大韓土木學會論文集
第17卷 第I-4號 · 1997年 7月
pp. 617 ~ 626

構造工學

신경망을 이용한 구조물의 손상평가

Damage Assessments of Structures Using Neural Networks

정환진* · 김주태 · 오주원*** · 이인원******
Choung, Hwan Chin · Kim, Ju Tae · Oh, Ju Won · Lee, In Won

요 지

시간 영역에서 신경망을 이용한 구조물의 손상평가를 수행하였다. 손상평가에는 역전파 네트웍이 사용되었다. 구조물에 손상이 발생하면 구조물의 동적응답이 변하고 변화된 동적응답은 구조물의 손상위치와 정도에 관한 정보를 포함하게 된다. 동적응답의 변화를 통해서 신경망은 구조물의 손상위치와 정도를 동시에 감지한다. 신경망을 이용한 손상평가에 관한 기존 연구들에서는 정적하중과 지진하중에 의해 동적응답을 얻었다. 그런 경우 실질적으로 응답을 얻기가 불가능하다. 그러나 본 연구에서 사용된 보 구조물의 동적응답은 이동하중에 의하여 유발되었다. 신경망의 입력은 가속도이고, 이 가속도에 의해 신경망이 학습을 한다. 수치해석결과 손상평가에 신경망을 이용하는 것이 매우 효율적이라는 사실과 신경망의 입력값을 얻기 위한 샘플링 시간에 대한 결론을 얻었다.

핵심용어 : 손상평가, 신경망, 학습, 이동하중

Abstract

In this paper, damage assessments are accomplished by using neural networks in time domain. Backpropagation neural network is adopted to fulfill the damage assessments. When the damages occur on a structure, the structural responses changed contain the information about damage locations and extents, implicitly. Throughout the changes of structural responses, the neural networks can detect damage locations and extents, simultaneously. In the conventional damage assessments using neural networks, the vibration signatures are measured by means of impacting the seismic loads and static loads on structures. In such cases, it is difficult to obtain the data surveying on the structures, practically. However, the vibration of beam structure considered in this study is induced by the moving load. The acceleration is the input of neural networks. The neural network is trained and tested with the acceleration data. The simulation informs that it is efficient to use neural network in damage assessments and how

*정회원 · 한국과학기술원 토목공학과 석사과정
**정회원 · 한국과학기술원 토목공학과 박사과정
***정회원 · 한남대학교 토목환경공학과 교수
****정회원 · 한국과학기술원 토목공학과 교수

to determine the sampling time of input data for neural networks.

Keywords : *damage assessment, neural network, learning, moving loads*

1. Introduction

Damage assessments are defined as finding the damage locations and extents of structures. It is very important to detect the damage locations and extents so that the structural safety and serviceability may be maintained. In-built civil structures are damaged by various loads.

To fulfill the damage assessments directly, the method to observe with the eyes and various in-situ tests exist. Recently, the indirect methods to assess the damages have been developed by many researchers.[6] The nondestructive evaluation of structures have been a promising field. The nondestructive evaluation of structures can be classified as frequency-domain based or time-domain based. Before 1990s, the natural frequency and mode shape[7] are the general tools for damage assessments. Because the dynamic characteristics are very sensitive to the mass and stiffness changes, the data on dynamic characteristics are useful for detecting the local damages of structures. However, the frequency-based researches using the natural frequencies and mode shapes are mainly dependent on a few lower modes and needs the mathematical models of structures. In addition the behavior of individual structural elements are not evaluated. The damage assessments in the time domain can overcome these drawbacks.

As a pilot study, Ghaboussi's paper[11] has covered the damage assessments of frame structures using neural networks. The backpropagation neural network was trained and tested with the acceleration of structures in the frequency domain for damage assessments. The response of structures contains the information regarding its natural frequencies and mode shapes. The damage of structure affects these dynamic characteristics. The response measured on the structure, compared with the undamaged structure, has different response and the response contains the information regarding damage locations and extents. Based on this idea, the damage assessments of simple three-story frame was accomplished. The possibility of damage assessments using neural network was presented in the Ghaboussi's paper.

