방 재 연 구 논 문 집
제2권 제3호 · 2000년 9월
PP. 95 ~ 104

# Neural Network Technique for Structural Vibration Control under Earthquakes

Kim, Dong-Hyawn · Lee, In-Won · Jung, Gil-Ho

## SUMMARY

Neuro-controller training algorithm based on cost function is applied to multi-degree of freedom system. And sensitivity evaluation algorithm replacing emulator neural network is proposed. In conventional methods, emulator neural network is used to evaluate sensitivity of structural response to control signal. To use emulator, it should be trained to predict the dynamic response of the structure. Much of time is usually spent on training of emulator. In proposed algorithm, however, it takes only one sampling time to obtain the sensitivity. Therefore, training time for emulator is eliminated. In result, only one neural network is used for neuro-control system. In numerical example, three-story building structure with linear and nonlinear stiffness is controlled by trained neural network. The actuator dynamics and control time delay are considered in simulation. Numerical examples show that the proposed control algorithm is valid in structural control.

*KEY WORDS*

***Neural network, Control, Structural vibration, Sensitivity, Emulator, Training***

## INTRODUCTION

Artificial neural networks have been widely used in many engineering fields such as system identification, optimization, vibration control and etc. Learning capability of neural networks gives the opportunity of solving complicated problems whose analytic or numerical solution is hard to obtain, if any. Structural vibration control is one of those problems. With high nonlinearity and modeling uncertainty, it is not easy or impossible to design a controller by conventional approaches that are based on mathematical model of structure. However, design methodology based on neural networks needs not the mathematical model of structure. Instead of the model, only the structural responses are used for design of neural network controller, or simply neuro-controller.

Pioneering studies by J. Ghaboussi et al.[4] and H. M. Chen et al.[10] simultaneously showed that neural networks can successfully control structures under ground excitations. Their work made the historic start of the neuro-control approach in structural control. Chen et al. showed the simulation results that the vibration of high-rise apartment structure under ground excitation could be suppressed by neuro-controller. They used standard multilayer perceptron for controller. Ghaboussi et al. showed the advanced simulation results that controller neural network can be trained when the dynamics of actuator is considered. And they also proposed the training algorithm aiming the structural vibration to be decreased smoothly. For the training of the network, the responses of a few future steps are predicted by the so-called emulator neural network. Emulator neural network is a neural network that learns the response due to dynamic loads and predicts the response with the appropriate inputs such as the response and the external loads of the previous steps. K. Nikzad et al.[8] proposed the delay compensation technique that occurs in real system. K. Bani-Hani et al.[6] proposed that neuro-controller could be also applied to the control of nonlinear structure having hysteretic property. They emphasized that only the neuro-controller trained with nonlinear dynamics can successfully control the vibration of nonlinear structure. K. Bani-Hani et al.[7] applied their neuro-control system to a benchmark problem initiated by the Structural Control Committee of ASCE (American Society of Civil Engineers).

There are some problems that are still not solved. One of the most important problems is that one should predetermine the desired structural response for the training of a neural controller. This is well understood in the training criterion of neural network. One should define the criterion to train a neural network. In most controller proposed until now, the criterion is defined as the squared sum of offset between the actual and the desired responses. By minimizing the criterion, the actual responses become close to the desired responses. Hence, one should define the desired structural response in advance of the training. Chen et al. defined the desired response as zero, which means the structural responses should disappear. This may cause the saturation of the actuator. J. Ghaboussi et al., however, used algorithm aiming the average of expected responses for a few future time steps to be zero. This scheme aims at the smooth reduction of structural responses. Although the desired response can be set by some strategy, the selection of desired response is not straightforward, and it may not be optimal. J. T. Kim et al.[2,] , however, proposed a new training algorithm that does not need desired response. They used cost function as a training criterion and showed that single degree of freedom structure under ground motion could be successfully controlled by neural network trained through minimization of the cost function. The algorithm is used and extended to the control of multi-degree of freedom structure in this study. A training rule for neuro-controller is derived by minimizing the cost function. There is no need to set the desired response at each training step. The controller trained by this scheme can be said to be optimal neuro-controller because the cost function is minimized after training.

Another problem may be the existence of the emulator neural network. In neuro-control system two neural networks are used. One is for the controller. The other is for the emulator. Emulator gives the sensitivity of structural response that is needed for the training of the controller. If the controller is trained without the help of emulator, the neuro-control system becomes compact and the total training time for neuro-control system can also be reduced. To do so, a sensitivity evaluation algorithm is proposed. Sensitivities of structural response to control input are obtained through the algorithm. Therefore, emulator neural network is not used in training of controller neural network any more. By doing so, time spent on the training of the emulator can be eliminated.

