University of Cincinnati

Date: 11/3/2015

I, Adityanarayan C Niranjana, hereby submit this original work as part of the requirements for the degree of Master of Science in Mechanical Engineering.

It is entitled:

Normalization of Complex Mode Shapes by Truncation of the Alpha-Polynomial

Student's name: **Adityanarayan C Niranjana**

This work and its defense approved by:

Committee chair: Randall Allemang, Ph.D.

Committee member: David L. Brown, Ph.D.

Committee member: Allyn Phillips, Ph.D.

19602

**Normalization of Complex Mode Shapes by Truncation of the
Alpha-Polynomial**

A thesis submitted to the
Graduate School
of the University of Cincinnati
in partial fulfillment of the
requirements for the degree of

Master of Science

in the Department of Mechanical and Materials engineering
of the College of Engineering and Applied Sciences
by

Adityanarayan C. Niranjana

B.E. University of Pune, India

June 2013

Committee Chair: Dr. Randall J. Allemang

Committee Members: Dr. Allyn W. Phillips

Dr. David L. Brown

Abstract

Finite element (FE) models are created to correctly predict the dynamic characteristics of any system without the need to test the system. This is a viable solution to test the system in situations and environments which might be either physically unfeasible or exorbitantly expensive. These FE models however need to be validated with the data obtained from actual tests to affirm that the model's predictions are indeed correct. One parameter that is needed to validate analytical model is the mode shape.

The mode shapes obtained from F.E. models are always real-valued due to the proportional nature of the assumed damping. The mode shapes obtained through experimental techniques, however, are more often than not complex-valued (complex modes) in nature. In order to validate the F.E. models, the complex mode shapes obtained from experimental data needs to be normalized. In this thesis, two such normalization techniques have been proposed.

Over the years, several post-processing techniques have been suggested which are discussed briefly in Chapter (2). In this thesis, of the two proposed techniques, the first method is a pre-processing normalization technique based on truncating the α -polynomial obtained from the Unified Matrix Polynomial Approach (UMPA) method of modal parameter estimation.

The proposed concept was tested on data obtained from a circular plate using frequency domain based high order algorithm Rational Fraction Polynomial (RFP). The proposed technique eliminated the damping and normalized the complex modal vectors effectively in low order polynomial based algorithms like Polyreference Frequency Domain - 2 (PFD - 2). The undamped natural frequencies, however, deviated from that of the actual system in case of polynomials with order higher than two. These digressions are explained with the help of 2 and 4 degree of freedom (DOF) analytical models.

The second normalization method proposed is a post-processing technique based on the partial fraction method of residue synthesis. This technique was tested on data obtained from a rectangular plate. The technique successfully normalized the complex modal vectors with a very good correlation with the actual (complex) mode shapes giving a very high modal assurance criterion (MAC) value.

Acknowledgements

At the end of every book, fiction, non-fiction, technical etc., the author usually thanks several people saying that the book would not have been possible if not for those people. I always wondered, how can so many people help the author to write a book? I mean, wasn't the book mostly his or her work? No doubt, these people must have helped him, encouraged him, supported him through tough times, times when the author doubted himself, but mostly it is the author who actually writes it. I think I now realize what the author means when he thanks so many people. It is not just the guidance or the support for the book. The author takes the opportunity to thank the people who helped him with his book not just directly but also indirectly. I believe, in some way or the other every person in one's life has an affect on the person, positively or negatively. They shape the person's life in several different ways. The authors take the opportunity to acknowledge that fact and proclaim their gratitude to the world.

My work in this thesis is not as significant as the authors I have talked about. But still, thinking about writing this acknowledgment for my thesis made me think of all the people who have been a part of my life here at Cincinnati and the ones since I can remember. I am not very good at saying it to their faces, so this opportunity to silently thank them without really looking into their eyes appealed to me.

Firstly, I would like to thank Dr. Randall Allemang and Dr. Allyn Phillips from the bottom of heart for going out of their way to assist me with my thesis. Without their advise, guidance and help I wouldn't have a thesis topic let alone a masters thesis. I cannot emphasize enough on how important their involvement in my masters has been. It was truly an honor to learn from both these individuals. I would also like to thank Dr. Phillips for funding me for five straight semesters and also for the several tweaks and updates he made in X-Modal to help me with my thesis. The Teaching Assistantship

not only helped me and my parents financially, but it helped me strengthen my basic understanding of structural dynamics and vibrations.

All the little things that I have accomplished in my life is only due to the constant, unparalleled, unwavering and unending support of my parents. If I decide to run the 100 meters in the next Olympics, I am pretty sure they will tell me that if I put in enough efforts and work hard enough, I will be able to outrun Usain Bolt. That is how unbelievable their support has been. They believe in me when I don't. I would like to thank them for always being there for me and for encouraging me to pursue my higher studies abroad. I would also like to thank my younger brother, Rounak. We usually don't talk about studies or career or life or such trivial stuff, but we make sure we keep up with the important things like football (soccer), movies, books and what Jason Bourne and Agent 47 are up to. Thank you for those random and unnecessary talks, Mangya. I would also like to thank my paternal grandfather and my maternal grandmother who in spite of not having had a lot of formal education, put a lot of importance on education and supported and encouraged me to pursue higher studies.

This has gone on for too long. I am quite sure I won't be writing an acknowledgment again (yes, mummy and papa, no PhD) so I want to thank everyone. So, to conclude, can't miss my friends. My best friends Abhishek, Anish, Mohit and Niyoti (in alphabetical order, so don't read much into it), you didn't really help me much with my thesis but this is not a thank you for supporting me through my thesis anyway. This is just for being a part of my life. Thank you very much! And my friends that I made here, after coming to Cincinnati, Achyut, Akhil, Anish B., Ankita, Asawari, Rohit, Sharang and Shounak. You guys made my life 8000 miles away from home far easy than I could have hoped for. Thank you!

And lastly, keep the blue flag flying high. Chelsea F.C. forever!

Contents

1	Introduction	1
2	Background	3
2.1	Real vs Complex Mode Shapes	3
2.2	Origin of Complex Modal Vectors	5
2.3	Other Normalization Techniques	6
2.3.1	Computation of Normal Modes from Identified Complex Modes .	7
2.3.2	A Time domain Subspace Iteration Method	10
2.3.3	Principal Response Analysis Approach	13
2.3.4	Rescaling of Complex Modal Vectors Technique	14
3	Overview of Experimental Modal Analysis	18
3.1	Frequency Response Function	18
3.1.1	Polynomial Model	19
3.2	Matrix Coefficient Polynomial	21
3.2.1	Companion Matrix	22
3.2.2	Determining the α Matrices	25
3.2.3	Frequency Domain Algorithms	26
3.3	Coefficient Condensation	28
3.3.1	Eigenvalue Decomposition	29
3.3.2	Singular Value Decomposition	30

3.4	Modal Assurance Criterion	31
4	Normalization Techniques	32
4.1	Normalization by Truncation of the α Polynomial	32
4.1.1	Background	32
4.1.2	Elimination of Odd Powered Terms	33
4.1.3	Application on a Circular Plate	34
4.1.4	Experimental Setup and Material Properties	34
4.1.5	Pole Setup and Modal Parameter Estimation	35
4.1.6	Even Powered Polynomials	37
4.1.7	Odd Powered Polynomials	39
4.1.8	Application on Data Obtained Using Polyreference Frequency Do- main - 2 (PFD - 2) Algorithm	42
4.2	Normalization by Manipulation of the Partial Fraction Method of Residue Synthesis	44
4.2.1	Background	44
4.2.2	Using Partial Fraction Method to Obtain the Residues	45
4.2.3	Normalizing the Residues Using Partial Fraction Method	46
4.2.4	Application on a Rectangular Plate	46
4.2.5	Pole Setup and Modal Parameter Estimation	47
4.2.6	Normalizing the Residues Using Partial Fraction Method	48
4.3	Comparison of the Two Techniques	51
5	Explanation for Deviation in Undamped Natural Frequencies	53
5.1	Background	53
5.2	2 DOF Analytical Example	55
5.3	4 DOF Analytical Example	57
5.3.1	Case I: Second Order Polynomial with 4×4 Matrix Coefficients	58

5.3.2	Case II: Fourth Order Polynomial with 2×2 Matrix Coefficients .	59
5.3.3	2 DOF and 4 DOF Example Conclusion	63
5.4	15 DOF Analytical Example	63
5.4.1	Case I: Analytical Solving of the 15 DOF Model	64
5.4.2	Case II: 15×15 FRF Matrix, Full Frequency Range	65
5.4.3	Case III: FRF Matrix 4×15 Complete Frequency Range	67
5.4.4	Case IV: FRF matrix of 4×15 with the Highest Three Modes Left Out	69
5.4.5	Conclusion for the 15 DOF Cases	70
5.5	Coefficient Condensation	71
6	Conclusions and Future Work	72
6.1	Conclusion	72
6.2	Future Work	73
A	MATLAB Scripts	77
A.1	Normalization by Truncation of the α Polynomial	77
A.1.1	Application on Data Obtained Using RFP Algorithm	77
A.1.2	Application on Data Obtained Using PFD - 2 Algorithm	79
A.2	Normalization by Manipulation of the Partial Fraction Method of Residue Synthesis	81
A.3	Analytical Explanation for Deviation in Undamped Natural Frequencies .	84
A.3.1	2 DOF System	84
A.3.2	4 DOF System	85
A.3.3	15 DOF System	89

List of Figures

3.1	Pole in s-plane	22
4.1	Frequency Response Function	35
4.2	Complex Mode Indicator Function	36
4.3	Modal Parameter Estimation	37
4.4	Modal Vector Complexity Comparison	39
4.5	Mode Shape at 362 Hz	40
4.6	Mode Shape at 364 Hz	41
4.7	Mode Shape at 557 Hz	41
4.8	Mode Shape at 761 Hz	41
4.9	Mode Shape at 764 Hz	42
4.10	Modal Vector Complexity Comparison	43
4.11	Selected Bandwidth for Rectangular Plate	47
4.12	Complex Mode Indicator Function	48
4.13	Modal Vector Complexity Residue Comparison for Rectangular Plate . .	49
4.14	Mode Shape at 95 Hz	49
4.15	Mode Shape at 104 Hz	50
4.16	Mode Shape at 119 Hz	50
4.17	Mode Shape at 137 Hz	50
4.18	Mode Shape at 176 Hz	51
4.19	Mode Shape at 203 Hz	51

5.1	2DOF system	53
5.2	4 DOF system	57
5.3	15 DOF CMIF	64
5.4	Selected Bandwidth for 15×15 FRF	66
5.5	Poles for 4×15 FRF Complete Frequency Range	70

List of Tables

4.1	Parameters for the selected bandwidth	36
4.2	Comparison of Modal Frequencies	38
4.3	Modal Assurance Criterion (MAC)	38
4.4	Comparison of Modal Frequencies	42
4.5	Modal Assurance Criterion (MAC)	44
4.6	Parameters for the selected bandwidth	47
4.7	Modal Assurance Criterion (MAC)	49
4.8	Mode Shape Comparison MAC	52
5.1	Comparison of Modal Frequencies	56
5.2	Modal Frequencies Comparison for Analytical 4 DOF System	59
5.3	Case I Modal Frequencies	65
5.4	Case I Modal Frequencies Comparison, 15×15 FRF	66
5.5	Case II MAC	67
5.6	Case III Modal Frequencies Comparison	68
5.7	Case III MAC	69
5.8	Case IV Modal Frequencies Comparison	69
5.9	Case IV MAC	70

Nomenclature

λ_r	Modal frequency corresponding to the r^{th} mode (rad/sec)
Ω	Undamped natural frequency (rad/sec)
ω_r	Damped natural frequency at the r^{th} mode (rad/sec)
σ_r	Modal damping at the r^{th} mode (rad/sec)
N	Number of modes
N_o	Number of output measurement degrees of freedom
$[\alpha]$	Denominator polynomial matrix coefficient
$[\beta]$	Numerator polynomial matrix coefficient
$[\Phi]$	Real modal vector matrix
$[M]$	Mass matrix (kg)
$[C]$	Damping matrix (Ns/m)
$[K]$	Stiffness matrix (N/m)
$[H]$	FRF matrix
$\{x(t)\}$	Displacement vector (m)
$\{\dot{x}(t)\}$	Velocity vector (m/s)

$\{\ddot{x}(t)\}$ Acceleration vector (m/s^2)

$\{f(t)\}$ Force vector (N)

$\{\phi\}$ State vector corresponding to the r^{th} mode

$\{\psi_r\}$ Modal vector corresponding to the r^{th} mode

$\{A_r\}$ Residues corresponding to the r^{th} mode

$\text{Im} ()$ Imaginary part of the matrix

$\text{Re} ()$ Real part of the matrix

Chapter 1

Introduction

Modal parameters of a system are often required to define its dynamic properties. These modal parameters can be estimated by experimental measurements or through analytical techniques like finite element (FE) models. The F.E. models, however, might need updating so that the results from the models are comparable to the experimentally obtained results. This process is called model calibration. Model calibration can be defined as the process of adjusting the physical modeling parameters of the system in the computational model to improve agreement with the experimental data [1][2].

Once calibration is completed, validation of the model can be evaluated based upon the application of the model. One such modal parameter which needs to be evaluated to validate the F.E. model is the mode shape. The nature of a mode shape, real or complex, depends on the damping characteristic of the system. Proportional or no damping in general yields real mode shapes and non-proportional damping typically yields complex mode shapes. There is no way, currently, to accurately determine the physical damping distribution of any system [3]. F.E. models, therefore, are comprised of the mass and stiffness properties and the damping is assumed to be proportional to mass and/or stiffness. Therefore, the mode shapes obtained from F.E. models are real-valued (normal

modes) in nature, whereas the mode shapes obtained from experimental measurements are complex-valued (complex modes). It may be important to obtain the real-normal mode shapes from the experimental measurements in order to be able to compare it with the mode shapes from analytical model and thereby calibrate or validate it.

Normalization of complex-valued modal vectors simply means changing the scaling of the modal vector. Real-normalization, however, means that the scaling of complex-valued modal vectors are changed and are also limited to positive or negative real numbers. In this thesis, two methods are evaluated that real-normalize the complex experimental mode shapes.

Several techniques of real-normalization proposed over the years, briefly discussed in Section (2.3), are post-processing techniques. The primary objective of this thesis is to identify a pre-processing real-normalization technique. The first technique proposed in this thesis is based on truncation of the α -polynomial developed in the Unified Matrix Polynomial Approach (UMPA) model, refer Chapter (3). This technique real-normalizes the modal vector, however, the natural frequencies obtained do not match with the ones of the system as will be shown in Chapter (4). An explanation for this deviation has been devised in Chapter (5). An alternative real-normalization technique based on the partial fraction method of residue synthesis is also proposed. The real-normalized modal vectors are compared with the original complex modal vectors using modal assurance criterion (MAC).

Chapter 2

Background

In this chapter, the difference between real and complex mode shapes will be briefly explained along with the possible origins of the complex modal vectors. Some of the normalization techniques proposed over the years will also be discussed.

2.1 Real vs Complex Mode Shapes

A mode shape is the pattern of displacement of all the points (degrees of freedom) of the system at its natural frequency. When all points are totally in phase or 180° out of phase with any other degree of freedom (DOF), the mode shape is said to be real or normal. If these points are not necessarily in or out of phase with other DOF, it is said to be complex in nature. In an experimental situation, the result is almost always a complex mode. In this situation, it is common to normalize the modal vector in some way so that the degree of complexity is easy to understand. Although there is no unique choice of normalization, this often involves a normalization procedure that adjusts the modal vector so that the largest coefficient is unity and all other coefficients are scaled to this one real value.

In case of real modes, with the possible exception of perfectly repeated modal frequencies, all points pass the minima or maxima at the same instant in time. They pass through zero at the same time. The real mode shapes, therefore, can be described as a standing wave which appears to have a fixed stationary node point. The modal vectors for such mode shapes are real valued numbers and the phase characteristic can be taken from the sign. Such real mode shapes are obtained in the case of an undamped or proportionally damped system. The mode shapes obtained from an undamped and a proportionally damped case are the same.

A system is said to be proportionally damped if the damping matrix of the system is proportional to either mass or stiffness or both, as follows,

$$[C] = \alpha[M]$$

or,

$$[C] = \beta[K]$$

or,

$$[C] = \alpha[M] + \beta[K]$$

where,

$[M], [C], [K]$ are the mass, damping and stiffness matrices respectively

α, β are constants

All the DOFs in the case of complex modes, however, do not pass through their maxima or minima at the same time. Likewise, all the points do not pass through zero at the same time but the zero locations appear to migrate back and forth in the physical system..

Complex modes can therefore be described as a traveling wave and appears to have a moving node on the structure. The modal vectors are complex in nature due to the non-proportional damping in the system [1]. However, non-proportional damping is not the only cause of complex mode shapes. Some other causes are discussed in the following section that may result from errors in the modal parameter estimation..