In 1992, Hajela[12] studied the damage assessments of a three-story frame using a counter-propagation neural network. He used ideal material properties of the structure. As the input to neural network, the displacement was used and in fulfilling the damage assessments, the noise and incomplete data were introduced.

Barai[9] investigated the study of truss bridges. The type of neural network adopted was the backpropagation neural network with two hidden layers. The characteristics of this paper is that the damage is defined as the area reduced in member. He presented that the performance of the network trained with single-point vibration signature at a suitable location was found to be reasonably good compared with that of three-point and five-point vibration signature measurements.

In this paper, the damage assessments which are easily applicable to the fields are fulfilled by using the material properties of real structures and accelerations induced by a moving load. Throughout this paper, the criterion on the number of data needed in training neural networks is presented.

The neural network is explained and the proposed method and the numerical example are presented.

2. Neural Networks

The fields that neural networks are applied are diverse: Signal Processing, Pattern Recognition, Control, Medicine and Damage Assessment etc. With the learning capability and nonlinear characteristics, artificial neural networks (ANNs) can solve many problems considered unsolvable in those fields. The characteristics of neural networks include developing the performance of networks throughout the learning and that networks may learn with the incomplete training data. ANNs are information-processing and parallel-processing system that performs certain function.

2.1 Neuron

An artificial neural network is modeled from human brains. Fig. 1 shows that a neuron is composed of dendrite, soma and axon. The dendrites receive signals from previous neurons. A signal is an electrical impact transmitting across the synaptic gap. The soma processes the signal received. The signal processed in a neuron is distributed to other neurons through the axon.

McCulloch and Pitts[3] devised a simple neuron model shown in Fig. 2. In Fig. 2, X, W, F, θ indicate neuron input, weight strength

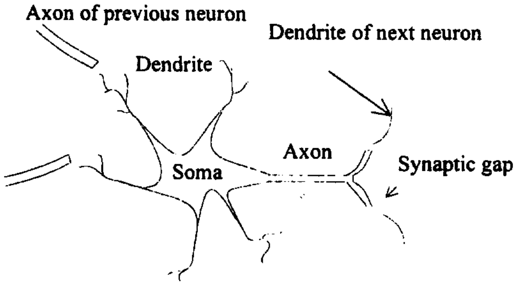


Fig. 1. Biological neuron.

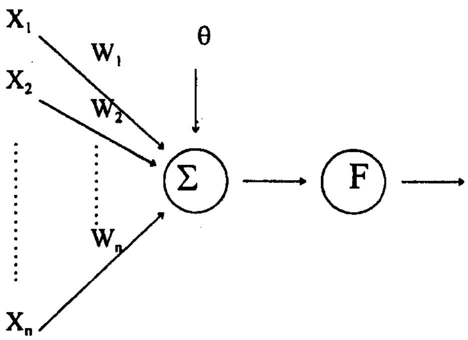


Fig. 2. McCulloch-Pitts model.

between neurons, activation function and bias, respectively. The symbol Σ means the process of inputs entering networks.

2.2 Backpropagation Network

The types of neural networks are Hopfield network, Kohonen network, Boltzmann machine, Backpropagation network, etc. The primary function of Hopfield network is to retrieve a pattern in memory in response to the presentation of an incomplete or noisy version of that pattern. Boltzmann machine is used with a binary input and output. Kohonen network is used for performing data compression. Backpropagation network is capable of performing any linear or nonlinear computation, and can approximate any reasonable value or function arbitrarily well.

Fig. 3 is the structure of Backpropagation

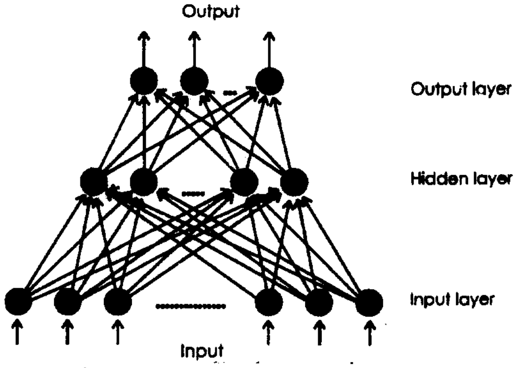


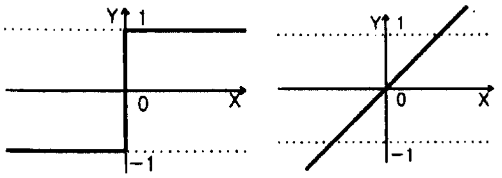
Fig. 3. Structure of backpropagation.