In the simulation study, three-story shear building is controlled. The structure is assumed to be linear and nonlinear, respectively, in the numerical examples. An active mass damper (AMD) is installed at the roof of the structure for the generation of control force. The dynamics of hydraulic actuator and time delay of the controller are also considered.

## VIBRATION CONTROL USING NEURAL NETWORK

### INTRODUCTION TO NEURAL NETWORK

Figure 1 shows a multilayer-type neural network. It consists of three layers; input, hidden, and output layer. There may be more hidden layers according to the complexity of problems. Each layer has $n1$, $n2$, and $n3$ nodes. When input layer takes information at each node, the information is weighted and transferred to the nodes of hidden layer. The hidden layer also transfers the information to output layer. Then, the output of the MLP is obtained. The number of layers and that of nodes in each layer should be chosen by experiences or trial and errors.

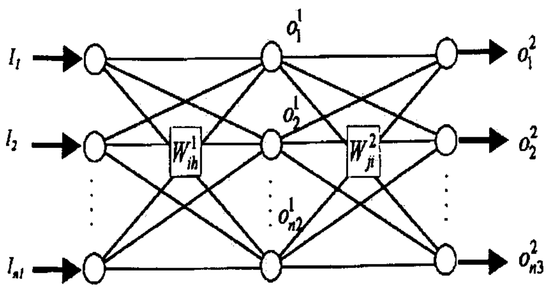



Figure 1. Structure of multilayer perceptron with one hidden layer

The inputs to the network are denoted by $I_h(h=1,2,\cdots,n1)$, and the net outputs of hidden layer are expressed as

$$o_i^1=f^1(net_i^1),(i=1,2,\cdots,n1) \quad (1)$$

where $net_i^1$ is the net input of $i$-th node of hidden layer and can be expressed as

$$net_i^1=\sum_{n+1}^{n1} W_{ih}^1 I_h + b_i^1 \quad (2)$$

where $W_{ih}^1$ is the connection weight between input and hidden layer, $b_i^1$ the bias of hidden layer. In equation (1), $f^1$ is activation function of hidden layer. Likewise, the relationship between the net input and output of the second layer is

$$o_j^2=f^2(net_j^2),(j=1,2,\cdots,n3) \quad (3)$$

In equation (3), the net input is

$$net_j^2=\sum_{i=1}^{n2} W_{ji}^2 o_i^1 + b_j^2 \quad (4)$$

where $W_{ji}^2$ denotes the connection weight between hidden and output layer, $b_j^2$ the bias of the output layer.

The weights and biases should be determined in order that the neural network predicts desired output. The determination of the optimal weights and biases is the so-called learning or training. Training is accomplished by minimizing a criterion defined by the squared sum of the offset values between the actual and the desired outputs of the neural network as

$$E=\sum |o_d-o_a|^2 \quad (5)$$

where $o_d$ and $o_a$ denote the desired output and the actual network output, respectively. By minimizing the error function through training rule, the network outputs become close to the desired outputs. Finally, when the neural network has input, it predicts the desired output related to the input. If the desired output is control signal, it becomes a controller. However, if the desired output is not directly available, the control problem belongs to this case, the training is not straightforward. One should obtain the desired output in some way to use the error function in training. To avoid this, J. T. Kim et al.[2, 3] proposed a training algorithm using cost function that is always available at hand. The algorithm was applied to the control of one-dimensional structure. The algorithm is used here to train a neural network controller. Detailed algorithm will be described in the next section.

## CONTROL ALGORITHM AND TRAINING RULE

Figure 2 shows the block diagram for the conventional neuro-control. The actual structural response is compared with the desired response that is determined in some way. The sensitivity of the structural response to input load is calculated by using emulator neural network. Then, using the desired response and the sensitivity, the controller neural network is trained. However, the sensitivity is obtained not by the emulator but by an evaluation algorithm in the proposed approach. Hence, the emulator is not required in the neuro-controller any more. In addition, a cost function is used to train the neuro-controller. One does not have to determine the desired output to train the controller. The control block diagram for the proposed method is shown in Figure 3.

