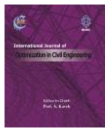
INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING
Int. J. Optim. Civil Eng., 2014; 4(1):77-91



STRUCTURAL RESPONSE OBSERVER BASED ON ARTIFICIAL NEURAL NETWORK

A. Gholizad[*, †] and S. D. Ojaghzadeh Mohammadi
Department of Engineering, University of Mohaghegh Ardabili

ABSTRACT

Structural vibration control is one of the most important features in structural engineering. Real-time information about seismic resultant forces is required for deciding module of intelligent control systems. Evaluation of lateral forces during an earthquake is a complicated problem considering uncertainties of gravity loads amount and distribution and earthquake characteristics. An artificial neural network (ANN) has been trained in this article to estimate these forces. This ANN was trained on the results of time history analysis of a three-story building under 702 different loadings. Results of numerical examples verify that the trained ANN can predict the expected forces with negligible deviations.

Received: 20 December 2013; Accepted: 24 March 2014

KEY WORDS: Structural vibration control, Dynamic response estimation, Multilayer perceptron artificial neural network

1. INTRODUCTION

Powerful theoretical methods and mechanical tools are now available to be applied for complicated engineering problems of the modern world. Vibration control is one of these issues in structural engineering. Precise and real time estimation of lateral seismic loads is required for intelligent control systems to mitigate the structural vibration. Reliable evaluation of these forces is not practical through closed form analytical methods due to uncertainties on gravity loads magnitude and pattern. Moreover, the ground motion accelerations is not usually available instantaneously during an earthquake. These make seismic loads prediction a complex multivariable dependent problem.

[*]Corresponding author: A. Gholizad Assistant Professor, Engineering Department, University of Mohaghegh Ardabili, Ardabil, Iran, PO BOX 56199-11367, Tel (+98-451) 5517373.
[†]E-mail address: : Gholizad@uma.ac.ir (A. Gholizad)

For analyzing a complex system, one may decompose it into smaller and naturally simpler elements. Finding a simple model for the behavior of these elements, then gathering those parts to construct the primary system is a common method [1]. Similar to finite element method, which is used for analyzing a structure, there are different approaches in the field of artificial intelligence (AI) in computational methods. AI is useful to simulate the behavior of systems with unknown or complicated governing equations.

The theory of soft computing which is related to the field of neural networks, fuzzy logic, evolutionary computing, probabilistic computing, and genetic algorithms is one of the best approaches to solve these problems [2]. In this research multilayer perceptron (MLP) neural network has been used for predicting the effective lateral forces on a structure during an earthquake. Artificial neural networks (ANNs) are able to adapt themselves with the natural relationships, which exist within an available set of data by using learning rules. After completing the adaptation procedure, the network can predict the behavior of a new sample, which does not exist in the data set [3]. There are many applications of AI in civil engineering:

Zhao *et al*. used the radial basis function neural network for preliminary design of concrete box girder bridges. They combined this ANN with fuzzy logic to perform the task of noisy data filtering, knowledge extraction, and candidate synthesis [4]. A multilayer-functional-link neural network was trained on the results of 55 experimental test results for predicting the confinement efficiency of concentrically loaded reinforced concrete columns with rectilinear transverse steel by Tang et al. [5]. Jeng and Mo modeled the critical structural response of a prestressed concrete bridge subjected to earthquake excitation of various magnitudes along different directions by using MLP neural networks. They trained the ANNs on the analytical data obtained from the nonlinear dynamic finite fiber element analyses of the bridge [6]. A MLP neural network was trained by Gupta *et al*. [7] on the results of 864 concrete specimens to predict the concrete strength based on concrete mix design parameters such as size and shape of specimen, curing technique, and environmental conditions. Pei and Smyth introduced a new architecture of multilayer feedforward neural network for problems with nonlinear and hysteric dynamic behaviors. They used this ANN for simulating the nonlinear dynamic response of single-degree-of-freedom oscillators [8], [9]. Von Neumann and Moor neighborhood model of cellular automata (CA) used to establish the CA numerical model for masonry wallets by Zhang et al. They used also a MLP neural network for predicting the cracking pattern of a wall based on the proposed CA model of the wallet and some data of recorded cracking at zones [10]. Ghaboussi and Joghataie presented a neurocontroller for linear structural control when the response of the structure remained within the linearly elastic range [11]. Bani-Hani and Ghaboussi used a neurocontroller for nonlinear structural control of a three-story steel frame with an actuator and tendon system in the first floor [12]. MLP neural network was used for structural control of a multi-story building by Chen *et al*. [13]. Tang used MLP neural network for active control of a single-degree-of-freedom system [14]. A counter-propagation neural network (CPNN) was trained by Madan to output the control forces required to reduce the structural vibrations without any feedback on the correctness of the output control forces. This ANN can learn from the control environment to compute the required control forces in an unsupervised manner [15]. Cho *et al*. designed a MLP neural network controller with a single hidden layer for single-degree-of-freedom and two-degree-of-freedom bridge systems. They found the optimal number of neurons in the hidden layer based on progressively adding one hidden neuron and observing the changes in performance of the neural network [16]. Li and Yang proposed a kind of multi-branch back propagation neural network (BPNN) to identify a structural dynamic system for predicting the structural future dynamic response. They used primary factors such as structure state variables and seismic inputs as branches of the model to increase the accuracy of prediction [17]. Li et al. used an artificial neural network for modeling the dynamic response of tall buildings. In addition, they used genetic algorithm for optimizing the structural control systems [18]. In this paper, the effective earthquake loads on different floors of a three-story building is predicted using a multilayer perceptron neural network. In fact, the building's behavior is modeled with a trained MLP neural network on different conditions of loadings. To find the building's reaction on a real situation, the time history analysis of 702 different combinations of dead, live, and earthquake loads was used. Results of numerical examples verify the high accuracy of trained neural network.

