Structural Engineering and Mechanics, Vol. 17, No. 3-4 (2004) 483-492
DOI: http://dx.doi.org/10.12989/sem.2004.17.3_4.483

Damage assessment of cable stayed bridge using probabilistic neural network

Hyo-Nam Cho†, Young-Min Choi‡, Sung-Chil Lee‡† and Choon-Kun Hur‡†

Department of Civil & Environmental Engineering, Hanyang University,
Ansan, Kyunggi 425-791, South Korea

(Received October 20, 2002, Accepted July 19, 2003)

Abstract. This paper presents an efficient algorithm for the estimation of damage location and severity in bridge structures using Probabilistic Neural Network (PNN). Generally, the Back Propagation Neural Network (BPNN)-based damage detection methods need a lot of training patterns for neural network learning process and the optimum architecture of a BPNN is selected by trial and error. In this paper, the PNN instead of the conventional BPNN is used as a pattern classifier. The modal properties of damaged structure are somewhat different from those of undamaged one. The basic idea of proposed algorithm is that the PNN classifies a test pattern which consists of the modal characteristics from damaged structure, how close it is to each training pattern which is composed of the modal characteristics from various structural damage cases. In this algorithm, two PNNs are sequentially used. The first PNN estimates the damage location using mode shape and the results of the first PNN are put into the second PNN for the damage severity estimation using natural frequency. The proposed damage assessment algorithm using the PNN is applied to a cable-stayed bridge to verify its applicability.

Key words: probabilistic neural network; damage identification; cable-stayed bridge.

1. Introduction

Artificial neural networks (ANN) have been being used for damage assessment by many researchers, but there are still some barriers that must be overcome to improve accuracy and efficiency. The major problems with the conventional neural network are the necessity of many training data set for the learning process of ANN and the ambiguity in the relation between neural network model and convergence of solution.

Wu *et al*. (1992) used 43 training patterns using the Fourier spectra of the computed relative acceleration time histories of the top floor for the damage assessment of a three storey frame. Szewczyk and Hajela (1992) used 3600 randomly generated training patterns using the relationship between structural elements and global static displacements. With these training patterns, they performed the damage assessment for the 9-bending element structure. Elkordy *et al.* (1993) used three BPNN for the damage assessment of five storey steel frame. They selected the optimum neural

† Professor
‡ BK Research Professor
‡† Graduate Student

network architecture by trial and error. Pandey and Barai (1995) used BPNN with 40 training patterns for the damage identification of 21-bar truss structure. They performed the detailed study with two neural network architectures to examine the performance of neural network. Yun and Bahng (2000) adopted BPNN for damage identification of 51-bar truss structure using 1200 training patterns. From many previous researches, it can be seen that the relation between the number of training patterns for the learning process of neural network and the architecture of neural network which gives the best performance need to be investigated further.

Though the architecture of the neural network affects the required number of training patterns, however, there is no firm method to determine the relationship. Baum and Haussler (1989) proposed the equation for the training pattern size for the case of a neural network containing a single hidden layer, and used as a binary classifier as follows.

$$N \geq \frac{32W}{\varepsilon}\ln\left(\frac{32M}{\varepsilon}\right) \tag{1}$$

where N denotes the required number of the training pattern; W denotes the total number of synaptic weights in the network; M denotes the total number of hidden computation nodes; and ε denotes the fraction of errors permitted on test. Ignoring the logarithmic factor in Eq. (1), it seems that, in practice, Eq. (2) is the required condition for a good generalization.

$$N \geq \frac{W}{\varepsilon} \tag{2}$$

Thus, with an error of 10 percent, the number of training patterns should be approximately 10 times the number of synaptic weights in the network.

Therefore, the training of neural network needs so many training patterns, and the number of training pattern means the same number of structural analysis to generate those training patterns in structural damage identification. As mentioned above, the fixing of architecture of neural network is also a difficult task. Generally, the architecture of neural network is selected by trial and error. It takes a lot of time to prepare a number of training pattern and determine optimum neural network architecture by trial and error.

