

# **Model Updating Using a Quadratic Form**

by

**Pablo A. Tarazaga**

Thesis submitted to the Faculty of the  
Virginia Polytechnic Institute and State University  
in partial fulfillment of the requirements for the degree of

**Master of Science**

in

**Mechanical Engineering**

Daniel J. Inman, Chair

Yoram Halevi

Donald J. Leo

July 2004

Blacksburg, Virginia

Keywords: symmetry, connectivity, closed-form, robustness

Copyright by Pablo A. Tarazaga, 2004

# Model Updating Using a Quadratic Form

Pablo A. Tarazaga

## Abstract

The research presented in this thesis addresses the problem of updating an analytical model using a parametric Reference Basis approach. In this method, some parameters are assumed to be accurate (e.g. natural frequencies, mode shapes and mass matrix), while others are adjusted so that the eigenvalue equation is satisfied. Updating is done with the use of principal submatrices, and the method seeks the best parameters multiplying these matrices. This is a departure from classical model reference, and is closer to the formulation of sensitivity methods. The submatrices allow updating of the stiffness matrix with certain freedom while preserving connectivity. Closed form solution can be achieved through multiple ways; two different approaches, denoted as the Quadratic Compression Method (QCM) and the Full Vector Method (FVM), are described in this paper. It is shown that the QCM possesses superior robustness properties with respect to noise in the data. This fact, as well as the simplicity offered by QCM, is demonstrated theoretically and experimentally. The experiments are presented to show the advantage of the QCM in the updating process.

Esta tesis esta dedicada a mis abuelos:

Italia (Nonna) Garcia

y

Cristóbal Tarazaga

Gracias por tanto Amor.

# Acknowledgments

I would like to thank my advisor, Dr. Daniel J. Inman for his support, guidance and sense of humor throughout my graduate studies. Special thanks go to Dr. Yoram Halevi for his insight into model updating, without which I would not have gone to far. I would also like to thank Dr. Donald J. Leo for taking the time to be part of this research.

Special thanks go to my precious wife Vanessa Tarazaga for her unconditional love and support in helping me achieve my goals, thank you for being a part of my life. I also cannot forget my parents Pablo and Susana Tarazaga for their support, encouragement and sacrifices they have gone through to allow me to be where I am today, I love you both. And last but not least one of my best friends, my sister Silvana Tarazaga, words can not describe how special you are.

This research has been supported by NSF ® grant number CMS 0200019.

# Contents

<b>LIST OF TABLES .....</b>	<b>VII</b>
<b>LIST OF FIGURES .....</b>	<b>VIII</b>
<b>CHAPTER 1 INTRODUCTION .....</b>	<b>1</b>
<b>1.1 MOTIVATION .....</b>	<b>1</b>
<b>1.2 LITERATURE REVIEW .....</b>	<b>1</b>
<b>1.3 THESIS OVERVIEW.....</b>	<b>3</b>
<b><i>1.3.1 Contribution</i> .....</b>	<b>3</b>
<b><i>1.3.2 Approach</i> .....</b>	<b>3</b>
<b>CHAPTER 2 QUADRATIC COMPRESSION METHOD .....</b>	<b>5</b>
<b>2.1 INTRODUCTION .....</b>	<b>5</b>
<b>2.2 PROBLEM STATEMENT .....</b>	<b>5</b>
<b>2.3 STIFFNESS MATRIX UPDATING.....</b>	<b>7</b>
<b>2.4 CHAPTER SUMMARY .....</b>	<b>11</b>
<b>CHAPTER 3 NOISE PROPAGATION.....</b>	<b>12</b>
<b>3.1 INTRODUCTION .....</b>	<b>12</b>
<b>3.2 ANALYSIS .....</b>	<b>12</b>
<b>3.3 CHAPTER SUMMARY .....</b>	<b>16</b>
<b>CHAPTER 4 EXPERIMENTS.....</b>	<b>17</b>
<b>4.1 INTRODUCTION .....</b>	<b>17</b>
<b>4.2 COMPUTER SIMULATED EXPERIMENTS .....</b>	<b>17</b>
<b>4.3 EXPERIMENT .....</b>	<b>24</b>

4.3.1 <i>Experimental Setup</i> .....	24
4.3.2 <i>Experimental Results</i> .....	26
4.3.3 <i>Model Updating</i> .....	28
4.4 CHAPTER SUMMARY .....	35
CHAPTER 5 CONCLUSION.....	36
BIBLIOGRAPHY.....	39
APPENDIX A MATLAB CODE OF THE QUADRATIC COMPRESSION METHOD .....	41
APPENDIX B MATLAB CODE OF THE FULL VECTOR METHOD .....	43
APPENDIX C MATLAB CODE OF THE FINITE ELEMENT MODEL .....	45
APPENDIX D MATLAB PROGRAM USED TO THE RUN THE UPDATING METHODS FOR THE EXPERIMENT.....	47

# List of Tables

4.2.1	Updating results obtained for different noise levels.....	21
4.2.2	MAC number and deviations obtained for the QCM and FVM.....	21
4.3.1	Beam properties.....	25
4.3.2	Updating parameters used.....	32
4.3.3	Natural Frequencies.....	33
4.3.4	Error in percent of the different methods.....	33
4.3.5	MAC of the experimental and analytical mode shapes.....	34
4.3.6	MAC of the experimental and QCM updated mode shapes.....	34
4.3.7	The MAC of the experimental and FVM updated mode shapes.....	34

# List of Figures

4.2.1	Spring-mass system.....	19
4.2.2	Error results for the QCM and FVM.....	23
4.3.1	Cantilever beam setup for model testing.....	25
4.3.2	Schematics of experimental setup.....	26
4.3.3	Frequency response functions of the cantilever beam.....	27
4.3.4	Schematic of correct spring location.....	29

# Chapter 1

## Introduction

### 1.1 Motivation

Having an accurate model is essential in analyzing systems under various excitations, boundary conditions and parameter changes. Unfortunately, the analytical model made through any of various methods, deviates from the true model. This is evident in features such as natural frequencies and mode shapes that are extracted from the test results, and do not agree with the predicted values from the analytical model. These errors can stem from various reasons, such as uncertainties in the governing equations of the system, simplifying assumptions and imprecise boundary conditions. It should be noted that the test results themselves may contain errors resulting from measurement noise. Thus, the creation of a model that takes into account the results of both the analytical model and experimental test results is sought after.

### 1.2 Literature Review

There are many approaches that can be taken to perform model updating. Mottershead and Friswell have compiled in [17] a wide variety of model updating techniques for the general case of finite element modeling. Certain methods of model

updating, such as formal system identification [5], completely disregard the analytical model. Most approaches, however, do use the analytical model, as the eigendata obtained experimentally are not complete and cannot be the sole source of a model. For example the sensitivity method, where the dependence of the natural frequencies and mode shapes on the physical parameters is calculated either numerically or analytically [12]. Link has also made use of this technique using local and global parameters for updating with the use of numerical optimization programs [14]. Soft computing methods, such as the ones applied by Atalla and Inman, use frequency domain and neural network to estimate the updating parameters quickly and yield a model representative of the measured data [2]. Piranda et al in [13] applied simulated annealing and genetic algorithm techniques to the problem.

In standard Reference Basis (RB) methods [3, 4], certain parameters are assumed to be accurate. Those that are free should satisfy the system eigenvalue equation with minimal deviation from the analytical values. The advantage of the Reference Basis method is its mathematical and numerical convenience. It provides closed-form solutions, requiring only elementary matrix operations and inversion of low-dimension matrices. The main disadvantages are that, one, it does not account for any dependence of the matrices on physical properties, two, it cannot incorporate any prior knowledge or engineering consideration, and three, it cannot preserve the connectivity of the stiffness matrix. This was partially modified in the Generalized Reference Basis by Kenigsbuch and Halevi in [11], where freedom in selecting the weighting matrix was added. The importance of this more general approach was the ability to incorporate prior knowledge regarding the accuracy of the model in specified areas into the method. The simplicity-connectivity tradeoff can clearly be seen when the Reference Basis method is compared with approaches that minimize a similar criterion; yet, impose certain connectivity on the stiffness matrix, either before the optimization, as in the case of Hemez and Farhat in [10], or after it, as in the case of Halevi and Bucher in [9].

## **1.3 Thesis Overview**

### **1.3.1 Contribution**

The purpose of this research is to provide a different method for model updating that is capable of combining the advantages of previous model updating approaches, such as the Reference Basis and Sensitivity methods, into one method. Taking this approach, an alternative method was developed that is able to preserve connectivity and will provide the results in a simple closed form solution. This new approach, referred to as the Quadratic Compression Method (QCM), has also shown robustness with respect to noise in the data. Thus, this research has proven to be successful in the completion of a new method with added gains in versatility, ease of calculation and noise robustness.

### **1.3.2 Approach**

Model updating will be looked at through the parametric Reference Basis (RB) method. To this method the use of principal submatrices will be incorporated to help update the stiffness matrix. Prior knowledge of the system can be taken into account when designing the principal submatrices to allow for engineering considerations. This problem formulation combines the Reference Basis method and the standard setup for sensitivity methods. The problem has a closed form solution that can be achieved through the Quadratic Compression Method or the Full Vector Method (FVM). It will be shown that while the two methods yield the same results under ideal conditions, the QCM is far more robust with respect to measurement noise.

The material is presented as follows: The problem and discussion of the stiffness matrix updating with the use of principal submatrices is stated in Chapter 2. In Chapter 3, an error analysis of the measurement noise propagation of the two methods is provided. Chapter 4 goes into greater detail by performing some numerical

experiments, as well as a full experiment of a cantilever beam. Concluding remarks are then offered in Chapter 5.

## Chapter 2

# Quadratic Compression Method

### 2.1 Introduction

This approach begins by addressing the problem, as well as related concerns. Among some of these concerns are the issues of symmetry, orthogonalization and normalization of the mode shapes with respect to the mass matrix. Also discussed in this chapter are the parameters that will be used as references for the updating process. The main idea of this chapter is the development of the Quadratic Compression Method for the updating of the stiffness matrix and how it differs from the more commonly used Full Vector Method.

