Earthq Eng & Eng Vib (2008) 7:13-24 **DOI**: 10.1007/s11803-008-0778-y

Optimal design of structures for earthquake loads by a hybrid RBF-BPSO method

Eysa Salajegheh[†], Saeed Gholizadeh[‡] and Mohsen Khatibinia[‡]

Department of Civil Engineering, University of Kerman, Kerman, Iran

Abstract: The optimal seismic design of structures requires that time history analyses (THA) be carried out repeatedly. This makes the optimal design process inefficient, in particular, if an evolutionary algorithm is used. To reduce the overall time required for structural optimization, two artificial intelligence strategies are employed. In the first strategy, radial basis function (RBF) neural networks are used to predict the time history responses of structures in the optimization flow. In the second strategy, a binary particle swarm optimization (BPSO) is used to find the optimum design. Combining the RBF and BPSO, a hybrid RBF-BPSO optimization method is proposed in this paper, which achieves fast optimization with high computational performance. Two examples are presented and compared to determine the optimal weight of structures under earthquake loadings using both exact and approximate analyses. The numerical results demonstrate the computational advantages and effectiveness of the proposed hybrid RBF-BPSO optimization method for the seismic design of structures.

Keywords: earthquake; optimization; binary particle swarm; neural networks; radial basis function

1 Introduction

Optimum design of structures is usually achieved by selecting the design variables such that an objective function is minimized while all of the design constraints are satisfied. Structural optimization requires that a structural analysis be performed many times for the specified external loads. This makes the optimal design process inefficient, especially when time history analysis (THA) is involved. This difficulty is magnified when the employed optimization method has a stochastic nature, such as evolutionary algorithms.

In recent years, several researchers (Lagaros *et al*., 2006; Zou and Chan, 2005; Kocer and Arora, 2002) have used traditional and evolutionary search techniques to optimize the seismic design of structures by using the response spectrum or THA. Salajegheh and Heidari (2005) incorporated a neural network, as an artificial intelligence technique, in the optimization process to predict structural time history responses.

In this study, a binary particle swarm optimization (BPSO) (Kennedy and Eberhart, 1997) is employed to find the optimal design of structures with dynamic constraints on the structural responses. In the BPSO algorithm, a new relationship has been proposed for updating the position of particles. This relationship changes the position of particles with respect to their previous position, and their stochastic state decreases with respect to the previous position of the particles. In the optimization process, the weight of the structure is considered as the objective function. The design variables are cross-sectional area assignments of structural elements, and the design constraints are taken as the desired time history responses of the structure. The stochastic nature of the BPSO algorithm makes the convergence of the process slow. Furthermore, evaluation of the structural responses using the finite element method during the optimization process can be computationally intensive with slow convergence. Because of the above considerations, the dynamic responses of the structures have been predicted using radial basis function (RBF) neural networks, and many successful applications have been reported (Deng, 2006; Roy and Ganguli, 2006; Gholizadeh *et al*., 2007). By such an approximation, the dynamic analysis of a structure is not necessary during the optimization process. In this work, the input is the cross-sectional property assignments of the structural members, and the output is the dynamic responses of the selected node displacement and internal stresses of members under ground shaking.

In order to perform THA and to provide training data, ANSYS (2004) is used. Also, MATLAB is utilized to design the neural networks.

Two illustrative examples are provided to investigate the efficiency of the suggested method for structural optimization subject to earthquake loads. A 72-bar space truss subjected to the El Centro (S-E 1940) earthquake and a 5-story steel shear frame subjected to the Chile

Correspondence to: Eysa Salajegheh, Department of Civil Engineering, University of Kerman, Kerman, Iran
Tel: +98 341 211 13 42; Fax: + 98 341 211 40 49
E-mail: eysasala@mail.uk.ac.ir

[†]Ph.D Student; [‡]Professor

Received October 24, 2007; **Accepted** January 25, 2008

(1985, Llolleo-N10E) earthquake are designed for optimal weight. The numerical results of optimization show that incorporating RBF neural networks in the framework of BPSO creates a powerful artificial intelligence tool for structural optimization against earthquake loads in a computationally efficient manner.

2 Formulation of optimization problem

In sizing optimization problems, the aim is usually to minimize the weight of the structure, under some constraints on stresses, displacements and frequencies. A discrete structural optimization problem can be formulated in the following form:

$$\begin{aligned}&\text{Minimize } f(X)\\&\text{Subject to } g_i(X)\le 0 \quad i=1,2,...,m \qquad (1)\\&X_j\in R^d,\ j=1,2,...,n\end{aligned}$$

where $f(X)$ represents the objective function, $g(X)$ is the behavioral constraint, and m and n are the number of constraints and design variables, respectively. A given set of discrete values is expressed by R^d and design variables X_j can take values only from this set.
The objective function is usually taken as:

$$f(X)=\sum_{i=1}^{N_e} r_i X_i L_i \qquad (2)$$

In structural optimization problems, constraints are usually taken as:

$$g_{si}(X)=\frac{S_{il}}{S_{il,\text{all}}}-1\le 0,\quad i=1,...,N_e \qquad (3)$$

$$g_{Ui}(X)=\frac{U_{jl}}{U_{jl,\text{all}}}-1\le 0,\quad j=1,...,N_j \qquad (4)$$

where r_i and L_i are the weight of unit volume and length of ith element, respectively; N_e and N_j are the number of the elements and nodes, respectively; S_{il} and U_{jl} are stress in the ith element and deflection of the jth node for the loading case l, respectively; and $S_{il,\text{all}}$ and $U_{jl,\text{all}}$ are allowable stress in the ith member and allowable deflection of the jth node for the loading case l, respectively.

3 Structural THA

The equilibrium for a finite element system subjected to an earthquake may be written in the usual form:

$$\boldsymbol{M}\ddot{\boldsymbol{U}}(t)+\boldsymbol{C}\dot{\boldsymbol{U}}(t)+\boldsymbol{K}\boldsymbol{U}(t)=-\boldsymbol{M}\boldsymbol{I}\ddot{u}_g(t) \qquad (5)$$

where $\boldsymbol{M}$, $\boldsymbol{C}$ and $\boldsymbol{K}$ are the mass, damping and stiffness matrices; $\ddot{\boldsymbol{U}}(t),\dot{\boldsymbol{U}}(t),\boldsymbol{U}(t)$ and $\boldsymbol{I}$ are the acceleration, velocity, displacement and unit vectors, respectively. Ground acceleration is shown by $\ddot{u}_g(t)$.

For analysis of the structures subjected to earthquake loading, an ANSYS based computer program is developed. The theory and solution procedures are based on the finite-element formulation of the displacement method with the nodal displacements as the unknown variables. It uses a step-by-step implicit numerical integration procedure based on Newmark's method to solve the resulting equations.

