DAMAGE DETECTION OF SPACE STRUCTURES USING CHARGED SYSTEM SEARCH ALGORITHM AND RESIDUAL FORCE METHOD[*]

M. SABERI[1] AND A. KAVEH[2][**]

[1]Dept. of Engineering, Science and Research Branch, Islamic Azad University, Tehran, I. R. of Iran
[2]Centre of Excellence for Fundamental Studies in Structural Engineering, Iran University of Science and Technology, Narmak, Tehran16, I. R. of Iran
Email: alikaveh@iust.ac.ir

Abstract– Changes in dynamic properties of structures indicate occurrence of damages in the structures. In this paper, a new two stage algorithm for damage detection of large structure is introduced. A modified Residual Force Method is utilized to locate the damage regions in structures, especially barrel vaults, and then the enhanced Charged System Search (CSS) Algorithm is used to quantify the amount of damage. The proposed method requires modeling the structure in its undamaged state to obtain the dynamic properties such as frequencies and mode shapes of the structure in its damaged state. The validation of the method is investigated by some numerical study on space structures. It is shown that if dynamic properties are not affected by noise, the method can still identify damaged elements, but when measurements are corrupted by noise, the exact determination of damaged region is not possible. To overcome this problem, it is proposed to repeat measuring modal properties several times, and then with utilizing residual force vector and statistical analysis, the damaged region is predicted accurately. In this paper, a modified vector is utilized for enhancing efficiency of the modal Residual Force method and suppressing effect of noise, and then a threshold is defined to distinguish DOFs associated with damaged elements. Then, using the CSS algorithm, severities of damaged elements are assessed.

1. INTRODUCTION

Maintenance of important structures has recently become a major challenge because of continuous increase of the social and economic costs of large structures. This becomes more important when a structure is subjected to seismic loading. Different methods are available for damage detection in structures.

Damage identification can be categorized into local and global methods. In the local case, the assessment of the state of structure is performed either by direct visual inspection or using experimental technique such as acoustic emission, ultrasonic, magnetic particle inspection, radiography and eddy current. For local inspection it is necessary to know about the damaged zone but this is difficult because damaged zones are covered either by fire protection or decorative materials. Thus, damage identification from their overall assessment has received much attention over the past two decades. Vibration-based methods generally have four levels: Excitation, Sensing, Signal Processing and Identification Algorithm [1].

Damage in structures changes the dynamic properties of the structures. Some identification methods are improved based on dynamic properties of damaged structures. These methods can be categorized into

---

two approaches: the first approach relates the variation in strain energy of the structure to the changes in structural frequencies [2, 3], and the second approach relates the change of stiffness and mass properties of the structures to the changes in structural frequencies and mode shapes [4].

Four levels of damage identification defined by Ritter [5] are as follows:

- Level 1: Determining whether the damage has occurred in the structure,
- Level 2: Determination of the geometric location of the damage,
- Level 3: Quantification of the severity of the damage,
- Level 4: Prediction of the remaining service life of the structure.

Assessment of methods can determine whether only the existence of damage in a structure is relatively simple, since this can be done by experimental data of the undamaged structure, without considering the damage state [1]. However, assessment of methods consisting of Levels 2 and 3 requires a finite element modeling, and the reproduction of a damage scenario is rather complex.

Maity and Tripayhy [6] used the genetic algorithm for detection of structural damage by utilizing the changes in natural frequencies. Gerist et al. [7] employed Basis Pursuit (BP) technique for increasing the performance of the genetic algorithm by generating initial population for this algorithm. Aktasoglu et al. [8] used both non-linear optimization and genetic algorithm for damage detection and utilized changing natural frequencies and mode shapes for objective function. Ge and Lui [4] utilized a new technique based on residual force method.

Damage identification in large structures with numerous elements is difficult, therefore utilizing only optimization methods does not lead to the desired result. Therefore, in these structures for increasing efficiency, the damaged region is limited with some techniques, and a two-stage process is usually utilized to detect damaged elements. Naseralavi at el. [9] presented an efficient method for structural damage detection using natural frequencies. The method was based on sensitivity analysis of the structure, consisting of two main stages. In the first stage, the structural elements were ordered based on their damage probability into a vector, referred to as elements damage probability ordering vector (EDPOV). In the second stage, a rather small subset of EDPOV elements was judiciously selected to form a nonlinear system of equations, which were subsequently solved to detect potential damages.

Salajegheh et al. [10, 11] used a new strategy with two aims: 1) reducing search space by elimination of some design variables during optimization process, 2) improving each individual by solving the linearized problem using Moore-Penrose pseudo inverse at the end of reproduction of genetic algorithm. In another paper, kinetic strain energy was employed to determine the location of structural damages. After determining the suspected damage locations, the severity of damages was obtained based on variations of modal strain energy between the analytical models and the responses measured in damaged models using time history dynamic analysis data.

Salajegheh at el. [12] introduced pseudo-eigenvectors because detection techniques are rendered ineffective in dome structures. The proposed pseudo-eigenvectors could be linearly approximated, and hence they adopted a sensitivity-based analysis.

Using single stage methods for large-scale structures does not lead to the desired result because of vast search space, also all of the techniques based on two stage utilizes sensitivity-based analysis and need to use static displacement, but measuring displacement in some space structures such as two-layer barrel vaults is difficult because of high stiffness, so in this study a new two stage method is provided based on dynamic properties. In this method, the damaged elements are detected only by measuring some dynamic properties.

In this paper, Residual Force Method and the recently developed Optimization Algorithm (CSS) are applied to identify damage in large structures, especially in space structures. An improved Charged System Search (CSS) Algorithm [13] is utilized to minimize the objective function. This algorithm with local and global search tools has efficient performance for our optimization problem.

The remainder of this paper is organized as follows: Section 2 briefly presents FE modeling. The CSS algorithm is introduced in Section 3. The objective function is presented in Section 4. The Residual Force Method is presented in Section 5 and the procedure of the presented method is provided in Section 6. Numerical study is the content of Section 7. Finally, the concluding remarks are provided in Section 8.

