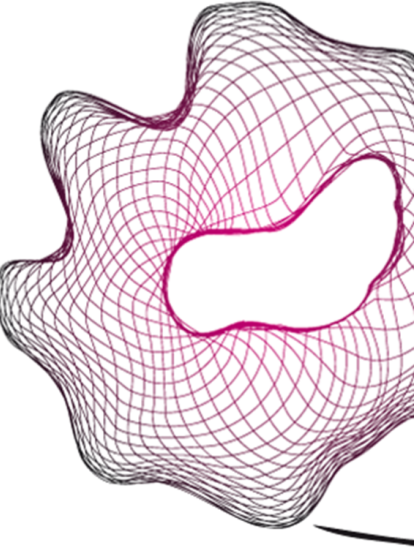


UNIVERSITY OF TWENTE.

Faculty of Mechanical Engineering
Automation & Control



APPLIED FREQUENCY-DOMAIN SYSTEM IDENTIFICATION IN THE FIELD OF EXPERIMENTAL MODAL ANALYSIS

Lennard Mulder

M.Sc. Thesis: Version 0.01

June 12, 2017

Supervisor:

Dr. ir. R. G. K. M. Aarts

Master Mechanics of Solids, Surfaces & Systems (MS3)
Faculty of Mechanical Engineering,
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Nomenclature

Symbol Discription

General

n_p	Order of polynomials
N_f	Number of frequencies
N_i	Number of inputs
N_o	Number of outputs
a	Analytical
H	Frequency response
γ	Coherence
Φ	Regression matrix
J	Jacobian matrix
Λ	Eigenvalue matrix
R	Residue matrix
L	Modal participation factor vector
\dagger	Moore-Penrose pseudoinverse
$\hat{H}_{oi}(\omega_f)$	Non-parametric estimated FRF
$H_{oi}(\omega_f)$	Parametric estimated FRF

Symbol Discription

n	mode shape index
m	master DOF index
s	slave DOF index
N	global DOF index
x	experimental
e	estimated
Ψ	Mode shape matrix
Ω	Complex Frequency
ω	Excitation frequency [rad/s]
ζ	Damping ratio
λ	Pole
ε	error term
i	Input index
o	Output index
N_n	Number of modes

Abstract

Within VDL ETG, the Structural Dynamics competence is responsible for the research to the dynamic behaviour of the fabricated products. As clients handle over more and more responsibility for the design improvements of entire systems, there is a need to streamline the process of this investigation. In this thesis the data acquisition is streamlined by means of a GUI which saves the measurements to a structured dataset.

Furthermore a user-friendly system identification technique is applied to this measured dataset to identify the modal parameters of the system under study. The developed algorithm is applied to experimental data by means of a simple steel plate and one typical mechanical system fabricated by VDL ETG to illustrate the practical applicability.

Once the modal parameters are estimated, correlation analysis is performed to verify the quality of the estimated model. These same metrics can be used to compare different models for validation purposes.

All treated theory is implemented in Matlab code and these scripts are extensively explained. All of this work is done to provide the Structural Dynamics competence at VDL ETG with a practical useful toolbox which will enable them to perform Experimental Modal Analysis (EMA) in a more convenient and efficient manner.

Contents

Nomenclature	iii
Abstract	v
List of acronyms	ix
1 Introduction	1
1.1 Motivation	1
1.2 Framework	1
1.3 Goal of the assignment	2
1.4 Report organization	2
2 Data Acquisition	3
2.1 Frequency Response Function (FRF) based Experimental Modal Analysis . .	3
2.1.1 Impact Testing	3
2.1.2 Calculation of the FRF	3
2.1.3 Performing a good impact test	5
2.1.4 Roving hammer vs. roving accelerometer	6
2.1.5 Windowing	6
2.2 Application to an experimental structure	7
2.3 Conclusion	9
3 System Identification	11
3.1 The Modal model	11
3.2 Curve fitting	12
3.2.1 Right Matrix Fractional Description model	14
3.2.2 Identification of a Right Matrix Fractional Description model	14
3.3 Poles identification	17
3.4 Residue identification	19
3.5 Implementation using simulated data	20
3.5.1 Rectangular plate	20
3.5.2 Bottom Frame	26
3.6 Application to an experimental structure	31
3.6.1 rectangular Plate	31