2. DYNAMIC MODEL OF A MULTI-STORY BUILDING

Dynamic model of a multi-story building has been shown in Figure 1. In general, the behavior of this structure is nonlinear. Step-by-step integration is one of the best methods for linear and nonlinear analysis of structures. In fact, the response of structure is calculated in a sequential series of time increments in this method. Since it is not necessary to determine mode shapes, frequencies, and decoupling the system of differential equations of motion, step-by-step integration method is under consideration more than modal analysis for systems of high degrees of freedom.

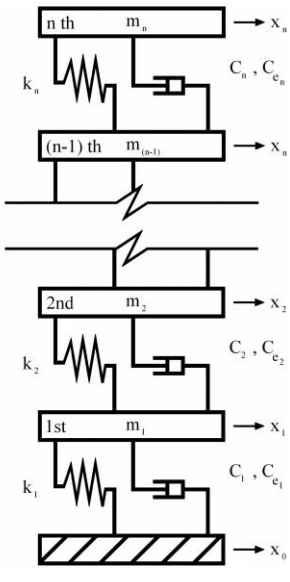


Figure 1. Dynamic model of a multi-story building

Equation of motion for *j*-th floor of structure, which is shown in Figure 2, is obtained by using D'Alembert's principle:

$$\begin{aligned}&\left(m_j+m_{e_j}+m_{e_{j+1}}\right)\ddot{u}_j-\left(c_{j+1}+c_{e_{j+1}}\right)\dot{u}_{j+1}+\left(c_{j+1}+c_j+c_{e_{j+1}}+c_{e_j}\right)\dot{u}_j\\&-\left(c_j+c_{e_j}\right)\dot{u}_{j-1}-k_{j+1}u_{j+1}+\left(k_j+k_{j+1}\right)u_j-k_ju_{j-1}=-m_j\ddot{x}_0\end{aligned} \tag{1}$$

where m and m_e are the mass of floor and damper respectively. c and c_e are the damping coefficient of story and damper respectively. k is the story stiffness and u, $\dot{u}$ and $\ddot{u}$ respectively are the displacement, velocity and acceleration of each floor relative to the base. $\ddot{x}_0$ denotes the absolute acceleration of the base.

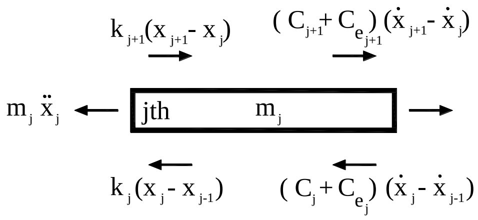


Figure 2. Free body diagram of j-th floor

The matrix form of Eq. (1) is:

$$[M]\{\ddot{u}\}+[C]\{\dot{u}\}+[K]\{u\}=-[M]\{1\}\ddot{x}_0 \tag{2}$$

where *[M]*, *[C]* and *[K]*are mass, damping coefficient and stiffness matrixes. Eq. (2) is a set of second order coupled differential equations, which governs the vibration of structure. Incremental equation of motion for step-by-step method is written as:

$$\begin{aligned}&[M]\{\Delta\ddot{u}\}+[C]\{\Delta\dot{u}\}+[K]\{\Delta u\}=\{\Delta P\}\\&\{P\}=-[M]\{1\}\ddot{x}_0\end{aligned} \tag{3}$$

where *{P}* can be regarded as base acceleration resultant load vector. considering linear variation for acceleration, the changes in velocity and displacement will be of the second and third order respectively. Substitution of incremental acceleration and velocity in term of incremental displacement into Eq. 3 gives:

$$\begin{aligned}&\left[\tilde{K}\right]\{\Delta u\}=\left\{\Delta\tilde{P}\right\}\\&\left[\tilde{K}\right]=[K]+\frac{6}{\Delta t^2}[M]+\frac{3}{\Delta t}[C]\\&\left\{\Delta\tilde{P}\right\}=\{\Delta P\}+[M]\left\{\frac{6}{\Delta t}\{\dot{u}\}+3\{\ddot{u}\}\right\}+[C]\left\{3\{\dot{u}\}+\frac{\Delta t}{2}\{\ddot{u}\}\right\}\end{aligned} \tag{4}$$