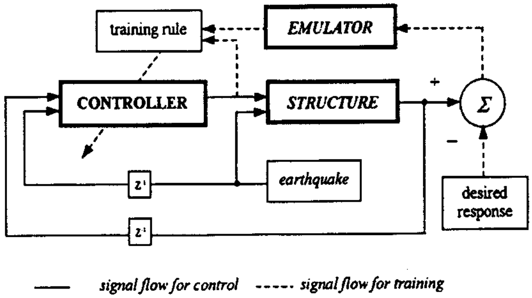



Figure 2. Control diagram for the conventional neuro-control

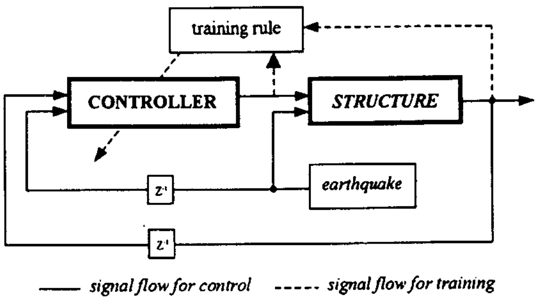



Figure 3. Control diagram for the proposed neuro-control

$$J=\frac{1}{2}\int_0^{T_f}(Z^TQ_Z+u^tRu)dt \tag{6}$$

where $z(n\cdot 1)$ and $u(m\cdot 1)$ are the state and the control signal; $Q(n\cdot n)$ and $R(m\cdot m)$ are the weighting matrices; $T_f$ is the final time. The first term of integrand means the vibration energy and the second term means the control energy. Each term is non-dimensionalized by weighting matrices $Q$ and $R$. If the neuro-controller is trained by minimizing the cost function, there is no need to predetermine the desired response and both the response and the control signal are available at every instant. In addition, the neuro-controller can be trained optimally. If the mathematical model of the structure is available, the so-called Riccati equation can be derived with variational approach. Then, the optimal control gain can be found by solving the equation. However, if the model is not available, or has some error, or has nonlinearity, it is impossible to control the structure. Neuro-controller is powerful in this case, because it can control even the structure by training.

If the cost function is expressed in the discrete-time domain, it has the form of

$$\hat{J}=\frac{1}{2}\sum_{k=0}^{N_f-1}\{Z_{k+1}^TQz_{k+1}+u_k^TRu_k\}=\frac{1}{2}\sum_{k=0}^{N_f-1} \tag{7}$$

where $k$, $N_f$, and $T_s$ are the sampling number, the total number of sampling time, and the sampling interval, respectively. By applying gradient descent rule to the cost at $k$-th step, the update for the weight, $W_{ji}^2$, at the $k$-th step can be expressed as

$$\triangle W_{ji}^2=-\eta\frac{\partial\hat{J}_k}{\partial W_{ji}^2} \tag{8}$$

where $\eta$ is the rate of training. By varying the rate, the convergence of training can be improved. Using chain rule, the partial derivative of equation (8) can be rewritten as

$$\frac{\partial\hat{J}_k}{\partial W_{ji}^2}=\frac{\partial\hat{J}_k}{\partial net_j^2}\frac{\partial net_j^2}{\partial W_{ji}^2} \tag{9}$$

Then, let's define the generalized error as

$$\delta_j^2=-\frac{\partial\hat{J}_k}{\partial net_j^2}=-\frac{\partial\hat{J}_k}{\partial o_j^2}\frac{\partial o_j^2}{\partial net_j^2} \tag{10}$$

Finally, the weight update can be simply expressed as

$$\triangle W_{ji}^2=\eta\delta_j^2 o_i^1 \tag{11}$$

where

$$\delta_j^2=\left(Z_{k-1}^TQ\left\{\frac{\partial Z_{k+1}}{\partial u_{k,j}}\right\}+u_k^Tr_j\right)G_j(f^2)'\Big|_{net_j^2} \tag{12}$$

In equation (12), gain factor, $G_j$, satisfies

$$u_j=G_jo_j^2 \tag{13}$$

and $r_j$ is the $j$-th column vector of $R$. The bias is also updated by

$$\triangle b_j^2=\eta\delta_j^2 \tag{14}$$

In equation (12), all the terms except the sensitivity of the state to the control signal are available at the $k$-th step. And the sensitivity can be calculated through the sensitivity evaluation algorithm that will be described in the next section. In the same manner, update for the weight, $W_{ih}^1$, can be obtained as

$$\triangle W_{ih}^1 = \eta \delta_i^1 I_h \tag{15}$$

where

$$\delta_i^1 = -\frac{\partial \hat{f}_k}{\partial net_i^1} = -\sum_{j=1}^{n3} \frac{\partial \hat{f}_k}{\partial net_j^2} \frac{\partial net_j^2}{\partial o_i^1} \frac{\partial o_i^1}{\partial net_i^1}$$

$$= \sum_{j=1}^{n3} \delta_j^2 W_{ji}^2 (f^1)' \Big|_{net_i^1} \tag{16}$$

And the bias of the hidden layer is updated by

$$\triangle b_i^1 = \eta \delta_i^1 \tag{17}$$

The sensitivity of response to control signal can be easily obtained if mathematical model is available. When, however, the model is not available, it can be indirectly calculated in the conventional neuro-control method; i.e. it is obtained from the emulator neural network that is trained to predict the dynamic response of the structure. Therefore, long time required for emulator training is inevitable. In addition, since the emulator is not trained to predict the sensitivity but to predict the response, the indirect calculation may contain prediction error. Even if the error is negligible, it takes calculation time at every training step and may delay the control signal in on-line learning. To overcome these problems, an evaluation algorithm for sensitivity is proposed in what follows.