3.6.2	Bottom frame	34
3.7	Conclusion	38
4	System Correlation	39
4.1	Correlation metrics	39
4.1.1	Conclusion	42
4.2	Obtaining the analytical modal model	42
4.2.1	Mass normalisation	43
4.3	Model Reduction/Expansion	44
4.3.1	Reducing the analytical model	45
4.3.2	Orthogonality checks	46
5	Conclusions and recommendations	49
5.1	Conclusions	49
5.2	Recommendations	49
	References	51
	Appendices	
A	Laplace, Fourier and Z-Transforms	53
B	Correlation metrics of identified systems	55
C	Deducing analytical models from ANSYS FEM	57
C.1	Matlab Code	60
D	The System Equivalent Reduction Expansion Process (SEREP) technique	65
D.1	Matlab Code	67
E	Correlation analysis evaluated for the LTA2 Bottom Frame (BF)	71
E.0.1	Conclusions	73

List of acronyms

BF LTA2 Bottom Frame

COMAC COordinate Modal Assurance Criterion

DOF Degree Of Freedom

DFT Discrete Fourier Transform

ECOMAC Enhanced COordinate Modal Assurance Criterion

EMA Experimental Modal Analysis

ETG VDL Enabling Technologies Group

FEM Finte Element Model

FRAC Frequency Response Assurance Criterion

FRF Frequency Response Function

FRI Frequency Region of Interest

FRM Frequency Response Matrix

LS Least Squares

MAC Modal Assurance Criterion

MFD Matrix Fraction Description

MSF Modal Scale Factor

NLS Non-linear Least Squares

ODS Operational Deflection Shape

pLSCF Poly-reference Least-Squares Complex Frequency

POC Pseudo-Orthogonality Check

RHP Right Half Plane

RHS Right Hand Side

RMFD Right Matrix Fraction Description

SEREP System Equivalent Reduction Expansion Process

SNR Signal to Noise Ratio

SVD Singular Value Decomposition

VDL Van Der Leegte groep

Introduction

1.1 Motivation

Van Der Leegte groep (VDL) VDL Enabling Technologies Group (ETG) is supplier of many complex mechatronic systems which are manufactured at site. These systems have to meet strict specifications which all have to be verified before the client accepts these products. Furthermore, clients hand over more and more responsibility to VDL ETG involving not only the manufacturing of certain products but also to development of new versions of these systems. So there is a need to perform EMA, in order to gain insight into the structural dynamics of the variety of systems fabricated by VDL ETG. This is key to make a well founded consideration about what modifications should be applied to obtain the desired dynamic specifications.

In the last decades several system identification methods have been developed. Thanks to nowadays developments of computational power of computers these methods are able to handle large datasets in a short matter of time, making the application of EMA more convenient.

1.2 Framework

The presented research is commissioned by the structural dynamics competence plan of VDL ETG, managed by ir. M. Bruin. The assignment is in collaboration between the competence group Structural Dynamics at VDL ETG and the chair Applied Mechanics, at the University of Twente.

There are already several commercial software packages¹ available which enable to perform such an EMA, but these packages are very expensive, contain many irrelevant functionalities and they are black box. A self-build GUI which is dedicated to the kind of systems which VDL ETG manufactures is preferred. Self-written matlab code is developed based on known methods. Primarily the data acquisition requires a GUI, since this significantly reduces the time spent to measure a complete column of the Frequency Response Matrix (FRM). A GUI

¹<https://www.plm.automation.siemens.com/en/products/lms/testing/test-lab/structures/modal-analysis.shtml>

can help the engineer to consistently save all different measurements into one dataset. Finally, once an EMA is performed, engineers at VDL ETG want to compare the identified modal model to compare it to the Finite Element Model (FEM) of the system under study. Any discrepancies between both models have to be investigated to gain insight in the correctness of the FEM. Later on this information can also be used to update the FEM to improve accuracy. Nowadays the verification of analytical models utilizing experimental data is very desirable.

1.3 Goal of the assignment

The main goal of this assignment is develop a stable, user-friendly, efficient algorithm which enables the Structural Dynamics competence of VDL ETG to perform a full EMA. In order to do so, the distinct goals must be met:

1. Implement a stable, user-friendly, efficient curve fitting technique
2. Estimate the modal model using the fitted model
3. Quantify the correlation between measured and estimated models to validate the fit
4. Develop a GUI which streamlines the process of acquiring a full FRM.