Eq. (4) is a set of algebraic equations, which may be solved by a standard method. Gauss method is used here to find the incremental displacements. Afterward, the velocity increment is obtained by:

$$\{\Delta\dot{u}\} = \frac{3}{\Delta t}\{\Delta u\} - 3\{\dot{u}\} - \frac{\Delta t}{2}\{\ddot{u}\} \tag{5}$$

Because of using initial slope of elastic and damping force curves for each time increment for k and c, the left hand side of Eq. (3) is an approximate value. To avoid of error accumulation, the acceleration at the beginning of each time increment is calculated based on equilibrium of all forces.

$$\begin{aligned} \ddot{u}(t+\Delta t) &= [M]^{-1}\{P(t+\Delta t) - F_D(t+\Delta t) - F_S(t+\Delta t)\} \\ F_D(t+\Delta t) &= [C(t)]\{\dot{u}(t+\Delta t)\} \\ F_S(t+\Delta t) &= [K(t)]\{u(t+\Delta t)\} \end{aligned} \tag{6}$$

This procedure should be repeated for each time increment [19, 20]. A computer program "Dynamic Analysis of Structures" (DAS) has been provided in Delphi to analyze multi-story buildings and find the time response of structure under applied loads or base excitations such as an earthquake record. Lateral forces on each story can be calculated in addition of displacements, velocities, and accelerations.

3. MULTILAYER PERCEPTRON NEURAL NETWORK

Multilayer feed-forward network consists of input layer, one or more hidden layers and an output layer. Each layer contains several computational nodes considered as neurons. Training of this supervised neural network is done by error back-propagation algorithm, which is the generalized form of least-mean-square (LMS) algorithm for the special case of a single linear neuron. This learning rule consists of two passes through different layers of the network: In a forward pass, one of the patterns is applied to the network to find the actual response of the network on the output layer after the sequential computations done layer by layer. In this step the difference between desired and obtained values are evaluated as error signal. On the backward pass, the error signal is propagated through the network to apply the back propagation algorithm for correcting the synaptic weights, which are constant in the forward pass.

Important characteristics of the multilayer perceptron network are:

- A sigmoidal nonlinearity is included in the model of each neuron. This nonlinearity should be smooth and usually is applied to neurons by logistic function.

$$y_j(n) = \frac{1}{1+\exp(-v_j(n))} = \varphi_j(v_j(n)) \tag{7}$$

where v_j is the weighted sum of all synaptic inputs of neuron j, known as induced local field of neuron j, and y_j is the output of neuron j for the n-th training example.

$$v_j(n) = \sum_{i=0}^{m} w_{ji}(n) y_i(n) \tag{8}$$

where m is the total number of inputs applied to neuron j The synaptic weight w_{j0} (corresponding to the fixed input $y_0=+1$) equals the bias applied to neuron j.

- Learning complex models or tasks is guaranteed by using hidden neurons
- The synapses of the network make high degrees of connectivity.

4. BACK-PROPAGATION ALGORITHM

After forward pass for each pattern, the error signal at the output layer for j-th neuron at the n-th training example is defined by:

$$e_j(n) = d_j(n) - y_j(n) \tag{9}$$

where d_j is the desired response for neuron j. The instantaneous error energy for neuron $j$is defined as $e_j^2(n)/2$. In addition, the total error energy is obtained by:

$$E(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n) \tag{10}$$

where the set C includes all the neurons in the output layer. The average squared error energy is obtained by

$$E_{av} = \frac{1}{N} \sum_{n=1}^{N} E(n) \tag{11}$$

where, $N$is the total number of records which are used for the training procedure. The learning performance of the network can be checked by evaluating the rate of decrease in the amount of E_{av}. the performance can be improved by correcting the synaptic weights and biases by using delta rule:

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ji}} = \eta \delta_j(n) y_i(n) \tag{12}$$

where η is the learning-rate parameter of the back-propagation algorithm and δ is the local gradient which is defined by:

$$\delta_j(n) = -\frac{\partial E(n)}{\partial v_j(n)} = e_j(n)\varphi_j'(v_j(n)) \tag{13}$$

The amount of learning-rate parameter η indicates the training celerity of MLP neural network. It means that using the smaller values for it leads to the smoother trajectory in weight space with lower speed of weight matrix correction. On the other hand, using large values for the learning-rate parameter speeds up the learning procedure but the network may become unstable. To avoid of the network instability, the generalized form of delta rule was presented by Rumelhart *et al*. [21],

$$\Delta w_{ji}(n) = \alpha \Delta w_{ji}(n-1) + \eta \delta_j(n) y_i(n) \tag{14}$$

α is called momentum constant and it is usually a positive number in the range of $0 \le |\alpha| < 1$ to guarantee the neural network convergence [22, 23]. An extensive study has been done on the optimal combination of η and α in this research.

