SHM
Structural Health Monitoring

Structural Damage Detection using Independent Component Analysis

C. Zang,[1,]* M. I. Friswell[1] and M. Imregun[2]

[1]*Department of Aerospace Engineering, University of Bristol, Queen's Building,*
Bristol BS8 1TR, UK
[2]*Mechanical Engineering Department, Imperial College, Exhibition Road,*
London SW7 2BX, UK

This paper presents a novel approach to detect structural damage based on combining independent component analysis (ICA) extraction of time domain data and artificial neural networks (ANN). The advantage of using time history measurements is that the original vibration information is used directly. However, the volume of data, measurement noise and the lack of reliable feature extraction tools are the major obstacles. To circumvent them, the independent component analysis technique is applied to represent the measured data with a linear combination of dominant statistical independent components and the mixing matrix [A]. Such a representation captures the essential structure of the measured vibration data. The vibration features represented by the mixing matrix provide the relationship between the measured vibration response and the independent components and are then employed to build the simplified neural network model for damage detection. Two examples are included to demonstrate the effectiveness of the method. First, a truss structure with simulated displacement data was used, and the results show that healthy and damage states located in the nine elements may be classified. Second, a bookshelf structure together with measured time history data from 24 piezoelectric single axis accelerometers was used to demonstrate the approach on a physical structure. The results show the successful detection of the undamaged and damaged states with very good accuracy and repeatability.

Keywords structural damage detection · independant component analysis · neural networks · vibration · time domain identification

1 Introduction

Structural damage detection based on measured vibration data is becoming increasingly important not only for preventing catastrophic failures but also for uninterrupted operation and prolonged service life. Detailed surveys of damage detection methods are given in [1–3]. Generally speaking, damage detection techniques can be classified according to the type of measured data on which they are based. Modal parameters (natural frequencies and mode shapes), provide a substantial reduction in measured vibration data, and are often used for damage detection [4–7]. However, modal parameters are not always easy to interpret in terms of the mathematical model of a linear vibrating system and often the lower frequency modes are insensitive to small levels of local

*Author to whom correspondence should be addressed.
E-mail: c.zang@bristol.ac.uk

Vol 3(1): 0069–83
[1475-9217 (200403) 3:1;69–83 10.1177/1475921704041876]

damage. Methods based on the measured frequency response functions (FRFs), such as the FRF curvature method and the FRF quotient method, are given in [8,9]. Most damage detection algorithms dealing with FRFs use a validated FE model as a reference and attempt to find discrepancies between this reference model and the FRFs of some damaged specimen [10–12]. But, practical difficulties still remain in obtaining a reference model for comparison purposes. Alternatively, methods based on the raw time signals are given in [13,14]. An obvious advantage of using raw time domain data is that the original vibration information is captured without any signal processing degradation. For instance, Farrar et al. [15] applied an auto-regressive (AR) model for measured acceleration–time histories and selected the coefficients of the AR model as damage indicators. However, in spite of promising results, there are some major obstacles that remain unresolved, such as the difficulties in dealing with large volumes of data, inherent measurement noise, and lack of reliable feature extraction tools.

In recent years, techniques based on multivariate statistics and neural networks have been applied to structural damage detection [1,16]. Sohn et al. [17] introduced a method using the coupling of principal component analysis (PCA) with statistical process control in order to enhance the discrimination between features from the undamaged and damaged structures. Worden and Manson [18] implemented visualization and dimension reduction with PCA for damage detection. Zang and Imregun [19–21] utilized a PCA technique to condense the FRF data. The basic idea is to compute the so-called principal components (PC) of a test matrix, the rows of which are the actual measured FRFs. The PC-compressed FRFs are represented by their projections onto the most significant principal components and have the additional benefit of containing less measurement noise. Furthermore, Zang and Imregun used both supervised and unsupervised artificial neural networks (ANNs) to successfully detect the damage in the case of two representative structures: a railway wheel and space antenna.

More recently, Zang et al. [22] used the independent component analysis (ICA) technique to extract the dynamic features from measured vibration time histories. Examples of a four degree of freedom (DOF) undamped system and a three DOF system with viscous damping show that the 'measured' time domain data represented with a linear combination of dominant statistical independent components and the mixing matrix [A] capture the essential dynamic characteristics. The additional benefits are not only the ability to deal with relatively high measurement noise, but also the availability of higher-order statistical data that can be used during the damage detection process. Back [23] applied ICA to multivariate financial time series such as a portfolio of stocks and the results were compared with those obtained using PCA. The results show that the overall stock price can be reconstructed surprisingly well by using a small number of ICs, in contrast to the poor perform of PCA. Biswal and Ulmer [24] separated multiple signal components present in functional MRI data sets, and obtained better results using ICA rather than PCA.

In this paper, independent component analysis is combined with neural networks for structural damage detection. The extracted dynamic features from the ICA technique, represented by the mixing matrix, are employed to build a simplified artificial neural network. The methodology is applied to a truss structure with simulated displacement data and a bookshelf structure with measured time histories from 24 piezoelectric single axis accelerometers, in order to assess its feasibility.

2 Methodology

Vibration-based structural damage detection may be considered as a kind of pattern recognition paradigm. It consists of data acquisition, signal processing, feature extraction and data reduction and detection analysis. Rytter [25] described the structural damage state by four levels: existence, location, extent and prediction. The focus of this paper is to detect the structural damage based on feature extraction for time histories from different sensors and a detection technique using multiple-layer neural networks. Figure 1 shows the framework of the damage detection process.

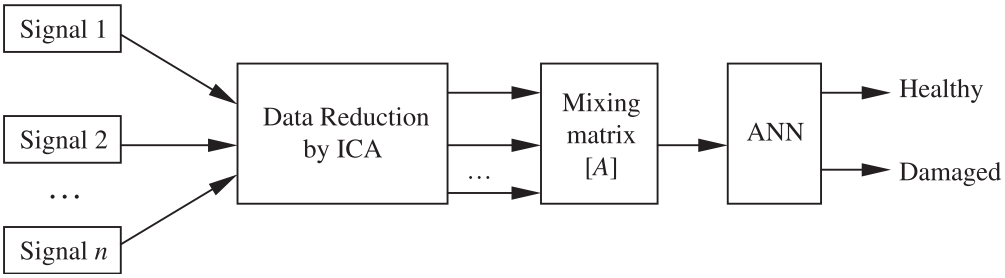

Figure 1 The damage detection process.

2.1 Data Reduction using the ICA Technique

In recent years, feature extraction methods such as blind signal separation (BBS) via independent component analysis (ICA) has been applied to speech enhancement, telecommunications and medical signal processing [26]. ICA techniques provide statistical signal processing tools for optimal linear transformations in multivariate data and these methods are well-suited to feature extraction, noise reduction, density estimation and regression [27, 28]. A summary of ICA techniques for extracting the dynamic features from measured vibration time histories will be given here for the sake of completeness [22].