In this assignment a broad research is done to the state-of-the-art of EMA and extensive comparison is made. Based on the applicability of these different techniques this knowledge is then implemented into Matlab code and dedicated to the typical systems fabricated at VDL ETG.

1.4 Report organization

The remainder of this report is organized as follows. In Chapter 2 the data acquisition is treated. It describes the considerations to accurately measure the dynamic behaviour of the system. The result of the step is the non-parametric FRM.

Then, in Chapter 3, this measured FRM is fitted by a modal model. A global curve fitting technique is used to identify the poles of this representation.

Once the modal model is estimated, the correlation between measured and estimated models are quantified using correlation analysis, described in Chapter 4. These metrics can be used in several ways; For instance to indicate discrepancies between representations. They can also be used to verify the quality of the estimated model

Finally, in Chapter 5, conclusions and recommendations are given.

Data Acquisition

This chapter describes the experiments required to perform a modal analysis on a mechanical system. Non-parametric system identification is applied to the in- and output spectra to obtain a set of FRFs; the FRM. This is an important step since it will involve the quality of the rest of the EMA.

2.1 FRF based Experimental Modal Analysis

There is a variety of algorithms in the field of parametric system identification which start directly from the measured in- and output time histories. For a typical EMA experiment this implies that a large dataset has to be fitted and it is recommended to apply pre-processing to the data to transform the measurement into the frequency domain. This reduces the size of the dataset and lowers the noise levels before the parametric identification is initialized. This step is called: *non-parametric identification*. The result of this step is the estimated FRF. These FRFs are combined into one matrix, the FRM, which will be used for the parametric system identification in the Chapter 3.

2.1.1 Impact Testing

The first step in the procedure is to measure the impulse response of a system, by means of an impact test. This is done using an impact hammer and multiple acceleration sensors. The in- and output signals between each excitation point and each measured output are measured using a data acquisition module, which converts the analog signals into discrete time signals which are communicated to the computer at fixed time intervals defined by the sampling frequency. These in- and output signals are transformed to the frequency domain by means of the Discrete Fourier Transform (DFT). These transformed signals are used to calculate the Frequency Response Function.

2.1.2 Calculation of the FRF

This step is also called, non-parametric system identification, since the time-signals are converted into a non-parametric curve in the frequency domain. Discrete Fourier Transformation

transforms the measurement data into the frequency domain by partitioning the entire set of measurement data, containing N_t datapoints, into smaller sections, so-called frequency points. To assure this representation is correct, the Shannon theorem should be obeyed: *If a function $x(t)$ contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced $1/(2B)$ seconds apart.* To assure that the algorithm works fast, the number of frequency points N_{FFT} should be a power of two. Therefore the preferred number of frequency points will be:

$$N_{FFT} = 2^A, \quad A = \left\lceil \text{floor} \left(\frac{\log(N_t)}{\log(2)} \right) \right\rceil \frac{1}{2^x} \quad (2.1)$$

In this the real positive integer x can be chosen equal to or larger than zero. In case of low Signal to Noise Ratio (SNR) increasing this value increases the accuracy of the transformation. The highest frequency which can be retrieved is determined by the Nyquist frequency, which is equal to half the sampling frequency. And the lowest frequency that can be retrieved equals $\frac{f_s}{N_{FFT}}$. In order to improve SNR of the measurements, the time-series are multiplied by weighting functions before the DFT conversion. This technique is called windowing and is treated more extensively in paragraph 2.1.5. The frequency representations of the measurements can now be used to calculate the Frequency Response Function. This can be done in several ways, depending on the expected noise of the measurement. The two most commonly used estimates are:

$$H(f) = \frac{Y(f)}{U(f)} \rightarrow H_1(f) = \frac{S_{yu}(f)}{S_{uu}(f)} \quad H_2(f) = \frac{S_{yy}(f)}{S_{uy}(f)} \quad (2.2)$$

in which S_{uy} , S_{uu} and S_{yy} are the Cross Spectral Density and the Auto Spectral Densities of frequency spectra of both in- and output-signals. In case of noiseless data both calculations will give the same result. First of all, by dividing the cross-power spectral density of both in and output-signal by the auto-power spectral density of the input signal, is based on the assumption that the output of the system is expected to be noisy when compared to the input. The second estimate assumes the input to the system to be noisy when compared to the output. The $H_1(f)$ estimator underestimates the actual $H(f)$ whereas the $H_2(f)$ estimator gives an overestimation. So by calculating both one can obtain the limits of the real FRF. There are also other estimates, but that is beyond the scope of this study. These two estimates can be used to calculate the coherence $\gamma_{uy}^2(f)$ of the measurement. This number is calculated by the ratio of both estimators:

$$\gamma_{uy}^2(f) = \left\| \frac{S_{yu}(f)S_{uy}(f)}{S_{uu}(f)S_{yy}(f)} \right\| = \left\| \frac{H_1(f)}{H_2(f)} \right\| \quad (2.3)$$