5. NUMERICAL EXAMPLES

A three-story building has been analyzed via provided dynamic analysis software and another reliable software to verify DAS results validity. This benchmark structure has been tested by Matsuoka et al. at the National Center for Research on Earthquake Engineering (NCREE) [24] and its characteristics are introduced on Table 1. Time response of the third floor of the benchmark structure under the north-south component of El Centro (1940) earthquake is shown in Figure 3. Earthquake record has been normalized to $0.32\, m/s^2$ for peak ground acceleration.

Table 1: Physical parameters of the 3-story benchmark structure [23].

Total height (m)		9
Width (m)		3
Span (m)		2
Mass of each story (tons)	m	6
Stiffness (N/m)	K_1	1.6609×10^6
	K_2	1.9152×10^6
	K_3	1.5694×10^6
Damping (Ns/m) coefficient	C_1	8.7314×10^2
	C_2	3.6620×10^3
	C_3	5.4533×10^3
Elasticity modulus (N/m^2)	E	1.9620×10^{11}

DAS analysis results have been compared on Table 2 both with experimental results and ETABS analyses results. According to results compatibility and proofed accuracy of DAS it has been employed for time history analyses in this study.

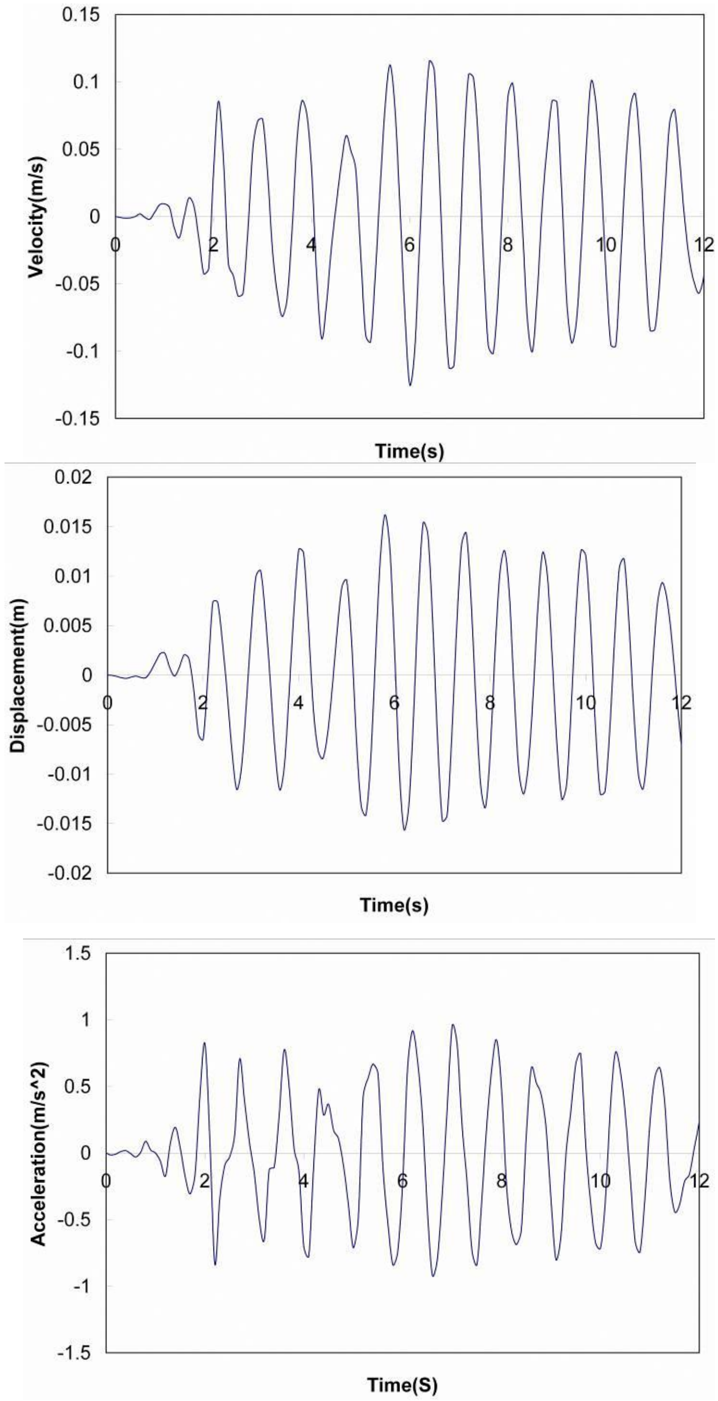


Figure 3. Time response of the benchmark structure.