network. The black circles are neurons. Backpropagation network in Fig. 3 is composed of the input layer, hidden layer and output layer. The arrows indicate the flow of information. The training information propagates backward. While the training information propagates backward, the neural networks learn the characteristics of training set. Therefore, this network is called Backpropagation network. The learning of neural networks is simulated by the changes in weights and thresholds.

The neurons process input data and distribute their output to other neurons. The communication of information in the same layer does not happen in the backpropagation neural network. Fig. 3 is the backpropagation network with a hidden layer. If networks have more hidden layers and neurons, learning time is increased.

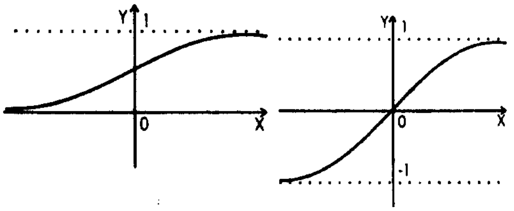
2.2.1 Activation Function

Fig. 4 represents simple activation function of neural network. Fig. 5 represents the activation function of Backpropagation networks. According to the outputs of neural network, activation function is chosen adequately. Activation represents that neurons receive and process input signal. Neurons in the input layer receive the signal and then transmit the signal to the hidden layer. In the hidden layer, the received signals are summed and then applied to the activation function. Same process occurs in the hidden layer.



$$y = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases} \qquad y = x$$

Fig. 4. Simple activation function.



$$y = 1/(1+\exp(-x)) \qquad y = 2/(1+\exp(-x)) - 1$$

Fig. 5. Activation function of backpropagation networks.

2.2.2 Learning Algorithm

Step 1. Initialize weights randomly.

Step 2. The signal from input layer is distributed to the hidden layer.

Step 3. Each hidden neuron sums the weighted input signal.

$$h_j = b_{0j} + \sum_i x_j w_{ij} \tag{1}$$

where, x_j : output in the j-th neuron
w_{ij}: weight between i-th layer and j-th layer
b_{0j} : bias in the j-th layer

Equation (1) is applied to activation function to compute the output:

$$z_j = f(h_j) \tag{2}$$

where, f: activation function

Step 4. Each output neuron sums the weighted input signal.

$$s_k = b_{0k} + \sum z_j w_{jk} \tag{3}$$

Equation (3) is applied to the activation function to compute the output.

$$y_k = f(s_k) \tag{4}$$

Step 5. The each output neuron computes its error information, the rate of change of weights and bias.

$$\delta_k = (t_k - y_k) f'(y_k) \tag{5}$$

$$\Delta w_{jk} = \alpha \delta_k z_j \tag{6}$$

$$\Delta b_{ok} = \alpha \delta_k \tag{7}$$

where, α: learning rate

Step 6. Each hidden neuron sums the weighted error.

$$\delta_{jk} = \sum_{k=1}^{m} \delta_k w_{jk} \tag{8}$$

Each hidden neuron computes its error information, the rate of change of weights and bias:

$$\delta_j = \delta_{jk} f'(z_j) \tag{9}$$

$$\Delta w_{ij} = \alpha \delta_j x_i \tag{10}$$

$$\Delta b_{0j} = \alpha \delta_j \tag{11}$$

Step 7. Update weights and Bias:

$$w_{jk}^{new} = w_{jk}^{old} + \Delta w_{jk} \tag{12}$$

$$b_{0j}^{new} = b_{0j}^{old} + \Delta b_{0j} \tag{13}$$

Step 8. Check the stopping conditions.

Step 9. Until stopping conditions are satisfied, the algorithm is repeated from Step 2 to Step 8.