This indicates the quality of the measurement. If the noise level is low, the uncertainty will be small and both bounds will be close to each other resulting in a coherence approaching unity. At resonance frequencies the ratio will drop to zero, since the upper bound approaches infinity for low damped systems. Therefore the bounds will be far apart, resulting in a drop in ratio value. If the coherence is approaching unity for all other frequencies in the entire frequency region of interest, the measurement is done correctly and the calculated FRF is reliable.

2.1.3 Performing a good impact test

First of all the excitation location and direction must be considered. Therefore the local flexibility of the structure must be considered to assure that the whole system is excited sufficiently. If this is done correctly the measurement itself can be analysed. For a good measurement the input force must have the following characteristics:

- One single peak, described by multiple data points; to assure accurate DFT
- A short impulsive excitation; lasting less than 5% of the sample interval
- High amplitude for a broad range of frequencies, well above the noise level of the instrumentation
- Amplitude must be evenly distributed throughout this frequency range

This results in a FRF in which the peaks indicate the natural frequencies, since for all frequencies the input is equal. Such a spectrum is achieved by using a hammer. The length of time of the impact pulse determines the width of the range of excited frequencies. The shorter the impact time, the broader the frequency domain response. To clarify this; there is an inverse correlation between the time and frequency domain. Signals with a short duration in time, have a broad response in frequency domain and vice versa. For instance a broad time signal, a sine wave, will have a very narrow frequency spectrum will show up in the frequency domain at only one frequency. So the shorter the impact time the more frequencies can be excited.

The impact hammer is used to achieve this short impact force. The hammer can be adjusted for a specific case, i.e. to excite the frequencies which are of interest. This input frequency range can be adjusted by means of adding mass or changing the stiffness of the tip. This will have the following effect on the input spectrum.

- Changing the stiffness; The time in which the hammer is in contact with the structure can be changed by choosing a more rigid tip, for instance replacing the rubber tip with a metal one.
- Changing the mass; Increased mass lengthens the time in which the direction of the hammer is reversed.

The desired result is a clean FRF over the full frequency range of interest together with a relatively evenly distributed input spectrum for that same frequency range. The right hammer configuration depends always on the specific case. If all modes are excited, but also modes beyond the Frequency Region of Interest (FRI), the FRF maybe have a lower SNR at the lower frequency region, caused by the so called *out-of-band* overloads. This indicates that the tip may be too hard. Another effect which can show up in the measured FRF, is noise at the higher frequencies. This indicates that the hammer is too soft. So the user should always make a well-founded choice of the configuration of the impact hammer since the quality of the measurement will have influence on all further steps in the modal analysis

procedure. The correct configuration will induce sufficient energy to excite the full frequency range of interest, while the frequencies beyond this region will contain significantly lower energy. Furthermore, the configuration must be such that there is enough force is begin input to the structure, to excite all modes of the structure. So for a giant structure, e.g. a bridge, also a giant impact hammer with sufficient mass will be required. Finally, the quality of the measurement can also be indicated by means of the coherence. If the coherence approaches one for entire frequency region of interest the measurement will be sufficiently accurate. These aspects should be checked by the user for every measurement, to assure that the FRM contains accurate measurements.

2.1.4 Roving hammer vs. roving accelerometer

In order to determine the mode shapes of a structure, one use either measure a full row or column of the FRM. This corresponds to either roving the hammer over all measurement points while holding the accelerometer at a fixed place, or excite the system at one place and direction while roving the accelerometer. It will depend on the case which procedure is preferred. The following considerations have to be made.