The equation of motion of linear system can be written as

$$\dot{Z} = A_Z + B_u \tag{18}$$

where $A(n \times n)$ and $B(n \times m)$ denote the system matrix and the location vector of the controller, respectively. By $z$-transform, equation (18) is transformed to

$$Z_{k+1} = Gz_k + Hu_k \tag{19}$$

where $G(n \times n)$ and $H(n \times m)$ are expressed in terms of $A$ and $B$ as

$$G = e^{AT_s} \tag{20}$$

$$H = (e^{ATs} - I)A^{-1}B \tag{21}$$

In linear system, it can be easily found that the matrix $H$ is the function of only the sampling time and that it is constant for a fixed sampling time. By differentiating equation (19) with respect to control signal at $k$-th step, the following sensitivity equation is obtained.

$$\frac{\partial z_{k+1}}{\partial u_k} = H \tag{22}$$

Equation (22) means that the matrix $H$ equals the response sensitivity to control signal. Therefore, if $H$ is found in some way, the emulator neural network becomes obsolete. $H$ can be simply found as follows. If all the initial states are set to zero, and unit control signal is applied only to the $i$-th controller, namely

$$z_k = 0 \tag{23}$$

$$u_{j,k} = \begin{cases} 1 & (\text{if } j \neq i) \\ 0 & (\text{if } j = i) \end{cases} \quad j = 1, 2, \cdots, m \tag{24}$$

Then, equation (19) becomes

$$z_{k+1} = h_i \tag{25}$$

where $h_i$ denotes the $i$-th column of $H$. By measuring the state at ($k+1$) step, the response sensitivity to control signal of $i$-th controller can be found. If another unit input signal is applied to the system, sensitivity vector corresponding to the controller is obtained. Therefore, the full matrix $H$ can be found. In conventional neuro-control, sensitivity is found from emulator neural network that requires much of time for training. It takes tens of minutes or hours. In this algorithm, however, the sensitivity is found in one sampling time. Therefore, time for obtaining sensitivity can be almost eliminated.

Displacement and velocity only after one sampling time are too small to measure with sensors. Therefore, it is recommended to measure acceleration instead of displacement and velocity. Then, displacement and velocity can be obtained by integrating the measured acceleration signal.

Although it is derived under linear condition, it can be used for the nonlinear system. Decreasing learning rate can compensate the sensitivity error due to nonlinearity. In addition, the controlled structural response of nonlinear system shows linear behavior as the training converges. Hence, the sensitivity data are useful in the neuro-control of nonlinear structures. This will be verified in numerical examples.

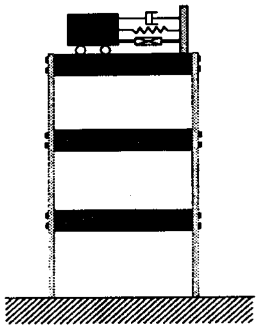


Figure 4. Shear building with active mass damper(AMD)

## STRUCTURE WITH ACTIVE MASS DAMPER SYSTEM

The structure of the reference[6] is slightly modified for numerical examples. An active tendon installed in the first floor controls the vibration of structure in the paper. In this research, however, an active mass damper (AMD) installed on the roof does. The hydraulic actuator is excited by the control signal generated from the neuro-controller. Then, the AMD attached to the actuator applies reaction force to the structure. Since the dynamics of the actuator and the structure are coupled, actuator dynamics should be included in analysis.

*Hydraulic actuator*

The equation of motion of hydraulic actuator can be divided in two parts, the valve dynamics and the piston equation. First, valve equation is expressed as

$$\frac{\tau}{g_1 g_2}\dot{q} + \frac{1}{g_1 g_2} q = u \tag{26}$$

where $g_1$ and $g_2$ denote the valve gains; $\tau$ the time constant of the valve; $q$ and $u$ the flow rate of oil and the control signal, respectively. The change of the oil flowing through the valve induces the motion of the piston and the relationship can be modeled by

$$a_r \dot{x}_r - \frac{c_l}{a_r} f + \frac{V}{2\beta a_r}\dot{f} = q \tag{27}$$

where $a_r$, $\beta$, $c_l$ and $V$ denote the area of the piston, the compressibility coefficient, leakage coefficient, and the volume of the cylinder respectively; $x_r$ the relative displacement between the roof and the piston; $f$ the force exerted on the structure by the AMD.

*Three-story shear building*

The equation of motion of the 3-story building with an AMD can be expressed as

$$M\ddot{x} + C\dot{x} + K(x, \dot{x}) = Lf - M[1]\ddot{x}_g \tag{28}$$

where $M$ and $C$ are $(4\times4)$ mass and damping matrices; $x$ is the $(4\times1)$ relative displacement vector consisting of three stories and an AMD; $K(x, \dot{x})$ is the $(4\times1)$ restoring-force vector; $L$ is the $(4\times1)$ vector indicating the location of the actuator; $\ddot{x}_g$ is ground acceleration; [1] is the direction vector of ground motion.

