_j - U_j)^2}{N_{out}}}, \text{ where } j = 1, 2, \cdots, N_{out} \quad (15)$$

where N_{out} is the number of output units.

4. NUMERICAL EXAMPLES

A ten-story shear building shown in fig. 2 is analyzed, and several factors are set as follows:

A. factors of inspected building
(1) the rate of damaged story of the building, Prob =0.4
(2) the maximum stiffness loss rate, $D_{max} = 0.4$

B. Factors of training examples:
(1) the number of training examples, $N_{exam} = 500$
(2) the number of desired mode, $N_{mode} = 10$

C. Factors of neural network
(1) the learning rate, $\eta = 1.0$
(2) the momentum factor, $\alpha = 0.5$
(3) the number of hidden layers, $N_{layer} = 1$
(4) the number of hidden units, $n_{hidden} = 10$

The results of the implementation of the problem are shown in Fig. 3. It is found that the error rate is converged while training example set (i.e. 500 training examples) is implemented about 50 cycles. The results of several testing examples are shown in Table 1. The procedure takes about 5 minutes on a 80386 PC with math coprocessor.

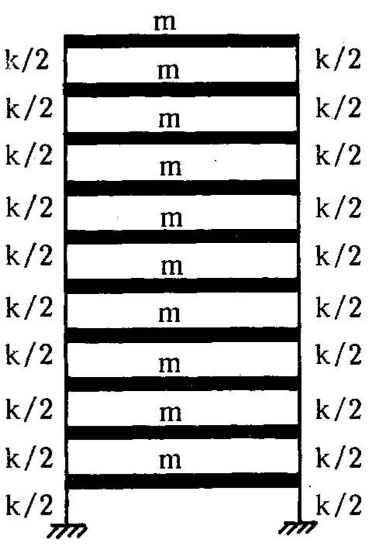


Fig.2 Ten-story shear building model of numerical example. ($m = 1.0 \cdot 10^5 kg$, $k = 2.0 \cdot 10^8 Nt/M$)

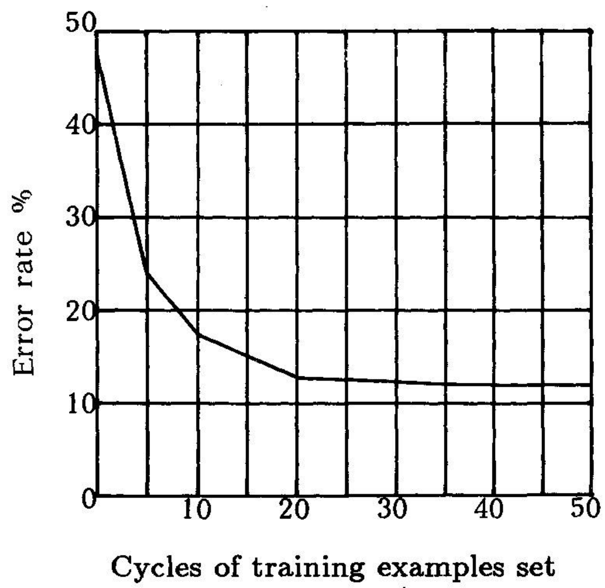


Fig.3 Learning results for cycles of training examples.

TABLE 1. The results of several testing examples.

Test Exam	Output unit									
No.	1	2	3	4	5	6	7	8	9	10
1	0.29	0.00	0.97	0.00	0.28	0.38	0.47	0.34	0.00	0.00
	0.12	0.00	0.93	0.08	0.29	0.60	0.48	0.47	0.02	0.02
2	0.00	0.77	0.65	0.69	0.30	0.99	0.00	0.30	0.00	0.00
	0.01	0.71	0.95	0.14	0.16	0.50	0.25	0.67	0.07	0.07
3	0.00	0.78	0.84	0.00	0.56	0.00	0.60	0.98	0.00	0.62
	0.01	0.47	0.93	0.02	0.82	0.11	0.75	0.95	0.05	0.48
4	0.00	0.96	0.81	0.61	0.00	0.00	0.00	0.23	0.00	0.28
	0.08	0.76	0.82	0.61	0.32	0.02	0.00	0.12	0.18	0.37
5	0.00	0.00	0.11	0.24	0.38	0.00	0.00	0.00	0.28	0.00
	0.01	0.02	0.05	0.14	0.56	0.01	0.00	0.03	0.26	0.01
6	0.30	0.00	0.00	0.00	0.03	0.09	0.00	0.00	0.00	0.00
	0.15	0.01	0.02	0.00	0.03	0.01	0.00	0.01	0.01	0.01
7	0.00	0.16	0.00	0.58	0.07	0.94	0.00	0.00	0.38	0.00
	0.02	0.10	0.08	0.52	0.03	0.97	0.02	0.08	0.27	0.09
8	0.00	0.00	0.00	0.00	0.08	0.95	0.00	0.00	0.00	0.23
	0.15	0.02	0.12	0.00	0.00	0.96	0.02	0.01	0.04	0.33
9	0.18	0.94	0.58	0.00	0.00	0.48	0.47	0.00	0.00	0.00
	0.09	0.81	0.26	0.04	0.27	0.85	0.03	0.01	0.13	0.07
10	0.54	0.00	0.00	0.95	0.00	0.00	0.59	0.01	0.00	0.00
	0.39	0.01	0.04	0.90	0.07	0.01	0.46	0.07	0.02	0.06