At first sight, roving the hammer has some practical advantages over roving the accelerometer, since one does not have un- and remount the sensor. In case of very light structure, moving the accelerometer will also influence the weight distribution, which influences the measured dynamical behaviour of the system. There are also drawbacks on moving the hammer, since the hammer is a single input and most accelerometers are tri-axial. So one must note that all directions are excited. In practise it will be not possible to excite each point in all there dimensions, for instance for a plate, which can only be excited in perpendicular direction of all points. If not all directions are excited or measured incomplete columns will be measured, which will also result in incomplete mode shapes. So for some cases, only one method is applicable.

2.1.5 Windowing

In order to reduce the presence of noise in the measured signal the data is edited before the DTFT analysis is applied, in order to improve the SNR. This technique is called *windowing* and the windows are weighting functions that have been developed for the transient signals. For impulse testing two common time domain windows are applied; the *force* and *exponential* window which are applied to the in- and output signals respectively.

Ideally, the impact of the hammer is described by one peak in the input signal and all other samples are equal to zero. Due to noise all other samples will be non-zero and the force window will turn them into zeros to get a more reliable signal, after all the hammer has no contact with the system outside the impulse, so all measured force will not be applied to the measured system.

The output signals will be edited by the exponential window. The effect of this window is to increase the apparent damping of the measured system. For instance, if a system contain a

very lightly damped suspension frequency it can take a large amount of time before the system comes at rest. To reduce measurement data, the measurement can be stopped before the system is at rest. However, since the signals will not be decayed to zero, leakage¹ will occur. This has the effect an underestimation of the magnitude, distortion of the phase and a drop in coherence at the peaks. The weights of exponential window can make the decay closer to zero to reduce the leakage. The amount of added damping which is predictable and can be compensated when the modal parameters are estimated. This trick is governed by the shift property of the Laplace Transform. However, one must beware of the fact that adding too much damping to the signals can make two closely spaced peaks undistinguishable, since the two peaks will combined into one in case of high damping, as illustrated in figure 2.1. So noise can be reduced, but not be removed entirely by the applying windows to the transient signals.

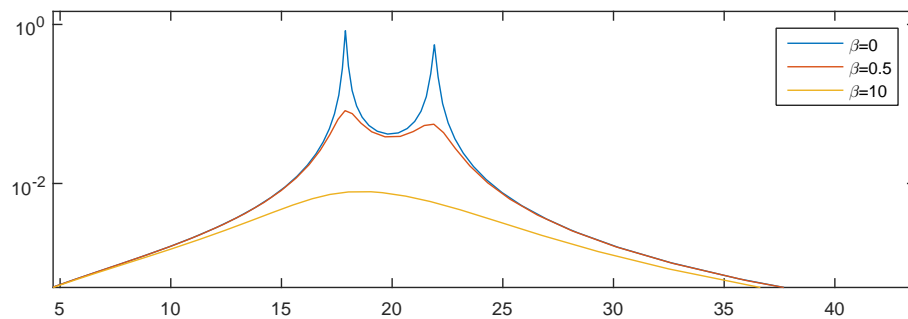


Figure 2.1: FRF of lightly damped system with closely spaced peaks. The increase in damping by application of exponential window

2.2 Application to an experimental structure

In this project the data acquisition device DT9837B device is used: This device is a high

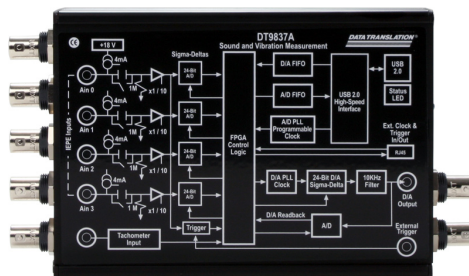


Figure 2.2: DT9837 data acquisition module

accuracy dynamic signal acquisition module ideal for vibration measurements. It contains 4 synchronized 24-bit sensor inputs which provide a data stream that is matched in time,

¹ Leakage: a signal processing bias error which is caused by the fact that the begin and end conditions of the signal are unequal to zero

which can be used for signal analysis. During the tests only one SISO transfer is measured, using one hammer and one uniaxial accelerometer. Each experiment was repeated at least 5 times, and the best 5 measurements were used to determine an averaged *Frequency Response Function* for the corresponding SISO transfer. In this the best is based on a selection of 5 measured responses, which result in the highest coherence