4 Dynamic constraints treatment

All of the stress and displacement constraints are time dependent. These constraints need to be imposed at each desired time instant. The consideration of all the constraints requires an enormous amount of computational effort and, therefore, treatment with a vast number of time history responses is a challenging problem for most numerical optimization algorithms (Zou and Chan, 2005). Various numerical techniques exist for treating such time-dependent constraints (Arora, 1999). The basic idea of these methods is to eliminate the time parameter from the optimization problem. In other words, a time-dependent problem is transformed into a time-independent one. In the present study, the conventional method (Arora, 1999) is employed. This method is quite simple and convenient to implement where the time interval is divided into p subintervals and the time-dependent constraints are imposed at each time grid point. Let the ith time-dependent constraint (stress or displacement) be written as:

$$g_i(X,t)\le 0,\quad 0\le t\le T \qquad (6)$$

where T is the time interval over which the constraints need to be imposed.

Because the total time interval is divided into n_{gp} subintervals, the constraint Eq. (6) is replaced by the constraints at the $n_{\text{gp}}+1$ time grid points as:

$$g_i(X,t_j)\le 0,\quad j=0,1,\cdots,n_{\text{gp}} \qquad (7)$$

The constraint function $g_i(X, t_j)$ can be evaluated at each time grid point after the structure has been analyzed and stresses and displacements have been evaluated at each time point. If fewer grid points are used, the time-dependent constraints may be violated between the grid points. Use of a finer grid can capture these points.

5 Optimization method

There are two major steps in computer implementation of the optimal design process of structures: the analysis step and the optimization step. As mentioned previously, the THA of structures is performed using Newmark's method, and the optimization is implemented by using the BPSO.

The particle swarm optimization has been inspired by the social behavior of animals such as fish schooling, insects swarming and birds flocking (Kennedy and Eberhart, 2002). It involves a number of particles, which are initialized randomly in the search space of an objective function. These particles are referred to as swarm. Each particle of the swarm represents a potential solution of the optimization problem. The particles fly through the search space and their positions are updated based on the best positions of individual particles in each iteration. The objective function is evaluated for each particle and the fitness values of particles are obtained to determine which position in the search space is the best (Bergh and Engelbrecht, 2003).

In each iteration, the swarm is updated using the following equations:

$$V_i^{k+1} = w^k V_i^k + c_1 r_1 (P_i^k - \bar{X}_i^k) + c_2 r_2 (P_{\mathrm{g}}^k - \bar{X}_i^k) \tag{8}$$

$$\bar{X}_i^{k+1} = \bar{X}_i^k + V_i^{k+1} \tag{9}$$

where $\bar{X}_i$ and V_i represent the current position and the velocity of the *i*th particle, respectively; P_i is the best previous position of the *i*th particle (called *pbest*) and P_g is the best global position among all the particles in the swarm (called *gbest*); r_1 and r_2 are two uniform random sequences generated from interval [0, 1]; w^k is the inertia weight used to discount the previous velocity of the particle preserved. Shi and Eberhart (1997) proposed that the cognitive and social scaling parameters c_1 and c_2 be selected such that $c_1 = c_2$ =2.0 to allow the product c_1r_1 or c_2r_2 to have a mean of 1. Each component of V_i is constrained to a maximum value defined as $V_i^{\max}$ and a minimum value defined as $V_i^{\min}$.

The algorithm flow can be represented as follows:

(1) Initialize

(a) Set counter $k = 0$

(b) Randomly initialize particle positions $\bar{X}_i^0$ for $i = 1,\ldots,p$

(c) Randomly initialize particle velocities $-V_i^{\max} \le V_i^0 \le V_i^{\max}$ for $i = 1,\ldots,p$

(d) Evaluate fitness function values F_i^0 using design space coordinates $\bar{X}_i^0$ for $i = 1,\ldots,p$

(e) Set $F_i^{\mathrm{best}} = F_i^0$ and $P_i^0 = \bar{X}_i^0$ for $i = 1,\ldots,p$

(f) Set $F_{\mathrm{best}}^{\mathrm{g}} = \min(F_i^{\mathrm{best}})$ and $P_{\mathrm{g}}^{\,0}$ to corresponding $\bar{X}_i^0$.

(2) While $k < k_{\max}$

(a) Update particle velocity vectors V_i^{k+1} using Eq. (8).

(b) If V_i^{k+1} for any component, then set that component to its minimum and maximum allowable value.

(c) Update particle position vectors $\bar{X}_i^{k+1}$ using Eq. (9).

(d) Evaluate fitness function values F_i^{k+1} using design space coordinates $\bar{X}_i^{k+1}$ for $i = 1,\ldots,p$

(e) If $F_{\mathrm{best}}^i < F_i^{k+1}$ then $F_{\mathrm{best}}^i = F_i^{k+1}, P_i^{k+1} = \bar{X}_i^{k+1}$ for i=1,⋯, p

(f) If $F_{\mathrm{best}}^i < F_{\mathrm{g}}^{k+1}$ then $F_{\mathrm{best}}^i = F_{\mathrm{g}}^{k+1}, P_{\mathrm{g}}^{k+1} = \bar{X}_i^{k+1}$ for i=1,⋯, p

(g) Increment k.

End while.

When the design variables are discrete, the binary model of PSO algorithm must be used. The positions of the particles in the binary model are indicated by two values: one and zero. Therefore, any particle moves in a limited space related to one and zero. So, new definitions for velocity, distance and movement path must be presented in terms of probability of being a bit in a specific position or in another one. The velocity of each particle states the probability of being its position equal to one. In binary model of PSO algorithm, Eq. (8) can be used without any changes for updating the velocity but must be brought to the interval [0, 1] as it is defined in terms of probability. For this purpose, a logistic converting function is used. For updating the particles positions, Eq. (9) is re-defined by the following rule (Kennedy and Eberhart, 1997):

$$\begin{aligned} &\text{if } \operatorname{rand}(0,1) < S(V_i^{k+1}) \quad \text{then} \\ &\qquad \bar{X}_i^{k+1} = 1 \\ &\text{else} \\ &\qquad \bar{X}_i^{k+1} = 0 \end{aligned} \tag{10}$$

where rand(0,1) is a random number in the interval [0,1] and $S(V_i^{k+1})$ is a limiting sigmoid function. The sigmoid function is expressed by (Kennedy and Eberhart, 1997):

$$S(V_i^{k+1}) = \mathrm{sig\,mod}\,e(V_i^{k+1}) = \frac{1}{1+\exp(-V_i^{k+1})} \tag{11}$$

Although the common binary PSO algorithm has been recognized as a powerful and efficient algorithm for finding the optimum solution and in spite of finding good solutions by this algorithm, its convergence properties are not good. In this paper, for resolving the difficulty mentioned, Eq. (10) is replaced by the following relationship:

$$\begin{aligned} &\text{if } \operatorname{rand}(0,1) < S(V_i^{k+1}) \quad \text{then} \\ &\qquad \bar{X}_i^{k+1} = \text{excahnge}(\bar{X}_i^k) \\ &\text{else} \\ &\qquad \bar{X}_i^{k+1} = \bar{X}_i^k \end{aligned} \tag{12}$$

In this case, the largeness of the velocities of the particles indicates their bad positions and causes them to change from zero to one, or from one to zero. The smallness of the particles velocities decreases the probabilities of changing the particles positions. Finally, when the particles velocities become equal to zero, their positions will be unchanged.

