CAIT-UTC-042

# Addressing the Issue of Insufficient Information in Data-Based Bridge Health Monitoring

Final Report
November 2015

**Raimondo Betti**
Professor
Columbia University
New York NY 10027

External Project Manager
Dyab Khazem
Parsons Transportation Group

In cooperation with
Rutgers, The State University of New Jersey
And
State of New York
Department of Transportation
And
U.S. Department of Transportation
Federal Highway Administration

# Disclaimer Statement

The contents of this report relfect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the Department of Transportation, University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

TECHNICAL REPORT STANDARD TITLE PAGE

| | | |
|---|---|---|
| 1. Report No.<br>CAIT-UTC-042 | 2. Government Accession No. | 3. Recipient's Catalog No. |
| 4. Title and Subtitle<br>Addressing the Issue of Insufficient Information in Data-Based Bridge Health Monitoring | | 5. Report Date<br>November 2015<br>6. Performing Organization Code<br>CAIT/Columbia |
| 7. Author(s)<br>Raimondo Betti | | 8. Performing Organization Report No.<br>CAIT-UTC-042 |
| 9. Performing Organization, Name and Address<br>Columbia University<br>New York, NY 10027 | | 10. Work Unit No.<br>11. Contract or Grant No.<br>DTRT12-G-UTC16 |
| 12. Sponsoring Agency Name and Address<br>Center for Advanced Infrastructure and Transportation<br>Rutgers, The State University of New Jersey<br>100 Brett Road<br>Piscataway, NJ 08854 | | 13. Type of Report and Period Covered<br>Final Report<br>12/01/13 - 11/30/2015<br>14. Sponsoring Agency Code |

15. Supplementary Notes

U.S Department of Transportation/Research and Innovative Technology Administration
1200 New Jersey Avenue, SE
Washington, DC 20590-0001

16. Abstract

One of the most efficient ways to solve the damage detection problem using the statistical pattern recognition approach is that of exploiting the methods of outlier analysis. Cast within the pattern recognition framework, damage detection assesses whether the patterns of the damage sensitive features extracted from the response of the system under unknown conditions depart from those drawn by the features extracted from the response of the system in a healthy state. The metric dominantly used to measure the testing feature's departure from the trained model is the Mahalanobis Squared Distance (MSD). Evaluation of MSD requires the use of the inverse of the training population's covariance matrix. It is known that when the feature dimensions are comparable to the number of observations, the covariance matrix is ill-conditioned and numerically problematic to invert. When the number of observations is smaller than the feature dimensions, the covariance matrix is not even invertible.

In this work, four alternatives to the canonical damage detection procedure were investigated to address the issue: data compression through Discrete Cosine Transform, use of pseudo-inverse of the covariance matrix, use of shrinkage estimate of the covariance matrix, and a combination of the three techniques. The performance of the four methods was first studied for solving the damage identification problem on simulated data from a four DOFs shear-type system, and on experimental data recorded on a four story steel frame excited at the base by means of the shaking table facility available at the Carleton Laboratory at Columbia University. Finally, the proposed techniques were also investigated in the context of damage location applications on simulated data from a bridge deck model.

| | | | |
|---|---|---|---|
| 17. Key Words<br>Structural Health Monitoring, Bridges, Damage Detection, Data-based Models | | 18 Distributional Statement | |
| 19. Security Classification<br>Unclassified | 20. Security Classification (of this page)<br>Unclassified | 21. No. of Pages<br>84 | 22. Price |

Form DOT F 1700.7 (8-09)

# Contents

# List of Figures

# List of Tables

## Executive Summary

One of the most efficient ways to solve the damage detection problem using the statistical pattern recognition approach is that of exploiting the methods of outlier analysis. Cast within the pattern recognition framework, damage detection assesses whether the patterns of the damage sensitive features extracted from the response of the system under unknown conditions depart from those drawn by the features extracted from the response of the system in a healthy state. The metric dominantly used to measure the testing feature's departure from the trained model is the Mahalanobis Squared Distance (MSD). Evaluation of MSD requires the use of the inverse of the training population's covariance matrix. It is known that when the feature dimensions are comparable to the number of observations, the covariance matrix is ill-conditioned and numerically problematic to invert. When the number of observations is smaller than the feature dimensions, the covariance matrix is not even invertible.

In this work, four alternatives to the canonical damage detection procedure were investigated to address this issue: data compression through Discrete Cosine Transform, use of pseudo-inverse of the covariance matrix, use of shrinkage estimate of the covariance matrix, and a combination of the three techniques. The performance of the four methods was first studied for solving the damage identification problem on simulated data from a four DOFs shear-type system, and on experimental data recorded on a four story steel frame excited at the base by means of the shaking table facility available at the Carleton Laboratory at Columbia University. Finally, the proposed techniques where also investigated in the context of damage location applications on simulated data from a bridge deck model.

# 1. Introduction

Since transportation infrastructure forms the backbone of the economic well-being and progress of any nation, proper maintenance and timely rehabilitation of bridges to ensure their uninterrupted functionality become a necessary pre-requisite in ensuring continued social and economic development. While historically such inspection and maintenance operations involved visual, and thus, manual effort, with the huge growth of the infrastructure systems, especially in large metropolis like New York City, continuous manual inspection-intervention operations for each and every bridge prove to be a very costly affair. This motivation, coupled with the enormous strides in computational efficiency during the latter part of the last century, paved the way for the development of vibration based structural health monitoring techniques for automatic inspection of civil infrastructure systems.

Vibration based Structural Health Monitoring (SHM) is the task of assessing structural integrity by analyzing the structural response to dynamic excitations. SHM may be pursued both by model-based and data-based techniques. When model-based techniques are engaged, the underlying premise is the assumption of the existence of a representative numerical model of the bridge. However, this representative model will suffer from modeling errors, due, for example, to uncertainties in boundary conditions or in the definition of material parameters, as well as environmental and operational variability induced uncertainties. Considerable computational effort makes these models possible only when a limited number of model parameters are to be optimized, thereby necessitating a considerable amount of reliability in the a priori model. Thus, while model-based monitoring may allow for a clearer description of the damage, such methods will firstly prove inefficient, and secondly may converge to an incorrect identified/updated system, thereby missing a damage or raising a false alarm. In the structural health monitoring jargon, declaring the structure damaged when it is not is called false alarm or Type I error, while declaring the structure undamaged when it is instead damaged is referred to as Type II error. From the bridge engineer standpoint, Type II error may seem the most dangerous one, while flagging a false alarm may be perceived as a venial mistake, solely requiring an additional, albeit unnecessary, inspection of the bridge. However, the primary reason why the bridge managers take recourse to structural health monitoring is that of saving budget on the bridge management. If the proposed structural health monitoring approach triggers too many false alarms, it will be soon deemed useless by the bridge manager, who will likely interrupt its use. Hence, for a structural health monitoring set-up to promote safety and meet the bridge manager needs, both Type I and Type II errors must be reasonably low.

In response to the limitations posed by unreliable models in model-based techniques, data-based techniques, which rely exclusively on the data recorded from the true structure, were developed by the structural health monitoring community. In essence, these methods attempt to identify patterns characterizing the structure by analyzing the recorded vibration signatures of the structure, and hence they are also classified as pattern recognition based methods. In fact, the task of assessing damage occurrence is naturally suited to be engaged within a pattern recognition framework [13,17], as in the definition of damage is inherent a comparison between two different states of the monitored system, one of which is the reference system, often assumed to be undamaged. Pattern recognition is the discipline concerned with the assignment of classes of membership to objects, which are represented by their main traits, called patterns. When pattern recognition is applied to SHM, the classes to assign are the healthy and damaged states of the system, while the characteristics describing the main traits of the structural response time histories are called *damage sensitive features* (dsf). Recognizing the patterns that these dsf follow will provide indication on whether or not damage occurred. However, damage is not the only factor contributing to the change in the structural properties; variability of the external conditions, as well as changes in the levels of temperature or traffic loadings may induce changes in the values of the damage sensitive features.

The statistical approach to pattern recognition is then gaining momentum in the civil engineering community as a convenient approach to distinguish changes in the dsf's induced by damage from those caused by fluctuations of the operational and/or environmental conditions [12, 14, 20, 23, 29].

Within the statistical pattern recognition framework, the damage detection process is developed according to the *training* and *testing* phases. During the training stage, the patterns of the dsf's extracted from the response of the system under undamaged conditions are learnt and used to construct a probabilistic model representative of the reference conditions. During the testing step, the trained statistical model is compared against the model constructed using the features extracted from the response of the system under unknown current conditions. If the new patterns depart from those extracted in the training phase more than a predefined threshold, the structure is declared damaged.

Usually, it is assumed that the amount of damage sensitive feature observations available to estimate the training model is large. Nonetheless, this condition may sometimes be not satisfied, as in the case, for example, when the monitoring system has been just implemented and there is not abundance of data yet. A requirement common to the majority of the approaches based on statistical modeling, such as Bayesian model updating [9], principal component analysis [31] or outlier analysis [7], is that of estimating the inverse of the covariance matrix of the features population. Usually, the unbiased sample covariance matrix estimator is exploited to define such a statistic. Nonetheless, when the matrix associated with such an estimator is ill-conditioned, the estimate of its inverse becomes problematic. This is the case, for example, when one is presented with a data set of features of dimension $p$ comparable or larger than the number of observations $n$; it is known that the estimate of the covariance matrix using its unbiased sample counterpart will be not reliable in the first case ($p \simeq n$), and not even invertible in the second ($p > n$). Another problem that may rise when dealing with small training data set is the definition of the threshold value necessary to distinguish instances of damage sensitive features representative of damaged structural conditions from those representative of an undamaged state. Threshold value definition is often pursued by exploiting assumptions on the probabilistic distribution properties of the damage sensitive features, but when the available observations of the dsf's are limited, there is no robust method able to estimate their distribution.

The objective of this work was to show that, using suitable manipulations, the task of damage detection may be solved within the statistical pattern recognition framework even when the number of available observations is limited. The methods proposed in this work can be employed to gain a greater numerical stability of the covariance matrix even when the number of usable records is large. Therefore, in addition to structural damage detection, other fields could benefit from the techniques discussed in this report. We tested the performance of the proposed approaches in solving damage identification and damage location problems. In order to validate and prove the efficiency of the proposed approaches in the context of damage identification, we employed the algorithm proposed in [5], where a modified version of the Mel-Frequency Cepstral Coefficients (MFCCs) was used as damage sensitive feature, and a metric popular in the field of outlier analysis, metric which is also becoming widely used in the field of of structural health monitoring ([4, 5, 10, 19, 24, 26]), namely the Mahalanobis Squared Distance (MSD), was used to measure the distance of the testing feature vector from the training model to assess damage occurrence. In this first application, the damage sensitive feature vector was constructed by stacking one after the other the individual damage sensitive feature vectors obtained from each of the available sensors, similarly to what done also by other authors when the task is simply that of identifying the presence of damage (see for example [14]). In fact, the combination of all spatial damage information into a single feature vector is more likely to amplify the effect of changes in the damage sensitive feature, when this is extracted from the response of the system under damaged conditions, then easing the task of assessing damage occurrence. Moreover, this approach gives rise to the specific problem that we wanted to analyze, i.e. the ill-conditioning of the unbiased sample covariance matrix. As aforementioned, the proposed covariance conditioning approaches were also tested in the context of damage location applications, for which the algorithm in [5] was applied to the damage sensitive feature vector extracted from each available time history separately. In this second application, the covariance matrix was made singular by assuming availability of a number of training observations smaller than the dimension of the damage sensitive feature vector extracted from each available structural response record.

As far as this report is concerned, in Chapter 2, the three possible methods we investigated to address the problem of the inversion of an ill-conditioned covariance matrix are presented. The first alternative focuses on decreasing the dimension of the feature vectors by employing a technique based on Discrete Cosine Transform (Section 2.1). The second approach overcomes the problem of covariance matrix inversion by employing its pseudo-inverse (Section 2.2). The third option proposes the use of the shrinkage covariance matrix in place of the sample one (Section 2.3). In Chapter 3, the details of the feature extraction procedure (Section 3.1), and of the damage identification and damage location algorithms (Sections 3.2.1 and 3.2.2, respectively) employed in this study are described. Chapter 4 inspects some numerical aspects of the three proposed methods, elaborating the discussion with the aid of a numerical example. From the insights gained from this application, a combination of the three aforementioned methods was elaborated. The issues inherent the practical implementation of the proposed methods in the context of damage identification were investigated using the acceleration time histories recorded on a four story steel frame excited at the base by means of the shaking table available at the Carleton Laboratory of Columbia University. The results from this first validation are presented in Section 5.1. The performance of the proposed conditioning methods in the context of damage location was verified using simulated data from a model of a bridge deck. The results obtained in this second kind of application are discussed in Section 5.2. Finally, Chapter 6 draws some conclusions on the presented work.

## 1.1 Damage Detection Using Outlier Analysis

In analyzing the problem of pattern recognition based damage detection with a limited availability of training data, we used the Mahalanobis Squared Distance to measure the departure of the testing feature vectors from the training population. In the $p$-dimensional feature space, let us denote by $\mathbf{x}$

a point representing a feature vector obtained from the testing set, and by $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ the mean vector and covariance matrix of the trained feature population, respectively; the Mahalanobis Squared Distance of $\mathbf{x}$ from the trained model is defined as:

$$D(\mathbf{x}) = (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}). \tag{1.1}$$

The Mahalanobis Squared Distance may be thought of as the weighted squared Euclidean distance of $\mathbf{x}$ from $\boldsymbol{\mu}$, where the weighting matrix is the covariance matrix. The MSD provides a measure of how distant a feature vector is from the training population in a statistical sense, as it takes into account the correlations among variables. In other words, two feature vectors having the same MSD from the mean of the training population have the same probability of having been generated from that training population. Although when introduced, no assumption regarding the underlying probabilistic distribution of the features was explicitly mentioned [25], the MSD is particularly suited to measure the 'statistical' distance of multi-variate normally distributed features from the mean of a multi-variate normally distributed population. In such a case, the distribution of the Mahalanobis Squared Distance is regulated by well defined rules [21], so that the definition of a threshold value able to distinguish anomalies from the reference data set is straightforward. Adversely, when the features constituting the training population are not normally distributed, there is no general approach as how to define the threshold value, so that one must engage into methods, such as resampling techniques, which determine the features distribution properties empirically.

As observed in equation (1.1), the definition of the Mahalanobis Squared Distance depends on mean vector and covariance matrix of the training population. In real applications, the values of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are unknown, so that an estimate of such statistics is required. Usually, the unbiased sample counterparts of first and second statistical moments are used for such purpose. Let us assume that the population of training features is collected into a matrix $\mathbf{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_n\} \in \mathbb{R}^{p \times n}$: each column of $\mathbf{Y}$ represents an observation of a $p$-dimensional feature vector. The sample mean of the training data set is given by:

$$\hat{\boldsymbol{\mu}} = \mathbf{m} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{y}_i, \tag{1.2}$$

while the unbiased sample covariance matrix of the training set is evaluated according to

$$\hat{\boldsymbol{\Sigma}} = \mathbf{S} = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbf{m})(\mathbf{y}_i - \mathbf{m})^T. \tag{1.3}$$

# 2. Alternative Approaches to Handle Large-Scale Data Sets

As explained in Chapter 1, the aim of the present work was that of performing damage detection using the statistical pattern recognition approach when the available training features sample is constituted of a number of individuals, $n$, smaller or comparable to the dimensions of the feature vectors, $p$. In practice, when the ratio $p/n$ is less than 1, but not negligible, the numerical inversion of the sample covariance matrix becomes ill-conditioned [22]. As mentioned in Section 1, such a condition may occur, for example, when the structural health monitoring system has been just implemented and not enough training data have been collected yet. To handle this problem, three alternatives were considered: reduction of the feature dimensions, use of pseudo-inverse to operate the covariance inversion, improvement of the covariance conditioning using a shrinkage estimator.

## 2.1 Method 1: Reduction of Feature Dimensions

The first solution we explored was that of reducing the feature dimensions, so as to make $p$ smaller than the number of observations. The tool typically employed for this purpose is the Principal Component Analysis applied through the Karhunen-Loève Transformation (KLT). Nonetheless, KLT depends on the estimation of the covariance matrix of the data set to reduce, which, in our applications, is not reliable due to scarcity of data. In place of KLT, the properties of the Discrete Cosine Transform, whose basis are data independent, were exploited to perform feature dimension reduction.

The objective of Principal Component Analysis (PCA) is that of projecting the original data set into a space whose basis are parallel to the principal components of the data set itself. The principle behind PCA is that if one starts with a data set of correlated features, it should be possible to decrease the dimensionality of such features by disregarding the dimensions associated with higher degree of correlation, and retaining only the ones associated with larger variance.