Story shears are the most useful data for decision making by an intelligent control system during an earthquake. These forces are dependent on the amount of gravity loads and earthquake severity which are uncertain parameters due to variable nature of live loads and stochastic characteristics of earthquakes. Since the exact values of these items are unknown during an earthquake, it is difficult to determine the lateral forces applied to floors. prediction of these forces provides information required for operating control devices such as MR-dampers. Artificial neural network is used for this purpose in this research.

Table 2: Benchmark structure analyses results verification

	Maximum Lateral Displacement (m)		
	First Floor	Second Floor	Third Floor
Experiment [23]	0.00815	0.01397	0.01826
DAS	0.00787515	0.01298839	0.01621277
ETABS	0.0079	0.0130	0.0162
	Maximum Velocity (m/s)		
	First Floor	Second Floor	Third Floor
Experiment [23]	-	-	-
DAS	0.05606566	0.09560065	0.12095745
ETABS	0.0560	0.0954	0.1210
	Maximum Acceleration (m/s^2)		
	First Floor	Second Floor	Third Floor
Experiment [23]	0.54	0.77	1.10
DAS	0.54461268	0.77584747	0.97494574
ETABS	0.5467	0.7753	0.9761

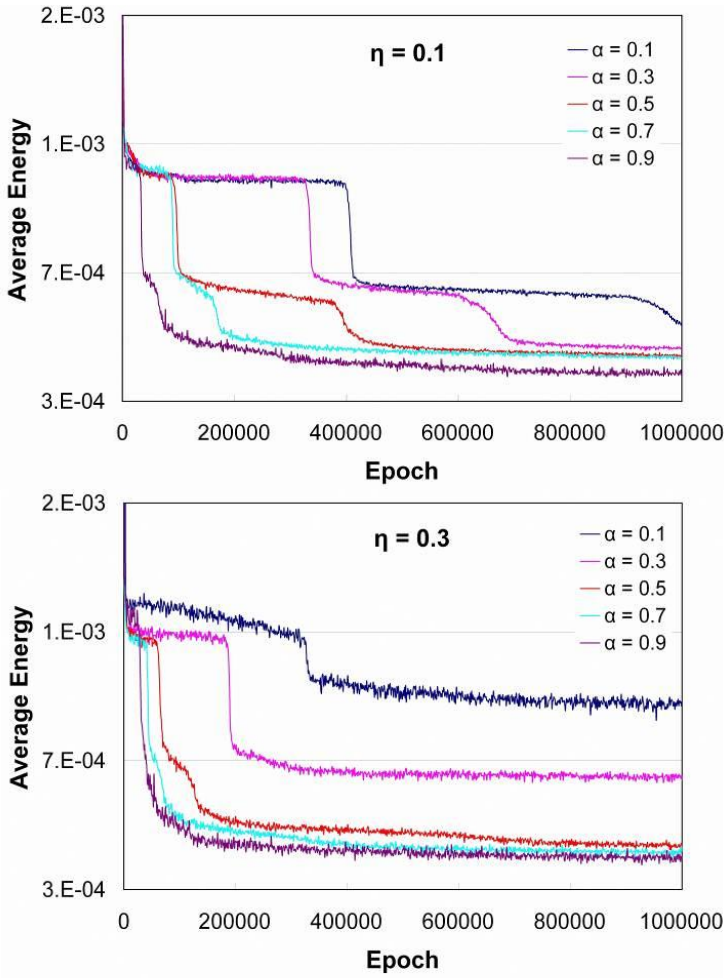
Different values of gravity loads have been applied in combination with El Centro excitation in different intensity levels for time history analysis of the benchmark structure. Applied loads variation and considered values of peak ground acceleration $a_{\max}$ are defined on Table 3.

Table 3: The range of different loads and accelerations applied to the benchmark structure

Load	Minimum	Maximum	Increment
Dead load (dN/m^2)	400	650	50
Live load (dN/m^2)	50	250	25
a_{max} (m/s^2)	0.2	1.4	0.1

A database has been provided through time history analyses of the benchmark structure performed by DAS considering 702 different load cases mentioned on Table3. Afterwards, a MLP neural network has been trained on the information of database. This ANN predicts the lateral forces due to earthquake vibrations based on the measured lateral displacements of floors. A computer program has been developed for "Prediction of Lateral Forces by Intelligent System" (PLFIS). Therefore, during an earthquake PLFIS is ready to receive a set of displacements of floors to calculate the lateral forces by a forward pass from input to output layer. These forces are useful for decision making in a control system. This procedure can be repeated as long as the PLFIS receives information about the displacements of structure.

