Computational Mechanics - New Frontiers for New Millennium
S. Valliappan and N. Khalili, editors.

NEURO-CONTROL OF STRUCTURES USING CMAC

SANG- WON CHO [1], D. H. KIM [2] AND IN-WON LEE [3]

[1,3]Department of Civil Engineering, Korea Advanced Institute of Science and Technology,
373-1 Kusong-Dong,Yusong-Gu, Taejon, 305-701 Korea
[2]Coastal & Harbor Engineering Laboratory, Korea Ocean Research & Development, 1270, Sa-dong,
Ansan, Kyunggi 425-744, Korea

E-mail: [1]wood@kaist.ac.kr, [2]eastlite@kordi.re.kr, [3]iwlee@kaist.ac.kr

ABSTRACT
Feasibility test for the application of Cerebeller Model Articulation Controller (CMAC) to vibration control is shown. CMAC is a kind of neural network known as having fast learning characteristics. Because the learning speed of MLP is very slow, it cannot be thought as a real-time controller. CMAC, however, can serve as a real-time controller. CMAC is trained as a controller and the convergence is compared with that of MLP. CMAC shows great possibility as a controller that learns very quickly. In the simulation study, the dynamics of hydraulic actuator and time delay of the controller are considered.

KEYWORDS
Neural networks, CMAC, Vibration control, Training

INTRODUCTION
Structural control has been one of the most frequent application areas of Artificial Neural Networks(ANNs) during last decade. ANNs have easily designed controller of structures with nonlinearity and uncertainty that had been challenging tasks of conventional control approaches. Learning property of ANNs make it possible to solve 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 ANNs needs not the mathematical model. Instead of the model, only the structural responses are used for design of ANN controller, simply called the neuro-controller.
A number of structural control algorithm using ANNs have been proposed by many researchers such as J. Ghaboussi et al., H. M. Chen et al., K. Nikzad et al., K. Bani-Hani et al., J. T. Kim et al. Multilayer neural network (MLNN) is used in all their works. It is the most widely-used type of neural network due to its simplicity of structure and learning algorithm. However, it has not been pointed out that the training convergence of MLNN is very slow. Hence, all of the control algorithms now proposed are based on off-line training. In result, one has to wait long for the end of training process before vibration control is applied by trained neuro-controller. But there are some needs for on-line learning

controller such as the control of damaged structures or time-varying structures. Therefore, a new training algorithms or even a new neural networks having fast learning convergence is required.

CEREELLAR MODEL ARTIULATION CONTROLLER(CMAC)

Moving Input Space

Each input space is first shifted to the right-half plane in real axis for the simple formulation of the addressing the activated memory. If x_i is the i-th element of input vector to CMAC, it can be done through

$$\bar{x}_i = x_i - x_{i,\min}\ ;\quad i = 1, 2, \cdots, \Omega \tag{1}$$

where $x_{i,\min}$ is the minimum of x_i, and Ω is the dimensionality of input space. Then, all the elements of input vectors have minimum values of zero and remain in positive domain.

Addressing Associate Memory (hash mapping)

When the generalization width is N_g and the dimensionality of input space is Ω, the indices of the activated weights in physical memory are calculated as

$$I_m = ceil\left(\frac{\bar{x}_1 - s_{m1}}{q_1}\right) + \sum_{n=2}^{\Omega}\left[ceil\left(\frac{\bar{x}_n - s_{mn}}{q_n}\right)\cdot\prod_{j=1}^{n-1}(b_j+1)\right] + I_m^*,\quad m = 1, 2, \cdots, N_g \tag{2}$$

where $ceil(y)$: the least integer greater than or equal to y,

s_{mn} : the amount of the shift of the n-th variable on the m-th mesh,

q_n : the quantization interval of the n-th variable,

b_j : the number of blocks covering the domain of the j-th variable,

I_m^* : the starting position of the address for the m-th mesh and can be expressed as

$$\begin{cases} I_{m-1}^* + \prod_{n=1}^{\Omega}(b_n+1), & (m \geq 2) \\ 1 & (m = 1) \end{cases} \tag{3}$$

Calculating Output and Training

The calculation of the output of CMAC is simple. The weights stored in the activated addresses are summed as

$$u = \sum_{m=1}^{N_g} w(I_m) \tag{4}$$

where $w(I_m)$ is the weight stored at the I_m-th location, N_g is the number of quantization mesh. To make an adequate output to a given input, the CMAC should be trained, i.e., the weights are updated so that the output is close to the desired one within allowable error. The training rule of CMAC as a controller will be described in the next chapter.