A popular approach to perform PCA is that of engaging into the Karhunen-Loève Transformation (KLT). Let us assume that a set of features is collected in a matrix $\mathbf{Y} = \{\mathbf{y}_1, ..., \mathbf{y}_n\} \in \mathbb{R}^{p \times n}$, whose $i^{th}$ column represents the $i^{th}$ observation of the $p$-dimensional feature vector $\mathbf{y}_i$. The first operation required to perform KLT is transforming the original data set $\mathbf{Y}$ into one of zero mean, $\hat{\mathbf{Y}}$. Subsequently, the covariance matrix of $\hat{\mathbf{Y}}$ may be estimated, typically through its unbiased sample estimate, $\hat{\mathbf{S}}$. The next operation requires the evaluation of the eigenvalues and eigenvectors of $\hat{\mathbf{S}}$, i.e. the definition of the matrix of eigenvectors, $\mathbf{V}$, and that of eigenvalues, $\mathbf{\Lambda}$, such that:

$$\mathbf{V}^T \hat{\mathbf{S}} \mathbf{V} = \mathbf{\Lambda}, \tag{2.1}$$

where $\mathbf{\Lambda}$ is a diagonal matrix, whose main diagonal elements are the eigenvalues of $\hat{\mathbf{S}}$ sorted in descending order, $\mathbf{V}$ is the corresponding matrix of eigenvectors, whose columns are the eigenvectors of $\hat{\mathbf{S}}$ arranged such that the first column represents the eigenvector associated with the largest eigenvalue of $\hat{\mathbf{S}}$, while the last column is the eigenvector associated with the smallest eigenvalue of $\hat{\mathbf{S}}$. It is noteworthy that, under the condition that $n$ be much larger than $p$, the covariance matrix is a symmetric, positive definite and real valued matrix of order $p$; for a matrix with such properties, the spectral decomposition theorem states that it is always possible to find a set of $p$ orthonormal eigenvectors forming a basis in $\mathbb{R}^p$. Therefore, the columns of $\mathbf{V}$ form a basis in $\mathbb{R}^p$ and the matrix $\mathbf{V}$ represents a rigid rotation transformation: by pre-multiplying the data points in $\hat{\mathbf{Y}}$ by $\mathbf{V}^T$, the data set $\hat{\mathbf{Y}}$ is rotated into a space whose principal axes are aligned with the eigenvectors of $\hat{\mathbf{S}}$. The first axis of the rotated data-set is associated to the direction along which $\hat{\mathbf{Y}}$ has the largest variance, while the last dimension is associated to the direction along which $\hat{\mathbf{Y}}$ has the least variance.

By picking only the first $d$, $d \leq p$, elements of the rotated data set, it is possible to reduce the dimensions of the features in $\hat{\mathbf{Y}}$ to a smaller dimension, without losing much information.

KLT is optimal in decorrelating the features in the transformed domain, in compacting the most information using only few coefficients and in minimizing the mean-square error (MSE) between the reconstructed and original feature vector. However, in our case, an important drawback of KLT is that its basis vectors are data dependent, since the basis functions of KLT are the eigenvectors of the covariance matrix of the features population, while the objective of this work was to investigate the case where the estimation of the covariance matrix is unreliable, due to scarcity of observations. To resolve this vicious cycle, we took recourse to the Discrete Cosine Transform [2]. As first proposed in [2], the Discrete Cosine Transform (DCT) of a data sequence $y[t]$, $t = 0, ..., N-1$, is defined as

$$g_y[k] = a_k \sum_{t=0}^{N-1} y[t] \cos\left[\frac{(2t+1)k\pi}{2N}\right] \text{ for } k = 0, ..., N-1, \tag{2.2}$$

where $a_k$ is equal to $\frac{\sqrt{2}}{N}$ for $k = 0$, and to $\frac{2}{N}$ otherwise, while $g_y[k]$ is the $k^{th}$ DCT coefficient. The expression for the Inverse Discrete Cosine Transform (IDCT) is given as:

$$y[t] = a_t \sum_{k=0}^{N-1} g_y[k] \cos\left[\frac{(2t+1)k\pi}{2N}\right] \text{ for } t = 0, ..., N-1. \tag{2.3}$$

It is worth noting that the basis vectors of DCT and IDCT are the same. In [2] it is stated that the motivation for the definition of the DCT as given in Equation (2.2) is driven by the fact that its basis vectors approximate those of a class of Toeplitz matrix of the form given in Equation (2.4), where $\rho$ is a scalar defined in the range between 0 and 1:

$$\mathbf{T} = \begin{bmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{N-1} \\ \rho & 1 & \rho & \cdots & \rho^{N-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho^{N-1} & \rho^{N-2} & \rho^{N-3} & \cdots & 1 \end{bmatrix} \tag{2.4}$$

Equation (2.4) provides a good representation of the covariance matrix of weakly stationary processes. In [2], it is further stated that DCT shares with KLT the same characteristics of data decorrelation, energy compaction, and minimum MSE between reconstructed and original signal, but its basis vectors are data independent. Moreover, its computational implementation is much more efficient than that required for KLT, since Fast Fourier Transform (FFT) algorithm can be exploited to estimate DCT, while KLT requires an eigenvalue analysis. Due to its energy compaction and computational efficiency properties, DCT is largely used in digital signal processing for data compression/decompression applications; for instance, the two-dimensional DCT was employed in the international image coding standards to compress still images (JPEG).

## 2.2 Method 2: Pseudo-Inverse of the Covariance Matrix

Another approach we considered was that of acting directly on the matrix inversion operation by replacing its inverse with its pseudo-inverse. It is known that the pseudo-inverse is the operation closest to inversion applicable also to singular matrices, as is the case of the covariance matrix when $n$ is less than $p$.

In this section, the evaluation of the Moore-Penrose pseudo-inverse, $\mathrm{S}^{\dagger}$, of a square matrix S of order $p$ is briefly recalled. Pseudo-inverse computation starts by evaluating the singular value decomposition of the matrix to be inverted:

$$\mathbf{V}\mathbf{D}\mathbf{V}^T = \mathbf{S}, \tag{2.5}$$

where $\mathbf{V} \in \mathbb{R}^{p\times p}$ is a unitary matrix containing the singular vectors of $\mathbf{S}$, while $\mathbf{D}$ is a diagonal matrix, whose main diagonal elements, $d_{ii}, i = 1, ..., p$, are the singular values associated with $\mathbf{S}$, sorted in descending order. The number of non-zero singular values is equal to the rank of the matrix. If $\mathbf{S}$ is ill-conditioned, some of its singular values are very close to zero. The rows and columns of $\mathbf{D}$ associated with the $r$ singular values lower than a prescribed tolerance, $\tau$, can be then deleted from $\mathbf{D}$, resulting in a new diagonal matrix $\hat{\mathbf{D}} \in \mathbb{R}^{(p-r)\times(p-r)}$ containing only non-zero singular values. In MatLab the tolerance value is set equal to $\epsilon \cdot p \cdot \max(d_{ii})$, where $\epsilon$ is the distance from 1 to the next largest double precision number, that is $\epsilon = 2^{(-52)}$ [1], and $\max(d_{ii})$ is the largest singular value. Subsequently, the last $r$ columns of $\mathbf{V}$ are discarded, resulting in the matrix $\hat{\mathbf{V}} \in \mathbb{R}^{p\times(p-r)}$. Finally, the pseudo-inverse of $\mathbf{S}$ can be defined as

$$\mathbf{S}^{\dagger} = \hat{\mathbf{V}}\hat{\mathbf{D}}^{-1}\hat{\mathbf{V}}^T. \tag{2.6}$$

In this work, the MatLab Function `svd` was used to perform the pseudo-inverse operation on the unbiased sample covariance matrix [1].

## 2.3 Method 3: Shrinkage Covariance Matrix

Another approach to deal with an ill-conditioned covariance matrix was that of operating on the covariance matrix itself, to make it numerically stable and properly invertible. Such operation had to be performed with great care in order to forestall the eventuality of distorting the information held by the covariance matrix, which governs the values of the damage index used to identify the presence of damage. The implementation we studied was that of the shrinkage covariance matrix, which delivers an estimate of the covariance matrix shrunk between a biased but stable and invertible matrix and the ill-conditioned but unbiased sample covariance matrix estimate.

In 2003, Ledoit and Wolf [22] were the first to introduce a method of estimation of the covariance matrix, $\mathbf{\Sigma}$, able to deliver a well-conditioned, distribution-free, and more accurate estimate of $\mathbf{\Sigma}$ than that given by the sample, $\mathbf{S}$, or maximum likelihood covariance matrix, $\mathbf{S}^{ML} = \frac{n-1}{n}\mathbf{S}$, where $n$ refers, as usual, to the number of observations available to estimate $\mathbf{S}$. The problem studied by Ledoit and Wolf in [22] is that of developing a well-conditioned estimator for a large dimensional covariance matrix, i.e. for a covariance matrix constructed from $n$ observations of $p$-dimensional feature vectors, where the ratio $\frac{p}{n}$ is not negligible.

It is well known that the sample covariance matrix is the maximum-likelihood estimator of the true covariance matrix of a normally distributed population of features, so at first it could seem surprising that a better estimate of such matrix can be found. Nonetheless, as highlighted by Efron in [11], maximum likelihood solution applies only asymptotically, i.e. sample covariance matrix gives an optimal estimate of the true covariance matrix only as long as the number of available observations is large. Moreover, the maximum likelihood solution is true when the population of features is normally distributed; when the number of individuals of the population is reduced, there is no robust way to estimate their distribution, therefore, also the condition of normality could not be satisfied in presence of limited data.

Ledoit and Wolf proposed a solution inspired by the work of Stein in [30]: the *shrinkage* covariance matrix. As stated by the same authors, shrinkage is a standard process used in fields like decision theory and Bayesian statistics. The shrinkage estimate of the covariance matrix is a weighted average between the sample covariance matrix, $\mathbf{S}$, and another matrix, $\mathbf{T}$, commonly referred to as *target* matrix, whose construction requires the estimation of a number of parameters smaller than that necessary for the estimation of $\mathbf{S}$. As explained by Shäfer and Strimmer in [28], while the expected error of the unbiased sample covariance matrix with respect to the true covariance matrix, i.e. the bias of $\mathbf{S}$, is null by construction, the variance of its elements is very large, as there is a great uncertainty on their estimation. On the other hand, an estimate of the covariance matrix constructed using less parameters, as for example the variance matrix (i.e. a diagonal matrix presenting on its main diagonal the elements of the main diagonal of $\mathbf{S}$), will be characterized by a greater bias but smaller variance than that observed for $\mathbf{S}$. It makes then sense to literally shrink the sample covariance matrix towards the greater bias-less variance target matrix, by means of the so called *shrinkage coefficient*, $\lambda$:

$$\mathbf{S}^* = \lambda \mathbf{T} + (1 - \lambda)\mathbf{S}. \tag{2.7}$$

The powerful contribution given by Ledoit and Wolf has been that of deriving an explicit formula for the shrinkage coefficient. Such formula is obtained by solving an optimization problem using the loss function $L(\lambda)$:

$$L(\lambda) = \|\mathbf{S}^* - \mathbf{\Sigma}\|_F^2, \tag{2.8}$$

where $\|\mathbf{A}\|_F^2$ is the squared Frobenius norm of the square matrix $\mathbf{A}$:

$$\|\mathbf{A}\|_F^2 = \text{trace}(\mathbf{A}\mathbf{A}^T) = \sum_{i=1}^{p}\sum_{j=1}^{p} A_{i,j}^2. \tag{2.9}$$

The optimization problem is solved by finding the value of $\lambda$ that minimizes the expected value of the loss function $L(\lambda)$, $R(\lambda)$:

$$\begin{aligned} R(\lambda) &= E\{L(\lambda)\} \\ &= E\left\{\|\mathbf{S}^* - \mathbf{\Sigma}\|_F^2\right\} \\ &= \sum_{i=1}^{p}\sum_{j=1}^{p} \lambda^2 \text{Var}\{T_{i,j}\} + (1-\lambda^2)\text{Var}\{S_{i,j}\} + 2\lambda(1-\lambda)\text{Cov}\{T_{i,j}, S_{i,j}\} + \\ &+ \left[\lambda E\{T_{i,j} - S_{i,j}\} + \text{bias}\{S_{i,j}\}\right]^2 \end{aligned} \tag{2.10}$$

where Var$\{A_{i,j}\}$ and Cov$\{A_{i,j}\}$ refer to the variance and covariance of the matrix element $A_{i,j}$, respectively, while bias$\{S_{i,j}\}$ is the bias on the $(i,j)^{th}$ element of the unbiased sample covariance matrix, which is shown for the sake of completeness, but which is known to be equal to zero. Taking the derivative of $R(\lambda)$ with respect to $\lambda$ and setting the result equal to zero, the value of the shrinkage coefficient yields

$$\lambda = \frac{\sum_{i=1}^{p}\sum_{j=1}^{p}\left[\mathrm{Var}\{S_{ij}\} - \mathrm{Cov}\{S_{ij}, T_{ij}\}\right]}{\sum_{i=1}^{p}\sum_{j=1}^{p} E\{(S_{ij} - T_{ij})^2\}}. \tag{2.11}$$

In Equation (2.11), the fact that bias$\{S_{i,j}\}$ is null has been taken into account. Since the second derivative with respect to $\lambda$ of $R(\lambda)$ is the mean square error between the target and sample covariance matrices, $E\{(S_{ij} - T_{ij})^2\}$, the function $R(\lambda)$ is concave and its critical point is a minimum: the value of $\lambda$ given in Equation (2.11) minimizes the mean square error between the shrinkage and the true covariance matrices. As observed in [28], all the quantities given in Equation (2.11) are asymptotical estimates of the corresponding statistics, which can be converted to their unbiased sample counterparts without great loss of accuracy.

By examining Equation (2.11), it is observed that the value of the shrinkage coefficient is directly proportional to the variance of the sample covariance matrix, so that if the uncertainty on the sample covariance matrix estimate increases, the value of $\lambda$ increases as well, putting more weight on the target matrix. The second term of the numerator, i.e. the covariance between target and sample covariance matrices, takes into account the fact that target and covariance matrix could be estimated from the same data-set, and thus it filters out the bias on the shrinkage estimate. Adversely, the shrinkage coefficient is inversely proportional to the mean square error between the sample and the target covariance matrices and this prevents the shrinkage covariance matrix from being largely affected by the wrong choice of the target matrix. Therefore, if one selects a target matrix whose structure is too far from that of the true covariance matrix, the value of $\lambda$ will be close to zero, and the shrinkage covariance matrix will basically be the sample covariance matrix. This observation clarifies how the overall shrinkage estimation can benefit from the right choice of the target.

# 3. Feature Extraction and Damage Detection Algorithm

## 3.1 Damage Sensitive Feature

Damage sensitive features should be able, ideally, to detect any change in the structural properties due to damage, irrespective of the external conditions. In this work, a modified version of Mel-Frequency Cepstral Coefficients (MFCC), namely the Warped-Frequency Cepstral Coefficients (WFCC), was used as damage sensitive feature [4,5]. As for the original MFCC, each step involved in the extraction of the WFCC is represented by a digital signal process, i.e. the final product of the feature extraction is really a compact version of information obtained from the structural response, no extrapolation of any sort is required to the user.

Figure 6.1 shows the steps involved in the cepstral feature derivation. The *framing* step is often employed when dealing with non-stationary signals, i.e. when handling time series whose statistical properties do change over time. In fact, framing is the operation of segmenting the original signal into pieces short enough to be considered stationary. It was proposed to employ 240 points long frames, which correspond to 0.6-2.4 seconds of recording length for the sampling frequency ordinarily used in civil engineering applications (100-400 Hz). The *windowing* operation is then performed by applying a non-rectangular window to each frame to remove edge effects. A popular choice is Hamming window, which was also the window selected in this work. The successive step requires the evaluation of the power spectrum of each frame. The consequent *frequency warping* operation is performed to compact the spectrum information into a reduced number of points, to smooth the spectrum and to emphasize meaningful frequencies by collecting the spectral components into $n_b$ frequency bins. When constructing MFCC, such bins are equally spaced into the so called Melody-Frequency Scale, which is optimal in representing sound perception by the human ear. Employing such scale when dealing with the structural response of the system is not appropriate, mainly because the Mel-Frequency scale was coined for a different scope, and secondly because it is set up for a frequency range well over the ranges normally considered in civil engineering applications. For these reasons, in [5], we proposed to warp the linear frequency scale into a scale, referred simply as *warped frequency scale*. The two scales are related through the following expression:

$$f_{warping} = f_c \log_2 \left( 1 + \frac{f}{f_c} \right) \tag{3.1}$$

where $f_{warping}$ is the value of the frequency in the warped scale associated to the value $f$ of the linear frequency scale, while $f_c$ is a user-defined cutoff frequency representing the boundary of the major power content in the system spectrum. The relation between $f_{warping}$ and $f$ is linear up to $f_c$, and becomes logarithmic after such value. Frequency warping is obtained by grouping together the spectrum values into $n_b$ critical bands, and weighting each cluster by a triangular filter. The series of $n_b$ triangular filters has centers equally spaced on the $f_{warping}$ scale. In this work, the value of $n_b$ was set equal to the integer part of $3\ln(f_s)$, as suggested in [15], where $f_s$ is the value of the sampling rate in Hz. Figure (6.2.a) represents the spectrum obtained averaging all time histories recorded for the undamaged scenario 1 described in Chapter 4, from which the selection of $f_c$ is made available, while Figure (6.2.b) shows the corresponding triangular filters. The successive step consists in the evaluation of the logarithm of the warped spectrum. The logarithm operation is instrumental to equally emphasize both peaks and valleys of the response spectrum.

