Chapter

# Construction of an Artificial Neural Network-Based Method to Detect Structural Damage

*Francisco Casanova-del-Angel, Daniel Hernández-Galicia and Xochicale-Rojas Hugo Alberto*

## Abstract

This chapter shows the framework used to obtain data with which the artificial neural network (ANN) was developed. It describes its geometry, properties of the material, sections of structural elements, and loads used. Then, the numerical model of the framework under study is developed in structural analysis using SAP2000® software in order to obtain its modal parameters. In addition, a program made in MATLAB® is shown, from which data with and without damage to the framework under study were obtained, and with which the ANN was developed. Data from the numerical model were used to corroborate data obtained with MATLAB®. The neural model used in this work to detect structural damage is described. Data on damage were obtained simulating a plastic hinge in various elements of a test framework, varying the position of the hinge. The above resulted in obtaining various damage conditions for the same framework, which data thus obtained were used to develop the network. Damage conditions were hierarchized based on their fundamental periods in order to know where is more damage, depending on location of the hinge within the framework. Upon completion of the research, we have concluded that the methodology implemented to detect structural damage is rather simple. It was carried out in four steps.

**Keywords:** damage detection, neural network, failure condition, framework, structural analysis

## 1. Introduction

Throughout their useful life, structures accumulate gradual damage as time passes. Such damage is caused by actions and natural phenomena such as earthquakes, winds, and explosions, among others. Therefore, and taking into account the functioning and the safety of structures, it is of paramount importance to detect damage, to follow up and monitor structures in order to know their physical conditions, and to increase safety and structural reliability. If damage is detected in a structure early, its evolution may be observed regarding magnitude and size, meaning it can be treated properly. Nowadays, there are various methods used to detect structural damage, all of which have advantages and disadvantages. There is no consensus that one method is the best. Therefore, for those who are not specialists on this topic, reading some specialized references would be useful [1–3]. In this article, an artificial neural network (ANN) is developed based on MATLAB's toolboxes [4]. Also, a program with source code was developed using MATLAB® [5], with which data were obtained (condensed matrices and modal parameters (frequencies, periods, and vibration modes) to train the network and, finally, to obtain an ANN capable to detect structural damage.

Damage has been defined as loss of stiffness. In order to identify such loss, plastic hinges in structural elements of the structure under study were simulated to obtain the condensed stiffness matrix with and without damage. Then, dynamic parameters were calculated, as well as their dynamic response with and without damage, which were used to develop the ANN. In this work, stiffness matrices with and without damage, that is, location of damage, are known.

## 2. Background and approach to the problem

For those devoted to analyze and design structures, it is very important to follow them up in order to know their conditions, specifically their physical conditions, in order to detect if there is any damage. Such follow-up on what has happened provides certainty regarding good structural safety for the structure and its inhabitants.

Some damage detection methods are visual and experimental, such as acoustic methods and methods based on magnetic, radiographic, and thermal fields. These experimental methods required access to the part of the structure where damage is located. Due to this limitation, these experimental methods detect damage in visible parts of the structure. The need to develop damage detection methods that may be applied to complex structures has led to develop methods that acknowledge changes in modal properties of structures. Every damage detection method using changes in modal properties of the structure is based on modal parameters (fundamental frequencies, vibration modes, and structure's vibration period), which are functions of physical properties of the structure (stiffness, mass, and damping). Therefore, changes in physical properties of structures such as stiffness cause changes in modal properties.

In civil engineering, studies have been developed to detect structural damage. They have applied ANNs with good results. However, these are recent developments in the area, and for this reason, researchers consider them significant for data development.

Rumelhart and his research team rediscovered the backpropagation learning algorithm (backpropagation). From 1986, the picture was encouraging with respect to the research and development of artificial neural networks [6, 7]. Wu et al. used a backpropagation neural network to identify damage in a three-stored building, modeled as a shear building in two dimensions, when being subject to earthquake excitation [8]. Damage was modeled, reducing stiffness of elements from 50 to 75%. Neural network was used to identify acceleration records and damage level of every element based on Fourier transform. The first 200 points in the fast Fourier transform (FFT) (0–20 Hz) were used as input in the network. A 200-node input layer and a 10-node hidden layer were selected as architecture of the network, and 42 training cases were used to train the network. In the first test, only acceleration data from the last floor were used, and the neural network was capable of identifying only the damage from the third floor with some exactness. The second network used acceleration data from two levels as input. This network was capable of diagnosing damage on the first and third levels with up to 25% approximation, but it was still incapable of predicting, with much precision, damage on the second level. The method is based on full knowledge of the acceleration record of two from three degrees of freedom.

Chen and Kim used a backpropagation neural network as a data processing technique for a metallic structure of a tridimensional bridge instrumented with accelerometers and strain gauges [9]. They studied feasibility to identify and locate structural damage using MATLAB's Neural Network Toolbox®. Vibration signals were obtained from a series of experiments on the scale structure. In their study, they developed four networks: the first one to identify damage using acceleration records, the second one to locate damage using data from strain gauges, the third one using signals simulated through analysis of finite elements and signals from the accelerometer, and the fourth one using acceleration records obtained from analysis of the finite elements model.

Marwala shows a technique based on using a set of neural networks with modal properties (frequencies and vibration modes) and the wavelet transform simultaneously in order to identify damage in structures [10]. The proposed method was tested with simulated data from a three-degrees-of-freedom spring-mass system. The result showed that performance of the method is not influenced by noise in data. Finally, the method was used to identify damage in 10 cylindrical shells. The set was capable to identify damage better than the three methods used separately.

In this work, artificial neural networks are applied to detect structural damage in steel buildings. An artificial neural network is designed to detect damage caused by actions present in the useful life of structures. In order to determine performance of the neural network, a structure is analyzed in various single and multiple damage scenarios. Damage is considered as loss of stiffness [11, 12]. The ANN is compared to data obtained in the numerical model and the program developed in MATLAB®. Learning of the developed neural network is based on epochs. The main idea here is that the network evolves for a period of time (called epoch), with an example sequence, under which it must render proper response. Upon completion of an epoch, the network is restarted in order that the new starting condition of such does not depend on the final condition of the previous epoch. Variation of weights is made upon completion of each epoch, and, as for the multilayer case, it is carried out by lots, that is, all the increases of patterns-epoch are calculated in an incremental manner, which is the one most used for recurrent networks.

As for hypotheses and objectives, it may be considered that side displacements were taken into account in order to make static condensation to horizontal displacements, that is, a stiff floor framework was considered. Only beams are axially stiff, and columns are completely flexible. Another hypothesis is that nodes would be absolutely stiff. In order to obtain data with and without damage (global stiffness matrices, condensed stiffness matrices, modal parameters, and dynamic response), plastic hinges (damage) were simulated on structural elements, based on the hypothesis that these hinges were near to the node, after the structure was subject to stress causing one or more elements to surpass their yield stress.

The objective is, throughout, to detect structural damage in steel buildings applying an ANN to monitor the structure in order to know and obtain proper knowledge of the physical conditions of such to be able to make reinforcement or rehabilitation decisions.

## 3. Method

A three-level framework is modeled in the structural analysis program SAP2000® [13]. From this modal parameter (frequencies, periods, and mode shapes) of the framework was obtained with and without damage. A program was developed in MATLAB® [5], from which the global and condensed stiffness matrices were obtained at horizontal degrees of freedom, with and without damage. Modal parameters and dynamic response with and without damage were calculated, which served for the development of the ANN, and a failure condition was considered to define a serious damage condition.

Data with damage were obtained by simulating a plastic hinge in various elements of the framework. The position of the hinges varied to obtain various damage conditions for the same framework. Data obtained were used to develop the network, and an ANN was obtained to detect structural damage. Regarding how damage conditions are determined, it must be considered that, in the moderate seism-resistant design of reinforced concrete frameworks, there is the fundamental philosophy to dimension and detail the elements of concrete frame in order that a plastic strong column-weak beam mechanism may be developed [14]. In addition, due to the complexity of the problem and for a quick and easy obtaining of data to train the ANN and for simplicity purposes, it was proposed that hinges or damages on the framework appear on the ends of structural elements (beams or columns). As for the question: Are all the damage conditions considered as possible? The answer is yes, since forming hinges in concrete frameworks usually takes place on the ends of their structural elements (beams or columns) [15]. It must be remembered that the ANN shown is in the development phase and still may not be compared to a real building. The ANN was trained for a three-story frame.

Damage detection was carried out in four steps: first, extraction of modal parameters and condensed matrix; second, establishing failure condition for a serious damage condition; third, treatment of modal data to be used in the development of the ANN; and fourth, detection of damage to be carried out with the ANN.

### 3.1 Static condensation

In order to carry out static condensation, the side stiffness matrix $K_L$ was considered. It is associated to side coordinates of the floor, since seismic analysis of flat frameworks includes a single degree of freedom per floor. This is a stiff floor model, which only works for the analysis in view of the horizontal component of soil movement. Since the nodes are considered to be stiff, only the beams are axially stiff and the columns are flexible; condensation is carried out as follows: the original structure has $n$ degrees of freedom (**Figure 1**). The condensed stiffness matrix is obtained first, at the desired degrees of freedom, GL. In our case, horizontal GLs are required, which are associated to displacement vector $\hat{u}$; in this case, the number of levels is identified as $p$, and $K_L$ is the condensed matrix of $p^*p$, thus reducing the total stiffness matrix to condensed matrix $K_L = k_{11} - k_{12}k_{22}^{-1}k_{21}$.