The PNN could be the answer to those problems because the PNN needs not so much training pattern as conventional BPNN does and there is no need to determine the architecture of the PNN. Moreover the PNN always shows the convergence. The characteristics and the architecture of the PNN will be described in the subsequent part. The objective of this paper is to develop the PNN-based damage assessment algorithm and apply it to cable-stayed bridges.

2. Probabilistic Neural Network (PNN)

Consider the two-category situation in which the state of nature θ is known to be either θ_A, or θ_B. If it is desired to decide whether $\theta = \theta_A$ or $\theta = \theta_B$ based on a set of measurements represented by the p-dimensional vector $\boldsymbol{X} = \{x_1\ x_2\ ...\ x_i\ ...\ x_p\}^T$, the Bayes decision rule becomes

$$\begin{aligned} d(\mathbf{X}) \in \theta_A \quad &\text{if} \quad h_A l_A f_A(\mathbf{X}) > h_B l_B f_B(\mathbf{X}) \\ d(\mathbf{X}) \in \theta_B \quad &\text{if} \quad h_B l_B f_B(\mathbf{X}) > h_A l_A f_A(\mathbf{X}) \end{aligned} \tag{3}$$

Where $d(\boldsymbol{X})$ is the decision on test vector(pattern) $\boldsymbol{X}$; h_A, h_B are the *a priori* probabilities of the categories θ_A and θ_B respectively; l_A is the loss associated with misclassifying $d(\boldsymbol{X}) \notin \theta_A$ when $\theta \in \theta_A$ and l_B is the loss associated with misclassifying $d(\boldsymbol{X}) \notin \theta_B$ when $\theta \in \theta_B$; and $f_A(\boldsymbol{X})$, $f_B(\boldsymbol{X})$ are the probability density functions for categories θ_A and θ_B respectively.

For the damage detection problem, h and l are usually assumed to be equal for all categories. However, if the probability densities of the patterns in the categories to be separated are unknown, and all that is given is a set of training patterns (training samples), then it is these samples which provide the only clue to the unknown underlying probability densities (Specht 1990). Here, the method of Parzen windows is used to estimate the PDFs in terms of kernel density estimators (Wasserman).

$$f_q(\boldsymbol{X}) = \frac{1}{n_q(2\pi)^{p/2}\sigma^p}\sum_{i=1}^{n_q}\exp\left[-\frac{(\boldsymbol{X}-\boldsymbol{X}_{qi})^T(\boldsymbol{X}-\boldsymbol{X}_{qi})}{2\sigma^2}\right] \tag{4}$$

Where $\boldsymbol{X}$ is the test vector to be classified; $f_q(\boldsymbol{X})$ is the value of the PDF of category q at point $\boldsymbol{X}$; n_q is the number of training vectors in category q; p is the dimensionality of the training vectors; $\boldsymbol{X}_{qi}$ is the i-th training vector for category q; and σ is the smoothing parameter. The PNN consists of three layers as shown in Fig. 1. The number of the input nodes is determined in the same way as for conventional neural network. But the number of hidden layer and nodes in hidden layer is determined automatically. The PNN has single hidden layer, and the number of the nodes in hidden layer is the same as the number of training patterns. The number of output nodes is the same as the number of classes. When a test pattern is presented to the PNN, the first layer computes distances from the test pattern to the training patterns, and produces a vector whose elements indicate how close the test pattern is to each training pattern. The second layer sums these contributions for each class of inputs to produce as its net output, a vector of probabilities. Finally, a compete transfer function on the output of the second layer picks the maximum of these probabilities.

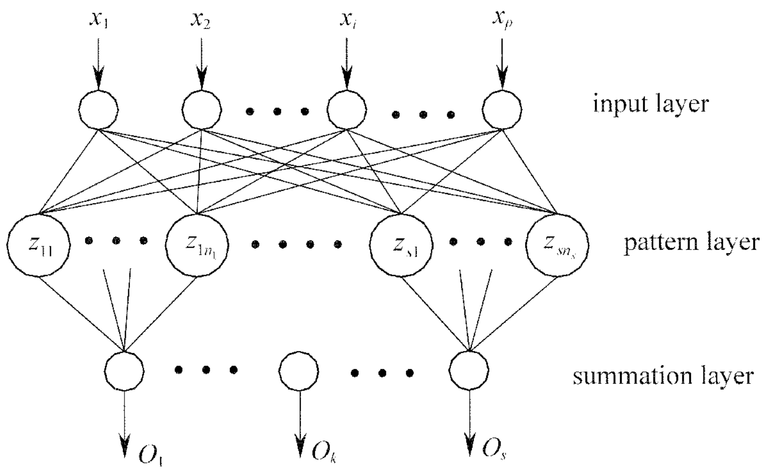

Fig. 1 Architecture of three-layer PNN (Ni *et al.* 2000)