2.2 Origin of Complex Modal Vectors

Even in proportionally damped systems, complex vectors are sometimes seen. The possible causes of such complexity are discussed below,

1. Measurement Errors [4]: Several types of measurement errors can lead to complexity of modal vectors. They are,
 - Uncommon errors: These errors are due to hardware failures such as a filter channel failure, phase mismatching between channels due to input gain failures, internal hardware calibration problems or lack of aliasing filters (generally a system error or user mistake) which would be uncommon today but still possible.
 - Leakage: This is a much more common cause of complexity in modal vectors. Due to incorrect determination of the poles, leakage can lead to complex modal vectors and incorrect estimations of frequency and damping.
 - Mass Loading: If the transducer used for the measurement is large enough to affect the mass matrix of the system when moved across the structure, it leads to mass loading which can cause complex modal vectors.
 - Noise: The presence of experimental noise pollutes the data which on processing leads to complex modes. Such complex modal vectors are computational.

2. Errors made during parameter estimation [4]: The errors made during parameter estimation lead to invalid frequency and damping estimates which cause complex modal vectors. The common errors are,
 - choosing a frequency range such that there are too few or too many poles generally results into an incorrect damping estimate.
 - choosing a frequency range which has poles too close to edge of the range especially the lower frequency, again, leads to incorrect estimation of damping.
3. Repeated roots: When there are two modes at nearly the same natural frequency, usually seen in axisymmetric structures, and their modal vectors have a phase difference, then the combination of those modal vectors which is valid modal vector for the eigenvalue will be complex [5][6].
4. Non-linearity: The complex modal vectors obtained from structures are generally assumed to be due to the non-proportional nature of the damping. However, this complexity could be due to the assumption of the structure being linear. M. Imregun and D. J. Ewins in [6] show that the non-linearity of the structure affects the modal parameters and it could lead to identification of false complex modes.
5. Gyroscopic and aerodynamic effects also can lead to complex modal vectors [6]

2.3 Other Normalization Techniques

Over the years, several normalization techniques have been proposed. Some of those techniques are discussed briefly in this section.

2.3.1 Computation of Normal Modes from Identified Complex Modes

Assuming that the modal parameters, namely the complex modal frequencies, λ_r , and modal vectors, $\{\psi_r\}$, are known, the displacement, velocity and acceleration responses can be given as,

$$\{x(t)\} = \sum_{r=1}^{2N} c_r \{\psi\}_r e^{\lambda_r t} \quad (2.1)$$

$$\{\dot{x}(t)\} = \sum_{r=1}^{2N} \lambda_r c_r \{\psi\}_r e^{\lambda_r t} \quad (2.2)$$

$$\{\ddot{x}(t)\} = \sum_{r=1}^{2N} \lambda_r^2 c_r \{\psi\}_r e^{\lambda_r t} \quad (2.3)$$

where,

$\{\psi\}_r$ is the modal vector with size of N_o

c_r is the complex modal scaling coefficient of $\{\psi\}_r$

λ_r is the modal frequency

$r = 1, 2 \dots N$

The characteristic equation for a mechanical multi degree of freedom system is given as,

$$[M]\{\ddot{x}(t)\} + [C]\{\dot{x}(t)\} + [K]\{x(t)\} = \{f(t)\} \quad (2.4)$$

where,

$\{f(t)\}$ is the vector of external forces

Equation (2.4) can also be represented as,

$$\begin{Bmatrix} \dot{x}(t) \\ \dot{x}(t) \end{Bmatrix} = \begin{bmatrix} [0] & [I] \\ -[M]^{-1}[K] & -[M]^{-1}[C] \end{bmatrix} \begin{Bmatrix} x(t) \\ \dot{x}(t) \end{Bmatrix} \quad (2.5)$$

Equation (2.5) can be represented as,

$$\{\dot{y}(t)\} = [E]\{y(t)\} \quad (2.6)$$

where,

$\{y(t)\}$ is the state vector of the system

$[E]$ is the companion matrix (refer Subsection (3.2.1))

They can be given as follows [7],

$$\{y(t)\} = \begin{Bmatrix} x(t) \\ \dot{x}(t) \end{Bmatrix} \quad [E] = \begin{bmatrix} [0] & [I] \\ -[M]^{-1}[K] & -[M]^{-1}[C] \end{bmatrix}$$

By repeating Equation (2.6) for $2N_o$ time instants, the equation can be modified as,

$$\begin{bmatrix} \dot{y}(t_1) & \dot{y}(t_2) & \cdots & \dot{y}(t_{2N_o}) \end{bmatrix} = [E] \begin{bmatrix} y(t_1) & y(t_1) & \cdots & y(t_{2N_o}) \end{bmatrix} \quad (2.7)$$

where,

$t_1, t_2 \cdots t_{2N_o}$ represent $2N_o$ time instants

The companion matrix can be computed from Equation (2.7) as follows,

$$[E]_{2N_o \times 2N_o} = [\dot{Y}]_{2N_o \times 2N_o} [Y]_{2N_o \times 2N_o}^{-1} \quad (2.8)$$

Once the companion matrix is computed, the real normal modal vectors and the undamped natural frequency of the system can be found using the following equation,

$$\left[[M]^{-1}[K] \right] [\Phi] = \Omega_r^2 [\Phi] \quad (2.9)$$

where,

$[\Phi]$ is the real modal vector matrix

Ω is the undamped natural frequency

In the physical coordinates, the number of measured degree of freedom (number of output sensors), N_o , is always more than the number of modes measured, N . This causes the $[Y]$ matrix to always be singular. Therefore, inverting the $[Y]$ matrix will be numerically difficult and unstable. To make the inversion of $[Y]$ possible, S. R. Ibrahim in [8] proposes adding small amount of noise as follows,

$$\{x(t)\} = \sum_{r=1}^{2N} \{p\}_r e^{\lambda_r t} + \sum_{r=2N+1}^{2N_o} \{N\}_r e^{\lambda_r t} \quad (2.10)$$

where,

$\{p(t)\}$ represents the N complex modal vector pairs

$\{N\}$ is the noise modeled as a combination of $(2N_o - 2N)$ complex exponential functions

Over the years, several different techniques have been proposed to overcome the problem of singularity of the $[Y]$ matrix. Two of these techniques have been discussed in Subsections (2.3.2) and (2.3.3). For more techniques refer [9].

2.3.2 A Time domain Subspace Iteration Method

M. L. Wei et. al. in [10] have proposed defining a new matrix $[\hat{\Phi}]$ of size $N_o \times N$ which is the transformation matrix of the identified complex modal vector, $[\psi]$, which can be given as,

$$[\psi] = [\hat{\Phi}][W] \quad (2.11)$$

where, $[W]$ is a $N \times N$ complex matrix and can be determined by pseudo inverse technique,

$$[W] = [\hat{\Phi}]^+[\psi] \quad (2.12)$$

The newly defined matrix, $[\hat{\Phi}]$, cannot be selected arbitrarily and has to satisfy the following three conditions,

1. It should not be orthogonal to the real modal vectors of its corresponding undamped system.
2. It should be close to $[\Phi]$.
3. Its rank should be N .

The displacement response of the system can be given as,

$$\{x(t)\} = \sum_{r=1}^{2N} c_r \{\psi\}_r e^{\lambda_r t} \quad (2.13)$$

This displacement vector in the physical coordinates can also be transformed to the N dimensional subspace as follows,

$$\{x(t)\} = [\hat{\Phi}]\{p(t)\} \quad (2.14)$$

where,

$\{p(t)\}$ is the transformed displacement vector

$\{p(t)\}$ can be determined, again, using the pseudo inverse technique,

$$\{p(t)\} = [\hat{\Phi}]^+ \{x(t)\} \quad (2.15)$$

Substitute Equation (2.11) and (2.13) in Equation (2.15),

$$\begin{aligned} \{p(t)\} &= [\hat{\Phi}]^+ 2Re\{[\psi]\{e^{\lambda_r t}\}\} \\ &= 2[\hat{\Phi}]^+ [\Phi] Re\{[W]\{e^{\lambda_r t}\}\} \\ &= 2Re\{[W]\{e^{\lambda_r t}\}\} \end{aligned} \quad (2.16)$$

where,

' Re ' represents the real part of the matrix

Similarly, the velocity and acceleration can be given as,

$$\{\dot{p}(t)\} = 2Re\{[W]\{\lambda e^{\lambda_r t}\}\} \quad (2.17)$$

$$\{\ddot{p}(t)\} = 2Re\{[W]\{\lambda^2 e^{\lambda_r t}\}\} \quad (2.18)$$

The companion matrix can now be computed as follows,

$$\{q(t)\} = [E_N] \{\dot{q}\} \quad (2.19)$$

where,

$\{q(t)\}$ is the state vector in the transformed subspace

which can be given as,

$$\{q(t)\} = \begin{Bmatrix} p(t) \\ \dot{p}(t) \end{Bmatrix} \quad \{\dot{q}(t)\} = \begin{Bmatrix} \dot{p}(t) \\ \ddot{p}(t) \end{Bmatrix} \quad (2.20)$$

And $[E_N]$ is the companion matrix,

$$[E_N] = \begin{bmatrix} [0] & [I] \\ -[M_N]^{-1}[K_N] & -[M_N]^{-1}[C_N] \end{bmatrix} \quad (2.21)$$

where,

$[M_N]$, $[C_N]$ and $[K_N]$ are the reduced mass, damping and stiffness matrices respectively

These matrices can be given as,

$$[M_N]_{N \times N} = [\hat{\phi}]^T [M] [\hat{\phi}] \quad (2.22)$$

$$[C_N]_{N \times N} = [\hat{\phi}]^T [C] [\hat{\phi}] \quad (2.23)$$

$$[K_N]_{N \times N} = [\hat{\phi}]^T [K] [\hat{\phi}] \quad (2.24)$$

Once $\{q(t)\}$ and $\{\dot{q}(t)\}$ are computed at $2N$ time instants, $[E_N]$ can be computed,

$$\begin{bmatrix} \dot{q}(t_1) & \dot{q}(t_2) & \cdots & \dot{q}(t_{2N}) \end{bmatrix} = [E_N] \begin{bmatrix} q(t_1) & q(t_2) & \cdots & q(t_{2N}) \end{bmatrix} \quad (2.25)$$

$$[E_N] = [\dot{Q}][Q]^{-1} \quad (2.26)$$

Since $[Q]$ has a full rank of $2N$, it is invertible and the companion matrix $[E_N]$ can be computed. The $N \times N$ matrix $[M]^{-1}[K]$ can be easily determined from the companion matrix which then can be used to compute the normal modes using Equation (2.9).

2.3.3 Principal Response Analysis Approach

In the same paper, [10], the authors have proposed another technique of determining the companion matrix and this technique is also based on defining the transformation matrix, $[\hat{\Phi}]$, which transforms the complex modal matrix $[\psi]$. In this case, using the Singular Value Decomposition (SVD) technique, the $[\hat{\Phi}]$ matrix was decomposed as follows,

$$[\hat{\Phi}] = [P][\Sigma][S]^H \quad (2.27)$$

where,

$[P]$ is the orthonormal matrix ($N_o \times N$)

$[\Sigma]$ is the diagonal matrix which consists of eigenvalues of $[\hat{\Phi}]^H[\hat{\Phi}]$ ($N \times N$)

$[S]$ is unitary matrix which consists of eigenvectors of $[\hat{\Phi}]^H[\hat{\Phi}]$ ($N \times N$)

The matrix $[P]$ can be used to transform displacement from the physical coordinates to the N dimensional subspace as follows,

$$\{x(t)\} = [P]\{p(t)\} \quad (2.28)$$

$$\{p(t)\} = [P]^H\{x(t)\} \quad (2.29)$$

Substitute Equation (2.13) and (2.27) in (2.29),

$$\begin{aligned} \{p(t)\} &= [P]^H 2Re\{[\psi]\{e^{\lambda_r t}\}\} \\ &= 2[P]^H[\hat{\phi}]Re\{[W]\{e^{\lambda_r t}\}\} \\ &= 2[P]^H[P][\Sigma][S]^H Re\{[W]\{e^{\lambda_r t}\}\} \\ &= 2[\Sigma][S]^H Re\{[W]\{e^{\lambda_r t}\}\} \end{aligned} \quad (2.30)$$

Similarly, velocity and acceleration can be given as,

$$\{\dot{p}(t)\} = 2[\Sigma][S]^H Re\{[W]\{\lambda e^{\lambda_r t}\}\} \quad (2.31)$$

$$\{\ddot{p}(t)\} = 2[\Sigma][S]^H Re\{[W]\{\lambda^2 e^{\lambda_r t}\}\} \quad (2.32)$$

The companion matrix $[E_N]$ can now be computed using Equations (2.19) to (2.21). The reduced mass, damping and stiffness matrices in Equation (2.21) can be given, in this case, as follows,

$$[M_N]_{N \times N} = [P]^H [M] [P] \quad (2.33)$$

$$[C_N]_{N \times N} = [P]^H [C] [P] \quad (2.34)$$

$$[K_N]_{N \times N} = [P]^H [K] [P] \quad (2.35)$$

The procedure in Subsection (2.3.2) can then be followed to compute the real normal modes of the system.

2.3.4 Rescaling of Complex Modal Vectors Technique

In [9], S. Sinha proposed a technique of normalization of complex modes obtained from an undamped or proportionally damped system perturbed to the first order by a non-proportional damping matrix. In this technique, the largest modal coefficient corresponding to each of the modal frequencies was rescaled such that it is $1 + 0j$. After scaling, the imaginary part becomes negligible and is discarded. The real modal vectors retain the underlying characteristic of the mode shape. The proof is given below.

The characteristic equation for a MDOF damped system is,

$$(\lambda_r^2 [M] + \lambda_r [C] + [K])\{\psi\}_r = \{0\} \quad (2.36)$$

The equation for the corresponding undamped system will be,

$$(\Omega_r^2[M] + [K])\{\Phi\}_r = \{0\} \quad (2.37)$$

where,

$\{\Phi\}$ and $\{\psi\}$ is the real and complex modal vector respectively

Ω and λ is the undamped natural frequency and the modal frequency of the system respectively

The non-proportional damping introduced in an undamped or proportionally damped system can be represented as,

$$[C] = \epsilon[\bar{C}] \quad (2.38)$$

where,

$[\bar{C}]$ is the damping matrix of the proportionally damped system

$[C]$ is the damping matrix of the non-proportionally damped system

ϵ is the small perturbation

The complex modal vectors obtained were then scaled such that the largest modal coefficient is $1 + 0j$. Due to the introduction of non-proportionality in the system, it was assumed that the natural frequencies and modal vectors were affected in the following way,

$$[\lambda] = j[\Omega] + \epsilon[L] \quad (2.39)$$

$$[\psi] = [\phi] \left[[I] + j\epsilon[P] \right] \quad (2.40)$$

The matrices $[L]$ and $[P]$ are assumed complex initially, but will be proved real due to the orthogonality relations between modal vectors. Substituting Equations (2.38), (2.39) and (2.40) in Equation (2.36),

$$[M][\phi] \left[[I] + j[\epsilon][P] \right] \left[j[\Omega] + \epsilon[L] \right]^2 + \epsilon[\bar{C}][\phi] \left[[I] + j\epsilon[P] \right] \left[j[\Omega] + \epsilon[I] \right] + [K][\phi] \left[[I] + j\epsilon[p] \right] = 0 \quad (2.41)$$

Equating all the coefficients of $\epsilon(\epsilon^0, \epsilon^1, \epsilon^2, \epsilon^3)$ to zero,

$$-[M][\phi][\Omega^2] + [K][\phi] = 0 \quad (2.42)$$

$$2[M][\phi][L][\Omega] - [M][\phi][P][\Omega^2] + [K][\phi][P] = 0 \quad (2.43)$$

$$[M][\phi][L^2] - 2[M][\phi][P][L][\Omega] + [\bar{C}][\phi][L] - [\bar{C}][\phi][P][\Omega] = 0 \quad (2.44)$$

$$[M][\phi][P][L^2] + [\bar{C}][\phi][P][L] = 0 \quad (2.45)$$

The real modal vectors follow the orthogonality conditions, hence,

$$[\phi]^T [M][\phi] = [M_r] \quad (2.46)$$

$$[\phi]^T [\bar{C}][\phi] = [C_r] \quad (2.47)$$

$$[\phi]^T [K][\phi] = [K_r] \quad (2.48)$$

Pre-multiplying Equations (2.42) through (2.45) with $[\phi]^T$ and applying Equations (2.46) through (2.48),

$$-[M_r][\Omega^2] + [K_r] = 0 \quad (2.49)$$

$$2[M_r][L][\Omega] - [M_r][P][\Omega^2] + [C_r][\Omega] + [K_r][P] = 0 \quad (2.50)$$

$$[M_r][L^2] - 2[M_r][P][L][\Omega] + [C_r][L] - [C_r][P][\Omega] = 0 \quad (2.51)$$

$$[M_r][P][L^2] + [C_r][P][L] = 0 \quad (2.52)$$

Substituting $[K_r]$ from Equation (2.49) into Equation (2.50) and re-arranging,

$$2[M_r][L][\Omega] + [C_r][\Omega] + [M_r] \left[[\Omega^2][P] - [P[\Omega^2]] \right] = 0 \quad (2.53)$$

The diagonal elements of the last term of Equation (2.53), $[M_r] \left[[\Omega^2][P] - [P[\Omega^2]] \right]$, will be zero. The matrices $[M_r]$, $[C_r]$ and $[\Omega]$ are all real diagonal matrices. Therefore, if the modal assurance criterion (2.53) has to hold true, the matrix $[L]$ has to be a real diagonal matrix. From Equations (2.53), (2.51) and (2.52), it can also be concluded that the matrix $[P]$ also has to be a real matrix. Therefore, the following conclusions can be made,

- $[L]$ is a real diagonal matrix. The effect of first order perturbation damping matrix on the modal frequencies is purely real.
- $[P]$ is a real matrix. The effect of first order perturbation damping matrix on the modal vectors is purely imaginary.