Note: The upper values are the actual output of the output unit.
The lower values are the inference output value of the output unit.

For the reason of comparison, the above mentioned implementation is reimplemented by the cases applying various factors. When each of the special cases with the change of certian mentioned factor is implemented, all of the other factors are maintained as the value set in above paragraph.

A. Factors of inspected building

(1) The rate of damaged story of the building, prob = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9. The results of the implementation are shown in Fig. 4.

(2) The maximum stiffness loss rate, D_{max} = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9. The results are shown in Fig. 5.

B. Factors of training examples:

(1) The number of training examples, N_{exam} =50, 100, 250, 500, 750, 1000, 1500, 2000. The results of the implementation are shown in Fig. 6.

(2) The number of desired mode, N_{mode} = 3, 4, 5, 6, 7, 8, 9, 10. The results are shown in Fig. 7.

From the above example, some observations can be given as follows

A. Factors of inspected building

(1)The rate of damaged story of the building

The results show that there is a increasing monotonical relationship between the rate of damage story of the building and the error rate.

(2) The maximum stiffness loss rate

The results show that there is a increasing monotonical relationship between the maximum stiffness loss rate of the building and the error rate.

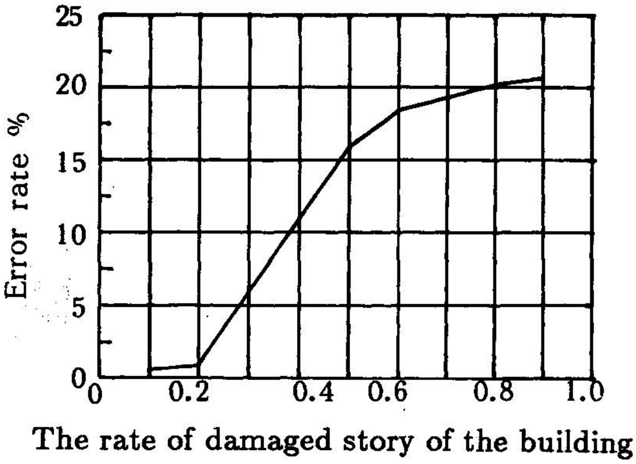


Fig.4 Comparison of the learning results with the rate of damaged story of the building.

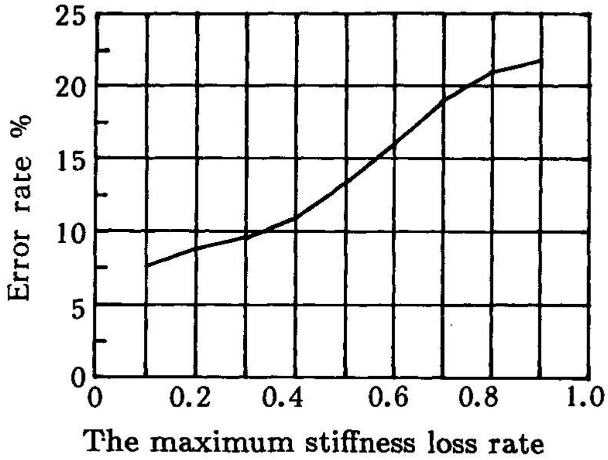


Fig.5 Comparison of the learning results with the maximum stiffness loss rate.

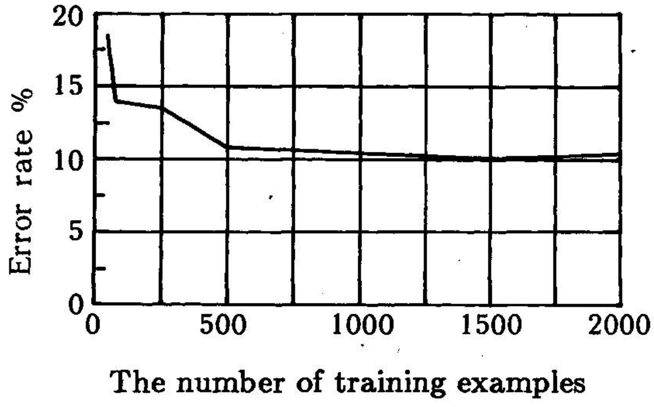


Fig.6 Comparison of the learning results with the number of training examples.