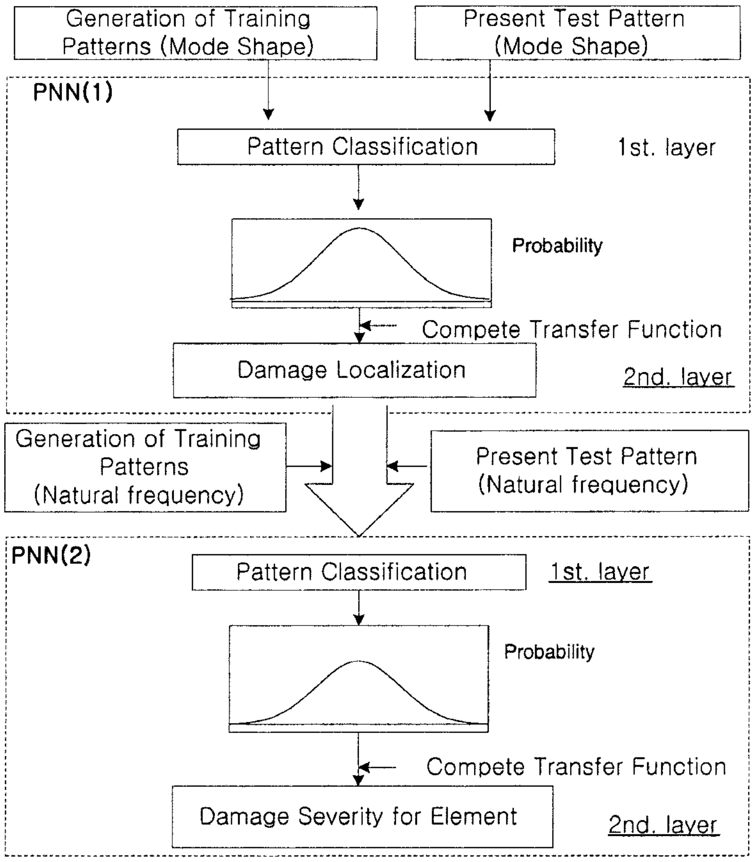

Fig. 2 Algorithm for damage assessment

3. Algorithm for damage assessment

Two PNNs are used for the proposed damage assessment algorithm. Damage locations are identified through the first PNN (PNN(1)), and damage severity is estimated by the second PNN (PNN(2)). The procedure for the damage assessment using the two PNNs is shown in Fig. 2. The mode shapes of a damaged structure are used as both training patterns and test patterns in PNN(1). The training patterns of the PNN(1) are generated based on many damage cases. But there are so many members in a cable-stayed bridge that it is not easy to determine which element is damaged. Therefore, whole bridge is divided into several classes (Fig. 3). The whole bridge could be divided geometrically or functionally. In this paper, a cable-stayed bridge is divided geometrically and functionally for damage assessment, since though cable, deck and pylon element are interrelated each other in a cable-stayed bridge, they could be distinguished geometrically and functionally. If a test pattern is presented to the PNN(1), damage identification is implemented for the class which has damaged elements. After damaged class is identified, then, damage localization is predicted at element level.

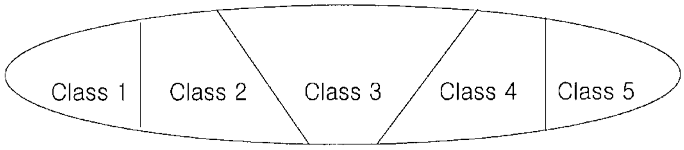

Fig. 3. Classification of a structure

When a test pattern which is composed of mode shape of damaged structure is presented, the first layer of PNN(1) determines how close the test pattern is to a training pattern and then the output of the first layer are summed for each class. The third layer of the PNN(1) produces the probability for each class. Once the damaged class is determined, the expected damaged members are reduced to the members of that class. In PNN(2), the natural frequencies are used as both training patterns and test pattern. The natural frequencies of training patterns for damage scenarios are generated by Finite Element Analysis. The natural frequencies of damaged structure are put into the PNN(2) and the damage severity is estimated by the similar procedure for the PNN(1).