Therefore, if an undamped or proportionally damped system is perturbed with a first order damping, the real part of the modal vectors obtained by scaling the modal coefficients such that the largest modal coefficient corresponding to each mode is $1+0j$ and the damped natural frequencies can be considered as a good approximation of the real modal vectors and undamped natural frequencies of its corresponding undamped system.

Chapter 3

Overview of Experimental Modal Analysis

In this chapter, the Unified Matrix Polynomial Approach [11] to experimental modal analysis will be discussed. It is a technique of representing the most commonly used modal parameter estimation algorithms under a common framework. This approach highlights the similarities between all the different algorithms [12]. The normalization technique which will be discussed in the following chapter is based on the α -polynomial which is formulated in this model.

3.1 Frequency Response Function

The characteristic equation of a MDOF system in the Laplace domain can be given as,

$$[[M]s^2 + [C]s + [K]]\{X(s)\} = \{F(s)\} \quad (3.1)$$

And, the transfer function of the system is given as,

$$[H_{pq}(s)] = \frac{\{X_p(s)\}}{\{F_q(s)\}} = [[M]s^2 + [C]s + [K]]^{-1} \quad (3.2)$$

where,

$[M]$, $[C]$ and $[K]$ are the mass, damping and stiffness matrices respectively

$\{X_p(s)\}$ is the response at point ‘p’

$\{F_q(s)\}$ is the input force at point ‘q’

3.1.1 Polynomial Model

The transfer function can also be represented in a polynomial form. The numerator and denominator both are represented by two polynomials with independent variable ‘s’. The coefficients of these polynomials are different numerical combinations of the discrete values of mass, damping and stiffness of the system. The roots of the denominator are the modal frequencies of the system. The polynomial form for a single input single output (SISO) is given as,

$$H_{pq} = \frac{X_p(s)}{F_q(s)} = \frac{\beta_n(s)^n + \beta_{n-1}(s)^{n-1} + \dots + \beta_1(s)^1 + \beta_0(s)^0}{\alpha_m(s)^m + \alpha_{m-1}(s)^{m-1} + \dots + \alpha_1(s)^1 + \alpha_0(s)^0} \quad (3.3)$$

This model can be written in concise form as follows,

$$\frac{\{X_p(s)\}}{\{F_q(s)\}} = \frac{\sum_{k=0}^n \beta_k(s)^k}{\sum_{k=0}^m \alpha_k(s)^k} \quad (3.4)$$

Re-arranging,

$$\sum_{k=0}^m \alpha_k(s)^k \{X_p(s)\} = \sum_{k=0}^n \beta_k(s)^k \{F_q(s)\} \quad (3.5)$$

For a multiple input, multiple output system, the model can be written as follows,

$$\sum_{k=0}^m [[\alpha_k](s)^k] \{X_p(s)\} = \sum_{k=0}^n [[\beta_k](s)^k] \{F_q(s)\} \quad (3.6)$$

By substituting $s = j\omega$, the model can be represented in frequency domain as follows,

$$\sum_{k=0}^m [[\alpha_k](j\omega)^k] \{X_p(j\omega)\} = \sum_{k=0}^n [[\beta_k](j\omega)^k] \{F_q(j\omega)\} \quad (3.7)$$

To obtain the frequency response function (FRF) of the system, both sides of the equation can be multiplied with $\{F_q(j\omega)\}^H$. This gives output-input cross power on the left hand side and output-output cross power matrix on the right hand side as follows,

$$\sum_{k=0}^m [[\alpha_k](j\omega)^k] [G_{pq}(j\omega)] = \sum_{k=0}^n [[\beta_k](j\omega)^k] [G_{qq}(j\omega)] \quad (3.8)$$

Then, the equation can be post-multiplied with $[G_{qq}(j\omega)]^H$ to obtain the frequency response function of the system [11],

$$\sum_{k=0}^m [[\alpha_k](j\omega)^k] [H_{pq}(\omega)] = \sum_{k=0}^n [[\beta_k](j\omega)^k] [I] \quad (3.9)$$

The dimensions of the $[\alpha]$, $N_i \times N_i$ or $N_o \times N_o$ depends on the type of algorithm chosen. More details can be found in Subsection (3.2.3)

3.2 Matrix Coefficient Polynomial

The modal frequencies and modal vectors of the system can be estimated from the denominator of Equation (3.2) as stated in the Subsection (3.1.1). For a MIMO system, the modal parameters can be found by solving for the roots of the following polynomial with matrix coefficients,

$$\left| [\alpha_m](s)^m + [\alpha_{m-1}](s)^{m-1} + \cdots + [\alpha_1](s)^1 + [\alpha_0](s)^0 \right| = 0 \quad (3.10)$$

The roots of the polynomial, λ_r , can be given as,

$$s_r = \lambda_r = \sigma_r \pm j\omega_r \quad (3.11)$$

where,

σ_r is the modal damping corresponding to the mode r

ω_r is the damped natural frequency corresponding to mode r

The poles of a single degree of freedom system can be represented in a s -plane as shown in Figure (3.1). The damping ratio, i.e. the ratio of the actual damping in the system to the critical damping, can be given as,

$$\zeta = \frac{C}{C_c} = -\frac{\sigma_r}{\Omega_r} \quad (3.12)$$

where,

ζ is the damping ratio

C is the actual damping of the system

C_c is the critical damping of the system

Ω_r is the undamped natural frequency of the system for the r^{th} mode

Figure 3.1: Pole in s-plane

3.2.1 Companion Matrix

To find the roots of a polynomial, companion matrices are often used. The eigenvalue decomposition of a companion matrix gives the modal frequencies and vectors of the system. There are different ways in which a companion matrix can be formulated. For instance, the companion matrix of a highest order normalized polynomial as follows,

$$[I](s)^m + [\alpha_{m-1}](s)^{m-1} + [\alpha_{m-2}](s)^{m-2} + \cdots + [\alpha_1](s)^1 + [\alpha_0](s)^0 = 0 \quad (3.13)$$

can be given as,

$$[C] = \begin{bmatrix} -[\alpha_{m-1}] & -[\alpha_{m-2}] & \cdots & -[\alpha_1] & -[\alpha_0] \\ [I] & [0] & \cdots & [0] & [0] \\ [0] & [I] & \cdots & [0] & [0] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [0] & [0] & \cdots & [I] & [0] \end{bmatrix} \quad (3.14)$$

The developed companion matrix can be used in the following eigenvalue decomposition formulation to obtain the modal frequencies and vectors,

$$[C][\phi] = [\lambda][I][\phi] \quad (3.15)$$

where,

$[\phi]$ is the eigenvector matrix

$[\lambda]$ is a diagonal matrix of modal frequencies

The companion matrix of a polynomial whose lowest order is normalized similar to the equation shown below,

$$[\alpha_m](s)^m + [\alpha_{m-1}](s)^{m-1} + [\alpha_{m-2}](s)^{m-2} + \cdots + [\alpha_1](s)^1 + [I](s)^0 = 0 \quad (3.16)$$

can be given as,

$$[C] = \begin{bmatrix} -[\alpha_{m-1}] & -[\alpha_{m-2}] & \cdots & -[\alpha_1] & -[I] \\ [I] & [0] & \cdots & [0] & [0] \\ [0] & [I] & \cdots & [0] & [0] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [0] & [0] & \cdots & [I] & [0] \end{bmatrix} \quad (3.17)$$

And the roots can be obtained by solving,

$$[C][\phi] = [\lambda] \begin{bmatrix} [\alpha_m] & [0] & \cdots & [0] & [0] \\ [0] & [I] & \cdots & [0] & [0] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [0] & [0] & \cdots & [I] & [0] \\ [0] & [0] & \cdots & [0] & [I] \end{bmatrix} [\phi] \quad (3.18)$$

If the α matrices are of size $n \times n$, and the order of the polynomial is m , then there will be $n \times m$ modal frequencies, λ . The eigenvector, $[\phi]$, will have the size $(n \times m) \times (n \times m)$. The eigenvectors will not be the actual modal vectors of the system. The meaningful modal vectors will be the last n rows of the $[\phi]$ matrix. The remaining rows are the meaningful modal vectors multiplied by the modal frequencies raised to an increasing

power from 1 to $(m - 1)$. For the r^{th} mode it can be given as follows,

$$\left\{ \phi_r \right\} = \begin{pmatrix} \lambda_r^{m-1} \left\{ \psi \right\}_r \\ \lambda_r^{m-2} \left\{ \psi \right\}_r \\ \vdots \\ \vdots \\ \lambda_r^1 \left\{ \psi \right\}_r \\ \lambda_r^0 \left\{ \psi \right\}_r \end{pmatrix} \quad (3.19)$$

3.2.2 Determining the α Matrices

In real world system, the mass, damping, and stiffness matrices are not known. However, the frequency response function (FRF) of the system can be obtained. Using the FRF matrix of a system, the α matrices can be obtained. Re-arranging Equation (3.9),

$$\sum_{k=0}^m [[\alpha_k](j\omega)^k][H(\omega)] - \sum_{k=0}^n [[\beta_k](j\omega)^k][I] = 0 \quad (3.20)$$

In the Equation (3.2.2), the frequency response of the system is known. To determine the α and β matrices, one of the unknown coefficients can be normalized to an identity matrix. There are two common ways of normalizing,

- High Order normalization: $[\alpha_m] = [I]$

$$\sum_{k=0}^{m-1} [[\alpha_k](j\omega)^k][H_{pq}(\omega)] - \sum_{k=0}^n [[\beta_k](j\omega)^k][I] = -[[\alpha_m](j\omega)^m][H_{pq}(\omega)] \quad (3.21)$$

- Low Order normalization: $[\alpha_0] = [I]$

$$\sum_{k=1}^m [[\alpha_k](j\omega)^k][H_{pq}(\omega)] - \sum_{k=0}^n [[\beta_k](j\omega)^k][I] = -[[\alpha_0](j\omega)^0][H_{pq}(\omega)] \quad (3.22)$$

3.2.3 Frequency Domain Algorithms

Equation (3.21) can be re-arranged in various matrix arrangements depending on the order of the algorithm. The basic matrix equation for a second order algorithm like Polyreference Frequency Domain (PFD) is,

$$\begin{bmatrix} [\alpha_0] & [\alpha_1] & [\beta_0] & [\beta_1] \end{bmatrix} \begin{bmatrix} (s_i)^0[H(\omega_i)] \\ (s_i)^1[H(\omega_i)] \\ -(s_i)^0[I] \\ -(s_i)^1[I] \end{bmatrix} = -(s_i)^2[H(\omega_i)] \quad (3.23)$$

The size of the matrix coefficients α and β in the case will be $N_l \times N_l$, where, N_l is the largest dimension of the FRF matrix. The companion matrix for this case can be given as,

$$[C] = \begin{bmatrix} -[\alpha_1] & -[\alpha_0] \\ [I] & [0] \end{bmatrix} \quad (3.24)$$

And, the generalized form for any high order frequency domain algorithms like Rational Fraction Polynomial (RFP) is,

$$\begin{bmatrix} [\alpha_0] & [\alpha_1] & \cdots & [\alpha_{m-1}] & [\beta_0] & [\beta_1] & \cdots & [\beta_n] \end{bmatrix} \begin{bmatrix} (s_i)^0[H(\omega_i)] \\ (s_i)^1[H(\omega_i)] \\ \vdots \\ (s_i)^m[H(\omega_i)] \\ -(s_i)^0[I] \\ -(s_i)^1[I] \\ \vdots \\ -(s_i)^n[I] \end{bmatrix} = -(s_i)^m[H(\omega_i)] \quad (3.25)$$

The size of the matrix coefficients α and β in this case will be $N_s \times N_s$. Where, N_s is the smallest dimension of the FRF matrix. The companion matrix for this case can be given as follows,

$$[C] = \begin{bmatrix} -[\alpha_{m-1}] & -[\alpha_{m-2}] & \cdots & -[\alpha_1] & -[\alpha_0] \\ [I] & [0] & \cdots & [0] & [0] \\ [0] & [I] & \cdots & [0] & [0] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [0] & [0] & \cdots & [I] & [0] \end{bmatrix} \quad (3.26)$$

The size of the matrix coefficients α and β in the second order algorithm will be $N_l \times N_l$ and in the higher order case they will be $N_s \times N_s$. Where, N_l and N_s are the largest and smallest dimension of the FRF matrix.

3.3 Coefficient Condensation

Typically, for lower order algorithms, the number of physical outputs i.e. the number of output sensors in the long dimension, N_l , is much more than the number of modes, N . The size of long dimension dictates the size of the alpha coefficient matrices, $[\alpha]$. Which causes the system to solve for N_l modal frequencies which is time consuming. There are two alternative techniques namely the Singular Value Decomposition (SVD) and Eigenvalue Decomposition (ED) which can reduce the physical coordinates to a virtual modal space with approximate number of effective modal frequencies, N_e ($N_e \approx N$). These techniques reduce the coordinates without losing the principal modal information [12][11].

Both the techniques give a transformation matrix, $[T]$, containing complex valued modal vectors, which can be used to transform from physical to virtual coordinate system. If the modal density and/or the level of damping increases and if the complete FRF, $[H]$, matrix is used for computation, both, SVD and ED will give erroneous results. To get better results, the transformation matrix, $[T]$, is forced to be real valued by using just the imaginary part of the FRF matrix in the SVD and ED techniques.

The difference between the two techniques is in the way in which the transformation matrix is computed. Once, the transformation matrix is calculated, FRF matrix can be condensed as,

$$[\bar{H}] = [T][H] \quad (3.27)$$

where,

$[\bar{H}]$ is the transformed FRF matrix

$[T]$ is the transformation matrix

$[H]$ is the original FRF matrix

Once the FRF matrix is condensed using Equation (3.27), the same parameter estimation technique as the one used for a full size matrix will be used to determine the modal parameters. The poles obtained from the condensed data are the same as the ones from the full data set. The condensed modal vectors, however, need to be expanded back to the physical space as follows,

$$[\psi] = [T]^T [\bar{\psi}] \quad (3.28)$$

where,

$[\psi]$ is the full size modal vector

$[\bar{T}]$ is the condensed modal vector

3.3.1 Eigenvalue Decomposition

In this technique, the transformation matrix, $[T]$, is the modal vectors corresponding to N_e largest eigenvalues of the power spectrum of the FRF matrix as follows,

$$[H(\omega)]_{N_o \times N_i N_s} [H(\omega)]_{N_i N_s \times N_o}^H = [V][\Lambda][V]^H \quad (3.29)$$

The transformation matrix is then given by the eigenvectors corresponding to the N_e largest eigenvalues,

$$[T]_{N_e \times N_o} = \left[\{v_1\} \quad \{v_2\} \quad \cdots \quad \{v_{N_e}\} \right]^T \quad (3.30)$$

The input subspace can also be condensed using the same technique. In this case, however, the FRF matrix will have to be transposed such that it is $N_i \times N_o$ before the power

spectrum is found,

$$[H(\omega)]_{N_o \times N_i N_s} [H(\omega)]_{N_i N_s \times N_o}^H = [V][\Lambda][V]^H \quad (3.31)$$

The transformation matrix can be found similar to Equation (3.30),

$$[T]_{N_e \times N_i} = \begin{bmatrix} \{v_1\} & \{v_2\} & \cdots & \{v_{N_e}\} \end{bmatrix}^T \quad (3.32)$$

3.3.2 Singular Value Decomposition

This technique operates directly on the FRF matrix instead of the power spectrum. The FRF matrix, $[H]$, will be decomposed into three matrices as follows,

$$[H]_{N_o \times N_i N_s} = [U]_{N_o \times N_o} [\Sigma]_{N_o \times N_i N_s} [V]_{N_i N_s \times N_i N_s}^H \quad (3.33)$$

where,

$[U]$ is the left singular vector

$[\Sigma]$ is a diagonal matrix consisting the singular values

$[V]$ is the right singular vector

The transformation matrix can be obtained from the first N_e columns of the left singular matrix, $[U]$, which corresponds to the N_e largest singular values,

$$[T]_{N_e \times N_o} = \begin{bmatrix} \{u_1\} & \{u_2\} & \cdots & \{u_{N_e}\} \end{bmatrix}^T \quad (3.34)$$

Similarly, the input subspace can be condensed by transposing the FRF matrix,

$$[H]_{N_i \times N_o N_s} = [U]_{N_i \times N_i} [\Sigma]_{N_i \times N_o N_s} [V]_{N_o N_s \times N_o N_s}^H \quad (3.35)$$

The transformation matrix in this case can be given as,

$$[T]_{N_e \times N_i} = \begin{bmatrix} \{u_1\} & \{u_2\} & \cdots & \{u_{N_e}\} \end{bmatrix}^T \quad (3.36)$$

3.4 Modal Assurance Criterion

Modal assurance criterion (MAC) [13] is used to determine the degree of linearity between two modal vectors. As long as the dimensions of two modal vectors are same, MAC can be defined. A common application of MAC is to compare experimental modal vectors with the ones obtained from different sources like analytically developed modal vectors from finite element (FE) models. MAC can be formulated as follows,

$$MAC_{cdr} = \frac{\{\psi_{cr}\}^H \{\psi_{dr}\} \{\psi_{dr}\}^H \{\psi_{cr}\}}{\{\psi_{cr}\}^H \{\psi_{cr}\} \{\psi_{dr}\}^H \{\psi_{dr}\}} \quad (3.37)$$

where,

$\{\psi_{cr}\}$ is the modal vector corresponding to the r^{th} mode and degree of freedom c

$\{\psi_{dr}\}$ is the modal vector corresponding to the r^{th} mode and degree of freedom d

The MAC can take values between 0 and 1. If the MAC value is near zero, the two modal vectors can be said to have no linear dependency with each other i.e. they are linearly independent. If the MAC value is close to 1, the modal vectors can be said to have a linear relationship with each other i.e. the modal vectors represent the same modal vector with a different arbitrary scaling.