Constraints are handled by using the concept of penalty functions, which penalize infeasible solutions, i.e.,

$$f_{\mathrm{s}}(X)=\begin{cases} f(X) & if \quad X\in\tilde{\Delta} \\ f(X)+f_{\mathrm{p}}(X) & \text{otherwise} \end{cases} \tag{13}$$

where $f_{\mathrm{s}}(X)$ and $f_{\mathrm{p}}(X)$ represent supplemental and penalty functions, respectively. Also, $\tilde{\Delta}$ denotes the feasible search space.

In the case of earthquakes, a simple form of the penalty function is employed as:

$$f_{\mathrm{p}}(X,t_j)=r_{\mathrm{P}}\left[\sum_{i=1}^{N_{\mathrm{e}}}\left(\max(g_{Si}(X,t_j),0)\right)^2+\sum_{i=1}^{N_j}\left(\max(g_{Ui}(X,t_j),0)\right)^2\right]$$

$$j=0,1,\cdots,n_{\mathrm{gp}} \tag{14}$$

where r_{p} is an adjusting coefficient.

Despite the serious reducing effects of BPSO on the optimization time, the computational burden of the process due to implementing the time history dynamic analysis is very high. Therefore, using neural networks to reduce the computer effort is very effective. In the present work, RBF neural networks are employed.

6 RBF neural networks

RBF neural networks are widely used in the field of structural engineering due to their fast training, performance generality and simplicity (Zhang, 2004; Rafiq *et al.*, 2001). These networks are feed forward networks of two layers. In order to train the hidden layer of RBF networks, no training is accomplished and the transpose of the training input matrix is taken as the layer weight matrix. Also, a supervised training algorithm is employed to adjust output layer weights (Wasserman, 1993). The topology of RBF networks is shown in Fig. 1.

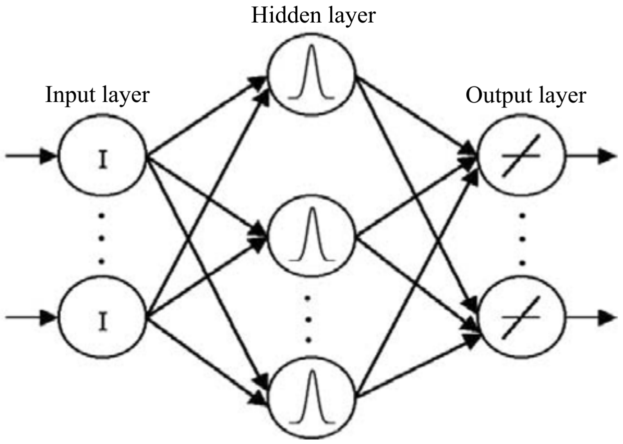


Fig. 1 Typical topology of RBF neural networks

7 Error monitoring

In order to evaluate the accuracy of approximate structural responses predicted by neural networks, two evaluation metrics are used: the relative root mean square (RRMS) error e_{RRMS} and R-*square* (R^2) statistic measurement R_{square} (Jiang *et al.*, 2006).

The RRMS error between the exact and predicted responses is defined as follows:

$$e_{\mathrm{RRMS}}=\sqrt{\frac{\frac{1}{r\text{-}1}\sum_{i=1}^{r}(\lambda_i-\tilde{\lambda}_i)^2}{\frac{1}{r}\sum_{i=1}^{r}(\lambda_i)^2}} \tag{15}$$

where λ_i and $\tilde{\lambda}_1$ are the ith component of the exact and predicted responses, respectively. The vectors dimension is expressed by r.

To measure how successful fitting is achieved between exact and approximate time history responses, the R-*square* statistic measurement is employed. Statistically, the R^2 is the square of the correlation between the predicted and the exact responses. It is defined as follows:

$$R_{\mathrm{square}}=1-\frac{\sum_{i=1}^{r}(\lambda_i-\tilde{\lambda}_i)^2}{\sum_{i=1}^{r}(\lambda_i-\bar{\lambda})^2} \tag{16}$$

where $\bar{\lambda}$ is the mean of exact vectors component.

8 Main steps of optimization

The fundamental steps in the optimization by BPSO using RBF network for earthquake loading are as follows:

(a) Selecting some particle vectors from the design variables space.

(b) Evaluating the time history responses of the structure employing RBF.

(c) Evaluating the objective function.

(d) Checking the constraints at grid points for feasibility of particle vectors.

(e) Update p_{best} and g_{best}.

(f) Update particle velocity and position.

(g) Predicting the structural time history responses for the particles using trained RBF.

(h) Evaluating the objective function.

(i) Checking the constraints at grid points; if satisfied continue, else change the vector and go to step (g).

(j) Checking convergence; if satisfied stop, else go to step (c).

As the size of populations in BPSO is small, the method is rapidly converged. It can be observed that during the optimization, the exact dynamic analysis is not needed and the necessary responses are found by the trained RBF.

9 Numerical examples

Two illustrative examples of structures are provided for achieving a minimum value of their weights. The time of optimization is computed in clock time by a personal Pentium IV 2000 MHz computer. The earthquake records are applied in the *x* direction. Young's modulus is 2.1×10^{11} Pa, and the mass density is 7850 kg/m^3. Cross-sectional properties of the members are selected from the pipe, with a radius to thickness ratio of less than 50, and box standard sections available in European profile list. The optimization is carried out by the BPSO using the following analysis methods:

(a) Exact Analysis (EA).

(b) Approximate analysis by RBF neural networks (RBF).

9.1 Space truss with 72-bars

The 72-bar truss is shown in Fig. 2. The mass of 10000 kg is lumped at nodes of 1 to 4. The truss is subjected to 15 s of the El Centro (S-E 1940) earthquake record.

Due to simplicity and practical demands, the truss members are divided into 9 groups based on cross-sectional areas, given in Table 1. The cross-sectional areas of the elements can be chosen from the values given in Table 2.