The operations applied to the response signal up to this point deliver the log-warped spectral frames, which can be represented by means of as many $n_b$-dimensional vectors as the number of frames in which the original signal has been framed. The elements of such vectors are highly correlated. Therefore, the last operation, consisting of applying a $d$-point IDCT to each log-warped spectra has the objective of decorrelating the vector dimensions. Here, the properties of IDCT discussed in section 2.1 are exploited to further reduce the vector dimensions. The value of the first cepstral coefficient is proportional to the logarithm of the energy of the corresponding time history frame, and it is then dependent on the characteristics of the instrument used to measure the structural response. For this reason, the first cepstral coefficient is often discarded [8, 18], as it was also done in this work.

Let us now clarify the criterion we proposed to select the number of IDCT points, $d$. Let us denote as $\ell$ the number of frames in which a time history is segmented. At the end of the feature extraction process, $\ell$ $(d-1)$-dimensional vectors are extracted from a single response history: the average of such vectors is evaluated leading to a single vector consisting of $(d-1)$ elements. Let us now assume that $n_s$ sensors are available. From each of the $n_s$ response time histories, a $(d-1)$-point long feature vector is extracted and stacked with the other $(n_s-1)$ vectors, so that each data set is represented by a feature vector $\mathbf{y} \in \mathbb{R}^{n_s \cdot (d-1) \times 1}$. For the sake of notation brevity and consistency with the previous treatments, let us denote as $p$ the dimension of a feature vector extracted from a data set, i.e. $p$ is equal to $n_s \cdot (d-1)$. Let us refer to $n_{tr}$ as the number of data sets available for training. At the end of the feature extraction procedure, if the technique described in section 2.1 is engaged, the value of $d$ as the number of IDCT points is given as:

$$d = \left\lceil \frac{0.8 n_{tr}}{n_s} + 1 \right\rceil , \tag{3.2}$$

where the symbol $\lceil a \rceil$ indicates the closest integer greater to or equal to $a$.

## 3.2 Damage Detection Algorithm

In this study, we investigated the use of two algorithms to perform damage detection. The objective of the first algorithm was that of assessing whether or not the structure is damaged. We refer to this task as *damage identification*. The scope of the second approach was that of assessing the presence of damage and its location. We refer to this task as *damage location*.

### 3.2.1 Damage Identification

In any data-based damage detection algorithm, an important step is represented by the definition of the threshold value for the dsf, value that can be defined only upon the observed data when the number of available feature observations to estimate the distribution statistics is too low. In fact, when the number of feature vector observations available for training is large, the feature data set may be considered normally distributed. The distribution of the MSD of a $p$-variate point $\mathbf{x}$, when $\mathbf{x}$ is not involved in the estimation of the sample mean and the sample covariance matrix, follows a scaled F-distribution with degrees of freedom $p$ and $n_{tr} - p$, where $n_{tr}$ is the number of features observations used to evaluate the sample statistics' estimators. This property can then be used to define confidence intervals upon which the threshold is then defined. When the number of feature vectors to estimate the training statistics is too low, no assumption can be made on the feature distribution, so that the confidence intervals cannot be set up using the aforementioned property of the MSD values of normally distributed features, and the threshold value must be then defined empirically.

For the damage identification application, when using the methods described in sections 2.1 and 2.2, we proposed to construct the training model evaluating the sample mean $\mathbf{m}_{tr}$:

$$\mathbf{m}_{tr} = \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \mathbf{y}^{(i)} \tag{3.3}$$

and the unbiased sample covariance matrix $\mathbf{S}_{tr}$:

$$\mathbf{S}_{tr} = \frac{1}{n_{tr} - 1} \sum_{i=1}^{n_{tr}} \left(\mathbf{y}^{(i)} - \mathbf{m}_{tr}\right) \left(\mathbf{y}^{(i)} - \mathbf{m}_{tr}\right)^T \tag{3.4}$$

where $\mathbf{y}^{(i)}$ represents the realization of the feature vector extracted from the $i^{th}$ training data set. If the technique described in section 2.3 is engaged, the shrinkage estimate of the covariance matrix, $\mathbf{S}^*$, is evaluated in place of $\mathbf{S}$.

Once the model is determined, the threshold to distinguish a dsf representative of a healthy state from a damaged one can then be defined as explained in the following paragraphs. As per the introductory discussion of this section, the paucity of data prevents the features to be assumed normally distributed and a resampling technique was proposed to set the boundary between damaged and undamaged conditions. In order to define the threshold, we used the exclusive Mahalanobis Squared Distance of the $n_{tr}$ training feature vectors:

$$D(\mathbf{x}^{(i)}) = (\mathbf{x}^{(i)} - \mathbf{m}^{(i)})^T \mathbf{S}^{(i)-1} (\mathbf{x}^{(i)} - \mathbf{m}^{(i)}) \text{ for } i = 1, ..., n_{tr}, \tag{3.5}$$

where $\mathbf{m}^{(i)}$ and $\mathbf{S}^{(i)}$ are the sample mean vector and sample covariance matrix of the training sample set, from where $\mathbf{x}^{(i)}$ has not been included. The inverse of the unbiased sample covariance matrix is replaced by the pseudo inverse of $\mathbf{S}^{(i)}$, when the second approach is used to address the covariance conditioning issue, or by the inverse of the shrinkage covariance matrix if the approach discussed in Section 2.3 is employed. The value exceeded by $\alpha\%$ of the $n_{tr}$ instances of exclusive MSD is picked as threshold value, $\gamma$, where $\alpha$ is a value ranging from 1 to 5. In this work the 5% percentile of $D(\mathbf{x}^{(i)})$ was used.

Finally, let us denote as $n_{te}$ the number of data sets available for testing. From each available data set, a $p$-dimensional feature vector is extracted. If feature dimension reduction is required, a $d$-point IDCT is employed as last step of the WFCC features extraction, where $d$ is selected according to the criterion given in Equation (3.2). If more than one data set is available, the mean of the available testing vectors, $\mathbf{m}_{te}$, is computed. If $n_{te}$ is equal to 1, as the case for short-time applications, $\mathbf{m}_{te}$ is simply the single $p$-dimensional vector extracted from the only available data set. The structure can be then declared damaged if

$$D(\mathbf{m}_{te}) = (\mathbf{m}_{te} - \mathbf{m}_{tr})^T \mathbf{S}_{tr}^{-1} (\mathbf{m}_{te} - \mathbf{m}_{tr}) > \gamma. \tag{3.6}$$

where $\mathbf{S}_{tr}^{-1}$ is replaced either by the pseudo-inverse of $\mathbf{S}_{tr}$, $\mathbf{S}_{tr}^{\dagger}$, or by the inverse of the shrinkage estimate of the covariance matrix, $(\mathbf{S}_{tr}^{*})^{-1}$.

### 3.2.2 Damage Location

The algorithm proposed to perform damage location is very similar to that used to perform damage identification, but each recorded response time history is used separately to solve a damage identification problem. This algorithm is especially suited for cases where the number of available observations is quite small.

Let us again denote as $n_s$ the number of available recorded time histories. During training, the $p$ dimensional WFCC vector is extracted from each of the $n_s$ time histories following exactly the same procedure described in Section 3.1. For damage location applications, the resulting feature vectors are not stacked one after the other. Hence, for damage location applications, $p$ is simply equal to $d - 1$, where $d$ is evaluated according to Equation (3.2), with $n_s = 1$, if feature reduction is required. Assuming $n_{tr}$ data-sets are available for training, for each one of the $n_s$ $p$-dimensional WFCC feature vectors, the exclusive Mahalanobis Squared Distance is evaluated as defined in Equation (3.5), where again if Method 2 is employed, the inverse of the sample covariance matrix is replaced by its pseudo-inverse, and, if Method 3 is employed, the inverse of the sample covariance matrix is replaced by the inverse of the shrinkage covariance matrix. The value exceeded by only $\alpha\%$ of the $n_s$ sets of exclusive MSD values is then evaluated. Also for this application $\alpha$ was taken equal to 5. Let us denote as $D^{(s)}_{\alpha\%}$ the $\alpha\%$ percentile identified from the set of exclusive Mahalanobis Squared Distance values estimated from the $s^{th}$ sensor. These values indicate that the probability for the MSD of a feature vector extracted from the response of the healthy system at sensor $s$ to be smaller than $D^{(s)}_{\alpha\%}$ is equal to $\alpha\%$.

For reporting the results of damage location, we investigated an approach somewhat similar to the p-values in standard hypothesis testing problems. The p-value is the probability of obtaining at least the observed sample results when the null hypothesis is true. In the application under analysis, the null hypothesis is that the area around sensor $s$ is under reference, usually undamaged, conditions. In the training phase of the statistical pattern recognition approach is implicitly assumed the structure be subject to a variety of analysis to ensure its performance is aligned with predefined criteria of safety and functionality; the structural conditions characterizing the system in this phase are often referred to as *reference* conditions. The farther the p-value from the confidence interval value, the more confident we are in accepting the null hypothesis. Let us denote as $D_{test}^{(s)}$ the value of the MSD of the testing feature extracted from the response recorded at the $s^{th}$ sensor. The p-value associated to $D_{test}^{(s)}$ is computed as

$$p_{value}^{(s)} = 1 - P(D < D_{test}^{(s)}), \tag{3.7}$$

where $P(D < D_{test}^{(s)})$ is the probability that the MSD of the testing feature vector from the training model be smaller than the observed testing value $D_{test}^{(s)}$. When many observations are available, Equation (3.7) specializes into

$$p_{value}^{(s)} = 1 - \tilde{F}_{n_{tr}-p,p}(D_{test}^{(s)}) \tag{3.8}$$

where $\tilde{F}_{n-p,p}(D_{test}^{(s)})$ is the cumulative distribution function of a scaled F-distribution of degrees of freedom $p$ and $n_{tr} - p$ evaluated at $D_{test}^{(s)}$. In fact, we already mentioned that when $n_{tr}$ is large, the MSD of the testing feature from the training model is distributed according to a scaled F-distribution with degrees of freedom $p$ and $n_{tr} - p$. Since we could not exploit this result for the application of interest, we approximated $P(D < D_{test}^{(s)})$ as

$$P(D < D_{test}^{(s)}) \simeq \frac{\alpha}{100} \frac{D_{test}^{(s)}}{D_{\alpha\%}^{(s)}}. \tag{3.9}$$

Therefore, the p-value for the testing feature extracted from the response time history recorded at the $s$ sensor is approximated as:

$$p_{value}^{(s)} = 1 - \frac{\alpha}{100} \frac{D_{test}^{(s)}}{D_{\alpha\%}^{(s)}}. \tag{3.10}$$

The farther the $p_{value}^{(s)}$ is from $1 - \alpha/100$, the more confident we are in declaring the zone close to sensor $s$ under healthy conditions. Using p-values in place of the threshold comparison used for damage identification allowed us to better understand the degree of confidence on the damage location result.

# 4. Numerical Considerations

## 4.1 Physical Interpretation of Mahalanobis Squared Distance

In the previous sections, it has been shown how the Mahalanobis Squared Distance may be used as an index to assess damage occurrence. In Section 1.1 it has also been stated that MSD can be thought of as a weighted Euclidean distance of a vector $\mathbf{x}$ from the mean of the training distribution $\mathbf{m}_{tr}$, where the weights take into account the correlation and variance properties of the training vectors. In order to better understand the performance of the three proposed covariance conditioning methods, a further analysis of the Mahalanobis Squared Distance had to be elaborated. In the following, the steps of this analysis are discussed.

The definition of MSD, in terms of the sample estimates of the training statistics, is rewritten in Equation (4.1) to ease the ensuing discussion:

$$D(\mathbf{x}) = (\mathbf{x} - \mathbf{m}_{tr})^T \mathbf{S}_{tr}^{-1} (\mathbf{x} - \mathbf{m}_{tr}), \tag{4.1}$$

where **x** is a $p$-dimensional testing feature vector, $\mathbf{m}_{tr}$ is the $p$-dimensional sample mean vector of the training population and $\mathrm{S}_{tr}$ is the unbiased sample covariance matrix of order $p$ of the training population.

Let us denote by $\mathbf{V}$ the orthonormal matrix of order $p$ of eigenvectors of $\mathbf{S}_{tr}$, and by $\mathbf{\Lambda}$ the diagonal matrix of order $p$ of eigenvalues of $\mathbf{S}_{tr}$, such that:

$$\mathbf{S}_{tr} = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T. \tag{4.2}$$

As already noted in section 2.1, the columns of $\mathbf{V}$ form a basis in $\mathbb{R}^p$ and by multiplying a vector in $\mathbb{R}^p$ by $\mathbf{V}$ a rigid rotation of the vector into a new coordinate system is produced.

Making use of Equation (4.2), Equation (4.1) may be rewritten as follows:

$$D(\mathbf{x}) = \left[\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{V}^T(\mathbf{x}-\mathbf{m}_{tr})\right]^T \left[\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{V}^T(\mathbf{x}-\mathbf{m}_{tr})\right]. \tag{4.3}$$

From Equation (4.3) it is possible to interpret how the Mahalanobis Squared Distance measures the departure of a testing vector $\mathbf{x}$ from the mean of the training population, $\mathbf{m}_{tr}$. First, the testing vector is projected into a new system of reference, which has origin at the point corresponding to the mean vector of the training population and axes directed along the principal components of the training population covariance matrix. Let us denote with $\mathbf{y}$ the point in the $p$-dimensional space obtained by translating $\mathbf{x}$ by $\mathbf{m}_{tr}$ and rotating it through $\mathbf{V}$:

$$\mathbf{y} = \mathbf{V}^T(\mathbf{x}-\mathbf{m}_{tr}); \tag{4.4}$$

the second operation involved in the evaluation of the MSD of $\mathbf{x}$ from $\mathbf{m}_{tr}$ consists of scaling the coordinates of $\mathbf{y}$ by the inverse of the square root of the matrix of eigenvalues, $\mathbf{\Lambda}^{-\frac{1}{2}}$, leading to a transformed point $\mathbf{z}$:

$$\mathbf{z} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{y} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{V}^T(\mathbf{x}-\mathbf{m}_{tr}). \tag{4.5}$$

It is noteworthy to remind that the covariance matrix eigenvalues are assumed sorted in ascending order, i.e. eigenvalues are sorted in order of data variance. Hence, the first (last) eigenvalue is the largest (smallest) and is associated to the eigenvector representing the direction of maximum (minimum) dispersion of the data. Scaling $\mathbf{y}$ by $\mathbf{\Lambda}^{-\frac{1}{2}}$ normalizes $\mathbf{y}$ with respect to the standard deviation of the training feature vectors coordinates. For this reason, here onwards, the scalars obtained as the square root of the inverse of the covariance eigenvalues will be referred to as standard deviation normalizing factors. In order to finally obtain the measure of the departure of $\mathbf{x}$

from $\mathbf{m}_{tr}$ in terms of Mahalanobis Squared Distance, the squared norm of $\mathbf{z}$ must be evaluated:

$$D(\mathbf{x}) = \mathbf{z}^T\mathbf{z} = \left[\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{V}^T(\mathbf{x}-\mathbf{m}_{tr})\right]^T \left[\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{V}^T(\mathbf{x}-\mathbf{m}_{tr})\right]. \tag{4.6}$$

Evaluating the MSD of $\mathbf{x}$ from $\mathbf{m}_{tr}$ is then equivalent to evaluate the Euclidean squared norm of the testing vector in a system of reference where the training vector variables are uncorrelated and normalized with respect to the training variables standard deviation. In order to further clarify this point, Figures (6.3) illustrate the effects of the projection and normalization, on a set of 1000 realizations of bi-variate Gaussian random vectors, of mean $\boldsymbol{\mu} = [1,1]^T$, standard deviation associated to the $x$ variable equal to $\sigma_x = \sqrt{10}$, standard deviation of the $y$ variable equal to $\sigma_y = \sqrt{2}$ and coefficient of correlation between the two variables equal to 0.67. The effects of said transformations are tracked on a point P, represented by a red cross in Figures (6.3).

Figure (6.3a) shows the original data set: the data set are arranged in an elliptical cluster whose major axis forms a non-zero angle with respect to the Oxy system of reference abscissa, indicating correlation between the two variables. In Figure (6.3b), it is shown the result of the projection of the data points into the system of reference with origin in the point represented by the cluster's sample mean vector, $\mathbf{m}_{tr}$, and axis parallel to the direction traced by the eigenvectors of the cluster's sample covariance matrix, $\mathbf{S}_{tr}$. It is observed that in the new frame of reference, the data are maximally dispersed along the axis labeled as $\bar{x}_1$, and minimally dispersed along its orthogonal axis, labeled as $\bar{x}_2$. In the new system of reference, the vectors variables are uncorrelated. Nonetheless, the cluster is still an ellipse, indicating different values of data dispersion along the two principal directions. Figure (6.3c) displays the effect of standard deviation normalization; by multiplying the data set by $\boldsymbol{\Lambda}^{-\frac{1}{2}}$, the resulting sample data points get equally dispersed along each direction, as in fact the data cluster is now represented by a circle rather than an ellipse. The set of operations performed on the data is known as affine transformation and permits to compare the data on a common scale of standard deviation. The MSD of the points in Figure (6.3a) from their mean is equivalent to Euclidean squared norm of the points in Figure (6.3c).