### 3.2 Obtaining modal parameters

To obtain modal parameters, the diagonal mass matrix is $M = [m_1\ m_2\ m_3]$. Once the condensed stiffness matrix $K_L$ and the mass matrix $M$ have been obtained, the problem of eigenvalues and eigenvectors is solved: $(K_L - \omega^2 M)\Phi = 0$, with nonsingular $M$ and symmetric and positive defined $K_L$. Then, there are $N$ real roots $\omega_i$, being $\omega_1 \leq \ldots \leq \omega_N$, for which reason to every eigenvalue corresponds an eigenvector $\phi_i$ which is called own vibration modes $\Phi = (\varnothing_1, \ldots, \varnothing_n)$ where $\Phi$ is the modal matrix. To find the period, ratio $T = \frac{2\pi}{\omega} \forall \omega = \sqrt{\frac{K}{M}}$ has been used. Therefore.

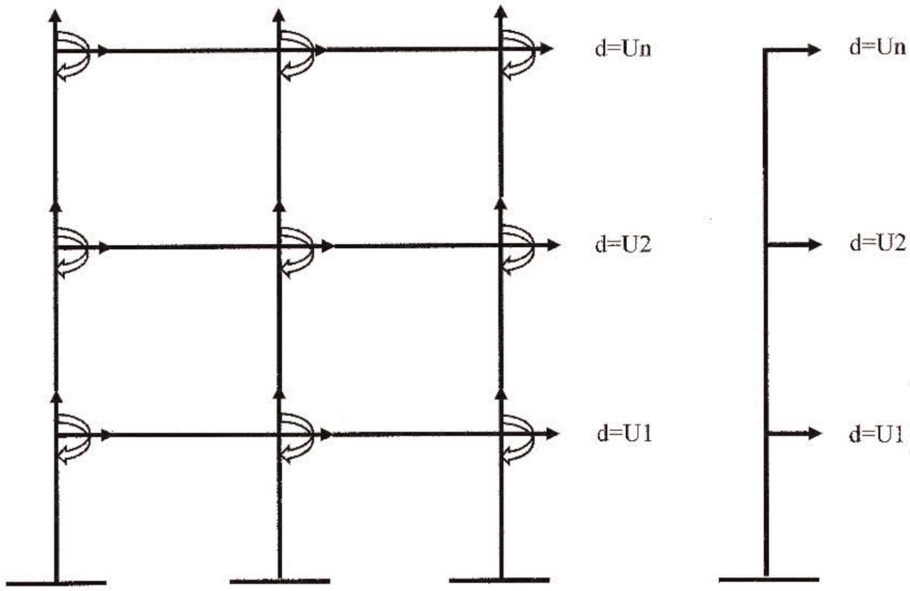



**Figure 1.**
*n degrees of freedom system and condensed only at horizontal GL.*

$$T = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad \omega = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} \omega_{11} & \omega_{21} & \omega_{N1} \\ \omega_{12} & \omega_{22} & \omega_{N2} \\ \omega_{N3} & \omega_{N3} & \omega_{NN} \end{bmatrix} \tag{1}$$

where $T$ is the period and $\omega$ is frequency.

To obtain the dynamic response where there is movement on the base, the second-order linear differential equation $M\ddot{u} + C\dot{u} + K_L u = -mj\ddot{u}_g(t)$ is considered, where $u$ is the vector of $N$ horizontal displacements related to the movement of the soil, $\ddot{u}$ is acceleration, and $\dot{u}$ is speed. In addition, $M$, $C$, and $K_L$ are the mass matrix, the damping matrix, and the side or condensed stiffness matrix, respectively. Every value of $j$ influence vector is equal to one. The right side of the differential equation may be interpreted as the seismic forces $P(t) = -mj\ddot{u}_g(t)$ which is the function of time, $t$.

The distribution of these forces in height is defined by vector $s = mj$ and variation by time $\ddot{u}_g(t)$, which is acceleration of the ground. Such distribution of forces may be expressed as the summation of the distribution of modal inertial forces $s_n$; thus, $mj = \sum_{n=1}^{N} s_n = \sum_{n}^{N} \Gamma_n m \phi_n$, where $\phi_n$ is the ith vibration mode of the structure, $\Gamma_n$ is the factor of modal participation, and $m$ is the mass. Seismic forces may be expressed as $P(t) = \sum_{n=1}^{N} P(t)_n = \sum_{n=1}^{N} -s_n \ddot{u}_g(t)$.

The contribution of the ith mode to $s$ is $s_n = \Gamma_n m \phi_n$ and to P(t) is $P(t) = -s_n \ddot{u}_g(t)$. Now, it may be seen that the response of the multiple degrees of freedom system $P(t)$ is completely in the ith mode, without any contribution from other modes. The equation representing the response of the system is $M\ddot{u} + C\dot{u} + K_L u = -s_n \ddot{u}_g(t)$. Using orthogonal properties of modes, it may be known that none of them, except the ith mode, contributes to the response. Therefore, the displacement of the floor is $u_n(t) = \phi_n q_n(t)$, where the modal coordinate $q_n(t)$ is governed by $\ddot{q}_n(t) + 2\zeta_n \omega_n q_n(t) + \omega^2{}_n q_n(t) = -\Gamma_n \ddot{u}_g(t)$. Consequently, $q_n(t)$ is the response of a single degree of freedom system. $\zeta_n$ is the damping coefficient, and $\omega_n$ is the vibration frequency. Therefore, floor displacements for multiple degrees of freedom system are given by $u_n(t) = \phi_n q_n(t)$.

## 4. Failure condition

The network will identify if there is damage or not. If there is any, it will consider it a failure condition, in order to determine if the structure has failed or not. Therefore, failure condition is defined as a limit service condition, that is, the presence of displacements affecting proper functioning of the building (**Figure 2**) not damaging its capacity to bear loads. In accordance with Article 209 of the Rules for Buildings in Mexico City [16, 17], admissible displacements are as shown in **Table 1**.

### 4.1 Presentation of a 2D framework

The flat framework of **Figure 3** was analyzed considering that beams are infinitely stiff regarding the columns and that the columns are fixed on the base. Stiffness of columns and beams and the spaces between galleries are in SI units. All the elements are made of steel with a I W14X311 section (see **Table 2**).

For the dynamic analysis of the framework, a program was developed in MATLAB®. With it condensed stiffness matrices and modal data (periods, frequencies, and modal shapes) were obtained. To obtain structural response, the framework was placed under stress by the seismic acceleration record in accordance with the department of communication and transportation for the seism that occurred in 1985 and a 5% damping. Data for analysis are as follows (see **Table 3**).

The geometry of the structure is described in **Table 4**. Column five of this table shows *Condition of the bar*, which may have four cases: (i) elements without hinges, (ii) elements with hinges *N* end, (iii) elements with hinges *F* end, and (iv) elements with both end hinges. For the example shown in **Table 4**, all the elements are in case i-elements without hinges. In addition, it must be mentioned that punctual

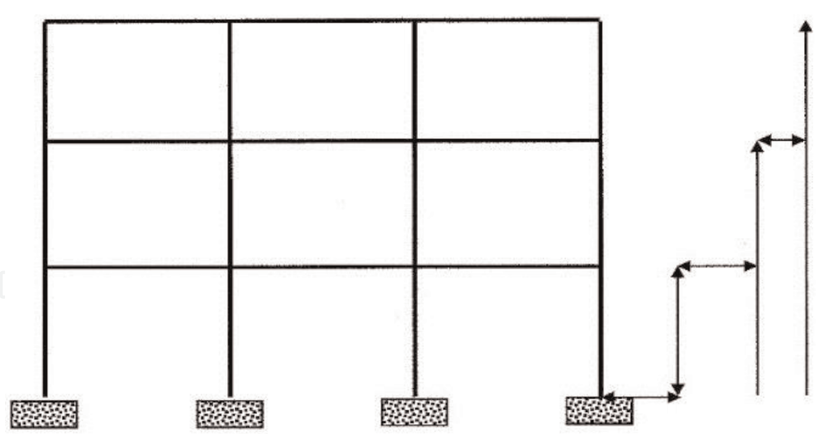



**Figure 2.**
*Relative displacement of mezzanine.*

| | |
|---|---|
| For walls integrated into the framework | Δadm = 0.006 *h* |
| When walls are not integrated into the framework | Δadm = 0.012 *h* |

**Table 1.**
*Admissible values for horizontal displacements in accordance with RCDF [16], where h is the height of space between galleries under study.*