VIBRATION CONTROL UNSING CMAC

Control Algorithm and Training Rule

To train the CMAC neuro-controller, a cost function comprised of the structural response and the control signal is defined as

$$
\begin{aligned}
J &= \sum_{k=0}^{N_f-1} J_k \\
&= \sum_{k=0}^{N_f-1} \frac{1}{2}\left(\mathbf{z}_{k+1}^T \mathbf{Q} \mathbf{z}_{k+1} + \mathbf{u}_k^T \mathbf{R} \mathbf{u}_k\right)
\end{aligned}
\tag{5}
$$

where $\mathbf{z}(n\times 1)$ and $\mathbf{u}(m\times 1)$ are the state and the control signal; $\mathbf{Q}(n\times n)$ and $\mathbf{R}(m\times m)$ the weighting matrices; k, N_f sampling number, total number of sampling time, respectively. By applying the gradient descent rule to the cost at the k-th step (J_k), the weight update at the k-th step can be expressed as

$$\Delta w = \eta\delta \tag{6}$$

where

$$\delta = -\left(\mathbf{z}_{k+1}^T \mathbf{Q}\left[\frac{\partial \mathbf{z}_{k+1}}{\partial \mathbf{u}_k}\right] + \mathbf{u}_k^T \mathbf{R}\right) \tag{7}$$

The response sensitivity of Eq. (7) can be obtained through the sensitivity evaluation algorithm proposed by D. H. Kim et al.

Limitations of CMAC

Although CMAC quickly learns functional input-output relationships, it should be noticed that it has limitation in application. Since it stores information to a few localized memories, the size of memory is enormously increased in storing full information. In addition, since any one instance of training data is used to train only a few localized weights, the training data should be scattered as uniformly as possible for having generalization potential. This usually increases the size of training data. Therefore, the memory mapping and generalization algorithm should be developed to widen application areas of CMAC.

NUMERICAL EXAMPLE

Structure with Active Mass Damper (AMD) System

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

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}(\mathbf{x}, \dot{\mathbf{x}}) = \mathbf{L}f - \mathbf{M}[1]\ddot{x}_g \tag{8}$$

where $\mathbf{M}$ and $\mathbf{C}$ are (4×4) mass and damping matrices; $\mathbf{x}$ is the (4×1) relative displacement vector consisting of three stories and an AMD; $\mathbf{K}(\mathbf{x}, \dot{\mathbf{x}})$ is the (4×1) restoring-force vector; $\mathbf{L}$ is the (4×1) 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 is used to simulate the motion of nonlinear structure. The model has been used for the control simulation in many works. The restoring force of the model is composed of the linear and the nonlinear terms as

$$k_s(x_s, \dot{x}_s) = \alpha k_0 x_s + (1-\alpha) k_0 d y \tag{9}$$

where x_s denotes the inter-story displacement; k_0 and α 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.

$$\dot{y} = \frac{1}{d}(\rho \dot{x}_s - \mu |\dot{x}_s| |y|^{p-1} y - \sigma \dot{x}_s |y|^p) \tag{10}$$

where ρ, μ and σ are the constants that affect the hysteretic behavior.

Training CMAC

To train a CMAC, cost function of Eq. 5 is defined. 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

$$J_k = \mathbf{z}_{3,k+1}^T \mathbf{Q} \mathbf{z}_{3,k+1} + r u_k^2 \tag{11}$$

where $\mathbf{z}_{3,k+1}$ and u_k denote the state of third floor and control signal, respectively. And weighting matrix $\mathbf{Q}$ and r are as follows.

$$\mathbf{Q} = \begin{bmatrix} \left|\frac{1}{\tilde{x}_3}\right|^2 & 0.0 \\ 0.0 & \left|\frac{1}{\tilde{\dot{x}}_3}\right|^2 \end{bmatrix}, \quad r = 0.1\left|\frac{1}{\tilde{u}}\right|^2 \tag{12), (13}$$

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.