MAC will be used in thesis to compare the modal vectors obtained after truncating with the original experimental modal vectors. As the modal vectors obtained after truncation still correspond to the same mode, a high MAC value is desirable.

Chapter 4

Normalization Techniques

A normalization technique based on the α -polynomial will be discussed in this chapter. The normalized mode shapes and modal frequencies will be compared with the experimentally obtained mode shapes and modal vectors.

4.1 Normalization by Truncation of the α Polynomial

4.1.1 Background

For a multiple degree of freedom system, the characteristic equation is given as shown in Equation (3.1),

$$[[M]s^2 + [C]s + [K]]\{X(s)\} = \{F(s)\}$$

In the above equation, the damping associated with the system is given by the $[C]$ matrix. Therefore, on eliminating this matrix, the damping can be eliminated from the system which will give the undamped natural frequencies and real mode shapes, as can be seen

the damping matrix is the coefficient of the odd powered variable, s^1 . Therefore, it can be said that by eliminating the odd powered term in the polynomial, the undamped solution for the system can be obtained. This understanding will be tested further in next few sections on polynomials with an order larger than two.

4.1.2 Elimination of Odd Powered Terms

Using the UMPA model, the second order characteristic system equation is expanded to an α -polynomial during the modal parameter estimation. The order of this α -polynomial varies with every iteration. For the RFP algorithm, the lowest order of the polynomial will be two (i.e. iteration 1). Therefore at iteration 3, the order of the polynomial will be four. The α -polynomial with order four can given as,

$$\left| [\alpha_4](s)^4 + [\alpha_3](s)^3 + [\alpha_2](s)^2 + [\alpha_1](s)^1 + [\alpha_0](s)^0 \right| = 0 \quad (4.1)$$

The α matrices for this case can be estimated as follows,

$$\begin{bmatrix} [\alpha_0] & [\alpha_1] & [\alpha_2] & [\alpha_3] & [\beta_0] & [\beta_1] & [\beta_2] \end{bmatrix} \begin{bmatrix} (s_i)^0[H(\omega_i)] \\ (s_i)^1[H(\omega_i)] \\ (s_i)^2[H(\omega_i)] \\ (s_i)^3[H(\omega_i)] \\ -(s_i)^0[I] \\ -(s_i)^1[I] \\ -(s_i)^2[I] \end{bmatrix} = -(s_i)^4[H(\omega_i)] \quad (4.2)$$

After eliminating the odd powered terms in the polynomial Equation (4.1), it reduces to,

$$\left| [\alpha_4](s)^4 + [\alpha_2](s)^2 + [\alpha_0](s)^0 \right| = 0 \quad (4.3)$$

The companion matrix for this polynomial will be,

$$[C] = \begin{bmatrix} [0] & -[\alpha_2] & [0] & -[\alpha_0] \\ [I] & [0] & [0] & [0] \\ [0] & [I] & [0] & [0] \\ [0] & [0] & [I] & [0] \end{bmatrix} \quad (4.4)$$

The eigenvalue decomposition of Equation (4.4) will give the eigenvalues and eigenvectors through which the modal frequencies and mode shapes can be obtained.

4.1.3 Application on a Circular Plate

In this section, the normalization technique discussed in Section (4.1) will be used on the data obtained from a circular plate. The modal frequencies after truncation will be compared with the absolute values of the modal frequencies before truncation i.e. the undamped natural frequencies of the original system. The modal vectors before and after truncation will also be compared using the modal assurance criterion (MAC).

4.1.4 Experimental Setup and Material Properties

A steel circular plate with diameter of 0.91 *m* (3 *ft*) and 0.20 *m* (8 *in*) diameter hub, thrice the thickness of the circular plate fixed at the center, with 7 references was excited with an impact hammer at 36 points on the plate. The data was acquired from 0 to 2555 Hz with Δf of 5 Hz. The FRF data for the circular plate was obtained using this setup.

Since the setup had 36 inputs and 7 outputs and 512 frequency points, the dimension of the FRF matrix was $7 \times 36 \times 512$.

4.1.5 Pole Setup and Modal Parameter Estimation

For ease of processing and to obtain stable poles the entire frequency range was not processed at once. If too big a frequency range is selected, the number of frequency points will be very high causing the over determination factor to be larger than desirable which may lead to unstable poles [12]. Therefore, two separate bandwidths were selected and processed. Of the two, the normalization technique was executed on the poles obtained from one of the two selected bandwidths and will be compared with the default poles obtained from X-Modal, a modal parameter estimation software package used at the University of Cincinnati, Structural Dynamics Research Laboratory (UC-SDRL). The bandwidth shown in Figure (4.1) was selected.

Figure 4.1: Frequency Response Function

The parameters of the selected bandwidth are shown in Table 4.1.

Minimum Frequency	145 Hz
Maximum Frequency	825 Hz
Minimum Time	0.0047 s
Maximum Time	0.0998 s
Number of Frequency Lines	137
Number of Time Points	131

Table 4.1: Parameters for the selected bandwidth

From the Complex Mode Indicator Function (CMIF) shown in Figure 4.2, it can be seen that, for the selected frequency range, there are five modes of the system. This data was

Figure 4.2: Complex Mode Indicator Function

processed using the RFP algorithm discussed in Subsection (3.2.3). Highest order of the polynomial was normalized. The poles obtained can be seen in fig 4.3. In X-Modal, for high order low basis polynomials, as explained in (4.1.2), the order of the polynomial starts with two and increases with each iteration (Y-axis in Figure (4.3)).

Figure 4.3: Modal Parameter Estimation

4.1.6 Even Powered Polynomials

For iteration 3, the order of the polynomial is 4. Therefore, for this iteration, there are five α matrices as can be seen in Equation (4.1). These matrices were extracted directly from X-Modal. X-Modal uses the same technique as explained in Subsection (3.2.3) to compute the α matrices. Since the FRF matrix was 7×36 and a high order short basis frequency domain algorithm was used, the α matrices obtained were of dimension 7×7 . The α matrices were then used to find the modal frequencies and mode shapes using the procedure explained in Subsection (4.1.2). The modal frequencies determined from this technique and the ones obtained from X-Modal, i.e. by considering the complete polynomial are compared in Table (4.2).

In Table 4.2, the modal frequencies in the first column are the original system poles obtained from X-Modal, i.e. the roots obtained by solving the complete polynomial of the system. In the second column are the absolute values of the corresponding modal frequencies in the first column. These absolute values are the undamped natural frequencies of the system. In the third column are the modal frequencies obtained after truncating

Original (Hz)	Absolute (Hz)	Truncated (Hz)	Deviation (%)
$-3.1498 + 362.3564i$	362.3701	$0.0000 + 362.7741i$	0.1115
$-3.1498 - 362.3564i$	362.3701	$0.0000 - 362.7741i$	0.1115
$-3.5025 + 363.6586i$	363.6755	$0.0000 + 363.7470i$	0.0197
$-3.5025 - 363.6586i$	363.6755	$0.0000 - 363.7470i$	0.0197
$-2.9191 + 557.0450i$	557.0526	$0.0000 + 556.7307i$	-0.0578
$-2.9191 - 557.0450i$	557.0526	$0.0000 - 556.7307i$	-0.0578
$-5.1902 + 761.2215i$	761.2392	$0.0000 + 761.4137i$	0.0229
$-5.1902 - 761.2215i$	761.2392	$0.0000 - 761.4137i$	0.0229
$-2.5540 + 764.1789i$	764.1831	$0.0000 + 764.1483i$	-0.0046
$-2.5540 - 764.1789i$	764.1831	$0.0000 - 764.1483i$	-0.0046

Table 4.2: Comparison of Modal Frequencies

the polynomial. As can be seen, the truncation technique eliminates the damping (real part of the modal frequency) but the obtained frequencies are not exactly the undamped natural frequencies of the system. The percent deviation can be seen in the fourth column of Table (4.2). This deviation in the natural frequency is explained in Chapter (5) with the help of 2 and 4 degree of freedom analytical models.

Modal vector complexity plots show the complexity in a particular mode shape or modal vector. The comparison of the complexity of modal vectors before and after truncation is shown in Figure (4.4). From the figure, it can be concluded that the modal vectors after truncation have been normalized. The mode shape comparisons are shown in Figures (4.5) to (4.9).

The modal assurance criterion for the modal vectors before and after truncation can be seen in Table (4.3). From the MAC, it can be seen that the modal vectors pre and

Frequency (Hz)	MAC
362	0.8455
364	0.9931
557	0.9998
761	0.9986
764	0.9994

Table 4.3: Modal Assurance Criterion (MAC)

Figure 4.4: Modal Vector Complexity Comparison

post truncation are very similar, except the one at 362 Hz , with a MAC value as high as 0.9994. For the RFP algorithm, the number of references are 7 which is probably not enough to identify the actual mode shape of the system which is causing the low MAC value. If the residues are synthesized and then compared, a better MAC should be obtained.

Therefore, from the MAC at other four frequencies, it can be concluded that the truncation technique works as far as normalization of complex mode shapes is concerned. However, as stated earlier, the undamped natural frequencies have a slight variation, the maximum being 0.1115%, from that of the system's.

4.1.7 Odd Powered Polynomials

The iteration 2 in the MPE shown in Figure (4.3) corresponds to the poles obtained by solving the polynomial with an order of 3 which can be given as follows,

$$\left| [\alpha_3](s)^3 + [\alpha_2](s)^2 + [\alpha_1](s)^1 + [\alpha_0](s)^0 \right| = 0 \quad (4.5)$$

The companion matrix for this polynomial can be given as,

$$[C] = \begin{bmatrix} -[\alpha_2] & -[\alpha_1] & -[\alpha_0] \\ [I] & [0] & [0] \\ [0] & [I] & [0] \end{bmatrix} \quad (4.6)$$

On using the elimination technique, the companion matrix will change to,

$$[C] = \begin{bmatrix} -[\alpha_2] & [0] & -[\alpha_0] \\ [I] & [0] & [0] \\ [0] & [I] & [0] \end{bmatrix} \quad (4.7)$$

But, in case of a companion matrix, as shown in Equation (3.13), the coefficient of the term with the highest order (or lowest, depending on the type of normalization chosen) is normalized to an identity matrix. Therefore, if the odd terms of any odd powered polynomial are eliminated from the companion matrix as shown in Equation (4.7), not all the odd powered terms will be eliminated. The term with the highest odd power which is not in the companion matrix is still present in the calculations. If all the odd powered terms are removed from the polynomial before forming the companion matrix, the polynomial will no longer be a third order one, it will be reduced to a second order polynomial. Therefore, in case of odd powered polynomials, this technique does not work.

Figure 4.5: Mode Shape at 362 Hz

(a) Default

(b) After Truncation

Figure 4.6: Mode Shape at 364 Hz

(a) Default

(b) After Truncation

Figure 4.7: Mode Shape at 557 Hz

(a) Default

(b) After Truncation

Figure 4.8: Mode Shape at 761 Hz

Figure 4.9: Mode Shape at 764 Hz

4.1.8 Application on Data Obtained Using Polyreference Frequency Domain - 2 (PFD - 2) Algorithm

The circular plate data was processed using the Polyreference Frequency Domain - 2 (PFD - 2) algorithm on the same bandwidth as shown in Figure (4.1) by setting the coefficient condensation off. The reason why coefficient condensation was set to off will be discussed in Chapter (5). The truncation technique was then applied on the α matrices obtained using PFD - 2 algorithm. Since PFD is a low order algorithm, the size of the α matrices in this case will be $N_l \times N_l$ i.e. 36×36 . Table (4.4) compares the modal frequencies obtained before and after applying the truncation technique.

Original (Hz)	Absolute (Hz)	Truncated (Hz)	Deviation (%)
$-3.1990 + 362.4229i$	362.4371	$0.0000 - 360.9763i$	-0.4030
$-3.1989 - 362.4229i$	362.4371	$0.0000 + 360.9763i$	-0.4030
$-2.4864 + 365.6700i$	365.6785	$0.0000 - 365.1040i$	-0.1571
$-2.4864 - 365.6700i$	365.6785	$0.0000 + 365.1040i$	-0.1571
$-2.9389 + 556.8717i$	556.8794	$0.0000 - 562.9960i$	1.0984
$-2.9389 - 556.8717i$	556.8794	$0.0000 + 562.9960i$	1.0984
$-5.1497 + 761.0664i$	761.0838	$0.0000 - 761.6937i$	0.0801
$-5.1496 - 761.0663i$	761.0838	$0.0000 + 761.6937i$	0.0801
$-2.8034 + 764.3707i$	764.3759	$0.0000 - 762.7500i$	-0.2127
$-2.8034 - 764.3707i$	764.3759	$0.0000 + 762.7500i$	-0.2127

Table 4.4: Comparison of Modal Frequencies

In Table (4.4), the first column shows the original modal frequencies of the system. The

frequencies in second column are the absolute values of the modal frequencies in the first column i.e. the undamped natural frequency of the system. In the third column are the modal frequencies obtained after applying the truncation technique. The last column shows the percent deviation in the undamped natural frequencies obtained pre and post truncation.

From the third column in Table (4.4) it can be seen that the truncation technique eliminates the damping (real part) from the modal frequencies and similar to the RFP case, the undamped natural frequencies deviate. The deviation however is very low, the maximum deviation being 1.0984 %.

The modal vector complexity plots are shown in Figure (4.10) and MAC values can be seen in Table (4.5).

Figure 4.10: Modal Vector Complexity Comparison

From the modal vector complexity plots in Figure (4.10), it can be concluded that the modal vectors have been successfully normalized. The modal vectors before and after truncation compare well except at frequencies 365 Hz and 764 Hz , where the MAC is comparatively less. Similar to the RFP case, better MAC can be obtained at those frequencies by synthesizing the residues.

Frequency	MAC
362	0.9746
365	0.8753
556	0.9845
761	0.9018
764	0.9874

Table 4.5: Modal Assurance Criterion (MAC)

4.2 Normalization by Manipulation of the Partial Fraction Method of Residue Synthesis

4.2.1 Background

Another method of representing a transfer function is through partial fractions. This is one of the methods of determining residues [12]. Equation (3.2), can be represented, for a single DOF system, as product of roots as follows,

$$H(s) = \frac{1}{Ms^2 + Cs + K} = \frac{1/M}{s^2 + \frac{C}{M}s + \frac{K}{M}} = \frac{1/M}{(s - \lambda)(s - \lambda^*)} \quad (4.8)$$

This equation can be represented in a partial fraction format as follows,

$$H(s) = \frac{A_1}{(s - \lambda_1)} + \frac{A_1^*}{(s - \lambda_1^*)} \quad (4.9)$$

Similarly, the frequency response function can be represented by substituting $s = j\omega$ as,

$$H(\omega) = \frac{A_1}{(j\omega - \lambda_1)} + \frac{A_1^*}{(j\omega - \lambda_1^*)} \quad (4.10)$$

where,

$$\lambda_1 = \sigma + j\omega; \lambda_1^* = \sigma - j\omega,$$

A_1, A_1^* are the residues associated with the pole λ_1

For a MDOF system, Equation (4.10), can be modified to,

$$H_{pq}(\omega) = \frac{A_{pq1}}{(j\omega - \lambda_1)} + \frac{A_{pq1}^*}{(j\omega - \lambda_1^*)} + \frac{A_{pq2}}{(j\omega - \lambda_2)} + \frac{A_{pq2}^*}{(j\omega - \lambda_2^*)} \quad (4.11)$$

This can be written in a compact form as,

$$H_{pq}(\omega) = \sum_{r=1}^2 \frac{A_{pqr}}{(j\omega - \lambda_r)} + \frac{A_{pqr}^*}{(j\omega - \lambda_r^*)} \quad (4.12)$$

Generalizing for n degrees of freedom,

$$H_{pq}(\omega) = \sum_{r=1}^n \frac{A_{pqr}}{(j\omega - \lambda_r)} + \frac{A_{pqr}^*}{(j\omega - \lambda_r^*)} \quad (4.13)$$

4.2.2 Using Partial Fraction Method to Obtain the Residues

For a system with a single reference, N modes and s spectral lines, Equation (4.13) can be expressed as,

$$\left\{ H_{pq}(\omega) \right\}_{N_s \times 1} = \left[\frac{1}{j\omega_{N_s} - \lambda_{2N}} \right]_{N_s \times 2N} \left\{ A_{pqr} \right\}_{2N \times 1} \quad (4.14)$$

where,

$$\left[\frac{1}{j\omega_{N_s} - \lambda_{2N}} \right]_{N_s \times 2N} = \begin{bmatrix} \frac{1}{(j\omega_1 - \lambda_1)} & \frac{1}{(j\omega_1 - \lambda_2)} & \cdots & \frac{1}{(j\omega_1 - \lambda_{2N})} \\ \frac{1}{(j\omega_2 - \lambda_1)} & \frac{1}{(j\omega_2 - \lambda_2)} & \cdots & \frac{1}{(j\omega_2 - \lambda_{2N})} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{1}{(j\omega_{N_s} - \lambda_1)} & \frac{1}{(j\omega_{N_s} - \lambda_2)} & \cdots & \frac{1}{(j\omega_{N_s} - \lambda_{2N})} \end{bmatrix}$$

$$\left\{ H_{pq}(\omega) \right\}_{N_s \times 1} = \begin{Bmatrix} H_{pq}(\omega_1) \\ H_{pq}(\omega_2) \\ \vdots \\ H_{pq}(\omega_{N_s}) \end{Bmatrix}; \quad \left\{ A_{pqr} \right\}_{2N \times 1} = \begin{Bmatrix} A_{pq1} \\ A_{pq2} \\ \vdots \\ A_{pq2N} \end{Bmatrix}$$

By pre-multiplying both sides of the Equation (4.14) with pseudo-inverse of $\left[\frac{1}{j\omega_{N_s} - \lambda_{2N}} \right]$ and solving FRF by FRF, residues can be obtained.