### 2.2 Problem Statement

The general statement that model updating attempts to solve is as follows: let the ‘true’ system be defined as

$$M_T \ddot{x}(t) + K_T x(t) = f(t) \quad x \in R^n, \quad (2.1)$$

where  $M_T$  is the true mass matrix,  $K_T$  is the true stiffness matrix,  $x(t)$  is the displacement vector,  $\ddot{x}(t)$  represents the second derivative with respect to time of the

displacement vector,  $f(t)$  is any external force to the system, and  $n$  represents the number of degrees of freedom. The absolute accurate system is non-linear, has infinite degrees of freedom, etc. Thus ‘true’, refers to a system that is accurate enough for all practical purposes. The true natural frequencies and mode shapes are defined by  $\omega_{Ti}$  and  $\phi_{Ti}$  respectively. The analytical system is given by

$$M_A \ddot{x}(t) + K_A x(t) = f(t) \quad x \in R^n, \quad (2.2)$$

where  $M_A$  is the analytical mass matrix,  $K_A$  is the analytical stiffness matrix and typically  $M_A \neq M_T$  and  $K_A \neq K_T$ . After obtaining the results of a modal test all that is left is a subset  $m < n$  of experimental natural frequencies  $\omega_{Ei}$  and experimental mode shapes  $\phi_{Ei}$ . The experimental results are arranged in the following matrix form

$$\Omega_E = \text{diag}(\omega_{Ei}^2) \in R^{m \times m} \quad (2.3)$$

$$\Phi_E = [\phi_{E1} \dots \phi_{Em}] \in R^{n \times m}. \quad (2.4)$$

The assumption that the experimental natural frequencies are correct is made. In other words,

$$\Omega_E = \Omega_T = \Omega. \quad (2.5)$$

The physical justification for this is that they are global variables, common to all measurements in the system, and the errors in extracting them are relatively small [2]. With the previous in mind, the model updating problem tries to combine the analytical information  $(M_A, K_A)$  and the experimental results  $(\Omega_E, \Phi_E)$  to obtain a model that is closer to  $(M_T, K_T)$ . In the sequel it is also assumed that the mass matrix is accurate, i.e.  $M_A = M_T = M$ , and only the stiffness matrix is updated. The extension to the case of both mass and stiffness updating is left for future considerations.

## 2.3 Stiffness Matrix Updating

The updating method discussed in this paper is a combination of the Reference Basis and parametric methods. To perform updating of the stiffness matrix, three parameters have been chosen to be accurate, similar to RB; which in this case, will be our Reference Basis. The first one is the matrix of natural frequencies squared  $\Omega$ , the second is the mass matrix  $M$ , and third, the matrix of mode shapes  $\Phi_E$ , after it has been normalized and orthogonalized with respect to  $M$ , i.e.  $\Phi_E = \Phi$  satisfying

$$\Phi^T M \Phi = I. \quad (2.6)$$

The first two assumptions regarding the natural frequencies and the mass matrix were discussed in the previous section. The assumption concerning the mode shapes is not completely accurate and errors in this matrix will propagate in the updating of the stiffness matrix. This topic is discussed in further detail in Chapter 3. Now that the references have been established, updating of the stiffness matrix can proceed. The key element of the Reference Basis method is satisfying the eigenvalue equation in matrix form:

$$K \Phi = M \Phi \Omega. \quad (2.7)$$

Notice that only  $K$  can be changed after the assumption that  $M$ ,  $\Omega$  and  $\Phi$  are accurate and therefore fixed. The standard Reference Basis method [3,4,11] looks for the unstructured  $K$  whose deviation from the  $K_A$  is minimal. However, because it is unstructured, the solution does not preserve the existing connectivity in the structure. Instead, the process of updating the stiffness matrix will be done with the aid of principal submatrices as is customary in sensitivity methods [12,14] and was introduced to Reference Basis in [9,10]. The equation

$$K_u = K_A + \sum_{k=1}^p \sigma_k P_k, \quad (2.8)$$

will be used to update the analytical stiffness matrix, where  $P_k$  represents the principal submatrices, defined with respect to the nature and structure of the stiffness matrix that is being updated, and  $\sigma_k$  represents the optimization parameters. Thus, by determining the structure of  $P_k$ , an appropriate updating pattern can be generated that will match the form of the analytical stiffness matrix. The principal submatrices are basically placeholder matrices with components that are placed strategically in order to update different parts of the stiffness matrix. Since these matrices are sparse, the problem can be isolated to any particular part of the physical system, and restrict certain areas from being updated.  $K_u$  will be symmetric and will preserve any connectivity existing in the system. The form (2.8) also makes the relationship between the updating parameters,  $\sigma_k$ , and the physical system easier to relate.

The Sensitivity Method makes use of equation (2.8) as well. This approach [14] uses this parameterization to generate the objective function

$$\min J = \Delta Z^T W \Delta Z + s^T W_s s, \quad (2.9)$$

where  $W, W_s$  are weighting matrices,  $\Delta Z$  are the residual vectors between the experimental and corresponding analytical values of the frequencies and mode shapes, and  $s$  is the vector of correction parameters. As it can be seen this approach tries to minimize the error between the experimental and analytical frequencies and mode shapes, while the approach presented in continuance tries to minimize the error in the eigenvalue equation.

Substituting equation (2.8) into equation (2.7),

$$\left( K_A + \sum_{k=1}^p \sigma_k P_k \right) \Phi = M \Phi \Omega \quad (2.10)$$

is obtained. Expanding equation (2.10) and placing all the constant terms to the right side results in

$$\sum_{k=1}^p \sigma_k P_k \Phi = M \Phi \Omega - K_A \Phi. \quad (2.11)$$

This is a linear equation in the vector  $\sigma \in R^{p \times 1}$ , whose elements are the  $\sigma_k$ . After some rearrangement, equation (2.11) can be expressed in the following way

$$A\sigma = r, \quad A \in R^{nm \times p}, \quad r \in R^{nm \times 1}. \quad (2.12)$$

Seemingly, equation (2.12) provides  $nm$  equations, however, due to the symmetry of  $P_k$ , the number of independent equations is only  $nm - m(m-1)/2$ . In some approaches, equation (2.12) is an actual step of the algorithm while in others it is only a conceptual step. In any event, the solution is given by (or can be thought of as given by)

$$\sigma = A^L r, \quad (2.13)$$

where  $A^L$  is any generalized left inverse of  $A$ . In the case of perfectly accurate measured mode shapes, the same results will be obtained for every choice of  $A^L$ . If the mode shapes contain noise this is not necessarily so.

Two alternatives are presented to solve for the optimization parameters. The first one, which will be referred to as the Full Vector Method (FVM), uses the format in equations (2.12)-(2.13) directly, with

$$A = [Vec(P_1\Phi) \dots Vec(P_p\Phi)] \quad , \quad r = Vec(M\Phi\Omega - K_A\Phi) \quad (2.14)$$

where  $Vec$  stands for the operation of stacking the columns of a matrix into a single vector. The left inverse is the Penrose-Moore generalized inverse [18] and is associated with Least Squares. The solution is given in this case by

$$\sigma_{FVM} = (A^T A)^{-1} A^T r. \quad (2.15)$$

A different approach, which will be referred as the Quadratic Compression Method (QCM), can be taken to solve equation (2.11). It is proposed to multiply equation (2.11) on the left by  $\Phi^T$  to obtain

$$\sum_{k=1}^p \sigma_k \Phi^T P_k \Phi = \Phi^T M \Phi \Omega - \Phi^T K_A \Phi = \Omega - \Phi^T K_A \Phi. \quad (2.16)$$

Following the same procedure as before

$$G_p \sigma = b \quad (2.17)$$

is obtained, where

$$G_p = [Vec(\Phi^T P_1 \Phi) \dots Vec(\Phi^T P_p \Phi)] \quad , \quad b = Vec(\Omega - \Phi^T K_A \Phi) \quad (2.18)$$

This is again a linear equation in  $\sigma$ , but its dimension is smaller.  $G_p$  has dimensions  $(m^2 \times p)$ ,  $\sigma$  has dimensions  $(p \times 1)$ , and  $b$  has dimensions  $(m^2 \times 1)$ . The system described in equation (2.16) provides  $m^2$  equations. However, due to the symmetry of  $P_k$  the number of independent equations is  $d = m(m+1)/2$ . Equation (2.17), after deleting the dependent equations, can encompass three main forms for which closed form solution can be obtained. If

1.  $p < d$ , the system is over-determined and can be solved through Least Squares by:

$$\sigma = (G_p^T G_p)^{-1} G_p^T b. \quad (2.19)$$

With previous knowledge of the system the possibilities of using Weighted Least Squares can be implemented.

2.  $p = d$ , the system is square and has a unique solution.
3.  $p > d$ , the system is under-determined and can be solved using the Karush Kuhn - Tucker condition [6]. The Karush Kuhn Tucker conditions have a closed form solution for

$$\min f(\sigma) = \|K_u - K_A\|_F^2 = \left\| \sum_{k=1}^p \sigma_k P_k \right\|_F^2 \quad (2.20)$$

subject to:

$$G_p \sigma = b,$$

where  $\| \cdot \|_F$  is the Frobenious Norm. If  $f(\sigma)$  is convex then  $\sigma^*$  is the minimizer if and only if

$$\begin{aligned} G_p \sigma^* &= b \\ \nabla f(\sigma^*) &= G_p^T \delta^* \end{aligned} \quad (2.21)$$

where  $\nabla f(\sigma)$  is the gradient of  $f(\sigma)$  and  $\delta^*$  is a vector of Lagrange multipliers. In this case  $f(\sigma)$  is clearly convex. After restructuring this system we get

$$\begin{bmatrix} G_p & 0 \\ \nabla f(\sigma) & -G_p^T \end{bmatrix} \begin{bmatrix} \sigma^* \\ \delta^* \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}. \quad (2.22)$$

This system is square and linear and can be solved for the unique solution.

The same approach can be taken to solve equation (2.12) of the FVM, although the system will be predominantly over-determined and will be solved through least squares.

## 2.4 Chapter Summary

In this chapter, the development of the Quadratic Compression Method was presented. This new approach provides user intervention by using the ability to define the principal submatrices and preserves connectivity in the updated stiffness matrix. In addition, a solution for this new method was obtained in a closed-form for any possible condition that the system may encounter (i.e. over-determined, under-determined, and square). Usually, the system will be over determined and a Least Squares solution is used to solve for the updating parameters. A not so visible fact is that QCM uses a smaller set of equation to solve for the same updating parameters, making it numerically more stable when larger degrees of freedom are considered. Also discussed, was the more common approach referred to in this chapter, and for the rest of this thesis, as the Full Vector Method. In the next chapter, comparisons of how these two methods update when noise affects the collected data, will be presented.

## Chapter 3

# Noise Propagation

### 3.1 Introduction

In the previous chapter, two methods of calculating the updating parameter were shown. In the ideal case of perfect measurement they yield the same solution. However, inaccuracies in model updating due to noise in the data are inevitable. Therefore, identification of a method that shows more robustness with respect to noise is of great interest to model updating. Hence, this chapter compares the two methods in terms of their ability to update in the presence of noise.