2. FINITE ELEMENT MODEL

The stiffness and mass matrices of a space truss elements in global coordinate can be expressed as [14].

$$K_e = T^t k_e T \tag{1}$$

$$= \begin{bmatrix} l_{ij}^2 & l_{ij}m_{ij} & l_{ij}n_{ij} & -l_{ij}^2 & l_{ij}m_{ij} & l_{ij}n_{ij} \\ l_{ij}m_{ij} & m_{ij}^2 & m_{ij}n_{ij} & -l_{ij}m_{ij} & m_{ij}^2 & -m_{ij}n_{ij} \\ l_{ij}n_{ij} & m_{ij}n_{ij} & n_{ij}^2 & l_{ij}n_{ij} & -m_{ij}n_{ij} & -n_{ij}^2 \\ -l_{ij}^2 & -l_{ij}m_{ij} & -l_{ij}n_{ij} & l_{ij}^2 & l_{ij}m_{ij} & l_{ij}n_{ij} \\ -l_{ij}m_{ij} & -m_{ij}^2 & -m_{ij}n_{ij} & l_{ij}m_{ij} & m_{ij}^2 & m_{ij}n_{ij} \\ -l_{ij}n_{ij} & -m_{ij}n_{ij} & -n_{ij}^2 & l_{ij}n_{ij} & m_{ij}n_{ij} & n_{ij}^2 \end{bmatrix}$$

and

$$M_e = T^t m_e T \tag{2}$$

$$= \frac{\rho AL}{6} \begin{bmatrix} 2l_{ij}^2 & 2l_{ij}m_{ij} & 2l_{ij}n_{ij} & l_{ij}^2 & l_{ij}m_{ij} & l_{ij}n_{ij} \\ 2l_{ij}m_{ij} & 2m_{ij}^2 & 2m_{ij}n_{ij} & l_{ij}m_{ij} & m_{ij}^2 & m_{ij}n_{ij} \\ 2l_{ij}n_{ij} & 2m_{ij}n_{ij} & 2n_{ij}^2 & l_{ij}n_{ij} & m_{ij}n_{ij} & n_{ij}^2 \\ l_{ij}^2 & l_{ij}m_{ij} & l_{ij}n_{ij} & 2l_{ij}^2 & 2l_{ij}m_{ij} & 2l_{ij}n_{ij} \\ l_{ij}m_{ij} & m_{ij}^2 & m_{ij}n_{ij} & 2l_{ij}m_{ij} & 2m_{ij}^2 & 2m_{ij}n_{ij} \\ l_{ij}n_{ij} & m_{ij}n_{ij} & n_{ij}^2 & 2l_{ij}n_{ij} & 2m_{ij}n_{ij} & 2n_{ij}^2 \end{bmatrix}$$

In which

$$l_{ij} = \cos(x, X) = \frac{X_j - X_i}{L} \tag{3}$$

$$m_{ij} = \cos(x, Y) = \frac{Y_j - Y_i}{L}$$

$$n_{ij} = \cos(x, Z) = \frac{Z_j - Z_i}{L}$$

Here A, E, L, I, T and ρ are cross-sectional area, modulus of elasticity, length, second moment of inertia, an orthogonal matrix and material density of the member, respectively. The length of the member can be calculated using the global coordinates of the two end nodes of the member by

$$L = \sqrt{(X_j - X_i)^2 + (Z_j - Z_j)^2 + (Z_j - Z_j)^2} \tag{4}$$

3. OPTIMIZATION ALGORITHM

Charged system search algorithm was first introduced by Kaveh and Talatahari [13]. Here, for increasing the performance of the algorithm some modifications are made to enhance the algorithm. In this section, first the standard CSS is introduced in a concise form and then the enhanced version is presented. Some other applications of CSS may be found in [14, 15].

a) *The standard CSS*

Charged system search is a population based meta-heuristic algorithm proposed by Kaveh and Talatahari [13]. This algorithm is based on laws from electrostatic of physics and Newtonian mechanics. The coulomb and Gauss laws provide the magnitude of the electric field at a point inside and outside a charged insulating solid sphere as:

$$E_{ij} = \begin{cases} \frac{k_e q_i}{a^3} r_{ij} & if \quad r_{ij} < a \\ \frac{k_e q_i}{r_{ij}^2} & if \quad r_{ij} \geq a \end{cases} \tag{5}$$

Where k_e is a constant known as the Coulomb constant; r_{ij} is the separation of the centre of sphere and the selected point; q_i is the magnitude of the charge; and a is the radius of the charged sphere. Using the principle of superposition, the resulting electric force due to *N* charged spheres is equal to:

$$F_j = k_{eq} \sum_{i=1}^{n} (\frac{q_i}{a^3} r_{ij} . i_1 + \frac{q_i}{r_{ij}^2} r_{ij} . i_2) \frac{r_i - r_j}{|r_i - r_j|} \begin{cases} i_1 = 1, \ i_2 = 0 \leftrightarrow r_{ij} < a \\ i_1 = 0, \ i_2 = 1 \leftrightarrow r_{ij} \geq a \end{cases} \tag{6}$$

Also, according to Newtonian mechanics, we have:

$$\Delta r = r_{new} - r_{old} \tag{7}$$

$$v = \frac{r_{new} - r_{old}}{\Delta t}$$

$$a = \frac{v_{new} - v_{old}}{\Delta t}$$

Where r_{old} and r_{new} are the initial and final positions of the particle, respectively; v is the velocity of the particle; and a is the acceleration of the particle. By combining the above equations and using Newton's second law, the displacement of any object as a function of time can be obtained as:

$$r_{new} = \frac{1}{2} \frac{F}{M} \Delta t^2 + v_{old} \Delta t + r_{old} \tag{8}$$

By the above electrostatic and Newtonian mechanics laws, the pseudo-code of the CSS algorithm is presented as follows:

Level 1: Initialization

Step 1. *Initialization*. Initialize the parameters of the CSS algorithm. Initialize an array of charged particles (CPs) with random positions. The initial velocities of the CPs are taken as zero. Each CP has a charge of magnitude, q, defined considering the quality of its solution as:

$$q_i = \frac{fit(i) - fit_{worst}}{fit_{best} - fit_{worst}} \quad i = 1,2,...,n \tag{9}$$

Where fit_{best} and fit_{worst} are best and the worst fitness of all the particles; $fit(i)$ represents the fitness of agent *i*. The separation distance r_{ij} between two charged particles is defined as:

$$r_{ij} = \frac{\|X_i - X_j\|}{\left\| \frac{(X_i + X_j)}{2} - X_{best} \right\| + \varepsilon} \tag{10}$$

where X_i and X_j are the position of the *i*th and *j*th CPs, respectively; X_{best} is the position of the best current CP; and ε is a small positive to avoid singularities.