CMAC has two inputs, relative displacement and velocity of the third floor. Each input space has two quantization blocks. And the generalization width is 200. Therefore, the total number of weights, N_w, equals $200\prod_{i=1}^{2}(2+1) = 1800$.

CMAC was trained during 10sec ground motion of El Centro earthquake. After 500 epochs the final costs and epochs for CMAC and MLNN are compared in Table 1. The cost for CMAC is slightly lager than that for MLNN. Epochs for both linear and nonlinear cases are dramatically reduced.

Control Results

Figure 1. shows the response of controlled and uncontrolled structure at the third floor. And figure 2. shows the restoring force to displacement relation between base and second floor. After control action, vibration of structure was successfully reduced.

CONCLUSIONS

CMAC is exploited for the control of structural vibration. The structure and training algorithm of CMAC is presented. The training convergence of CMAC is quite fast compared with MLNN. In the numerical example, when CMAC is used for controller, it takes only 15% (linear case) and 8% (nonlinear case) of the epoch taken by MLNN for cost function to reach below the same values. It is expected, therefore, that CMAC can be used for structural control.

Table 1. Cost and epoch after training

Network	Linear (α=1.0)		Nonlinear (α=0.5)	
	J_{min}	Epoch	J_{min}	epoch
MLNN	1.77×10^{-2} (1.00)	412 (1.00)	1.91×10^{-2} (1.00)	427 (1.00)
CMAC	1.94×10^{-2} (1.09)	65 (0.15)	2.02×10^{-2} (1.00)	34 (0.08)

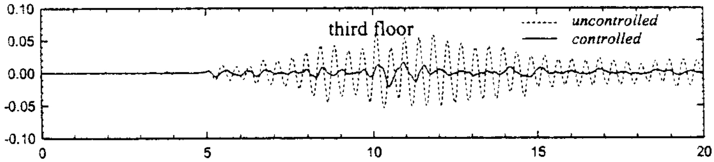

Figure 1. Displacement under Northridge earthquake

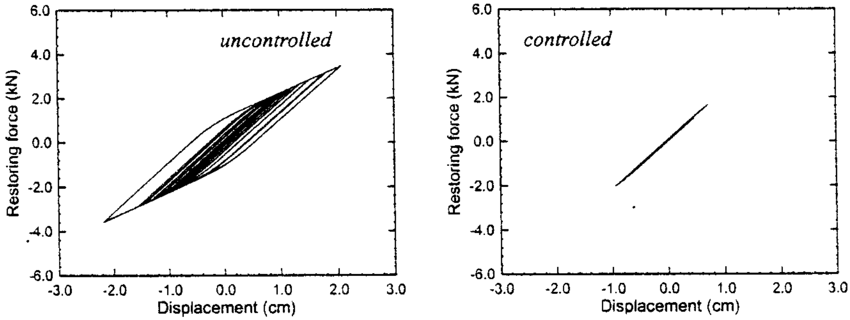

Figure 2. Restoring force vs. displacement under Northridge earthquake

ACKNOWLEDGEMENT

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

References

Albus, J.S (1975), "A new approach to manipulator control : the cerebellar model articulation controller," *ASME Trans. on Dynamic Systems Measurements & Control*, 97, **220-227**.

Chen, H.M., K.H. Tsai, G.Z. Qi, J.C.S. Yang, and F. Amini (1995), "Neural network for structural control," *Journal of Computing in Civil Engineering, ASCE*, **9(2), 168-176**.

Ghaboussi, J. and A. Joghataie (1995), "Active control of structure using neural networks," *Journal of Engineering Mechanics, ASCE,* **121(4), 555-567**.

Kim, J.T., H.J. Jung, and I.W. Lee (2000), "Optimal structural control using neural networks," *Journal of Engineering Mechanics, ASCE*, **126(2), 201-205**.

Kim, D.H., C.B. Yun, and I.W. Lee (2000), "Vibration control of steel structure using neuro-controller trained with sensitivity data," *Proceedings of Korena Society of Steel Construction*, Seoul, Jun. 3, 2000, **328-333**.