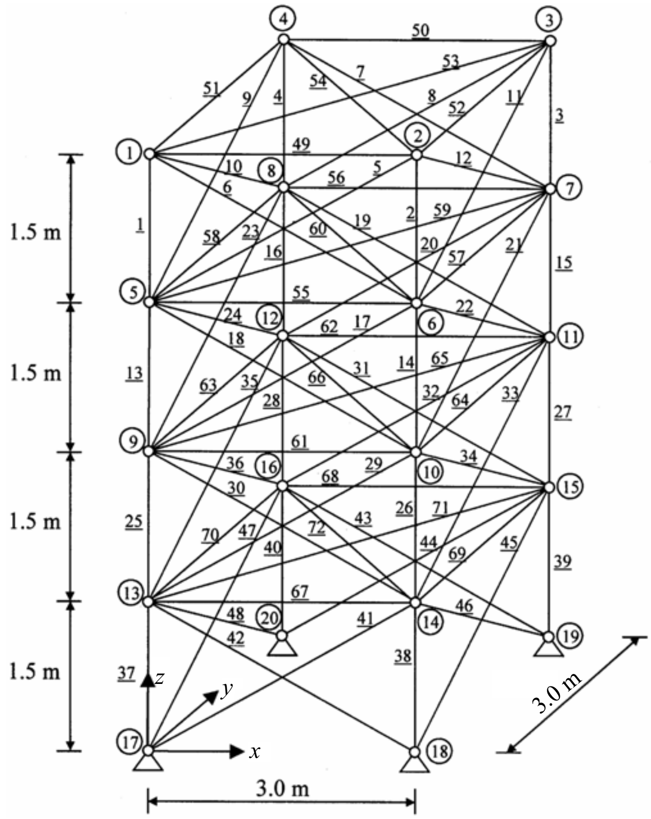


Fig. 2 Seventy two-bar space steel truss

Because of the insignificant internal stresses of elements of group 9 under the earthquake excitation, a minimum cross-sectional area of 2.54 cm^2 is assigned to them. For all the element groups, allowable stress is chosen to be 1.2×10^8 Pa. Also, for the top node of the structure, the allowable horizontal displacement is chosen to be 2 cm.

9.1.1 Data selection for training the RBF neural network

In order to train the RBF neural networks, 300 structures are randomly generated based on cross-sectional areas and are subjected to the El Centro earthquake record. Their cross-sectional areas are selected to be inputs of RBF networks. The node 1 displacement and axial stresses of element groups 1 to 8 are chosen to be the outputs of RBF networks. From the generated samples, 220 and 80 samples are employed to train and to test the performance generality of the RBF networks, respectively.

The results of testing the RBF networks are only shown for node 1 displacement and axial stress of group 7 elements in Figs. 3-6. There are similar results for other responses.

The average R_{square} and e_{RRMS} for the RBF neural networks for all training samples are 0.8956 and 0.3207, respectively. In this example, the total time for the data generation and networks training was 460 min.

9.1.2 Optimization of the 72-bar truss

Now employing the RBF networks, the 72-bar

Table 1 Element groups of the 72-bar truss

Group No.	Elements
1	1-4
2	5-12
3	13-16
4	17-24
5	25-28
6	29-36
7	37-40
8	41-48
9	49-72