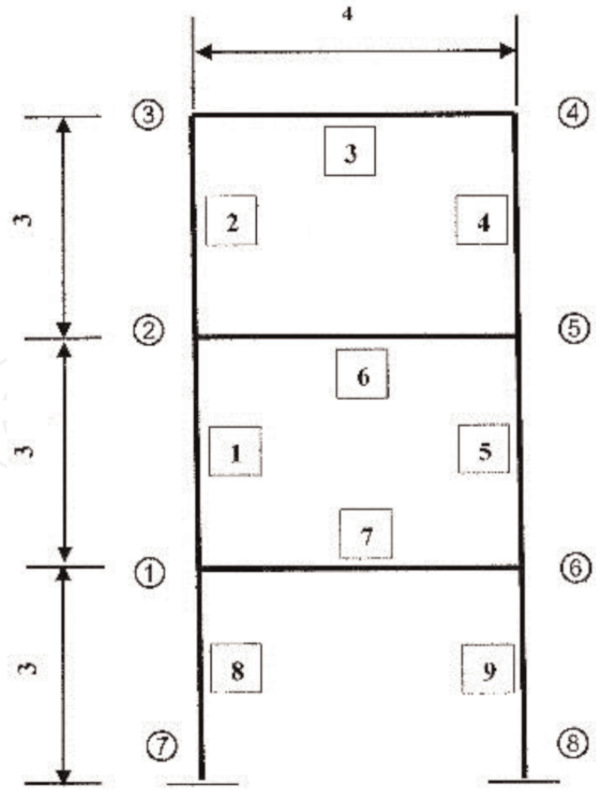



**Figure 3.**
*Framework under analysis.*

| Beam I W 14X311 | |
|---|---|
| Dimensions(m) | |
| t3 | 0.4348 |
| t2 | 0.4122 |
| Tf | 0.0574 |
| Tw | 0.0358 |
| t2b | 0.4122 |
| Tfb | 0.0574 |

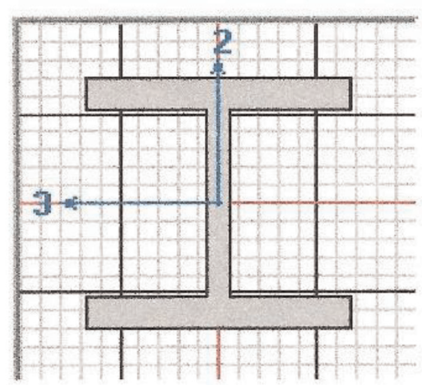



**Table 2.**
*Dimensions of the section under study.*

loads were placed on the nodes to simulate weights that would fall on the framework in service conditions. Magnitudes of loads were placed so that when carrying out the seismic analysis, the period of the framework was within the constant part of the seismic design spectrum of complementary technical standards [16, 17] and RCDF and has a behavior similar to that of most structures.

Loads applied on the framework are shown in **Figure 4**. Magnitude of loads was fixed at 25 ton in order that, when carrying out the seismic analysis, the vibration periods would fall on the constant part of the design spectrum of the NTC, Construction Rules for Mexico City.

| | |
|---|---|
| Volumetric weight of steel | 7.8 Ton/m$^3$ |
| Number of degrees of freedom | 24 |
| Number of bars | 9 |
| Number of unknown displacements | 18 |
| Number of known forces | 18 |

**Table 3.**
*Degrees of freedom, number of bars and known/unknown forces on the framework.*

| Bar | L | I | E | Condition of the bar | | A |
|---|---|---|---|---|---|---|
| | m | m$^4$ | Kg/m$^2$ | | | m$^2$ |
| 1 | 3 | 0.001802 | 20904200000 | 1 | Element without hinge | 0.059 |
| 2 | 3 | 0.001802 | 20904200000 | 1 | Element without hinge | 0.059 |
| 3 | 4 | 0.001802 | 20904200000 | 1 | Element without hinge | 0.059 |
| 4 | 3 | 0.001802 | 20904200000 | 1 | Element without hinge | 0.059 |
| 5 | 3 | 0.001802 | 20904200000 | 1 | Element without hinge | 0.059 |
| 6 | 4 | 0.001802 | 20904200000 | 1 | Element without hinge | 0.059 |
| 7 | 4 | 0.001802 | 20904200000 | 1 | Element without hinge | 0.059 |
| 8 | 3 | 0.001802 | 20904200000 | 1 | Element without hinge | 0.059 |
| 9 | 3 | 0.001802 | 20904200000 | 1 | Element without hinge | 0.059 |

**Table 4.**
*Number of bars of the framework under study, lengths of elements, their geometric properties, and properties of material.*

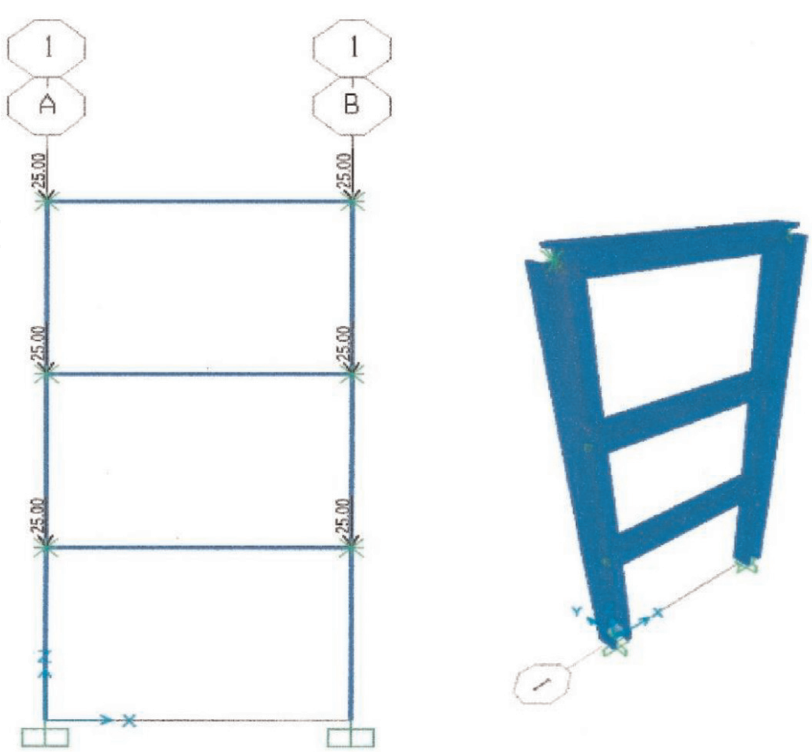



**Figure 4.**
*Numerical model made in SAP2000® structural analysis software, with position of loads and their magnitudes.*

*DOI: http://dx.doi.org/10.5772/intechopen.87193*

### 4.2 2D numerical model

The framework shown in **Figure 3** was modeled with a structural analysis software, with the same geometry and properties of material, in order to obtain modal data (periods, frequencies, and modal shapes), as well as structural response for a I W14X311 section. The numerical model used was carried out to compare data obtained from such with those obtained with the program in MATLAB® in order to calibrate the numerical model.

### 4.3 Program in MATLAB® to simulate 2D framework

A program was developed in MATLAB®, with which the framework was analyzed, obtaining data with and without damage for the development of the ANN. For calculation of data with damage, the total stiffness matrix $K_T$ was obtained from stiffness matrices with hinges on one or both ends. These conditions are defined in the program in order to carry out a structural and a dynamic analysis.

The program must be supplied with the geometry and properties of material, number of floors, damping, degrees of freedom to be condensed, NGLC, displaced degrees of freedom, NGLD, mass per floor, and accelerogram. The stiffness method is used to obtain the total stiffness matrix $K_T$. From this the reduced total stiffness matrix KTR, the reduced reordered total stiffness matrix KTRR, and finally, the

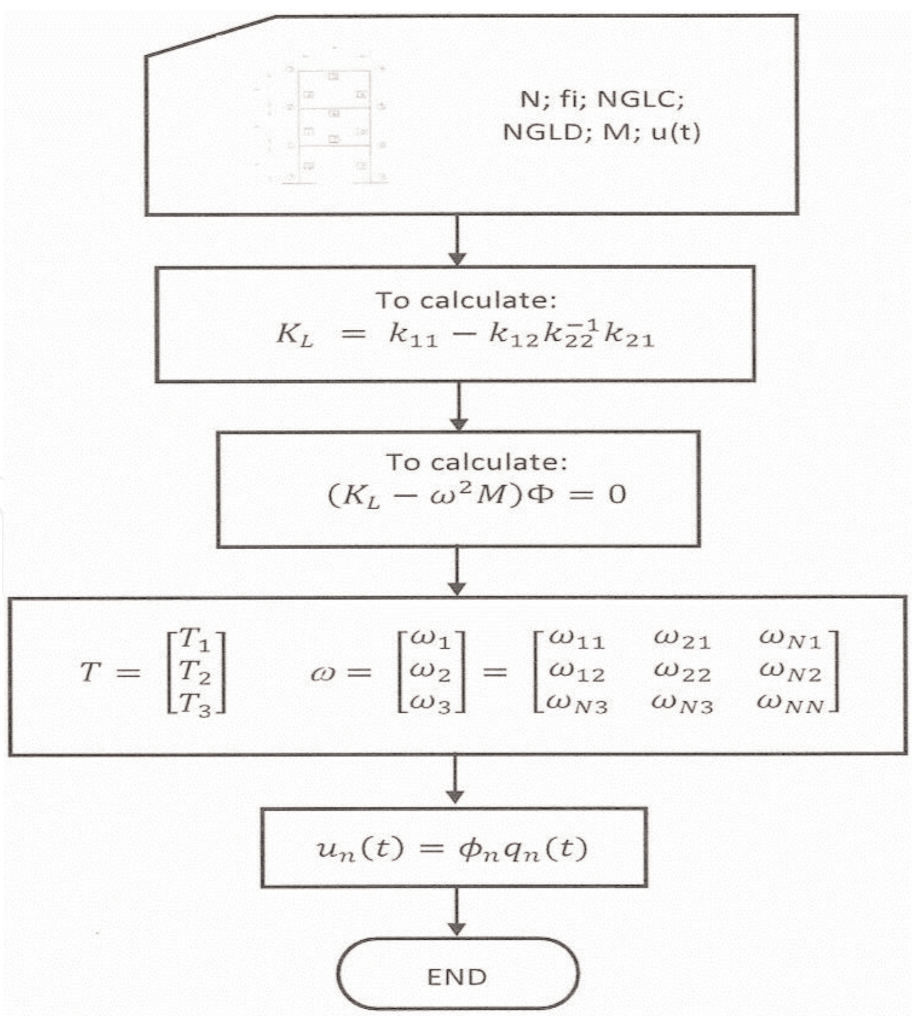



**Figure 5.**
*Flowchart to obtain data with and without damage.*

condensed matrix $K_L$ are obtained. This is obtained to reduce the system in order to make a dynamic analysis to calculate horizontal displacements per floor, with and without damage. After that, the problem of eigenvalues and eigenvectors is solved to obtain frequencies and modal shapes of the framework. Finally, the response per floor is obtained applying an excitation on the base of the structure. The flowchart (**Figure 5**) shows the sequence of steps:

## 5. Application of the neural network to detect structural damage

Let us see three cases of application of the ANN which were created. A three-level framework was analyzed, in which plastic hinges on the ends of its elements were simulated. Plastic hinges varied in position to obtain various damage conditions. In addition, to simplify the problem, the $n$ degrees of freedom system, GL, was reduced to a three-degrees-of-freedom system (**Figure 3**), from which 48 different damage conditions were obtained to train the network. In Case 1, 16 different conditions were used, from which only one is without damage while the remaining 15 have damage. For Case 2, 48 different conditions were used. For Case 3, the ANN was trained with modal matrices from 48 cases. The failure condition described was taken into account (to determine if the structure fails or not). For every state the condensed stiffness matrix $K_L$, its periods, frequencies, and modal matrices were obtained. These data were used to train the ANN.

### 5.1 Case 1: test of the ANN

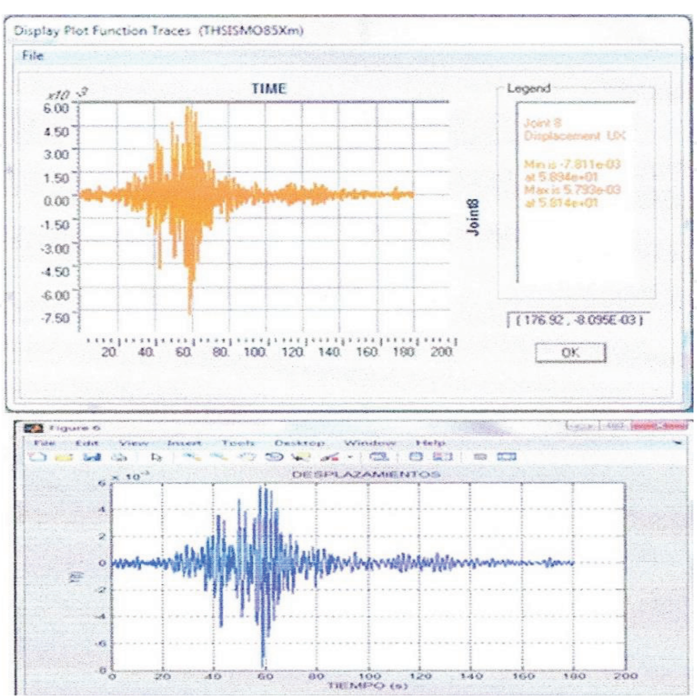
For this example, the periods of every damage case were used to carry out the test of the ANN. From the above framework, a condition without damage was obtained, as well as 15 damage conditions. **Table 5** shows the elements damaged for every condition, as well as the location of damage. It should be mentioned that data were obtained with the program developed in MATLAB®. Structural analysis software SAP2000® was used only to verify that data obtained in the program created in MATLAB® were correct. **Figure 6** shows the structural response of Case 1 obtained with SAP2000® and the program created in MATLAB®, which were used to verify that results obtained with the program developed using MATLAB® were correct and, therefore, to train the ANN.

In addition, the network used is of the backpropagation type, which uses the error backpropagation algorithm or generalized delta rule (**Figure 7**). The network is provided with the following data to be trained: error = 0.001 and maximum number of epochs = 10,000. From the analysis of the framework, condensed stiffness matrices $K_L$ were obtained for every condition. Since the framework has three levels, the condensed matrix $K_L$ is $3 \times 3$. The principal diagonals were extracted to every $K_L$ matrix to create a $3 \times 16$ matrix with the eigenvalues. This was made because the principal diagonal contains the stiffness coefficients to directions of displacements and spins, that is, coefficient $K_{11}$ is the force generated in direction 1 when applying a unit displacement in the same direction. Also, a $3 \times 16$ matrix was created, with the periods obtained for every case. These two matrices were created in order to correlate them. Correlation between these two matrices created the weights matrix, to train the network. The matrix resulting from the correlation is $3 \times 3$. Once the weight matrix to start the network was obtained, the period's matrix was the input matrix for the network and, in turn, the objective matrix. This was made in order to verify that the network worked properly, that is, the output of the network should be the input period's matrix. With these last data, the network was trained. Obtaining data close to the periods wanted, **Table 6** shows the error obtained by the network, the number of epochs required to train the ANN, and the output weights.

| Case | Damaged element | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | SD | SD | SD | SD | SD | SD | SD | SD | SD |
| 1 | SD | SD | D-EF | SD | SD | SD | SD | SD | SD |
| 2 | SD | SD | D-EN | SD | SD | SD | SD | SD | SD |
| 3 | SD | D-EF | SD | SD | SD | SD | SD | SD | SD |
| 4 | SD | SD | SD | D-EN | SD | SD | SD | SD | SD |
| 5 | SD | D-EN | SD | SD | SD | SD | SD | SD | SD |
| 6 | SD | SD | SD | D-EF | SD | SD | SD | SD | SD |
| 7 | SD | SD | SD | SD | SD | D-EF | SD | SD | SD |
| 8 | SD | SD | SD | SD | SD | D-EN | SD | SD | SD |
| 9 | D–EF | SD | SD | SD | SD | SD | SD | SD | SD |
| 10 | SD | SD | SD | SD | D-EN | SD | SD | SD | SD |
| 11 | D–EN | SD | SD | SD | SD | SD | SD | SD | SD |
| 12 | SD | SD | SD | SD | D-EF | SD | SD | SD | SD |
| 13 | SD | SD | SD | SD | SD | SD | D-EF | SD | SD |
| 14 | SD | SD | SD | SD | SD | SD | D-EN | SD | SD |
| 15 | SD | SD | SD | SD | SD | SD | SD | D-EN | SD |

*D, damage; SD, without damage. D-EN damage in extreme N (or mode vibration 1), and D-EF damage in extreme F (mode vibration 2).*

**Table 5.**
*Damaged elements and corresponding case.*




Maximum
displacements m.

0.005793
-0.007811

Periods
0.318279
0.104332
0.051539

Maximum
displacements m.

0.005787
-0.007804

Periods
0.318255
0.104709
0.052938

**Figure 6.**
*The structural response obtained with SAP2000® and MATLAB® from Case 1, with maximum displacements and periods is shown.*

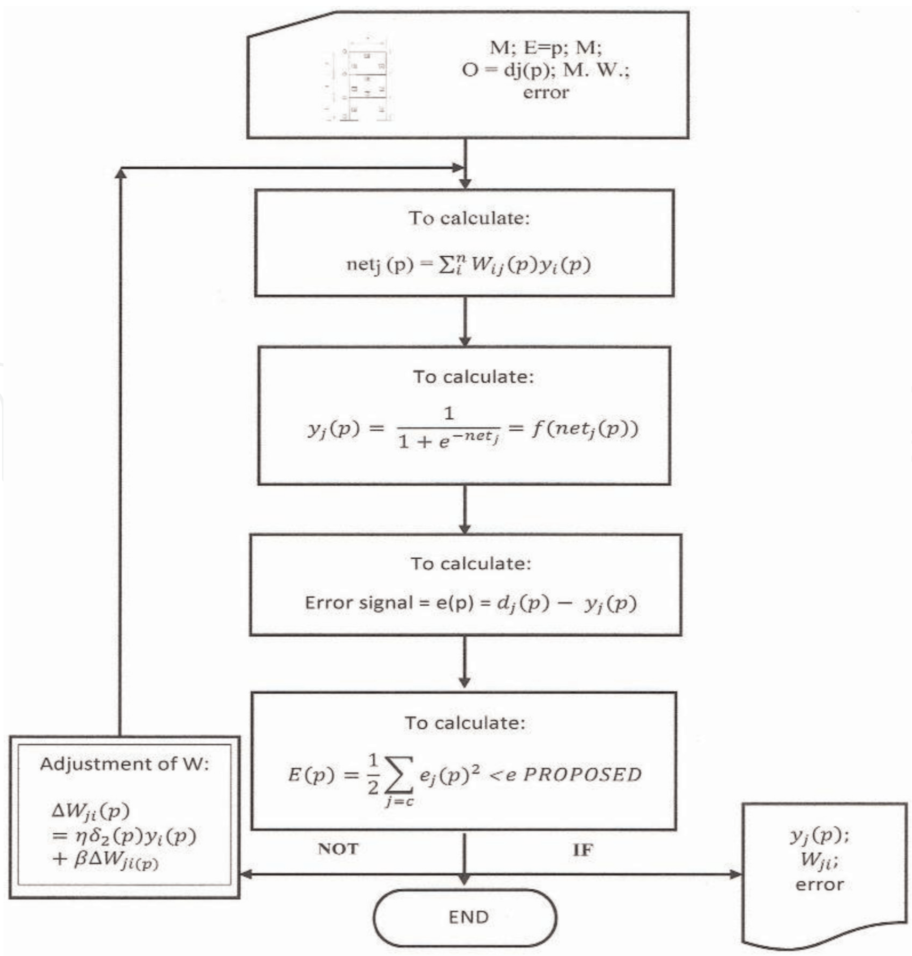



**Figure 7.**
*Flowchart of the error backpropagation algorithm or the generalized delta rule for the case under study.*

| | Output weights | | | Average error |
|---|---|---|---|---|
| **1** | 3.47499926 | 0.08577985 | -0.88603064 | 1.048724 |
| **2** | -0.43881538 | -0.17416166 | -0.88476444 | 5.329752 |
| **3** | 0.41886369 | -0.85005851 | 0.22275651 | 5.459212 |