Step 2. *CP ranking*. Evaluate the values of the fitness function for the CPs, compare them with each other and sort them in ascending order.

Step 3. *CM creation*. Store the number of the first CPs equal to charged memory size (CMS) and their related values of fitness functions in the charged memory (CM).

Level 2: Search

Step 1. Attracting force determination. Determine the probability of moving each CP toward the others considering the following probability function:

$$P_{ij} = \begin{cases} 1 & \frac{fit(i)-fit_{best}}{fit(j)-fit(i)} > rand \quad \forall \quad fit(i) > fit(j) \\ 0 & else \end{cases} \qquad (11)$$

And calculate the attracting force vector for each CP as follows:

$$F_{ij} = q_j \sum_{i,i \neq j} \left(\frac{q_i}{a^3} r_{ij}.i_1 + \frac{q_i}{r_{ij}^2} r_{ij}.i_2 \right) p_{ij} (X_i - X_j) \begin{cases} j = 1,2,\dots,n \\ i_1 = 1, i_2 = 0 \leftrightarrow r_{ij} < a \\ i_1 = 1, i_2 = 0 \leftrightarrow r_{ij} \geq a \end{cases} \qquad (12)$$

where F_{ij} is the resultant force affecting the jth CP.

Step 2. Solution construction. Move each CP to the new position and find its velocity using the following equation:

$$X_{j,new} = rand_{j1}.k_a.\frac{F_j}{m_j} \Delta t^2 + rand_{j2}.k_v.v_{j,old} + X_{j,old} \qquad (13)$$

$$V_{j,new} = \frac{X_{j,new} - X_{j,old}}{\Delta t}$$

Where $rand_{j1}$ and $rand_{j2}$ are two random numbers uniformly distributed in the range (1,0); m_j is the mass of the CPs, which is equal to q_i in this paper. Δt is the time step, and it is set to 1. k_a is the acceleration coefficient; k_v is the velocity coefficient to control the influence of the previous velocity. Here, k_a and k_v are taken as:

$$k_a = c_1 \left(1 + \frac{iter}{iter_{max}} \right) \qquad (14)$$

$$k_v = c_2 \left(1 - \frac{iter}{iter_{max}} \right)$$

where c_1 and c_2 are two constants to control the exploitation and exploration of the algorithm; *iter* is iteration number and $iter_{max}$ is the maximum number of iterations.

Step 3. CP position correction. If each CP exits from the allowable search space, correct its position using the HS-based handling as described by Kaveh and Talatahari [13].

Step 4. CP ranking. Evaluate and compare the values of the fitness function for the new CPs; and sort them in an ascending order.

Step 5. CM updating. If some new CP vectors are better than the worst ones in the CM, in terms of their objective function values, include the better vectors in the CM and exclude the worst ones from the CM.

Level 3: Controlling the terminating criterion
Repeat the search level steps until a specified terminating criterion is satisfied.

a) Enhanced CSS algorithm

An alternative to the conventional mathematical programming approaches, meta-heuristic optimization techniques have been used to obtain global or near-global optimum solutions. Due to their capability of exploring and finding promising regions in the search space in an affordable time, these methods are quite suitable for global searches and furthermore, alleviate the need for continuous cost function and variables used for mathematical optimization methods. Though these are approximate methods, their solutions are quite satisfactory, but not necessarily optimal. Therefore, the algorithm may be repeated to obtain an adequate result. In this study, for improving the performance of the CSS algorithm, optimal result in each execution of the optimization algorithm is added to the input in the subsequent iteration. In other words, optimal result of the search process in each stage is used for subsequent stage, for improving the solution. For better comparison, the standard CSS flowchart is

depicted in Fig. 1, and that of the enhanced CSS is illustrated in Fig. 2, and the difference between two processes is designated with asterisks.

Initialization

Step1	Step2	Step3
Initialize specification of optimization Problem and algorithm parameters, Initialize the initial position of charged particales with random positions and their associated velocities.	Find the values of the objective function for the CPs, compare with each other and sort in the increasing order.	Generate the initial charged memory [some of the best solution vectors] (as many as CMS)

Search

Step1	Step 2	Step 3
Determine the probability of moving and the attracting force vector for each CP.	Determine the new positions and velocities of the CPs.	If a cp swerves the side limits, correct its position using HS-based algorithm.

Step 5	Are the new CPs better than the stored ones in CM?	Step 4
Include the better vectors in the CM and exclude the worst ones from the CM.	**Yes** →	Evaluate the objective function and rank the CPs according to their quality.

No

Termination criterion?

No →

Yes

Stop

Fig. 1. The flowchart of the standard CSS [13]

Initialization

Step1*	Step2	Step3*
Initialize algorithm parameters [first iteration], initialize the initial position of charged particles with random positions and their associated velocities, then add all percales from CMS into initial particles[in first iteration CMS is empty].	Find the values of the objective function for the CPs, compare with each other and sort in the increasing order.	Include the best solution vector into CM memory.

Search

Step1	Step 2	Step 3
Determine the probability of moving and the attracting force vector for each CP.	Determine the new positions and velocities of the CPs.	If a cp swerves the side limits, correct its position using HS-based algorithm.

Step 5*	Are the new CPs better than the stored ones in CM?	Step 4*
Include the best vector in the CM and exclude the worst one from the CM.	**Yes** →	Evaluate the objective function and rank the CPs according to their quality. and eliminate duplicate ones

No

Termination criterion?

No →

Yes

Step 7*	Step 6*
Increment the counter.	Add CP from CM memory into CMS memory [CM memory will be empty].

Is the counter less than threshold?

Yes →

No

Stop

Fig. 2. The flowchart of the enhanced CSS

4. OBJECTIVE FUNCTION

The objective function can be considered as the difference between modal properties of the initial and modeled structure. Kaveh and Zolgadr [17] have used the difference of the frequencies as:

$$F(X) = \sqrt{\frac{1}{n}\Sigma(f_i^a - f_i^c)^2} \tag{15}$$

where X is a solution vector; and n is the number of modes; f_i^a is the frequency of initial structure (measured frequency); f_i^c is the frequency of modeled structure.