4. Numerical example

The proposed algorithm is applied numerically to a typical cable-stayed bridge model as shown in Fig. 4. For the Finite Element Analysis of the structure, MIDAS Civil is used.

4.1 Classes

The applied cable-stayed bridge model has three spans and its total length is 484 m. The finite element model of the bridge contains 226 nodes and 285 elements. In Fig. 4, the whole bridge is divided into six classes. From the first class to the fourth class, each class is composed of main cable element. The fifth and the sixth class contain main girder element. A cable element of the FE model is denoted by C and deck element by B for convenience. For example, the left cables of left tower are classified as class 1, and composed of C1~C6 in Fig. 4. Table 1 shows the summary of classes and elements belonging to each class.

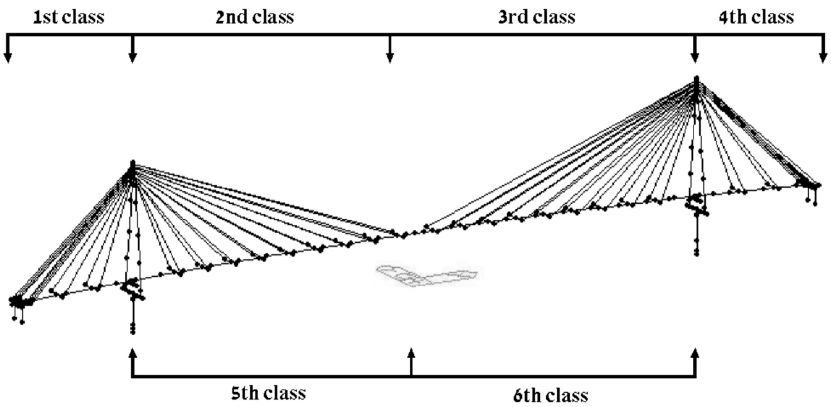

Fig. 4 FE model and classes of numerical example

Table 1 Class and its elements

Class	Element
1	C1-C6
2	C7-C15
3	C16-C24
4	C25-C30
5	B16-B35
6	B36-B56

Table 2 Simulated damage

Damage case	No. of class	No. of element	Stiffness reduction
1	5	B35	40%
2	3	C17	40%
3	5, 6	B35, B44	40%, 40%

4.2 Training patterns and test patterns

In damage detection using the PNN, the selection of the proper training patterns is a very important thing. Damages of the structure must be reflected in training patterns. The training patterns are made from the natural frequency and the mode shape of the structure. The vibration mode shapes of the applied model are obtained by FEA. From the FEA results, the first and the fourth vibration modes are horizontal mode and the second and the third one are vertical mode. When monitoring the behavior of bridge in the field, the vertical components are mainly acquired. So it is more reasonable to use vertical vibration mode and the natural frequency of that mode as training patterns. In this paper, the first vertical vibration mode shape of the model is used for damage assessment. The training patterns are made as follows;

$$\textit{Training Patterns} = \left[\left(\frac{\phi^* - \phi}{Max\ v}\right)\right] \tag{5}$$

Where ϕ^*= the first vertical mode shape of a damaged model, ϕ = the first mode shape of undamaged model; and $Max\ v$ = the maximum value of the $(\phi^* - \phi)$.

The training patterns are normalized by the maximum difference value between the damaged mode shape and the undamaged mode shape. The test patterns used for damage identification are made similar to the procedure for training patterns. If the damage severity for generation of training patterns is too low or too high, a training pattern could not represent the characteristics of the damage for various damage levels. Thus for the generation of the training pattern, the stiffness reduction by 30% is applied to each element in a class. For the test pattern, damage is introduced by reducing the stiffness of element by 40%. Table 2 shows the damage cases, damaged element, and stiffness reduction in element for each damage case.

