Paper No. 1266. (quote when citing this article)
Eleventh World Conference on Earthquake Engineering
ISBN: 0 08 042822 3

ACTIVE VIBRATION CONTROL OF HYSTERETIC OSCILLATORS USING NEURAL NETWORKS

H.I. HANSEN, S.R.K. NIELSEN & P. THOFT-CHRISTENSEN

Department of Building Technology and Structural Engineering, Aalborg University, Sohngaardsholmsvej 57, DK-9000 Aalborg, Denmark.

ABSTRACT

This paper deals with active vibration control of single degree-of-freedom (SDOF) hysteretic oscillators, where it is assumed that only the displacement and the velocity of the oscillator may be observed whereas the hysteretic component is hidden. Hence, Incomplete State Observation is assumed. A partially recurrent neural network, structured as a Multi Layer Perceptron (MLP) neural network is trained to estimate the response of the oscillator. The identified model is the well-known Innovation State Space Model and the identification is based on known measurements of the loading, the displacement and the velocity, so in fact the Extended Kalman Filter problem is solved. The closed-loop controller is also modelled as a MLP partially recurrent neural network, which is trained in a way that the mean square of the responses of the oscillator is minimized. The method has been applied to a Bouc-Wen oscillator subjected to Gaussian white noise filtered through a Kanai-Tajimi filter. The output of the filter is assumed to be observed and forms the input to the system.

KEY WORDS

Hysteretic oscillators, incomplete state observation, innovation state space model, active closed-loop control, neural networks.

1 INTRODUCTION

The idea of using automatic control systems to reduce the response of civil engineering structures has been actively pursued by several researchers since it was proposed by Yao (1972). Much of the theory applied by civil engineering researches is rooted in classical and modern control techniques, which basically have been developed within the aerospace, electrical and mechanical fields, see e.g. Soong (1988).

The actual control process is initiated by sensors placed at strategic points on the structure. These sensors measure the appropriate structural response, and this information is then used to compute the feedback control forces. One well-known technique for state space estimation for control purpose is the Extended Kalman Filtering (EKF) technique, see e.g. Thesbjerg (1992). Although the EKF has been successfully used in applications, it is known that the EKF might show poor convergence properties in certain cases. The estimates may be biased and even divergence may occur. Further, the computation time required for convergence, if at all, increases quadratic with the number of degree-of-freedom of a structural system. However, the main problem with the state space estimation techniques is the extensive computing which is necessary to convert the raw measured response data to control forces, i.e. real-time control does not seem to be feasible and cost effective due to the computing time and resources to individual structures. It should also be mentioned that one has to do assumptions when using the traditional structural control algorithms. The assumptions that limit the accuracy of the solution are that the real structure, material, sensors and actuators do not behave exactly as their mathematical model, i.e. that the model is controlled in stead of the real structure.

In order to overcome the mentioned problems much research has been done in the field of active control based on artificial neural networks. Artificial neural networks have been investigated for control of complex nonlinear systems with applications to model the nonlinear behaviour of the system, from which a control algorithm can be developed, or through learning, to develop and represent the control algorithm. A survey of artificial neural networks in the realm of modelling, identification and control of nonlinear systems is given in Hunt *et al.* (1992). Artificial neural networks have been used for process control in different fields such as biological and electrical systems, see e.g. Hunt *et al.* (1992). However, in the area of structural control only few investigations have been done. Doo *et al.* (1989) use an artificial neural network control algorithm to suppress vibrations in linear single degree-of-freedom systems subjected to dynamic loadings. Miccoli *et al.* (1994) estimate the state in the problems of noise and vibration active control using artificial neural networks. In Rehak and Garrett (1992) the applicability and use of neural computing for structural control is discussed.

The aim of this paper is to investigate the possibility of using artificial neural networks to estimate the response of a single degree-of-freedom hysteretic oscillator for incomplete state observations. The ability of the neural network to act as a simulator is investigated. Further, the performance of the neural network controller is investigated for incomplete state observation. The response of a single degree-of-freedom hysteretic oscillator is described in Section 2. In Section 3 the principle of the Multi Layer Perceptron neural networks used to estimate the response of the oscillator and to estimate the control force is shown. Neural network models of the hysteretic oscillator and the closed-loop controller are formulated in Section 4 and 5, respectively. As an example in Section 6 the neural network models are applied to a Bouc-Wen oscillator subjected to Gaussian white noise filtered through a Kanai-Tajimi filter. Conclusions and references are given in Section 7 and 9, respectively.