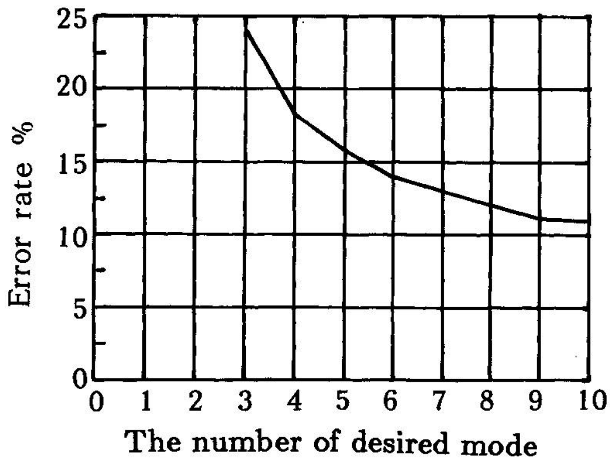


Fig.7 comparison of the learning results with the number of desired mode.

B. Factors of training examples:

(1) The number of training examples

The results show that there is a decreasing monotonical relationship between the number of story of the building and the error rate.

(2) the number of desired mode

The results show that there is a decreasing monotonical relationship between the number of story of the building and the error rate.

5. CONCLUSION

In this paper, artificial neural system is employed to inspect the stiffness loss and the damage location of multistory shear building. Results obtained with various factors of inspected building and various factors of training examples are compared. The results show that the error rate of inspection of stiffness loss and the damage location is dependent on these factors. Therefore, selection of the factors of training examples is an important task in order to minimizing the error rate. However, it can be concluded that an ANS can be sufficiently worked as a tool for damage detection.

Appendix I. Reference

1. Cassaro, M. A. and Romero, E. M. (eds.) (1986). The Mexico Earthquakes-1985, Factors Involved and Lessons Learn, ASCE, New York, N.Y.
2. Stone, W. C. (1986), "Preliminary dynamic analyses of the ministry of agriculture building," in The Mexico Earthquakes-1985, Factors Involved and Lessons Learn, 233-254, Cassaro, M. A. and Romero, E. M. (eds.), ASCE, New York, N.Y.
3. Yao, J. T. P. (1980). "Damage assessment of existing structures," J. of Engineering Mechanics, ASCE, 106(4), 785-799.
4. Ogawa, H., Fu, K. S., and Yao, J.T.P. (1985). "An inexact inference for damage assessment of existing structures," Int. J. Man-Machine Studies, 22, 295-306.
5. Wang, T. Y., Bertero, V. V. and Popov, E. P. (1975) "Hysteretic behavior of reinforced concrete framed walls", Report No. EERC 75-23, Earthquake Engineering Research Center, University of California, Berkeley, California.
6. Hecht-Nielsen, R. (1988). "Neurocomputing: picking the human brain", IEEE Spectrum, March 1988, 36-42.
7. Maren, A. E. (1989), "The need-to-know of neural networks", J. of Neural Network Computing, Summer, 1989, 57-65.
8. McAulay, A. D. (1987). "Engineering design neural networks using split inversion learning", in Proceedings of the First Int. Conf. on Neural Networks, Vol. IV, 635-641, M. Caudhill and C. Butler (eds.), IEEE, San Diego, California, June 21-24, 1987.
9. Rehak D. R., Thewalt, C. R., and Doo, L. B. (1989). "neural network approaches in structural mechanics computations", in Computer Utilization in Structural Engineering, Proceedings of the Sessions at Structures Congress '89, 168-176.
10. Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). "Learning internal representation by error propagation", in Parallel Distributed Processing, Vol. 1, 318-362, Rumelhart, D. E. and McClelland, J. L. (eds.), M.I.T. Press, Cambrideg, MA.

R_j = change rate of the jth story stiffness.
S_i = actual input of the ith input unit.
H_k = output value of the kth hidden unit.
U_j = inference output value of the jth output unit.
T_j = actual output of the jth output unit.
V_{ik} = weight on the link between the ith input unit and the kth hidden unit.
W_{kj} = weight on the link between the kth hidden unit and the jth output unit.
η = learning rate
α = momentum factor
λ_i = ith mode eigen value of the original building
λ_i' = ith mode eigen value of the existing building