### 3.2 Analysis

The analysis of how the error propagates is rather complex. Therefore, an explanation will be given in this section only for the case where one mode shape is measured and one updating parameter and principal submatrix is used. These assumptions simplify the analysis and enable tracking of the error through all the steps of the algorithm. Hence the following assumptions are made:

1. One measured mode, i.e.  $\Phi \in R^{n \times 1}$  is a vector.

2. One parameter  $\sigma$  and principal submatrix  $P \in R^{n \times n}$ .
3.  $\sigma$  represents the only parameter that changes in the structure. Hence, with the true value  $\sigma_0$ ,

$$K_T = K_A + \sigma_0 P. \quad (3.1)$$

Equation (2.11) in this case reduces to

$$P\Phi\sigma = -K_A\Phi + M\Phi\omega^2. \quad (3.2)$$

Because of the assumption that  $\Phi$  is a vector, there is no need for vectorization of the equation since (2.11) and (2.12) coincide in this case. In terms of (2.12)

$$A = P\Phi, \quad r = (M\omega^2 - K_A)\Phi. \quad (3.3)$$

The FVM solution is based on the generalized inverse of  $A$ , and is given in equation (2.14) as  $\sigma_{FVM} = (A^T A)^{-1} A^T r$ , or specifically

$$\sigma_{FVM} = (\Phi^T P^2 \Phi)^{-1} \Phi^T P (M\omega^2 - K_A) \Phi. \quad (3.4)$$

The QCM solution, on the other hand, is given in this case by

$$\sigma_{QCM} = (\Phi^T P \Phi)^{-1} (\omega^2 - \Phi^T K_A \Phi). \quad (3.5)$$

In the absence of noise, i.e.  $\Phi = \Phi_T$ , and recalling assumption 3, direct substitution of (3.1) verifies that both methods result in the same accurate parameter, i.e.

$$\sigma_{FVM} = \sigma_{QCM} = \sigma_0. \quad (3.6)$$

In a practical application, the mode shapes always contains some error due to noise. This can be expressed in the following manner

$$\bar{\Phi}_E = \Phi_T + \bar{e}, \quad \bar{e} \in R^{n \times 1}, \quad (3.7)$$

where  $\bar{e}$  is noise. As explained in the previous section, the first step of the algorithm is normalization and orthogonalization of  $\bar{\Phi}_E$ . To investigate the meaning of this step, it will first be looked at in the general case, and then, in the single vector case. After orthogonalization and normalization, the measured  $\bar{\Phi}_E$  is replaced by

$$\Phi_E = \Phi_T + e \quad (3.8)$$

Now

$$\begin{aligned}
I &= \Phi_E^T M \Phi_E \\
&= (\Phi_T + e)^T M (\Phi_T + e) \\
&= \Phi_T^T M \Phi_T + e^T M \Phi_T + \Phi_T^T M e + e^T M e \\
&= I + e^T M \Phi_T + \Phi_T^T M e + e^T M e .
\end{aligned} \tag{3.9}$$

Assuming that the error is small, the quadratic term in the last equation can be neglected and thus, results in

$$e^T M \Phi_T + \Phi_T^T M e = 0 . \tag{3.10}$$

This means that the diagonal of  $e^T M \Phi_T$  is zero, which implies that  $e_i$  does not have a component in the direction of  $\Phi_{T_i}$ . In the case of a single mode shape the conclusion is that

$$e^T M \Phi_T = 0 . \tag{3.11}$$

Taking the eigenvalue equation and multiplying on the left by  $e^T$  results in

$$e^T K_T \Phi = e^T M_T \Phi_T \Omega_T . \tag{3.12}$$

Due to (3.11) the right side is equal to zero and

$$e^T K_T \Phi_T = 0 \tag{3.13}$$

is also obtained.

Returning now to computing  $\sigma_{FVM}$  and  $\sigma_{QCM}$ , when the measured modes are used, equations (3.4) and (3.5) become

$$\sigma_{FVM} = [(\Phi_T + e)^T P^2 (\Phi_T + e)]^{-1} (\Phi_T + e)^T P (M \omega^2 - K_A) (\Phi_T + e) \tag{3.14}$$

$$\sigma_{QCM} = [(\Phi_T + e)^T P (\Phi_T + e)]^{-1} [\omega^2 - (\Phi_T + e)^T K_A (\Phi_T + e)] . \tag{3.15}$$

Calculation of how the error in  $\Phi$ , propagates to an error in  $\sigma$ , allow for comparison of the two methods. A formula, which is useful in these derivations, is the derivative of the inverse of a square matrix [19], shown in equation (3.16)

$$d(X^{-1}) = -X^{-1} (dX) X^{-1} . \tag{3.16}$$

Using this formula and assuming small  $e$  so that quadratic terms can be neglected, results in

$$\begin{aligned}
\sigma_{FVM} - \sigma_0 &= -2(\Phi_T^T P^2 \Phi_T)^{-2} e^T P^2 \Phi_T \Phi_T^T P (M\omega^2 - K_A) \Phi_T + (\Phi_T^T P^2 \Phi_T)^{-1} [e^T P (M\omega^2 - K_A) \Phi_T + \Phi_T^T P (M\omega^2 - K_A) e] \\
&= (\Phi_T^T P^2 \Phi_T)^{-1} [-\sigma_0 e^T P^2 \Phi_T - \sigma_0 e^T P^2 e + e^T P (M\omega^2 - K_A) \Phi_T + \Phi_T^T P (M\omega^2 - K_A) e] \\
&= (\Phi_T^T P^2 \Phi_T)^{-1} [e^T P (M\omega^2 - K_A - \sigma_0 P) \Phi_T + \Phi_T^T P (M\omega^2 - K_A - \sigma_0 P) e] \\
&= (\Phi_T^T P^2 \Phi_T)^{-1} [e^T P (M\omega^2 - K_T) \Phi_T + \Phi_T^T P (M\omega^2 - K_T) e] \\
&= (\Phi_T^T P^2 \Phi_T)^{-1} \Phi_T^T P (M\omega^2 - K_T) e .
\end{aligned} \tag{3.17}$$

In this derivation, the definition of  $\sigma_0$  was used in the sense that a scalar is the transpose of itself, and of assumption 3. The important conclusion from (3.17), is that the error in  $\sigma_{FVM}$  is linear in the error in  $\Phi$ . Applying the same procedure to  $\sigma_{QCM}$  yields,

$$\begin{aligned}
\sigma_{QCM} - \sigma_0 &= -2(\Phi_T^T P \Phi_T)^{-2} \Phi_T^T P e (\omega^2 - \Phi_T^T K_A \Phi_T) - 2(\Phi_T^T P \Phi_T)^{-1} \Phi_T^T K_A e \\
&= -2(\Phi_T^T P \Phi_T)^{-1} \Phi_T^T (P \sigma_0 e + K_A e) \\
&= -2(\Phi_T^T P \Phi_T)^{-1} \Phi_T^T (K_A + P \sigma_0) e \\
&= -2(\Phi_T^T P \Phi_T)^{-1} \Phi_T^T K_T e = 0 .
\end{aligned} \tag{3.18}$$

In this case, the linear term is zero and the dominant term is quadratic in the error  $e$ . As a result, for relatively small levels of noise, the error is much smaller in  $\sigma_{QCM}$  than it is in  $\sigma_{FVM}$ . A less fortunate consequence of this result is that, using statistical terms, the estimation of  $\sigma_{QCM}$  is biased. However, the gain in the reduced variance is more meaningful than the loss in bias, as is demonstrated in the next chapter.

The difference between the two methods can be explained by examining equation (2.12) once more. However, this time written with the noise effect

$$[A_0 + \Delta A(e)]\sigma = r + \Delta r(e) . \tag{3.19}$$

It is well known that the Least Squares solution is optimal when the  $A$  matrix is accurate and only the right-hand side is subjected to noise. However, in this case, the  $A$  matrix is stochastic as well. QCM provides in this case a better estimator of  $\sigma$ .

### 3.3 Chapter Summary

In this chapter we have presented the analysis for noise propagation for both QCM and FVM methods. The simple case, where only one mode shape is measured and one updating parameters is used, was presented. It has been shown that the error in FVM is linear and the error in QCM is quadratic in  $e$ . This means that for relatively small levels of noise, the error is much smaller for QCM. The same idea can be expected for multiple updating parameters and multiple mode shapes.

As is known, the accuracy of the mode shapes and frequencies is critical in any updating method, therefore, any improvement in robustness is translated immediately to more accurate updated models. Taking these facts into account, QCM provides great advantages in the area of model updating. As explained in Chapter 2, its advantages in providing a closed-form solution that maintains the symmetry and connectivity of the stiffness matrix and now the robustness to noise add for a great rounded method in the area of model updating. The next chapter will demonstrate these advantages through the use of some experiments.

## Chapter 4

# Experiments

### **4.1 Introduction**

In this chapter, examples will be shown to demonstrate the applicability of the suggested method. Also the results of this new method will be compared with the more general approach of FVM. The first section will exhibit some computer simulated experiments. These examples show in greater detail the advantages of QCM over FVM. The following section discusses the process of an experiment of a cantilever beam and the model updating results.

### **4.2 Computer Simulated Experiments**

By a computer simulated experiment, it is meant that a set of two models that are generated analytically with a difference in the stiffness matrix. For example, in the case of a spring-mass system, a difference in one of the modeled springs could simulate a wrong calculation of one of the springs used in a system. This way, one model is assigned to be the true or “experimental” model and only its frequencies and mode shapes are retrieved and used to update the analytical model. This kind of computer simulated experiments are able to provide fast results and insight into the

method used for updating. In continuation, there are three examples that use this kind of computer simulated example to show the advantages of QCM.

