Optimal Active Control of Building Structures Based on Acceleration Feedback Using Artificial Neural Networks

Maciej Przychodzki* and Roman Lewandowski
Institute of Structural Engineering, Poznan University of Technology
ul. Piotrowo 5, 60-965 Poznan, Poland
maciej.przychodzki@ikb.poznan.pl
roman.lewandowski@ikb.poznan.pl

Abstract

This paper is devoted to a new approach to the active control of building structures. The control is based on the acceleration feedback that enables direct usage of data from accelerometers. The active reduction of vibrations of building structures realized according to classical optimisation theory is saddled with some difficult mathematical problems. These problems can be avoided by using artificial neural networks in active structural control. The numerical simulations were made for elastic structures loaded by the wind.

Keywords: artificial neural networks, optimisation, active control

1. Introduction

The active control of building structures is one of the most effective methods of the reduction of structural vibrations. The use of the active control in decreasing vibrations of slender structures excited by environmental loads like the wind or earthquakes is especially profitable. This method consists in installing special mass or tendon actuators which act with additional forces to the structure. In the classical approach to the active control of building structures, the active control forces are calculated on the basis of the actual dynamical state of the building. However, the determination of relations between the actual dynamical state of the structure and the designed control forces involves a few mathematical problems. The solution to the Riccatti equation is especially difficult. Moreover, the active control based on the classical optimisation theory has application in the reduction of the vibrations of elastic structures only. The control of buildings using artificial neural networks is more universal. An additional advantage of this method is that it allows avoiding difficult mathematical calculations. The relations between the control forces and the state-space variables are determined during the learning process. The use of artificial neural networks in active structural control with the state-space feedback is described in Reference [1].

In the control method based on the state-space feedback the displacements and velocities of the structure have to be measured. However, direct measurement of structure displacements is quite a difficult process. Measuring building accelerations is easier and more precise. The state-space variables can be calculated from measured data but the calculations are often saddled with miscounts. That is why the research on the possibility of using the data directly from accelerometers in calculations of the active control forces seems to be justified. The issue of the use of the acceleration feedback in classical control of structures is described in References [2] and [3].

In this research, the active control with acceleration feedback using the artificial neural network consisted of two layers, whereof one is non-linear, is analyzed.

2. Formulation and description of the method

The correct working of an artificial neural network depends on the appropriate choice of the learning method. In this approach, the method of the network learning is one of the supervised learning methods. The most popular rule of learning (delta rule) consists in choosing the weights of connections between neurons in such a way that the cost function is minimal. Often the cost function takes the following form:

$$J = \int_0^{T_f} (\mathbf{d} - \mathbf{y})^2 \, dt \,, \tag{1}$$

where the symbol $\mathbf{d}$ denotes the desired network output vector and $\mathbf{y}$ is the actual network output vector. From the formula (1) follows that the prototype of the neural network behaviour is necessary. In other words, the set of function values assigned to a given set of arguments has to be known. In the case of active structural control, finding the prototype is quite difficult. In order to avoid this problem, the cost function (1) can be replaced by a function, for which the control will be optimal:

$$J = \int_0^{T_f} \left(\dot{\mathbf{x}}(t)^T \mathbf{Q} \dot{\mathbf{x}}(t) + \mathbf{u}(t)^T \mathbf{R} \mathbf{u}(t) \right) dt \,. \tag{2}$$

Symbols $\mathbf{Q}$ and $\mathbf{R}$ denote weight matrices and $\mathbf{u}$ is the control forces vector. The vector $\dot{\mathbf{x}}(t)$ is the first derivative of the state-space vector, which consists of displacements, and velocities of the structure. The state-space equation for an elastic structure can be written in the following form:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{H}\mathbf{f}(t) \,. \tag{3}$$

The matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{H}$ are defined by

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D} \end{bmatrix},$$
$$\mathbf{B} = \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1}\widetilde{\mathbf{B}} \end{bmatrix},\ \mathbf{H} = \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1}\widetilde{\mathbf{H}} \end{bmatrix}, \tag{4}$$

where $\mathbf{K}$, $\mathbf{D}$, $\mathbf{M}$ and $\widetilde{\mathbf{B}}$, $\widetilde{\mathbf{H}}$ denote stiffness, damping, mass and location matrices respectively. Symbol $\mathbf{f}$ denotes the external forces vector.