Table 2 Available cross-sectional areas

No.	Area (cm^2)
1	11.2
2	12.3
3	13.9
4	15.2
5	17.2
6	18.9
7	21.4
8	25.7

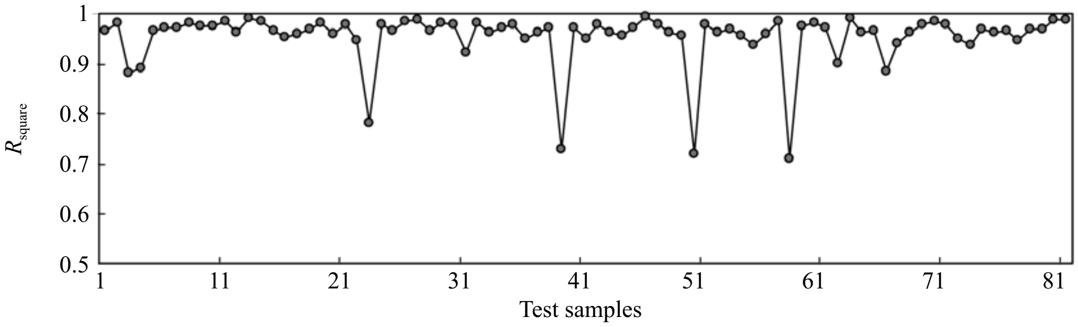


Fig. 3 R_{square} of approximate displacement of node 1

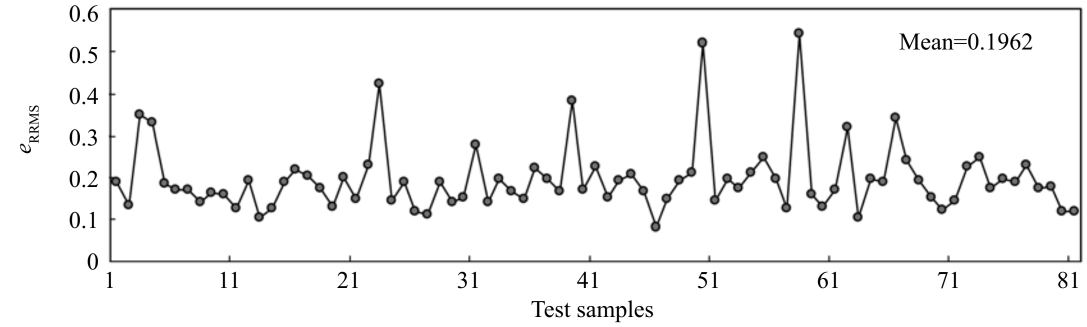


Fig. 4 e_{RRMS} of approximate displacement of node 1

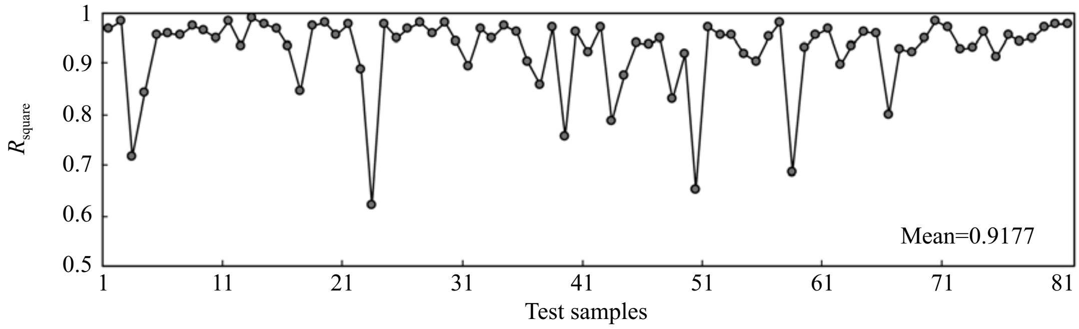


Fig. 5 R_{square} of approximate stress of group 7 elements

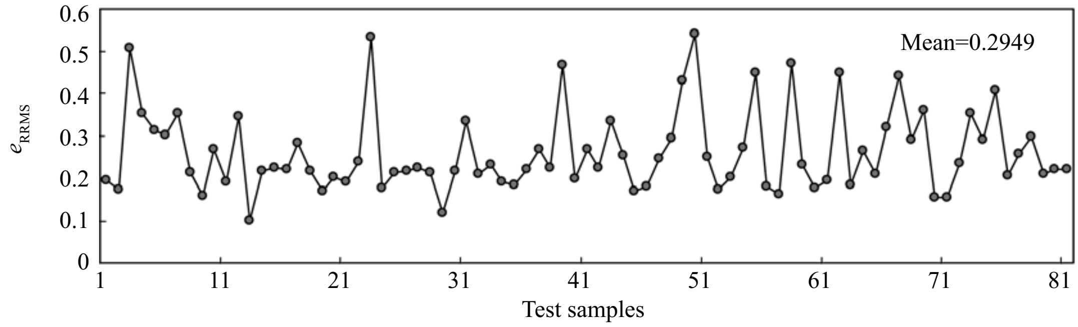


Fig. 6 e_{RRMS} of approximate stress of group 7 elements

truss is designed for optimal weight. The results of optimization using exact and approximate analysis are given in Table 3.

As given in this table, the optimum design obtained using exact analysis is better than the other solution, but the former is very extensive in terms of the optimization over all time.

Time history responses of optimum designs obtained

Table 3 Optimum designs obtained by BPSO using exact and approximate analyses

Element Groups No.	Optimum areas (cm^2)	
	EA	RBF
1	11.20	11.20
2	11.20	12.30
3	17.20	15.20
4	11.20	11.20
5	25.70	25.70
6	11.20	12.30
7	25.70	25.70
8	11.20	11.20
9	2.54	2.54
Mass (kg)	1506.60	1544.11
Generations	57	70
Time (min)	2538.0	6.4

using approximate analysis for displacements of nodel and some typical element groups are compared with their corresponding actual ones as shown in Figs. 7-8. A brief summery is given in Table 4. The comparisons reveal the appropriate conformance between all of the approximate and corresponding actual responses.

It is important to note that in this example, the time of optimization employing neural network including data generation and training the neural networks is about 0.18 time of the optimization with exact THA.

9.2 Five-story steel shear frame

The steel frame structure is shown in Fig. 9. Mass of 45.38 t is considered on the diaphragms. The structure is subjected to the Chile (1985, Llolleo-N10E) earthquake record.

It is assumed that all the columns in each story have the same properties. Cross-sectional properties of the columns are selected from the discrete values listed in Table 5. Also, an IPE 500 profile is assigned to all the beams. Therefore, in this example, only the cross-sectional properties of the columns are taken into account as the design variables.

For stories 1 to 5, the allowable inter-story drift ratio is taken as 1/300. The constraints are checked at 12001 grid points where the time interval between adjacent points is 0.005 s.

9.2.1 Data selection for training the RBF neural network

In this example, 150 structures are randomly generated based on cross-sectional properties and are analyzed for the Chile earthquake induced loads. The design variables (cross-sectional properties of the columns) and the inter-story drifts of all stories are treated as the inputs and outputs of RBF networks, respectively. From the generated samples, 100 and 50 are used to train and test the performance generality of the RBF networks, respectively.

The results of testing the RBF networks, in terms of R_{square} and e_{RRMS}, are only shown for inter-story drifts of stories 2 and 4 in Figs. 10-11, respectively. There are similar results for the other responses.

The average R_{square} and e_{RRMS} for the RBF neural networks for all training samples are 0.9991 and 0.0269, respectively. In this example, the total time spending to data generation and networks training is equal to 170 min.

9.2.2 Optimization of the 5-story steel shear frame

The results of 5-story steel shear frame optimization using exact and approximate analysis are given in Table 6. As listed in this table, the optimum design obtained using exact analysis is better than the other solution, but it is very extensive in terms of the optimization over all time.

Time history responses of optimum designs obtained using approximate analysis are compared with their corresponding actual ones in Fig. 12. A brief summery is given in Table 7. The comparisons reveal appropriate conformance between all of the approximate and corresponding actual responses.

In this example the time of optimization employing neural networks, including data generation and training the neural networks, is about 0.15 times of optimization with exact THA.