It is clear that the standard deviation normalization is a crucial step in MSD evaluation and it is basically what distinguishes the Mahalanobis Squared Distance metric from the Euclidean squared norm. It is then important to investigate how the standard deviation normalizing factors change when using the covariance matrix estimators proposed in Chapter 2. In order to perform such analysis, the four proposed methods were tested in the context of damage identification. In such an analysis, the eigenvalues of the different covariance estimates were compared, assuming the different estimators be rotation equivalent, i.e. assuming the different covariance estimators have the same eigenvectors of the true covariance matrix, but different eigenvalues. The shrinkage estimate of the covariance matrix is rotation invariant by construction [22], while the assumption of rotation invariance for the unbiased sample covariance matrix of a small data set can be speculated to be reasonable if one thinks of the physical interpretation of the covariance matrix eigenvalues and eigenvectors. The covariance matrix eigenvalues give a measure of how much each feature vector variable is dispersed, then we expect this measure to be very sensitive to the number of available feature vector observations. Adversely, the covariance matrix eigenvectors indicate the direction of dispersion of the various feature components, and such directions are not expected to vary by changing the number of feature observations. For example, if a smaller sample of data points was generated from the bi-variate normal distribution used to plot Figures (6.3), we would have expected the majority of the data points to lie along direction $\bar{x}_1$, and only a few along $\bar{x}_2$. Thus, the direction of maximum and minimum data dispersion would have remained unchanged; on the contrary, due to the smaller amount of points available along each direction, the ‘amount of dispersion’, i.e. the variance, of each dimension would have changed with respect to the values observed for the larger sample, likely increasing along direction $\bar{x}_1$ and decreasing along direction $\bar{x}_2$.

## 4.2 Numerical Example

The acceleration response time histories of a 10-story shear-type system, modeled according to the common mass-spring-viscous damper chain, were considered. The nodes were numbered in ascending order, so that the node closest to the fixed end was labeled as 1. The inter-story stiffness between the $(i-1)^{th}$ and $i^{th}$ nodes was denoted as $k_i$. The energy dissipation properties of the system were modeled through the Rayleigh damping mechanism. The baseline system was assumed to be characterized by inter-story stiffness, $k_i^0 = 6 \times 10^6$ N/m, and mass, $m_i^0 = 2 \times 10^3$ Kg, for $i = 1, ..., 10$. Another undamaged scenario was considered, by reducing the stiffness of the first 5 inter-stories by 2% of their baseline values. This second undamaged condition was considered in order to account for those systematic changes to the structural properties imposed by fluctuations of the environmental conditions. For example, the second undamaged scenario could represent a condition where only the first five storeys of the frame are subject to a temperature increase due, for instance, to the relative position of sun and structure at a particular time of the day. Additionally, two damaged conditions were modeled by decreasing to 85% of their original values $k_3$ (Damage Scenario 1) and $k_5$ (Damage Scenario 2).

The acceleration response to a white Gaussian noise input was simulated. The input time histories, of mean 0 and standard deviation 1, were 2 minutes long, sampled at 0.01 seconds and applied at each DOF. To simulate the effects of measurement noise, 10% RMS white Gaussian noise was added to each record of the simulated response. To reproduce operational variability, for each data set realization, the values of the structural properties of the four considered scenarios were perturbed by a small amount, randomly picked from a set of values, uniformly distributed between -0.01 and 0.01. For the purpose of this analysis, only the response recorded at DOFs 2, 5 and 8 were considered, i.e. the number of available instruments, $n_s$, was assumed to be equal to 3. The simulated response data were standardized and normalized prior of being used for feature extraction:

$$x_{normalized}[n] = \frac{x[n] - \bar{x}}{\sigma_x}, \tag{4.7}$$

where $x[n]$ is the original signal, $\bar{x}$ its mean and $\sigma_x$ its standard deviation. The normalization procedure was performed on each frame. The number of cepstral coefficients extracted from each frame was selected based on the criterion given in [15], i.e. selecting the closest integer greater or equal to $3\ln(f_s)$. Since the sampling frequency was 100 Hz, the selected number of cepstral coefficients was set to 14. Once discarded the first coefficient of each cepstral feature vector, and appended the resulting feature vector to the other $n_s - 1$ vectors, the feature dimension, $p$, was equal to 39.

Two training data sets, $Y^L$ and $Y^S$ were then constructed. $Y^L$ consisted of 3900 training feature observations, so that the ratio between the feature dimensions, $p$, and the number of observations, $n_{tr}$, was equal to 0.01; $Y^S$ was instead constructed so that the number of training feature observations be equal to that of the feature dimensions, so that $\frac{p}{n_{tr}} = 1$. Here onwards, $\mathrm{S}_{tr}^L$ will be used to denote the sample covariance matrix of the larger training data set, $Y^L$, while $\mathrm{S}_{tr}^S$ will refer to the sample covariance matrix of the smaller training sample, $Y^S$. The form of the true covariance matrix of the population of feature coefficients was unknown, and the results obtained using $Y^L$ were then treated as a benchmark.

The testing data set consisted of 50 realizations for each of the four conditions (two undamaged and two damaged conditions), for a total of 200 testing data sets. Each data set was used individually to perform a short-term kind of damage identification assessment. Thus, two hundred tests were performed: 100 using response realizations of the healthy structure, and 100 using those of the damaged system.

## 4.3 Results obtained using the sample covariance matrix $\mathrm{S}_{tr}^S$

Figures (6.4) compare the standard deviation normalizing factors obtained from $\mathrm{S}_{tr}^L$ with those obtained using the different covariance estimators discussed in the previous sections. Figure (6.4a) presents the comparison between the variance normalizing factors obtained from the sample covariance matrix of the training data set $Y^L$ with those obtained from $\mathrm{S}_{tr}^S$, which is simply the covariance matrix of the small training data set without any of the proposed conditioning. The plot shows only the real part of such factors, as indeed $\mathbf{S}_{tr}^{S}$ is non-positive definite. Sample covariance matrices of small samples can occur to be non-positive definite as the scarcity of information on the correlation between some of the feature components can lead to an improper definition of the component statistics [3], as is the case for the data set considered in this example. Since the standard deviation normalizing factors are obtained as the square root of the inverse of the covariance matrix eigenvalues, the factors obtained from the negative eigenvalues are complex numbers. Furthermore, from Figure (6.4a) it is clearly apparent that the standard deviation normalizing factors are largely magnified with respect to the reference ones. In order to better appreciate such characteristic, in Figure (6.4a) the values of the antepenultimate and penultimate standard deviation normalizing factors are plotted as scaled by a factor of 3, while the last factor value is scaled by a factor of $10^6$. The difference in magnitude of the normalizing factors is again due to the scarcity of available feature vectors observations. The last components of the feature vectors are by definition the ones characterized by low variance values, i.e. the ones associated with the low covariance eigenvalues. When only few data are available, this trend is intensified and it is likely that very few data are observed along the direction of low data dispersion; the variance and consequently the eigenvalue associated to that direction is then very low and numerically equal to zero. The reciprocal of the very low eigenvalues results in very large normalizing factors. The combination of the two effects, namely the presence of both negative and very low eigenvalues, produces unreasonable values of the MSD of the testing points, motivating the necessity of seeking alternatives to the unbiased sample covariance matrix when dealing with small sample training sets.

## 4.4 Results obtained using Method 2

Figure (6.4b) shows the comparison between the reference normalizing factors obtained from $\mathbf{S}_{tr}^{L}$ and those obtained using Method 2, which employs the pseudo-inverse of the covariance matrix $\mathbf{S}_{tr}^{S}$, $(\mathbf{S}_{tr}^{S})^{\dagger}$, when evaluating the MSD of the testing points from the training mean vector. It is worth reminding that the factors plotted in Figure (6.4b) are the square root of the eigenvalues of matrix $(\mathbf{S}_{tr}^{S})^{\dagger}$. It is noteworthy that, although for this case a numerically stable representation of the inverse of the covariance matrix is used, still, the variance properties of the feature components are described by means of the ill-conditioned unbiased sample covariance matrix. As mentioned before, the variance of the last components is very low, producing the large values of the normalizing factors weighting the last feature components in Figure (6.4b). In particular, the values of the normalizing factors weighting the last two components are within one and two orders of magnitudes larger than all the other factors, in addition to being two orders of magnitude larger than the values of the reference factors. This causes the value of the resulting Mahalanobis Squared Distance to be totally influenced by only the two last components of the feature vector, as well as to be considerably larger than the MSD obtained using the statistics estimated from the large training data set, $Y^{L}$. It is important to mention that here the term *components* refers to the elements of the feature vector projected into the principal component space, i.e. in the system of reference centered on the training mean vector and with basis represented by the eigenvectors of the training covariance matrix. The effects of such conditions on the MSD values are presented in Figure (6.5b), where the magnitudes of the Mahalanobis Squared Distance estimated using Method 2 are shown. The first 100 tests are performed on the healthy structure, while the tests from 101 to 150 are performed on the structure under the damaged condition D1 and tests from 151 to 200 on the system under damaged condition D2. The dashed line represents the computed threshold: any value of the MSD below the threshold is considered representative of an undamaged scenario, while an instance with an MSD value greater than the threshold triggers the declaration of damage occurrence. The two orders of magnitude difference between the factors obtained from the large and small samples is reflected in the MSD values, as it can be readily observed by comparing Figure (6.5a) with Figure (6.5b), which show the MSD values of the testing features from the large and small training data sets, respectively. Figure (6.5a) shows the results for the case where 3900 training observations are available, and the unbiased sample covariance matrix is used in order to estimate the departure of the testing features from the training mean vector.

Further inspecting the results obtained by use of Method 2 on the small sample data set, it is recognized that only 2 out of the 100 tests performed on data simulating undamaged scenarios incorrectly identify the structure as damaged, i.e. Type I or false alarm error is only equal to 2%; nonetheless, 39 out of the 100 tests performed on data representative of damaged scenarios incorrectly identify the structure as undamaged, i.e. Type II or false acceptance error is equal to 39%. In particular, 79% of Type II error is due to the missed recognition of state D2 as damaged. This is because the last two feature components of the majority of the feature vectors representing the second damage condition do not depart from the training ones enough to raise the MSD value over the threshold. This is empirically observed by plotting the components of the testing feature vectors transformed as in Equation (4.5), where the matrix of the square root of the matrix of eigenvalues of $(\mathbf{S}_{tr}^{S})^{\dagger}$ is used in place of $\mathbf{\Lambda}^{-\frac{1}{2}}$. Figure (6.6) shows said plots for the last six pairs of components obtained by affine transformation of all 200 testing feature vectors using Method 2. The components pairs are plotted on the same scale to reiterate that only the last two components do actually influence the value of MSD. This means that the damage identification process is solely based on the departure of the last two components from the training ones. This observation may be generalized stating that if there are only few components influencing the value of the MSD, the assignment of damage detection is devolved solely upon the sensitivity to damage of such components, thereby reducing the chance of detecting damage.

However, it is important to stress out that the scaling factors defined using the pseudo-inverse propose a conceptually ideal way of scaling the feature components. In fact, the first components are weighted by very low values, then not participating to the value of the MSD. On the contrary, the last components are weighted by very high values, then largely contributing to the value of the MSD. It is worth to remember that the MSD is used as a measure of discordance with the training data: we would like the MSD to be large for data which are not realized from the training population, known as an *outlier*, while we want the MSD of a feature extracted from any of the healthy conditions, i.e. a realization of the training sample, also known as *inlier*, to be ideally zero. The major differences between inliers and outliers will be observed in the last feature components, then weighting more such components would amplify any difference from the training samples. On the other hand, under-weighting the first components will aid to decrease the effect of a wrong variance estimation, often occurring for small sample data sets. In conclusion, the scaling approach proposed by the pseudo-inverse would be ideal if applied on a stable covariance matrix.

## 4.5 Results obtained using Method 1

Figure (6.4c) compares the standard deviation normalizing factors obtained from $\mathbf{S}_{tr}^{L}$ with those obtained using Method 1, which employs features of lower dimensions to form a full rank sample covariance matrix. In particular, the factors shown in Figure (6.4c) refer to the condition where an 11-points IDCT is used to complete the feature extraction, leading to a 30-dimensional feature vector. Also in this case, it is observed that the factors weighting the last components are over-magnified with respect to the reference values. Nevertheless, when employing Method 1, the standard deviation normalizing factors are only one order of magnitude greater than the reference ones. In fact, by using an 11-points in place of a 14-points IDCT to extract the feature vectors allows us to get rid of feature variables characterized by very low variance values. This results in smaller covariance eigenvalues compared to those obtained through Method 2. However, the paucity of available data still causes an overestimate of the variance of some components, inducing the values of the factors weighting the last feature components to be larger than the reference ones. The results of the damage detection exercise performed using Method 1 are given in Figure (6.5c). Once again it can be seen that the Type I error is very low, namely equal to 2%, while Type II error is very large and equal to 40%. As for the previous case, also in this instance the presence of dominant normalizing factors reduces the influence of many components sensitive to damage. However, for the case under analysis, the major part of missed damaged instances is caused by an unduly large value of the threshold. The reason of such phenomenon may be grasped by observing that the proportion between the factors weighting the last and penultimate feature components is distorted, if compared to that of the large sample data set. The distortion is symptomatic of a feature components dispersion that departs from that of the benchmark sample. It can be speculated that the majority of the components of the training features extracted from the small data set be dispersed as those of the large one, while only a few depart from it. It must be remembered that the threshold value is evaluated using the exclusive Mahalanobis Squared Distance of the training data themselves: the exclusive MSD of training features that are dispersed in a different fashion than the majority of the other training features is large. It is enough that 5% of the training features be abnormally dispersed to excessively raise the value of the threshold, as for the case being currently considered. The same phenomenon is partly present also when the pseudo-inverse approach is applied; however, for that case, the effect of the predominance of the last components to the value of the MSD prevails in causing high Type II error.

## 4.6 Results obtained using Method 3

The last step of the feature extraction process employed in this work aims at decorrelating the feature coefficients. Therefore, the covariance matrix of the training feature vectors is expected to be diagonal. For this reason, we considered the diagonal variance matrix as target for the construction of the shrinkage covariance estimator. Here, we refer to as variance matrix the diagonal matrix whose main diagonal entries are equal to the main diagonal elements of the sample covariance matrix.

Figure (6.4d) plots the standard deviation normalizing factors obtained from $\mathbf{S}_{tr}^{L}$ with those estimated from the shrinkage covariance matrix of the training set $Y_S$, $\mathbf{S}_{tr}^{*S}$. It is apparent how the order of magnitude of the two types of scaling factors is comparable, although this time the shrinkage scaling factors result under-amplified with respect to the reference ones. The de-emphasizing effect is observed on almost all components, except the first four, and the factor of de-amplification is almost the same for all the components; hence, the MSD values are expected to be slightly smaller than the reference ones, and no distortion of the data variance should be observed. This is actually proved to be true by looking at the results of the damage identification exercise presented in Figure

(6.5d), and again comparing such results with those obtained using the large training sample data set, $Y^L$, given in Figure (6.5a). Use of the shrinkage covariance matrix of the training feature vectors in solving the damage identification problem produces 2% Type I error and 0% Type II errors.

These results are obviously satisfactorily, but it is interesting to explore a way to combine the advantages of the three approaches. It is in fact apparent that the over-shrinkage of some of the variance normalizing factors results in a reduced proportion of the MSD values of damaged and undamaged instances. In fact, the average value of the MSD from the mean of the large training sample of a testing feature representing an undamaged scenario is 0.23 times that of a testing feature representing a damaged condition, the average value of the MSD estimated via shrinkage covariance matrix from the mean of the small training data set of a testing feature from an undamaged condition is 0.31 times that of a testing feature representing a damage condition, then indicating a reduced difference between undamaged and damaged MSD values. This condition could cause an increase of Type II error in presence of low severity damage conditions.

## 4.7 Results obtained by a combination of the three proposed methods

Based on the results described in the preceding sections, a method combining the advantages of the three discussed techniques was finally proposed. Here the results and the methodology proposed to apply the 'combination method' are reported.

The methods are combined as follows: the dimensions of the features are first reduced employing the IDCT approach (Method 1). Then, the covariance matrix of the reduced features is evaluated and shrunk towards a diagonal target matrix, $\mathbf{T}$. The diagonal terms of $\mathbf{T}$ are the singular values of the sample covariance matrix. In this way, it is possible to exploit the ideal distribution of the standard deviation normalizing factors obtainable via pseudo-inverse. We would not get the same results by applying the pseudo-inverse to the covariance matrix shrunk towards the variance matrix of the features of reduced dimensions, as the shrinkage covariance estimate of the training features of reduced dimension is not ill-conditioned and the pseudo-inverse of a square non ill-conditioned matrix is only a computationally expensive way to evaluate the inverse of the matrix itself [1]. What we want to use of the pseudo-inverse approach (Method 2) is the distribution of the standard deviation normalizing factors, modified so as to make them more regularly distributed, i.e. avoiding the presence of factors unduly weighting the last components. The action of the feature dimension reduction is that of getting rid of components which are the most unlikely to bring information regarding actual patterns related to the structural behavior. The shrinkage operation is then instrumental to "bend" the covariance matrix towards a matrix whose eigenvalues are the singular values of the covariance matrix itself.