4.2.3 Normalizing the Residues Using Partial Fraction Method

Normalized residues were obtained by manipulating the residue synthesis procedure explained in the Subsection (4.2.2). The imaginary part of $\left[\frac{1}{j\omega_{N_s} - \lambda_{2N}} \right]$ and the real part of the FRF matrix, $\left\{ H_{pq}(\omega) \right\}$, were used for the synthesis of residues as shown below,

$$Re \left(\left\{ H_{pq}(\omega) \right\}_{N_s \times 1} \right) = Im \left(\left[\frac{1}{j\omega_{N_s} - \lambda_{2N}} \right]_{N_s \times 2N} \right) \left\{ A_{pqr} \right\}_{2N \times 1} \quad (4.15)$$

Equation (4.15) was pre-multiplied with the pseudo-inverse of $Im \left(\left[\frac{1}{j\omega_{N_s} - \lambda_{2N}} \right]_{N_s \times 2N} \right)$ and then solved FRF by FRF to obtain the residues.

4.2.4 Application on a Rectangular Plate

A rectangular steel plate of dimensions of 34.00" $\times$ 22.50" $\times$ 0.25" with 21 references was excited over 160 points with an impact hammer. The plate was excited over a range 0 to 250 Hz with a Δf of 0.1563 Hz. The FRF for 160 inputs, 21 references and 1601 frequency lines will therefore be 21 $\times$ 160 $\times$ 1601 in size. For the application of this technique on the data, however, only one of the references was selected. Therefore, the

FRF matrix will have the dimensions $1 \times 160 \times 1601$.

4.2.5 Pole Setup and Modal Parameter Estimation

Similar to first technique, a smaller bandwidth was selected, mainly, to avoid contamination of modal vectors due to repeated roots (refer [5] for details). The selected bandwidth can be seen in Figure (4.11).

Figure 4.11: Selected Bandwidth for Rectangular Plate

The parameters of the selected bandwidth are shown in Table (4.6). From the Complex

Minimum Frequency	59.8438 Hz
Maximum Frequency	237.1875 Hz
Minimum Time	0.0480 s
Maximum Time	3.1980 s
Number of Frequency Lines	1136
Number of Time Points	1119

Table 4.6: Parameters for the selected bandwidth

Mode Indicator Function (CMIF) shown in Figure (4.12), it can be seen that, for the selected frequency range, there are six poles of the system.

Figure 4.12: Complex Mode Indicator Function

The data was processed using RFP algorithm with the highest order term normalized to identity. The residues were then processed for the selected frequency range.

4.2.6 Normalizing the Residues Using Partial Fraction Method

On application of the normalization technique explained in Subsection (4.2.3) on the rectangular plate, purely imaginary residues were obtained. The complexity of the residues before and after normalization can be seen in Figure (4.13) and the mode shapes from Figure (4.14) to (4.19). The residues in both the cases were also compared using the modal assurance criterion (MAC), shown in Table (4.7).

(a) Default

(b) After Truncation

Figure 4.13: Modal Vector Complexity Residue Comparison for Rectangular Plate

Frequency (Hz)	MAC
95	0.9833
104	0.9999
119	0.9918
137	0.9973
176	0.9975
203	0.9945

Table 4.7: Modal Assurance Criterion (MAC)

From the complexity plots, it can be concluded that this technique normalizes the complex mode shapes and the normalized mode shapes compare well with the original with a very high MAC value as can be seen in Table (4.13), for this case.

(a) Default

(b) After Truncation

Figure 4.14: Mode Shape at 95 Hz

Figure 4.15: Mode Shape at 104 Hz

Figure 4.16: Mode Shape at 119 Hz

Figure 4.17: Mode Shape at 137 Hz

Figure 4.18: Mode Shape at 176 Hz

Figure 4.19: Mode Shape at 203 Hz

4.3 Comparison of the Two Techniques

To check how well the mode shapes obtained from truncation technique compare with the mode shapes from the residue technique, the former technique was applied on the rectangular plate data used for the residue technique of real-normalization. The same bandwidth was used and Polyreference Frequency Domain - 2 (PFD - 2) was used to obtain the α matrices with coefficient condensation switched off.

The truncation technique was then applied and the obtained real-normalized mode shapes were compared with the real-normalized mode shapes from the residue technique at the afore mentioned frequencies using the modal assurance criterion. The resulting MAC

values are shown in Table (4.8).

Frequency (Hz)	MAC
95	0.9938
104	0.9978
119	0.9904
137	0.9943
176	0.8849
203	0.9572

Table 4.8: Mode Shape Comparison MAC

It can be seen from Table (4.8) that the mode shapes obtained from both the techniques are very similar to each other. Therefore, regardless of damping distribution, the truncation technique of real-normalization should work as an acceptable truncation technique in case of lightly damped system.

Chapter 5

Explanation for Deviation in Undamped Natural Frequencies

It was shown in Subsection (4.1.6) that the frequencies obtained after truncating the α -polynomial deviate from the undamped natural frequencies of the system as can be seen in Table (4.2). In this chapter, a general explanation for this deviation will be discussed.

5.1 Background

Figure 5.1: 2DOF system

Consider a 2 DOF grounded system shown in Figure (5.1). Where M_1 and M_2 are the two masses, C_1 , C_2 and C_3 are the three dampers and K_1 , K_2 and K_3 are the three springs in the system. The respective mass, damping and stiffness matrices for this system can

be given as,

$$[M] = \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} \quad (5.1)$$

$$[C] = \begin{bmatrix} C_1 + C_2 & -C_2 \\ -C_2 & C_2 + C_3 \end{bmatrix} \quad (5.2)$$

$$[K] = \begin{bmatrix} K_1 + K_2 & -K_2 \\ -K_2 & K_2 + K_3 \end{bmatrix} \quad (5.3)$$

The homogeneous form of Laplace domain model,

$$\left[\begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} s^2 + \begin{bmatrix} C_1 + C_2 & -C_2 \\ -C_2 & C_2 + C_3 \end{bmatrix} s + \begin{bmatrix} K_1 + K_2 & -K_2 \\ -K_2 & K_2 + K_3 \end{bmatrix} \right] \left\{ X(s) \right\} = \left\{ 0 \right\} \quad (5.4)$$

For non-trivial solutions of $\left\{ X(s) \right\}$, the characteristic equation can be given as,

$$\left| \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} s^2 + \begin{bmatrix} C_1 + C_2 & -C_2 \\ -C_2 & C_2 + C_3 \end{bmatrix} s + \begin{bmatrix} K_1 + K_2 & -K_2 \\ -K_2 & K_2 + K_3 \end{bmatrix} \right| = 0 \quad (5.5)$$

Adding the matrices and solving the determinant, Equation (5.5) expands to,

$$\begin{aligned} (M_1 M_2) s^4 + & \underline{(C_1 M_2 + C_2 M_1 + C_2 M_2 + C_3 M_1) s^3} + \boxed{(C_1 C_2 + C_1 C_3 + C_2 C_3) s^2} \\ + & (K_1 M_2 + K_2 M_1 + K_2 M_2 + K_3 M_1) s^2 + \underline{(C_1 K_2 + C_2 K_1 + C_1 K_3 + C_3 K_1 + C_2 K_3 + C_3 K_2) s} \\ & + K_1 K_2 + K_1 K_3 + K_2 K_3 = 0 \end{aligned} \quad (5.6)$$

In Equation (5.6), the terms in red are all the odd powered damping terms in the system and the terms in green are the damping terms with an even power. As discussed in

Subsection (3.1.1), any transfer function can be represented in a polynomial model. From Equation (3.2), a MDOF system represented by the second order characteristic equation in polynomial form with an order of 2 can be represented as follows,

$$[H_{pq}(s)] = \frac{\{X_p(s)\}}{\{F_q(s)\}} = [[M]s^2 + [C]s + [K]]^{-1} = [\beta_0] [[\alpha_2]s^2 + [\alpha_1]s + [\alpha_0]]^{-1} \quad (5.7)$$

From Equation (5.7), for a second order system, it can be concluded that the M, C and K matrices correspond to α_2 , α_1 and α_0 respectively. Therefore, the expansion in Equation (5.6) can be said to equate with the α -polynomial of the analytical 2 DOF system with an order of 4 which can be compared to the iteration 3 of the RFP algorithm in X-Modal for comparison.

In the Equation (5.6), it can be seen that the influence of terms from the damping matrix affect more than the odd ordered polynomial terms. The damping terms are spread throughout the polynomial from $(n - 1)$ to 1 powered terms, where n is the highest order. Therefore, the variation in the undamped natural frequency of the system as shown in Table (4.2) can be said to be due to the presence of damping in the even powered terms. This hypothesis will be proved in the following sections.

5.2 2 DOF Analytical Example

For the 2 DOF system shown in the Figure (5.1), the following mass and stiffness values were assumed,

$$M_1 = 100 \text{ kg}; M_2 = 50 \text{ kg}$$

$$K_1 = K_2 = K_3 = 1 \times 10^6 \frac{N}{m}$$

The damping was assumed proportional to both mass and stiffness with the mass proportionality constant, α , as 0.1 s^{-1} and the damping proportionality, β , as 0.001 s ,

$$[C] = \alpha[M] + \beta[K]$$

For the assumed proportionality constants, α and β , the damping in the system varies from moderate to heavy damping as can be seen in the second column of Table (5.1). The above assumed values for M, C and K were then substituted in Equation (5.6) to determine the poles of the system, s . On obtaining the poles after solving the complete polynomial, the polynomial was truncated using the technique discussed in Section (4.1) and by manually eliminating all the damping terms in the equation. On solving the polynomial in those three ways, the results shown in Table (5.1) were obtained.

Original (Hz)	ζ (%)	Absolute	Truncated (Hz)	Deviation (%)	No damping (Hz)
$-6.3897 - 112.4218i$	5.6746	112.6033	$0.0000 - 111.6370i$	-0.8581	$0.0000 - 112.6033i$
$-6.3897 + 112.4218i$	5.6746	112.6033	$0.0000 + 111.6370i$	-0.8581	$0.0000 + 112.6033i$
$-23.7102 - 216.2367i$	10.899	217.5328	$0.0000 - 219.4156i$	0.8655	$0.0000 - 217.5328i$
$-23.7102 + 216.2367i$	10.899	217.5328	$0.0000 + 219.4156i$	0.8655	$0.0000 + 217.5328i$

Table 5.1: Comparison of Modal Frequencies

In the first column of Table (5.1) the original modal frequencies of the system, i.e. the frequencies obtained by solving the complete polynomial, are listed. The second column shows the damping ratios at the modes. In the third column are the absolute values of the modal frequencies in the first column, these are the undamped natural frequencies of the system. The fourth column represents the modal frequencies obtained after truncating the polynomial i.e. eliminating the odd powered terms, the terms highlighted by red in Equation (5.6), in the polynomial. In the fifth column is the percent deviation of the modal frequencies in column four from three. And, in the sixth column are the modal frequencies obtained on solving the polynomial without any damping terms, i.e. all the terms highlighted by green and red in Equation (5.6) have been eliminated.

From the Table (5.1), it can be seen that the damping terms distributed across the

polynomial between $(n - 1)$ and 1 powered terms cause the undamped natural frequency to deviate. Upon eliminating all the damping terms in the polynomial, the red and green terms, the undamped natural frequency of the system can be obtained.

5.3 4 DOF Analytical Example

Figure 5.2: 4 DOF system

Figure (5.2) is a 4 DOF analytical system where M_1 , M_2 , M_3 and M_4 are the masses, C_1 , C_2 , C_3 , C_4 and C_5 are the dampers and K_1 , K_2 , K_3 , K_4 and K_5 are the springs. The mass, damping and stiffness matrices for the system can be given as follows,

$$[M] = \begin{bmatrix} M_1 & 0 & 0 & 0 \\ 0 & M_2 & 0 & 0 \\ 0 & 0 & M_3 & 0 \\ 0 & 0 & 0 & M_4 \end{bmatrix} \quad (5.8)$$

$$[C] = \begin{bmatrix} C_1 + C_2 & -C_2 & 0 & 0 \\ -C_2 & C_2 + C_3 & -C_3 & 0 \\ 0 & -C_3 & C_3 + C_4 & -C_4 \\ 0 & 0 & -C_4 & C_4 + C_5 \end{bmatrix} \quad (5.9)$$

$$[K] = \begin{bmatrix} K_1 + K_2 & -K_2 & 0 & 0 \\ -K_2 & K_2 + K_3 & -K_3 & 0 \\ 0 & -K_3 & K_3 + K_4 & -K_4 \\ 0 & 0 & -K_4 & K_4 + K_5 \end{bmatrix} \quad (5.10)$$

Therefore, the poles of the system can be obtained by solving Equation (5.11),

$$\left| \begin{bmatrix} M_1 & 0 & 0 & 0 \\ 0 & M_2 & 0 & 0 \\ 0 & 0 & M_3 & 0 \\ 0 & 0 & 0 & M_4 \end{bmatrix} s^2 + \begin{bmatrix} C_1 + C_2 & -C_2 & 0 & 0 \\ -C_2 & C_2 + C_3 & -C_3 & 0 \\ 0 & -C_3 & C_3 + C_4 & -C_4 \\ 0 & 0 & -C_4 & C_4 + C_5 \end{bmatrix} s + \begin{bmatrix} K_1 + K_2 & -K_2 & 0 & 0 \\ -K_2 & K_2 + K_3 & -K_3 & 0 \\ 0 & -K_3 & K_3 + K_4 & -K_4 \\ 0 & 0 & -K_4 & K_4 + K_5 \end{bmatrix} \right| = 0 \quad (5.11)$$

5.3.1 Case I: Second Order Polynomial with 4×4 Matrix Coefficients

The following mass and stiffness values were assumed for the 4 DOF system,

$$M_1 = 200 \text{ kg}; M_2 = 150 \text{ kg}; M_3 = 100 \text{ kg}; M_4 = 50 \text{ kg}$$

$$K_1 = K_2 = K_3 = K_4 = K_5 = 1 \times 10^6 \frac{N}{m}$$

The damping was again assumed proportional to both mass and stiffness with the proportionality constants, α and β , were assumed to be 0.1 s^{-1} and 0.001 s respectively. For the assumed proportionality constants, α and β , the damping in the system varies from moderate to heavy damping as can be seen in the second column of Table (5.2). The assumed values were substituted in Equation (5.11) and solved. The truncation technique was then applied i.e. the odd powered term, the damping matrix, was eliminated and then again solved for the poles of the system. The modal frequencies obtained are compared in Table (5.2).