2 SINGLE DEGREE-OF-FREEDOM HYSTERETIC OSCILLATOR

A linear single degree-of-freedom hysteretic oscillator is described by the stochastic differential equation, see e.g. Suzuki and Minai (1985)

$$\ddot{X}(t) + 2\zeta\omega_0\dot{X}(t) + \omega_0^2\left(\alpha X(t) + (1-\alpha)Z(t)\right) = F(t) \tag{1}$$

$X(t)$ is the displacement, $\dot{X}(t)$ is the velocity, $\ddot{X}(t)$ is the acceleration and $F(t)$ is the excitation. ζ is the damping ratio and ω_0 is the circular eigenfrequency of the corresponding linear oscillator. α is the elastic fraction of the restoring force. $Z(t)$ is the hysteretic component of the restoring force, which is assumed to be modelled by the Bouc-Wen model, see Wen (1976)

$$\dot{Z}(t) = g\left(\dot{X}(t), Z(t)\right) = \left(1 - \beta\,\mathrm{sign}(\dot{X}(t))\frac{|Z(t)|^{n-1}Z(t)}{z_0^n} - \gamma\frac{|Z(t)|^n}{z_0^n}\right)\dot{X}(t) \tag{2}$$

n is a parameter, which controls the level of non-linearity. If $\beta + \gamma = 1$, z_0 can be interpreted as the yield displacement of the oscillator.

(1) and (2) can be written on state vector form

$$\dot{\boldsymbol{X}}(t) = \boldsymbol{a}(\boldsymbol{X}(t)) + \boldsymbol{b}F(t), \quad t > 0, \quad \boldsymbol{X}(0) = \boldsymbol{0} \tag{3}$$

$$\boldsymbol{X} = \begin{bmatrix} X \\ \dot{X} \\ Z \end{bmatrix}, \quad \boldsymbol{a}(\boldsymbol{X}) = \begin{bmatrix} \dot{X} \\ -2\zeta\omega_0\dot{X} - \omega_0^2\left(\alpha X + (1-\alpha)Z\right) \\ g(X,Z) \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tag{4}$$

Normally, the state variables $X(t)$ and $\dot{X}(t)$ can be observed, whereas $Z(t)$ is non-observable (hidden). Assuming the excitation $F(t)$ is constant at the value $F_{k-1} = F(t_{k-1})$ throughout the interval $[t_{k-1}, t_k[$, where $t_k = k\Delta t$, (3) can then be integrated on the following form

$$\boldsymbol{X}_k = \boldsymbol{\mathcal{F}}(\boldsymbol{X}_{k-1}, F_{k-1}) \tag{5}$$

where $\boldsymbol{X}_k = \boldsymbol{X}(t_k)$. The vector function $\boldsymbol{\mathcal{F}}(\boldsymbol{X}_{k-1}, F_{k-1})$ is a non-linear and non-analytical function of $\boldsymbol{X}_{k-1}$ because of the non-analytical constitutive relation (2).

A neural network model of (5) involves in principle a replacement of $\boldsymbol{\mathcal{F}}(\boldsymbol{X}_{k-1}, F_{k-1})$ by an analytical function. It is one of the aims of this paper to investigate the limits of such an approximation.

3 NEURAL NETWORKS

The neural networks are structured as Multi Layer Perceptron (MLP) neural networks containing:

- An input layer of n_I neurons with linear neuron functions without offsets.
- A fictive input layer with one neuron with constant value '1'.
- A hidden layer with a sufficient number of neurons n_H with non-linear neuron functions including offsets. The offsets are organized by adding the fictive input neuron. The non-linear function used is the tanh-function. The necessary number of hidden neurons can be found by trial and error.
- An output layer of n_O neurons with linear neuron functions without offsets.