**Example 1:** Using QCM for a spring mass system.

Consider the system shown in Figure 4.2.1, whose model in matrix form is

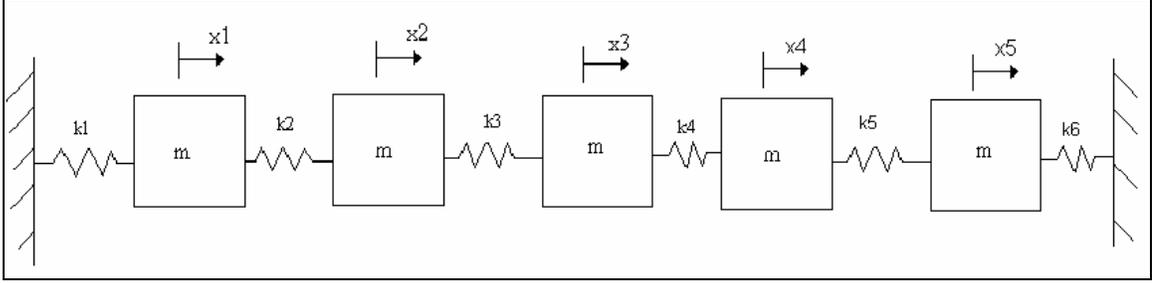
$$\begin{bmatrix} m_1 & 0 & 0 & 0 & 0 \\ 0 & m_2 & 0 & 0 & 0 \\ 0 & 0 & m_3 & 0 & 0 \\ 0 & 0 & 0 & m_4 & 0 \\ 0 & 0 & 0 & 0 & m_5 \end{bmatrix} \ddot{x}(t) + \begin{bmatrix} k_1 + k_2 & -k_2 & -k_{10} & 0 & 0 \\ -k_2 & k_2 + k_3 & -k_3 & 0 & 0 \\ -k_{10} & -k_3 & k_3 + k_4 & -k_4 & 0 \\ 0 & 0 & -k_4 & k_4 + k_5 & -k_5 \\ 0 & 0 & 0 & -k_5 & k_5 + k_6 \end{bmatrix} x(t) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (4.1)$$

The values for the true system and the analytic model are  $m_i = 10$ ,  $i = 1 : 5$  (common to both), and

$$k_{Ti} = [50 \quad 70 \quad 50 \quad 50 \quad 50 \quad 50] \quad , \quad k_{Ai} = [50 \quad 50 \quad 50 \quad 50 \quad 50 \quad 50]$$

It can be clearly seen that the second component of the spring-constant vector is incorrect in the analytical model. Thus, the top left corner of the analytical stiffness matrix will change. This error will propagate to the analytical frequencies and mode shapes. Assuming that only a subset of  $m = 2$  (out of  $n = 5$ ) could be obtained, the corresponding natural frequencies and mode shapes extracted “experimentally” and those of the analytical model, would be

$$\Omega = \begin{bmatrix} 1.4086 & 0 \\ 0 & 5.0000 \end{bmatrix}, \Phi = \begin{bmatrix} 0.1004 & 0.1581 \\ 0.1519 & 0.1581 \\ 0.1812 & 0.0000 \\ 0.1594 & -0.1581 \\ 0.0928 & -0.1581 \end{bmatrix} \quad \Omega_A = \begin{bmatrix} 1.3397 & 0 \\ 0 & 5.0000 \end{bmatrix}, \Phi_A = \begin{bmatrix} 0.0913 & -0.1581 \\ 0.1581 & -0.1581 \\ 0.1826 & 0.0000 \\ 0.1581 & 0.1581 \\ 0.0913 & 0.1581 \end{bmatrix}$$



**Figure 4.2.1:** Spring-mass system.

Referring to the stiffness matrix of equation (4.1) it can be seen that the generation of a principal matrix for every spring can be postulated, although only one is needed to alter the spring constant  $k_2$  where the error is located. Two principal submatrices will be created to show that the method will only alter the value of the spring constant that has the error. With a slightly different notation, the principal submatrices for a spring-mass system are defined as

$$(P_{ij})_{tq} = \begin{cases} 1 & i = q = t \text{ or } j = t = q \\ -1 & (t, q) = (i, j) \text{ or } (j, i) \\ 0 & \text{other wise} \end{cases}, \quad (4.2)$$

where  $t$  and  $q$  represent the positions in the principal submatrices. The index  $k$  could be any combination with  $i = 1, \dots, n$ ,  $j = 1, \dots, n$  and  $i \leq j$  meant to match the springs modeled. In our case we will define  $i = 1, j = 2$  and  $i = 2, j = 3$ . The results of equation (4.2) for this particular case are shown as

$$P_{1,2} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad P_{2,3} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Referring back to equation (4.1) and examining the stiffness matrix, it can be seen that the matrices  $P_{1,2}$  and  $P_{2,3}$  represent the positions of springs 2 and 3. Thus, these

matrices should be chosen in such a way that they are symmetric and alter only the specified locations. As mentioned in Chapter 2, they should follow the nature and structure of the stiffness matrix. In this case, the spring locations  $k_2$  and  $k_3$  were chosen. It is known in this example that the error in the analytical model is only located in the position of the 2<sup>nd</sup> spring, but the locations of the 3<sup>rd</sup> spring was also added as a test to see if the method updates the correct positions only.

Since  $p = 2$  and  $d = 3$  the system in equation (2.17) can be solved using equation (2.19); obtaining

$$\sigma = [\sigma_{1,2} \quad \sigma_{2,3}]^T = [20 \quad 0]^T .$$

These results show that the method has used only the first parameter, which corresponds to the position of the spring constant  $k_2$ , to update the analytical stiffness matrix. Thus, this simple example proves that the method will update the correct spring location 2 and not alter spring location 3 while trying to find the updating parameters.

**Example 2:** Comparing QCM and FVM for inaccurate measurements.

For this next example, noise will be incorporated to the data in Example 1. The noise was added as a percent of the magnitude of the components of  $\Phi$  (given in the tables as ‘Noise’) multiplying an  $N(0,1)$  random variable. The FVM and the QCM shown on equation (2.15) and (2.19), respectively, will be compared in the following areas: The Frobenius norm between the true stiffness matrix and the updated stiffness matrix, the updating parameters  $\sigma_{1,2}$  and  $\sigma_{2,3}$  and the diagonal of the MAC matrix for different noise levels. The Frobenius norm of the differences between the true stiffness matrix and the updated stiffness matrix is defined as

$$Kdif = \|K_T - K_u\|_F . \quad (4.3)$$

The MAC [1] is defined as

$$MAC(i, j) = \frac{|\phi_{Ei}^* \phi_{Aj}|^2}{\phi_{Aj}^* \phi_{Aj} \phi_{Ei}^* \phi_{Ei}}, \quad (4.4)$$

where  $\phi_{Ei}$  and  $\phi_{Aj}$  represent the  $i^{\text{th}}$  experimental and the  $j^{\text{th}}$  analytical mode shapes respectively. Thus, if the experimental and analytical mode shapes are well correlated, it is expected that when  $i = j$  a value of unity and a value of zero for  $i \neq j$ . These results are summarized in Table 4.2.1 and Table 4.2.2. The values presented in these tables are the results of 100,000 simulations, with converged mean and standard deviation (SD) of the results.

**Table 4.2.1:** Updating results obtained for different noise levels.

Noise		Kdif QCM	Kdif FVM	$\sigma_{1,2}$ QCM	$\sigma_{1,2}$ FVM	$\sigma_{2,3}$ QCM	$\sigma_{2,3}$ FVM
0.001	Mean	0.0127	0.4159	19.9984	20.0000	-0.0001	0.0002
	SD	0.0085	0.2915	0.0030	0.2445	0.0074	0.0523
0.01	Mean	0.3998	4.1550	19.8444	20.0208	-0.0121	0.0008
	SD	0.3095	2.9102	0.1836	2.4419	0.0753	0.5229
0.02	Mean	1.5008	8.3004	19.3771	20.0871	-0.0492	0.0045
	SD	1.2898	5.8598	0.7424	4.8927	0.1553	1.0450
0.03	Mean	3.3641	12.4473	18.5829	20.2036	-0.1110	-0.0023
	SD	3.0976	8.9502	1.7525	7.3843	0.2439	1.5619

**Table 4.2.2:** MAC number and deviations obtained for the QCM and FVM.

Noise		Mac True and Updated QCM					Mac True and Updated FVM				
0.001	Mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	SD	1.09E-09	2.29E-09	5.04E-09	4.10E-09	7.35E-09	8.00E-08	1.10E-07	1.28E-06	1.29E-05	1.29E-05
0.01	Mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9999	0.9991	0.9991
	SD	1.20E-07	2.40E-07	2.12E-06	1.91E-05	1.93E-05	6.00E-06	1.00E-06	1.00E-04	1.00E-04	1.30E-03
0.02	Mean	1.0000	1.0000	1.0000	0.9998	0.9998	1.0000	1.0000	0.9996	0.9964	0.9964
	SD	2.00E-06	1.10E-06	3.88E-05	3.92E-04	3.92E-04	2.70E-05	4.50E-05	5.00E-03	5.30E-03	5.20E-03
0.03	Mean	1.0000	1.0000	0.9999	0.9990	0.9990	0.9999	0.9999	0.9991	0.9921	0.9921
	SD	1.80E-05	2.60E-06	4.00E-04	3.10E-03	3.10E-03	1.00E-04	1.00E-04	1.20E-03	1.18E-02	1.16E-02

It can be seen that the solutions of the QCM have some bias. This is due to the fact that the error is quadratic and creates a shift in the mean. However, the bias is small compared to the difference in the standard deviation. It is clear from these results that the QCM is superior to the FVM as it shows less sensitivity to noise in the data.

To directly compare between the two methods, the error between the true values of the perturbation was selected, as well as the value that was actually obtained for each of 100,000 simulation runs, to be the single quality indicator. This is shown in equation (4.5).

$$Error = (\sigma_{1,2} - 20)^2 + (\sigma_{2,3} - 0)^2 \quad (4.5)$$

The QCM proved to have a smaller error than FVM 92% of the time for a noise level of 0.03 and 99.7% for a noise level of 0.001.

**Example 3.** Verification of the results of Chapter 3.

In this example, only one experimental mode shape and one frequency will be used. The same system shown in Figure 4.2.1 will be used for this example. The values of the true system and the analytical models are given by

$$k_{Ti} = [50 \ 50 \ 50 \ 50 \ 50 \ 50] \quad , \quad k_{Ai} = [70 \ 50 \ 50 \ 50 \ 50 \ 50]$$

It can be seen that in this case the error in the analytical model lies in the first spring constant. Since the stiffness matrix has the 1<sup>st</sup> spring in the position (1,1) the principal submatrix has been defined using equation (4.2) for  $i=1$  and  $j=1$ , which results in

$$P_{1,1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} .$$

This way, only the first position where the error is suspected, can be modified.