4.3 Numerical results

The damage identification for Damage Case 1 is shown in Fig. 5. In Damage Case 1, the element

B35 is damaged. From Fig. 5, the damage probability of Class 5 is higher than any other classes. It can be determined that the damages are present in some elements of Class 5.

But the damage probability of Class 6 is similar to that of Class 5 in Fig. 5. It is thought that because the element B35, which belongs to the Class 5, is near the Class 6, the effect of damage in the element B35 is also reflected to Class 6.

Thus, it may be stated if the damaged elements are near the boundary of classes, damage estimation show high probability for both the adjacent classes. Fig. 6 shows the damage localization results at element level. From the Fig. 6, the damage probability of the element B35 is highest.

The numerical results for Damage Case 2 are shown in Figs. 7 and 8. For Damage Case 2, the damage is introduced to the cable element C17 of Class 3 reducing the stiffness by 40%. Fig. 7

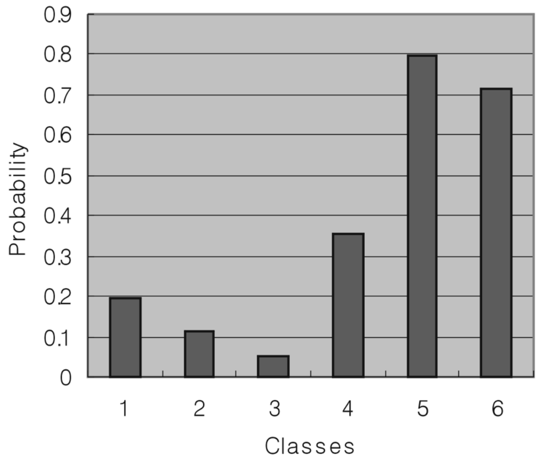

Fig. 5 Damaged element B35 (40%)

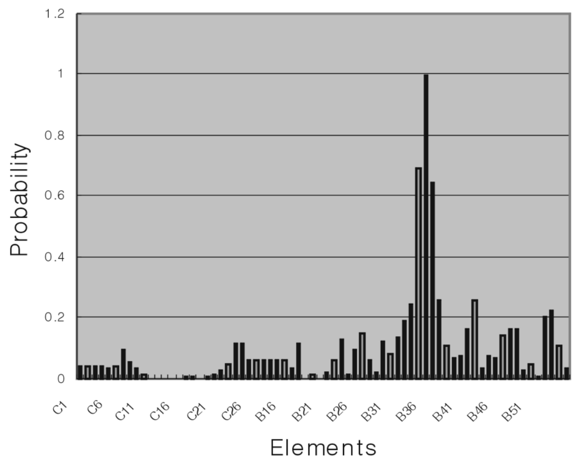

Fig. 6 Damaged element B35 (40%)

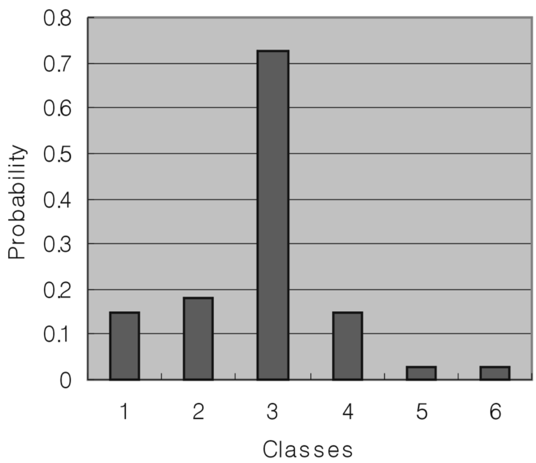

Fig. 7 Damaged element C17 (40%)

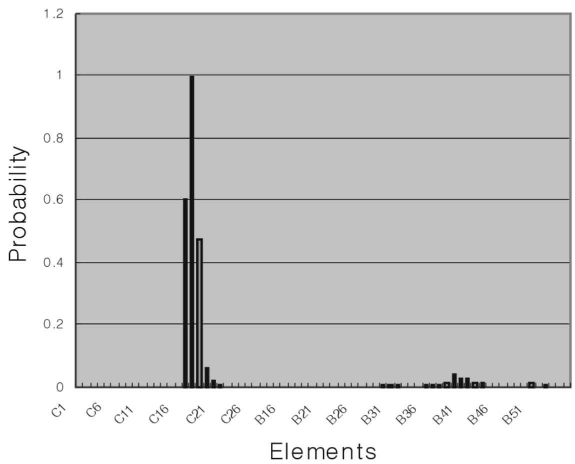

Fig. 8 Damaged element C17 (40%)