There are connections between the n_I input neurons and all hidden neurons, with weights w_{hi}. In the same way there are connections between all hidden neurons and all output neurons, with weights w_{oh}. Further, there are connections between the fictive input neuron and all hidden neurons, with weights w_{h0}.

The output of the network $O_1, O_2, \ldots, O_{n_O}$ can be calculated by feedforward based on the weights and the input to the network $I_1, I_2, \ldots, I_{n_I}$. The output O_o of the o'th neuron in the output layer is

$$O_o = \sum_{h=1}^{n_H} w_{oh} \tanh\left(w_{h0} + \sum_{i=1}^{n_I} w_{hi} I_i\right) \tag{6}$$

4 MODELLING OF HYSTERETIC OSCILLATOR BY NEURAL NETWORKS

A neural network model of the hysteretic oscillator described by (5) is formulated by a Non-linear Innovation State Space Model where p state variables are observed, see Sørensen (1994)

$$\hat{\boldsymbol{X}}(k) = \boldsymbol{\mathcal{F}}\left(\hat{\boldsymbol{X}}(k-1), F(k-1), \boldsymbol{E}(k-1), \boldsymbol{w}\right) \tag{7}$$

$$\boldsymbol{Y}(k) = \boldsymbol{H}\hat{\boldsymbol{X}}(k) + \boldsymbol{E}(k) \tag{8}$$

$\hat{\boldsymbol{X}}(k)$ is the estimate of the state vector at time step k and $\boldsymbol{E}(k)$ is the prediction error vector of order p at time step k. $\boldsymbol{w}$ is a vector of weights in the MLP network. $\boldsymbol{Y}(k)$ is the measured observation vector of order p at time step k. $\boldsymbol{H}$ is the observation matrix of order $p \times 3$.

Incomplete State Information may occur, i.e., $\hat{\boldsymbol{X}}$ is not completely measurable. The matrix $\boldsymbol{H}$ can be chosen to $\boldsymbol{H} = [\mathbf{1}_{p,p}\mathbf{0}_{p,3-p}]$, where $\mathbf{1}_{p,p}$ is a $p \times p$ unity matrix and $\mathbf{0}_{p,3-p}$ is a $p \times (3-p)$ zero matrix. In this way the elements in $\boldsymbol{H}\hat{\boldsymbol{X}}(k)$ are equal to the first p elements of $\hat{\boldsymbol{X}}(k)$. For the hysteretic oscillator the number of observed states p can then be $p = 1$ if only the displacement $X(k)$ is observed, $p = 2$ if the displacement $X(k)$ and the velocity $\dot{X}(k)$ are observed and $p = 3$ if all three state variables $X(t)$, $\dot{X}(k)$ and $Z(t)$ are observed. The net is a partially recurrent network as output from the net is used as input in the next step.

4.1 Training of Neural Network

The weights $\boldsymbol{w}$ in the neural network $\boldsymbol{\mathcal{F}}$ given by (7) are found by training. The training set consists of N input time series, each of length K. Feedforward as described by (6) is used to estimate the output of the network. There are several methods for updating the weights $\boldsymbol{w}$, see e.g. Hertz *et al.* (1991).

Let $\boldsymbol{E}(n, k, \boldsymbol{w})$ be the error vector between the estimates and measurements of the nth training time series at the kth time step. The following performance index $J(\boldsymbol{w})$ is minimized

$$J(\boldsymbol{w}) = \frac{1}{2} \sum_{n=1}^{N} \sum_{k=1}^{K} \boldsymbol{E}^T(n, k, \boldsymbol{w}) \boldsymbol{E}(n, k, \boldsymbol{w}) \tag{9}$$

The gradient of $J(\boldsymbol{w})$ with respect to the weights $\boldsymbol{w}$ is needed as search direction in the iteration scheme. This is calculated from

$$\frac{dJ(\boldsymbol{w})}{d\boldsymbol{w}^T} = -\sum_{n=1}^{N} \sum_{k=1}^{K} \boldsymbol{E}^T(n, k, \boldsymbol{w}) \boldsymbol{H} \frac{d\hat{\boldsymbol{X}}(n, k, \boldsymbol{w})}{d\boldsymbol{w}^T} \tag{10}$$