The flow of input data is forward (from step 1 to step 4), and the direction of training is backward (from step 5 to step 7).

2.2.3 Momentum

Momentum can decrease learning time and elevate the efficiency of learning. It is the role of momentum that stabilizes networks and determines the learning speed.

$$w_{jk}(t+1) = w_{jk}(1) + \Delta w_{jk}(t+1)$$

$$\Delta w_{jk}(t+1) = \alpha \delta_k z_j + \mu \Delta w_{jk}, \quad 0 < \mu < 1$$

where, μ: momentum constant.

If momentum constant lies between 0 and 1, learning time is decreased and the efficiency of learning is elevated. However, if momentum constant lies in other zones, the error is increased.

3. Neural Network for Damage Assessments

To assess the damage, many methods have been investigated and developed by many researchers. In recent years, neural networks have been introduced to damage assessments and rigorous research continues in this area.

In general, when a structure has damages, the structural responses will be changed. Though damages may be in the invisible areas, the response of structures will be changed. The response of damaged structures will differ according to damage locations and extents. Throughout the change of structural responses, the damage locations and extents will be detected by the neural network. The damage assessments using neural networks are to find damage locations and extents. Moreover, neural networks do not need the mathematical model of structures and regardless of the linearity or nonlinearity of structures, the neural networks can be applied to the damage assessments of structures.

The conventional damage assessments using neural networks made use of ideal material properties, seismic loadings, static loadings, etc. In these cases, it is not easy to measure the responses of the structure. The structural responses induced by the moving load, however, are easy to obtain.

The damage assessments using neural networks are composed of three processes: preparation of training set, training neural network and damage assessments using trained neural network.

3.1 Preparation of Training Set

The acceleration of structures is measured in time domain. The characteristics of structures are better described by acceleration than by displacement or velocity of structures. The measurement points of structures are adopted as the locations to consider several modes. The meaning of sampling rate is the time interval to measure acceleration, and affects the efficiency of damage assessments. If sampling rate is too slow or too fast, the damage is not efficiently assessed.

It is impossible that researcher imposes damages to the real structures for the field experiments and then measures the structural responses. However, if researcher knows the material properties of real structures, it is possible that he measures the responses of the damaged structures. The data measuring on one point or more can be utilized. The number of data, also, can be varied. In this paper, one point is utilized, and the number of data is 200, 400, 600 and 800.

3.2 Training Neural Network

The backpropagation neural network is adopted for damage assessments. The structure of backpropagation network is simple and learning algorithms are highly developed. The backpropagation neural network is composed of one input layer, one hidden layer and one output layer. Through the hidden layer, the complicated computations are performed. The initialization of weights is between -1 and 1. As the outputs of neural network lies between 0 and 1, the activation function like the one in Fig. 7 is chosen. The outputs in the output layer represent the residual stiffness of each element. The neural network generates the output as the number between 0 and 1. The 0 represents the complete loss of stiffness in one section of structures, and 1 represents no damage. The results are the data obtained from structural analysis, and the causes are damage locations and extents. In training the neural networks, researcher knows the causes and results.

3.3 Damage Assessments with Neural Network

The testing of neural networks uses not only the trained data but also the untrained data. The task of finding damage locations and extents may be failed. The cause of this problem is that networks dip into local minimum. Therefore the optimal structure of neural networks may be determined by trial and error. Neural network is used as in Fig. 6 when trained, as in Fig. 7 when tested.

3.4 Numerical Example

In this paper, the neural network is trained with the dynamic responses of structure and the network assesses the damages in structure. The input data is originated from the structural analysis program ADINA. The bridge in Kyung-bu high speed train is modeled as the Bernoulli-Euler beam. The real bridge has complicated shapes, but the model for simulation is idealized as a beam with equivalent stiffness. The damage is defined as reduction in flexural stiffness.