In Table (5.2), the first column contains the modal frequencies obtained by solving the complete polynomial in Equation (5.11). The second column shows the percent damping in the system with respect to the critical damping of the system. In the third column are the absolute values of the modal frequencies in the first column. The fourth column

Original (Hz)	ζ (%)	Absolute (Hz)	Truncated (Hz)	Deviation (%)
$-1.4601 + 54.0195i$	2.7020	54.0392	$0.0000 + 54.0392i$	0.0000
$-1.4601 - 54.0195i$	2.7020	54.0392	$0.0000 - 54.0392i$	0.0000
$-5.5386 + 105.1022i$	5.2624	105.2480	$0.0000 + 105.2480i$	0.0000
$-5.5386 - 105.1022i$	5.2624	105.2480	$0.0000 - 105.2480i$	0.0000
$-10.7879 + 146.4903i$	7.3444	146.8870	$0.0000 + 146.8870i$	0.0000
$-10.7879 - 146.4903i$	7.3444	146.8870	$0.0000 - 146.8870i$	0.0000
$-23.8801 + 217.2323i$	10.9274	218.5409	$0.0000 + 218.5409i$	0.0000
$-23.8801 - 217.2323i$	10.9274	218.5409	$0.0000 - 218.5409i$	0.0000

Table 5.2: Modal Frequencies Comparison for Analytical 4 DOF System

contains the modal frequencies obtained by truncating the polynomial in Equation (5.11). The last column shows the percent deviation between the modal frequencies pre and post truncation. It can be concluded from Table (5.2) that the truncation technique of normalization works on second order polynomials.

The corresponding modal vectors are shown in Equation (5.12). Since proportional damping was assumed, the modal vectors are already real.

$$[\psi] = \begin{bmatrix} 0.9417 & 0.9417 & 1.0000 & 1.0000 & -0.5759 & -0.5759 & -0.0408 & -0.0408 \\ 1.0000 & 1.0000 & -0.1616 & -0.1616 & 1.0000 & 1.0000 & 0.2313 & 0.2313 \\ 0.5705 & 0.5705 & -0.5365 & -0.5365 & -0.5363 & -0.5363 & -0.7760 & -0.7760 \\ 0.1539 & 0.1539 & -0.1855 & -0.1855 & -0.2911 & -0.2911 & 1.0000 & 1.0000 \end{bmatrix} \quad (5.12)$$

5.3.2 Case II: Fourth Order Polynomial with 2×2 Matrix Coefficients

To show that the damping terms are not just present in the odd powers of a polynomial, the above assumed 4 DOF analytical model with a second order characteristic equation was sub-spaced into a 4th order polynomial with 2×2 sized matrix coefficients such that it gives the same modal frequencies as the second order polynomial did.

As discussed earlier, in Subsection (3.2.1), a second order polynomial with 4×4 matrix coefficients will give eight modal frequencies. Similarly, a fourth order polynomial with matrix coefficients of 2×2 will give eight modal frequencies as well. So, there is no change in the system as far as modal frequencies are concerned, however, instead of 4 DOF, the system will be assumed to have just 2 DOF and hence the modal vectors in this case will be 2×8 and not 4×8 as in the first case.

To expand the second order polynomial to a fourth order polynomial, the usual eigenvalue decomposition process was performed in the reverse order. Since the modal frequencies and modal vectors that were supposed to be obtained after the eigenvalue decomposition are known, the companion matrix can be deduced using Equation (3.15) shown below,

$$[C][\phi] = \lambda[I][\phi]$$

Modal vectors corresponding to the first and third DOF of the four output DOF calculated in the first case, Subsection (5.3.1), were selected as the new modal vector, $\bar{\psi}$. Using these selected modal vectors, the eigenvector was created using Equation (3.19). The modal frequencies, λ , however, will remain the same. To obtain the new companion matrix, $\bar{C}$, Equation (3.15) was post-multiplied with $\{\bar{\phi}\}^{-1}$,

$$\begin{aligned} [C][\phi][\bar{\phi}]^{-1} &= [\phi][\lambda][\bar{\phi}]^{-1} \\ [C] &= [\phi][\lambda][\bar{\phi}]^{-1} \end{aligned} \tag{5.13}$$

Using this technique, the following companion matrix was obtained,

$$\begin{bmatrix} -23.3333 & 0 & -23433.3333 & 66.6667 & -200000 & 133333.3333 & -100000000 & 66666666.6667 \\ 26.6667 & -60 & 26966.6667 & -60533.3333 & 600000 & -1066666.6667 & 300000000 & -533333333.3333 \\ 1.0000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9999 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.9999 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.9999 & 0 & 0 \end{bmatrix} \quad (5.14)$$

The polynomial represented by this companion matrix can be given as,

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} s^4 + \begin{bmatrix} 23.3333 & 0 \\ -26.6667 & 60 \end{bmatrix} s^3 + \begin{bmatrix} -23433.3333 & 66.6667 \\ -26966.6667 & 60533.3333 \end{bmatrix} s^2 + \begin{bmatrix} 200000 & -133333.3333 \\ -600000 & 1066666.6667 \end{bmatrix} s + \begin{bmatrix} 100000000 & -66666666.6667 \\ -300000000 & 533333333.3333 \end{bmatrix} = 0 \quad (5.15)$$

This polynomial on solving will give the modal frequencies obtained in Case I. To check the effect of no damping in the modal frequencies on the companion matrix, the absolute values of the modal frequencies in Table (5.2) were used in Equation (5.13) along with

the same eigenvector, $\bar{\phi}$. On solving, the following companion matrix was obtained,

$$\begin{bmatrix} 0 & 0 & -23333.3206 & -0.0001 & 0 & 0 & -99999882.3648 & 66666608.4228 \\ 0 & 0 & 26666.65557 & -59999.97178 & 0 & 0 & 299999702.4389 & -533332787 \\ 0.9999 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9999 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.9999 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.9999 & 0 & 0 \end{bmatrix} \quad (5.16)$$

The polynomial represented by this companion matrix can be given as,

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} s^4 + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} s^3 + \begin{bmatrix} 23333.3206 & 0.0001 \\ -26666.65557 & 59999.97178 \end{bmatrix} s^2 + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ + \begin{bmatrix} -99999882.3648 & 66666608.4228 \\ 299999702.4389 & -533332787 \end{bmatrix} = 0 \quad (5.17)$$

On comparing the matrices in Equations (5.16) and (5.17), it can be seen that eliminating the odd powered terms in the polynomial might remove the damping (real part) from the modal frequencies, but it will not yield the undamped natural frequencies of the system in case of a fourth order polynomial.

From the matrix coefficients in Equation (5.17), it can be seen that the coefficients of the odd powered terms have been reduced to a null matrix but the other matrices have also been changed. The change in the values of the elements in Equation (5.17) suggests that the damping in any system is not restricted to the odd powered terms and is spread across all the terms in the polynomial.

5.3.3 2 DOF and 4 DOF Example Conclusion

From the cases on 2 DOF and 4 DOF analytical model, it can be concluded that the truncation technique works successfully for second order polynomials yielding the exact undamped natural frequencies of the system. Sub-spacing a polynomial into higher degrees causes the damping to distribute across the polynomial thereby causing the undamped natural frequencies to deviate slightly. The maximum deviation observed in all of the above cases was under 1% in spite of the system being heavily damped at certain modes.

5.4 15 DOF Analytical Example

From the 2 DOF and 4 DOF examples, it was shown that the truncation technique should work as long as it is applied on second order polynomials. Polyreference Frequency Domain - 2 (PFD - 2) algorithm is such a low order frequency domain algorithm. The order of polynomials obtained using this algorithm is 2. Therefore, the hypothesis that the truncation technique should in fact work on second order polynomials will be tested on a 15 DOF analytical system by processing it using the PFD - 2 algorithm.

For the theoretical model, damping was assumed proportional to the stiffness. The FRF matrix was created for the assumed analytical model with 1000 frequency points. The dimensions of the FRF matrix was therefore, $15 \times 15 \times 1000$. The CMIF plot for the analytical model is shown in Figure (5.3).

The FRF matrix was then processed using the PFD - 2 algorithm in three different ways. The 15 DOF model was first solved analytically, similar to the way the 4 DOF system was solved in Subsection (5.3.1). The modal frequencies obtained after processing with PFD - 2 will be compared with the analytical solution.

Figure 5.3: 15 DOF CMIF

5.4.1 Case I: Analytical Solving of the 15 DOF Model

The 15 DOF system was solved analytically, i.e. by making a companion matrix of the mass, damping and stiffness matrices and doing an eigenvalue decomposition of the companion matrix. The truncation technique was then applied on the analytical 15 DOF model i.e. the damping matrix, $[C]$, was eliminated and the truncated polynomial was then solved for the poles. The modal frequencies obtained after truncation are compared with the original poles of the system in Table (5.3).

In the first column are the poles obtained on solving the complete second order polynomial consisting of the M , C and K matrices. The second column shows the percent damping in the system with respect to the critical damping in the system. In the third column are the absolute values of the frequencies in the first column. The last column corresponds to the modal frequencies obtained on application of the truncation technique. As shown with the help of 4 DOF analytical system in Subsection (5.3.1), the undamped natural frequency obtained after truncation match with the system's.

Original (Hz)	ζ (%)	Absolute (Hz)	Truncated (Hz)
-0.1605 + 15.9852i	1.0044	15.9860	0.0000 + 15.9860i
-0.5986 + 30.8588i	1.9393	30.8646	0.0000 + 30.8646i
-1.1950 + 43.5941i	2.7401	43.6104	0.0000 + 43.6104
-1.3567 + 46.4474i	2.9196	46.4672	0.0000 + 46.4672i
-1.7889 + 53.3281i	3.3526	53.3581	0.0000 + 53.3581i
-1.7932 + 53.3923i	3.3566	53.4224	0.0000 + 53.4224i
-2.2217 + 59.4229i	3.7363	59.4644	0.0000 + 59.4644i
-2.3883 + 61.6064i	3.8738	61.6526	0.0000 + 61.6526i
-2.9818 + 68.8240i	4.3284	68.8890	0.0000 + 68.8890i
-3.4152 + 73.6472i	4.6324	73.7263	0.0000 + 73.7263i
-12.9451 + 142.9519i	9.0187	143.5368	0.0000 + 143.5368i
-12.9703 + 143.0896i	9.0274	143.6762	0.0000 + 143.6762i
-13.0077 + 143.2943i	9.0405	143.8835	0.0000 + 143.8835i
-13.0492 + 143.5210i	9.0549	144.1130	0.0000 + 144.1130i
-13.0826 + 143.7027i	9.0664	144.2970	0.0000 + 144.2970i

Table 5.3: Case I Modal Frequencies

5.4.2 Case II: 15×15 FRF Matrix, Full Frequency Range

The $15 \times 15 \times 1000$ FRF matrix was imported into X-Modal and the bandwidth shown in Figure (5.4) was processed using the PFD - 2 algorithm. The α matrices obtained on processing were then used to apply the truncation technique.

The α matrices obtained from X-Modal were used to calculate the system poles and modal vectors. The odd powered term, $[\alpha_1]$, was then eliminated and the modal frequencies and vectors were calculated again thereby applying the truncation technique. Modal frequencies obtained pre and post truncation are shown in Table (5.4).

 Figure 5.4: Selected Bandwidth for 15×15 FRF

Original (Analytical)	Original (X-Modal)	Absolute	Truncated	% Deviation
$-0.1605 + 15.9852i$	$-0.1606 + 15.9852i$	15.9860	$0.0000 + 15.9860i$	0.0000
$-0.5986 + 30.8586i$	$-0.5986 + 30.8588i$	30.8646	$0.0000 + 30.8646i$	0.0000
$-1.19450 + 43.5941i$	$-1.195 + 43.5940i$	43.6104	$0.0000 + 43.6104i$	0.0000
$-1.3567 + 46.4474i$	$-1.3567 + 46.4474i$	46.4672	$0.0000 + 46.4672i$	0.0000
$-1.7888 + 53.3281i$	$-1.7889 + 53.3281i$	53.3581	$0.0000 + 53.3581i$	0.0000
$-1.7931 + 53.3923i$	$-1.7932 + 53.3923i$	53.4224	$0.0000 + 53.4224i$	0.0000
$-2.2217 + 59.4229i$	$-2.2214 + 59.4231i$	59.4646	$0.0000 + 59.4646i$	0.0000
$-2.3885 + 61.6064i$	$-2.3884 + 61.6063i$	61.6526	$0.0000 + 61.6526i$	0.0000
$-2.98180 + 68.8245i$	$-2.9818 + 68.8246i$	68.8891	$0.0000 + 68.8891i$	0.0000
$-3.41527 + 73.6472i$	$-3.4149 + 73.6473i$	73.7264	$0.0000 + 73.7264i$	0.0000
$-12.9451 + 142.9519i$	$-12.9435 + 142.9579i$	143.5427	$0.0000 + 143.5427i$	0.0000
$-12.9702 + 143.0896i$	$-12.9714 + 143.0817i$	143.6685	$0.0000 + 143.6685i$	0.0000
$-13.0077 + 143.2943i$	$-13.0096 + 143.2829i$	143.8723	$0.0000 + 143.8723i$	0.0000
$-13.0493 + 143.5210i$	$-13.0493 + 143.5240i$	144.1160	$0.0000 + 144.1160i$	0.0000
$-13.0826 + 143.7027i$	$-13.0798 + 143.7053i$	144.2993	$0.0000 + 144.2993i$	0.0000

 Table 5.4: Case I Modal Frequencies Comparison, 15×15 FRF

In the Table (5.4), the first column corresponds to the modal frequencies obtained by solving the 15 DOF system analytically in Case (5.4.1). The second column corresponds to the poles obtained from X-Modal for the $15 \times 15 \times 1000$ FRF matrix. The third column consists of the absolute values of the modal frequencies in the second column. The

fourth column consists of the modal frequencies obtained after applying the truncation technique on data obtained from X-Modal. And in the last column is the percent deviation between the undamped natural frequencies in the third and fourth column. It can be concluded from fifth column that the truncation technique works efficiently on second order polynomials when all of the degrees of freedom and all the modes are selected.

The modal vectors before and after truncation are compared using modal assurance criterion (MAC) in Table (5.5).

Frequency	MAC
15.9860	1.0000
30.8646	1.0000
43.6104	1.0000
46.4672	1.0000
53.3581	1.0000
53.4224	1.0000
59.4646	1.0000
61.6526	1.0000
68.8891	1.0000
73.7264	1.0000
143.5427	0.9997
143.6685	0.9998
143.8723	0.9998
144.1160	0.9998
144.2993	1.0000

Table 5.5: Case II MAC

5.4.3 Case III: FRF Matrix 4×15 Complete Frequency Range

In this case, FRF matrix of size $4 \times 15 \times 1000$ was used instead of $15 \times 15 \times 1000$ used in Case (5.4.2). The bandwidth selected for this case is exactly the same as shown in Figure (5.4).

The exact procedure in Case (5.4.2) was then followed. The obtained modal frequencies pre and post truncation are compared in Table (5.6).

Original (Hz)	Absolute (Hz)	Truncated (Hz)	Deviation (%)
$-0.1606 + 15.9852i$	15.9860	$0.0000 + 15.9860$	0.0000
$-0.5985 + 30.8588i$	30.8646	$0.0000 + 30.8646$	0.0000
$-1.1949 + 43.594i$	43.6104	$0.0000 + 43.6104$	0.0000
$-1.3567 + 46.4472i$	46.4670	$0.0000 + 46.4670$	0.0000
$-1.7889 + 53.3283i$	53.3583	$0.0000 + 53.3583$	0.0000
$-1.793 + 53.3921i$	53.4222	$0.0000 + 53.4222$	0.0000
$-2.222 + 59.4230i$	59.4645	$0.0000 + 59.4645$	0.0000
$-2.3893 + 61.6055i$	61.6519	$0.0000 + 61.6519$	0.0000
$-2.9823 + 68.8247i$	68.8892	$0.0000 + 68.8892$	0.0000
$-3.4152 + 73.6465i$	73.7257	$0.0000 + 73.7257$	0.0000
$-13.0336 + 142.8375i$	143.4309	$0.0000 + 143.4208$	0.0071
$-13.1572 + 143.0060i$	143.6100	$0.0000 + 143.6171$	-0.0049
$-13.0888 + 143.4708i$	144.0666	$0.0000 + 144.0752$	-0.0060
$-13.1774 + 143.5807i$	144.1842	$0.0000 + 144.1787$	0.0038

Table 5.6: Case III Modal Frequencies Comparison

Similar to the previous two sections, the first column consists of the original frequencies of the system, i.e. the modal frequencies obtained by solving the complete polynomial. The second column corresponds to the absolute values of the frequencies in the first column. In the third column are the modal frequencies obtained on applying the truncation technique and in the last column is the percent deviation of undamped natural frequencies in the third and second column.

The MAC for the modal vectors before and after truncation are shown in Table (5.7).

Frequency (Hz)	MAC
15.9860	1.0000
30.8646	1.0000
43.6104	1.0000
46.4670	1.0000
53.3583	0.9999
53.4222	0.9998
59.4645	1.0000
61.6519	1.0000
68.8892	1.0000
73.7257	1.0000
143.4309	0.9357
143.6100	0.9378
144.0666	0.8488
144.1842	0.9760

Table 5.7: Case III MAC

5.4.4 Case IV: FRF matrix of 4×15 with the Highest Three Modes Left Out

In this case, the same FRF matrix used in Case (5.4.3) was used, however, the entire frequency range was not selected for processing with the PFD algorithm. The selected frequency range is shown in Figure (5.5).