The gradient of the estimated state variables $\hat{\boldsymbol{X}}(n, k, \boldsymbol{w})$ with respect to the weights $\boldsymbol{w}$ can be updated from the recursive equation

$$\boldsymbol{\psi}(n, k) = \frac{d\hat{\boldsymbol{X}}(n, k, \boldsymbol{w})}{d\boldsymbol{w}^T} = \boldsymbol{\varphi}(n, k) + \left[\hat{\boldsymbol{\Phi}}(n, k) - \hat{\boldsymbol{K}}(n, k)\boldsymbol{H}\right] \boldsymbol{\psi}(n, k-1) \tag{11}$$

where

$$\boldsymbol{\varphi}(n, k) = \frac{\partial \hat{\boldsymbol{X}}(n, k, \boldsymbol{w})}{\partial \boldsymbol{w}^T}, \quad \hat{\boldsymbol{\Phi}}(n, k) = \frac{\partial \hat{\boldsymbol{X}}(n, k, \boldsymbol{w})}{\partial \hat{\boldsymbol{X}}^T(n, k-1, \boldsymbol{w})}, \quad \hat{\boldsymbol{K}}(n, k) = \frac{\partial \hat{\boldsymbol{X}}(n, k, \boldsymbol{w})}{\partial \boldsymbol{E}^T(n, k-1, \boldsymbol{w})} \tag{12}$$

$\boldsymbol{\varphi}(n, k)$, $\hat{\boldsymbol{\Phi}}(n, k)$ and $\hat{\boldsymbol{K}}(n, k)$ can be found by differentiation of the output of the network in (6) with respect to the weights and input. $\hat{\boldsymbol{\Phi}}(n, k)$ and $\hat{\boldsymbol{K}}(n, k)$ can be interpreted as the dynamic transfer matrix of an equivalent linear system and the corresponding extended Kalman filter gain, respectively.

4.2 Simulation and State Prediction by Trained Neural Network

When the trained neural network $\boldsymbol{\mathcal{F}}$ given by (7) is used as a simulator only the initial values $\boldsymbol{Y}(0) = \boldsymbol{H}\boldsymbol{X}(0)$ of the measurable state variables and the complete excitation time series $F(k)$ are measured without any error. The non-observed state variables are set to $\boldsymbol{0}$ to obtain the initial values $\boldsymbol{X}(0) = \boldsymbol{X}_0$. As no further measurements of the state variables are made the prediction error vector $\boldsymbol{E}(k)$ is fixed to $\boldsymbol{0}$. The state variables in the simulator neural network are predicted by

$$\hat{\boldsymbol{X}}(k) = \boldsymbol{\mathcal{F}}\left(\hat{\boldsymbol{X}}(k-1), F(k-1), \boldsymbol{E}(k-1), \boldsymbol{w}\right), \quad \boldsymbol{X}(0) = \boldsymbol{X}_0, \quad \boldsymbol{E}(k-1) = \boldsymbol{0} \tag{13}$$

When the trained neural network is used as a one-step ahead predictor the measurable state variables $\boldsymbol{Y}(k) = \boldsymbol{H}\boldsymbol{X}(k)$ and the excitation $F(k)$ are measured at every time step. In this case the state variables are predicted by

$$\hat{\boldsymbol{X}}(k) = \boldsymbol{\mathcal{F}}\left(\hat{\boldsymbol{X}}(k-1), F(k-1), \boldsymbol{E}(k-1), \boldsymbol{w}\right), \quad \boldsymbol{E}(k-1) = \boldsymbol{Y}(k-1) - \boldsymbol{H}\hat{\boldsymbol{X}}(k-1) \tag{14}$$

5 MODELLING OF ACTIVE CONTROLLER BY NEURAL NETWORKS

After the model of the hysteretic oscillator (7) has been settled, an active closed-loop controller is coupled to the system. The control force $C(k)$ is used to minimize the mean square of the observable part of the response $\boldsymbol{X}(k)$ of the hysteretic oscillator. It is assumed that the total excitation of the oscillator can be written $F(k)+C(k)$. In the final active closed-loop control configuration the neural network model of the hysteretic oscillator is used as a one-step ahead predictor to estimate the state vector to the next time step. The control force can then be estimated by the controller network based on the estimated state variables.