*Nonlinear dynamic model*

Nonlinear model proposed by Baber and Wen[15] is used to simulate the motion of nonlinear structure. The model has been used for the control simulation in many works[6, 9]. The restoring force of the model is composed of the linear and the nonlinear terms as

$$k_s(s_s, \dot{x}_s) = \alpha k_o x_s + (1-\alpha) k_o d y \tag{29}$$

where $x_s$ denotes the inter-story displacement; $k_o$ and are the linear stiffness and its contribution to restoring force, respectively. And $d$ and $y$ are the constant and the variable, respectively, satisfying the following equation.

$$\dot{y} = \frac{1}{d}(\rho \dot{x}_s - \mu|\dot{x}_s||y|^{p-1}y - \sigma \dot{x}_s|y|^p) \tag{30}$$

where $\rho$, $\mu$ and $\sigma$ are the constants that affect the hysteretic behavior.

*Time delay*

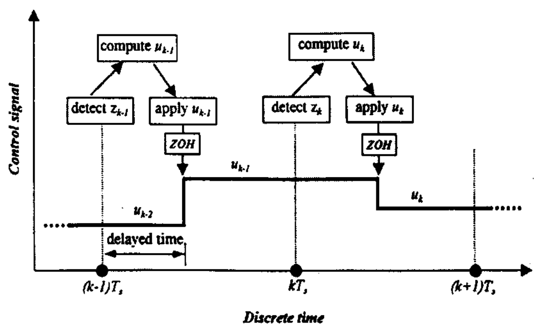



Figure 5. Time delay mechanism

Time delay is inevitable in the implementation of the controller. It is caused by the computation of the control signal that will be applied to the actuator. It is clearly shown in Figure 5. At the *k*-th step of sampling time, the state of the structure, $z_k$, is identified and used as a feedback signal to compute the control signal. Although it is quite short, the computation of the control signal takes time. Therefore, the control signal is not applied at the time $kT_s$ but at the time $kT_s$ plus the delayed time. If the time delay is not considered in the design, the performance of the controller may be worse than expected. Hence, the effect of time delay should be considered.

*Simulation parameters*

The structural properties are as follows: story mass equals 200kg; inter-story stiffness 2.25($10^5$N/m; damping ratios of three modes are 0.6, 0.7 and 0.3% respectively. The parameters of AMD are designed based on the suggestions[14] for optimal tuned mass damper (TMD). Mass of AMD is 18kg that corresponds to the 3% of the total mass of the structure. The actuator parameters are adopted from the reference[6] as follows: $g_1=32.14\times10^2$ and $g_2=2.8\text{m}^3/\text{sec}$; $\tau=0.1$; $a_r=1.52\times10^{-3}\text{m}^2$; $V=4.56\times10^{-4}\text{m}^3$; $c_l=1.0\times10^{-11}$; $\beta=2.1\times10^9$. For the linear structure $\alpha=1.0$ is used and the nonlinear structure $\alpha=0.5$. For nonlinear model $d$ equals 0.01; $\rho, \mu, \sigma$ and $p$ are 1.0, 0.5, 0.5 and 5.0 respectively. The sampling time is 0.005sec, delay time is assumed to 0.0005sec. The equation of motion is integrated at every 0.00025sec on Matlab 5.1[16].

*Sensitivity*

When the states of structure are all zero, 1.0volt control signal is applied to the actuator. Then, the acceleration response of the third floor is measured during one sampling time. The measured acceleration is shown in Figure 6. Random white noise with amplitude 0.1g is added to the measured signal. Then, velocity and displacement responses are obtained by integrating the measured acceleration. The sensitivities corresponding to the response after one sampling time are listed and compared with exact ones in Table 1. In the conventional method, sensitivities are obtained through emulator that needs much of time in training. However, it is obtained in one sampling time. That is, it takes almost no time in obtaining sensitivities.