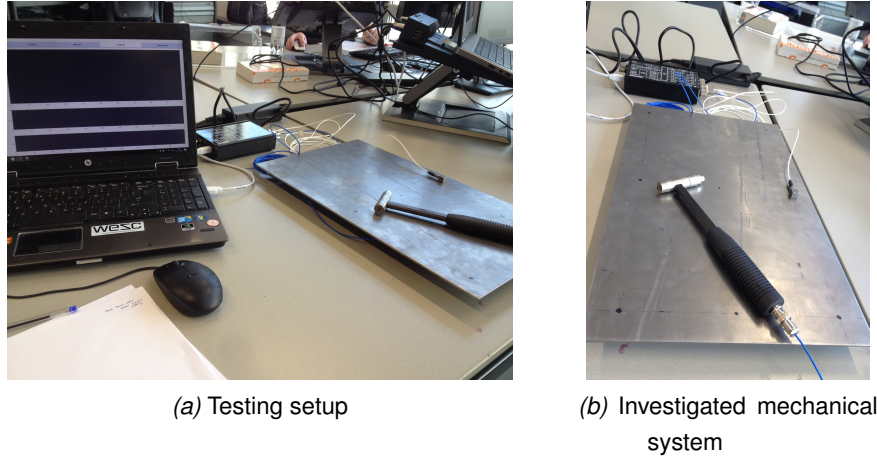


Figure 2.3: Equipment used during experiments

To verify the correctness of the data acquisition, an simple rectangular plate is used. The mode shapes of this mechanical system are well known and therefore it can illustrate the correctness of the algorithms. A (5×5) grid is applied to the plate, resulting in 25 possible in- and output positions, see figure 2.4. A square (25×25) *Frequency Response Matrix* can be made from the dynamic measurements between all grid points in which each entity contains the corresponding measured FRF. This results in large data set will be fitted with a mathematical model in the next phase of the the modal analysis: System Identification.

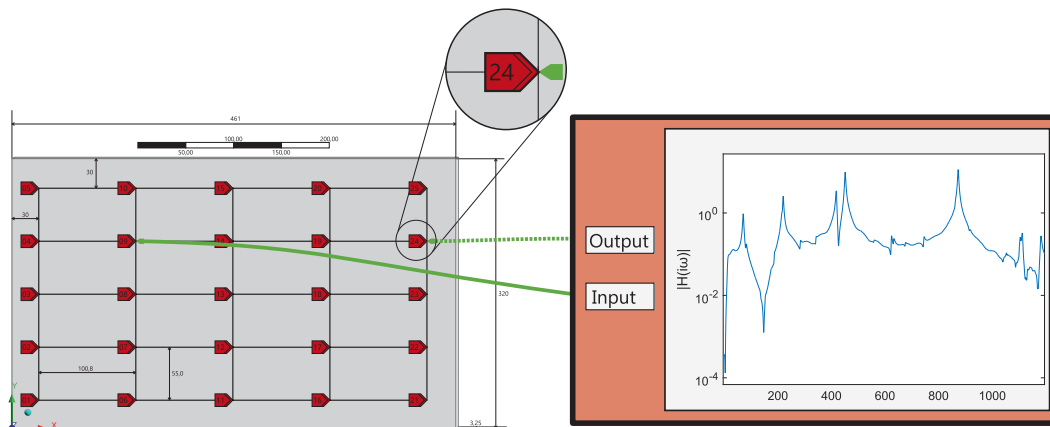


Figure 2.4: Measurement of $H(24, 9)$

All measured FRFs are shown in figure 2.5 and this already shows the large amount of data which must be fitted in the next phase of the experimental modal analysis. The shown

FRFs are equal to the H_2 estimates of the response spectra. The FRFs show 15 clear lightly damped peaks which can be explained by the fact the measured system is a the rectangular plate. One can also see that the first two resonances and also the 10th and 11th resonance are closely spaced, indicated by the black curves. The identification methods must be able to distinguish all the natural frequencies, since in practise many mechanical systems will contain symmetry which results in closely spaced natural frequencies.