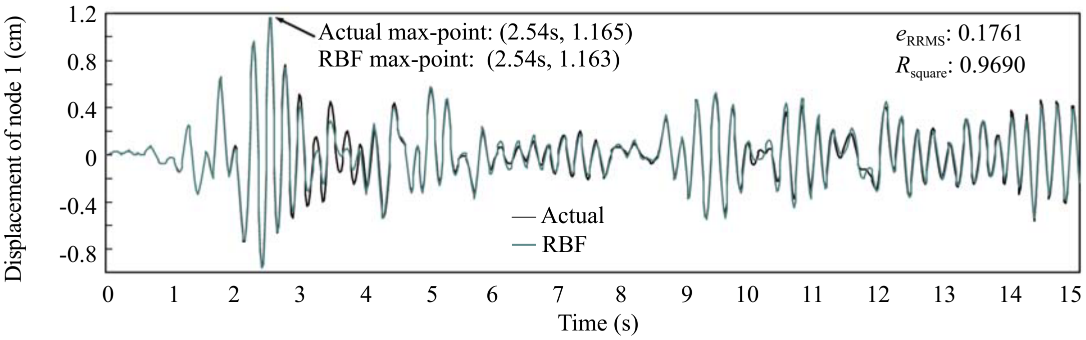
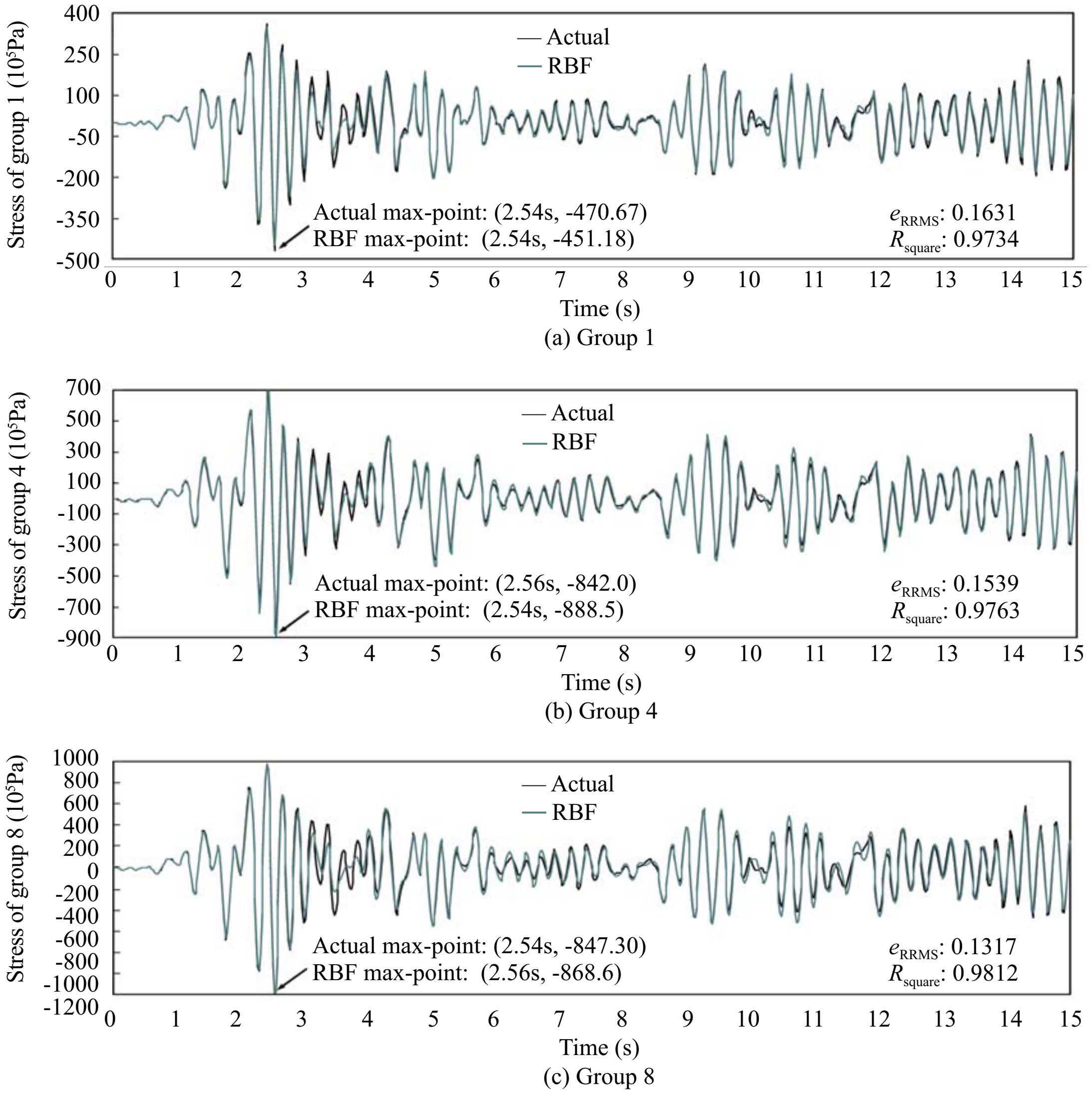


Fig. 7 Displacement of node 1 of the optimized 72-bar space steel truss



(a) Group 1

(b) Group 4

(c) Group 8

Fig. 8 Stress responses in different element groups of the optimized 72-bar space steel truss

Table 4 Mean R_{square} and mean e_{RRMS} of optimum designs

Structural parameters	RBF	
	R_{square}	e_{RRMS}
Node 1 displacement	0.9690	0.1761
Group 1 elements	0.9734	0.1631
Group 2 elements	0.9745	0.1596
Group 3 elements	0.8701	0.3604
Group 4 elements	0.9763	0.1539
Group 5 elements	0.9341	0.2567
Group 6 elements	0.9734	0.1630
Group 7 elements	0.9459	0.2327
Group 8 elements	0.9812	0.1371
Average	0.9553	0.2003

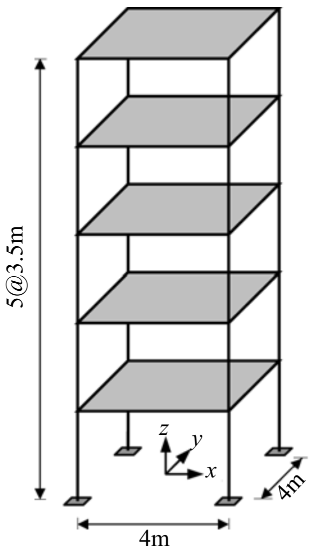


Fig. 9 Five-story steel shear frame

Table 5 Available box cross-sections

No.	Moment of inertia (cm^4)	Area (cm^2)
1	67272.60	210.24
2	71983.92	215.04
3	75801.52	244.16
4	77436.16	264.96
5	81422.47	271.36
6	84081.62	277.76
7	87314.78	296.64
8	90345.81	303.84

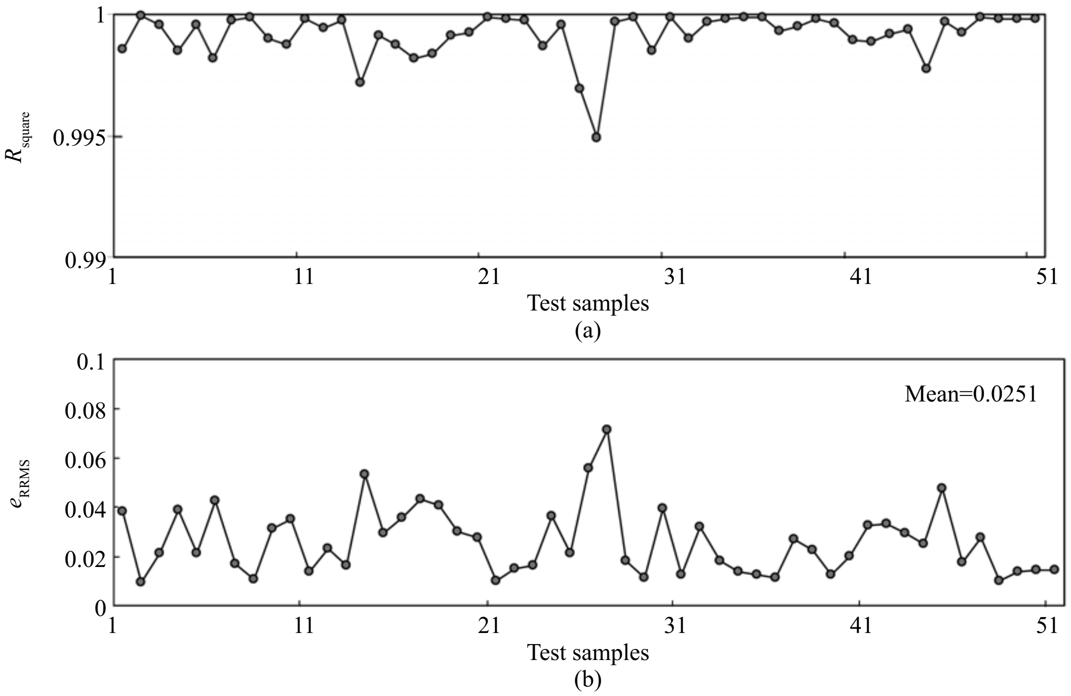


Fig. 10 R_{square} (a) and e_{RRMS} (b) of the approximate inter-story drift of story 2