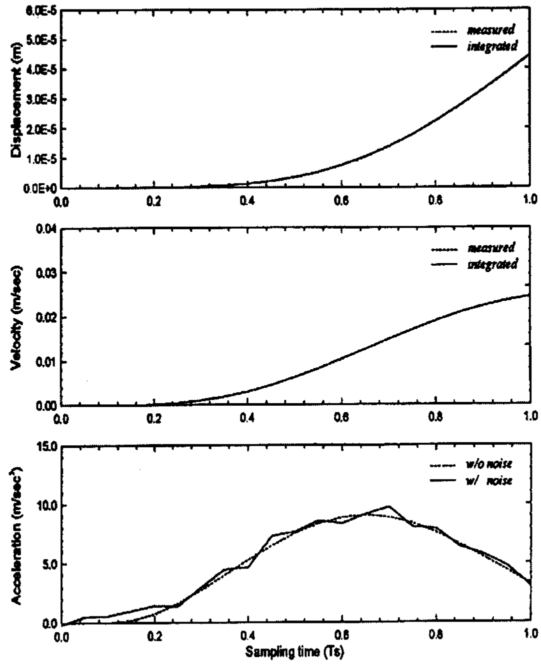



Figure 6 Structural response during one sample time

Table 1. Sensitivity evaluation results

| sensitivity | exact | Proposed |
|---|---|---|
| displacement (*m/volt*)<br>(ratio) | $4.433\times10^{-5}$<br>(1.00) | $4.436\times10^{-5}$<br>(1.001) |
| velocity (*m/sec/volt*)<br>(ratio) | $2.443\times10^{-2}$<br>(1.00) | $2.441\times10^{-2}$<br>(0.999) |

*Training the controller*

The neuro-controller has three layers: input layer, hidden layer, and output layer. The input layer has three nodes that will have the feedback signals of the displacement, velocity of the third floor and the ground acceleration. The hidden layer has three nodes. The output has only one node that will produce control signal. The sigmoid function is used as the activation function of the hidden layer and the linear function for the output layer. Only the state of the third floor normalized to the uncontrolled responses participates in the cost function. The cost at the $k$-th step used for training criterion is

$$\hat{J}_k = z^T_{3,k-1} Q z_{3,k-1} + r u_k^2 \qquad (31)$$

where $z3, k+1$ and $u_k$ denote the state of third floor and control signal, respectively. And weighting matrix $Q$ and $r$ are as follows.

$$Q = \begin{bmatrix} \left|\frac{1}{\tilde{x}_3}\right|^2 & 0.0 \\ 0.0 & \left|\frac{1}{\tilde{\dot{x}}_3}\right|^2 \end{bmatrix}, \quad r = 0.1\left|\frac{1}{\tilde{u}}\right|^2 \qquad (32), (33)$$

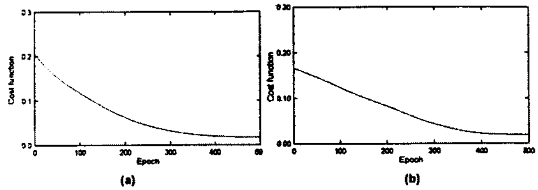
In these equations, $\tilde{x}_3$ and $\tilde{\dot{x}}_3$ are the maximum displacement and velocity of third floor under El Centro earthquake when control input is off. $\tilde{u}$ is the maximum control input voltage.

2000 time intervals among the dynamic responses under El Centro earthquake (1940), whose peak ground acceleration (PGA) is 0.348g, are used for the training data. This is the response during 10sec. Training was repeated until the number of epochs reaches 500. The history of cost functions for linear and nonlinear case is shown in Figure 7. The cost function for both the linear and the nonlinear structure converges slowly. In nonlinear case, training also converges within 500 epochs, though the sensitivity data found by the proposed algorithm are used.

*Control results*

The linear structure is controlled by the trained neuro-controller. Figure 8 shows control signal, AMD displacement and control force exerted on the structure by the AMD during El Centro earthquake. Controlled and uncontrolled displacements under the other two earthquakes, Northridge earthquake (PGA, 0.334g) and Kern County earthquake (PGA, 0.158g), are compared for each story in Figures 9 to 11. The response of each story is considerably reduced under control action. It should be noted that neuro-controller can reduce the response due to earthquakes which are not included in training.

The nonlinear structure is also controlled by the neuro-controller trained the nonlinear response under El Centro earthquake. Restoring force versus displacement relationships under Northridge earthquake are compared in Figures 12. It can be also concluded that the neuro-controller trained by the proposed method can control the nonlinear structure.




(a) (b)

Figure 7. Cost function vs. epoch: (a) linear case; (b) nonlinear case

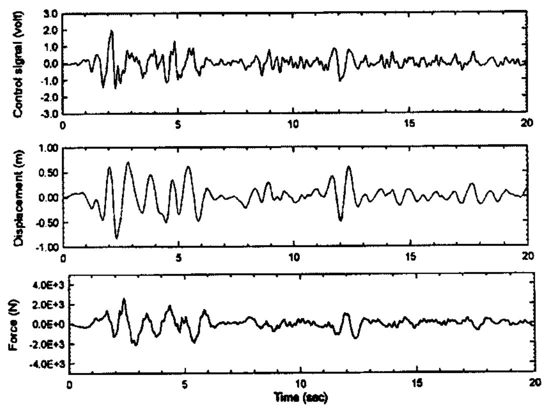



Figure 8. Control signal, AMD displacement, control force (El Centro earthquake)