The zero-mean matrix $[X]_{N\times T}=\{x_{ik}\}$ $(i=1,2,\ldots,N;\ k=1,2,\ldots,T)$ has N rows of measured time histories $\{x\}_i$ (where the mean of the data, $\bar{x}_i$, has been subtracted from each row individually) from N sensors, each with T time points. Each observation $x_i(t)$ can be considered to be a linear combination of M statistically independent sources, which are the individual elements of the vector $\{S(t)\}=\{s_1(t),s_2(t),\ldots,s_M(t)\}^T$. The sources $s_i(t)$ are called the independent components and have unit variance. The linear relationship between $\{X\}$ and $\{S\}$ is,

$$\{X(t)\}_{N\times 1}=[A]_{N\times M}\{S(t)\}_{M\times 1}=\sum_{i=1}^{M}\{a_i\}\ s_i(t) \quad (1)$$

where $[A]$ is an unknown *mixing* matrix and $\{a_i\}$ is the ith column of matrix $[A]$. The main issue is then the estimation of the mixing matrix $[A]$ and the realization of the source vector $\{S(t)\}$ using only the measured data vector $\{X(t)\}$.

The ICA algorithms normally find the independent components of a data set by minimizing or maximizing some measure of independence. Cardoso [29] gave a review of the solution to the ICA problem using various information theoretic criteria, such as mutual information, negentropy, maximum entropy and infomax, as well as the maximum likelihood approach. Here we will use the fixed-point algorithm [30,31] because of its suitability for handling raw time-domain data and good convergence properties. This algorithm will now be described briefly.

The first step is to pre-whiten the measured data vector $\{X\}$ by a linear transformation, to produce a vector $\{\tilde{X}\}$ whose elements are mutually uncorrelated and all have unit variance. A singular value decomposition (SVD) of the covariance matrix $[C]=\mathrm{E}[\{X(t)\}\{X(t)\}^T]$ yields,

$$[C]=[\Psi][\Sigma][\Psi]^T \quad (2)$$

where $[\Sigma]=\mathrm{diag}(\sigma_1,\sigma_2,\ldots,\sigma_n)$ is a diagonal matrix of singular values and $[\Psi]$ is the associated singular vector matrix. Then, the vector $\{\tilde{X}\}$ can be expressed as,

$$\{\tilde{X}(t)\}=[\Sigma]^{-1/2}[\Psi]^T\{X(t)\}. \quad (3)$$

An advantage of using an SVD-based technique is the possibility of noise reduction by discarding singular values smaller than a given threshold.

The second step is to employ the fixed-point algorithm. Define a separating matrix $[W]$ that transforms the measured data vector $\{X(t)\}$ to a vector $\{Y(t)\}$, such that all elements $y_i(t)$ are both mutually uncorrelated and have unit variance. Independent random variables are always uncorrelated, and hence ensuring the transformed variables are uncorrelated reduces the number of free parameters available and simplifies the problem [33]. The fixed-point algorithm then determines $[W]$ by maximising the absolute value of the kurtosis of $\{Y(t)\}$. The vector $\{Y(t)\}$ has the properties required for the independent components, and thus,

$$\{\hat{S}(t)\} = \{Y(t)\} = [W]\{\tilde{X}(t)\}. \tag{4}$$

From Equation (3), $\mathrm{E}[\{\tilde{X}(t)\}\{\tilde{X}(t)\}^T] = I$, and by definition we require $\mathrm{E}[\{Y(t)\}\{Y(t)\}^T] = I$. Hence, from Equation (4),

$$[W][W]^T = I, \tag{5}$$

and thus $[W]$ is an orthogonal matrix.

If we consider only one source signal at a time, the problem of estimating the filter matrix $[W]$ can be somewhat simplified. From Equation (4),

$$\hat{s}_i(t) = y_i(t) = \{w_i\}^T\{\tilde{X}(t)\} \tag{6}$$

where $\{w_i\}^T$ denotes the ith row of $[W]$. Using the deflation approach ([32]), $[W]$ may be estimated on a row-by-row basis, where each independent component is estimated separately. To estimate M independent components, the algorithm must be run M times. To ensure that a different independent component is estimated each time, a simple orthogonalising projection must be used inside the loop. Such a projection is possible because the rows of the filtering matrix $[W]$ are orthonormal to each other, from Equation (5). Hence the independent components are estimated by projecting the current solution vector $\{w_i\}$ onto the space orthogonal to the previously found rows of the filtering matrix $[W]$.

The kurtosis of the estimated signal $y_i(t)$ is defined as

$$\mathrm{kurt}(y_i(t)) = E[y_i^4(t)] - 3(E[y_i^2(t)])^2 \tag{7}$$

Kurtosis may be viewed as a normalised fourth-order moment of a signal and has the linearity properties $\mathrm{kurt}(x_1 + x_2) = \mathrm{kurt}(x_1) + \mathrm{kurt}(x_2)$ and $\mathrm{kurt}(\alpha x_1) = \alpha^4 \mathrm{kurt}(x_1)$, where α is a scalar. For a Gaussian distribution the kurtosis is zero, and thus kurtosis is used as a measure of the non-Gaussianity of a signal. The key to ICA is maximising the non-Gaussianity and thus maximising the absolute value of kurtosis [33].

From Equations (5)–(7), since the variance of $y_i(t)$ is unity,

$$\begin{aligned}\mathrm{kurt}(y_i(t)) &= \mathrm{kurt}(\{w_i\}^T\{\tilde{X}(t)\}) \\ &= E\left[(\{w_i\}^T\{\tilde{X}(t)\})^4\right] - 3,\end{aligned} \tag{8}$$

and the ith filtering vector $\{w_i\}$ may be obtained by maximising the kurtosis of $y_i(t)$ using a gradient descent type algorithm [30,33,34].

Following the estimation of the filtering matrix $[W]$ and the vector of independent components $\{S\}$, the mixing matrix $[A]$ is obtained by writing the independent components from Equations (3) and (4) as,

$$\{\hat{S}(t)\} = \{Y(t)\} = [W]\{\tilde{X}(t)\} = [W][\Sigma]^{-1/2}[\Psi]^T\{X(t)\} \tag{9}$$

and thus the mixing matrix, $[A]$, may be estimated, using the orthogonality of $[W]$ and $[\Psi]$, as,

$$[A] = [\Psi][\Sigma]^{1/2}[W]^T. \tag{10}$$

Therefore, using the ICA algorithm, the time responses from different sensors may be transformed into a linear mixture of higher-order statistically independent components and the original time histories from the ith spatial sensor may be reconstructed as

$$\hat{x}_i(t) = \sum_{j=1}^{M} a_{ij}s_j(t) + \bar{x}_i; \quad (i = 1, 2, \ldots, N; \quad j = 1, 2, \ldots, M) \tag{11}$$

where $\bar{x}_i$ denotes the mean response of the ith sensor. The matrix $[A]$ represents the relationship between the measured responses (inputs) and the independent components (outputs). Thus $[A]$ may be viewed as a transformation matrix between the time domain data and the characteristic dynamic features in the data. For very lightly damped, undamaged structures this transformation will be closely related to the most important modes of a structure, where the relative importance depends on the frequency range of interest, and the observability and controllability of the modes. For damaged structures, the independent components will also account for the effect of the damage on the dynamic response, and are therefore sensitive to the presence, location and severity of damage. Thus the mixing matrix $[A]$ may be used to build a simplified neural network model for damage detection.