It is known that the objective function based on natural frequencies and mode shapes are more efficient [18, 19]. In this study, for increasing the sensitivity of the objective function, both frequencies and mode shapes are used in non-dimensional form as follows:

$$F(X) = \left|\frac{w_j^{opt} - w_j^{ex}}{w_j^{ex}}\right| + \omega\sqrt{\frac{\Sigma_{i=1}^{ndf}(\emptyset_{ij}^{opt} - \emptyset_{ij}^{ex})^2}{\Sigma_i^{ndf}(\emptyset_{ij}^{ex})^2}} \tag{16}$$

where ndf is the number of DOF with available experimental data; f_j^{opt} and f_j^{ex} are the *j*th frequency of modeled and initial structure, respectively; $\emptyset_{ij}^{opt}$ and $\emptyset_{ij}^{ex}$ are the *j*th displacement or curvature mode shape in the *i*th DOF of the modeled and initial structure, respectively.

This new objective function has two non-dimensions parts. Part one is related to frequency and part two is related to mod shapes. The sensitivity of frequency and mod shape are not identical in detecting damage, and depend on type of structure and damage. Thus it seems to be logical to introduce a weight factor ω to change the amount of their contributions. In general, the weight factor is determined using numerical study and it is considered constant in some cases, but in this study, a process is introduced to enable determination of the value of weight factor via an optimization process. The flowchart of the optimization is shown in Fig. 3.
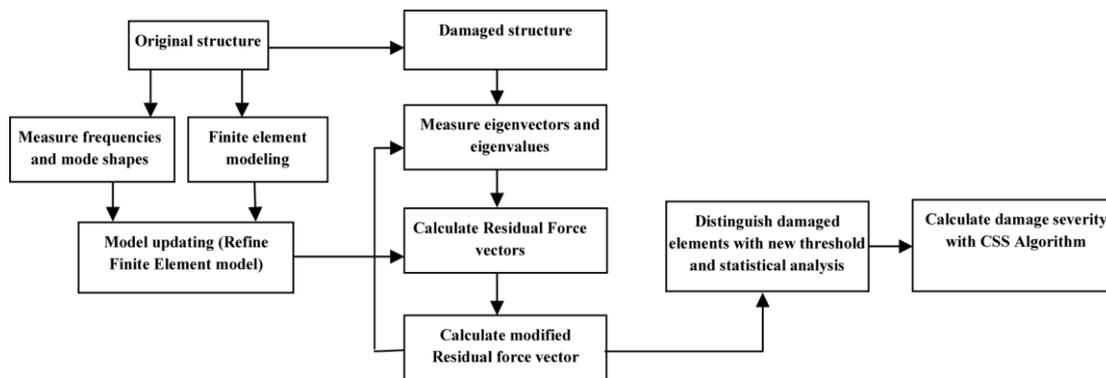
Fig. 3. Flowchart for identification of damage

5. RESIDUAL FORCE METHOD

Residual Force Method [20] is based on identifying modal properties between the undamaged and damaged structures. The *j*th eigenvalue equation of the damaged structure can be derived by substitution of the structural stiffness matrix by that of the damaged one as:

$$[k_d]\{\emptyset_{jd}\} - \lambda_{jd}[M]\{\emptyset_{jd}\} = \{0\} \tag{17}$$

where λ_{jd} and $\emptyset_{jd}$ are the *j*th natural frequency and *j*th mode shape of the damaged structure, respectively. Subscript d is used to denote the damaged structure, and we have

$$k_d = k_u + \Delta k \tag{18}$$

$$M_d = M_u + \Delta M$$

where k_u and M_u are the stiffness and mass matrices of undamaged structure, respectively.

Substituting Eq. (18) in Eq. (17) we obtain

$$\Delta k \emptyset_{jd} - \lambda_{jd} \Delta M \emptyset_{jd} = (-[k_u] + \lambda_{jd}[M_u])\{\emptyset_{jd}\} \tag{19}$$

The left-hand side of the above equation has the unit of force, and if we call it R_i, Eq. (19) can be written as

$$R_i = (-[k_u] + \lambda_{jd}[M_u])\{\emptyset_{jd}\} \tag{20}$$

It can be readily inferred from Eq. (19) that if the measured frequencies and mode shapes are without noise, the *j*th entry of R_i representing the *j*th degree-of-freedom will be zero if none of the elements associated with this DOF is damaged, but it will be a non-zero value if any element associated with this DOF is damaged.

a) *The modified residual force vector R*

If the measured frequencies and mode shapes are not corrupted by noise, this algorithm can determine the exact location of the damaged elements and then an adequate optimization process can precisely obtain the severity of damage; however, normally modal displacement values are measured only at selected points on a structure and they are expanded to all degrees-of-freedom through an eigenvector expansion algorithm. Experimental errors introduced in the measurements and numerical errors in the mode expansion processing both contribute noise to the mode shapes. Therefore, it is possible that some entries associated with undamaged elements DOF be non-zero, so this causes inaccurate result for detecting damage location. To suppress the effect of noise, the Residual Force vector is modified. If the measured values for *k* damaged frequencies and mode shapes are available, we can define the modified Residual Force vector, *R*, as

$$R = \{r_1, r_2, r_3, \dots, r_n\} \tag{21}$$

Where r_j is calculated from the equation

$$r_j = \prod_{i=1}^{k} (|r_j|_i) \tag{22}$$

In which $|r_j|_i$ is the absolute value of the r_j entry of R_i calculated from Eq. (20). This operation magnifies the true data and suppresses false data in calculating *R*. In unpolluted cases, one can pinpoint the location of damaged element by identifying non-zero entries in modal residual force vector, but with presence of noise many entries in R_i will be non-zero, but the value of entries associated with undamaged elements DOFs in modified *R* are reduced, but these entries will still not be zero, so a non-zero threshold is needed to distinguish the DOFs associated with damaged elements.

b) *Recognition criterion*

If all elements associated with the *j*th degree-of-freedom are undamaged, then the sub matrices in Δk and ΔM associated with this DOF will be zero, and so the *j*th entry of R_i will have zero entry. Therefore, by identifying non-zero entries in modal residual force vector, we can find the location of damaged elements, but because of existing of noise in dynamic properties, we will have non-zero entries for undamaged elements DOFs, so a modified residual force vector *R* is introduced in section 5.1 which, in this vector, the effect of noise is compressed significantly and the value of these entries will be small but still not zero. In this study, a criterion was introduced as

$$cr = 0.15 * norm(R) \tag{23}$$

where R is the modified residual force vector, and can be calculated from Eq. (21).