The result of such an operation is presented in Figure (6.4e) in terms of standard deviation normalizing factors. It may be noted that while the first half of the factors is under-weighted, the second half is over-weighted, when compared to the standard deviation normalizing factors of the larger data sample. The effects of such factors distribution may be appreciated in Figure (6.5e), where again the results of the damage identification exercise are depicted in terms of the MSD. It is observed that both Type I and Type II errors are equal to 1%, then representing an improvement with respect to all methods, even the one where a well-conditioned covariance matrix may be constructed, except than for the Method 3, which presents the optimal solution for the given set of available sensors and observations. The higher value of the last weighting factors contributes to a threshold value slightly greater than needed. In fact, the average value of the MSD of the tests from 1 to 100 is 61.37, while the threshold is equal to 140.8. Nonetheless, the fact that the last components are over-weighted counteracts the effect of an excessively large threshold, keeping the Type II error within very low values.

# 5. Validation

The proposed methods were validated performing damage identification on a laboratory scale steel frame excited at the base at the shaking table facility available at the Carleton Laboratory of Columbia University and conducting damage location on simulated data from a bridge deck model. The following sections outline the obtained results.

## 5.1 Experimental Steel Frame

The laboratory structure used to investigate the issues involved in the practical implementation of the proposed approaches was a four-story A36 steel frame with an inter-story of 533 mm and floor plate dimensions of 610×457×12.7 mm (Figure (6.7)). The floors are braced diagonally in only one direction, hereafter denoted as strong direction. The brace and the column elements have 50.8×6.4 mm and 50.8×9.5 mm cross section dimensions, respectively. All structural connections are bolted. In addition to said reference configuration, denoted as **U1** in the following, two undamaged scenarios (**U2** and **U3**) were considered, in order to simulate operational variability. A summary of the different frame states considered in this work is given in Table (6.1). Undamaged condition **U2** was simulated by adding one mass at the third floor between columns A and B, on edge C, while the undamaged scenario **U3** was represented by a setting where two masses were placed between columns A and B, on both edges C and D. Undamaged scenarios **U2** and **U3** were considered to simulate the effects of systematic changes in the structural properties induced by variations of operational conditions or functionality, such as the addition of a library at the third floor of the building. Moreover, the addition of mass provokes a change in the frequency content of the structural response time histories similar to that induced by a decrease in stiffness. Hence, with the addition of masses in localized areas of the frame, we can also simulate localized changes in stiffness properties of the structure, as the ones that can occur due to changes in environmental conditions. The first damage scenario (**D1**) was simulated by replacing the two column elements on side A of the third inter-story with elements with reduced cross-section along the input direction; the second damage scenario (**D2**) was obtained by replacing all the column elements of the third inter-story by elements of reduced cross-section; finally, the third damage scenario (**D3**) was achieved by reducing the cross section of two column elements, one at the second and one at the third inter-story, both along column A. The column element of reduced dimensions was characterized by 50.8×6.5 mm cross-section: replacing one of the original columns with such an element induced a 15% inter-story stiffness reduction. Therefore, the first damage condition was characterized by 30% stiffness reduction at the third inter-story, the second by 60% stiffness reduction at the third inter-story, while the third damage scenario represented a condition where the stiffness of both second and third inter-stories were reduced to 85% of their original values.

The structure was excited along the weak direction of bending by means of a medium-scale uniaxial hydraulic shake table. However, preliminary studies showed that the structure is not of ideal shear-type kind, but torsional modes in the strong direction of bending may be identified, even when the load is applied along the weak direction ([6, 16, 27]). Two types of time histories were used as inputs: four ground acceleration time histories recorded during El Centro (1940), Hachinohe (1983), Northridge (1994) and Kobe (1995) earthquakes, and the acceleration time history obtained from the design spectrum of the EuroCode08. To ensure the structure be excited by the proper range of the time histories power spectra, a time scale of $\frac{1}{\sqrt{3}}$ was introduced and, to prevent yielding and additional unexpected damage, the input time histories were properly scaled in magnitude. Inputs and outputs were sampled at 400 Hz. The structure was instrumented with 8 piezo-electric accelerometers located as displayed in Figure 6.7, measuring accelerations along the weak direction.

The training set consisted of 20 data sets from each of the three undamaged scenarios, **U1**, **U2** and **U3**, so that the training model was evaluated using 60 data sets. The testing data set comprised 105 data sets, 10 from each one of the undamaged scenarios, 15 from damage scenario **D1** and 30 from each one of the damage scenarios **D2** and **D3**. Each testing data set was used individually, so to perform 105 tests; in other words, 105 tests were run, each characterized by $n_{te}$ equal to 1. Five different sensor setups were considered, using the response recorded from only some of the available instruments. The characteristics of the five sensor setups, labeled as **S1**, **S2**, etc., are depicted in Table 6.2, where the ratio between the feature dimension, $p$, and the number of observations employed to construct the training model, $n$, is also presented.

Table 6.3 presents the results in terms of Type I error, and Table 6.4 those in terms of Type II error, showing how much each scenario concurs to the overall error value. In Tables 6.3 and 6.4 the three approaches presented in Sections 2.1, 2.2 and 2.3 are labeled as *Method 1*, *Method 2* and *Method 3*, respectively, while the combination of the three methods is labeled as *Methods Combination*. In both tables, in the first column, under the label of the considered sensor setup, the values of various parameters are indicated. The value of $d$ indicates the dimensions of the feature vectors when the Discrete Cosine Transform method is employed, $p$ indicates the original dimensions of the feature vectors, $\lambda_3$ denotes the value of the shrinkage coefficient obtained using the variance matrix as target, while $\lambda_4$ refers to the value of the shrinkage coefficient when the matrix of the singular values of the covariance matrix is used as target.

The percentage error values presented in Table 6.3 for the individual undamaged states were obtained on a set of 10 individuals. Then, for example, the 10% error observed for State **U1** when using *Method 1* and the sensor setup **S2** is due to the fact that 1 out of the 10 tests performed on the structure under state **U1** is erroneously labeled as damaged. On the other hand, the overall Type I error observed when using *Method 1* and the second sensor setup (**S2**) is equal to 6.67% because 2 out of the 30 tests performed on the system under undamaged conditions were incorrectly identified as representative of damaged conditions, 1 by erroneously flagging damaged an instance from state **U1** and the other from incorrectly labeling as damaged an instance from state **U2**. The results are presented in this fashion in order to emphasize the fact that state **U2** is the most complex to identify.

In fact, for all of the methods and the sensor setups, state **U2** is almost always the only responsible for a non zero Type I error. This undamaged scenario induced a strong torsion to the frame, as one eccentric mass is added at the third floor of the frame to obtain this second healthy condition. Therefore, even though instances of such condition were used to construct the training, it may be speculated that such instances will be characterized by high values of the MSD. During testing, instances of the second healthy condition are then the most likely to overcome the threshold value, being then flagged as instances of a damaged condition. However, it must be noted that the overall Type I error maintains very low values, and even the partial errors can be considered low, as in fact never more than 3 instances were incorrectly labeled as damaged. Moreover, it is worth noting that the approach which employs the combination of the three methods gave the minimum Type I error for sensor setups **S1**, **S3** and **S5**, while it gave the second best results in terms of Type I error for the other two sensor setups. It is worth mentioning that the value of the shrinkage coefficient is very low both for *Method 3* and for the 'combined' approach, suggesting that the obtained covariance matrix was very similar to the original sample covariance matrix, especially when using the combined approach. This observation allows us to speculate that the transformation of the eigenvalues as operated by the shrinkage estimate is indeed the only transformation acted on the sample covariance matrix, and that its overall structure is preserved.

For what concerns Type II error, the results are again presented in terms of partial and total Type II error. In this case, there were no particular damage conditions consistently responsible for misidentifications. It is certainly observed that Type II error is much lower than the Type I, albeit it is important to remember that the overall Type II error is evaluated on a population of 75 individuals: 15 from damage state **D1** and 30 for damage states **D2** and **D3**. Probably, if we were able to perform tests on more undamaged instances, the Type I error would decrease. It is yet important to stress out the fact that the *methods combination* gave always 0% Type II error, proposing itself as a robust approach to solve the problem of damage detection using pattern recognition with limited number of training observations.

## 5.2 Bridge Deck

To look at the applicability of the proposed approach to a bridge-type structure, we considered the lumped mass model of a bridge deck of Figure 6.8. The model consisted of 12 lumped masses and 20 flexural links, and was assumed to vibrate along the $z$ direction. The energy dissipation characteristics of the system were modeled through proportional damping, by assigning $1\%$ damping ratio to each mode. Table 6.5 lists the 9 different states considered. States **U1** to **U5** represent different healthy conditions. State **U1** can represent the baseline state, characterized, for example, by normal traffic conditions and average temperature. State **U2** may represent a scenario where only the $-y$ side of the bridge deck is subjected to a temperature increase, while state **U3** corresponds to a state where the same edge of the bridge deck is subjected to a temperature decrease. Such local changes in the stiffness properties due to temperature fluctuations may occur as a result of daily changes of the relative positions of sun and bridge. Likewise, **U4** may represent a scenario where the $+y$ edge of the deck is subject to a temperature increase, while state **U5** could depict the condition where the same side of the deck undergoes a temperature decrease. States **D1** to **D4** represent different damage scenarios, with local damages defined with respect to different undamaged states. States **D1** and **D2** represent conditions where damage occurred between degrees of freedom 5 and 6, where **D1** represents a damage condition more severe than that characterizing the damage state **D2**, as for state **D2** only $15\%$ decrease in local stiffness of element $k_6$ is simulated, versus the $20\%$ decrease in stiffness of the same element simulated for state **D1**. Similarly, damage states **D3** and **D4** represent scenarios where damage occurred between degrees of freedom 2 and 8, where state **D3** represents a damage condition more severe than that depicted by state **D4**.

To construct the training data sets, 5 tests were simulated, one from each of the five healthy states. For a given test $r$, the value of the $i^{th}$ stiffness parameter, $k_i^{(r)}$, $r = 1, 2, \ldots, 5$, was chosen as $k_i^{U_m} + \mathcal{U}(-\pm 0.01) k_i^{Um}$, where $k_i^{Um}$ is the mean value of the flexural stiffness $k_i$ for the undamaged scenario $\mathbf{U}_m$, $m = 1, 2, \ldots, 5$, (Table 6.5), and $\mathcal{U}(\pm 0.01)$ is the uniform probability distribution between the limits - 0.01 and 0.01. The variability of the stiffness parameters depicted in Table 6.5

was adopted to model systematic changes induced, for instance, by environmental effects, while the variability induced by the perturbation obtained via $\mathcal{U}(\pm 0.01)$ was used to mimic the effects of operational and modeling assumptions. Each of the 5 sets of healthy structural parameters was used to simulate the response of the system to Gaussian white noise input force applied at all the DOFs. The resulting response accelerations were corrupted by adding 10$\%$ root mean square Gaussian white noise sequences, to represent measurement noise. In this analysis, only the response accelerations measured at DOFs 4 - 9, 11 and 12 (shaded in a darker color in Figure 6.8) were considered, so to represent a partial instrumentation set-up. Note that while two sensors are adjacent to damaged element $k_6$ of states **D1** and **D2**, located at degrees of freedom 5 and 6, only one sensor located at degree of freedom 8 is available to detect damage of element $k_{16}$, representing the damaged element of conditions **D3** and **D4**. Hence, identification and location of damage for scenarios **D1** and **D2** should result easier than that for conditions **D3** and **D4**.

To construct the testing ensemble, 30 tests were performed on each of the 9 states of Table 6.5, and the structural response was simulated adopting the same procedure used to construct the training data sample.

Figures from 6.9 to 6.17 show the results of the damage location exercises for the 9 different states and for the four different methods in terms of average p-values. In order to obtain Figures 6.9 to 6.17, 270 tests were performed according to the indications in Section 3.2.2. Then, the 30 p-values corresponding to the 30 tests performed on the time histories recorded from the same sensor under same structural state were averaged. In Figures 6.9 to 6.17 one bar represents such an average value. In said figures, ‘Method 4’ refers to the combination of the three methods. It is worth reminding that we expect the p-values associated to all sensors be larger than 0.95 for the undamaged states **U1** to **U5**. On the contrary, we expect the p-values associated to sensors 5 and 6 to be smaller than 0.95 for states **D1** and **D2**, while we expect the p-value associated to sensor 8 to be smaller than 0.95 for states **D3** and **D4**. The threshold p-value 0.95 is marked with a dashed red line in Figures 6.9 - 6.17.

It is observed that the shrinkage method (Method 3) is again the one delivering the best results, as the p-values associated to undamaged areas of the bridge are always above 0.95 while those associated with damaged areas of the bridge are always below 0.95. The only exception is represented by the average p-values associated to sensor 7, that are always approximately equal to 0.92, irrespective of the structural condition. Moreover, when using the shrinkage estimator of the covariance matrix in evaluating the MSD, the magnitude of the p-value decreases at increasing damage extent: damage scenario **D1** is more severe than damage scenario **D2**, likewise, damage scenario **D3** is more severe than damage scenario **D4**. This is reflected in the p-values, as in fact the average p-values associated to sensors 5 and 6 are 0.86 and 0.72 for state **D1** and 0.90 and 0.81 for state **D2**. Similarly, the average p-value associated to sensor 8 is equal to 0.85 for state **D3** and to 0.90 for state **D4**.

Similarly to what observed in Section 4.4, decreasing the feature vector dimensions via DCT (Method 1) gives optimal results in terms of Type I error, as the undamaged areas of the bridge are always identified as undamaged. However, the method is unable to locate damage between sensors 5 and 6, while for state **D3** the average p-value associated to sensor 8 is equal to 0.95, indicating low confidence in declaring the structure healthy around that sensor. It is worth mentioning that although for state **D4** the average p-value associated to sensor 8 is equal to 0.97, the average p-values associated with the other sensors are all around 1. Then looking at these results, the engineer would be encouraged to further investigate the structural conditions in the vicinity of channel 8.

On the other hand, the use of the pseudo-inverse (Method 2) delivers very conservative results: the average p-values associated to sensors 6, 7 and, sometimes, 12 are always below or around 0.95, even for undamaged conditions. It is noteworthy, though, that the average p-values associated to sensors 5 and 6 for states **D1** and **D2** and those associated to sensor 8 for states **D3** and **D4** are correctly well below 0.95.

Finally, the combination of the three methods (Method 4) does not improve the results obtained using the shrinkage estimator of the covariance matrix. On the contrary, while the method is always successful in identifying damage location or, at least, triggering an alarm for further inspection in the damage areas, using the combination of the methods does not allow to locate damage around sensor 5. This means that, if the combination approach were used for states **D1** and **D2**, one would be led to inspect the structural conditions around the whole zone surrounding sensor 6, rather than just that between sensors 5 and 6.

In order to better understand these results, further analysis of the relationship between the cepstral features and structural properties should be performed.

# 6. Conclusions

The problem of statistical pattern recognition based structural damage detection when only small training data sets are available was studied. The fundamental problem one must deal with when operating with a reduced number of training instances is that of inverting a possibly ill-conditioned covariance matrix. Four alternatives were proposed in order to solve the structural health monitoring assignment under the condition of scarcity of training data: reduction of feature dimensionality via discrete cosine transform, inversion of the unbiased sample covariance matrix via pseudo-inverse, use of the shrinkage estimate in place of the more common sample estimate of the covariance matrix and a combination of the three techniques.

The behavior of each approach was studied by analyzing the patterns of the eigenvalues of the covariance matrix estimated using the different techniques. It was observed that while the recursion to the discrete cosine transform and pseudo-inverse delivered a covariance matrix characterized by under-estimated values of the variance associated to the most dispersed feature components, the use of the shrinkage covariance matrix estimate restrained the covariance eigenvalues to be regularly distributed within a bounded range, though slightly over-estimating the variance of the most dispersed feature components.

The performance of the four techniques were further investigated when cast within the statistical pattern recognition scheme to solve damage identification and location problems. In this work, the Mahalanobis Squared Distance was used to measure the departure of the testing feature vectors from the probabilistic model representing the training features. It was shown how the Mahalanobis metric value is intimately related to the eigenvalues of the training sample covariance matrix. In fact, it was clarified how the Mahalanobis Squared Distance may be thought of as a weighted Euclidean distance where the weighting factors are the square roots of the reciprocal of the covariance matrix eigenvalues. The presence of components whose magnitude largely exceeds that of other components induces the value of the MSD to be mainly influenced solely by those components, upon which the whole damage detection outcomes depend. In the context of damage identification, the negative consequences of such condition were particularly evident when the pseudo-inverse of the covariance matrix was used in place of its inverse in evaluating the MSD, and resulted in very high false acceptance errors. High Type II error was also observed when the threshold value is overestimated, occurrence observed when features, whose dimensions have been reduced employing the inverse discrete cosine transform, are employed in the damage detection assignment. The overestimation of the threshold was found to be due to the distortion operated by the weighting factors to the training data, distortion that led some of the training data to be associated with high values of exclusive Mahalanobis Squared Distance causing an increase of the threshold value. The performance of the proposed methods was also studied in the context of damage location, where results were reported in terms of indices that may be interpreted similarly to the p-values used to report the results in standard binary hypothesis testing problems. The results obtained using the shrinkage covariance estimator were particularly promising, but farther analysis on the relationship between cepstral coefficients and structural parameters are needed to better understand the results obtained in the damage location applications. In conclusion, the shrinkage estimator of the covariance matrix seems to represent the most promising technique able to correctly characterize the healthy and damaged conditions of the structure even in presence of limited training data, by giving reasonably low values of both Type I and Type II errors. It is in fact worth stressing once again that a robust SHM algorithm must guarantee high accuracy both in detecting damage when damage has in fact occurred (low Type II error), but also not triggering too many false alarms (low Type I error) so as to not unduly increase the cost of inspections, being that of management cost saving the primary reason why the bridge manager will take recourse to structural health monitoring in the first place.