The function (2) is the performance index of the optimal control of the structure. The index is non-standard because the state- space vector in the standard approach is replaced by its first derivative. The discrete in time version of the index (2) can be written as follows:

$$\widetilde{J} = \sum_n \widetilde{J}_n = \sum_n \left(\dot{\mathbf{x}}_{n+1}{}^T \mathbf{Q} \dot{\mathbf{x}}_{n+1} + \mathbf{u}_n{}^T \mathbf{R} \mathbf{u}_n \right) \Delta t \,. \tag{5}$$

The weights of connections between neurons in a network have to be matched in such a way that the value of the performance index (5) is minimal. In order to fulfil this condition, the weights updates are calculated according to formula

$$\Delta W_{kj} = -\eta \frac{\partial \widetilde{J}_n}{\partial W_{kj}} \,. \tag{6}$$

Symbol η denotes learning rate. Introducing $\widetilde{J}_n$ defined by equation (5) in formula (6) the weights updates take the following form:

$$\Delta W_{kj} = -\eta \Delta t \left(\dot{\mathbf{x}}_n{}^T \mathbf{Q} \frac{\partial \dot{\mathbf{x}}_n}{\partial \mathbf{u}_n} + \mathbf{u}_n{}^T \mathbf{R} \right) \frac{\partial \mathbf{u}_n}{\partial W_{kj}} \tag{7}$$

The control forces vector can be written as

$$\mathbf{u} = g\left(\mathbf{net}^W\right), \tag{8}$$

where the function g is the transfer function of the output layer and $\mathbf{net}^W$ is the input vector of this layer. The elements of this vector are described by the formula

$$net_k^W = \sum_j f\left(net_j^V\right) \cdot W_{kj} \,. \tag{9}$$

The symbol f denotes the transfer function of a hidden layer. A function which is most widely used in neural networks non-linear continuous transfer function is the so called the hyperbolic tangent sigmoid transfer function which is described by equation

$$f(net) = \frac{2}{1+\exp(-net)} - 1 \,. \tag{10}$$

Its chart is shown in Fig.2. The output values of the hidden layer are calculated from the equation

$$net_j^V = \sum_i \dot{x}_i V_{ji} \,, \tag{11}$$

where x_i is the state-space variable and V_{ji} is the weight of the hidden layer.

Taking into account relations (8) and (9), the first derivative of the k-th element of the vector $\mathbf{u}_n$ can be written as

$$\frac{\partial u_k}{\partial W_{kj}} = \frac{\partial u_k}{\partial net_k^W} \cdot \frac{\partial net_k^W}{\partial W_{kj}} = g'\left(net_k^W\right) \cdot f\left(net_j^V\right) \tag{12}$$

The matrix $\dfrac{\partial \dot{\mathbf{x}}}{\partial \mathbf{u}}$, which occurs in equation (7), is called the sensitivity matrix. For linear dynamic systems, this matrix can be calculated in a direct way. The solution to equation (3) in the discrete form is following:

$$\mathbf{x}_{n+1} = \mathbf{G}\mathbf{x}_n + \mathbf{H}\mathbf{u}_n \,, \tag{13}$$

where the matrices $\mathbf{G}$ and $\mathbf{H}$ are given by

$$\mathbf{G} = e^{\mathbf{A}\Delta t}$$
$$\mathbf{H} = \left(e^{\mathbf{A}\Delta t} - \mathbf{I}\right)\mathbf{A}^{-1}\mathbf{B} \,. \tag{14}$$