**Table 6.**
*Output weights and average error between objective and calculated matrices.*

### *5.1.1 Result*

Results obtained where as follows:

It may be seen that the maximum displacements obtained with SAP2000® are almost equal to those obtained with MATLAB®, as well as the periods.

Inputs are the weight and period matrices and the objective. The output is a modified weight matrix and a calculated matrix close to the objective matrix. This diagram describes the error backpropagation algorithm or generalized delta rule explained above.

In tables in Annex I, it can be seen that the ANN works properly with a small error. In order to reduce the error obtained by the network, the ANN may be provided with a smaller error than that proposed, to reduce the difference between the required input and output. Here, the error of the network is 0.00099999 and the number of epochs required is 5976. **Table 6** for weights shows learning of the network.

Since the error of the network is lower than the error proposed, training is completed. In addition, the number of epochs required for training was 5976 of the 10,000 that were proposed at the beginning. As was previously mentioned, if the output calculated by the network is closer to the objective matrix given to it, a smaller error must be proposed. **Table 6** also shows the average error percentage between the objective and calculated matrices, which is acceptable. It must be considered that the error is a percentage measured between periods to be calculated by the ANN and the ones actually calculated.

### 5.2 Case 2: damage detection

In this application, condensed stiffness matrices were used to carry out all the training of the ANN, different from Case 1, where the periods were used for test purposes and to verify that the ANN was working properly. **Table 7** shows 48 damage conditions and which bar is damaged for every case.

### 5.3 Training of the network

To train the network, there are 48 condensed stiffness matrices, from which one belongs to the framework without damage matrix zero, and the remaining 47 are matrices with damage. Matrix number 47 was selected as the objective matrix, leaving it fixed for every training case. Matrix number 47 belongs to a framework with damage (**Figure 8**).

**Figure 3** framework selected to train the ANN. Damage may be observed in bars 7 and 3. Framework number 47 was selected since it is known that an articulated column from the base of the framework is more unfavorable than an intermediate one, though numerically it was found that damage is higher if the hinge is in an intermediate column (**Figure 9**). Case 42 shown may be seen in **Figure 8**.

The input data to train the network were (i) condensed matrices for every framework; (ii) the weight matrix, created from the correlation of two matrices, the stiffness matrix containing the principal diagonal of each condensed matrix and the period matrix; and (iii) bias matrix, started with a unit matrix. The training began with condensed matrix number zero, the matrix without damage, and matrices were given in increasing order until matrix number 47. For the weight's matrix, the starting matrix was the one obtained in the correlation; the matrix obtained upon completion of training for each case became the input weight matrix for the following case and so on, until reaching matrix number 47. The same was carried out with the bias matrix. It should be mentioned that stiffness matrices were multiplied by a factor that reduced values to decimal numbers, because the network used only accepts values lower than the unit since they have a sigmoid activation function. In addition, training was supervised for every case. The maximum number of epochs (training cycles) proposed was 30,000, while the error calculated by the network should be lower than 0.001, and the $\eta$ learning speed of 1 was proposed. It must be stressed that for each training case, and as was mentioned above, the objective matrix was matrix number 47 (damage case). This matrix was fixed and was the objective matrix for the 48 cases. For this purpose, the ANN had to find a weight matrix making the input matrix adjusted to the objective matrix, that is, it had to be equal or almost equal to the objective matrix. That means the ANN was working and learning properly.

#### *5.3.1 Results*

**Table 8** shows the data proposed for training of Case Zero (Damage Zero) and data obtained by the ANN upon completion of training.

| Case | Damaged element | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| 0 | SD | SD | SD | SD | SD | SD | SD | SD | SD |
| 1 | SD | SD | D-EF | SD | SD | SD | SD | SD | SD |
| 2 | SD | SD | D-EN | SD | SD | SD | SD | SD | SD |
| 3 | SD | D-EF | SD | SD | SD | SD | SD | SD | SD |
| 4 | SD | SD | SD | D-EN | SD | SD | SD | SD | SD |
| 5 | SD | D-EN | SD | SD | SD | SD | SD | SD | SD |
| 6 | SD | SD | SD | D-EF | SD | SD | SD | SD | SD |
| 7 | SD | SD | SD | SD | SD | D-EF | SD | SD | SD |
| 8 | SD | SD | SD | SD | SD | D-EN | SD | SD | SD |
| 9 | D–EF | SD | SD | SD | SD | SD | SD | SD | SD |
| 10 | SD | SD | SD | SD | D-EN | SD | SD | SD | SD |
| 11 | D–EN | SD | SD | SD | SD | SD | SD | SD | SD |
| 12 | SD | SD | SD | SD | D-EF | SD | SD | SD | SD |
| 13 | SD | SD | SD | SD | SD | SD | D-EF | SD | SD |
| 14 | SD | SD | SD | SD | SD | SD | D-EN | SD | SD |
| 15 | SD | SD | SD | SD | SD | SD | SD | D-EN | SD |
| 16 | SD | SD | SD | SD | SD | SD | SD | SD | D-EN |
| 17 | SD | SD | D-ENF | SD | SD | SD | SD | SD | SD |
| 18 | SD | D–EN | D-ENF | SD | SD | SD | SD | SD | SD |
| 19 | SD | SD | D-ENF | D–EF | SD | SD | SD | SD | SD |
| 20 | D–EF | SD | D-ENF | SD | SD | SD | SD | SD | SD |
| 21 | SD | SD | D-ENF | SD | D–EN | SD | SD | SD | SD |
| 22 | D–EN | SD | D-ENF | SD | SD | SD | SD | SD | SD |
| 23 | SD | SD | D-ENF | SD | D–EF | SD | SD | SD | SD |
| 24 | SD | SD | D-ENF | SD | SD | SD | SD | D–EN | SD |
| 25 | SD | SD | D-ENF | SD | SD | SD | SD | SD | D–EN |
| 26 | SD | SD | SD | SD | SD | D-ENF | SD | SD | SD |
| 27 | SD | SD | D–EF | SD | SD | D-ENF | SD | SD | SD |
| 28 | SD | SD | D–EN | SD | SD | D-ENF | SD | SD | SD |
| 29 | SD | D–EF | SD | SD | SD | D-ENF | SD | SD | SD |
| 30 | SD | SD | SD | D–EN | SD | D-ENF | SD | SD | SD |
| 31 | D–EN | SD | SD | SD | SD | D-ENF | SD | SD | SD |
| 32 | SD | SD | SD | SD | D–EF | D-ENF | SD | SD | SD |
| 33 | SD | SD | SD | SD | SD | D-ENF | SD | D–EN | SD |
| 34 | SD | SD | SD | SD | SD | D-ENF | SD | SD | D–EN |
| 35 | SD | SD | SD | SD | SD | SD | D-ENF | SD | SD |
| 36 | SD | SD | D–EF | SD | SD | SD | D-ENF | SD | SD |
| 37 | SD | SD | D–EN | SD | SD | SD | D-ENF | SD | SD |
| 38 | SD | D–EF | SD | SD | SD | SD | D-ENF | SD | SD |
| 39 | SD | SD | SD | D–EN | SD | SD | D-ENF | SD | SD |
| 40 | SD | D–EN | SD | SD | SD | SD | D-ENF | SD | SD |
| 41 | SD | SD | SD | D–EF | SD | SD | D-ENF | SD | SD |
| 42 | D–EF | SD | SD | SD | SD | SD | D-ENF | SD | SD |
| 43 | SD | SD | SD | SD | D–EN | SD | D-ENF | SD | SD |
| 44 | D–EN | SD | SD | SD | SD | SD | D-ENF | SD | SD |
| 45 | SD | SD | SD | SD | D–EF | SD | D-ENF | SD | SD |
| 46 | SD | SD | SD | SD | SD | SD | D-ENF | D–EN | SD |
| 47 | SD | SD | SD | SD | SD | SD | D-ENF | SD | D–EN |