Because no state variables can be measured during training of the controller neural network the trained network $\boldsymbol{\mathcal{F}}$ given by (7) is used as a simulator to estimate the response of the controlled oscillator $\hat{\boldsymbol{X}}^T(k)=[\hat{X}(k),\hat{\dot{X}}(k),\hat{Z}(k)]$. Hence

$$\hat{\boldsymbol{X}}(k)=\boldsymbol{\mathcal{F}}\left(\hat{\boldsymbol{X}}(k-1),F(k-1)+C(k-1),\boldsymbol{E}(k-1),\boldsymbol{w}\right),\quad \boldsymbol{E}(k-1)=\boldsymbol{0} \tag{15}$$

The control force $C(k)$ is modelled by a MLP neural network by using closed-loop control

$$C(k)=\mathcal{G}\left(\hat{\boldsymbol{Y}}(k),\boldsymbol{w}_c\right),\quad \hat{\boldsymbol{Y}}(k)=\boldsymbol{H}\hat{\boldsymbol{X}}(k) \tag{16}$$

where $\boldsymbol{w}_c$ is a vector of weights in the network modelling the controller. Obviously, the feedback part of the control law can only depend on the state variables $\hat{\boldsymbol{Y}}(k)=\boldsymbol{H}\hat{\boldsymbol{X}}(k)$ which are measured (observed), and no noise term is present in the model.

The weights $\boldsymbol{w}_c$ of (16) are estimated, so the response predicted by (15) becomes minimum. At the training the same input time series $F(k)$ are used as applied at the training of the neural network (7).

5.1 *Training of Neural Network*

The following performance index, which is widely used in structural control, see e.g. Soong (1990), is minimized for the controller network (16)

$$J_c(\boldsymbol{w}_c)=\frac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{K}\left(\hat{\boldsymbol{Y}}^T(n,k,\boldsymbol{w}_c)\hat{\boldsymbol{Y}}(n,k,\boldsymbol{w}_c)+aC^2(n,k,\boldsymbol{w}_c)\right) \tag{17}$$

$$\hat{\boldsymbol{Y}}(n,k,\boldsymbol{w}_c)=\boldsymbol{H}\hat{\boldsymbol{X}}(n,k,\boldsymbol{w}_c) \tag{18}$$

where a is a positive scaling factor and $\hat{\boldsymbol{X}}(n,k,\boldsymbol{w}_c)$ specifies the output of (15) for the nth time series at the time step k using the controller (16) with the weights $\boldsymbol{w}_c$.

The gradient of $J_c(\boldsymbol{w}_c)$ with regard to the weights $\boldsymbol{w}_c$ is

$$\frac{dJ_c(\boldsymbol{w}_c)}{d\boldsymbol{w}_c^T}=\sum_{n=1}^{N}\sum_{k=1}^{K}\left[\hat{\boldsymbol{Y}}^T(n,k,\boldsymbol{w}_c)\boldsymbol{H}\frac{d\hat{\boldsymbol{X}}(n,k,\boldsymbol{w}_c)}{d\boldsymbol{w}_c^T}+aC(n,k,\boldsymbol{w}_c)\frac{dC(n,k,\boldsymbol{w}_c)}{d\boldsymbol{w}_c^T}\right] \tag{19}$$

The gradient of the estimated state variables $\hat{\boldsymbol{X}}(n,k,\boldsymbol{w}_c)$ and the control force $C(n,k,\boldsymbol{w}_c)$ with respect to the weights organized in the vector $\boldsymbol{w}_c$ can be updated from the recursive equations, cf. (11)

$$\boldsymbol{\psi}_c(n,k)=\frac{d\hat{\boldsymbol{X}}(n,k,\boldsymbol{w}_c)}{d\boldsymbol{w}_c^T}=\hat{\boldsymbol{\Phi}}(n,k)\boldsymbol{\psi}_c(n,k-1)+\hat{\boldsymbol{\Gamma}}(n,k)\boldsymbol{v}_c(n,k-1) \tag{20}$$