2.2 Neural Networks for Damage Detection

Artificial Neural Networks (ANNs) provide a general, non-linear parameterised mapping between a set of inputs and a set of outputs. Once trained on available sample data, they can recognise patterns and hence they are ideally suited to signature analysis. Although many types of ANNs are used in practice, the multi-layer perceptron (MLP), trained using an error back-propagation (BP) algorithm, will be used here [35]. The network consists of an input layer, hidden layers and an output layer, and each layer is composed of a variable number of nodes. The mixing matrix from the ICA is used to define the data for the input nodes, while the output nodes show the state of the structure. The relationship between the input and output can be non-linear or linear, and the network's characteristics are determined by the weights assigned to the connections between the nodes in adjacent layers. Changing these weight will change the input to output behaviour of the network, and allow the network to be trained.

An ANN analysis consists of two stages, namely training and validation. During the training stage, an input-to-output mapping is determined iteratively using the available sample data. The actual output error, propagated from the current input set, is compared with the target output and the required compensation is transmitted backwards to adjust the node weights so that the error can be reduced at the next iteration. The learning stage is terminated once a pre-set error threshold is reached and the node weights are frozen at this point. During the validation stage, data from specimens with unknown properties are provided as input and the corresponding output is calculated using the fixed node weights. This provides a check on how well the network is able to *generalize* from the training data set to other data.

The design of the ANN requires the determination of the number of nodes in the hidden layer. The optimum number for a particular application may be found by looking at the results from the validation data set to ensure that the network generalizes sufficiently. Different numbers of hidden nodes may be tried, and the number is chosen that gives the lowest validation error. The resulting network is likely to provide the most robust result when further data is presented to the network, and provides a trade off between over fitting the data (too many hidden nodes) and having insufficient freedom to fit the data (too few hidden nodes). Experience has shown that the optimum number of hidden nodes is approximately half of the sum of the number of input and output nodes [36], and this choice has been used in this paper.

The remaining choice is the number of output nodes. To determine whether a structure is damaged or not requires only one output node, with, say, 0 for a healthy structure and 1 for a damaged structure. However, when more choices or patterns are available, for example to distinguish low or high damage levels, experience has shown that the number of output nodes in neural network should equal the number of patterns [37]. For consistency this approach will also be used for the detection of damage.

3 The Truss Structure

3.1 Preliminaries

The truss structure of Figure 2 was used to investigate the combined ICA/ANN damage detection technique. All nine elements had an elastic modulus of 200 GPa and a cross-sectional area of $2.5 \times 10^{-3}\,\mathrm{m}^2$. Damping was neglected. The horizontal and vertical displacements of Node 1 and the horizontal displacement of Node 6 were constrained. An impact load of 200 N was applied at Node 5 and the nine resulting displacement time histories at the nodes were computed. Displacement data is used in this example, and acceleration data in the bookshelf example considered later, to demonstrate the versatility of the approach in terms of the data requirements. Of course, in practice, the acceleration response would be measured.

Damage cases denoted DS1, DS2,..., DS9 were then introduced to the truss structure by reducing the stiffness of the nine axial members by 50% respectively, thus obtaining 81 time histories of damaged specimens. Including the healthy state of the truss, there was a total of 90 time histories. Figure 3 shows a time history comparison of damaged (dotted line) and healthy (solid line) specimens at node 5 in the horizontal direction. To provide a more realistic example, the 90 time history data sets were contaminated with 0, 5, 10, 15, 20, 25, and 30% noise at each time point, with a uniform distribution on the interval [0,1]. The detection process is robust with respect to the noise properties, and an example using Gaussian noise will be demonstrated later. Thus, 630 noise-corrupted sample data sets were obtained.

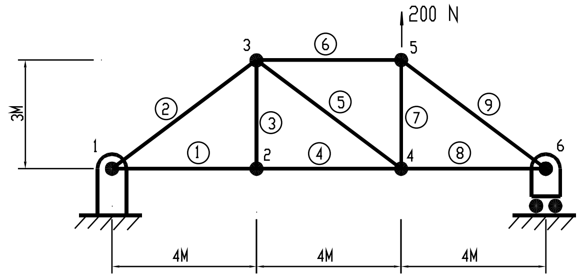

Figure 2 Truss structure example for damage detection.

3.2 ICA for Data Reduction

The 630 time histories, 63 × 1 corresponding to the healthy state and 63 × 9 corresponding to the damaged states from DS1 to DS9, were used to form a 630 × 301 matrix, where 301 is the number of points in each time history. Clearly, such a large data set cannot be used with ANNs

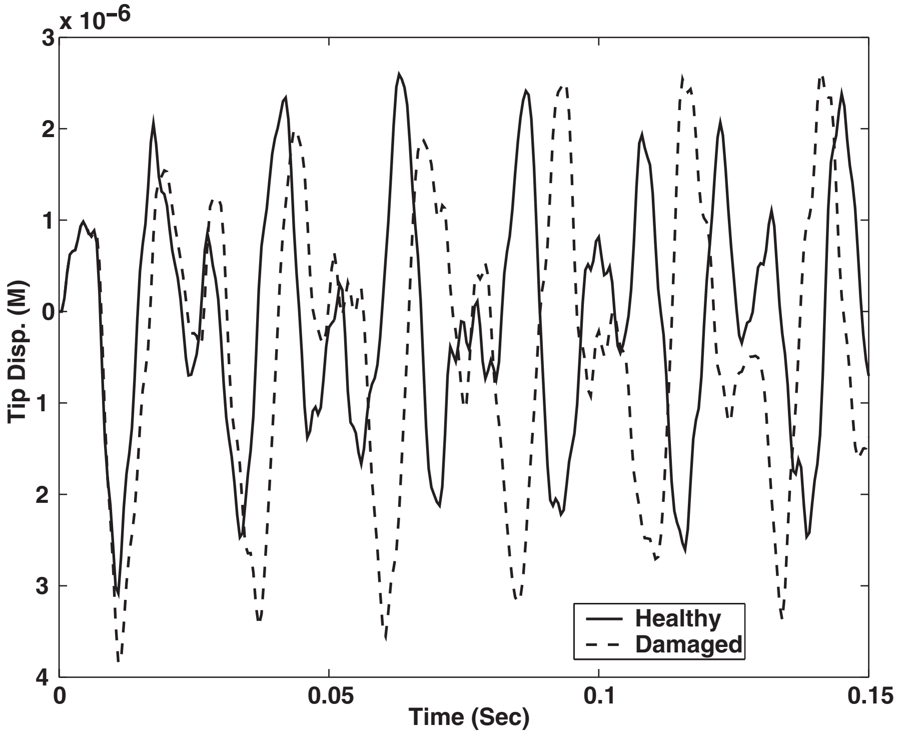

Figure 3 Time histories for healthy and damaged truss structure – node 5, horizontal direction.