*D, damage; SD, without damage; EN, end N; EF, end F; ENF, ends N and F.*

**Table 7.**
*Identification of damaged elements and in which case.*

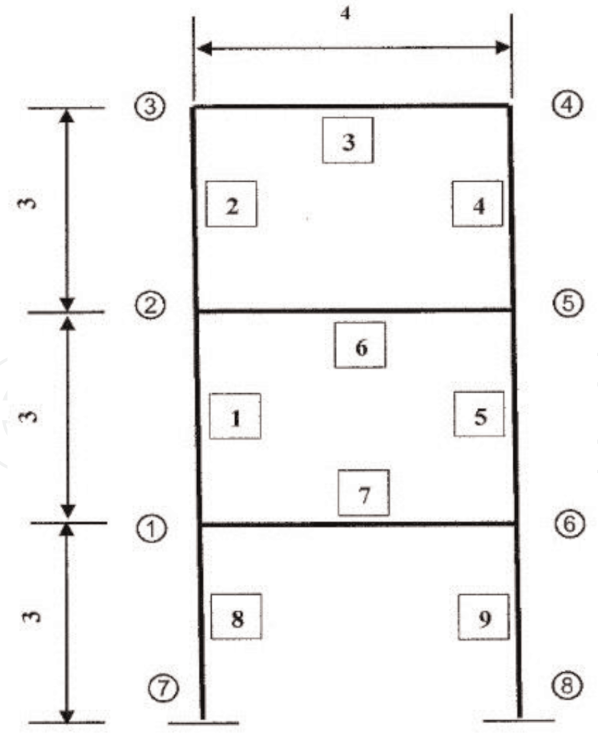



**Figure 8.**
*Framework with the most serious damage (Case 42).*

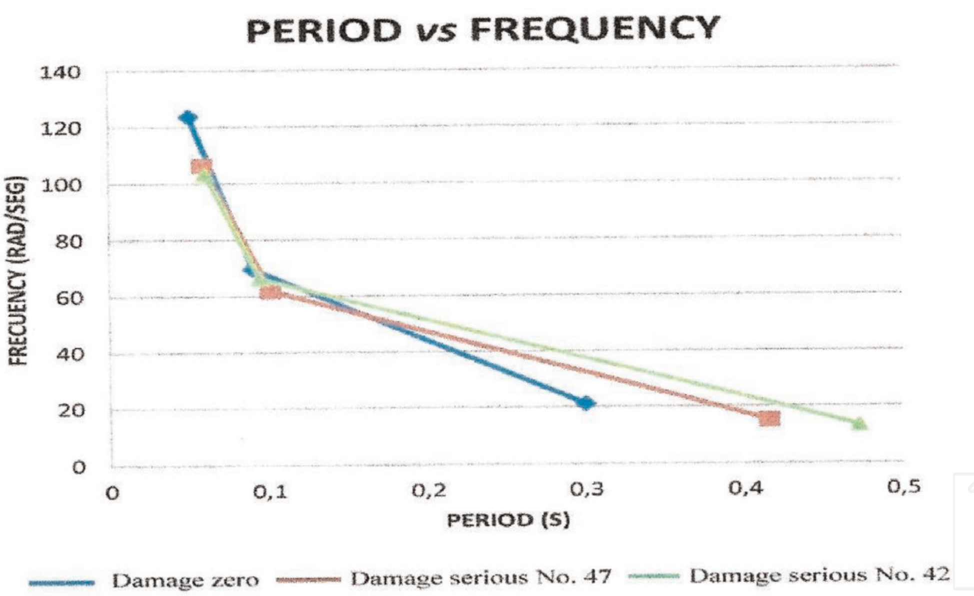



**Figure 9.**
*The plot shows that damage is higher for Case 42 than for Case 47.*

It can be observed that the 30,000 epochs proposed were required to train the ANN. In addition, it may be seen that the error calculated by the ANN is higher than that proposed as objective error. This error was left out, since it is not significant. The error calculated by the ANN could be reduced by increasing the number of epochs, but then training would be longer. Therefore, it was decided to leave these results out and, as it was already mentioned, the input matrix for the ANN, in this case the condensed matrix zero (Damage Zero), the objective matrix that was the condensed matrix of Case 47 (case with damage), and the output matrix calculated by the ANN. The ANN searches for a weight matrix to make the input matrix adjust to the objective matrix, which tells us how efficient its learning is. As may be seen in **Table 9**, the output matrix is almost equal to the objective matrix, with a 16% error.

| Training data | |
|---|---|
| Start | 10:05 a.m. |
| Lr | 1 |
| Error | 0.001 |
| Max. No. of epochs | 30000 |

| Data obtained | |
|---|---|
| Final | 11:48 a.m. |
| Lr | 1 |
| Error red | 0.16 |
| No of epochs required | 30000 |

**Table 8.**
*Data proposed for training, calculated by the ANN for the Damage Zero case.*

This procedure was made for the 48 cases considered, obtaining 48 matrices almost equal to the objective matrix (Case 47).

The dynamic properties of the framework were used to graph the period and the frequency, as well as to graphically observe the damage to the framework. **Figure 10** shows the case with Damage Zero, case with Damage 47, case with intermediate damage (Case 22), and the case with serious damage (Case 4) of 48 cases considered. These graphs show something already known, that is, that frequency is a function of stiffness and that period is a function of frequency. If stiffness changes, dynamic properties of the framework will also change. If stiffness decreases, frequency also decreases; therefore, the period increases. This shows that the structure lost stiffness (damage) and, therefore, became more flexible, thus increasing the vibration time (period). Therefore, the vibration period of Damage Zero framework is lower than the vibration period of the framework with serious damage (Case 42) (**Figure 10**). It should also be noted that mass was constant in every case. Graphs have been used to define an epsilon value (**Table 10**) both for the period and the frequency, respectively. With this parameter, through the ANN, the case of damage may be specified, as well as its range.

The damage conditions could also be placed hierarchically, based on their fundamental periods, in order to know where there is more damage, depending on the location of the hinge within the framework. It was observed that, when hinges were placed on the higher corners of the framework, damage was the same, regardless of the hinge being in the column or the girder. The same happened when the hinges were placed on the corners of the lower part of the framework. It should be highlighted that the damage is the same whether hinges are in the higher or lower corners of the framework. In other cases, the same did not happen. An important observed aspect is that cases where there was higher damage are those where hinges were inside the framework, that is, in intermediate elements. Case 47 was expected to be a serious

| | Input matrix | | | Objective matrix | | | Output matrix | | |
|---|---|---|---|---|---|---|---|---|---|
| | D0 | | | D47 | | | D47′ | | |
| | Condensed matrix Kg/m | | | Condensed matrix Kg/m | | | Condensed matrix Kg/m | | |
| 1 | 0.550475 | -0.303859 | 0.059508 | 0.358309 | -0.21264 | 0.050552 | 0.358558 | 0.000295 | 0.047987 |
| 2 | -0.303859 | 0.439864 | -0.205054 | -0.21264 | 0.343081 | -0.192423 | 0.000015 | 0.342884 | 0.000253 |
| 3 | 0.059508 | -0.205054 | 0.151989 | 0.050552 | -0.192423 | 0.150256 | 0.051943 | 0.000408 | 0.149131 |

**Table 9.**
*The first table shows the input matrix, the second one shows the objective matrix, and the third one shows the output matrix calculated by the ANN.*

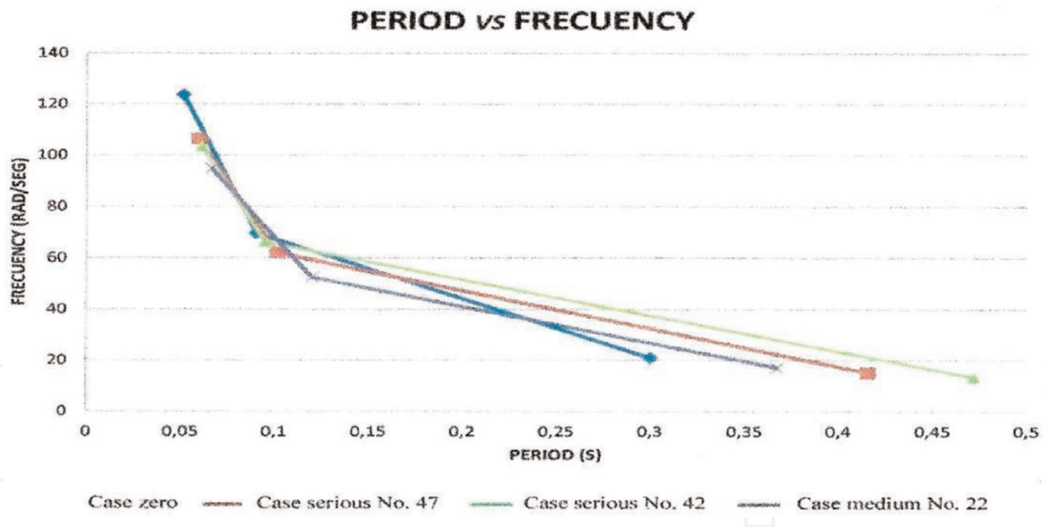



**Figure 10.**
*This plot shows cases with Damage Zero, case with Damage 47, case with serious Damage 42, and case with intermediate Damage 22.*

| Period's epsilon | | | |
|---|---|---|---|
| Case 43 | Highest period 0.472126659 | Case 0 | Lowest period 0.299830019 |
| Difference | 0.17229664 | Seconds | |
| **Frequency's epsilon** | | | |
| Case 0 | Highest frequency 20.95582465 | Case 42 | Lowest frequency 13.30826207 |
| Difference | 7.647562583 | Rad/s | |

**Table 10.**
*Epsilon value for period and frequency.*

damage case, where elements 7 and 9 of the frameworks are articulated. However, the serious damage case is Case 42, where hinges are in elements 1 and 7 of the frameworks.