Equation (13) is correct with the assumption that the external loads are neglected. The dynamic system has to fulfil the equation of motion, which in discrete time can be written as follows

$$\dot{\mathbf{x}}_{n+1} = \mathbf{A}\mathbf{x}_{n+1} + \mathbf{B}\mathbf{u}_n \tag{15}$$

After introducing formula (13) into equation (15) and differentiating equation (15) with respect to $\mathbf{u}_n$ the sensitivity matrix is obtained in the following form

$$\frac{\partial \dot{\mathbf{x}}_{k+1}}{\partial \mathbf{u}_k} = \mathbf{A}\mathbf{H} + \mathbf{B} \,. \tag{16}$$

From relation (16) follows that the elements of the sensitivity matrix are constant in the time domain.

The learning process of the hidden neurons layer is done according to the same assumptions as for the output layer. Weights updates are calculated using the formula

$$\Delta V_{ji} = -\eta \frac{\partial \widetilde{J}_n}{\partial V_{ji}} \,. \tag{17}$$

After the transformation, equation (17) is written in the following form

$$\Delta V_{ji} = -\eta \Delta t \left(\dot{\mathbf{x}}^T{}_n \mathbf{Q} \frac{\partial \dot{\mathbf{x}}_n}{\partial \mathbf{u}_n} + \mathbf{u}_n{}^T \mathbf{R}_n \right) \frac{\partial \mathbf{u}_n}{\partial V_{ji}} \,. \tag{18}$$

The first derivative of the k-th element of the vector $\mathbf{u}_n$ appearing in equation (18) is obtained in the similar way as formula (12):

$$\frac{\partial u_k}{\partial V_{ji}} = \frac{\partial u_k}{\partial net_k^W} \cdot \frac{\partial net_k^W}{\partial net_j^V} \cdot \frac{\partial net_j^V}{\partial V_{ji}} = g'\left(net_k^W\right) \cdot f'\left(net_j^V\right) \cdot W_{kj} \cdot \dot{x}_i \tag{19}$$

The sensitivity matrix appearing in equation (18) has the same form as in formula (7) and is given by relation (16).

3. Numerical calculations

The numerical calculations were made for the 4-storey building, exposed to forces excited by the wind. The calculation model of building is the so-called shear frame described in Reference [4]. The weight of the building is concentrated at the