# Acknowledgements

- Dyab Khazem of the Parsons Transportation Group for his valuable contributions, comments and suggestions during the entire project.

- Financial support from from the Center for Advanced Infrastructure and Transportation (CAIT), Rutgers, the State University of New Jersey.

- Dr. Luciana Balsamo for her help with the research discussed herein.

- Adrian Brugger, Prof. Manolis Chatzis, Prof. Suparno Mukhopadhyay and Prof. Andrew Smyth for their collaboration during the experimental phase of the work.

# Bibliography

[1] *MATLAB 7.9 (R2009b), The MathWorks, Inc., Natick, Massachusetts, United States.*, 2009.

[2] N. Ahmed, T. Natarajan, and K. R. Rao. Discrete Cosine Transform. *IEEE Transactions on Computers*, 23(1):90–93, 1974.

[3] J.C. Anderson and D.W. Gerbing. The effect of sampling error on convergence, improper solutions, and goodness-of-fit indeces for maximum likelihood confirmatory factor analysis. *Psychometrika*, 4931:155–173, 1984.

[4] L. Balsamo, R. Betti, and H. Beigi. Structural damage detection using speaker recognition techniques. *Proceedings of the 11th International Conference on Structural Safety and Reliability (ICOSSAR)*, pages 1871–1879, 2013.

[5] L. Balsamo, R. Betti, and H. Beigi. A structural health monitoring strategy using cepstral features. *Journal of Sound and Vibration*, 333:4526–4542, 2014.

[6] L. Balsamo, S. Mukhopadhyay, and R. Betti. A statistical framework for structural health monitoring with stiffness proportional damage sensitive features (under review). *Smart Structures and Systems*, 2014.

[7] V. Barnett and T. Lewis. *Outliers in statisitcal data*. Chilchester: Wiley, 1994.

[8] H. Beigi. *Fundamentals of Speaker Recognition*. Springer US, 2011.

[9] S. H. Cheung and J. L. Beck. Bayesian model updating using hybrid Monte Carlo simulation with application to structural dynamic models with many uncertain parameters. *Journal of Engineering Mechanics*, 135:243–255, 2009.

[10] E. J. Cross, G. Manson, K. Worden, and S. G. Pierce. Features for damage detection with insensitivity to environmental and operational variations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 468:4098–4122, 2012.

[11] B. Efron and C. Morris. Stein's paradox in statistics. *Scientific American*, (237):119–127, 1977.

[12] C. R. Farrar, T. A. Duffey, S. W. Doebling, and D. A. Nix. A statistical pattern recognition paradigm for vibration-based structural health monitoring. In *2nd International Workshop on Structural Health Monitoring, Stanford, CA*, 1999.

[13] C. R. Farrar and K. Worden. *Structural health monitoring: A machine learning perspective*. John Wiley and Sons, 2013.

[14] E. Figueiredo, G. Park, C. R. Farrar, K. Worden, and J. Figueiras. Machine learning algorithms for damage detection under operational and environmental variability. *Structural Health Monitoring*, 10(6):559–572, 2011.

[15] R. Fraile, N. Senz-Lechn, J. I. Godino-Llorente, V. Osma-Ruiz, and P. Gmez-Vilda. Use of Mel-Frequency Cepstral Coefficients for automatic pathology detection on sustained vowel phonations: mathematical and statistical justification. *Proc. 4th International Symposium on Image/Video Communications over Fixed and Mobile Networks (Bilbao, Brazil, July 2008)*, 2013.

[16] G. Fraraccio, A. Brugger, and R. Betti. Experimental studies on damage detection in frame structures using vibration measurements. *Shock and Vibration*, 17:697–721, 2010.

[17] D. George, N. Hunter, C. R. Farrar, and R. Deen. Identifying damage sensitive features using nonlinear time-series and bispectral analysis. *Proceedings of the International Modal Analysis Conference - IMAC*, 2:1796–1802, 2000. cited By (since 1996)11.

[18] T. Granchev. *Contemporary Methods for Speech Parameterization*. Springer, 2011.

[19] M. Gul and F. N. Catbas. Statistical pattern recognition for structural health monitoring using time series modeling: Theory and experimental verifications. *Mechanical Systems and Signal Processing*, 23:2192–2204, 2009.

[20] M. Gul and F. N. Catbas. Damage assessment with ambient vibration data using a novel time series analysis methodology. *Journal of Structural Engineering*, 137(12):1518–1526, 2011.

[21] J. Hardin and D. M. Rocke. The distribution of robust distances. *Journal of Computational and Graphical Statistics*, 14:928–946, 2005.

[22] O. Ledoit and M. Wolf. A well-conditioned estimator for large dimensional covarinace matrices. *Journal of Multivariate Analysis*, 88:365–411, 2004.

[23] Y. Lei, A. S. Kiremidjian, K. K. Nair, J. P. Lynch, K. H. La, T. W. Kenny, E. Carryer, and A. Kottapalli. Statistical damage detection using time series analysis on a structural health monitoring benchmark problem. *Proceedings of the 9th International Conference on Applications of Statistics and Probability in Civil Engineering, San Francisco, CA, USA.*, July 6-9, 2003.

[24] Y.-Q. Lin and W.-X. Ren. Stochastic state space model-based damage detection of engineering structures. *Zhendong Gongcheng Xuebao/Journal of Vibration Engineering*, 20(6):599–605, 2007.

[25] P. C. Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences of india*, 2:49–55, 1936.

[26] A. A. Mosavi, D. Dickey, R. Seracino, and S. Rizkalla. Identifying damage locations under ambient vibrations utilizing vector autoregressive models and mahalanobis distances. *Mechanical Systems and Signal Processing*, 26(1):254–267, 2012. cited By (since 1996)3.

[27] S. Mukhopadhyay, H. Lus, and R. Betti. Structural identification with base excitation, part ii. *(under review)*, 2014.

[28] J. Schäfer and K. Strimmer. A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical Applications in Genetics and Molecular Biology*, 4(32), 2005.

[29] H. Sohn, C. R. Farrar, N. F. Hunter, and K. Worden. Structural Health Monitoring using statistical pattern recognition techniques. *Transactions of the ASME*, 123:706–711, 2001.

[30] C. Stein. Inadmissibility of the usual estimator for the mean of a multivaritate distirbution. *Proceedings of the Third Berkeley Symposium on Mathematics, Statistics and Probability, Berkeley, University of California Press*, 1:197–206, 1956.

[31] A.-M. Yan, G. Kerschen, P. De Boe, and J.-C. Golinval. Structural damage diagnosis under varying environmental conditions - Part II: Local PCA for non-linear cases. *Mechanical Systems and Signal Processing*, 19:865–880, 2005.

# List of Figures

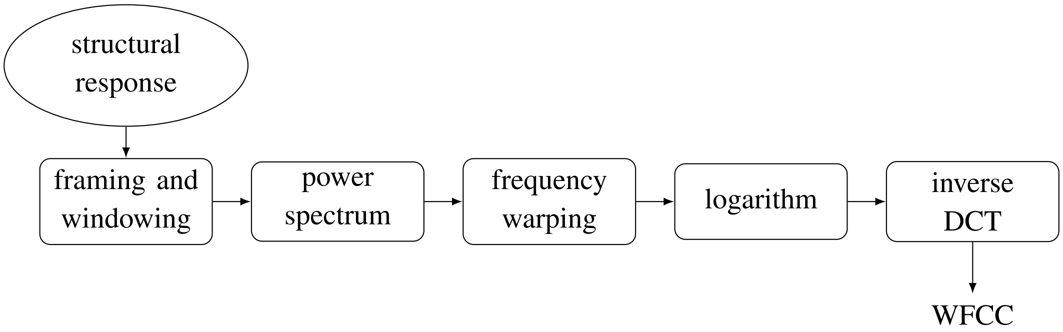


Figure 6.1: Mel-Frequency Cepstral Coefficients Extraction

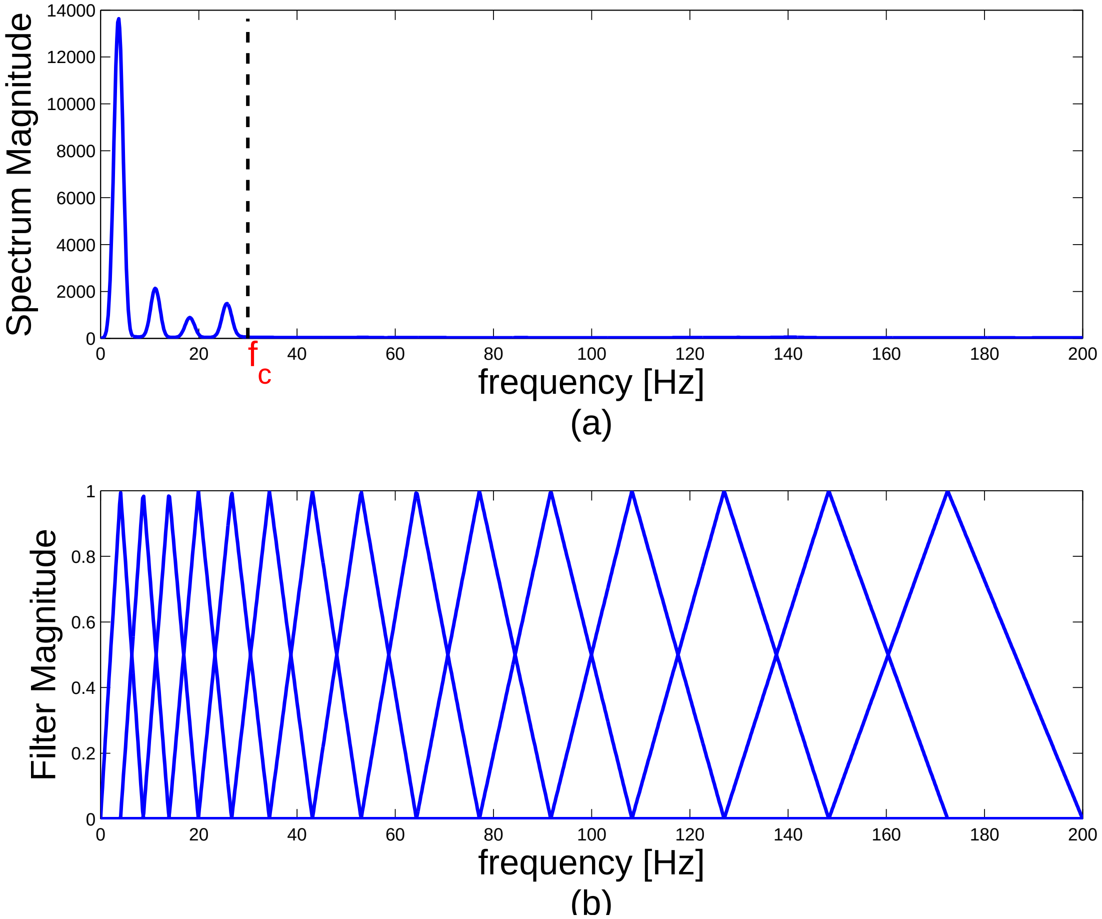


Figure 6.2: Warping procedure: (a) Cutoff frequency selection, (b) Triangular filters.

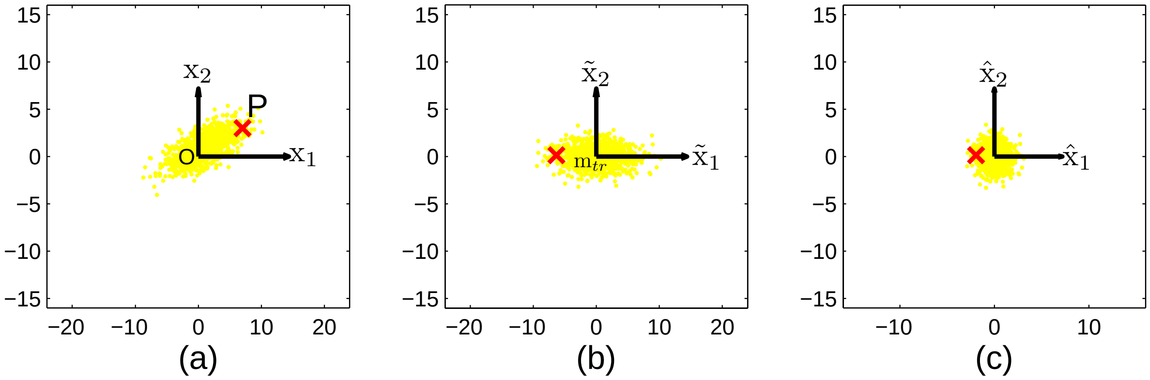
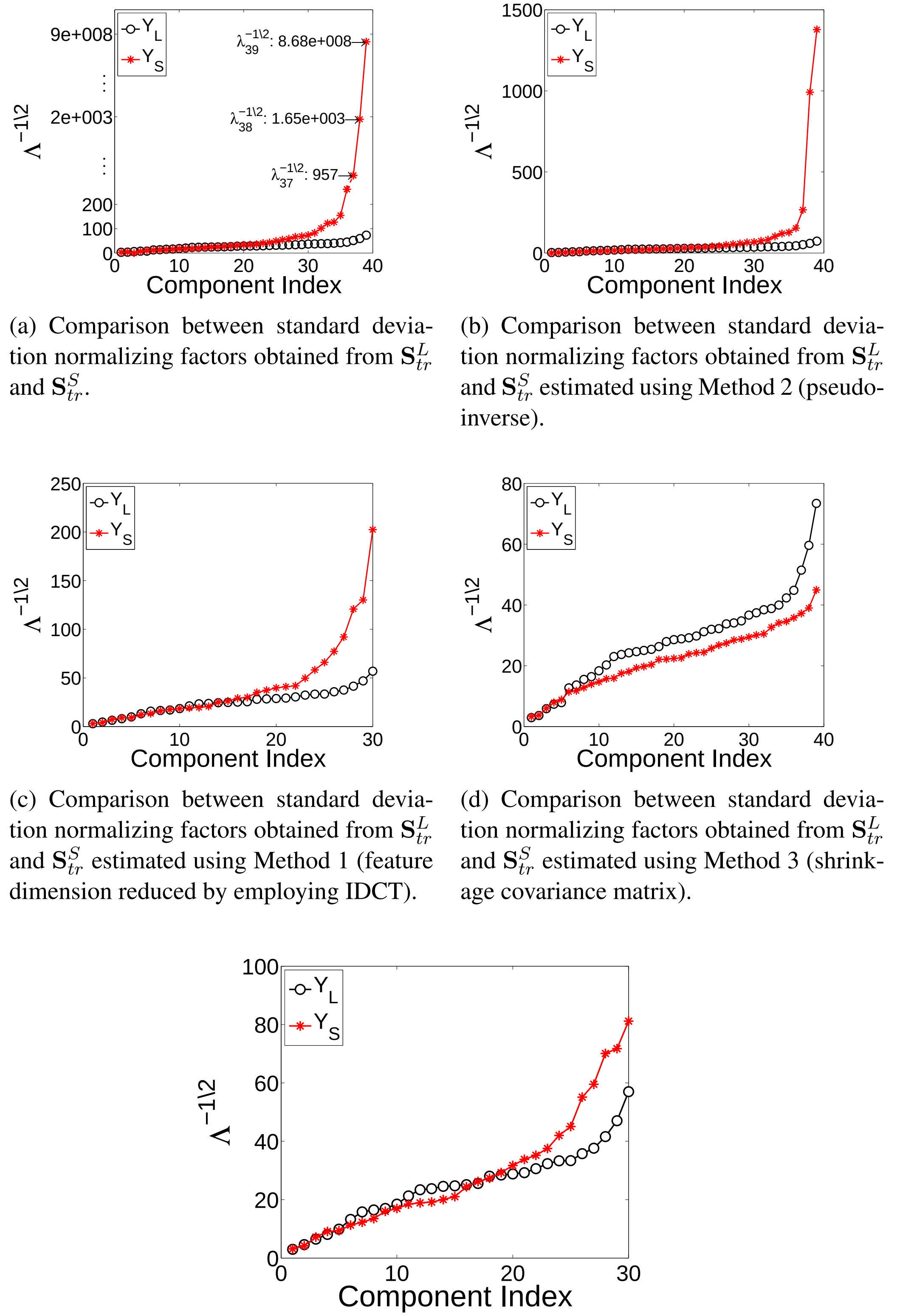


Figure 6.3: Mahalanobis Squared Distance evaluation: (a) Realizations of a bi-variate Gaussian distribution; (b) Effect of projection of the original data set into a system of reference with origin in $\mathbf{m}_{tr}$ and axis parallel to the principal components of $\mathbf{S}_{tr}$; (c) Effect of variance normalization.