### 5.4 Case 3: damage detection

In this example, the ANN was trained with modal matrices of every damage case. As it has already been mentioned, there are 48 different cases, from which one is the case without damage. For Case Zero, its condensed stiffness matrix $K_L$, its mass matrix $M$, its frequencies $\omega$, and its modal matrix $\Phi$ are known (**Figure 3**).

There are also 47 cases of damage, through which the same information can be known.

$K_L{}^*, M^*, \omega^*, \Phi^*$ are, respectively, the condensed stiffness matrix with damage, the masses matrix with damage, and the modal matrix with damage (**Figure 11**). Since it is possible to know the dynamic properties of the framework $\omega^*; \Phi^*$ through an environmental vibration test, it was selected to train the ANN with the

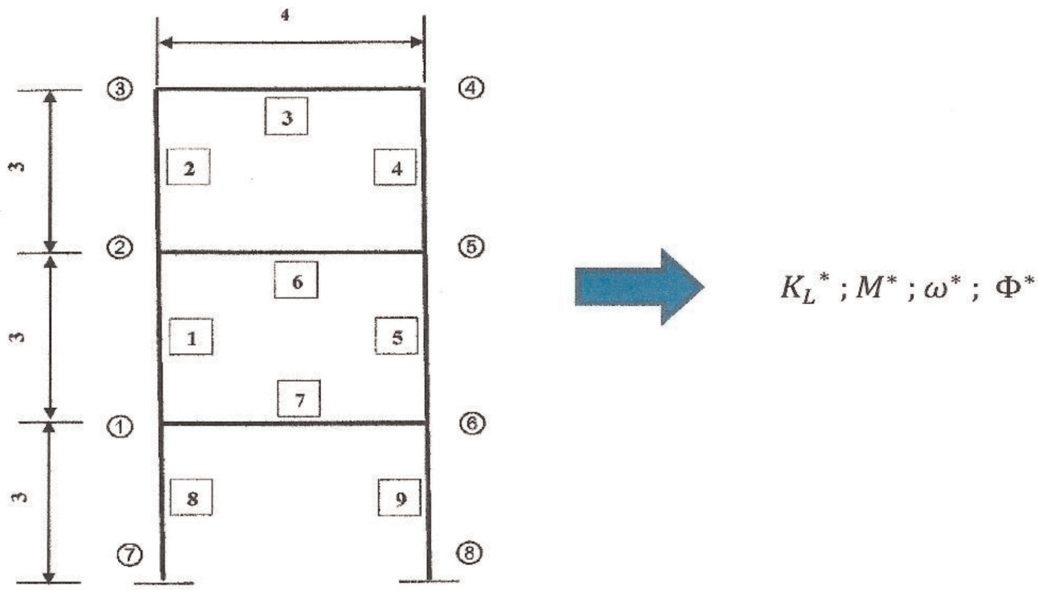



**Figure 11.**
*Framework with damage, points represent hinges (damage).*

modal matrices [18, 19]. With this, we aimed at the ANN defining matrix $K_L{}^*; M^*$, which would correspond to the said modal matrix. Once the ANN was trained and it provided searched for data for $K_L{}^*; M^*$, the failure condition was taken into account, in this case a service limit condition. Such service limit condition was defined as displacement of mezzanine $u^i_o$ and $u^i_j$ as well as differential displacements $\Delta^i_o$ y $\Delta^i_j$. Therefore, if

$$u^i_j > u^i_o \ \textit{Structure fails}$$

$$\Delta^i_j > \Delta^i_o \ \textit{Structure fails}$$

Here the subscript shows the case of damage, and the superscript shows the number of levels. Thus, $u^i_o$ shows the case of Damage Zero of level $i$. To obtain $u_o$, $\Delta_0$, $u_j$, and $\Delta_j$, the load condition was defined as the spectrum of design in Mexico City. For the calculation of $u_o$ and $\Delta_0$, the Damage Zero framework data were used $K_L$ and $M$, but for the calculation of $u_j$ and $\Delta_j K_L{}^*$ and $M^*$, the ANN gives as a result after having been trained were used, providing a modal matrix that may be obtained through an environmental vibration test.

## 6. Conclusions and discussion

According to Test 1, the ANN works properly calculating the objective matrix in an efficient manner, with an average 5% error. This may be lower if the number of training epochs of the ANN increases.

From Test 2 it can be concluded that, if training cases increase, the number of epochs required for learning of the ANN is higher. This is because input data of the ANN is larger, and therefore, more time is required for training. Also, from Test 2 it is concluded that the ANN has a 16% error when calculating the objective matrix, which corresponds to the increase of training cases. However, if the number of training epochs increases, the error of the ANN will decrease because the ANN will have more learning time.

Cases of serious damage are those where hinges are in intermediate elements, and cases where damage is of the same magnitude are those where hinges are in the higher and lower corners of the framework.

A hinge column on the base is more unfavorable than a hinge column in higher floors, but numerically, damage is higher when there are hinge elements in higher floors.

It may be interesting to see the damage level and period relation. How is period and damage related, the level of damage and period elongation is consistent. How much different damage levels coincide with the same period value and vice versa. But these points of interest may not be addressed in this work, since there was no parameter to measure the damage level on the framework. Damage is only found with changes in modal parameters, as well as with the increase of the vibration period.

Upon completion of the research, we have concluded that the methodology implemented to detect structural damage is rather simple. It was carried out in four steps: (i) extraction of modal parameters and condensed matrix, (ii) establishing failure condition for a serious damage condition, (iii) treatment of modal data to be used in the development of the ANN, and (iv) detection of damage to be carried out with the artificial neural network.

## Acknowledgements

This article and its corresponding research were carried out, in part, with the research projects IPN SIP-20181253 and IPN SIP-20196119. The authors thank the reviewers for carefully reading the paper and for their constructive comments and suggestions which have improved this paper.

## Annex I

Constructiong of an artificial neural network for detection of structural damage (**Tables I1–I4**).

| | | Principal stiffness kg/m | | |
|---|---|---|---|---|
| **Damage type** | **Level** | **1** | **2** | **3** |
| SD0 | *0* | 55047511.35 | 43986421.33 | 15198916.05 |
| D1 | *1* | 54877708.7 | 36097371.39 | 9399138.702 |
| D2 | *2* | 54771544.01 | 35317166.31 | 9121527.737 |
| D3 | *3* | 54771544.01 | 35317166.31 | 9121527.737 |
| D4 | *4* | 54877708.7 | 36097371.39 | 9399138.702 |
| D5 | *5* | 52859001.57 | 30346342.81 | 9268800.799 |
| D6 | *6* | 53635426.09 | 32269306.5 | 9737254.051 |
| D7 | *7* | 51836691.31 | 43952306.34 | 13441604.94 |
| D8 | *8* | 51573961.66 | 43939537.94 | 13450263.27 |
| D9 | *9* | 44998259.72 | 29296754.08 | 13808325.01 |

| Principal stiffness kg/m | | | | |
|---|---|---|---|---|
| **Damage type** | **Level** | **1** | **2** | **3** |
| D10 | *10* | 46785291.09 | 31313923.98 | 14088832.2 |
| D11 | *11* | 38598400.77 | 34400828.39 | 14998080.93 |
| D12 | *12* | 41125388.66 | 35987149.14 | 15086548.41 |
| D13 | *13* | 54,908,398 | 41044048.98 | 15126241.06 |
| D14 | *14* | 54923344.34 | 40895188.06 | 15112225.05 |
| D15 | *15* | 36667939.84 | 41919304.71 | 15169346.77 |

**Table I1.**
*Matrix stiffness formed with the principal diagonal of each condensed matrix $K_L$.*

| Periods | | | | |
|---|---|---|---|---|
| **Damage type** | **Level** | **1** | **2** | **3** |
| SD0 | *0* | 0.299830 | 0.090200 | 0.050792 |
| D1 | *1* | 0.318255 | 0.104710 | 0.052938 |
| D2 | *2* | 0.318365 | 0.105415 | 0.053246 |
| D3 | *3* | 0.318365 | 0.105415 | 0.053246 |
| D4 | *4* | 0.318255 | 0.104710 | 0.052938 |
| D5 | *5* | 0.303981 | 0.104247 | 0.055981 |
| D6 | *6* | 0.303965 | 0.103230 | 0.054903 |
| D7 | *7* | 0.348754 | 0.096427 | 0.051103 |
| D8 | *8* | 0.348938 | 0.096669 | 0.051148 |
| D9 | *9* | 0.323188 | 0.090897 | 0.060636 |
| D10 | *10* | 0.323013 | 0.090790 | 0.058691 |
| D11 | *11* | 0.319120 | 0.092516 | 0.060238 |
| D12 | *12* | 0.318989 | 0.092179 | 0.058335 |
| D13 | *13* | 0.362777 | 0.092909 | 0.051041 |
| D14 | *14* | 0.362999 | 0.093013 | 0.051068 |
| D15 | *15* | 0.311588 | 0.100941 | 0.055899 |