For increasing accuracy in detection damaged elements, when noise level is high, it is expected to repeat measuring modal properties of structures several times. Therefore, we can utilize the above criterion to detect damaged elements in R vector in every process of repetition. Finally, damaged elements can be detected accurately by using statistical analysis.

6. DAMAGE DETECTION METHODOLOGY

a) Algorithm

This algorithm has two stages as follow:

Stage 1: detection damaged elements

For detecting damaged elements it is necessary to follow these steps:

a)   Define the finite element model of the undamaged structure, and obtain the corresponding dynamic parameters.

b)   Measure the dynamic parameters (frequency and mode shapes) of the damaged structure.

c)   Calculate modal Residual Force vectors as Eq. (20)

d)   Calculate modified R vector from modal residual force vectors as Eq. (22)

e)   Distinguish DOFs associated with damaged elements by proposed Cr criterion

Stage 2: determining severity of damage

a)   Define the objective function

b)   Apply a suitable optimization algorithm to determine severity of damage.

c)   Assess the results.

After identifying the DOFs associated with damaged elements and consequently damaged elements in second stage, one can use an optimization algorithm to calculate the severities of damage for detected elements.

In this study enhanced CSS is utilized to minimize the objective function. Each response vector in CSS algorithm is as charged particles (CP). In these vectors, number of variables is equal to number of damaged elements that they have detected in stage 1. The values of these variables indicate the severities of damage in elements. Since the number of variables in response vector is known, search space will be confined, and optimization process will run faster and more accurately.

The larger values of variables obtained from optimization algorithm indicate existence of damage in corresponding elements, but inconsiderable value for each variable indicates that the elements are healthy. This process is depicted in Fig. 3.

7. NUMERICAL STUDIES

a) Example 1: Modified R vector

A spatial truss is considered as shown in Fig. 4b, and residual force vector is calculated in two cases:

Case 1: Without noise

As aforementioned, the DOFs associated with damaged elements in modal residual force vector will get non-zero values and other DOFs will be zero. This is shown in Fig. 5.

Case 2: With noise

This spatial truss is considered with 4% noise according to Case 4 in Table 1. If the dynamic properties of damaged structures were smeared with noise, we have non-zero value in DOFs of healthy elements. This is shown in Fig. 6. The DOFs associated with damaged elements are signed with small

circle. This figure shows that we have perturbation because of presence of noise, and this makes the detection DOFs associated with damaged elements difficult. The *R* vector introduced in Eq. (21) is illustrated in Fig. 7, and shows the effect of noise suppression when modified residual vector *R* is used instead of modal residual force vector.

(a)