directly and is the reason for the data reduction using ICA. During the pre-whitening stage (which is essentially a principal component analysis) the first ten dominant principal components containing 97.84% of the original information were retained. Then ten independent components and the corresponding mixing matrix, [A] (630 × 10), were estimated.

3.3 Neural Network for Damage Detection

Using ICA the original time history specimens are transformed to a linear combination of the mixing matrix [A] and the higher order independent components. The mixing matrix [A], which represents the dynamic characteristics of the structure and the damage, is then used as the input data to the neural network model for damage detection. The available 630 specimens were first divided into a training group of 540 and a validation group of 90. The training group contained 54 data sets for the healthy state with up to 30% added noise, and 54 data sets for each of the nine damage states, again with added noise. The validation group consisted of 90 specimens containing nine of each state respectively. The output of the network consisted of two nodes, representing the healthy and damage states, as shown in Table 1.

The number of nodes in the hidden layer was set at five, using the approach discussed earlier. Thus, a simple three-layer MLP network with ten input nodes, a five-node hidden layer, and two output nodes was built for further training and testing. During training, the learning and momentum rates for network were set at 0.6 and 0.3 respectively in the early stages but fell to 0.01 and 0.001 later (see, for example, [35] for the definition of learning and momentum rates). The network converged smoothly to RMS errors of 1.38% for training and 0.53% for validation in 3000 iterations. This result indicates that the neural network was stable and well trained. The output from the network is plotted in Figure 4, where the target and actual outputs are represented by solid and dotted lines respectively. All of the actual output closely matched the target output, although some small discrepancies are apparent in the healthy state and damage state 3 during the training. All specimens are classified correctly.

Table 1 Output node definition for damage detection of the truss structure.

State	*Node 1*	*Node 2*
Healthy (HS)	1	0
Damaged (DS1, DS2, . . . , DS9)	0	1

After the successful training of the network, a set of 90 data sets corrupted with noise, nine from each of the healthy and damaged states, were selected for validation. All of the data in each set was fed sequentially into the network and Figure 5 shows the actual output of the network, compared with the target output. All of the testing specimens were classified successfully.

The next example considers various damage levels and uses a different noise model, for the same truss structure. Damage was introduced to the truss structure by reducing the stiffness of the fourth axial member (corresponding to case DS4) by 10, 20, 30, 50, 70 and 90% respectively. Nine data sets for each healthy or damaged state were used. The time history data sets were then contaminated by Gaussian noise with a standard deviation in the range 0–30%, in steps of 5%, to give 441 data sets. As before, the first ten independent components were estimated, containing 96.56% of the original information and the corresponding mixing matrix [A] (441 × 10) obtained. The neural network model was built with a 10-node input layer, one six-node hidden layer, and a three-node output layer. Table 2 gives the output definition. A stiffness reduction of less than 50% is defined as light damage, while a stiffness reduction of 50% or above is taken to be heavy damage.

The network was trained with 378 specimens and validated with 63. The network output from the training phase is shown in Figure 6, where the target and actual outputs are represented by a dotted line and asterisks respectively. The actual output closely matched the target output, although some small fluctuations are present in the healthy and light damage states in output nodes 1 and 2. Figure 7 gives the network output for the validation phase, compared to the target

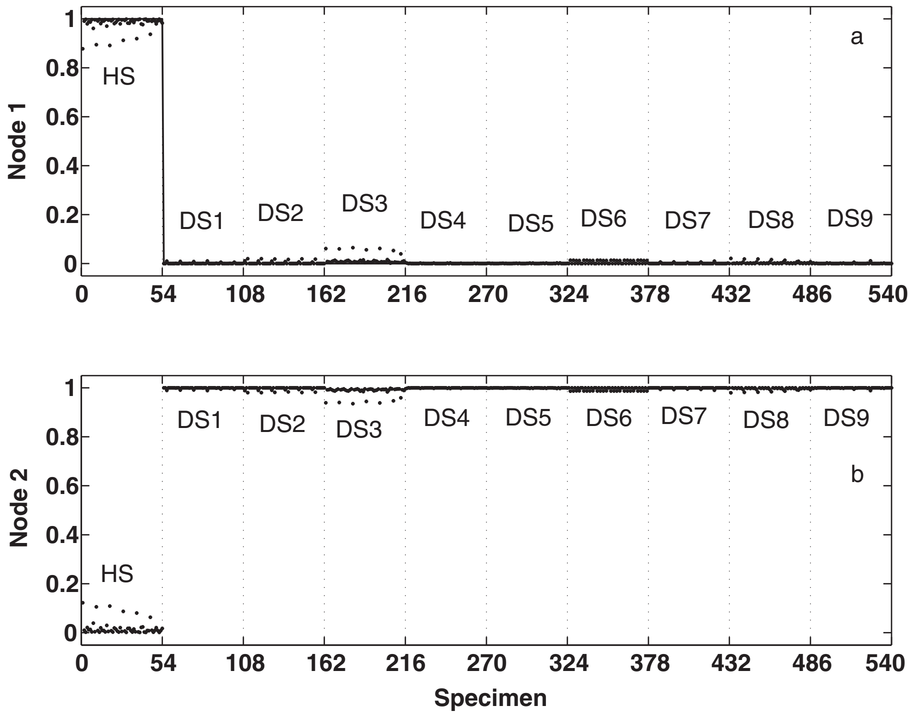

Figure 4 Network outputs from the training phase for the truss structure (Target output: solid line, actual output: dotted line).