(a) Comparison between standard deviation normalizing factors obtained from $\mathbf{S}_{tr}^{L}$ and $\mathbf{S}_{tr}^{S}$.

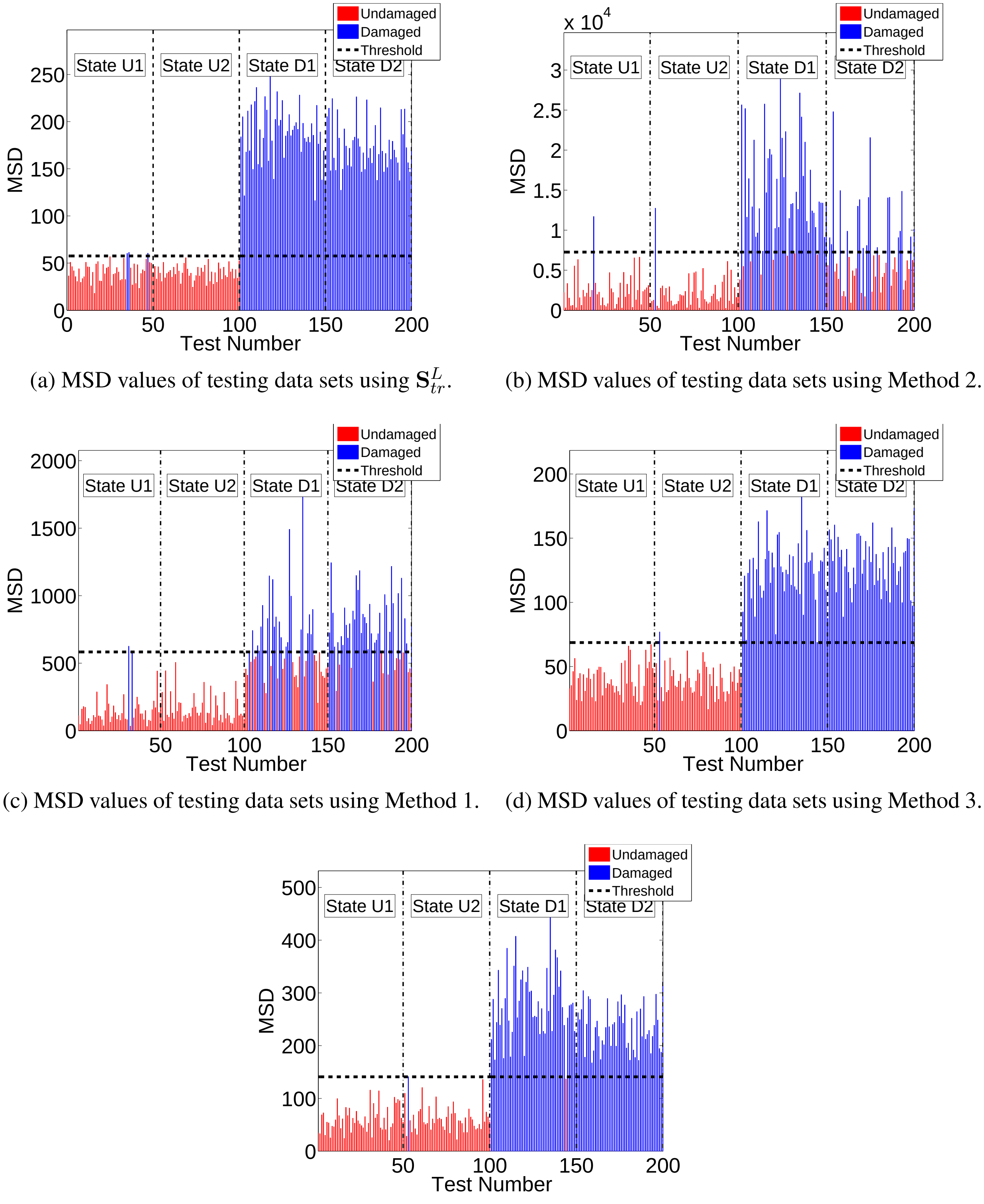
(b) Comparison between standard deviation normalizing factors obtained from $\mathbf{S}_{tr}^{L}$ and $\mathbf{S}_{tr}^{S}$ estimated using Method 2 (pseudo-inverse).

(c) Comparison between standard deviation normalizing factors obtained from $\mathbf{S}_{tr}^{L}$ and $\mathbf{S}_{tr}^{S}$ estimated using Method 1 (feature dimension reduced by employing IDCT).

(d) Comparison between standard deviation normalizing factors obtained from $\mathbf{S}_{tr}^{L}$ and $\mathbf{S}_{tr}^{S}$ estimated using Method 3 (shrinkage covariance matrix).

(e) Comparison between standard deviation normalizing factors obtained from $\mathbf{S}_{tr}^{L}$ and $\mathbf{S}_{tr}^{S}$ estimated combining the the advantages of the three proposed methods.

Figure 6.4: Comparison between standard deviation normalizing factors obtained from $\mathbf{S}_{tr}^{L}$ and from the different covariance estimators discussed in Section 2.



(a) MSD values of testing data sets using $\mathbf{S}_{tr}^{L}$.

(b) MSD values of testing data sets using Method 2.

(c) MSD values of testing data sets using Method 1.

(d) MSD values of testing data sets using Method 3.

(e) MSD values of testing data sets combining the the advantages of the three proposed methods.

Figure 6.5: MSD values of testing data sets.

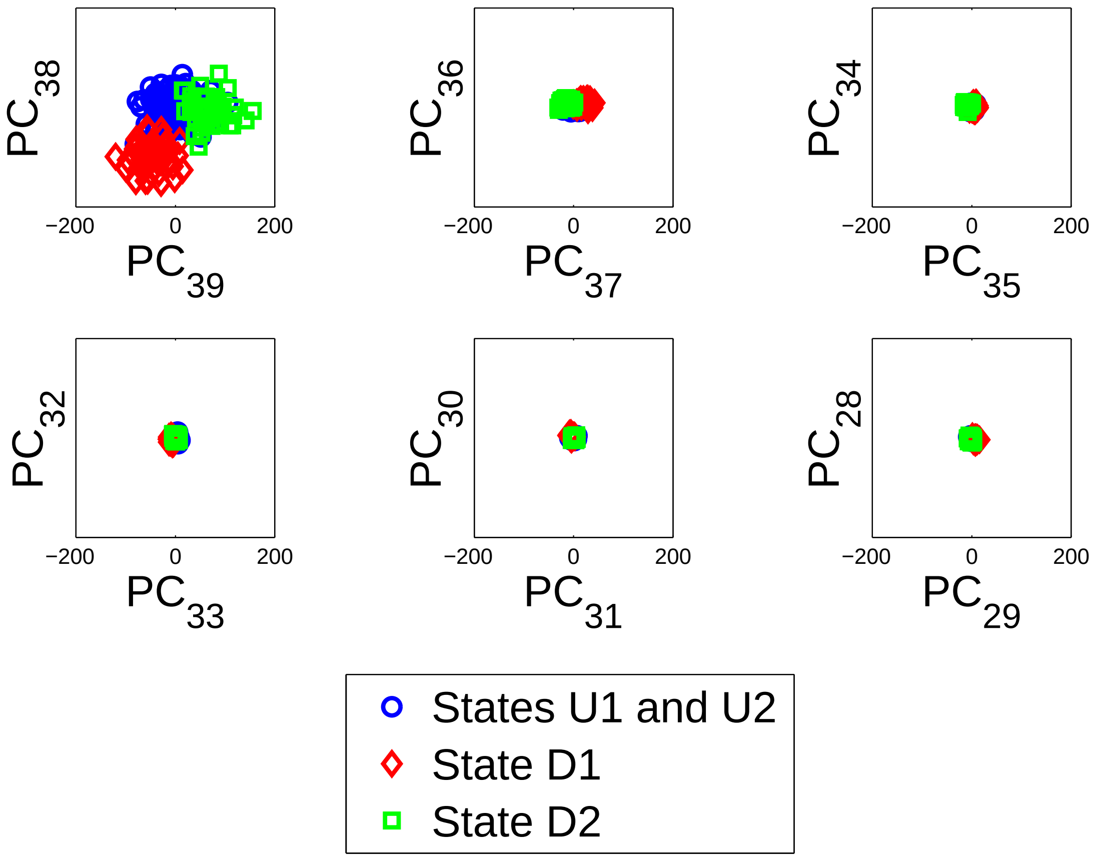


Figure 6.6: Testing feature components obtained through affine transformation applied using Method 1.

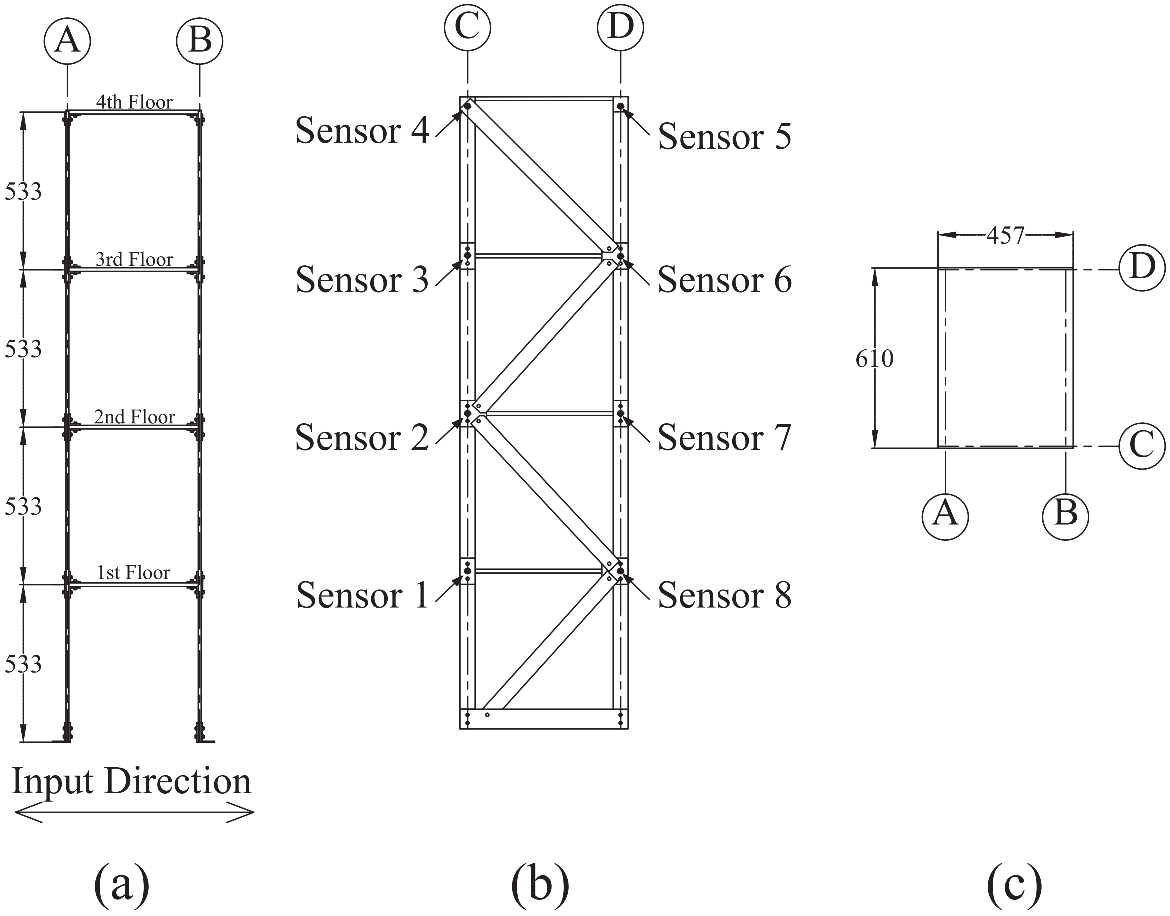


Figure 6.7: Sensors location. Dimensions are in millimeters.

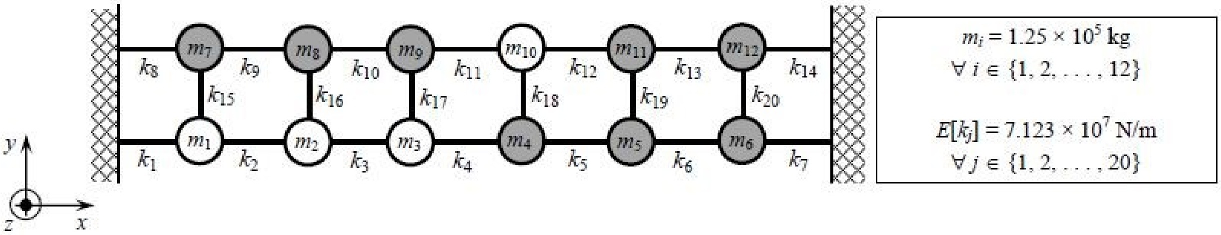


Figure 6.8: Model of the bridge deck used to validate the proposed methods.

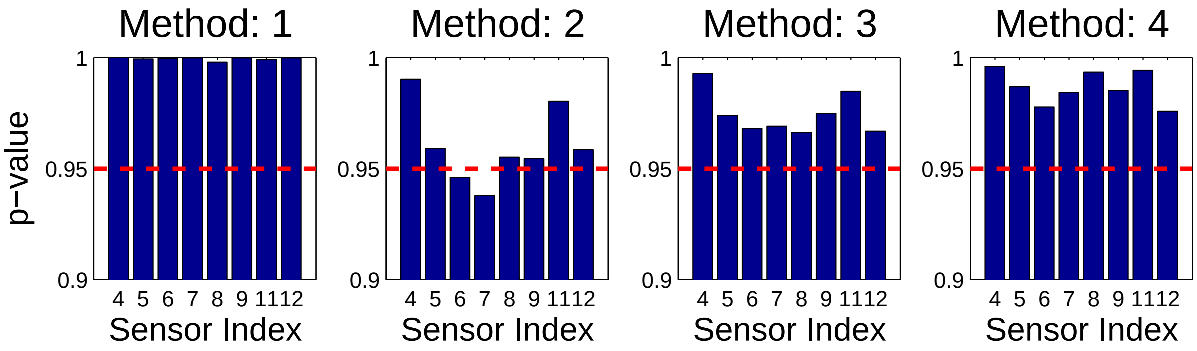


Figure 6.9: p-values for damage location for state **U1** - healthy state. A bar corresponding to the $i^{th}$ sensor index portraying a value lower than 0.95 (red dashed line) flags damage occurrence in the vicinity of sensor $i$.

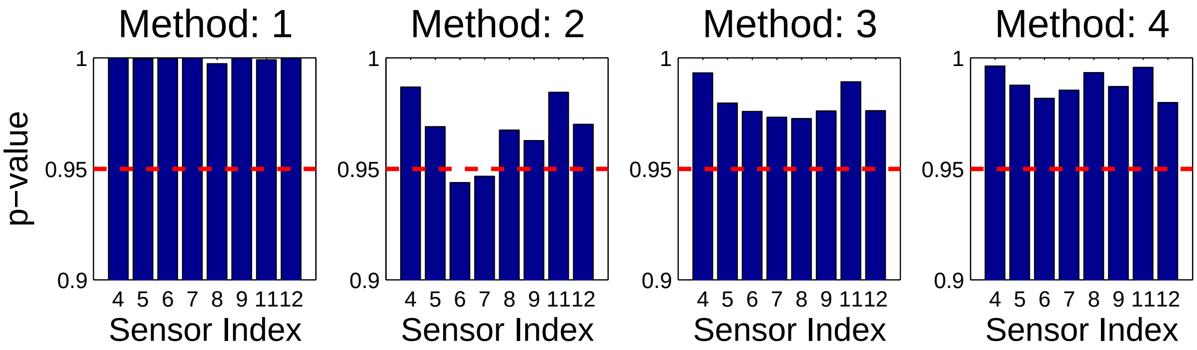


Figure 6.10: p-values for damage location for state **U2** - healthy state. A bar corresponding to the $i^{th}$ sensor index portraying a value lower than 0.95 (red dashed line) flags damage occurrence in the vicinity of sensor $i$.