$$\boldsymbol{v}_c(n,k)=\frac{dC(n,k,\boldsymbol{w}_c)}{d\boldsymbol{w}_c^T}=\boldsymbol{\varphi}_c(n,k)+\hat{\boldsymbol{\Phi}}_c(n,k)\boldsymbol{H}\boldsymbol{\psi}_c(n,k) \tag{21}$$

where

$$\hat{\boldsymbol{\Gamma}}(n,k)=\frac{\partial\hat{\boldsymbol{X}}(n,k,\boldsymbol{w}_c)}{\partial C(n,k-1,\boldsymbol{w}_c)},\quad \boldsymbol{\varphi}_c(n,k)=\frac{\partial C(n,k,\boldsymbol{w}_c)}{\partial\boldsymbol{w}_c^T},\quad \hat{\boldsymbol{\Phi}}_c(n,k)=\frac{\partial C(n,k,\boldsymbol{w}_c)}{\partial\hat{\boldsymbol{Y}}^T(n,k,\boldsymbol{w}_c)} \tag{22}$$

As seen the gradient is no longer dependent on variations of the observation noise, because this has been fixed to $\boldsymbol{E}=\boldsymbol{0}$ during the training phase.

Corresponding relations $F(n,k)$ and $\boldsymbol{X}(n,k)$ between the excitation process and the response process are obtained by numerical integration of (3). Realizations of the excitation process are obtained by filtering non-stationary white noise through a Kanai-Tajimi filter. The excitation process is then obtained from the stochastic differential equations, see Tajimi (1960)

$$F(t) = 2\zeta_s \omega_s \dot{U}(t) + \omega_s^2 U(t) \tag{23}$$

$$\ddot{U}(t) + 2\zeta_s \omega_s \dot{U}(t) + \omega_s^2 U(t) = -r(t)W(t) \tag{24}$$

$F(t)$ can then be interpreted as the negative of the ground surface acceleration, and ζ_s and ω_s are the damping ratio and circular eigenfrequency of a single degree-of-freedom shear model of the underlying subsoil. $\{W(t),\ t \in]0,\infty[\}$ is unit intensity white noise with the auto-spectral density function $\frac{1}{2\pi}$. The deterministic modulation function is given as follows, see Saragoni and Hart (1974)

$$r(t) = r_0 \exp\left(-\ln(2)\frac{\frac{t}{t_1} - \ln\left(\frac{t}{t_1}\right) - 1}{\frac{t_2}{t_1} - \ln\left(\frac{t_2}{t_1}\right) - 1}\right) \tag{25}$$

The amplitude r_0 defines the strength of the excitation. This is selected so the peak value of all excitations is normalized to $0.5\,g$, where g is the acceleration of gravity. t_1 and t_2 are respectively the instants of time of maximum intensity and the time where the intensity has dropped to half value. The following parameters are used for the hysteretic oscillator:

$$\zeta = 0.01, \quad \omega_0 = 2\pi \text{ s}^{-1}, \quad \alpha = 0.05, \quad \beta = \gamma = 0.5, \quad n = 1, \quad z_0 = 0.01 \text{ m}$$

$$\zeta_s = 0.5, \quad \omega_s = 30 \text{ s}^{-1}, \quad t_1 = 3 \text{ s}, \quad t_2 = 15 \text{ s}$$

Three input-output time series are generated. The first two of these, labeled A and B are for training. The last one labeled C is selected for verification. The displacement responses of cases A, B and C are classified as small, large and medium. The minimum and maximum of the displacement x, the velocity $\dot{x}$ and the hysteretic component z are shown in Table 1.

Table 1: Minimum and maximum values for data series used for training and verification.