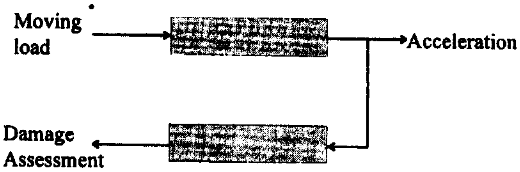
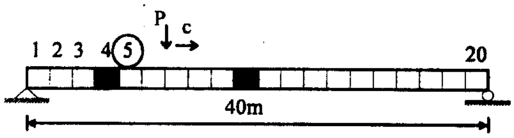


Fig. 6. Block diagram of training.

Fig. 7. Block diagram of testing.

Fig. 8 represents the mid-span finite element model of the bridge in Kyung-bu high speed train with span length of 40m. The model consists of 20 beam elements, in which element 4 or 10 (hatched box) are assumed to have damages. Because large damages are easily detected with the eyes, the ranges of assumed damages are limited to 0~40% reduction of undamaged flexural stiffness. The location for acceleration measurement is element 5 where the response is dependent on several modes. The input data are the acceleration induced by a moving load. Because the input data induced by a



Density (ρ): 3850 kg/m^3 Area (A): 11.332 m^2
Poisson's Ratio (ν): 0.17 Load (P): 17 ton
Moment of Inertia (I): 18.638 m^4
Young's Modulus (E): 3.3×10^{10} N/m^2
Velocity of Moving Load (c): 300 km/hr

Fig. 8. Finite Element Model of Bridge.

moving load is obtained, traffic control is not needed and the task of measurement is easy. The sub-critical velocity of Kyung-bu highspeed train occurrs at 300 km/hr which is less than the design velocity of 350 km/hr.[1] Therefore, the velocity of a moving load is chosen to be 300 km/hr.

When the numbers of input data are 200, 400, 600 and 800, the time intervals are 0.004, 0.002, 0.0013 and 0.001 seconds, respectively. The number of hidden and output neurons are 11, 20, respectively. The output from the output layer represents the residual stiffness in each element. The input data are normalized with the largest number. Unless the input data are normalized, the differences between the initial weight and the input data are large so the learning of the neural networks may not be successful.

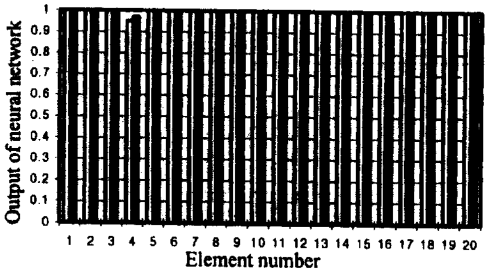
3.5 Cases to Train Elements 4 and 10

The training cases are shown in Table 1. The damage locations to be trained are in element 4 and 10. Cases 1~4 represent that damage has occurred only in element 4, cases 5~8 only in element 10 and cases 9~12 both in element 4 and element 10 with same extents of damages. The damage extents of element 4 or 10 are 10, 20, 30 and 40%. The initial values of learning rate and momentum constant are 0.01, 0.95, respectively. The stopping condition is either less than sum-squared error of 0.001 or more than epoch of 10000.

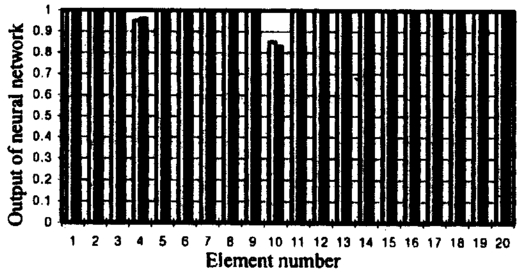
Fig. 9 shows that neural network detects the damage locations and extents. Right and left bar represent the output of neural network and residual stiffness, respectively. Fig. 9a represents the testing of neural network when only element 4 has damage (5%) and Fig. 9b shows the case where element 4 (5%) and 10 (15%) have damage. The summary of damage assessments is shown in Table 2. In Table 2a, columns 2~4 represent the damage conditions of each element and

Table 1. Training Cases (Measurement: Element 5)

Case	Damage Extents (%)	
	Element 4	Element 10
1	10	
2	20	
3	30	
4	40	
5		10
6		20
7		30
8		40
9	10	10
10	20	20
11	30	30
12	40	40



(a)



(b)

Fig. 9. (a) Testing of the trained neural network when only element 4 (5%) has the damage. (b) Testing of the trained neural network when element 4 (5%) and 10 (15%) have the damage.
Left bar: residual stiffness of the structure.
Right bar: output from neural network.