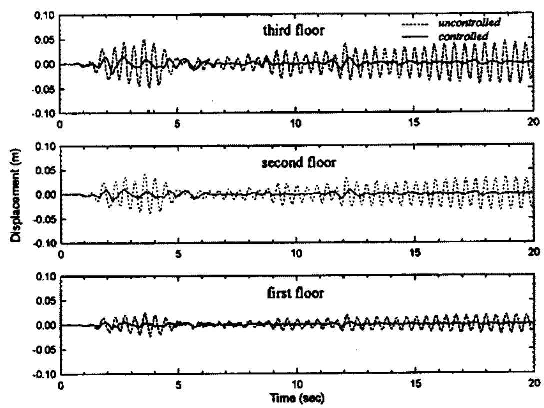



Figure 9. Displacement (El Centro earthquake)

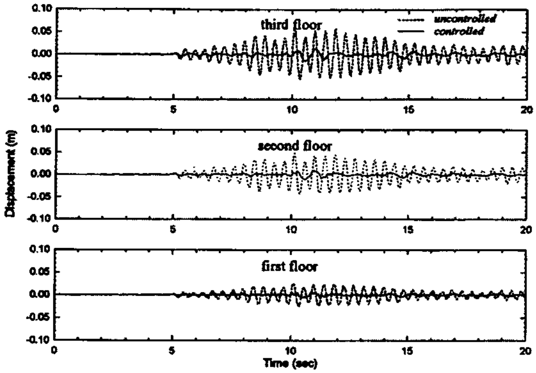



Figure 10. Displacement (Northridge earthquake)

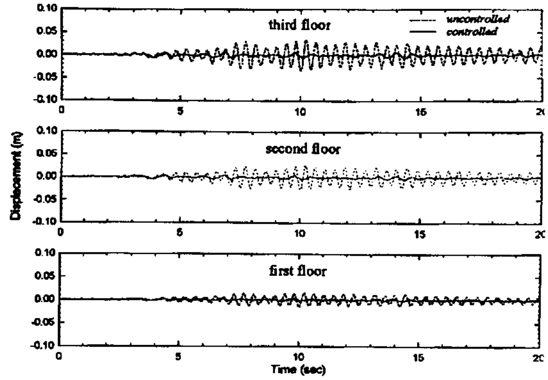



Figure 11. Displacement (Kern County earthquake)

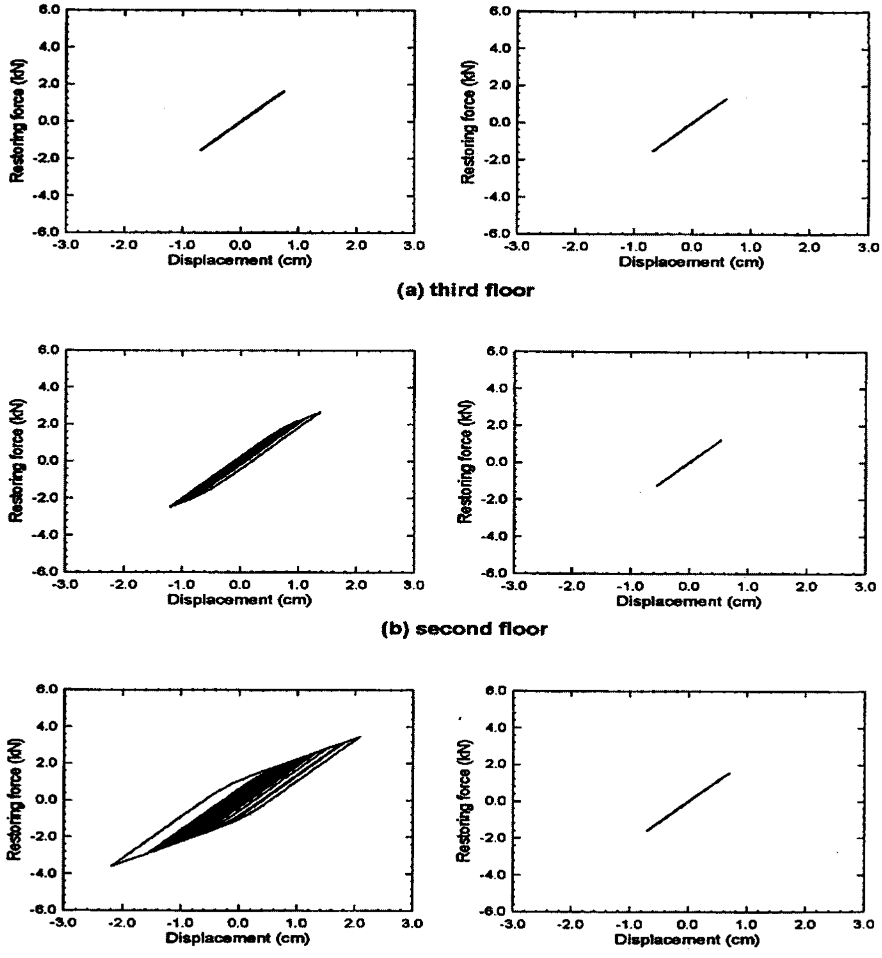



Figure 12. Restoring force vs. displacement (Northridge earthquake, left column-uncontrolled; right column-controlled)