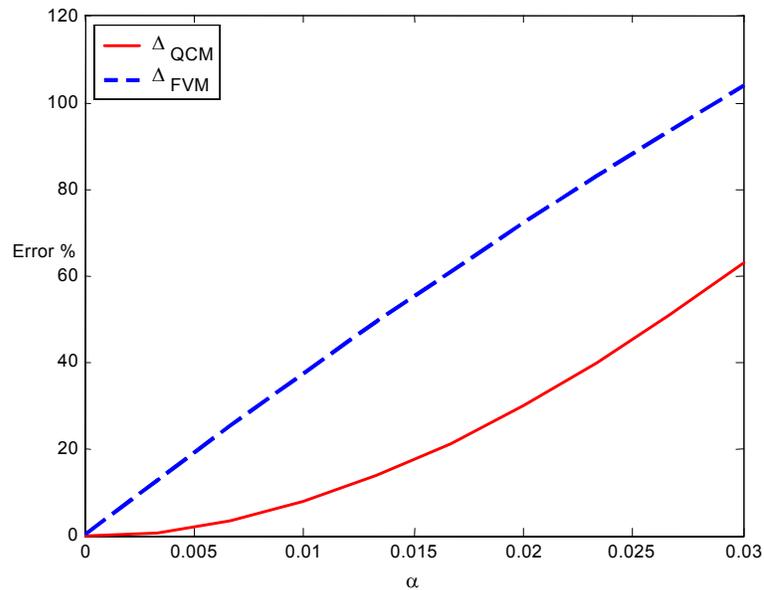
Since only one mode shape is used,  $\Phi$  is a vector in  $R^{5 \times 1}$ . Noise to the mode shape will be added using the equation

$$\Phi_E = \Phi + e\alpha, \quad (4.6)$$

where  $e$  is a fixed vector in  $R^{5 \times 1}$  and  $\alpha$  is a parameter; used to control the level of noise implemented ( $\alpha$  is not the same as the noise level in Example 2 and its values cannot be regarded as percentage standard deviation). The error due to noise of the two methods are defined as

$$\Delta_{QCM}(\alpha) = \frac{\sigma_0 - \sigma_{QCM}}{\sigma_0}, \quad \Delta_{FVM}(\alpha) = \frac{\sigma_0 - \sigma_{FVM}}{\sigma_0}, \quad (4.7)$$

where  $\sigma_0$  is the true value. Calculating these errors for various values of  $\alpha$  Figure 4.2.2 is obtained. It can be seen that QCM maintains a lower error than that of the FVM. Furthermore, it is evident from Figure 4.2.2 that the QCM shows a quadratic increase in error that stays below the linear increase of the FVM; as was explained in Chapter 3. The QCM error may exceed the linear error of the FVM at some noise level; however, by then the noise in the system provides such a large disturbance it makes the data useless. Figure 4.2.2 is another demonstration of the QCM being more insensitive to noise than the FVM.



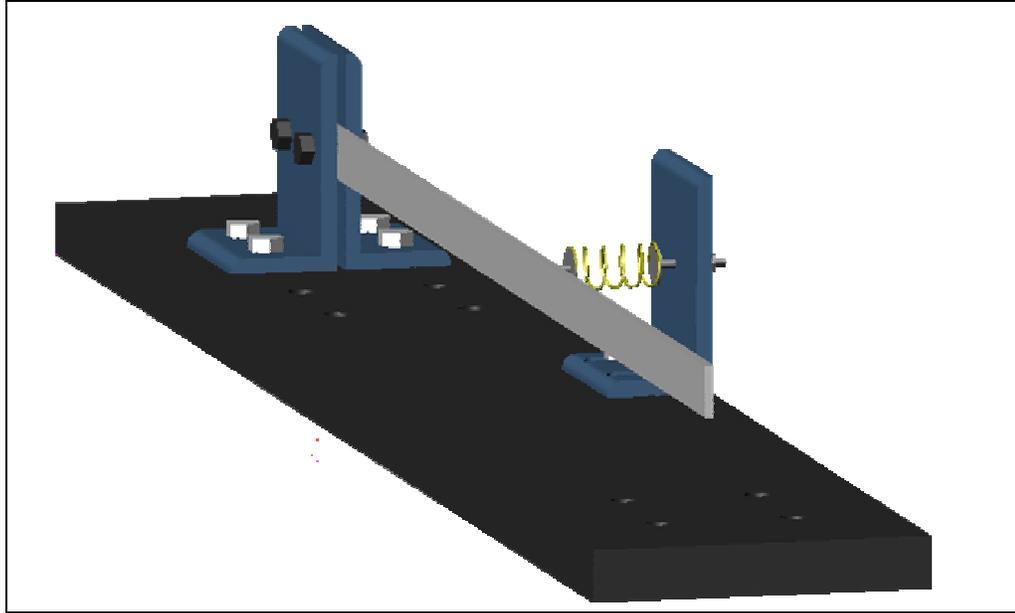
**Figure 4.2.2:** Error results for the QCM and FVM.

## **4.3 Experiment**

Although computational experiments give an idea of how things work, and provide fast insight, there is no substitute for an actual experiment. In this section a modal test was conducted on a cantilever beam. The data that was collected was used to perform model updating. Further details will be explained in the following sections.

### **4.3.1 Experimental Setup**

In order to corroborate this new approach for model updating, modal testing on the cantilever beam with attached spring shown in Figure 4.3.1 was conducted. During the experiment, the beam is clamped down at one end, while free at the other. The clamps are bolted down to a steel base that is heavy enough to simulate ground. The properties of the aluminum beam that was used are summarized in Table 4.3.1. The test setup also included a spring that was connected to the beam and was grounded to the steel base. The spring is located 25.4 cm from the clamped end of the cantilever beam and the spring constant is 3.34 N/mm.



**Figure 4.3.1:** Cantilever beam setup for model testing.

**Table 4.3.1:** Beam Properties.

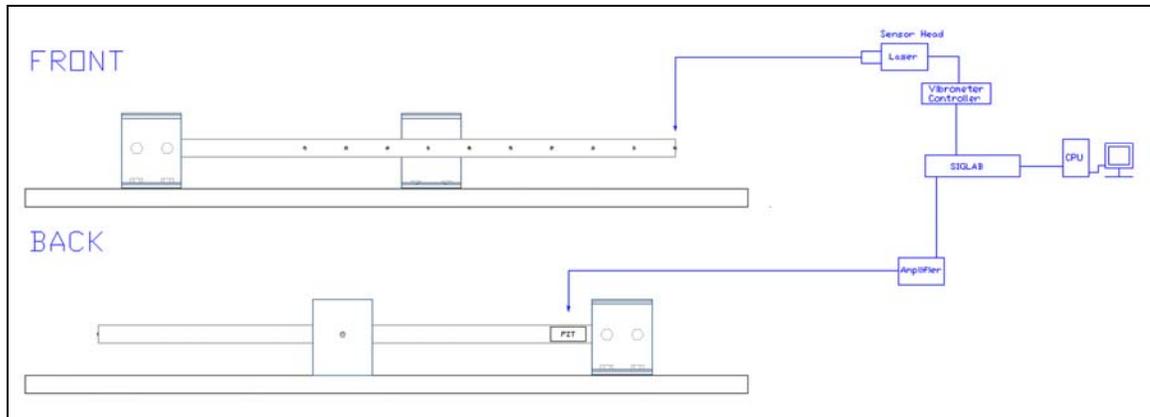
	Beam
Length (m)	0.5307
Width (m)	1.91E-02
Thickness (m)	3.17E-03
Area (m <sup>2</sup> )	6.04E-05
Moment of Inertia (m <sup>4</sup> )	5.05698E-11
Modulus Of Elasticity (N/m <sup>2</sup> )	7.10E+10
Density (kg/m <sup>3</sup> )	2.70E+03

The schematics of the experiment can be seen in Figure 4.3.2. To excite the system a PZT was used (model PSI-5H-S4-ENH). The size of the PZT patch was 3.81 cm long by 1.9 cm wide and .2667 mm thick. The PZT was connected to an *AVC* instrumentation amplifier (model 790A01) to adjust the voltage sent to the PZT patch. The PZT patch was neglected in the model used for the cantilever beam. Although its mass and stiffness will contribute to the dynamics of the beam it can be expected that this decision will not affect the results too much. Later this could be considered to see its effect.

The measurements were taken with a laser vibrometer, which is composed of a sensor head and a vibrometer controller (model OFV 303 and OFV 3001 respectively with

resolution 1000mm/s/V). The beam was divided into 12 section generating 13 points of measurements. One point is eliminated because it is the zero displacement due to the clamped end. The two closest points to the clamped section were also eliminated, because of interference with the location of the PZT patch. Thus, 10 points were measured in total, as shown in Figure 4.3.2.

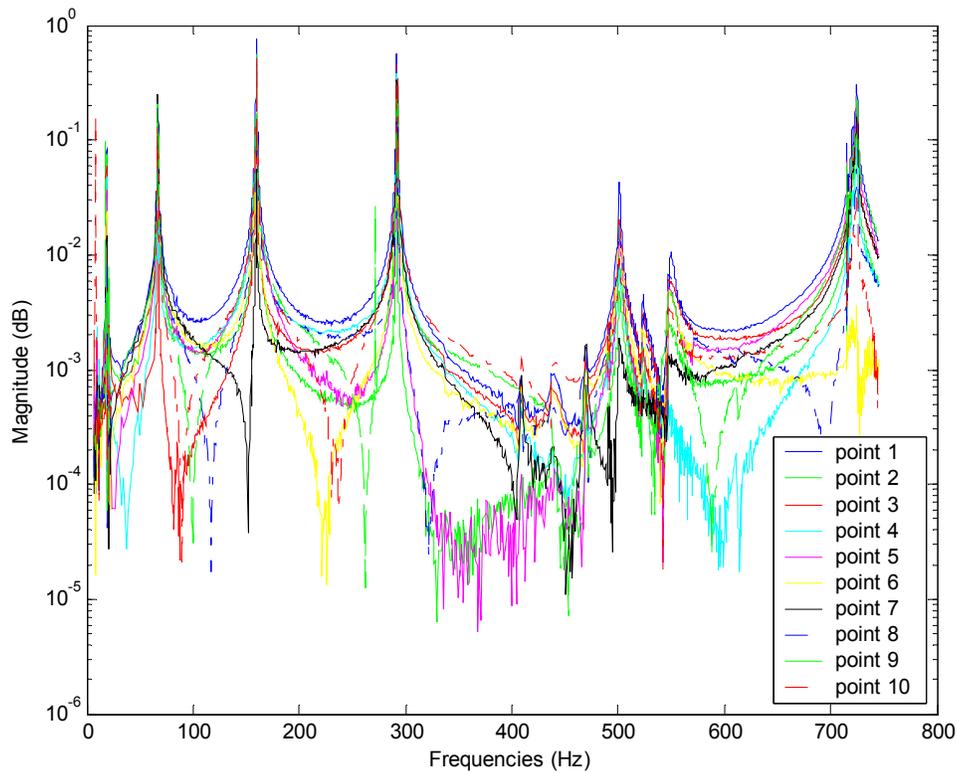
The data collected by the laser vibrometer was fed into a SigLab DSP board. SigLab is a signal processing plug-in compatible with Matlab. The Virtual Sine Sweep (VSS) toolbox in SigLab was used to control the excitation of the structure.



**Figure 4.3.2:** Schematics of experimental setup.

### 4.3.2 Experimental Results

Using the SigLab DSP board, 10 frequency response functions (FRF) were obtained for the experiment. The data collected can be seen in Figure 4.3.3. Since the beam was actuated with a PZT patch the bending and the torsion modes were being excited simultaneously. It is clear that the six major peaks in this figure correspond to the bending modes. The finite element model was used to estimate the natural frequencies to make sure of this. From this data, only the bending modes and frequencies were obtained and used for model updating.



**Figure 4.3.3:** Frequency response functions of the cantilever beam.

By using the Virtual Sine Sweep form SigLab, it was possible to control the frequency resolution. Thus, high frequency resolution was used in the regions surrounding the natural frequencies and low other wise. This was mainly done to reduce the time consumption of the experiment. The frequencies locationed close to the natural frequencies showed almost perfect coherence. This gave confidence in the experimental results.