Table 2a. Output from neural Network (Measurement: Element 5)

Case	Damage Condition (%)			Error in Damage Assessments* (%)			
				Number of Data			
	Element 4	Element 10	Element 15	200	400	600	800
1	5			-4	-2	-1	-2
2	15			-14	3	5	7
3	25			-22	-2	-1	-6
4	35			-46	0	5	1
5		5		-4	-2	-3	1
6		15		-16	2	-3	-2
7		25		-32	-1	6	13
8		35		32	3	-1	-15
9			5	-4	-4	-4	-4
10			15	-16	-16	-16	-16
11			25	-32	-32	-32	-32
12			35	-52	-52	-52	-52

*error=(residual stiffness-output of neural network)/output of neural network.

Table 2b. Output from neural network (Measurement: Element 5)

Case	Damage Condition (%)		Error in Damage Assessments* (%)							
			Number of Data							
			200		400		600		800	
	Element 4	Element 10	Element 4	Element 10	Element 4	Element 10	Element 4	Element 10	Element 4	Element 10
13	5	5	-4	-4	-1	-2	0	-1	-2	-2
14	5	15	-4	-15	-1	2	0	-2	-2	2
15	5	25	-4	-30	-2	1	-2	5	-4	6
16	5	35	-4	-50	-4	9	-3	-1	-4	6
17	15	15	-9	-16	-3	1	-5	0	-16	-3
18	15	25	-16	-32	-12	12	-7	-4	0	1
19	15	35	-16	-52	-12	1	-10	3	26	-52
20	25	25	-26	-32	-6	-17	-22	34	19	-32
21	25	35	-26	-52	6	7	-25	33	12	0
22	35	35	-52	-52	-30	-41	26	-46	-12	0

*error=(residual stiffness-output of neural network)/output of neural network.

columns 5~8 the errors in damage assessments with the increase in the number of data. In the case where the number of data is 200, neural network cannot detect the damage locations and extents. In cases 9~12, the damage assessments are not successful. In case where the number of data are 400, 600 and 800, the damage assessments are successful. Table 2b represents the output from neural network in case where two elements are damaged simultaneously. In case where the number of data is 200, neural network cannot detect the damage locations and extents.

3.6 Cases to Train All Elements Twice

All elements are assumed to be damaged and trained twice by ANN and the number of data is 400. For the training, the damage extents of all elements are 10, 20, 30 and 40% reduction of undamaged flexural stiffness. The initial value of learning rate and momentum constant are 0.01, 0.95, respectively. The stopping condition is either less than sum-squared error of 0.001 or more than epoch of 10000.

The summary of damage assessments is shown in Table 3. Table 3a and 3b are the cases to be trained and not to be trained, respectively. As training is fulfilled in case of one damage only, neural network cannot assess two or more damages. For the cases not to be trained, consequently, the neural network can not detect the damage locations and extents.

If the number of input data is under certain limit, damage is not assessed correctly. When the number of input data is 200, damage assessment fails in spite of the cases to be trained. The first, second and third natural frequency of structure are 22 Hz, 87 Hz and 114 Hz. When the number of input data is 200, measurement interval is 0.004 second. As the first natural frequency is within 10 percent of 250/2=125 Hz, or 12.5 Hz, the input data has reliability. Because the first natural frequency, however, is more than 12.5 Hz, the number of input data, 200, has no reliability. In the case where the number of input data is 400 because the first

Table 3a. Output from Neural Network (Measurement: Element 5)

Case	Damage Condition (%)	Error in Damage Assessments* (%)		
		Element 6	Element 12	Element 16
1	5	-4	-4	-4
2	15	-2	-3	-4
3	25	-13	-8	-4
4	35	24	7	0

*error=(residual stiffness-output of neural network)/output of neural network.