## CONCLUSIONS

Improved neuro-control method is presented in this study. The training algorithm that was first proposed by J. T. Kim et al.[2, 3] is extended to the control of multi-degree-of freedom structure. Because the cost function is used as a training criterion, there is no need to pre-determine the desired response of structure. Only actual responses, control signals, and sensitivities of responses are used for training. The sensitivity has been found by the emulation of dynamics through trained neural network in the conventional method. It is, however, found by the sensitivity evaluation algorithm in the proposed method. Therefore, the time for training emulator neural network is eliminated. And only one neural network, neuro-controller, is used for neuro-control system.

The sensitivity is evaluated by assuming that the controlled structure is linear. But the sensitivity of nonlinear structure may not be significantly different from the evaluated result. And as the training converges, the nonlinear structure behaves linearly. Consequently, the obtained sensitivity can be used for the training of neural network in nonlinear structural control. It is verified in the numerical examples that the neuro-controller for nonlinear structure can be trained with the sensitivity found by the proposed method.

For the numerical examples, nonlinearity of stiffness, actuator dynamics, and time delay are considered. The neuro-controller successfully reduces the vibration induced by ground motions. Nonlinear hysteretic responses disappear when the control action is applied to the nonlinear structure. Finally, it is shown that neural network is a promising tool for the control of structures of which dynamics model is unknown or uncertain, and which have highly nonlinear properties.

## ACKNOWLEDGEMENT

This research was supported by the National Research Laboratory (NRL) program for Aseismic Control of Structures. The financial support is gratefully acknowledged.

## REFERENCES


1. D. H. Kim, 'Vibration control of structures using neural networks', Ph.D dissertation, Dept. of Civil Engineering, Korea Advanced Institute of Science & Technology (KAIST), Taejon, Korea, 2000.
2. J. T. Kim, H. J. Jung and I. W. Lee, Optimal structural control using neural networks', ASCE Journal of Engineering Mechanics, 126(2), 201-205 (2000).
3. J. T. Kim, J. W. Oh and I. W. Lee, Artificial Neural Networks for Structural Vibration Control', Proc. 1st Int. Conf. on Advances in Structural Engineering and Mechanics, Seoul, Korea, 23-25 August, 1999.
4. J. Ghaboussi, A. Joghataie, 'Active control of structure using neural networks', ASCE Journal of Engineering Mechanics, 121(4), 555-567 (1995).
5. J. N. Yang, Z. Li, S. Vongchavalitkul, 'Generalization of optimal control theory : linear and nonlinear control , ASCE Journal of Engineering Mechanics, 120(2), 266-283 (1994).
6. K. Bani-Hani, and J. Ghaboussi, 'Nonlinear structural control using neural networks', ASCE Journal of Engineering Mechanics, 124(3), 319-327 (1998).
7. K. Bani-Hani, and J. Ghaboussi, 'Neural networks for structural control of a benchmark problem, active tendon system', Earthquake Engineering & Structural Dynamics, 27, 1225-1245 (1998).
8. K. Nikzad, and J. Ghaboussi, 'Actuator dynamics and delay compensation using neurocontrollers', ASCE Journal of Engineering Mechanics, 122(10), 966-975 (1996).
9. H. Irschik, K. Schlacher and A. Kugi, 'Control of earthquake excited nonlinear structure using Liapunov's theory', Computers and Structures, 63, 83-90 (1998).
10. H. M. Chen, K. H. Tsai, G. Z. Qi, J. C. S. Yang, and F. Amini, 'Neural network for structural control', ASCE Journal of Computing in Civil Engineering, 9(2), 168-176 (1995).
11. P. Venini, 'Robust control of uncertain structures', Computers & Structures, 67, 165-174 (1998).
12. Y. He, J. Wu, 'Control of structural seismic response by self-recurrent neural networks (SRNN)', Earthquake Engineering and Structural Dynamics, 27, 641-648 (1998).
13. Y. Tang, 'Active control of SDF systems using artificial neural networks', Computer & Structures, 60(5), 695-703 (1996).
14. T. T. Soong and G. F. Dargush, Passive Energy Dissipation Systems in Structural Engineering, John Wiley & Sons, West Sussex, England, pp. 227-235 (1997).
15. T. T. Baber and Y. K. Wen, 'Random vibration of hysteretic degrading systems', ASCE Journal of Engineering Mechanics, 107(6), 1069-1087 (1981).
16. The Math Works, Inc., MATLAB, Natick, MA, 1997.