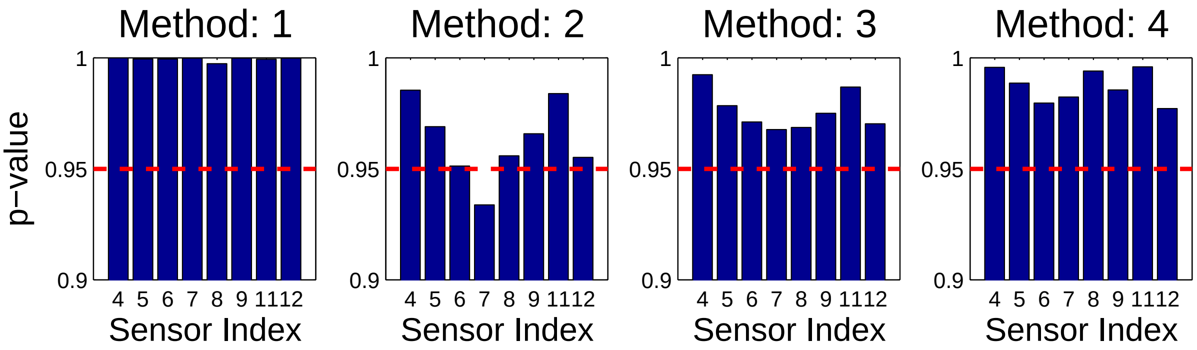


Figure 6.11: p-values for damage location for state **U3** - healthy state. A bar corresponding to the $i^{th}$ sensor index portraying a value lower than 0.95 (red dashed line) flags damage occurrence in the vicinity of sensor $i$.

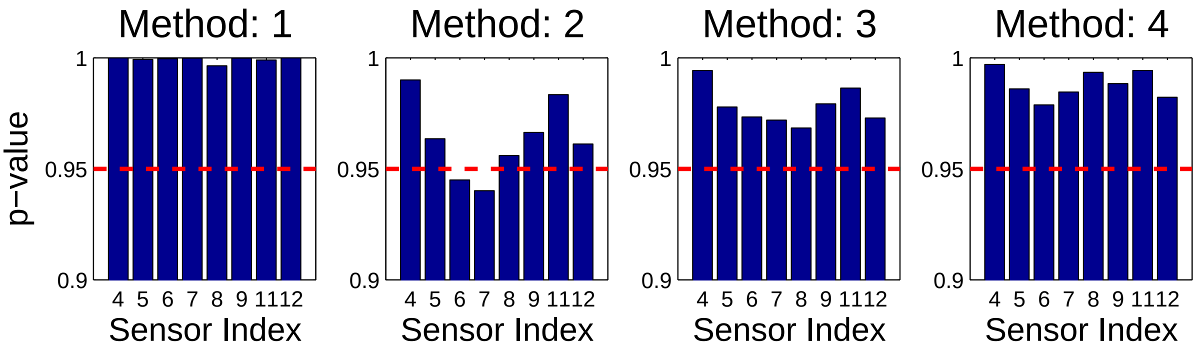


Figure 6.12: p-values for damage location for state **U4** - healthy state. A bar corresponding to the $i^{th}$ sensor index portraying a value lower than 0.95 (red dashed line) flags damage occurrence in the vicinity of sensor $i$.

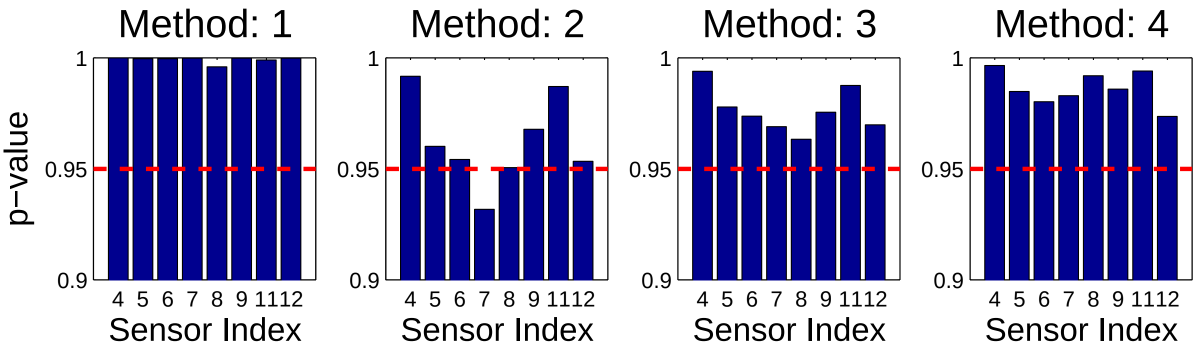


Figure 6.13: p-values for damage location for state **U5** - healthy state. A bar corresponding to the $i^{th}$ sensor index portraying a value lower than 0.95 (red dashed line) flags damage occurrence in the vicinity of sensor $i$.

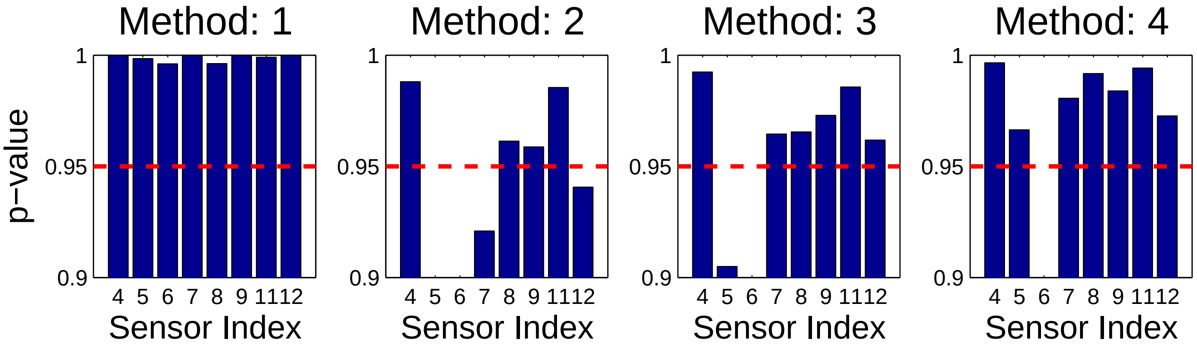


Figure 6.14: p-values for damage location for state **D1** - damage state, damage simulated as 20% stiffness decrease of the element between degrees of freedom 5 and 6. A bar corresponding to the $i^{th}$ sensor index portraying a value lower than 0.95 (red dashed line) flags damage occurrence in the vicinity of sensor $i$.

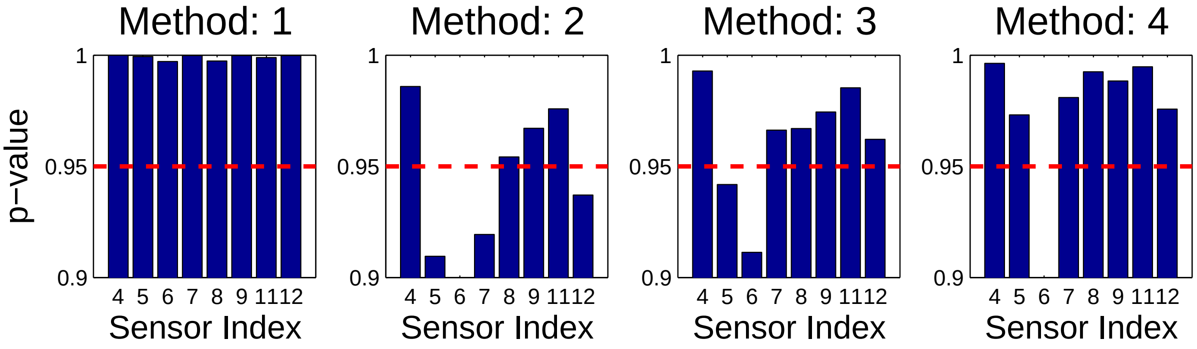


Figure 6.15: p-values for damage location for state **D2** - damage state, damage simulated as 15% stiffness decrease of the element between degrees of freedom 5 and 6. A bar corresponding to the $i^{th}$ sensor index portraying a value lower than 0.95 (red dashed line) flags damage occurrence in the vicinity of sensor $i$.

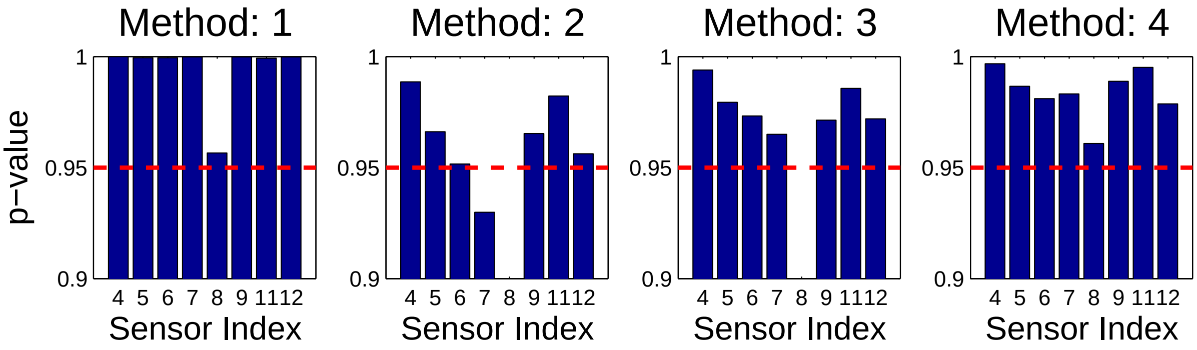


Figure 6.16: p-values for damage location for state **D3** - damage state, damage simulated as 15% stiffness decrease of the element in the vicinity of degree of freedom 8. A bar corresponding to the $i^{th}$ sensor index portraying a value lower than 0.95 (red dashed line) flags damage occurrence in the vicinity of sensor $i$.

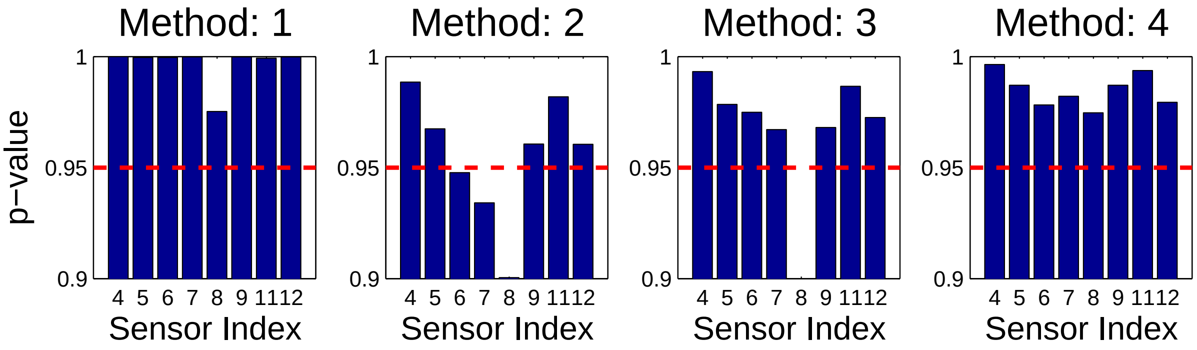


Figure 6.17: p-values for damage location for state **D4** - damage state, damage simulated as 20% stiffness decrease of the element in the vicinity of degree of freedom 8. A bar corresponding to the $i^{th}$ sensor index portraying a value lower than 0.95 (red dashed line) flags damage occurrence in the vicinity of sensor $i$.

# List of Tables

Table 6.1: Different States of the Steel Frame Considered for the Experimental Validation of the Proposed Damage Detection Techniques.

| **State** | **Condition** | **Description** |
|---|---|---|
| U1 | Undamaged | Baseline |
| U2 | Undamaged | 20% mass addition to the 3$^{rd}$ floor |
| U3 | Undamaged | 40% mass addition to the 3$^{rd}$ floor |
| D1 | Damaged | 15% stiffness reduction at 3$^{rd}$ floor |
| D2 | Damaged | 60% stiffness reduction at 3$^{rd}$ floor |
| D3 | Damaged | 15% stiffness reduction at 2$^{nd}$ and 3$^{rd}$ floors |

Table 6.2: Different Sensor Setups Employed for Experimental Validation.

| **Sensor Setup** | **Sensors** | $\frac{\mathbf{p}}{\mathbf{n}}$ |
|---|---|---|
| **S1** | 5, 6, 7, 8 | 1.0625 |
| **S2** | 1, 3, 5, 7 | 1.0625 |
| **S3** | 2, 4, 5, 7 | 1.0625 |
| **S4** | 3, 4, 7, 8 | 1.0625 |
| **S5** | all 8 sensors | 2.125 |

Table 6.3: Results in terms of Type I error (Type I error: error committed when declaring the structure damaged when it is instead undamaged).

| ***Sensor Setup*** | **State** | ***Method 1*** | ***Method 2*** | ***Method 3*** | ***Methods Combination*** |
|---|---|---|---|---|---|
| ***S1:***<br>$d$=48<br>$p$=68<br>$\lambda_3$=0.059<br>$\lambda_4$=0.033 | **U1** | 0.00% | 10.00% | 0.00% | 0.00% |
| | **U2** | 20.00% | 20.00% | 20.00% | 20.00% |
| | **U3** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **Type I** | **6.67**% | **10.00**% | **6.67**% | **6.67**% |
| ***S2:***<br>$d$=48<br>$p$=68<br>$\lambda_3$=0.059<br>$\lambda_4$=0.033 | **U1** | 10.00% | 10.00% | 0.00% | 10.00% |
| | **U2** | 10.00% | 10.00% | 10.00% | 10.00% |
| | **U3** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **Type I** | **6.67**% | **6.67**% | **3.33**% | **6.67**% |
| ***S3:***<br>$d$=48<br>$p$=68<br>$\lambda_3$=0.066<br>$\lambda_4$=0.036 | **U1** | 0.00% | 10.00% | 10.00% | 0.00% |
| | **U2** | 10.00% | 30.00% | 10.00% | 10.00% |
| | **U3** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **Type I** | **3.33**% | **13.33**% | **6.67**% | **3.33**% |
| ***S4:***<br>$d$=48<br>$p$=68<br>$\lambda_3$=0.060<br>$\lambda_4$=0.034 | **U1** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **U2** | 0.00% | 10.00% | 0.00% | 10.00% |
| | **U3** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **Type I** | **0.00**% | **3.33**% | **0.00**% | **3.33**% |
| ***S5:***<br>$d$=48<br>$p$=136<br>$\lambda_3$=0.058<br>$\lambda_4$=0.030 | **U1** | 0.00% | 10.00% | 0.00% | 0.00% |
| | **U2** | 0.00% | 10.00% | 30.00% | 0.00% |
| | **U3** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **Type I** | **0.00**% | **6.67**% | **10.00**% | **0.00**% |

Table 6.4: Results in terms of Type II error (Type II error: error committed when declaring the structure healthy when it is instead damaged).

| ***Sensor Setup*** | **State** | ***Method 1*** | ***Method 2*** | ***Method 3*** | ***Methods Combination*** |
|---|---|---|---|---|---|
| ***S1:***<br>$d$=48<br>$p$=68<br>$\lambda_3$=0.059<br>$\lambda_4$=0.033 | **D1** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **D2** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **D3** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **Type II** | **0.00%** | **0.00%** | **0.00%** | **0.00%** |
| ***S2:***<br>$d$=48<br>$p$=68<br>$\lambda_3$=0.059<br>$\lambda_4$=0.033 | **D1** | 20.00% | 13.33% | 0.00% | 0.00% |
| | **D2** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **D3** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **Type II** | **4.00%** | **2.67%** | **0.00%** | **0.00%** |
| ***S3:***<br>$d$=48<br>$p$=68<br>$\lambda_3$=0.066<br>$\lambda_4$=0.036 | **D1** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **D2** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **D3** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **Type II** | **0.00%** | **0.00%** | **0.00%** | **0.00%** |
| ***S4:***<br>$d$=48<br>$p$=68<br>$\lambda_3$=0.060<br>$\lambda_4$=0.034 | **D1** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **D2** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **D3** | 0.00% | 3.33% | 10.00% | 0.00% |
| | **Type II** | **0.00%** | **1.33%** | **4.00%** | **0.00%** |
| ***S5:***<br>$d$=48<br>$p$=136<br>$\lambda_3$=0.058<br>$\lambda_4$=0.030 | **D1** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **D2** | 0.00% | 0.00% | 0.00% | 0.00% |
| | **D3** | 3.33% | 0.00% | 0.00% | 0.00% |
| | **Type II** | **1.33%** | **0.00%** | **0.00%** | **0.00%** |

Table 6.5: Different states of the bridge deck model.

| **State** | **Condition** | **Description** |
|---|---|---|
| U1 | Undamaged | $k_i^{U1} = E[k_i] \forall i \in \{1, ..., 20\}$ |
| U2 | Undamaged | $k_i^{U2} = 0.99E[k_i] \forall i \in \{1, ..., 7\}$ |
| U3 | Undamaged | $k_i^{U3} = 1.01E[k_i] \forall i \in \{1, ..., 7\}$ |
| U4 | Undamaged | $k_i^{U4} = 0.99E[k_i] \forall i \in \{8, ..., 14\}$ |
| U5 | Undamaged | $k_i^{U5} = 1.01E[k_i] \forall i \in \{8, ..., 14\}$ |
| D1 | Damaged | $k_i^{D1} = 0.80k_6^{U1}$ |
| D2 | Damaged | $k_i^{D2} = 0.85k_6^{U2}$ |
| D3 | Damaged | $k_i^{D3} = 0.80k_{16}^{U3}$ |
| D4 | Damaged | $k_i^{D4} = 0.85k_{16}^{U4}$ |