The Peak Picking method was used to retrieve the natural frequencies and mode shapes. Since, as explained before, the frequency resolution at the peaks was increased, very well defined peaks were obtained making this method acceptable. The experimental results obtained were,

$$\omega_E = [18.4000 \quad 67.1495 \quad 159.7900 \quad 292.0320 \quad 500.9667 \quad 724.4350]$$

for the natural frequencies in Hertz, and

$$\Phi_E = \begin{bmatrix} -0.1112 & -7.9564 & -33.8689 & -39.1449 & 13.0470 & -16.6492 \\ -0.1982 & -11.5200 & -35.3576 & -19.0606 & -12.7661 & -111.0134 \\ -0.3783 & -14.2965 & -23.9185 & 12.0069 & -18.6938 & -25.6685 \\ -0.5738 & -14.4752 & -3.2823 & 28.1345 & -2.0260 & 79.3508 \\ -0.9220 & -13.7575 & 18.0222 & 17.8037 & 19.5025 & -1.1971 \\ -1.3746 & -10.4865 & 30.4185 & -10.7296 & 17.5321 & -97.4717 \\ -1.8616 & -5.0665 & 29.0250 & -32.8030 & -8.7362 & -36.2786 \\ -2.3956 & 1.4383 & 13.1741 & -28.1158 & -17.7243 & 74.4720 \\ -2.9256 & 9.1317 & -14.3789 & 3.2774 & -3.9466 & 48.9005 \\ -3.5378 & 15.9052 & -45.9741 & 48.0287 & 34.8089 & -104.5533 \end{bmatrix}$$

for the mode shapes.

As explained in the previous section, these mode shapes correspond to the 10 points taken from the free end of the beam. In the next section, it will be explained how it is planned to expand these experimental mode shapes in order to use them for model updating of a finite element model.

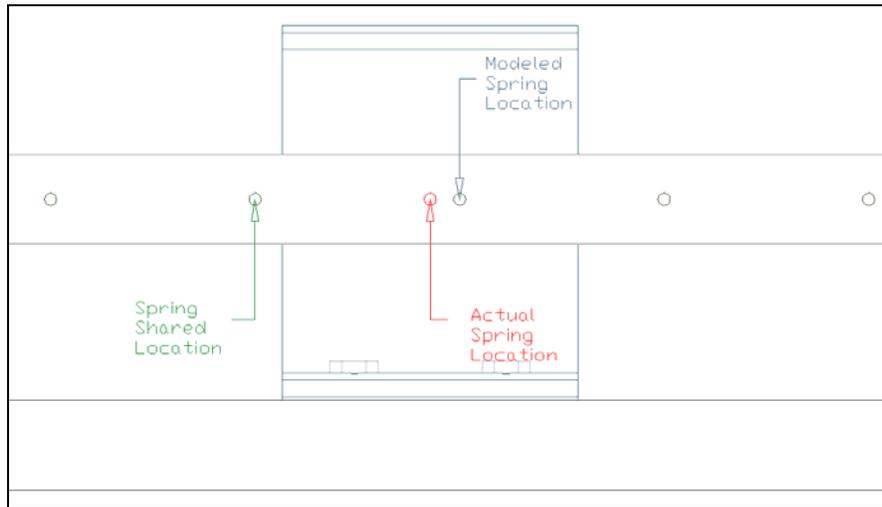
### 4.3.3 Model Updating

The frequency range used to test the beam was chosen with the idea of being able to retrieve five or six frequencies and mode shapes. Although there are six frequencies and mode shapes, only five were used in the model updating process to be able to use the last natural frequency for validation purposes. It was also found that the more frequencies that were tried to match the better results that were obtained. This is due to the fact that the more information that is used in the updating process, the higher possibilities of finding a better fit is achieved. This is the same effect of fitting a curve, the more points used to fit the better fit obtained.

A finite element method approach was used for the model. The finite element model of a cantilever Euler Bernoulli beam was programmed in MatLab and can be seen in Appendix C. The assumption of small shear deformation (Euler Bernoulli beam) is valid in this case since

$$\frac{L}{b} = 28 \quad \frac{L}{h} = 167,$$

which complies with  $L/b$  and  $L/h \geq 10$ , the characteristics of a slender beam. The finite element model was also modified to account for the additional spring that is incorporated into the experiment. A spring constant of 3.0 N/mm was incorporated into a degree of freedom of the model to account for the added local stiffness provided by the spring. Closer examination of the structure revealed that the spring was offset from that particular degree of freedom with which it was being modeled. Thus, the spring acted on the element and should be shared by two degrees of freedom when modeled. Figure 4.3.4 (a closer look of Figure 4.3.2) shows this in a more graphical sense. It can be expected that the model updating techniques are able to see this and correct it. Thus, this will be left as a foreseen error in the model.



**Figure 4.3.4:** Schematic of correct spring location.

The references for model updating of the cantilever beam were taken to be  $M_A$ ,  $\omega_E$  and  $\Phi_E$ , where the first one is the mass matrix provided by the finite element model and the rest are the experimental natural frequencies and mode shapes respectively.

Before using the mode shapes that were extracted from the experiment, expansion needed to be performed on them. Expansion is done because the data collected experimentally and the finite element model used do not match in dimensions. Therefore, the process of completing the experimental mode shapes, in order to bring the analytical and experimental data to the same number of degrees of freedom, is called expansion.

In this case, the finite element model takes into account the linear displacement coordinates at 12 points and the rotational coordinates (angles) in the beam. This accounts for a total of 24 degrees of freedom in the system. The experimental data retrieved only measured 10 linear displacement degrees of freedom, thus the inconsistency. The idea behind the expansion process is to supplement the 10 degrees of freedom obtained experimentally to be able to compare them to the analytical model. Thus, the analytical data and the experimental data are used to expand the experimental data to 24 degrees of freedom, in order to be able to use them for model updating.

There are many approaches that can be taken to expand the experimental mode shapes as illustrated by Gysin in [8]. Among them are eigenvector-mixing, dynamic expansion, modal coordinate method and SEREP. Eigenvector-mixing is the simplest expansion method that fills in the elements from the finite element mode shapes in the corresponding, missing experimental mode shape group. Dynamic expansion is based on eigenvalue equation. It uses the eigenvalue equation to find a relationship, transformation matrix, between the experimental and analytical mode shapes. Once the transformation matrix is obtained it is then used to estimate the missing data. In the modal coordinate method the experimental mode shapes are considered as a linear combination of the corresponding analytical mode shapes and thus solves for the

missing with this same relationship. SEREP is similar to modal coordinate method with the difference that the transformation matrix is a global curve fitting function which projects the mode shapes elements of the correlated data to the full system.

Without a doubt, this incorporates some erroneous data into the experimental data, but this is a common and acceptable way of getting around the missing data. The smoothing and expansion algorithm suggested by Lipkins and Vandeurzen in [15] was used to expand the mode shape data obtained experimentally. This method is based on the assumption that the experimental mode shapes are linear combinations of the analytic ones. This method, described as the most effective and reliable [16,8] can be regarded as a special case of SEREP.

Once the mode shapes were expanded, analysis of the appropriate locations that needed to be updated now needs to be addressed. Looking back at Figure 4.3.4, which shows the fact that the spring acts on the element and not on a particular degree of freedom, was accounted for. This was done by creating two updating parameters to update these two stiffness locations. Also added, was the beam's stiffness as another updating parameter. This can be thought of as the Young's modulus, but the change in stiffness can also be attributed to other factors such as the beam's thickness. Another parameter found to be useful in updating the stiffness matrix, is one that will be referred to as the surrounding stiffness parameter. This parameter slightly changed the stiffness of the locations that were very close to the spring. The parameters and their updated values for the QCM and FVM cases are shown in the Table 4.3.2. The program used in the updating process can be seen in Appendix A for the QCM, Appendix B for the FVM and the program that puts them all together can be seen in Appendix C.

**Table 4.3.2:** Updating parameters used.

	Nominal Value	Updated Value QCM	Updated Value FVM
Beam Stiffness	7.1E10 N/m	-17%	-24.50%
Modeled Spring Location	3000 N/m	0.006 N/m	0 N/m
Shared Spring Location	0	253.1 N/m	-639.9 N/m
Surrounding Stiffness	0	21 N/m	30.6 N/m

Note: The value of zero in this table does not necessarily represent the value of the stiffness matrix at that location. They mainly represent that the value or position has not been updated.

The beam stiffness parameter, although large for the QCM case, seems to be acceptable. In the case of FVM, it seems to have reached an even larger value. The beam did have its cross-sectional area vary by at least 3%, which would reflect as 5% in the updating parameter. In the same way, Young's Modulus for an aluminum beam can vary about 7%. Considering the initial length of the beam, it was found that we had around 2% error in that as well. This made the results of the QCM reasonable and not too far behind FVM. For the Modeled Spring location, both systems returned close to zero updating, but for the Shared Spring location, QCM returned a very reasonable number as opposed to FVM. If the effects of both spring locations are added, QCM holds a value very close to the manufacturer's specified spring stiffness of 3.34 N/mm. The FVM value would be 29.3 % off, while QCM provided a value that is only 2.6% off. The main difference in the surrounding stiffness parameter was that both methods required different updating locations. The updating values given by both methods were considerably small with respect to the other updating parameters, which lead to the assumption that the spring might have slightly altered the interaction between the elements where it was placed.

Table 4.3.3 lists the first six natural frequencies of the experiment, the initial finite element model, as well as the updated frequencies by the QCM and by the FVM for comparison. Table 4.3.4 shows the error in percent with respect to the experimental frequencies of the initial finite element model and both updated methods. It can be seen that the QCM is superior in updating than the FVM. The only frequency that the FVM is able to update reasonably is the fourth one, while QCM does a great job at matching all of them quite accurately.

As mentioned before, only five frequencies were used in the updating process and the last frequency, the sixth, was left out to use later as a form of validation. It can clearly be seen from the last row of Table 4.3.4 that QCM matched the last frequency within 0.16 % of error. This shows how well QCM works in updating the adequate parameters so that it can still match those frequencies that are not being used in the updating process.