Input layer, first and second hidden layers, and output layer with 3, 10, 10, and 3 neurons respectively configure the multilayer perceptron used in PLFIS. Input vector includes the lateral drift of the stories and the output vector consists of seismic forces on different floors. Different combinations of η and α have been tried to evaluate the influence of learning-rate parameter and momentum constant on the performance of MLP network used in this research. Considering 25 different combinations of the values 0.1, 0.3, 0.5, 0.7, and 0.9 for η and α the MLP neural network has been trained on the results of benchmark structure analyses. Figure 4 illustrates the rate of changes in average energy of the neural network for these combinations.



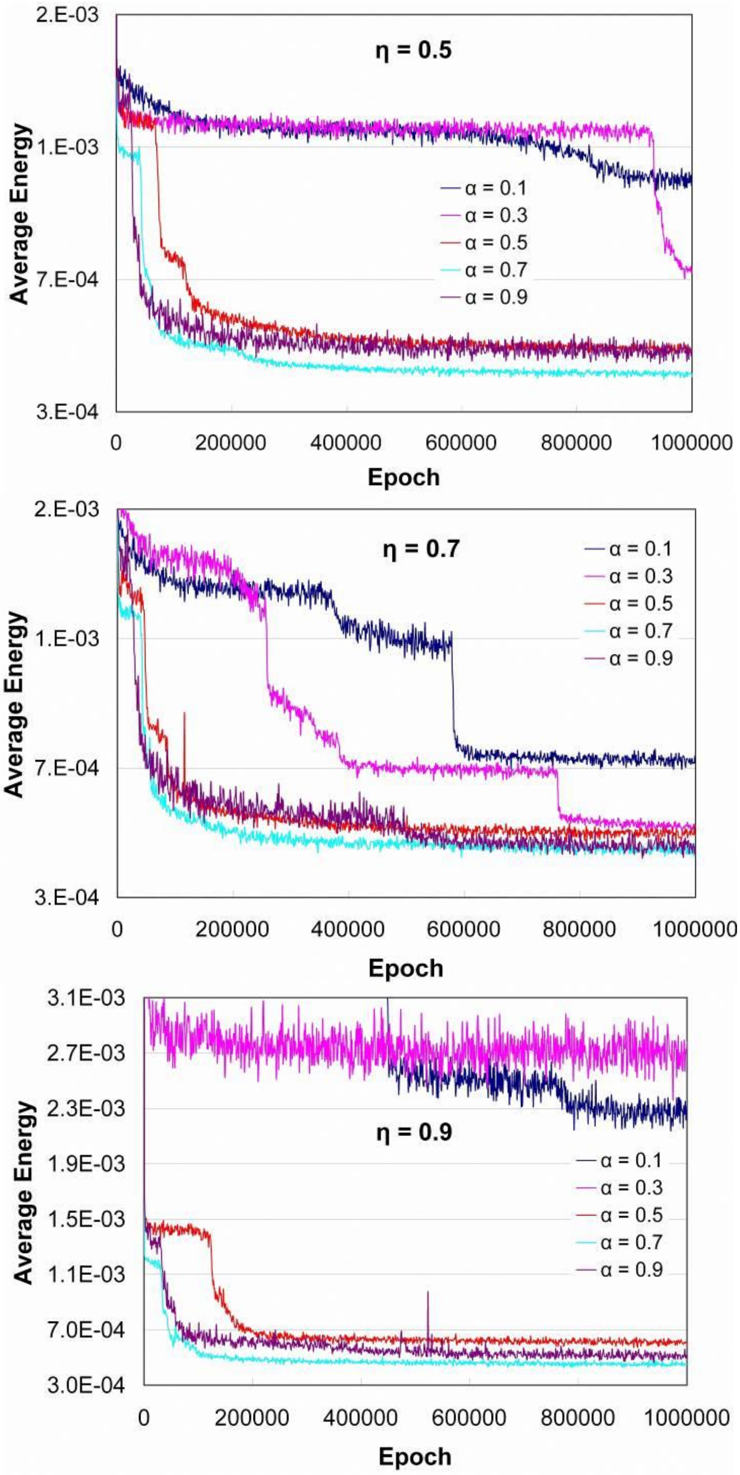


Figure 4. Rate of changes in the average energy for different values of η and α

According to Figure 4, for each selection of learning-rate parameter η, there is a specific value for momentum constant α , which leads the MLP neural network to its best training trajectory. Table 4 shows these values.

Table 4. Best combinations of parameters η and α

Learning-Rate Parameter	Momentum Constant	Average Energy in the Last Epoch
0.1	0.9	0.000381
0.3	0.9	0.000395
0.5	0.7	0.000405
0.7	0.7	0.000442
0.9	0.7	0.000453

Table 4 indicates that the learning rate parameter increase results in a slight decrease in momentum constant α . The best training graphs of Figure 4 are represented in Figure 5. All graphs shown in Figure 5 are closed to each other, although the best training trajectory is related to $\eta = 0.1$ and $\alpha = 0.9$.