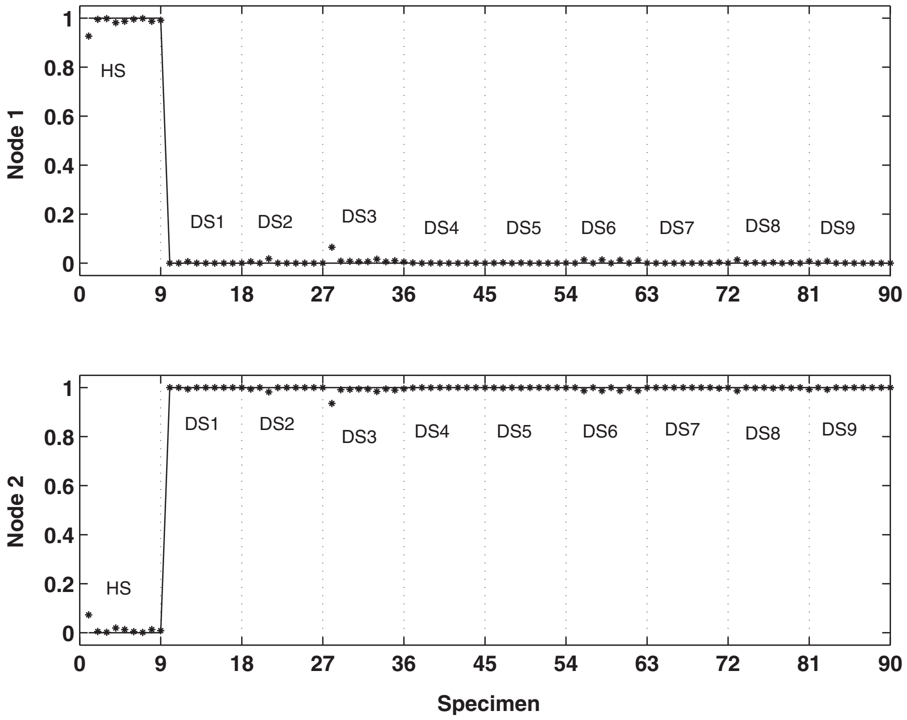

Figure 5 Network outputs from the validation phase for the truss structure (Target: solid line, actual output: asterisk).

output, and shows that all of the validation data sets are classified correctly.

4 The Bookshelf Structure

4.1 Experimental Data

The experimental example is a three-story bookshelf structure, shown in Figure 8. The testing was performed at the Los Alamos National Laboratory and the time histories data are available from the Los Alamos website [38]. The structure was constructed from Unistrut columns and aluminium floor plates with two-bolt connections to brackets on each Unistrut. Twenty-four piezoelectric single axis accelerometers, two per joint, were mounted on aluminium blocks, giving a total of eight for each plate. A shaker was attached at corner D using a stinger connected to a tapped hole at the mid-height of the base plate.

Table 2 Output definition for damage assessement of the truss structure.

State of DS4	*Node 1*	*Node 2*	*Node 3*
Healthy	1	0	0
Light damage (< 50%)	0	1	0
Heavy damage (≥50%)	0	0	1

Consider damage introduced at the 1C location. Table 3 gives the corresponding damaged configurations for this location. The measurements from 24 sensors were repeated 10 times for the undamaged state and five times for each damaged state. Therefore, a total of 480 data sets were available.

4.2 Data Reduction and Damage Detection

The 480 time histories were used to form a 480×2048 matrix, where 2048 is the number of points in each measured time history. By discarding the less significant singular values in the step of data whitening, the first 31 significant independent components were selected and the corresponding mixing matrix $[A]$ (31×480) was obtained using the fixed-point ICA algorithm. To estimate the extraction quality, the time histories

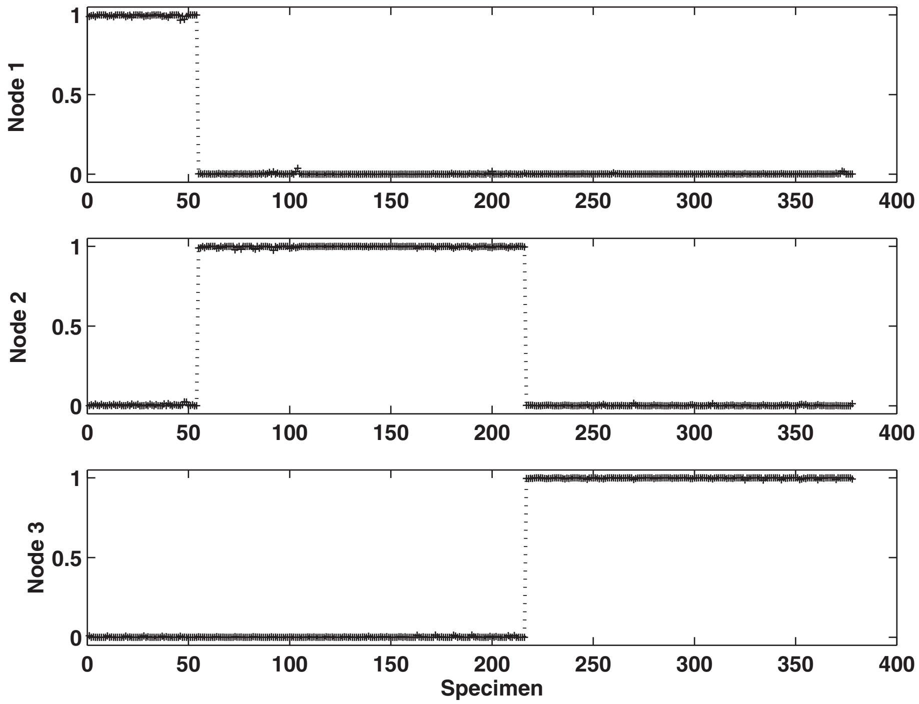

Figure 6 Network output from the training phase for the truss structure (Target output: dotted line, actual output: asterisk).

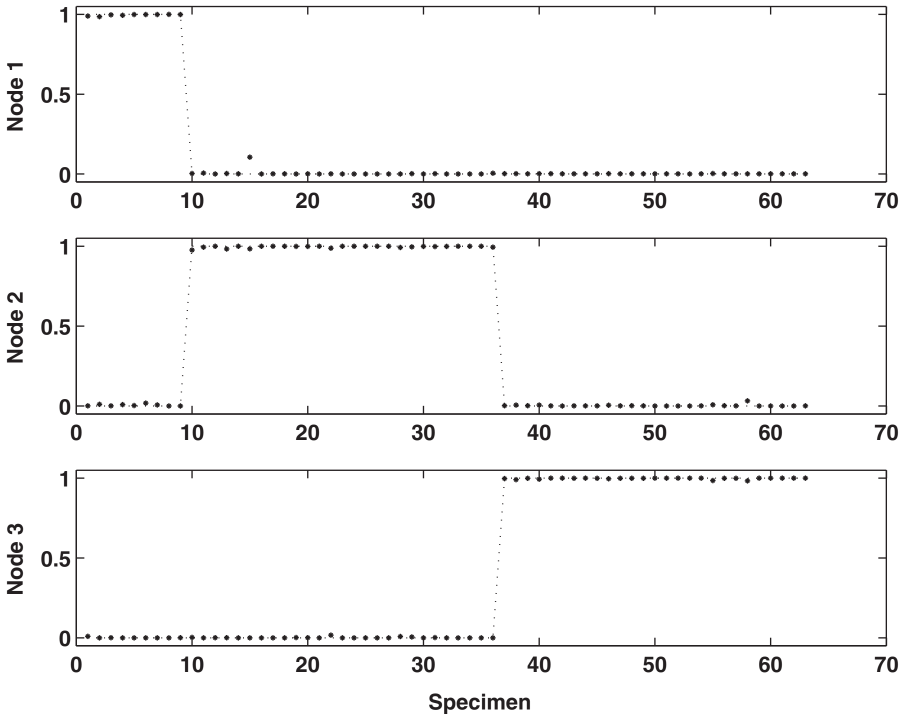

Figure 7 Network output from the validation phase for the truss structure (Target output: dotted line, actual output: asterisk).