**Table 4.3.3.** Natural frequencies.

Frequencies (Hz)			
EXP	FEM	QCM	FVM
18.40	18.43	18.40	20.35
67.15	72.86	67.29	63.70
159.79	163.63	158.38	154.92
292.03	323.53	294.50	292.40
500.97	530.65	492.00	479.77
724.44	794.98	723.28	691.33

**Table 4.3.4.** Error in percent of the different methods.

Percent Error		
$\Delta$ FEM	$\Delta$ QCM	$\Delta$ FVM
0.14	0.01	10.58
8.50	0.21	5.14
2.40	0.88	3.05
10.79	0.84	0.13
5.92	1.79	4.23
9.74	0.16	4.57

The MAC values for the initial and updated models can be seen on Tables 4.3.5-4.3.7. It can be seen that the MAC values were very good for all three cases and not much change was seen in the updated cases. The correlation between the experimental mode shapes and the analytical finite element model can be seen in Table 4.3.5. The first component of the diagonal in the table represents the correlation between the 1<sup>st</sup> analytical mode shape and the 1<sup>st</sup> experimental mode shape. A value of 1 represent perfect correlation and 0 no correlation at all. Thus, it is expected to have values

closer to 1 in the diagonal and close to 0 in the other positions. Evaluating this table as well as the following two, it can be seen that the mode shapes have very good correlation from the beginning and both updated models maintain it as well. But this also shows that the MAC number is very insensitive when the values get closer to one due to its non linearity.

**Table 4.3.5:** The MAC of the experimental and analytical mode shapes.

		Experimental Mode Shapes				
		1	2	3	4	5
Analytical Mode Shapes	1	0.9956	0.0062	0.0314	0.0791	0.0388
	2	0.0027	0.9822	0.0382	0.1213	0.0057
	3	0.0379	0.0352	0.995	0.0688	0.0366
	4	0.0748	0.1368	0.0602	0.9797	0.061
	5	0.0411	0.0231	0.0339	0.0377	0.9845

**Table 4.3.6:** The MAC of the experimental and QCM updated mode shapes.

		Experimental Mode Shapes				
		1	2	3	4	5
QCM Updated Mode Shapes	1	0.9957	0.0283	0.0127	0.0785	0.0366
	2	0.0179	0.983	0.0035	0.1209	0
	3	0.0203	0.0005	0.9948	0.0698	0.061
	4	0.0784	0.1322	0.0632	0.9799	0.0595
	5	0.0395	0.026	0.0578	0.0343	0.9836

**Table 4.3.7:** The MAC of the experimental and FVM updated mode shapes.

		Experimental Mode Shapes				
		1	2	3	4	5
FVM Updated Mode Shapes	1	0.9958	0.0052	0.0048	0.0827	0.0386
	2	0.0075	0.9793	0.0486	0.1377	0.0053
	3	0.0003	0.049	0.9953	0.037	0.0644
	4	0.0819	0.1534	0.0239	0.9792	0.0233
	5	0.0391	0.0323	0.056	0.0039	0.9816

## 4.4 Chapter Summary

In this chapter, both methods, the QCM and FVM, were put to the test. It is clear that under ideal conditions of perfect data, both methods are capable of providing the correct updating information needed to have a successful model. But, even under those conditions, QCM is able to achieve this with less computational burden, as it solves a smaller system of equations for the updating parameters. Under more realistic conditions, where the data retrieved contains noise, QCM pulls apart from the more general approach of FVM, with considerable robustness with respect to noise. This was proven in Chapter 3 from an analytical point of view and was backed with experimental results conducted in this chapter.

## Chapter 5

# Conclusion

Through this research, the usefulness of implementing principal submatrices to solve the model updating problem was shown. It provides a simple way of updating certain components of the analytical stiffness matrix and also helps preserve connectivity in the results. More importantly, it was shown that the Quadratic Compression Method offers greater robustness with respect to noise in the measured mode shapes, than any other similar methods, while also reducing the amount of equations that need to be solved. In small cases like the ones presented in this research, it might be irrelevant, but in larger systems that require thousands of degrees of freedom this will turn out to be a time saving advantage. It was also shown that the dominant term of the error in this method is quadratic as compared to linear in the Full Vector Method. The accuracy of the mode shapes is critical in both Sensitivity and Reference Basis methods, and mode shapes tend to be less accurate than natural frequencies, in particular when some of their components are obtained by expansion. Therefore, any improvement in robustness is translated immediately to more accurate updated models.

### **Contributions**

In a final note, this research has provided the area of Model Updating with a new method with combined advantages of previous ones. It has implemented the

advantages of the closed-form solution and ease of calculations common in RB Methods. It has also implemented the connectivity, symmetry, and user intervention advantages also common in Sensitivity Methods and on its own the reduced system of equations and the robustness with respect to noise. This makes the Quadratic Compression Method a special and unique case that can perform all of these things at once.

This research has contributed to the following papers:

Tarazaga, P.A., Halevi, Y., and Inman, D.J., 2004. "Model Updating with the Use of Principal Submatrices," Proceedings, International Modal Analysis Conference (IMAC XXII), January 26-29, 2004, Dearborn, MI.

In progress:

Tarazaga, P.A., Halevi, Y., and Inman, D.J, "Model Updating Using a Quadratic Form" for a journal paper.

Experiment was also used in:

Y. Halevi, P. A. Tarazaga, and D.J. Inman. "Connectivity Constrained Reference Basis Model Updating". International Conference on Noise & Vibration Engineering (ISMA) 2004, Leuven, Belgium.

Other:

Ruggiero, E. J., Tarazaga, P. A., and Inman, D.J., 2004. "Modal analysis of an ultra-flexible, Self rigidizing toroidal satellite component". Proceedings of 2004 ASME International Mechanical Engineering Congress and Exposition, November 13-19, 2004, Anaheim, CA.

## **Future Work**

While the issues that motivated this research have been addressed there are areas that could encourage future work. The incorporation of mass matrix into the updating method would result in a more rounded ample method for the model updating area. The incorporation of damping could also be beneficial to achieve better and more accurate models.

# Bibliography

1. Allemang, R.J. and Brown, D.L., “A correlation Coefficient for Modal Vector Analysis”, Proceedings of the International Modal Analysis Conference, pp. 110-116, 1982.
2. Atalla, M. J. and Inman, D. J., “On model updating using neural networks. Mechanical Systems and Signal Processing”, Vol 12, pp. 135-161 , 1998.
3. Baruch, M. and Bar Itzhack, Y., “Optimal Weighted orthogonalization of measured modes”. AIAA Journal 16, pp. 346-351, 1978.
4. Berman, A. and Nagy, J., “Improvement of a large analytical model using test data”. AIAA Journal 21, pp. 1168-1173, 1983.
5. Berman, A., “System Identification of Structural Dynamic Models Theoretical and Practical Bounds”. AIAA/ASME/ASCE/AHS 25<sup>th</sup> Structures, Structural Dynamics and Materials Conference, pp. 123-129,1984.
6. Bertsekas, Dimitri, “Nonlinear Programming”, Athena Scientific, 1995.
7. Ewins, D.J., “Modal Testing: Theory, Practice and Application”, 2<sup>nd</sup> Edition, Research Studies Press LTD, England 2000.
8. Gysin, H-P., “Comparison of Expansion Methods for FE Modeling Error Localization”, IMAC 8, Kissimmee, FL, pp. 195-204, 1990.
9. Halevi, Y. and Bucher, I., “Model updating via weighted reference basis with connectivity constraints”. Journal of Sound and Vibration, Vol 265, pp. 561-581, 2003.

10. Hemenz, F. and Farhat, C., "Updating finite element dynamic model using an element-by-element sensitivity methodology". *AIAA Journal* 31, pp. 1702-1711, 1993.
11. Kenigsbuch, R. and Halevi, Y., "Model updating in structural dynamics: a generalized reference basis approach". *Mechanical Systems and Signal Processing* 12, pp. 75-90, 1998.
12. Lallement, G., Piranda, J. and S. Cogan, "Parametric correction of finite element models by minimization of an output residual: improvement of the sensitivity method". *Proceedings of the 9<sup>th</sup> IMAC*, pp. 363-368, 1991.
13. Levin, R. I. and Lieven, N. A. J., "Dynamic finite element model updating using simulated annealing and genetic algorithms". *Mechanical Systems and Signal Processing*, Vol 12, pp. 91-120, 1998.
14. Link, M., "Updating analytical models by using local and global parameters and relaxed optimization requirements", *Mechanical Systems and Signal Processing*, Vol 12, pp. 7-22 1998.
15. Lipkins, J., and Vandeurzen, U., 1987, "The Use of Smoothing Techniques for Structural Modification Application", *Proceedings of the 12<sup>th</sup> International Seminar on Modal Analysis*, Leuven, Belgium. S1-3.
16. Maia, N.M.M, and Silva, J.M.M, (Ed.), 1997, "Theoretical and Experimental Modal Analysis", Research Study Press, Baldoc, England.
17. Mottershead and Friswell, "Finite Element Model Updating in Structural Dynamics", Kluwer Academic Publishers, 1995.
18. Rao, C.R. and Mitra, S.K., "Generalized Inverse of Matrices and Its Applications", John Wiley, New York 1971.
19. Rajendra Bhatia, "Graduate Texts in Mathematics: Matrix Analysis", Springer, New York, 1997.

## Appendix A

# MatLab Code of the Quadratic Compression Method

```
function [sigma,ResQCM,A,b]=QCM(x,y,Ka,P)
% QCM is the Quadratic Compression Method
% that solves for the optimal updating parameters.
%
%[sigma,ResQCM,A,b]=QCM(x,y,Ka,P)
%
%The inputs are
%       x   the experimental modesshapes.
%       y   the experimntal frequencies squared in a diagonal
%           matrix.
%       Ka  the analytical stiffness matrix.
%       P   the placement matrices fro the updating parameteres
%           they should be arranged in the form P(:, :, i).
%
%
%The outputs are
%       sigma  the updating parameters.
%       ResQCM is the residue of norm(A*sigma-b,'fro')
%       A & b  are the components of the system A*sigma=b

[n,m]=size(x);
num=size(P);
for i=1:num(3)
    for j=1:m
        A(j,i)=x(:,j)'+P(:, :, i)*x(:,j);,
        b(j)=y(j,j)-x(:,j)'+Ka*x(:,j);
```

```

        end
    end

    dd=m*(m+1)/2;
    for s=(m+1):dd
        for t=1:m-1

            for q=t+1:m
                for h=1:num(3)
                    A(s,h)=x(:,t)'*P(:, :, h)*x(:, q);
                    b(s)=-x(:,t)'*Ka*x(:, q);
                end
            end
        end
    end

    end
end
A;
b=b';
sigma=A\b;
ResQCM=A*sigma-b;

```

## Appendix B

# MatLab Code of the Full Vector Method

```
function [sigmaf,ResFVM,Af,bf]=FVM(x,y,Ka,Ma,P)
% FVM is the conventional Method for Model Updating
% that solves for the optimal updating parameters.
%
%[sigma,ResQCM,Af,bf]=FVM(x,y,Ka,P)
%
%The inputs are
%       x   the experimental modeshapes.
%       y   the experimntal frequencies squared in a diagonal
matrix.
%       Ka  the analytical stiffness matrix.
%       Ma  the analytical mass matrix
%       P   the placement matrices fro the updating parameteres
%           they should be arranged in the form P(:, :, i).
%
%The outputs are
%       sigmaf  the updating parameters.
%       ResFVM  is the residue of norm(Af*sigmaf-bf,'fro')
%       A & b   are the components of the system Af*sigmaf=bf

[n,m]=size(x);
num=size(P);
c=num(1);
for j=1:m

    bf(c*(j-1)+1:c*(j-1)+c,1)=Ma*x(:,j)*y(j,j)-Ka*x(:,j);

    for i=1:num(3)
```

```
Af(c*(j-1)+1:c*(j-1)+c,i)=P(:, :, i)*x(:, j);
```

```
end
```

```
end
```

```
sigmaf=Af\bff;
```

```
ResFVM=norm(Af*sigmaf-bff);
```