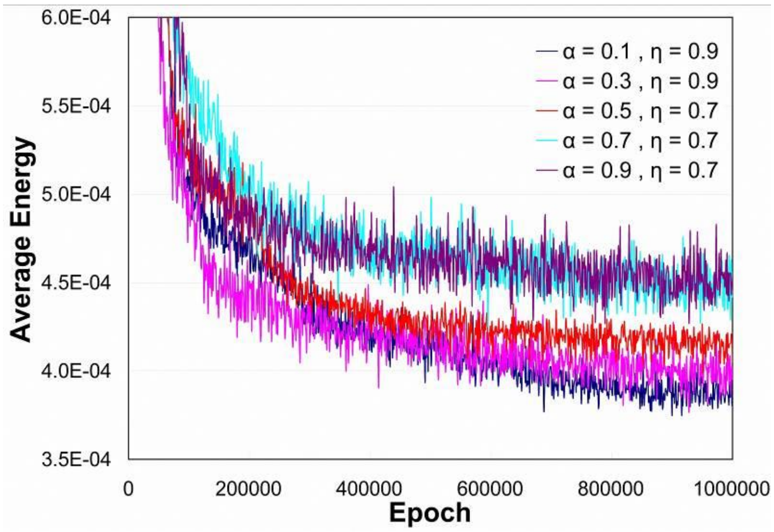


Figure 5. Different combinations of learning-rate parameter and momentum constant

The intelligent system has been trained with the best values of η and α indicated on Figure 5. Four examples are presented in Table 5 to show the efficiency and accuracy of the trained intelligent system. All examples analyzed by DAS and the exact values of lateral displacements of floors and the earthquake forces applied to them are available in Table 5.

If an earthquake happen while the exact values of gravity loads or the magnitude of the earthquake accelerations are unknown, the trained neural network will be able to predict the effective earthquake forces on floors when the lateral displacements of them are measured. Figure 10 compares the exact and estimated seismic loads on different floors of examples 1 to 4 if the measured displacements of floors are assumed equal to those values in table 5.

Table 5. Different loadings for benchmark structure and its exact results by DAS

Example	Dead Load (dN/m)	Live Load (dN/m)	a_{max} (m/s^2)	Floor	Lateral Displacement (m)	Effective Earthquake Load (kN)
1				1st	0.003153	14.072
	400	60	0.2	2nd	0.005146	18.357
				3rd	0.006422	22.623
2				1st	0.020699	81.422
	620	175	1.2	2nd	0.034698	119.758
				3rd	0.044207	153.127
3				1st	0.014259	55.215
	430	210	0.6	2nd	0.023753	81.557
				3rd	0.030077	103.387
4				1st	0.013237	49.628
	630	110	0.73	2nd	0.022192	75.009
				3rd	0.028261	93.636

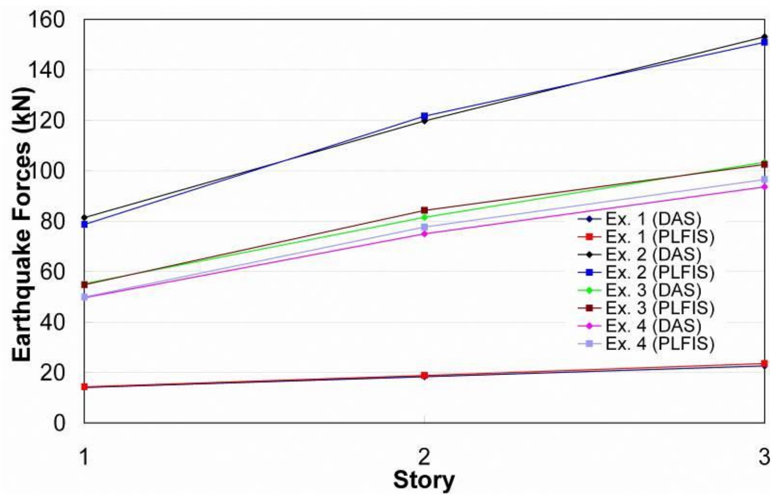


Figure 6. Comparison between the exact (DAS) and predicted (PLFIS) earthquake loads

It is clear from Figure 10 that the intelligent system determines the earthquake forces with a small amount of errors. In fact, the maximum error is related to the lateral forces on the third floor of example 1 which is 4.13%. Therefore it can be used as an estimating module of the controller system.