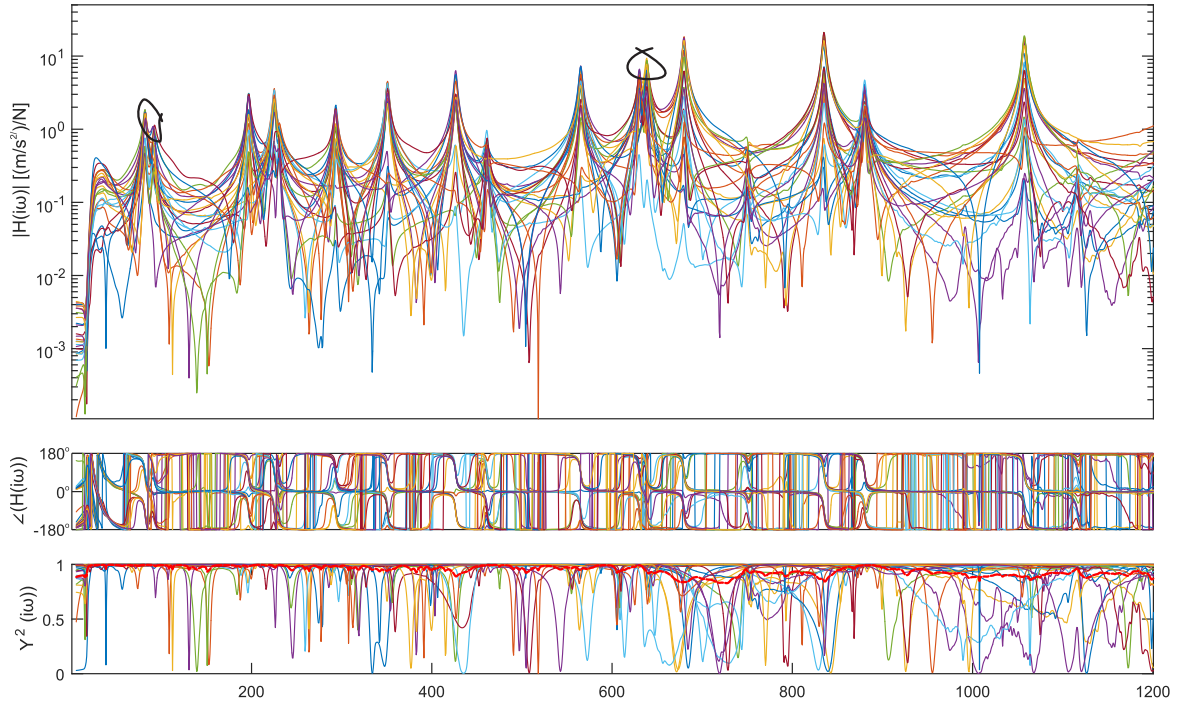


Figure 2.5: Frequency Response Matrix; 22nd column

Also one can see a clear drop of magnitude at low frequencies, this is due to the high-pass filter which is included in the data acquisition module. This part of the FRF will also be not be used in the system identification, thanks to the frequency domain transformation this part can be excluded easily.

2.3 Conclusion

The data acquisition is the key phase of the experimental modal analysis, since it influences the accuracy of all further steps in the process. The measurements and their considerations are evaluated and should be obeyed precisely to obtain an accurate FRM. In order to illustrate what such a data acquisition procedure involves a rectangular plate is used. In upcoming chapter also more complex systems will be treated. The main reason to start with a rectangular plate is the fact that the rectangular plate has a response which contains few damping and all mode shapes are well known. So this relatively simple system illustrates how the FRM is measured.

System Identification

This chapter describes all steps involved in the process in which the measured data is used to identify a *modal model*; a mathematical model which describes the dynamic behaviour of the measured mechanical system in terms of modal parameters. First the FRFs are fitted by algebraic equations, called *curve fitting*. The roots of these polynomials are related to the poles of the system. These poles λ_n of the system define the denominator part of the modal model. Subsequently this set of poles is fed into the *residue estimator*, which estimates the so called *residues* R_n . Finally these residues are combined into vectors, which describe the estimated mode shapes Ψ_{mn} . In the coming sections each step is explained extensively. Finally, two mechanical systems are used to illustrate the performance of the system identification algorithm. First of all the simple steel plate which was also used to illustrate the data acquisition, shown in section 2.2. The second mechanical system is a bottom frame which are produced by VDL ETG. In the first two cases, both FRMs are simulated using the FEMs of both systems. In case 3 and 4 the FRM contains experimental data of the steel plate and bottom frame respectively.