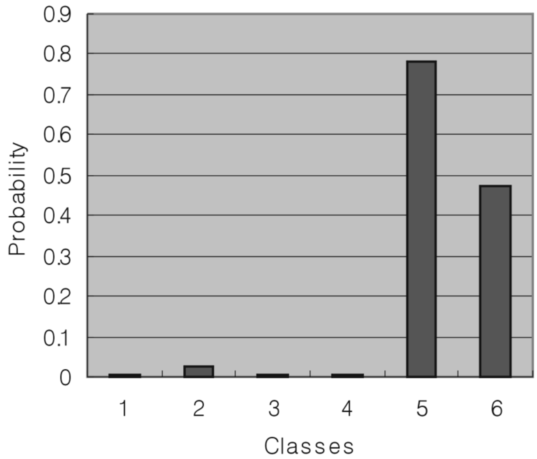

Fig. 9 Damaged elements B35, B44 (40%)

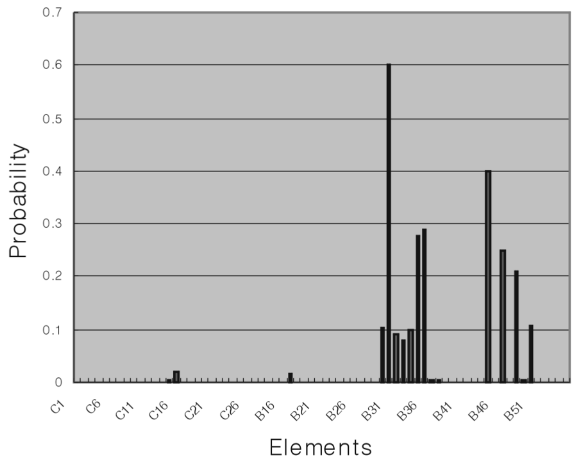

Fig. 10 Damaged element B35, B44 (40%)

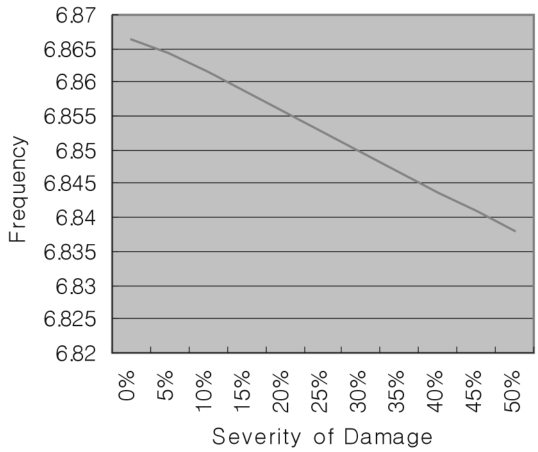

Fig. 11 The natural frequency versus damage severity of element B35

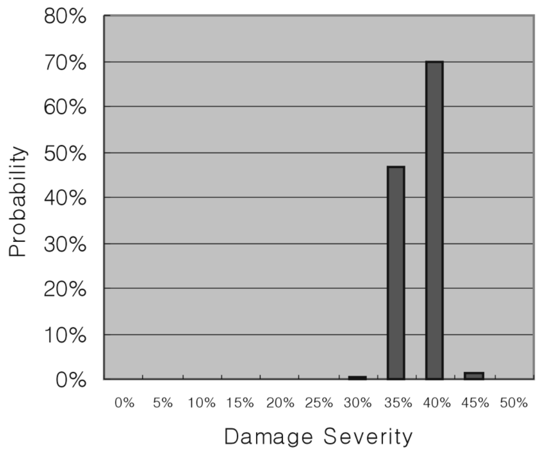

Fig. 12 Damage severity estimation for damage Case 1

shows that Class 3 has damaged elements and from Fig. 8, it can be determined that the element C17 is damaged. Although the element C17 is near the Class 2, the results from Figs. 7 and 8 pick clearly the damaged class and the damaged element, comparing with the results from Damage Case 1 in which Class 5 or 6 is supposed to be damaged.