Original	Abolute	Truncated	% Deviation
-0.1606 + 15.9852	15.9860	0.0000 + 15.986	0.0000
-0.5988 + 30.8589	30.8647	0.0000 + 30.8647	0.0000
-1.1935 + 43.5971	43.6135	0.0000 + 43.6135	0.0000
-1.3558 + 46.4472	46.4669	0.0000 + 46.4670	0.0000
-1.8594 + 53.2026	53.2351	0.0000 + 53.2348	0.0005
-1.7906 + 53.39	53.4200	0.0000 + 53.4192	0.0015
-2.3848 + 59.3207	59.3686	0.0000 + 59.369	-0.0006
-2.4746 + 61.6016	61.6513	0.0000 + 61.6521	-0.0013

Table 5.8: Case IV Modal Frequencies Comparison

MAC for the modal vectors before and after truncation are shown in the Table (5.9).

Figure 5.5: Poles for 4×15 FRF Complete Frequency Range

Frequencies	MAC
15.9860	1.0000
30.8647	1.0000
43.6135	1.0000
46.4669	1.0000
53.2351	0.9998
53.4200	0.9114
59.3686	0.9992
61.6513	0.9950

Table 5.9: Case IV MAC

5.4.5 Conclusion for the 15 DOF Cases

From the four cases, it can be seen that the undamped natural frequencies before and after truncation match very well and the modal vectors also MAC well. It can therefore be concluded that the truncation technique works effectively on low order algorithms likes PFD - 2. The small deviations in Case III and Case IV are probably resulting due to the reduction in the FRF matrix and/or due to the participation of out of band modes.

In Subsection (4.1.8), the truncation technique was applied on the data obtained from a

circular plate using PFD - 2 algorithm. In that case, the undamped natural frequencies did not match exactly with that of the system (maximum deviation of 1.0984 %) unlike in the cases shown in this section, due to the presence of noise in the obtained experimental data and/or the effect of the modes outside the selected band.

5.5 Coefficient Condensation

Another possible explanation for the distribution of damping across the polynomial is Coefficient Condensation. It is used to reduce the number of physical output DOFs, N_o to much more reasonable DOFs, N_e , to limit the time consumption for calculations. Using the techniques such as Eigenvalue Decomposition (ED) or Singular Value Decomposition (SVD), the physical coordinates are reduced to virtual coordinates. During this reduction, the information of the physical system is sub-spaced into a smaller virtual subspace which is then used to determine the modal parameters of the system. Due to the transformation, the mass, damping and the stiffness matrices are no longer restricted to the second, first and zero order terms respectively and they get distributed throughout the polynomial similar to distribution of the terms in the analytical examples shown in this chapter.

Chapter 6

Conclusions and Future Work

6.1 Conclusion

Two new normalization techniques for complex mode shapes were proposed and tested on experimental data. The first technique, normalization by truncation of the α -polynomial, successfully normalizes the complex modal vectors and yields the exact undamped natural frequencies of the system when used on second order modal parameter estimation methods. This method of normalization should therefore work well on second order time and frequency domain algorithms such as Eigensystem Realization Algorithm (ERA) and Polyreference Frequency Domain - 2 (PFD - 2) respectively. Reduction in FRF matrix, out of band mode effects, noise in the data and the use of coefficient condensation might, however, cause a small deviation in the undamped natural frequencies.

When the truncation technique is used on polynomials with an order greater than two, the undamped natural frequencies obtained after truncation, in general, deviates from that of the system's. This deviation is due to the fact that the damping in the system is not restricted to the odd powered terms in the α -polynomial but are spread throughout the polynomial as shown in Chapter (5). There are two possible reasons for this,

- When high order algorithm such as RFP is used: In case of RFP, the second order model is sub-spaced into higher order models which causes the damping terms to distribute across the polynomial as was shown in Subsection (5.3.2).
- When coefficient condensation can be used for second or first order algorithms such as PFD and ERA: For the lower order algorithms like PFD, coefficient condensation is used to minimize the computation time and memory footprint. If this is done, the damping distributes throughout the polynomial.

The deviation in the undamped natural frequency caused due to the distribution of damping in case of polynomials with order higher than two on application of the truncation technique is very low as can be seen in Table (4.2). Even in cases with moderate to high damping in the analytical models, the deviation in undamped natural frequencies is a maximum of 1%. The truncation technique might, therefore, be effective even in case of high order polynomials.

The second technique proposed was based on the partial fraction method of residue estimation. This technique was tested on data obtained from a rectangular plate. The technique successfully normalized the complex mode shapes as expected. The MAC values between the normalized and original complex modal vectors was very high. The lowest MAC value being 0.9833.

The mode shapes obtained from both the normalization techniques also compare well as can be seen in Table (4.8).

6.2 Future Work

The undamped natural frequencies obtained from the first normalization technique proposed were contaminated by the presence of damping even after eliminating the odd powered terms in the polynomial, thereby leading to the conclusion that the damping

is present throughout all the terms in the polynomial irrespective of its power in cases where the polynomial is of order greater than two. Therefore, to address this issue, a simple pre-processing technique to effectively eliminate all the damping terms from the polynomial is required.

The truncation technique also cannot be used on odd powered polynomials as shown in Section (4.1.7). A technique that can be used on both even and odd powered polynomials is needed.

In cases of low order algorithms like PFD - 2, the reduction in the FRF matrix size, modes from outside the selected band, noise in the data and coefficient condensation might have an impact on the effectiveness of the truncation technique. The extent of their impact needs to be investigated.

The residue synthesis method of normalization used in this thesis takes into account just the real part of the FRF matrix and the imaginary part of the $(j\omega_r - \lambda_r)^{-1}$ matrix. Therefore, a technique needs to be developed based on the partial fraction method of residue synthesis which takes into account the complete matrices thereby reducing the loss in data that is caused due to the proposed technique. Also, the residues were synthesized for a Single Input Multiple Output (SIMO) case. The technique should also be tested on a Multiple Input Multiple Output (MIMO) system.

References

- [1] **Avitable, P.**, *Modal Space Back to Basics*, Experimental Techniques, pp. 17–18, May/June 2002.
- [2] **Kohli, K.**, Structural Dynamics Model Calibration and Validation of a Rectangular Steel Plate Structure, Master’s thesis, University of Cincinnati, Ohio, USA, 2014.
- [3] **Niedbal, N.**, *Analytical Determination of Real Normal Modes from Measured Complex Responses*, *Proceedings of the 25th Structures, Structural Dynamics and Materials Conference, Palm Springs, CA*, pp. 292–5, 1984.
- [4] **Deblauwe, F.** and **Allemang, R. J.**, *A Possible Origin of Complex Modal Vectors*, *Proceedings of International Seminar on Modal Analysis, Katholieke Universiteit Leuven, Belgium*, 1986.
- [5] **Hiremaglur, R.**, Real-normalization of Experimental Complex Modal Vectors with Modal Vector Contamination, Master’s thesis, University of Cincinnati, Ohio, USA, 2014.
- [6] **Imregun, M.** and **Ewins, D. J.**, *Complex Modes - Origins and Limits*, *International Modal Analysis Conference*, 1995.
- [7] **Frazer, R. A.**, **Duncan, W. J.** and **Collar, A. R.**, *Elementary Matrices And Some Applications to Dynamics and Differential Equations*, Cambridge University Press, 1947, pp. 289.

- [8] **Ibrahim, S. R.**, *Computation of Normal Modes from Identified Complex Modes*, AIAA Journal, Vol. 21, No. 3, pp. 446–451, 1983.
- [9] **Sinha, S.**, Techniques for Real Normalization of Complex Modal Parameters for Updating and Correlation with FEM Models, Master’s thesis, University of Cincinnati, Ohio, USA, 2005.
- [10] **Wei, M. L., Zhang, Q., Allemang, R. J. and Brown, D. L.**, *A Time Domain Subspace Iteration Method for Normalization of Measured Complex Modes*, *Proceedings of International Seminar on Modal Analysis*, Katholieke Universiteit Leuven, Belgium, 1987.
- [11] **Allemang, R. J. and Phillips, A. W.**, *The Unified Matrix Polynomial Approach to Understanding Modal Parameter Estimation: An Update*, *Proceedings of International Seminar on Modal Analysis*, Katholieke Universiteit Leuven, Belgium, 2004.
- [12] **Allemang, R. J.**, *Vibrations: Analytical and Experimental Modal Analysis*, UC-SDRL-CN-20-263-662/663/664.
- [13] **Allemang, R. J.**, *The Modal Assurance Criterion—Twenty Years of Use and Abuse*, Sound and Vibration, Vol. 37, No. 8, pp. 14–23, 2003.

Appendix A

MATLAB Scripts

A.1 Normalization by Truncation of the α Polynomial

A.1.1 Application on Data Obtained Using RFP Algorithm

```
clc;
clear all;
load('C:\Users\Aditya\Documents\X-Modal\Exported Data\Thesis\Data_thesis
_cplate.mat')
load('C:\Users\Aditya\Documents\X-Modal\Exported Data\First_three
_peaks.mat')

%% Extracting the alpha matrices from the POLE_RESULTS:
clc;
A0=POLE_RESULTS.SubCase(1,3).A(1,1).coeff;
A1=POLE_RESULTS.SubCase(1,3).A(1,2).coeff;
A2=POLE_RESULTS.SubCase(1,3).A(1,3).coeff;
A3=POLE_RESULTS.SubCase(1,3).A(1,4).coeff;
A4=POLE_RESULTS.SubCase(1,3).A(1,5).coeff;
Z=zeros(7);

[X,e1]=polyeig(A0,A1,A2,A3,A4);
e=sort(e1*825/2);

% Without Truncation
I=eye(7);
Z=zeros(7);
C1=[-A3 -A2 -A1 -A0;
    I Z Z Z;
    Z I Z Z;
    Z Z I Z];
```

```

[V,D]=eig(C1);
lambda1=825*pi*(diag(D)/(2*pi));
psi1=(V(22:28,:));
lambda1_abs=sort(abs(lambda1));

% Truncated
I=eye(7);
Z=zeros(7);
C2=[-Z -A2 -Z -A0;
     I Z Z Z;
     Z I Z Z;
     Z Z I Z];

[V2,D2]=eig(C2);
D2=sort(diag(D2));
lambda2=825*pi*sort((D2)/(2*pi));
psi2=(V2(22:28,:));
lambda2_abs=abs(lambda2);

%% Extracting the actual Poles and Vectors:
for i=1:28
    pole(i)=POLE_RESULTS.SubCase(1,3).EigParms(1,i).Pole;
    vec(:,i)=POLE_RESULTS.SubCase(1,3).EigParms(:,i).Vector;
end
poler=((pole.)/(2*pi));
vector=vec;

%% Extracting the needed modal vectors from the calculated ones:
P1=[27 28 25 26 13 14 3 4 1 2];
P1_trunc=[19 20 17 18 13 14 3 4 1 2];
P2=[12 11 14 13 16 15 26 25 28 27];

for i=1:10
    lambda_1_final(i,1)=lambda1(P1(i))
    lambda_1_final_abs=abs(lambda_1_final);
    lambda_2_final(i,1)=lambda2(P2(i))
    psi_final(:,i)=psi1(:,P1(i));
    psi_truncated(:,i)=psi2(:,P1_trunc(i));
    vector_final(:,i)=vector(:,P2(i));
end

%% Error
error=((lambda_1_final_abs-abs(lambda_2_final))./(lambda_1_final_abs))*100

%% Extracting the needed modal vectors from the Results matrix
for j=1:10
    for i=1:7
        psi_XM(i,j)=Results(1,j).Point(1,i).Y;
    end
end

%% Changing the matrix extracted from X-Modal
for j=1:10
    for i=1:7

```

```

        Results(1,j).Point(1,i).Y=psi_truncated(i,j);
    end
end

%% Scaling modal vectors such that the highest value in each column is 1

for i=1:10
    psi_final1(:,i)=(psi_final(:,i)./max(psi_final(:,i)));
    psi_truncated1(:,i)=(psi_truncated(:,i)./max(psi_truncated(:,i)));
    psi_XM1(:,i)=(psi_XM(:,i)./max(psi_XM(:,i)));
    vector_final1(:,i)=(vector_final(:,i)./max(vector_final(:,i)));
end

for i=1:28
    %X_scaled(:,i)=(X(:,i)./max(X(:,i)));
    psi1_scaled(:,i)=(psi1(:,i)./max(psi1(:,i)));
    psi2_scaled(:,i)=(psi2(:,i)./max(psi2(:,i)));
    vector_scaled(:,i)=(vector(:,i)./max(vector(:,i)));
end

%% MAC

% Extracted results original and MATLAB original:
for i=1:10
    MAC1(:,i)=(psi_final(:,i)'*psi_XM(:,i)*psi_XM(:,i)'*psi_final(:,i))
    / (psi_final(:,i)'*psi_final(:,i)*psi_XM(:,i)'*psi_XM(:,i));
end
MAC_act=real(MAC1).';

% Extracted results original and truncated vectors:
for i=1:10
    MAC2(:,i)=(psi_truncated(:,i)'*psi_XM(:,i)*psi_XM(:,i)'*psi_truncated(:,i))
    / (psi_truncated(:,i)'*psi_truncated(:,i)*psi_XM(:,i)'*psi_XM(:,i));
end
MAC_trunc=real(MAC2).';

% MATLAB original and truncated original:
for i=1:10
    MAC3(:,i)=(psi_final(:,i)'*psi_truncated(:,i)*psi_truncated(:,i)'*psi_
    final(:,i))(psi_final(:,i)'*psi_final(:,i)*psi_truncated(:,i)'*psi_
    _truncated(:,i));
end
MAC_mat=real(MAC3).';

```

A.1.2 Application on Data Obtained Using PFD - 2 Algorithm

```

clc;
clear all;
load('C:\Users\Aditya\Documents\X-Modal\Exported Data\PFD\PFD_nocoeffcond-newXM.3.mat')

%% Extracting the alpha matrices from the POLE_RESULTS:
clc;

```

```

N=19;
NM=1
A0=POLE_RESULTS.SubCase(1,NM).A(1,1).coeff;
A1=POLE_RESULTS.SubCase(1,NM).A(1,2).coeff;
A2=POLE_RESULTS.SubCase(1,NM).A(1,3).coeff;
U_xm=CondenseVector.Right;
maxfreq=POLE_RESULTS.Sieve.MaxFreq

Z=zeros(36);

% Without Truncation
I=eye(36);
Z=zeros(36);
C1=[-(A1) -(A0);
     I Z];

[V,D]=eig(C1);
D1=(diag(D));
lambda1=maxfreq*pi*(diag(D)/(2*pi));
lambda1_sort=sort(lambda1);
psi1=(V(37:72,:));
lambda1_abs=abs(lambda1_sort);

%% Truncated
I=eye(36);
Z=zeros(36);
C2=[-Z -A0;
     I Z];

[V2,D2]=eig(C2);
%D2=(diag(D2));
lambda2=maxfreq*pi*(diag(D2)/(2*pi));
lambda2_sort=sort(lambda2);
psi2=(V2(37:72,:));
lambda2_abs=abs(lambda2);

%% Extracting the actual Poles and Vectors:
for i=1:34+N*2
    pole(i)=POLE_RESULTS.SubCase(1,NM).EigParms(1,i).Pole;
    vec(:,i)=POLE_RESULTS.SubCase(1,NM).EigParms(:,i).Vector;
end
poler=(pole.)/(2*pi);
poler_abs=abs(poler);
vector=vec;

% Extracting the needed modal vectors:
P1=[69 70 67 68 21 22 5 6 3 4];
P1_trunc=[71 72 69 70 15 16 3 4 1 2];
P2=[11 12 13 14 23 24 33 34 35 36];

for i=1:10
    lambda1_final(i,1)=lambda1(P1(i))
    lambda2_final(i,1)=lambda2(P1_trunc(i))
    psi_final(:,i)=psi1(:,P1(i));

```

```

        psi_truncated(:,i)=psi2(:,P1_trunc(i));
        %vector_final(:,i)=vector_scaled(:,P2(i));
    end

    % Scaling the modal vectors:
    for i=1:10
        psi_final1(:,i)=(psi_final(:,i)./max(psi_final(:,i)));
        psi_truncated1(:,i)=(psi_truncated(:,i)./max(psi_truncated(:,i)));
    end

    %% Error
    error=(abs(lambda1_final)-abs(lambda2_final))./abs(lambda1_final))*100

    %% Changing the matrix extracted from X-Modal

    load('C:\Users\Aditya\Documents\X-Modal\Exported Data\PFD\results_nocoeffcond.mat')
    for j=1:10
        for i=1:36
            Results(1,j).Point(1,i).Y=psi_truncated1(i,j);
        end
    end

    %% MAC:

    %MAC between the extracted and truncated Modal Vectors
    for i=1:10
        nm=i
        MAC5(:,i)=(psi_final(:,nm)'*psi_truncated(:,i)*psi_truncated(:,i)'*psi_final(:,nm)).';
    end
    MAC_mat=real(MAC5).';