*It is the matrix of entry to the network and target matrix.*

**Table I2.**
*Matrix of periods.*

| Weights correlation *K vs. P* | | | |
|---|---|---|---|
| 1 | 0.19607059 | 0.36461043 | −0.75429818 |
| 2 | 0.47564924 | −0.19211267 | −0.73301055 |
| 3 | 0.39640392 | −0.89248216 | 0.149395599 |

**Table I3.**
*Input matrix to the network weights.*

| Periods, input matrix, and target matrix | | | | |
|---|---|---|---|---|
| **Damage type** | **Level** | **1** | **2** | **3** |
| SD0 | *0* | 0.299830 | 0.090200 | 0.050792 |
| D1 | *1* | 0.318255 | 0.104710 | 0.052938 |
| D2 | *2* | 0.318365 | 0.105415 | 0.053246 |
| D3 | *3* | 0.318365 | 0.105415 | 0.053246 |
| D4 | *4* | 0.318255 | 0.104710 | 0.052938 |
| D5 | *5* | 0.303981 | 0.104247 | 0.055981 |
| D6 | *6* | 0.303965 | 0.103230 | 0.054903 |
| D7 | *7* | 0.348754 | 0.096427 | 0.051103 |
| D8 | *8* | 0.348938 | 0.096669 | 0.051148 |
| D9 | *9* | 0.323188 | 0.090897 | 0.060636 |
| D10 | *10* | 0.323013 | 0.090790 | 0.058691 |
| D11 | *11* | 0.319120 | 0.092516 | 0.060238 |
| D12 | *12* | 0.318989 | 0.092179 | 0.058335 |
| D13 | *13* | 0.362777 | 0.092909 | 0.051041 |
| D14 | *14* | 0.362999 | 0.093013 | 0.051068 |
| D15 | *15* | .0311588 | 0.100941 | 0.055899 |
| **Periods, output matrix** | | | | |
| **Damage type** | **Level** | **1** | **2** | **3** |
| SD0 | *0* | 0.30661 | 0.09916 | 0.05424 |
| D1 | *1* | 0.32025 | 0.09805 | 0.05403 |
| D2 | *2* | 0.32029 | 0.09801 | 0.05401 |
| D3 | *3* | 0.32029 | 0.09801 | 0.05401 |
| D4 | *4* | 0.32025 | 0.09805 | 0.05403 |
| D5 | *5* | 0.30896 | 0.09837 | 0.05378 |
| D6 | *6* | 0.30914 | 0.09847 | 0.05381 |
| D7 | *7* | 0.34395 | 0.09714 | 0.05503 |
| D8 | *8* | 0.34409 | 0.09713 | 0.05502 |
| D9 | *9* | 0.32224 | 0.09747 | 0.05483 |
| D10 | *10* | 0.32248 | 0.09763 | 0.05481 |
| D11 | *11* | 0.31927 | 0.09763 | 0.05467 |
| D12 | *12* | 0.31953 | 0.09779 | 0.05466 |
| D13 | *13* | 0.35497 | 0.09666 | 0.05549 |
| D14 | *14* | 0.35515 | 0.09665 | 0.05549 |
| D15 | *15* | 0.31459 | 0.09814 | 0.05409 |
| **Difference, target matrix, and output matrix** | | | | |
| **Damage type** | **Level** | **1** | **2** | **3** |
| SD0 | *0* | −0.006784 | −0.008962 | −0.003451 |
| D1 | *1* | −0.001994 | 0.006660 | −0.001092 |
| D2 | *2* | −0.001921 | 0.007404 | −0.000760 |

| Difference, target matrix, and output matrix | | | | |
|---|---|---|---|---|
| **Damage type** | **Level** | **1** | **2** | **3** |
| D3 | *3* | −0.001921 | 0.007404 | −0.000760 |
| D4 | *4* | −0.001994 | 0.006660 | −0.001092 |
| D5 | *5* | −0.004983 | 0.005873 | 0.002200 |
| D6 | *6* | −0.005173 | 0.004756 | 0.001091 |
| D7 | *7* | 0.004801 | −0.3000715 | −0.003928 |
| D8 | *8* | 0.004845 | −0.000458 | −0.003877 |
| D9 | *9* | 0.000947 | −0.006573 | 0.005807 |
| D10 | *10* | 0.000530 | −0.006841 | 0.003883 |
| D11 | *11* | −0.000149 | −0.005118 | 0.005573 |
| D12 | *12* | −0.000541 | −0.005614 | 0.003680 |
| D13 | *13* | 0.007803 | −0.003753 | −0.004452 |
| D14 | *14* | 0.007851 | −0.003637 | −0.004426 |
| D15 | *15* | −0.003003 | 0.002805 | 0.001814 |

**Table I4.**
*The ANN output data.*

## Author details

Francisco Casanova-del-Angel[1]*, Daniel Hernández-Galicia[1]
and Xochicale-Rojas Hugo Alberto[2]

1 Instituto Politécnico Nacional, Mexico City, Mexico

2 Instituto Mexicano del Seguro Social, Mexico City, Mexico

*Address all correspondence to: fcasanova@ipn.mx

## References

[1] Futao Z, Wu Y. A rapid structural damage detection method using integrated ANFIS and interval modeling technique. Applied Soft Computing. 2014;**25**:473-484. DOI: 10.1016/j.asoc.2014.08.043

[2] Rezvani K, Maia NMM, Sabour MH. A comparison of some methods for structural damage detection. ResearchGate. 2017;**42**(4):649-659. DOI: 10.24200/sci.2017.4494

[3] Ancona Lazcano AR, Salgado Estrada R, Zamora Castro SA, Marcial MF. Evaluación de Métodos de Detección de Daño en Estructuras mediante uso de Vibraciones. In: XVIII Congreso Nacional de Ingeniería Sísmica. Aguascalientes, Mexico: Sociedad Mexicana de Ingeniería Sísmica; 2011

[4] Demuth H, Beale M. Neuronal Network Toolbox. For Use with MATLAB. User's Guide. Version 4. The MathWorks Inc; 1992–2004

[5] Gilat A. Matlab una introducción Con Ejemplos prácticos. Barcelona, España: Editorial Reverté; 2006. ISBN: 8429150358, 9788429150353

[6] Rumelhart DE, Hinton GE, Williams RJ. Learning internal representations by error propagation. In: Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol. I. Cambridge, MA, USA: MIT Press; 1986. pp. 318-362. ISBN: 0-262-68053-X

[7] Rumelhart DE, McClelland JL, The PDP Research Group. Parallel Distributed Processing Exploration in the Microstructure of Cognition. Volume 1: Foundation. Cambridge, MA: MIT Press; 1986

[8] Wu X, Ghaboussi J, Garrett JH. Use of neural networks in detection of structural damage. Computers and Structures. 1992;**42**(4):649-659. DOI: 10.1016/0045-7949(92)90132-5

[9] Chen SS, Kim S. Neural networks based sensor signal monitoring of instrumented structures. In: Proc. Struct. Congr. XIII. 1995

[10] Marwala T. Damage identification using committee of neural networks. Journal of Engineering Mechanics. 2000;**126**(1):43-50 ASCE. ISBN (Print): 0733-9399

[11] Doebling SW, Farrar CR, Prime MB, Shevitz DW. Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in their Vibration Characteristics: A Literature Review, United States. 1996 DOI: 10.2172/249299

[12] Barrios R, Doz G, Iturrioz I. Detección de daño en estructuras utilizando propiedades dinámicas. In: Información Tecnológica, Centro de Información Tecnológica. Vol. 11. La Serena, Chile; 2000. pp. 117-122

[13] Computers and Structures Inc. Getting Started with SAP2000 Linear and Nonlinear Static and Dynamic Analysis and Design of Three-Dimensional Structures. Berkeley, California, USA; 2009

[14] Solís Ortiz J, Martinez A, de Castro SM. Implicaciones en el Diseño de Marcos de Concreto Reforzado de los Criterios Para Asegurar el Mecanismo Plástico Columna Fuerte-Viga débil. Puebla, México: Sociedad Mexicana de Ingeniería Estructural, A.C; 2002

[15] Ávila J, Meli R. Análisis de la Respuesta de Edificios típicos Ante el Sismo del 19 de Septiembre de 1985. Mexico City: Instituto de Ingeniería, UNAM; 1987

[16] Reglamento de Construcciones para el Distrito Federal (RCDF). Reglamento publicado en Gaceta Oficial del Distrito

Federal, el 29 de enero de 2004. Última Reforma Publicada en la Gaceta Oficial de la Ciudad de México; 15 de Diciembre de 2017

[17] Luis AS, Max BS. Reglamento de Construcciones Para el Distrito Federal. 2005. ISBN: 968-24-7188-5

[18] Çalık İ, Bayraktar A, Türker T, Ayan AO. Dynamic identification of historical Molla Siyah mosque before and after restoration. Research on Engineering Structures and Materials. 2017;**3**(3): 164-175. DOI: 10.17515/resm2016.82st0728

[19] Salgado R. Damage detection methods in bridges through vibration monitoring: Evaluation and application [tesis doctoral]. Guimaraes, Portugal: Universidad de Minho-ISISE; 2008. p. 320