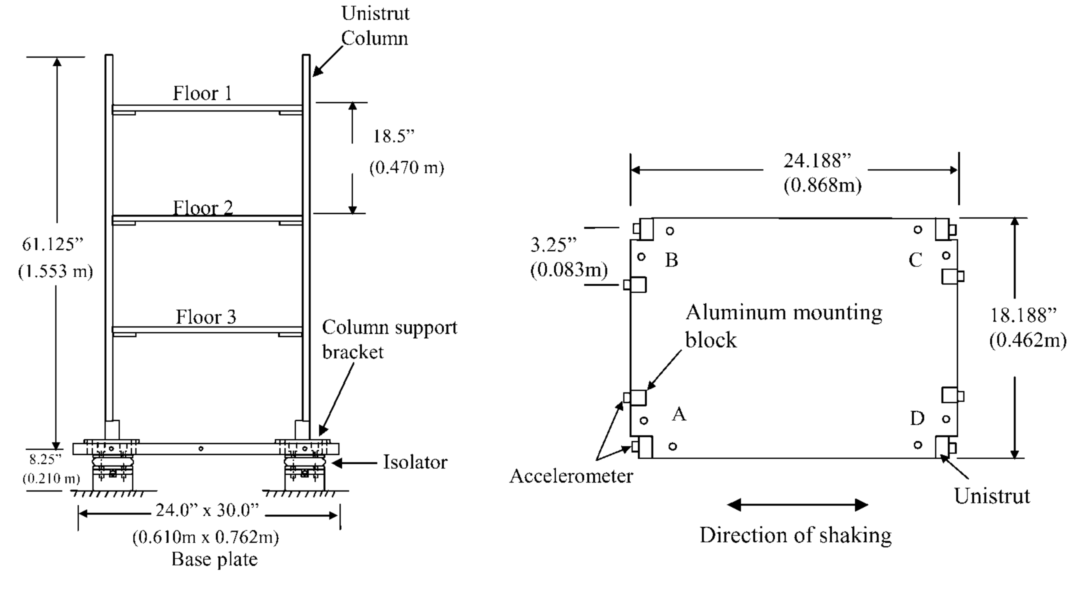

Figure 8 The three-story bookshelf structure used for damage detection (from [38]).

were reconstructed using the 31 independent components and the corresponding mixing matrix [*A*]. The reconstructed signals are compared with the original time histories, for a limited time range, in Figures 9–11. As can be seen, excellent matches between the original and the reconstructed responses were achieved, and the independent components clearly characterise both the

Table 3 The undamaged and damaged states for the bookshelf structure.

State	*Definition*	*Measured Time Histories*	*Data Sets*
Undamaged (UDS)	The original state of the structure	10	240
Damaged (DS1)	Removal of bolts between the bracket and the plate at the 1C location	5	120
Damaged (DS2)	Complete removal of the bracket at the 1C location	5	120

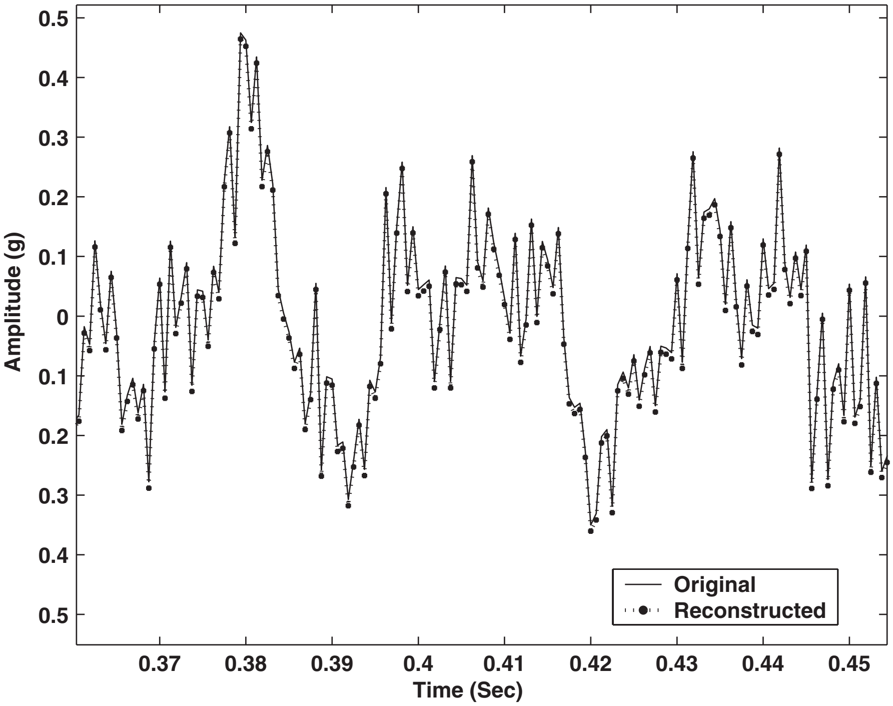

Figure 9 Original and the reconstructed time histories for sensor 10: undamaged state for the bookshelf structure.

dynamics of the healthy structure and the effects of damage.

A neural network model was built with a 31-node input layer, one 16-node hidden layer, and a two-node output layer. The output definition is given in Table 4. The available 480 data sets were randomly shuffled and divided into three groups: a training group with 300 data sets, a network validation group with 60 data sets and a detection or testing group with 120 data sets. The use of a validation data set to ensure the proper generalization of a network has already been discussed. In this case a third data set, the detection group, is used to access the overall performance of the approach. Table 5 gives the network performance during training, validation and detection.