(b)
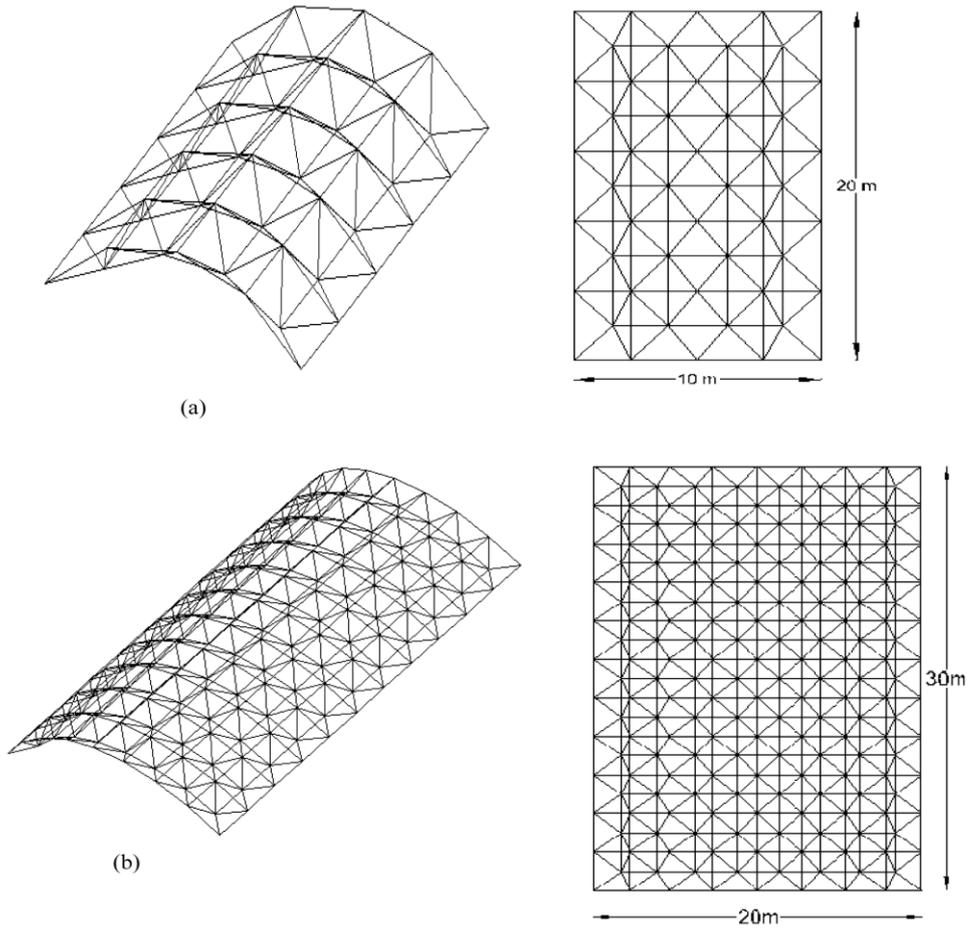
Fig. 4. (a) Barrel vaults: 160 elements with Rise=2m. (b) Barrel vaults: 704 elements with Rise=4m
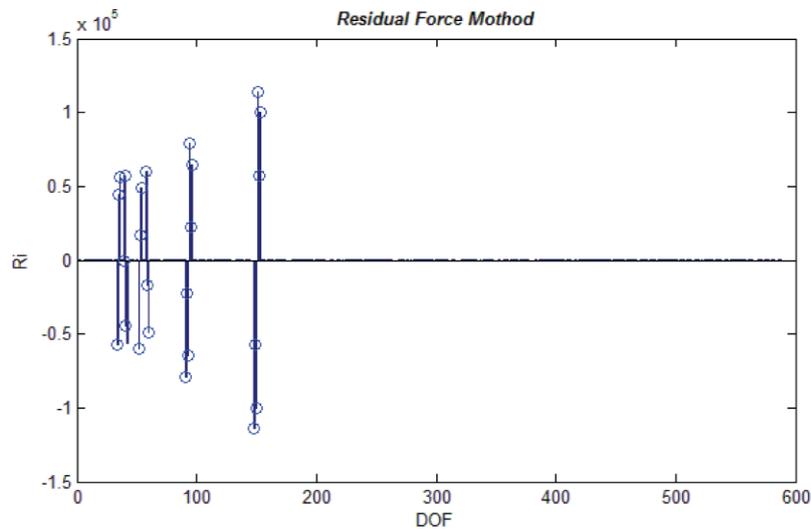
Fig. 5. Modal Residual Force Vector for barrel vaults with 704 elements without noise
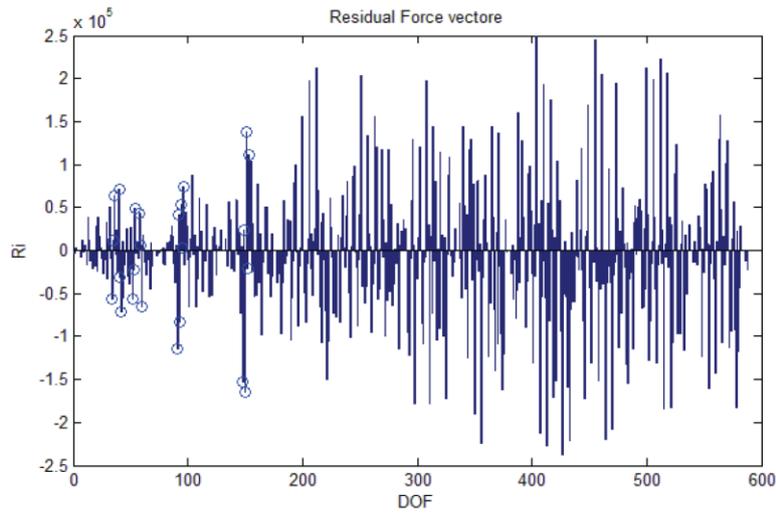
Fig. 6. Modal Residual Force Vector for barrel vaults with 704 elements with 4% noise
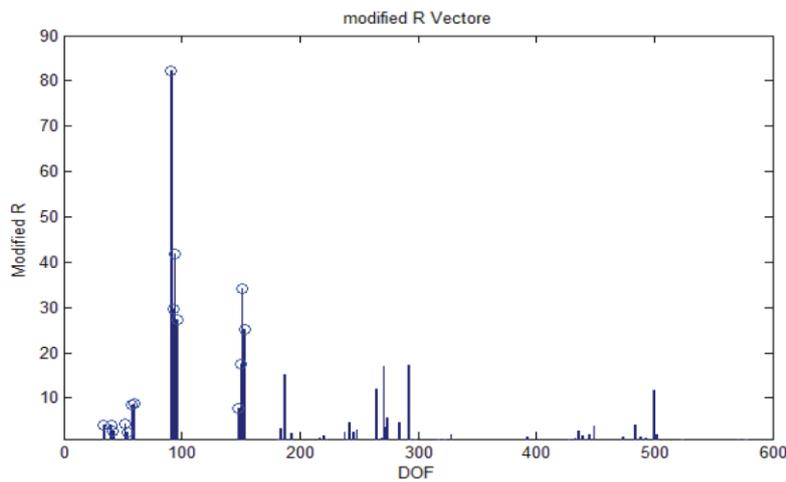
Fig. 7. Modified Residual Force Vector (R) for barrel vaults with 704 elements with 4% noise

Table 1. Different considered scenarios

Damaged scenario	Structural type	No. of FE element	No. of damaged elements	damaged element numbers	severity	No. of dynamic modes	Using case
1	Barrel vaults	164	4	43,82,94,118	40%	5	2% Noise
2	Barrel vaults	164	4	82,84,88,94	40%	5	4% Noise
3	Barrel vaults	704	4	16,411,383,601	40%	5	Without Noise
4	Barrel vaults	704	4	16,383,411,601	40%	5	2% Noise
5	Frame	704	4	16,24,43,79	40%	5	4% Noise

b) Example 2: Assessing damage criterion Cr

The spatial truss according to Fig. 4b and damage scenario as case 4 in Table 1 is considered. As mentioned before, in this study a new threshold was introduced instead of zero to detect DOFs associated with damaged elements in modified R vector. The estimated elements for 10 times execution are illustrated in Fig. 8, which shows extra damaged elements have less than 6 iterations out of 10 iterations.
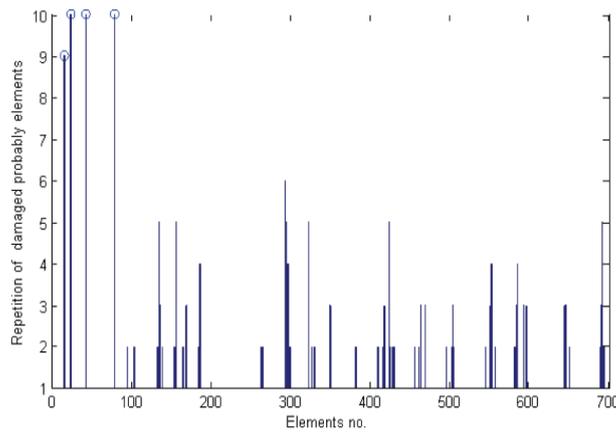
Fig. 8. Estimated elements by new criterion (Cr) from *R* vector in 10 times repetition with 4% noise

c) Example 3: Damage identification with new method

Two barrel vaults with 160 and 704 elements are considered as shown in Fig. 4. For better comparison, the algorithm is considered in two cases: corrupted and uncorrupted cases. All computation is performed in MATLAB using finite element method.