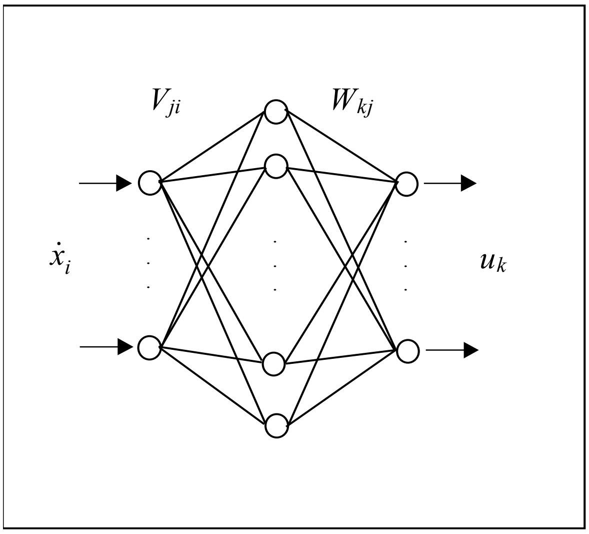

Figure 1: Scheme of the artificial neural network

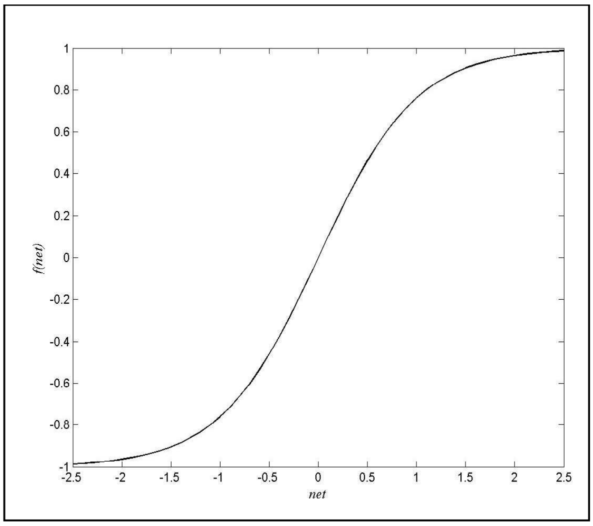

Figure 2: Chart of the tangent sigmoid transfer function

floor level, and the frame girders are infinitely rigid, compared with columns. The following characteristics of the building were assumed for the calculations: storey height *l*=7.0m, column rigidity *EJ*=2.00 10^8 kNm^2, dimensionless damping factor γ=0.05, storey weight *M*=30000 *kg*. The first natural frequency of building is 2.372 rad/s.

The wind is considered to be the stationary random process with zero mean value. The random characteristics of the wind velocity fluctuations are described by the spectral density function, proposed by Kaimal. Typical time variances of the wind velocity fluctuations $w_i(t)$ were generated as described in Reference [6].

The dynamic forces excited by the wind are calculated from the formula:

$$P_i(t) = \rho C_d A_i U_i w_i(t) \ , \tag{20}$$

where ρ, C_i, A_i and U_i denote the air density, the drag factor, the exposure area and the wind mean velocity related to the point "*i*". The chart of the wind force acted on the level of last structure floor is shown in Figure 3.

The weighting matrices were assumed as diagonal matrices and took the following form:

$$\mathbf{Q} = 10^6 \mathbf{I}\,,\ \ \mathbf{R} = 10^{-6} \mathbf{I}, \tag{21}$$

where **I** is the unit matrix.

There were two actuators generated control forces, located on first and second floor of the structure.

The learning process consisted of 50 000 cycles with the learning rate $\eta = 0.001$. The time step was equal $\Delta t = 0.01s$. The weights of the neural network were updated at each time step (pattern learning mode).

4. Results

The chart shown in Figure 4 shows the changes of performance index during learning process. It can be seen that there are two phases of learning. In the first phase the process is seemingly chaotic. The weights changed very fast in this time. In the second phase the changes of weights were less rapid and the performance index was decreasing continuously. After this part of the learning process, the weights achieved the equilibrium state.

In Figures 5 and 6 the displacements and accelerations of the last floor in time domain are shown, respectively. These charts prove that the active control of structures based on artificial neural networks gives correct results. Both, displacements and accelerations can be reduced significantly.

5. Final remarks

The presented method of the active control of building structures enables the direct use of data from accelerometers. The active control realized with artificial neural networks allows for avoiding difficult mathematical problems. The use of the performance index (2) as cost function in the learning process of the neural network causes the active control to be optimal. The numerical calculations show that the described method of active control can reduce structural displacements and accelerations significantly.

6. References

[1] Kim, J.T., Jung, H. J., Lee, I. W., Optimal structural control using neural networks, *Journal of Engineering Mechanics*, vol.126, pp.201-205, February, 2000.

[2] Kwak, S.K., Washington,G., Yedavalli, R.K., Acceleration-based vibration control of distributed parameter systems using the "reciprocal state-space framework", *Journal of Sound and Vibration*, vol.251, pp.543-557, 2002.

[3] R.Lewandowski, M.Przychodzki, Optimal active and semi-active control of building structures using the acceleration feedback and non-standard performance index. *15th International Conference on Computer Methods in Mechanics CMM-2003*, 3-6 June 2003, Gliwice/Beskidy Mountains, Poland.

[4] Paz, M., *Structural dynamics: Theory and computation,* Van Nostrand Reinhold Company, New York, 1985.

[5] Kim, D.H., Lee, I.W., Neuro-control of seismically excited steel structure though sensitivity evaluation scheme, *Earthquake Engng Struct. Dyn.*, 30, pp.1361-1377, 2001

[6] Lewandowski, R., Application of semi-empirical model to analysis of vortex-excited vibrations of beams near synchronisation region, *Computational Civil and Structural Engineering* (eds. G.De Roeck and B.H.V. Topping), Civil-Comp Press, Edinburgh, 2000.