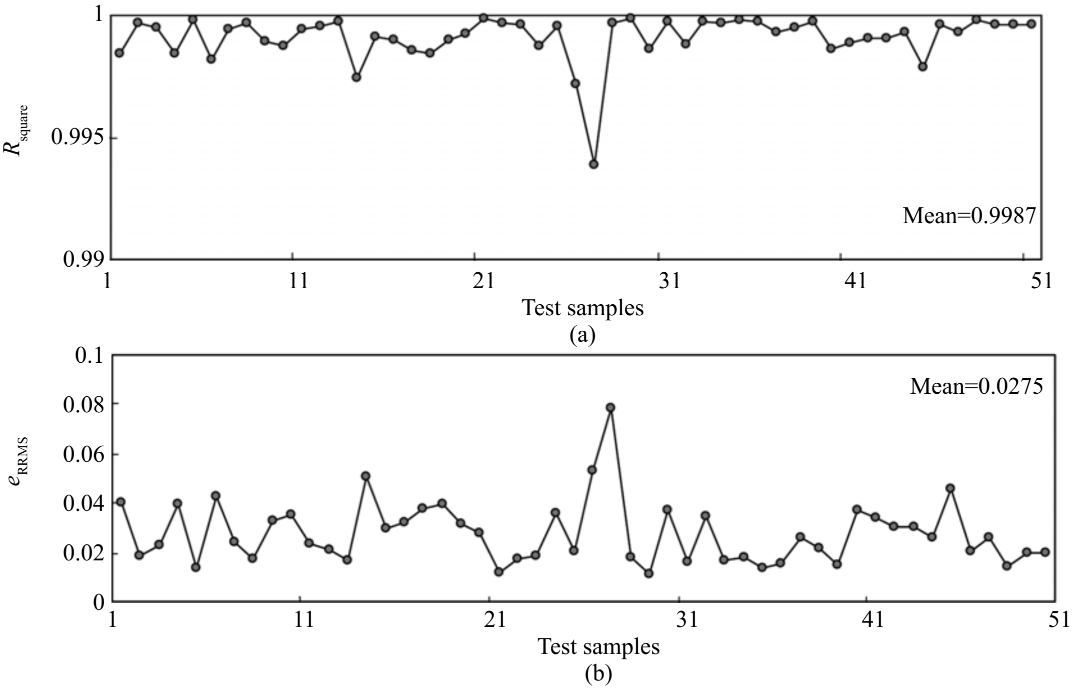
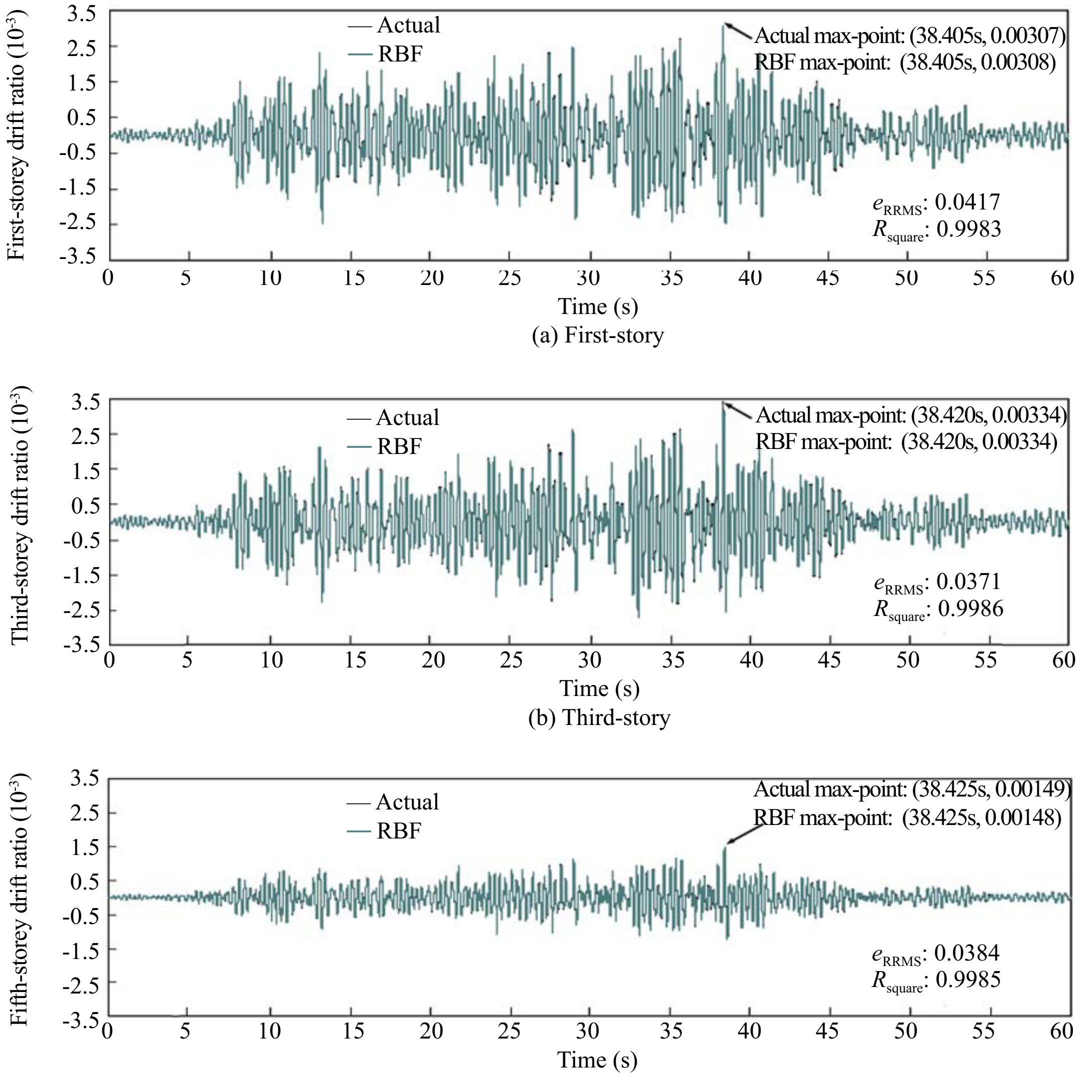