6. CONCLUSIONS

Determination of effective lateral forces applied on different floors of a structure, plays main role for deciding about the control forces to restrict the vibrations of the structure. These forces are functions of different parameters such as the amount of and the distribution of dead and live loads, the pattern of base excitation, and the stiffness of the structure. It is not possible to find the mathematical form of these functions by classic methods. MLP neural network used for this purpose. A three-story structure analyzed for 702 different loadings. Afterwards, the ANN was trained on the information of these cases. Thus, the trained ANN is able to predict the lateral forces on floors by receiving the lateral displacements, measured by precise devices, of the structure. An extensive study was done on the learning-rate parameter and momentum constant in the generalized form of delta rule. This study shows that in this problem, for different values of learning-rate parameter, the best values of momentum constant are 0.9 and 0.7 with an average value of 0.8. The predicted lateral forces on the floors of the benchmark structure for different examples show a high degree of accuracy. In fact, the maximum error is related to the lateral force on the third floor of example 1, which is 4.13%. It shows that MLP neural network is suitable for prediction of lateral forces during an earthquake.

REFERENCES

1. Bar-Yam Y. *Dynamics of Complex Systems*, Addison-Wesley, 1997.
2. Gupta MM, Jin L, Homma N. *Static and Dynamic Neural Network*, John Wiley & Sons, 2003.
3. Veelenturf LPJ. *Analysis and Applications of Artificial Neural Networks*, Prentice Hall, UK, 1995.
4. Zhao Z, He W, Fan SC. Preliminary design system FOR concrete box girder bridges, *J Comput Civ Eng*, 2001; **15**(3): 184-92
5. Tang CW, Chen HJ, Yen T. Modeling confinement efficiency of reinforced concrete columns with rectilinear transverse steel using artificial neural networks, *J Struct Eng*, 2003; **129**(6): 775-83.
6. Jeng CH, Mo YL. Quick seismic response estimation of prestressed concrete bridges using artificial neural networks, *J Comput Civ Eng*, 2004; **18**(4): 360-72.
7. Gupta R, Kewalramani MA, Goel A. Prediction of concrete strength using neural-expert system, *ASCE J Mater Civ Eng*, 2006; **18**(3): 462-6.
8. Pei JS, Smyth AW. New approach to designing multilayer feedforward neural network architecture for modeling nonlinear restoring forces. I: formulation, *J Eng Mech*, 2006; **132**(12): 1290-1300.
9. Pei JS, Smyth AW. New approach to designing multilayer feedforward neural network architecture for modeling nonlinear restoring forces. Ii: applications, *J Eng Mech*, 2006; **132**(12): 1301-12.
10. Zhang Y, Zhou GC, Xiong Y, Rafiq MY, techniques for predicting cracking pattern of masonry wallet using artificial neural networks and cellular automata, *J Comput Civ Eng*, 2010; **24**(2): 21-32.
11. Ghaboussi J, Joghataie A. Active control of structure using neural networks, *J Eng Mech*, 1995; **121**(4): 556-67.
12. Bani-Hani K, Ghaboussi J. Nonlinear structural control using neural networks, *J Eng Mech*, 1998; **124**(3): 319-27.

13. Chen HM, Tsai KH, Qi GZ, Yang JCS, Amini F. Neural network for structure control, *J Comput Civ Eng*, 1995; **9**(2): 168-76.
14. Tang Y. Active control of sdf systems using artificial neural networks, *J Struct Eng*, 1996; **60**(5): 695-703.
15. Madan A. Vibration control of building structures using self-organizing and self-learning neural networks, *J Sound Vib*, 2005; **287**(4-5): 759-84.
16. Cho HC, Fadali MS, Saiidi MS, Lee KS. Neural network active control of structures with earthquake excitation, *Int J Contr Autom Syst*, 2005; **3**(2): 202-10.
17. Li HN, Yang H. system identification of dynamic structure by the multi-branch BPNN, *J Neurocomput*, 2007; **70**(4-6): 835-41.
18. Li QS, Liu DK, Leung AYT, Zhang N, Tam CM, Yang LF. Modeling of structural response and optimization of structural control system using neural network and genetic algorithm, *Struct Des. Tall Spec Build*, 2000; **9**(4): 279-93.
19. Clough RW, Penzien J. *Dynamics of Structures*, Computers and Structures Inc., 3rd edition, Berkeley, 2003.
20. Chopra AK. *Dynamics of Structures*, Prentice Hall, 4th Edition, 2007.
21. Rumelhart DE, Hinton GE, and Williams RJ. Learning representations of back-propagation errors, *Nature*, 1986; **323**(9): 533-6.
22. Haykin S. *Neural Networks, A Comprehensive Foundation*, Prentice-Hall, 2007.
23. Freeman JA, Sakpura DM. *Neural networks: algorithms, applications, and programming techniques*, Addison-Wesley, 1st Edition, 1991.
24. Matsuoka T, Hiramoto K, Sunakoda K, Ohtake T, Roschke PN. *Numerical simulation of 3-story structure with vibration control device*, 4th International Conference on Earthquake Engineering, Taipei, Taiwan, 2006, 199.