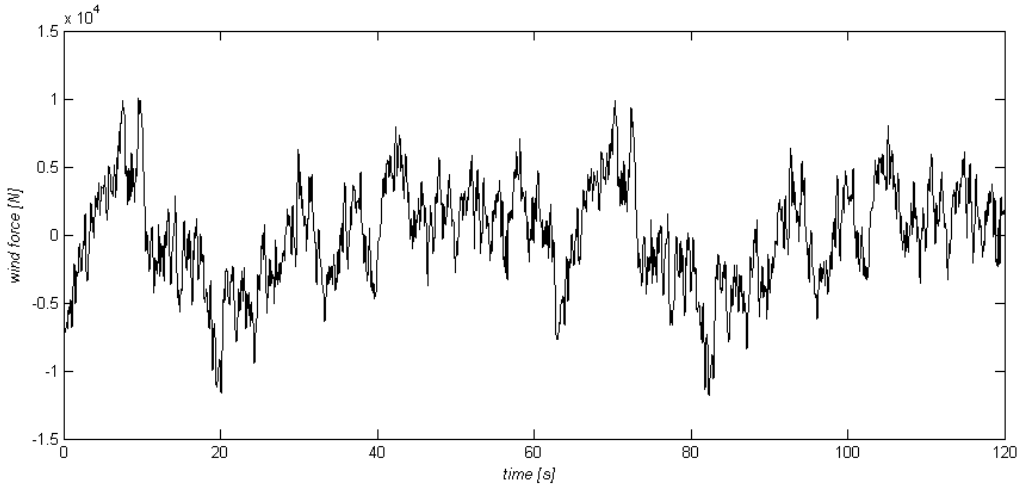

Figure 3: Wind force acted on the 4-th floor level.

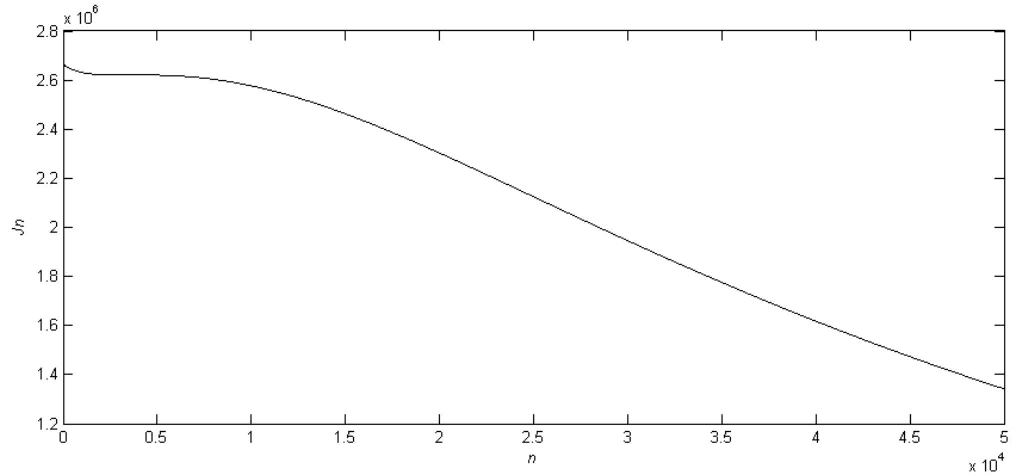

Figure 4: Learning history.