Table 3b. Output from neural Network (Measurement: Elment 5)

Case	Damage Condition (%)		Error in Damage Assessments* (%)	
	Element 6	Element 12	Element 6	Element 12
5	5	5	-3	-2
6	5	15	-4	-1
7	5	25	-4	-13
8	5	35	-4	-9
9	15	15	-14	-15
10	15	25	-15	-36
11	15	35	-11	-33
12	25	25	-4	-29
13	25	35	-10	-52
14	35	35	-52	-52

*error=(residual stiffness-output of neural network)/output of neural network.

Table 3c. Output from Neural Network (Measurement: Element 5)

Case	Damage Condition (%)			Error in Damage Assessments* (%)		
	Element 6	Element 12	Element 16	Element 6	Element 12	Element 16
15	5	5	5	-4	-4	-4
16	5	15	15	-4	-16	-15
17	5	15	25	-4	-16	-29
18	15	15	15	-16	-16	-14
19	15	25	25	-14	-32	-32

*error=(residual stiffness-output of neural network)/output of neural network.

natural frequency is within 25 Hz, this input data have reliability. Finally, in order to assess damage, the number of input data in the structure must be at least 400 (sampling time: 0.002 second).

4. Conclusions

Using the acceleration induced by a moving load, the damage assessments of high-speed train bridge are performed. Throughout the simulations, it is shown that the damage assessments of structures using neural networks are promising and the following conclusions are drawn.

1. When damage is on only one element and trained by ANN, damage locations are detected exactly, but the maximum of sum-squared error is 16%.

2. In the element to be trained, the errors in damage extents are almost the same in more than 400 sampled input data.

3. In the elements and cases not to be trained, damage assessments are not efficient. Moreover, as the data is increased, errors do not decrease.

4. When damages are on two elements simultaneously and trained by ANN, the damage locations are detected exactly, but the damage extents are less accurate than the case of one damaged element.

5. If damages occur in the trained element, the errors in damage extents are not increased or decreased with the increase of damage extents. But, in the element not to be trained, the errors of damage extents are increased with the increase of damage extents.

6. The criterion for the sampling frequency of acceleration of structures is recommended as 20 times of the first natural frequency of structures.

References

1. Ho-Chul Kwon (1996). *"Vibration control of bridges under moving loads,"* KAIST MD Thesis.
2. Howard B. Wilson and Louis H. Turcotte (1994). *Advanced Mathematics and Mechanics Applications Using MATLAB*, CRC Press.
3. John Hertz, Anders Krogh and Richard G. Palmer (1991). *Introduction to the Theory of Neural Computation*, Addison-Wesley Publishing Company.
4. J. Ghaboussi, J.H. Garrett Jr., X. and Wu (1991). "Knowledge-Based modeling of material behavior with neural networks," *Journal of Engineering Mechanics*, Vol. 117, No. 1, pp. 132-153.
5. Neural Ware, *Neural Computing*, 1989.
6. S.F. Masri, M. Nakamura, A.G. Chassiakos and T.K. Caughey (1996). "Neural network approach to detection of changes in structural parameters," *Journal of Engineering Mechanics*, Vol. 122, No. 4, pp. 350-360.
7. S. Hassiotis and G.D. Jeong (1993). "Assessment of structural damage from natural frequency measurements," *Computers & Structures*, Vol. 49, No. 4, pp. 679-691.
8. Simon Haykin, *Neural Networks*, Macmillan, 1994.
9. S.V. Barai and P.C. Pandey (1995). "Vibration signature analysis using artificial neural networks," *Journal of Computing in Civil Engineering*, Vol. 9, No. 4, pp. 259-265.
10. Valluru B. Rao and Hayagriva V. Rao (1993). C++ *Neural Networks and Fuzzy Logic*, MIS: Press.
11. X. Wu, J. Ghaboussi and J.H. Garrett, Jr. (1992). "Use of neural networks in detection of structural damage," *Computers & Structures*, Vol. 42, No. 4, pp. 649-659.
12. Zbigniew P. Szewczyk and P. Hajela (1992). "Neural networks based damage detection in structures," *Journal of Computing in Civil Engineering*, Vol. 8, No. 2, pp. 1163-1170.

(接受 : 1997. 5. 15)