Series	$\min(x)$ [m]	$\max(x)$ [m]	$\min(\dot{x})$ [m/s]	$\max(\dot{x})$ [m/s]	$\min(z)$ [m]	$\max(z)$ [m]
A	−0.044	0.043	−0.342	0.348	−0.033	0.033
B	−0.101	0.079	−0.588	0.376	−0.047	0.041
C	−0.080	0.054	−0.508	0.487	−0.043	0.041

In this example the Broyden-Fletcher-Goldfarb-Shanno variant of the Davidon-Fletcher-Powell minimization algorithm, see Press *et al.* (1988), is used to minimize the performance indices $J(\boldsymbol{w})$ and $J_c(\boldsymbol{w}_c)$ with respect to the weights $\boldsymbol{w}$ and $\boldsymbol{w}_c$. As stop criteria the performance index for case C is evaluated. When this index increases the network is overtrained. So the optimal neural network weights are obtained when the performance index has minimum for case C.

During training of the neural networks the input variables are scaled so the mean value of each variable is '0' and the standard deviation is '1'. Thereby, the components of the performance indices are of the same properties.

A MLP neural network is trained with the two training cases A and B. During training incomplete state observation is assumed with $p = 2$, i.e. the displacement and velocity are observed and the hysteretic component is hidden. The ability of the trained neural network to act as a simulator is shown in

Fig. 1 where the displacement for case C (with medium displacements) is shown. As seen the trained neural network is capable of approximating the non-linear and non-analytical vector function (5) very well when it as taken into account that only two of the three state variables are observed and that the neural network is trained with two time series with smaller and larger displacements, respectively.

Next, active control of the oscillator is considered. The trained MLP neural network $\mathcal{F}$ is used to estimate the response of the oscillator. A MLP neural network is trained with cases A and B to estimate the weights of the controller network (16). In Fig. 1 the measured displacement, the estimated uncontrolled displacement (estimated by the trained neural network $\mathcal{F}$ acting as simulator) and the estimated controlled displacement are shown. For simplicity $a = 0$ has been used in (17). Thereby, there are no limits on the applied control force. As shown in Fig. 1 the neural network controller is able to decrease the displacement of case C significantly.

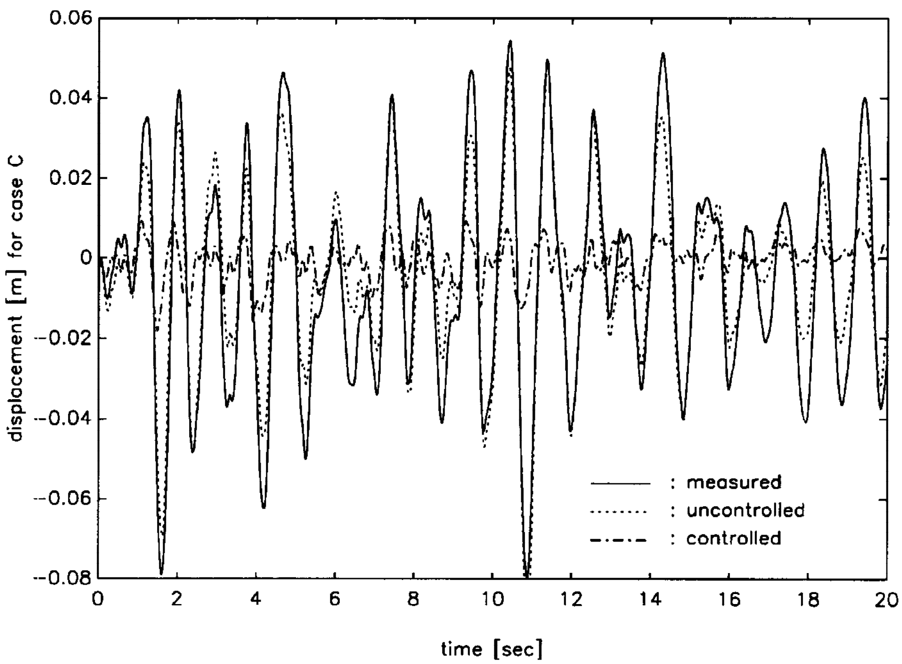

Fig. 1: Measured, uncontrolled and controlled displacement for case C.