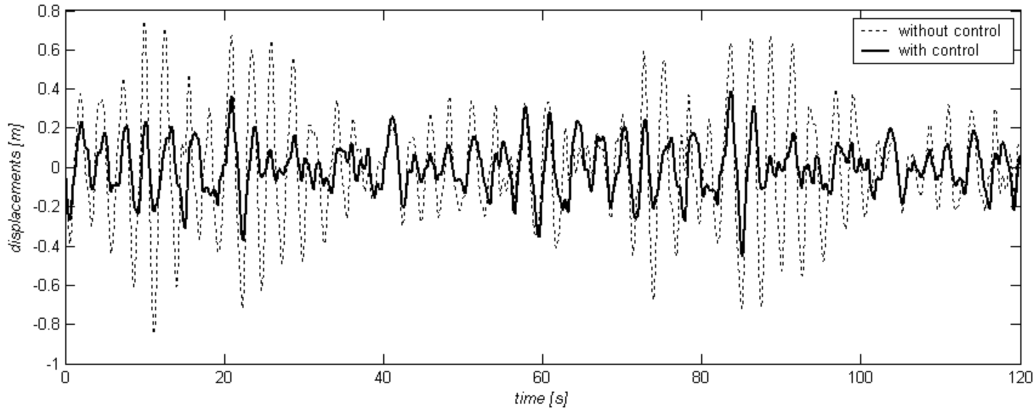

Figure 5: Displacements of the 4-th floor

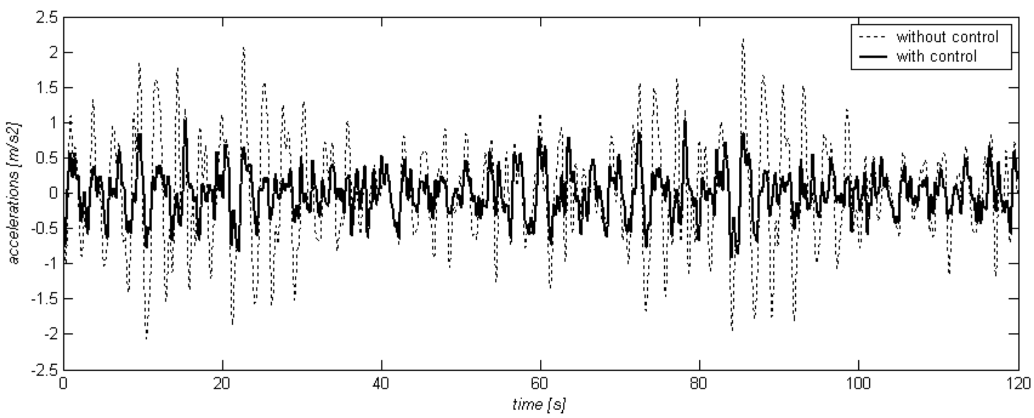

Figure 6: Accelerations of the 4-th floor

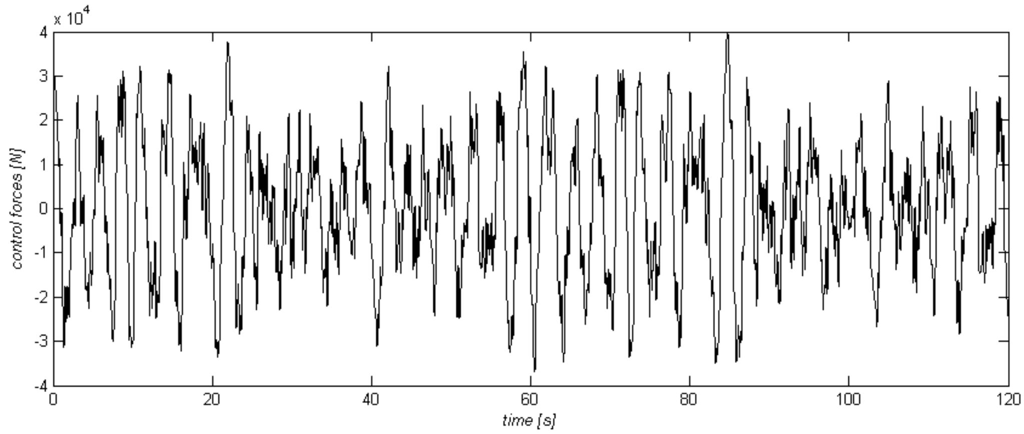

Figure 7: Control force in actuator on 1-st floor