Fig. 11 R_{square} (a) and e_{RRMS} (b) of the approximate inter-story drift of story 4

Table 6 Optimum designs obtained by BPSO using exact and approximate analyses

Story No.	Optimum cross-sectional properties			
	EA		RBF	
	Moment of inertia (cm^4)	Area (cm^2)	Moment of inertia (cm^4)	Area (cm^2)
1	90345.81	303.84	90345.81	303.84
2	84081.62	277.76	81422.47	271.36
3	67272.60	210.24	71983.92	215.04
4	67272.60	210.24	67272.60	210.24
5	67272.60	210.24	67272.60	210.24
Mass (kg)	13530.89		13361.20	
Generations	60		50	
Time (min)	1342.0		5.3	



(a) First-story

(b) Third-story

(c) Fifth-story

Fig. 12 Drift ratios of different stories of the optimized steel frame

Table 7 Mean R_{square} and mean e_{RRMS} of optimum designs

Structural parameters	RBF	
	R_{square}	e_{RRMS}
First-story drift ratio	0.9983	0.0417
Second-story drift ratio	0.9983	0.0411
Third-story drift ratio	0.9986	0.0371
Fourth-story drift ratio	0.9986	0.0369
Fifth-story drift ratio	0.9985	0.0384
Average.	0.9985	0.0390

10 Conclusions

An efficient optimization procedure has been developed for the optimal design of structures subjected to earthquakes using discrete design variables. In the procedure, a combination of the evolutionary algorithm and neural networks, as two artificial intelligence techniques, has been utilized. The employed evolutionary algorithm is a binary particle swarm optimization (BPSO) method. In this paper, a new relationship has been proposed for updating the position of the particles. This relationship changes the position of the particles with respect to their previous position. This relationship causes the BPSO algorithm to converge quickly and thus, the probability of achieving the global optimization is increased. Moreover, performing the structural optimization using the exact time history analysis for earthquake induced loads imposes a very large computational burden on the optimization process. That is, in each design point of the desired earthquake, the structure should be analyzed to evaluate the necessary responses. In order to reduce the computational effort of the optimization process due to the performing THA, radial basis function (RBF) neural networks are employed to approximate the structural time history responses. A simple method is used to treat the dynamic constraints. In this method, the time interval is divided into some subintervals and constraints are imposed at each time grid point. The numerical results of the optimization show that in the proposed method, the time of optimization including training time is reduced to about 0.2 of the time required for exact optimization; and the errors are small.

References

ANSYS Incorporated (2004), "*ANSYS Release 8.1*".

Arora JS (1999), "Optimization of Structures Subjected to Dynamic Loads," Leondes CT, ed. *Structural dynamic systems computational techniques and optimization*. Gordon and Breach Science Publishers.

Bergh V and Engelbrecht A (2003), "Using Neighbourhood with the Guaranteed Convergence PSO," *2003 IEEE Swarm Intelligence Symposium*, USA, pp.235-242.

Deng J (2006), "Structural Reliability Analysis for Implicit Performance Function Using Radial Basis Function Network," *International Journal of Solids and Structures*, **43:** 3255-3291.

Gholizadeh S, Salajegheh E and Torkzadeh P (2008), "Structural Optimization with Frequency Constraints by Wavelet Radial Basis Function Neural Networks," *Journal of Sound and Vibration*, **312**(1-2): 316-331.

Haykin S (1994), *Neural Networks: A Comprehensive Foundation*, New York: Macmillan.

Jiang X, Mahadevan S and Adeli H (2006) "Bayesian Wavelet Packet Denoising for Structural System Identification," *Structural Control and Health Monitoring*, **14:** 333-356.

Jung S and Ghabousi J (2006), "Neural Network Constitutive Model for Rate-dependent Materials," *Computers & Structures*, **84**(15-16): 955-963.

Kennedy J and Eberhart RC (1997), "A Discrete Binary Version of the Particle Swarm Algorithm," *Proceeding Conference Systems, Man, and Cybernetics*, Piscataway, NJ, pp. 4104-4108.

Kennedy J and Eberhart RC (2002), "Swarm Intelligence," Morgan Kaufman Publishers.

Kocer FY and Arora JS (2002), "Optimal Design of Latticed Towers Subjected to Earthquake Loading," *Journal of structural Engineering*, ASCE, **128**(2): 197-204.

Lagaros ND, Fragiadakis M, Papadrakakis M and Tsompanakis Y (2006), "Structural Optimization: A Tool for Evaluating Seismic Design Procedures," *Engineering Structures*, **28**(12): 1623-1633.

Madan A (2005), "Vibration Control of Building Structures Using Self-organizing and Self-learning Neural Networks," *Journal of Sound and Vibration*, **287**(4-5): 759-784.

Pu Y and Mesbahi E (2006), "Application of Artificial Neural Networks to Evaluation of Ultimate Strength of Steel Panels," *Engineering Structures*, **28**(8): 1190-1196.

Rafiq MY, Bugmann G and Easterbrook DJ (2001), "Neural Network Design for Engineering Applications," *Computers & Structures*, **79**(17): 1541-1552.

Roy N and Ganguli R (2006), "Filter Design using Radial Basis Function Neural Network and Genetic Algorithm for Improved Operational Health Monitoring," *Applied Soft Computing*, **6:** 154-169

Salajegheh E and Heidari A (2005), "Optimum Design of Structures Against Earthquake by Wavelet Neural Network and Filter Banks," *Earthquake Engineering & Structural Dynamics*, **34:** 67-82.

Shi Y and Eberhart RC (1997), "A Modified Particle Swarm Optimizer," *Proceeding IEEE International Conference on Evolutionary Computation*, pp.303-308.

The Language of Technical Computing (2006), "*MATLAB*", Math Works Inc.

Wasserman PD (1993), "*Advanced Methods in Neural Computing*," New York: Prentice Hall.

Yong L and Zhenguo T (2004), "A two-level neural network approach for dynamic FE model updating including damping," *Journal of Sound and Vibration*, **275**(3-5): 931-952.

Zhang, L (2004), "RBF Neural Networks for the Prediction of Building Interference Effects," *Computers & Structures*, **82**(27): 2333-2339.

Zhang QZ, Gan WS and Zhou YL (2006), "Adaptive Recurrent Fuzzy Neural Networks for Active Noise Control," *Journal of Sound and Vibration*, **296**(4-5): 935-948.

Zou XK and Chan CM (2005), "An Optimal Resizing Technique for Seismic Drift Design of Concrete Buildings Subjected to Response Spectrum and Time History Loadings," *Computers & Structures*, **83**(19-20): 1689-1704.