When two elements B35 and B44 are damaged, the damage localization results are shown in Figs. 9 and 10. The elements B35 and B44 belong to the Class 5 and 6 respectively. In Fig. 9, the PNN picks the Class 5 and 6 as the damaged class and in Fig. 10, the damage probabilities of the elements B34 and B44 are large enough to be considered as the damaged members. It is found that that the higher the damaged level is, the more accurate the damage detection result is. The damage severity can be estimated by the PNN(2). To make the training patterns for the PNN(2), the natural frequencies versus damage severities are needed. Once a test pattern is presented to the PNN(2), it produces the damage severity probability to that pattern. For example, Fig. 11 shows the frequency versus damage severity for the element B35. It can be seen that the relationship between the frequency and damage severity is almost linear. The damage severity estimation for Damage Case 1, in which the stiffness of the element B35 is reduced by 40%, is shown in Fig. 12. The damage severity probability for 40% is highest in Fig. 12, so it can be concluded that the damage severity for the element B35 is 40%.

For the prevention of structural failure, a damage detection method could detect damages in their earlier stage. The proposed damage detection method applied to the stiffness reduction of 10% showed a good result for small damage level.

5. Conclusions

A method of damage identification for structures has been presented in this paper. It is based on using two PNNs as a pattern classifier. The first PNN is applied for damage localization using mode shape and the second one is applied for the estimation of damage severity using natural frequency. To verify the damage assessment algorithm, it is numerically applied to cable-stayed bridge model. The training and test patterns for the damage localization are made from the first vertical vibration mode shape of the model. The damage severity is estimated by the training and test patterns which are made from the natural frequency of the model. To improve the accuracy of the damage localization, the applied model is subdivided into several classes.

From this study, the following conclusions could be drawn;

(1) Unlike the multi-layer neural network, especially BPNN, the PNN has the fixed architecture, and thus there is no need to waste time in determining the appropriate neural network architecture by trial and error.

(2) Most of artificial neural network-based damage identification algorithms need too many training patterns, when the structure is complicated and complex. But the PNN generally requires fewer training patterns than BPNN.

(3) From the numerical example, the proposed algorithm could detect damaged members and severity with small number of training patterns.

Acknowledgements

This work was supported by grant No. (1999-2-31100-002-5) from the Basic Research program of the Korea Science & Engineering Foundation.

References

Baum, E.B. and Haussler, D. (1989), “What size net gives valid generalization?”, *Neural Computation*, **1**, 151-160.

Elkordy, M.F., Chang, K.C. and Lee, G.C. (1993), “Neural networks trained by analytically simulated damaged states”, *J. Computing in Civil Engng*., ASCE, **7**, 130-145.

Ni, Y.Q., Zhou, X.T., Ko, J.M. and Wang, B.S. (2000), “Vibration-based damage localization in Ting Kau Bridge using probabilistic neural network”, *Advances in Structural Dynamics*, Elsevier Science Ltd., Oxford, UK, **2**, 1069-1076, *Proc. of the Int. Conf. on Advances in Structural Dynamics.*

Pandey, P.C. and Barai, S.V. (1995), "Multilayer perceptron in damage detection of bridge strucutures", *Comput. Struct.*, **54**, 597-608.

Specht, D.F. (1990), "Probabilistic neural networks", *Neural Networks* **3**, 109-118.

Szewczyk, Z.P. and Hajela, P. (1992), "Neural networks based damage detection in structures", Technical Report, RPI, Troy.

Wasserman, P.D., *Advanced Methods in Neural Computing*, Van Nostrand Reinhold, New York.

Wu, X., Ghaboussi, J. and Garrett, J.H. (1992), "Use of neural networks in detection of structural damage", *Comput. Struct.*, **42**, 649-659.

Yun, Chung-Bang and Bahng, Eun Young (2000), "Substructural identification using neural networks", *Comput. Struct.*, **77**, 41-52.