1. Example 1: Barrel vaults with 160 elements: Different scenarios are summarized in Table 1. The cross-section and the elastic modulus are $40\,cm^2$ and $2.01e + 6\ kg/cm^2$, respectively. The results are illustrated in Figs. 9 and 10. Noise is introduced by randomly changing in all eigenvectors and eigenvalues DOFs. All cases demonstrate the ability of the model to detect damage, but it is obvious that when the level of noise increases, iteration of measurement is necessary to get actual damaged elements with less extra elements. In this example, 10 iterations are considered.
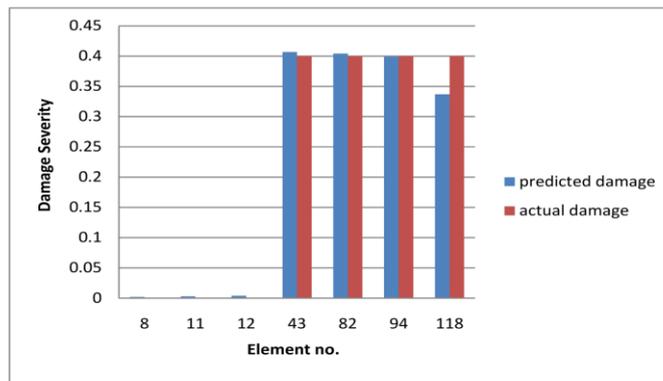
Fig. 9. Damage detection in barrel vaults with 160 elements and 2% noise (Case 1)
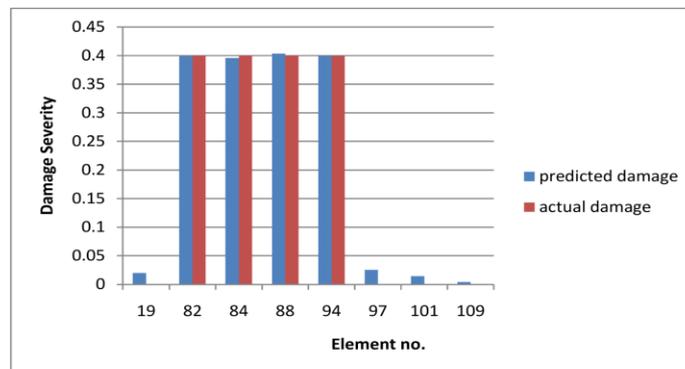
Fig. 10. Damage detection in barrel vaults with 160 elements and 4% noise (Case 2)

Existing extra element in stage 1 in the algorithm causes the need for more iterations in optimization algorithm to obtain a more accurate result.

2. Example 2: Barrel vaults with 740 elements: In this spatial truss, according Fig 4b, the cross-section and the elastic modulus are $40\ cm^2$ and $2.01e + 6\ kg/cm^2$, respectively. The results are illustrated in Figs. 11, 12 and 13.
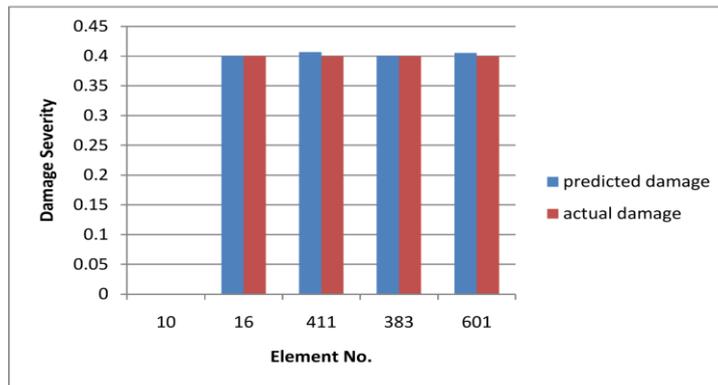
Fig. 11. Damage detection in barrel vaults with 740 elements and without noise (Case 3)
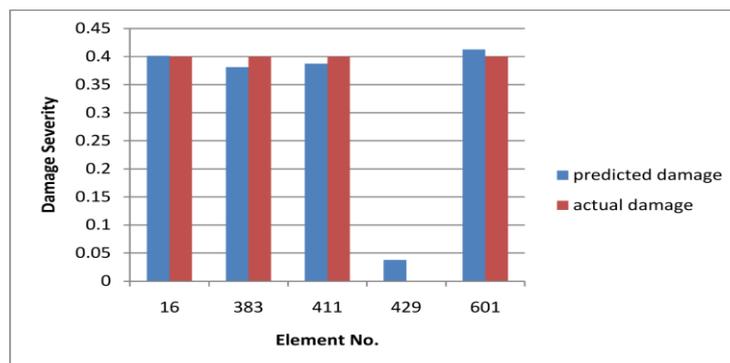
Fig. 12. Damage detection in barrel vaults with 740 elements and 2% noise (Case 4)
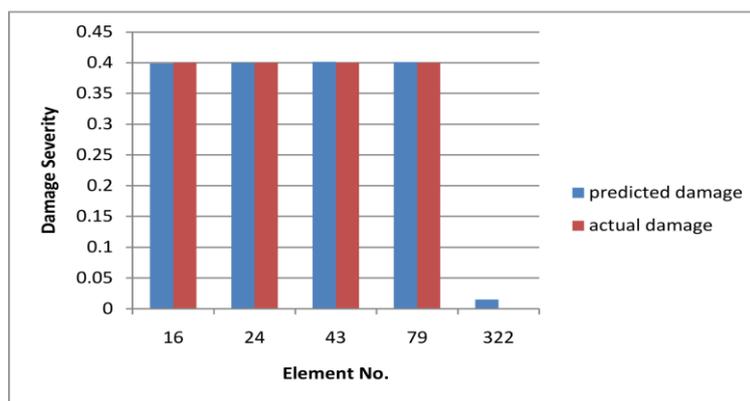
Fig. 13. Damage detection of barrel vaults with 740 elements and 4% noise (Case 5)

The following scenarios are considered for this space structure:

Case 3: Without noise
Results for this damage scenario are depicted in Fig. 11, which shows that without presence of noise, damaged elements are detected precisely and consequently the severity of damaged elements is also found accurately.

Case 4: Dynamic property with 2% noise

Results for this case are illustrated in Fig. 12. Presence of noise causes extra elements in damaged elements list, and therefore more iterations are necessary to find severity of damage accurately in stage 2.