The final errors for training and validation are very close and the detection error is less than 2.5%. In total, seven of the 480 data sets failed to be trained or estimated correctly by the network. The problems with all of the failed data sets were traced to errors in measured degrees of freedom. Four data sets corresponded to measurement point 1 (3BP) in the undamaged state test (UDS), and two samples to measurement point 24 (1DC) in the damaged states (one for DS1 and one for DS2). The seventh data

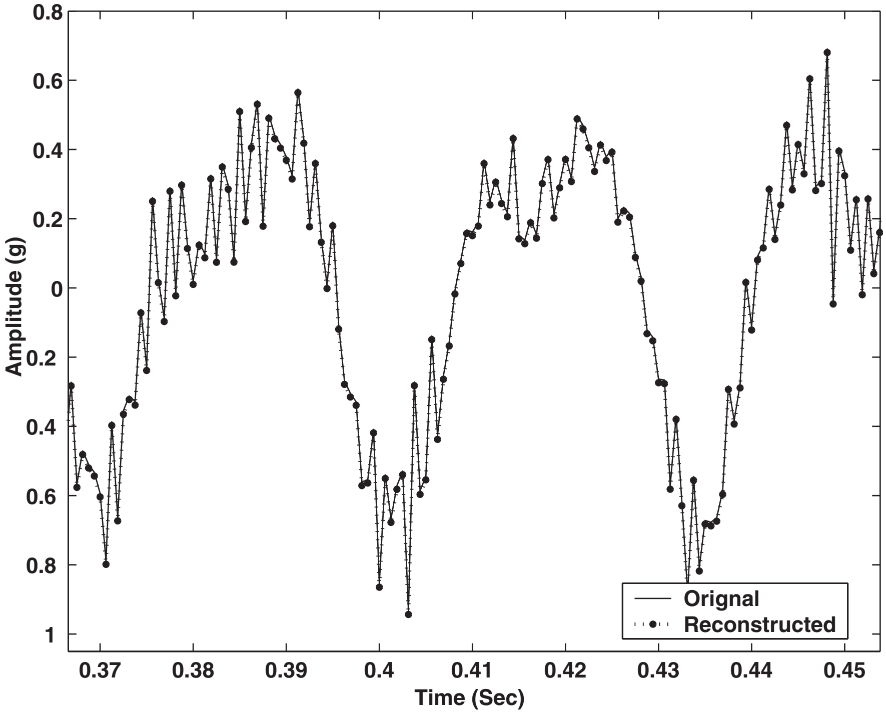

Figure 10 Original and the reconstructed time histories for sensor 10: damaged state (DS1) for the bookshelf structure.

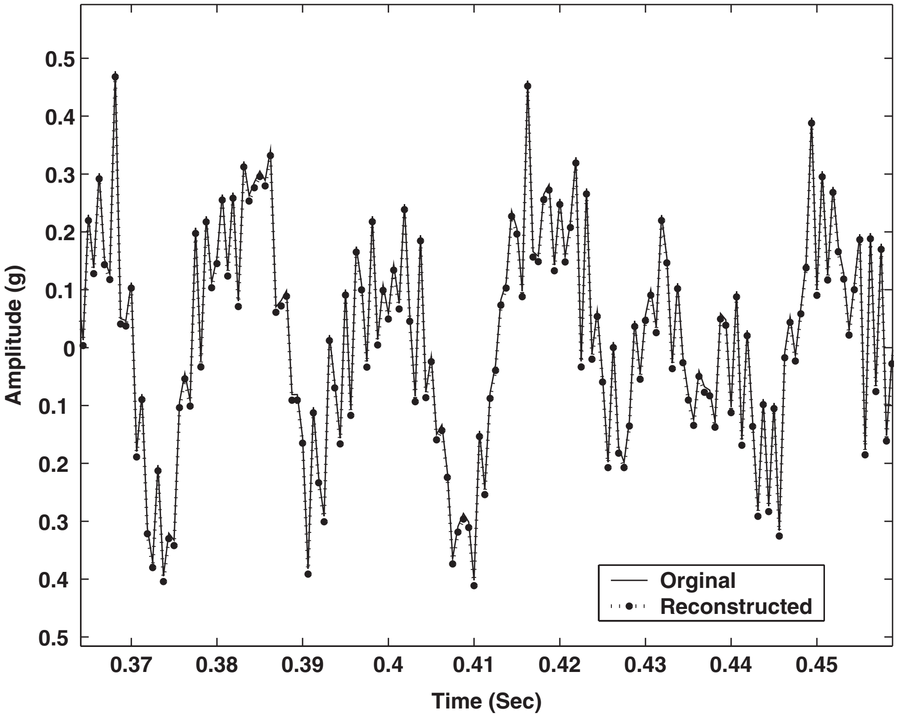

Figure 11 Original and the reconstructed time histories for sensor 10: damaged state (DS2) for the bookshelf structure.

Table 4 Output definition for the damage in the bookshelf structure.

State	*Node 1*	*Node 2*
Undamaged (UDS)	1	0
Damaged (DS1, DS2)	0	1

Table 5 The results of neural network training, validation and detection.

Name	*Samples*	*Success*	*Failed*	*Error Rate* (%)
Training	300	297	3	1.00
Validation	60	59	1	1.67
Detection	120	117	3	2.50

set corresponded to measurement point 23 (1DP) in the DS1 damaged state, and the problems may be due to measurement noise, or a lack of sensitivity of the measured degree of freedom to the damage.

5 Conclusions

The results demonstrate that data reduction from time domain data using independent component analysis, followed by neural networks for damage detection, provides a suitable methodology for structural health monitoring. Independent component analysis is a powerful tool to decompose signals into uncorrelated independent components. The corresponding mixing matrix encodes the dynamic characteristics of the structure and the effect of damage to enable structural damage detection using a neural network from the measured time domain data. Such a route has the advantage of reducing not only the size of the 'measured' data set, but also provides robustness to noise contamination. Furthermore, higher order statistics are available that may be useful for damage identification. A major benefit of the technique is that it requires the vibration response to be measured but not the excitation force, and this feature is very useful for on-line industrial applications. The multi-layer perceptron neural network, trained using an error back propagation algorithm is demonstrated to be effective for damage detection. However, this approach relies on supervised learning and requires sufficient data from several different damaged and healthy states to train the networks. Real data for this training is likely to be difficult to obtain in practical applications. Investigations are continuing into the use of simulated data to train networks, and the effect of modelling errors on the detection performance.

The number of independent components selected reduces the size of the measured data and filters unwanted measurement noise, but also determines how much original information is contained in the reconstructed signals. The use of too few components will lose some characteristics of the data that the neural network would find useful in the detection phase. The use of too many components requires the neural network to filter the noise as well as detect the damage. Work is continuing on the optimization of both the number of independent components and the network structure (i.e. the number of input and hidden nodes) and will be explored in a forthcoming paper.