7 CONCLUSIONS

In this paper the ability of a partially recurrent multi layer perceptron neural network to model the response of a single degree-of-freedom hysteretic oscillator is investigated. It is assumed that the displacement and the velocity of the oscillator at the previous time step can be measured whereas the hysteretic component is non-observable. The neural network trained with two time series with relatively small and large displacements of the oscillator shows a very good ability of simulating the displacements of the oscillator for a time series with medium displacements. Therefore, it seems possible to approximate the non-linear and non-analytical vector function modelling the response of the hysteretic oscillator by a neural network, i.e. an analytical function.

The ability of a neural network to model an active closed-loop controller of a hysteretic oscillator is next investigated. It is assumed that there is no limit on the applied control force. Again the time series with the relatively small and large displacements are used for training. During training the already trained neural network is used to simulate the response of the oscillator. The trained neural network controller is capable of reducing the displacements of the oscillator significantly when applied to a time series with medium displacements.

By using neural networks for real-time control the large computing time that usually is necessary for one-step ahead prediction of the response and determination of the control forces can be omitted as feedforward in a trained neural network is very fast. The neural networks can be updated by training, this may require a large computing time, but it can be performed off-line. Another advantage of using neural networks for control is that the neural network modelling the response can be trained with 'real' data from measurements, thus a non-linear model of the real structure is controlled—not a matematical model with assumptions about the behaviour of the structure, material, sensors and actuators.

8 ACKNOWLEDGEMENT

The present research was partially supported by The Danish Technical Council within the research program on Safety and Reliability.

9 REFERENCES

Doo, L.B., C.R. Thewalt and D.R. Rehak (1989). *Neural Networks for System Identification and Pulse Control of Linear Systems.* Technical Report, Department of Civil Engineering, Carnegie Mellon University, Pittsburgh, PA.

Hertz, J., A. Krogh and R.G. Palmer (1991). *Introduction to the Theory of Neural Computation.* Addison-Wesley, Redwood City, CA.

Hunt, K.J., D. Sbarbaro, R. Żbikowski and P.J. Gawthrop. Neural Networks for Control Systems—A Survey. *Automatica*, Vol. 28, No. 6, pp. 1083–1112.

Miccoli, G., A. Concilio and L. Lecce (1994). *Research Activity on Noise and Vibration Active Control by Piezoceramic Transducers.* Proceedings of the 12th International Modal Analysis Conference.

Press, W.H., B.P. Flannery, S.A. Teukolsky and W.T. Vetterling (1988). *Numerical Recipes in C.* Cambridge University Press.

Rehak, D.R., and J.H. Garrat, Jr. (1992). *Structural System Control using Neural Networks.* Proceedings of the Eighth Conference on Computing in Civil Engineering, ASCE, Dallas, TX.

Saragoni, G.R. and G.C. Hart (1974). Simulation of Artificial Earthquakes. *Earthquakes Engineering Structural Dynamics*, Vol. 2, pp. 249–267.

Soong, T.T. (1988). Active Structural Control in Civil Engineering. State-of-the-art Review. *Engineering Structures*, Vol. 10, No. 2, pp. 74–84.

Soong, T.T. (1990). *Active Structural Control. Theory & Practice.* Longman Scientific & Technical.

Suzuki, Y. and R. Minai (1985). *Seismic Reliability Analysis of Hysteretic Structures Based on Stochastic Differential Equations.* ICOSSAR'85, Proceedings of the 4th International Conference on Structural Safety and Reliability, Vol. 2, pp. 177–186.

Sørensen, O. (1994). *Neural Networks for Non-Linear Control.* Proceedings of the 3th IEEE Conference on Control Applications, Glasgow, pp. 161–166.

Tajimi, H. (1960). *A Standard Method of Determining the Maximum Response of a Building Structure during an Earthquake.* Proceedings of the 2nd World Conference on Earthquake Engineering, Tokyo, Vol. 2, pp. 781–797.

Thesbjerg, L. (1992). *Optimal Vibration Control of Engineering Structures.* Ph.D. Thesis, Aalborg University.

Wen, Y.K. (1976). Method for Random Vibration of Hysteretic Systems. *Journal of the Engineering Mechanics Division, American Society of Civil Engineers*, Vol. 102, No. EM2, pp. 249–263.

Yao, J.T.P. (1972). Concept of Structural Control. *Journal of the Structural Division*, Vol. ST7, pp. 1567–1574.