```

A.2 Normalization by Manipulation of the Partial Fraction Method of Residue Synthesis

```

clc;
clear all;
load('C:\Users\Aditya\Documents\X-Modal\Exported Data\Thesis\Data_thesis_rplate_frf.mat')
load('C:\Users\Aditya\Documents\X-Modal\Exported Data\Thesis\Data_thesis_rplate.mat')
tic;
H=Measurements.FRF.H;
f=Measurements.FRF.freq;
H1=squeeze(H(1, :, :));
odof=Measurements.FRF.odof(:,1);
idof=Measurements.FRF.idof(:,1);

%% Extracting the actual Poles and Vectors:
for i=1:84
    pole(i)=POLE_RESULTS.SubCase(1,3).EigParms(1,i).Pole;

```



```

        vec(:,i)=POLE_RESULTS.SubCase(1,3).EigParms(:,i).Vector;
    end
    poler=(pole. ');
    vector=vec;
    pole_hz=(pole/(2*pi)). ';

    %P=zeros(28,28);
    %f_range = linspace(23.4375,237.1875,1369);
    fn = 384:4:1519;
    for j=1:160
        for i=1:numel(fn)
            fnew(i)=f(fn(i));
            H_temp=H1(j,:);
            Hnew(i,j)=H_temp((fn(i)));
        end
    end
    wnew=fnew*pi*2;

    for j=1:160
        for i=1:numel(fn)
            wnew2(i+(i-1):i+i,1) = [wnew(i);-wnew(i)];
            H_conj(i+(i-1):i+i,j) = [Hnew(i,j);conj(Hnew(i,j))];
            H_abs(i+(i-1):i+i,j) = [abs(Hnew(i,j));-abs(Hnew(i,j))]*1j;
        end
    end

    for i=1:42
        pole_abs(i+(i-1):i+i,1)=[abs(pole(i+(i-1)))*1j;-abs(pole(i+(i-1)))*1j];
    end

    % Normal and Absolute Pole
    for j=1:84
        for i=1:numel(fn)*2
            P1(i,j)=(1./(1j*wnew2(i)-pole(j)));
            P2(i,j)=(1./(1j*wnew2(i)-pole_abs(j)));
        end
    end

    % Imaginary parts
    P_imag=imag(P1)*1j;
    H_imag=real(H_conj);

    %% Final residues:
    % Default:
    for j=1:160
        A_1(j,:)=pinv(P1)*H_conj(:,j); %Default
        A_2(j,:)=pinv(P2)*H_abs(:,j); %Absolute values of both
        A_3(j,:)=pinv(P_imag)*H_imag(:,j); %Just the imaginary part
    end

    %% Scaling the residue matrix:
    for i=1:84
        A_1_scaled(:,i) = A_1(:,i)./max(A_1(:,i));
        A_2_scaled(:,i) = A_2(:,i)./max(A_2(:,i));
    end

```

```

        A_3_scaled(:,i) = A_3(:,i)./max(A_3(:,i));
end

%% Extracting the needed modal vectors from the Results matrix
load('C:\Users\Aditya\Documents\X-Modal\Exported Data\Thesis\Data_thesis
_rplate_residue.mat')
for j=1:12
    for i=1:160
        A_XM(i,j)=Results(1,j).Point(1,i).Z;
    end
end
A_XM_imag=imag(A_XM)*1j;

%% Re-arranging the messed up residues:
for j=1:84
    for i=1:160
        A_1_temp=A_1(:,j);
        A_1_re(idof(i),j)=A_1_temp(i,:);

        A_2_temp=A_2(:,j);
        A_2_re(idof(i),j)=A_2_temp(i,:);

        A_3_temp=A_3(:,j);
        A_3_re(idof(i),j)=A_3_temp(i,:);
    end
end

%% Extracting the needed modal residues:
P2=[15 16 25 26 33 34 39 40 53 54 67 68];

for i=1:12
    A_1_final(:,i)=A_1(:,P2(i));
    A_1_re_final(:,i)=A_1_re(:,P2(i));
    A_2_final(:,i)=A_2(:,P2(i));
    A_3_final(:,i)=A_3(:,P2(i));
    A_3_re_final(:,i)=A_3_re(:,P2(i));
end
A_1_final_imag=imag(A_1_re_final)*1j;
pole_hz_final=pole_hz(P2)

%% Changing the matrix extracted from X-Modal
%load('C:\Users\Aditya\Documents\X-Modal\Exported Data\First.three
_peaks.mat')
for j=1:12
    for i=1:160
        Results(1,j).Point(1,i).Z=A_3_re_final(i,j);
    end
end

%% MAC
% MAC between original residue and case 2:
for i=1:12
    MAC11(:,i)=(A_1_final(:,i)'*A_3_final(:,i)*A_3_final(:,i)'*A_1_final(:,i))
        /(A_1_final(:,i)'*A_1_final(:,i)*A_3_final(:,i)'*A_3_final(:,i));
end

```

```

end
MAC1=real(MAC11).';

% MAC between calculated and extracted complete residues:
for i=1:12
    MAC12(:,i)=(A_1_re_final(:,i)'*A_XM(:,i)*A_XM(:,i)'*A_1_re_final(:,i))
        /(A_1_re_final(:,i)'*A_1_re_final(:,i)*A_XM(:,i)'*A_XM(:,i));
end
MAC2=real(MAC12).';

toc;

```

A.3 Analytical Explanation for Deviation in Undamped Natural Frequencies

A.3.1 2 DOF System

```

clc;
clear all;

%% Assuming the mass values:
m11=100; m12=0; m21=0; m22=50; %kg
m=[m11 m12;
   m21 m22];

%% Assuming the stiffness values:
k11=2*10^6; k12=-1*10^6; k21=-1*10^6; k22=2*10^6; %N/m
k=[k11 k12;
   k21 k22];

%% Defining damping:
Case=3;
if Case==1
    % CASE I: Defining the damping proportional to stiffness:
    beta=0.001; %sec, s
    c11=beta*k11; c12=beta*k12; c21=beta*k21; c22=beta*k22; %Ns/m

elseif Case==2
    % CASE II: Defining the damping proportional to mass:
    alpha=0.1; %1/sec
    c11=alpha*m11; c12=alpha*m12; c21=alpha*m21; c22=alpha*m22; %Ns/m
else
    % CASE III: Defining the damping proportional to both mass and stiffness:
    alpha=0.1; %1/sec
    beta=0.001; %sec, s
    c11=alpha*m11+beta*k11; c12=alpha*m12+beta*k12; %Ns/m
    c21=alpha*m21+beta*k21; c22=alpha*m22+beta*k22; %Ns/m
end
c=[c11 c12;
   c21 c22];

```

```

%% System equation on solving:
Damp=0;
if Damp==1
    A=m11*m22-m12*m21; %s^4
    B=c11*m22+m11*c22-m12*c21-c12*m21; %s^3
    C=m11*k22+c11*c22+k11*m22-m12*k21-c12*c21-k12*m21; %s^2
    D=c11*k22+k11*c22-c12*k21-k12*c21; %s
    E=k11*k22-k12*k21;
elseif Damp==0
    A=m11*m22-m12*m21; %s^4
    B=0; %s^3
    C=m11*k22+c11*c22+k11*m22-m12*k21-c12*c21-k12*m21; %s^2
    D=0; %s
    E=k11*k22-k12*k21;
else
    A=m11*m22-m12*m21; %s^4
    B=0; %s^3
    C=m11*k22+c11*c22+k11*m22-m12*k21-k12*m21; %s^2
    D=0; %s
    E=k11*k22-k12*k21;
end
u=[A B C D E];
cm=compan(u);
[V,DD]=eig(cm);
lambda=sort(diag(DD))
lambda_abs=abs(lambda)

% Companion matrix of M, C, K matrices:
I=eye(2);
Z=zeros(2);
companion1=[-Z -k/m
             I Z];
[Vc,Dc]=eig(companion1);
D_full=diag(Dc)

```

A.3.2 4 DOF System

```

clc;
clear all;

%% Assuming the mass values:
% Mass, kg
m11=200; m12=0; m13=0; m14=0;
m21=0; m22=150; m23=0; m24=0;
m31=0; m32=0; m33=100; m34=0;
m41=0; m42=0; m43=0; m44=50;

M11=[m11 m12;
      m21 m22];
M12=[m13 m14;
      m23 m24];

```

```

M21=[m31 m32;
      m41 m42];
M22=[m33 m34;
      m43 m44];
M=[M11 M12;
   M21 M22];

mass=[m11 m12 m13 m14;
      m21 m22 m23 m24;
      m31 m32 m33 m34;
      m41 m42 m43 m44];

%% Assuming the stiffness values:
% Stiffness, N/m
k11=2*10^6; k12=-1*10^6; k13=0; k14=0;
k21=-1*10^6; k22=2*10^6; k23=-1*10^6; k24=0;
k31=0; k32=-1*10^6; k33=2*10^6; k34=-1*10^6;
k41=0; k42=0; k43=-1*10^6; k44=2*10^6;
K11=[k11 k12;
      k21 k22];
K12=[k13 k14;
      k23 k24];
K21=[k31 k32;
      k41 k42];
K22=[k33 k34;
      k43 k44];
K=[K11 K12;
   K21 K22];

stiffness=[k11 k12 k13 k14;
           k21 k22 k23 k24;
           k31 k32 k33 k34;
           k41 k42 k43 k44];

%% Defining damping:
Case=1;
if Case==1
    % CASE I: Defining the damping proportional to stiffness:
    beta=0.001; %sec, s
    C11=beta*K11; %Ns/m
    C12=beta*K12; %Ns/m
    C21=beta*K21; %Ns/m
    C22=beta*K22; %Ns/m

    elseif Case==2
        % CASE II: Defining the damping proportional to mass:
        alpha=0.1; %1/sec
        C11=alpha*M11; %Ns/m
        C12=alpha*M12; %Ns/m
        C21=alpha*M21; %Ns/m
        C22=alpha*M22; %Ns/m

else
    % CASE III: Defining the damping proportional to both mass and stiffness:

```

```

alpha=0.1; %1/sec
beta=0.001; %sec, s
C11=alpha*M11+beta*K11; %Ns/m
C12=alpha*M12+beta*K12; %Ns/m
C21=alpha*M21+beta*K21; %Ns/m
C22=alpha*M22+beta*K22; %Ns/m
end

C=[C11 C12;
   C21 C22];

c11=C(1,1); c12=C(1,2); c13=C(1,3); c14=C(1,4);
c21=C(2,1); c22=C(2,2); c23=C(2,3); c24=C(2,4);
c31=C(3,1); c32=C(3,2); c33=C(3,3); c34=C(3,4);
c41=C(4,1); c42=C(4,2); c43=C(4,3); c44=C(4,4);

damping=[c11 c12 c13 c14;
          c21 c22 c23 c24;
          c31 c32 c33 c34;
          c41 c42 c43 c44];

%% Partition Matrix method - System equation on solving:
Damp=1;
if Damp==1
    A=M11*M22 - M12*M21; %s^4
    B=C11*M22 - C12*M21 - C21*M12 + C22*M11; %s^3
    C=C11*C22 - C12*C21 + K11*M22 - K12*M21 - K21*M12 + K22*M11; %s^2
    D=C11*K22 - C12*K21 - C21*K12 + C22*K11; %s
    E=K11*K22 - K12*K21;
else
    A=M11*M22 - M12*M21; %s^4
    B=zeros(2); %s^3
    C=K22*M11+C11*C22+K11*M22-K21*M12-C12*C21-K12*M21; %s^2
    D=zeros(2); %s
    E=K11*K22-K12*K21; %s^0
end

% Using companion matrix to find the roots:
I=eye(2);
Z=zeros(2);
cm=[-(B/A) -(C/A) -(D/A) -(E/A);
    I Z Z Z;
    Z I Z Z;
    Z Z I Z];

% Eigen value decomposition:
[V,DD]=eig(cm);
lambda=sort(diag(DD));
lambda_abs=abs(lambda);
vectors_fourth=V;
for i=1:8
    vectors_scaled_fourth(:,i)=vectors_fourth(:,i)./max(vectors_fourth(:,i));
end

```

```

%% Using companion matrix to find the roots:
I=eye(4);
Z=zeros(4);
if Damp==1
    cm_default=[-(damping/mass) -(stiffness/mass);
                I Z];
else
    cm_default=[-Z -(stiffness/mass);
                I Z];
end

[V_d,D_d]=eig(cm_default);
lambda_second=diag(D_d);
lambda_second_abs=abs(lambda_second);
vectors_second=V_d(5:8,:);
abs=[54.0392 -54.0392 105.2480 -105.2480 146.8870 -146.8870 218.5409
     -218.5409]*1j;
lambda_second_diag = diag(abs);
v2(1,:)=vectors_second(1,:);
v2(2,:)=vectors_second(3,:);

% Scaling vectors:
for i=1:8
    vectors_scaled_second(:,i)=vectors_second(:,i)./max(vectors_second(:,i));
    %vect2_scaled(:,i) = vect2(:,i)./max(vect2(:,i));
end
% Complete Roots:
for i=1:8
    vec_2(:,i)=[lambda_second(i)^3*v2(:,i);
                lambda_second(i)^2*v2(:,i);
                lambda_second(i)^1*v2(:,i);
                lambda_second(i)^0*v2(:,i)];
end
comp1=vec_2*D_d*pinv(vec_2);

% Absolute value of the roots:
for i=1:8
    vec_2(:,i)=[abs(i)^3*v2(:,i);
                abs(i)^2*v2(:,i);
                abs(i)^1*v2(:,i);
                abs(i)^0*v2(:,i)];
end
comp2=vec_2*lambda_second_diag*pinv(vec_2);

alpha3 = comp1(1:2,1:2);
alpha2 = comp1(1:2,3:4);
alpha1 = comp1(1:2,5:6);
alpha0 = comp1(1:2,7:8);

I=eye(2);
Z=zeros(2);
cm2=[(alpha3) (alpha2) (alpha1) (alpha0);
     I Z Z Z];

```

```

Z I Z Z;
Z Z I Z];

[V_d2,D_d2]=eig(cm2);
lamb2=diag(D_d2);
%lamb2abs=abs(lamb2);
vect2=V_d2(7:8,:);

```

A.3.3 15 DOF System

```

clc;
clear all;

NM=3
if NM==1
    N=9
    M=30
    load('C:\Users\Aditya\Documents\MATLAB\X-Modal\15 DOF\New folder
        \Dumped Poles\15.full.freq.mat')
elseif NM==2
    N=8
    M=28
    load('C:\Users\Aditya\Documents\MATLAB\X-Modal\15 DOF\New folder
        \Dumped Poles\4.full.freq.mat')
else
    N=2
    M=16
    load('C:\Users\Aditya\Documents\MATLAB\X-Modal\15 DOF\New folder
        \Dumped Poles\4.partial.freq.mat')
end

A0=POLE_RESULTS.SubCase(1,N).A(1,1).coeff;
A1=POLE_RESULTS.SubCase(1,N).A(1,2).coeff;
A2=POLE_RESULTS.SubCase(1,N).A(1,3).coeff;
MaxFreq=POLE_RESULTS.Sieve.MaxFreq
I=eye(15);
Z=zeros(15);

% Without truncation:
C1=[-A1 -A0;
    I Z];

[V,D]=eig(C1);
lambdal=MaxFreq*pi*(diag(D)/(2*pi));
psil=(V(16:30,:));
lambdalabs=sort(abs(lambdal))

% Truncated
C2=[-Z -A0;
    I Z];

[V2,D2]=eig(C2);

```



```

lambda2=(MaxFreq*pi*(diag(D2)/(2*pi)))
psi2=(V2(16:30,:));

%% Error
error=((lambda1abs-abs(lambda2))./(lambda1abs))*100

%% Extracting the actual Poles and Vectors:
for i=1:M
    pole(i)=POLE_RESULTS.SubCase(1,N).EigParms(1,i).Pole;
    vec(:,i)=POLE_RESULTS.SubCase(1,N).EigParms(:,i).Vector;
end
poler=(pole.)/(2*pi);
vector=vec;

%% Extracting the needed modal vectors from the calculated ones:
if NM==1
    P1=[11 15 21 19 27 29 25 23 17 13 1 7 9 5 3]; % Case I
    P1_trunc=[11 17 21 19 27 29 25 23 15 13 1 9 7 5 3];

    elseif NM==2
        P1=[9 11 17 19 25 27 23 21 15 13 1 3 7 5]; % Case III
        P1_trunc=[9 13 17 19 25 27 23 21 15 11 1 3 7 5];

        else
            P1=[1 3 5 7 15 13 11 9];
            P1_trunc=[1 3 5 7 15 13 11 9]; % Case IV
end

for i=1:(M/2)
    lambda1_final(i,1)=lambda1(P1(i))
    lambda1_final_abs=abs(lambda1_final)
    lambda2_final(i,1)=lambda2(P1_trunc(i))
    psi_final(:,i)=psi1(:,P1(i));
    psi_trunc(:,i)=psi2(:,P1_trunc(i));
end

%% Error
error=((lambda1abs-abs(lambda2))./(lambda1abs))*100

%% MAC

%Calculated and Truncated:
for i=1:(M/2)
    MAC2(:,i)=(psi_final(:,i)'*psi_trunc(:,i)*psi_trunc(:,i)'
        *psi_final(:,i))/(psi_final(:,i)'*psi_final(:,i)*psi_trunc(:,i)'
        *psi_trunc(:,i));
end
MAC=real(MAC2).'
```