References

1. Doebling, S.W., Farrar, C.R., Prime, M.B. and Shevitz, D.W. (1996). Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in their Vibration Characteristics: A Literature Review. Los Alamos National Laboratory Report No. LA-13070-MS.
2. Doebling, S.W., Farrar, C.R and Prime, M.B. (1998). Summary review of vibration-based damage identification methods. *Shock and Vibration Digest*, *30*(*2*), 91–105.
3. Friswell, M.I. and Penny, J.E.T. (1997). Is damage location using vibration measurements practical? *DAMAS 97, Structural damage assessment using advanced signal processing procedures*, pp. 351–362, Sheffield, UK.
4. Contursi, T., Messina, A. and Williams, E.J. (1998). A multiple damage location assurance criterion based on natural frequency changes. *Journal of Vibration and Control*, *4*(*5*), 619–633.
5. Rizos, P.F., Aspragathos, N. and Dimarogonas, A.D. (1990). Identification of crack location and magnitude in a cantilever beam from the vibration modes. *Journal of Sound and Vibration*, *138*(*3*), 381–388.
6. Topole, K.G. and Stubbs, N. (1995) Non-destructive damage evaluation in complex structures from a

minimum of modal parameters. *International Journal of Analytical and Experimental Modal Analysis*, *10*(*2*), 95–103.

7. Lam, H.F., Ko, J.M. and Wong, C.W. (1998). Localization of damaged structural connections based on experimental modal and sensitivity analysis. *Journal of Sound and Vibration*, *210*(*1*), 91–115.
8. Maia, N.M.M., Silva, J.M.M., Ribeiro, A.M.R. and Sampaio, R.P.C. (1999). On the use of frequency-response-functions for damage detection. *2nd International Conference on Identification in Engineering Systems*. (pp. 460–471), Swansea, UK.
9. Sampaio, R.P.C., Maia, N.M.M. and Silva, J.M.M. (1999). Damage detection using the frequency-response-function curvature method. *Journal of Sound and Vibration*, *226*(*5*), 1029–1042.
10. Fritzen, C.P. and Bohle, K. (1999). Identification of damage in large scale structures by means of measured FRFS – procedure and application to the I40-highway-bridge. *DAMAS 99*, Dublin, Ireland, Key Engineering Materials, (*167–168*, 310–319).
11. Wang, Z., Lin, R.M. and Lim, M.K. (1997). Structural damage detection using measured FRF data. *Computer Methods in Applied Mechanics and Engineering*, *147*(*1–2*), 187–197.
12. Thyagarajan, S.K., Schulz, M.J., Pai, P.F. and Chung, J. (1998). Detecting structural damage using frequency response functions. *Journal of Sound and Vibration*, *210*(*1*), 162–170.
13. Lew, J.S., Sathananthan, S. and Gu, Y. (1997). Comparison of damage detection algorithms using time domain data. *IMAC 15*, (pp. 645–651), Orlando, Florida.
14. Cattarius, J. and Inman, D.J. (1997). Time domain analysis for damage detection in smart structures. *Mechanical Systems and Signal Processing*, *11*(*3*), 409–423.
15. Farrar, C.R., Duffey, T.A., Doebling, S.W. and Nix D.A. (1999). A statistical pattern recognition paradigm for vibration-based structural health monitoring. *2nd International Workshop on Structural Health Monitoring*. (pp. 764–773), Lancaster, PA: Technomic Publishing.
16. Zeng, P. (1998). Neural computing in mechanics. *Applied Mechanics Reviews*, *51*(*2*), 173–197.
17. Sohn, H., Czarnecki, J.A. and Farrar, C.R. (2000). Structural health monitoring using statistical process control. *Journal of Structural Engineering, 126*(*1*), 1356–1363.
18. Worden, K. and Manson, G. (1999). Visualization and dimension reduction of high-dimensional data for damage detection. *IMAC 17*, (pp. 1576–1585), Kissimmee, Florida.
19. Zang, C. and Imregun, M. (2001). Structural damage detection using artificial neural networks and measured FRF data reduced via principal component projection. *Journal of Sound and Vibration*, *242*(*5*), 813–827.
20. Zang, C. and Imregun, M. (2000). FRF-based structural damage detection using Kohonen self-organizing maps. *International Journal of Acoustics and Vibration*, *5*(*4*), 167–172.
21. Zang, C. and Imregun, M. (2001). Combined neural network and reduced FRF techniques for slight damage detection. *Journal of the Archive of Applied Mechanics*, *71*(*8*), 525–536.
22. Zang, C., Friswell, M.I. and Imregun, M. (2002). Decomposition of time domain vibration signals using the independent component analysis technique. *3rd International Conference on Identification Engineering System*. (pp. 434–445), Swansea, UK.
23. Back, A.D. (1997). A first application of independent component analysis to extracting structure from stock returns. *International Journal of Neural Systems*, *8*(*4*), 473–484.
24. Biswal, B.B and Ulmer, J.L. (1999). Blind source separation of multiple signal sources of MRI data sets using independent component analysis. *Journal of Computer Assisted Tomography*, *23*(*2*), 265–271.
25. Rytter, T. (1993). Vibration based inspection of civil engineering structure. PhD Dissertation, Department of Building Technology and Structure Engineering, Aalborg University, Denmark.
26. Lee, T. (1998). *Independent component analysis*: *theory and applications*, Kluwer Academic Publishers, Boston.
27. Barlett, M.S., Lades, M.H. and Sejnowski, T.J. (1998). Independent component representation for face recognition. *SPIE symposium on electronic imaging*: *science and technology*. (pp. 528–539), San Jose, California.
28. Hyvärinen, A. (1999). Gaussian moments for noisy independent component analysis. *IEEE Signal Processing Letters*, *6*(*6*), 145–147.
29. Cardoso, J.F. (1998). Blind signal separation: statistical principles. *Proceedings of the IEEE*, *86*(*10*), 2009–2020.
30. Hyvärinen, A. and Oja, E. (1997). A fast fixed-point algorithm for independent component analysis. *Neural Computation*, *9*, 1483–1492.
31. Hyvärinen, A. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, *10*(*3*), 626–634.
32. Delfosse, N. and Loubaton, P. (1995). Adaptive blind separation of independent sources: a deflation approach. *Signal Processing*, *45*, 59–83.
33. Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks, 13*(*4-5*), 411–430.

34. Hyvärinen, A. and Oja, E. (1996). A neuron that learns to separate one independent component from linear mixtures. *IEEE International Conference on Neural Networks*. (*ICNN 96*) (pp. 62–67), Washington, DC.
35. Bishop, C.M. (1995). *Neural networks for pattern recognition*, Oxford: Oxford University Press.
36. Masters, T. (1993). *Practical neural network recipes in C++*. Academic Press, London.
37. Tarassenko, L. (1998). *A guide to neural computing applications*. Neural Computing Application Forum, Eliane Wigzell.
38. Los Alamos National Laboratory Website: http://www.lanl.gov/projects/damage_id/index.htm