## Appendix C

# MatLab Code of the Finite Element Model

```
function [M,K,S,Freq,P]=finbeam(l,numel,rou,A,I,E,kad,sploc)
%[M,K,S,FreqH,P]=finbeam(l,numel,rou,A,I,E,kad,sploc)
%
% This program generates the a Finite Element Model for an Euler-
Bernoulli
% of a Cantilever beam
% given the inputs
%
%           l           The length of the beam.
%
%           numel       The number of elements desired
%
%           rou         The density of the material
%
%           A           The cross sectional area
%
%           I           moment of inertia
%
%           E           Young's Modulus(Elastic Modulus)
%
%           kad         Additional spring
%
%           sploc       spring location
%
% The outputs are
%
%           M           The Mass Matrix
%
%           K           The stiffness Matrix
%
%           S           Mode shapes
%
%           Freq        Frequencies in rad/sec
%
%           P           The eigenvectors of  $M^{(-1/2)}KM^{(-1/2)}$ 

le=l/numel;
dof=numel*2+2;

m=rou*A*le/420*[ 156    22*le    54    -13*le;
                 22*le   4*le^2   13*le  -3*le^2;
                 54     13*le   156    -22*le;
                 -13*le -3*le^2 -22*le  4*le^2];
```

```

k=E*I/(1e^3)*[ 12      6*1e      -12      6*1e;
               6*1e      4*1e^2     -6*1e      2*1e^2;
              -12      -6*1e         12      -6*1e;
               6*1e      2*1e^2     -6*1e      4*1e^2];

M=zeros(dof,dof);
for i=1:2:dof-2
    point=[i i+1 i+2 i+3];
    M(point,point)=M(point,point)+m;
end
K=zeros(dof,dof);
Per=zeros(dof,dof);
for i=1:2:dof-2
    point=[i i+1 i+2 i+3];
    K(point,point)=K(point,point)+k;
    Per(point,point)=Per(point,point)+per;
end

%=====Additional Spring=====
q=sploc;
K(q,q)=K(q,q)+kad;
%=====
%=====
% Applying boundary conditions
% This part can be comented out for a free-free beam
Kw=K;
M=M([3:end],[3:end]);
K=K([3:end],[3:end]);
Per=Per([3:end],[3:end]);
%=====
%=====
[S, fre2]=eig(K,M);
Freq=sqrt(fre2);
Li=chol(M);
L=Li';
Kt=inv(L)*K*inv(L');
[P, fre2]=eig(Kt);

```

## Appendix D

# MatLab Program Used to the Run the Updating Methods for the Experiment

```
clear all
close all

%===== Experimental Frequencies
=====
wt=[18.4000  67.1495  159.7900  292.0320  500.9667  724.4350]*2*pi;
%===== rad/sec

% =====Experimental modeshapes=====
xt=100*[ -0.00111238117828  -0.07956421185995  -0.33868943401647
-0.39144939289723  0.13046950416082  -0.16649236447459
-0.00198246354125  -0.11520014034782  -0.35357612222274  -
0.19060575518837  -0.12766088167659  -1.11013391535590
-0.00378321426260  -0.14296519372388  -0.23918454440930
0.12006911584530  -0.18693831409527  -0.25668482005151
-0.00573827111388  -0.14475156855824  -0.03282348540351
0.28134518819457  -0.02025977323796  0.79350802228234
-0.00922015400937  -0.13757469665946  0.18022245358446
0.17803669041671  0.19502502664650  -0.01197127822764
-0.01374625680177  -0.10486537372963  0.30418498279790  -
0.10729595588689  0.17532110494457  -0.97471728504116
-0.01861568911042  -0.05066489517797  0.29024976850472  -
0.32802961091414  -0.08736203764278  -0.36278593787847
-0.02395634977155  0.01438334126394  0.13174104184099  -
0.28115770096820  -0.17724257336203  0.74472040702413
-0.02925607214447  0.09131657043776  -0.14378872650770
0.03277394701712  -0.03946576187644  0.48900504784727
```

```

-0.03537785508876    0.15905248232589   -0.45974112519435
0.48028708546717    0.34808863583374   -1.04553321096843];
%=====

%=====
%===== FEM ANALYTICAL MODEL=====
%=====
==

ch=3.87/22;
kk=19*ch*1000;
kk=3000; %Estimated spring constant! Experimental real value 3342

L=0.5307;b=19.05e-3;h=3.17e-3;rou=2700;A=b*h;I=b*h^3/12;
z=0;
noe=12+z;
loc=11+z;
E=7.1e10;[Ma,Ka,S,Freq,P,Per]=finbeamad(L,noe,rou,A,I,E,kk,loc);

FreqAn=diag(Freq)/2/pi; % Analytical Frequencies in Hertz

%S=fliplr(S);
S=S(:,[1:6]); % Analytical Mode Shapes

xa=S(:,[1:5]); % used at the end for comparison reasons
%=====
==
%===== EXPANSION METHODS
=====

Phxm=xt; % (10 X 6)
Pha=S; % (24 X 6)
M=Ma; % (24 X 24)

if z==0;
Im=[5:2:24]'; % (10,1)
else
Im=[11:4:48]';
end

Pham=Pha(Im,:);

% Dr. Link's method

```

```

c=pinv(Pham)*Phxm;
Phxxn_link=Pha*c*sqrtm(inv(c'*c));

% New method Dr. Halevi
c_new=sqrtm(inv(Pham'*Phxm*Phxm'*Pham))*Pham'*Phxm;
Phxxn_new=Pha*c_new;

xe=Phxxn_link; %==== LINK METHHOD selected =====
%xe=Phxxn_new; %==== Halevi ==alternative not used=====
xec=xe; % Data used for comparison at end
yec=diag(wt)^2; %Data used for comparison at end
%=====
%===== Amount of Mode Shapes we choose to use =====
gg=5;
y=diag(wt(1:gg))^2; % Frequencies Squared
xe=xe(:,[1:gg]);

%=====

%===== Updating =====
loc=9+z; % 9 or 19
P(:, :, 1)=zeros(size(Ka)); % Estimated error location
(Spring Location)
P(loc, loc, 1)=1;

P(:, :, 2)=Ka; % Beam Stiffness Location Geometry corection and/or
Modulus of Elasticity

%-----

%=====
% Updating Location for Surrounding Stiffness
%=====

P(:, :, 3)=zeros(size(Ka));
P(12+z, 9+z, 3)=-1;
P(9+z, 12+z, 3)=-1;

P(11+z, 9+z, 3)=-1;
P(9+z, 11+z, 3)=-1;

P(9+z, 10+z, 3)=-1;

```

```

P(10+z,9+z,3)=-1;

P(12+z,12+z,3)=1;

P(8+z,9+z,3)=1;
P(9+z,8+z,3)=1;

P(10+z,7+z,3)=-1;
P(7+z,10+z,3)=-1;

P(13+z,12+z,3)=-1;
P(12+z,13+z,3)=-1;

%===== CALLING Updating Program QCM =====

[sigma,ResQCM,A,b]=QCM(xe,y,Ka,P);

%-----FVM-----
Pf(:, :, 1)=zeros(size(Ka)); % Estimated error location (SPring
Location)
Pf(9,9,1)=1;

Pf(:, :, 2)=P(:, :, 2); % Beam Stiffness Location Geometry corection
and/or Modulus of Elasticity
Pf(:, :, 3)=zeros(size(Ka));

%=====
% Updating Location for Surrounding Stiffness
%=====
Pf(9+z,10+z,3)=-1;
Pf(10+z,9+z,3)=-1;

Pf(11+z,13+z,3)=1;
Pf(13+z,11+z,3)=1;

Pf(12+z,12+z,3)=1;

Pf(11,11,3)=1;
Pf(8,8,3)=1;

Pf(8+z,9+z,3)=1;

```

```

Pf(9+z,8+z,3)=1;
%=====

%===== CALLING Updating Program FVM =====

[sigmaf,ResFVM,Af,bf]=FVM(xe,y,Ka,Ma,Pf);

%=====

%===== Assembly of Updated Stiffness Matrix QCM =====
num=size(P);
Kup=zeros(size(Ka))+Ka;

for i=1:num(3)
    Ki=Kup+sigma(i)*P(:, :, i);
    Kup=Ki;
end

%=====

%===== Assembly of Updated Stiffness Matrix FVM!
numf=size(Pf);
Kupf=zeros(size(Ka))+Ka;
for i=1:numf(3)
    Kif=Kupf+sigmaf(i)*Pf(:, :, i);
    Kupf=Kif;
end

%=====

% =====Updated Mode Shapes and Frequencies QCM=====

[Sup,fre2pu]=eig(Kup,Ma);
Frequp=diag(sqrt(fre2pu)/2/pi);

% =====Updated Mode Shapes and Frequencies FVM=====

[Supf,fre2puf]=eig(Kupf,Ma);

```

```

Frequpf=diag(sqrt(fre2puf)/2/pi);
%=====

%===== Comparison Section =====

Freq_com=[(wt/2/pi)' Frequp(1:6) Frequpf(1:6) FreqAn(1:6)];

Freq_Per_Err=[abs((wt/2/pi)'-Frequp(1:6))./(wt/2/pi)'*100, ...
              abs((wt/2/pi)'-Frequpf(1:6))./(wt/2/pi)'*100, ...
              abs((wt/2/pi)'-FreqAn(1:6))./(wt/2/pi)'*100, ...
              (abs((wt/2/pi)'-Frequp(1:6))./(wt/2/pi)'*100)-
              (abs((wt/2/pi)'...
              -FreqAn(1:6))./(wt/2/pi)'*100),...
              (abs((wt/2/pi)'-Frequpf(1:6))./(wt/2/pi)'*100)-(abs((wt/2/pi)'...
              -FreqAn(1:6))./(wt/2/pi)'*100)];

%=====
disp('-----DATA-Comparisons-----')
disp('Frequencnies of -')
disp('.')
disp('  Expt          QCM          FVM          Analytical')
Fo=Freq_com
disp('=====')
disp('Percent Error Vs Experimental')
disp('  QCM          FVM          FEM          DifQCM(-)  DifFVM (-)')
PE=Freq_Per_Err
disp('-----')
%=====

```