Case 5: Dynamic property with 4% noise

In this case, due to the presence of high noise, measuring dynamic properties should be repeated several times. '*R'* vector is calculated in each iteration, and consequently, based on new threshold *Cr*, damaged elements can be extracted from *R* vector. Finally, the damaged elements can be selected accurately from among elements that have more iteration. In this example, the elements that have more than 60% iteration in total are selected. The results for case 5 are depicted in Fig. 13, and show this fact when damaged elements are calculated more accurately in stage 1, the severity will also be calculated precisely with low iteration in stage 2, although the number of elements and level of noise is high.

8. CONCLUDING REMARKS

In most cases, damage identification in large-scale structures by conventional methods does not lead to desired results. In this study, a new 2-stage method is introduced where in the first stage, damaged elements are detected and then in the second stage, severity of the damage is determined. This method is based on dynamic properties. Since the damaged elements are detected in first stage, the space of search in second stage significantly decreases, so the ability and accuracy of optimization algorithm in detecting damaged elements will be improved.

Some concepts are introduced in this research for efficient detection. These concepts are stated for each stage as follows:

In stage 1 we utilize residual force method for detecting damaged elements, but this method is unable to distinguish DOFs associated with damaged elements because of containing noise, thus a modified residual force vector is introduced as Eq. (21). In this vector, the effect of noise is suppressed significantly, and then a new criterion is introduced to distinguish DOFs associated with damaged elements.

It is suggested that measuring modal properties of structures be repeated several times, because detecting damaged elements accurately in first stage is vital to have successful damage identification. In each repetition, damaged elements are distinguished by introduced criterion as Eq. (23) in modified *R* vector, and then we can pinpoint the ones which have more iteration.

The severity of damage is determined in second stage of the algorithm. In this stage, the CSS algorithm is utilized to detect the severity of damage. The CSS algorithm is enhanced with a recursive process. The variables in CSS algorithm are the severity of the damage for elements that were detected in stage 1.

REFERENCES

1. Zapico, J. (2006). Numerical simulation of a method for seismic damage identification in buildings. *J Eng Struct*, Vol. 28, pp. 255-263.
2. Gudmundson, P. (1982). Eigenfrequency changes of structures due to cracks, notches or other geometrical changes. *J Mech Physics Solids*, Vol. 30, No. 5, pp. 339-353
3. Lee, Y. S. & Chung, M. J. (2000). A study on crack detection using eignfrequency test data. *Comput Struct.*, Vol. 77, pp. 327-42.
4. Ge, M. & Lui, E. M. (2005). Structural damage identification using system dynamic properties. *Comput Struct.*, Vol. 83, pp. 2185-2196

5. Ritter, A. (1993). Vibration inspection of civil engineering structure. Ph. D. thesis. Denmark: Department of Building Technology and Structural Engineering, University of Aalborg.

6. Maity, D. & Tripathy, R. R. (2005). Damage assessment of structure from changes in natural frequencies using genetic algorithm. *Struct Eng Mech*., Vol. 19, pp. 21-42

7. Gerist, S., Naseralavi, S. S. & Salajegheh, E. (2012). Basis pursuit based genetic algorithm for damage identification. *Int J Optim Civil Eng*., Vol. 2, No. 2, pp. 301-319.

8. Aktasoglu, S. & Sahin, M. (2012). Damage detection in beam structure using a combined genetic algorithm and nonlinear optimization system. *Proceedings of the Eleventh International Conference on Computational Structures Technology*. Civil-Comp Press. Stirlingshire, UK.

9. Naseralavi, S. S., Salajegheh, E., Salajegheh, J. & Fadaee, M. J. (2012). Detection of damage in cyclic structures using an eigenpair sensitivity matrix. *Comput Struct.*, Vol. 11, pp. 43-59.

10. Naseralavi, S. S., Salajegheh, E., Salajegheh, J. & Fadaee, M. J. (2010). An improved Genetic Algorithm using Sensitivity Analysis and Micro Search for Damage Detection. *Asian J Civil Eng*., Vol. 11, pp. 717-740.

11. Torkzadeh, P., Goodarzi, Y. & Salajegheh, E. (2013). A two-stage damage detection method for large-scale structure by kinetic and modal strain energies using heuristic particle swarm optimization. *Int J Optim Civil Eng*., Vol. 3, No. 3, pp. 465-482

12. Naseralavi, S. S., Salajegheh, E., Salajegheh, J. & Fadaee, M. J. (2012). Detection of damage in cyclic structures using an eigenpair sensitivity matrix. *Comput Struct*., Vol. 110, pp. 43-59.

13. Kaveh, A. & Talatahari, S. (2010). A novel heuristic optimization method: charged system search. *Acta Mech*., Vol. 213, pp. 267-289.

14. Kaveh, A. & Shokohi, F. (2014). Charged system search algorithm for the optimum cost design of reinforced concrete cantilever retaining walls. *Iranian J Sci Technol, Trans Civil Eng.*, Vol. 38, No. C+, pp. 235-249.

15. Kaveh, A., Massoudi, M. S. & Ghanooni Bagha, M. (2014). Structural reliability analysis using charged system search algorithm. *Iranian J Sci Technol, Trans Civil Eng.*, Vol. 38, No. C2, pp. 439-448.

16. Chandruputala, T. R. & Belegundu, A. D. (2002). *Introduction to finite elements in engineering*. 2nd ed, Prentice-Hall.

17. Kaveh, A. & Zolghadr, A. (2015). An improved CSS for damage detection of truss structures using changes in natural frequencies. *Adv Eng Softw.,* Vol. 80, pp. 93-100.

18. Kaveh, A. & Maniat, M. (2015). Damage detection based on MCSS and PSO using modal data, *Smart Struct Syst, Techno*, Vol. 15, No. 5, pp. 1253-1270.

19. Perera, R & Sevillano, E. (2012). Statistical multi-objective structural damage identification based on dynamic parameters. *Proceedings of the Eleventh International Conference on Computational Structures Technology*. Civil-Comp Press. Stirlingshire, UK.

20. Kosmatka, J. B. & Ricles, J. M. (1999). Damage detection in structures by modal vibration characterization. *J Struct Eng*., Vol. 125, No. 12, pp. 1384-1392.