3.1 The Modal model

This model assumes that the entire dynamic behaviour of a system can be described by a cumulation of a set of subsystems, which are called modes. The linearised equations of motion of a general mechanical system in a certain configuration are given by:

$$\bar{\mathbf{M}}\ddot{\mathbf{q}}(t) + \bar{\mathbf{D}}\dot{\mathbf{q}}(t) + \bar{\mathbf{K}}\mathbf{q}(t) = \mathbf{f}(t) \quad (3.1)$$

where $\bar{\mathbf{M}}$ is the system mass matrix, the damping matrix $\bar{\mathbf{D}}$ and the stiffness matrix $\bar{\mathbf{K}}$. The vector $\bar{\mathbf{B}}$ is called the input vector and $\bar{\mathbf{C}}$ is called the output matrix. Furthermore the vector \mathbf{q} contains the degrees of freedom of the system. The measured accelerations are stored in vector \mathbf{y} , and the measured input force of the hammer is stored in the scalar u . The matrices $\bar{\mathbf{M}}$, $\bar{\mathbf{D}}$ and $\bar{\mathbf{K}} \in \mathbb{R}^{N_m \times N_m}$, where N_m is the number of independent degrees-of-freedom; the length of \mathbf{q} . Using the Laplace transform and neglecting initial conditions results in the frequency-domain equivalent for above equation given by:

$$\mathbf{Z}(s)\mathbf{Q}(s) = \mathbf{F}(s) \quad (3.2)$$

in which $\mathbf{Z}(s) = \bar{\mathbf{M}}s^2 + \bar{\mathbf{D}}s + \bar{\mathbf{K}}$. Next step is to invert the above equation:

$$\mathbf{Q}(s) = \mathbf{H}(s)\mathbf{F}(s) \quad (3.3)$$

in which $\mathbf{H}(s) = \mathbf{Z}^{-1}(s)$ is called the *transfer function matrix*. This matrix can be expressed in its modal form; in terms of the modal parameters, $(\lambda_m, \phi_m$ and $L_m)$ respectively the pole, mode shape and modal participation factor of mode m .

$$\mathbf{H}(s) = \phi [s\mathbf{I}_{N_m} - \mathbf{\Lambda}]^{-1} L^T + \phi^* [s\mathbf{I}_{N_m} - \mathbf{\Lambda}^*]^{-1} L^H \quad (3.4)$$

$$= \sum_{n=1}^{N_m} \left(\frac{\phi_n L_n^T}{s - \lambda_n} + \frac{\phi_n^* L_n^H}{s - \lambda_n^*} \right) = \sum_{n=1}^{N_m} \left(\frac{\mathbf{R}_n}{s - \lambda_n} + \frac{\mathbf{R}_n^*}{s - \lambda_n^*} \right) \quad (3.5)$$

in which $\mathbf{\Lambda}$ is a diagonal matrix containing all λ_n and N_m the number of modes. The poles $\lambda_n = \sigma_n + j\omega_n$ contain the natural frequencies $f_n = \omega_n/(2\pi)$ and corresponding damping ratios $d_n = -\omega_n/\sqrt{\sigma_n^2 + \omega_n^2}$. And the numerator terms can be substituted for the residues \mathbf{R}_n . Now the linearised equations of motion are rewritten in terms of modal parameters. This modal model can be used to compare it with other modal models based on analytical or other experimental models. The estimated modal parameters can be used to update a FEM or e.g. to check if the mechanical system meets its specifications. If any discrepancies between these datasets are measured, the mode shapes can give the insight into from which part of the system these discrepancies originate, which can be very helpful in the verification of a assembled mechanical system.

3.2 Curve fitting

There is a variety of curve fitting methods as to see in [1] and each method has its pros and cons. Frequency Domain estimators were preferred, since estimation in the frequency domain is a more convenient estimation process as this enable the user to select a FRI, to specify which part of the spectrum is relevant for the modal analysis. This particularly is given preference for so called *stiff systems*; systems that have both very slow and very fast modes, which are difficult to handle using time-domain measurements. In order to capture the slow modes a very long measurement time is required and a high sampling frequency to capture the fast modes. As a results the dataset to be fitted will be very large, which has negative influence on the computational afford of the identification process. The advantage of the frequency domain identification algorithms is the fact that the frequency domain measurement do not have to be equidistant. This can keep the size of the dataset relatively small, while the system can still be identified accurately. For this assignment a stable, user-friendly algorithm was found in the Poly-reference Least-Squares Complex Frequency (pLSCF) method, in which a Right Matrix Fraction Description (RMFD) model is estimated. More information about this model is given in section 3.2.1.

Complex frequencies The important contribution to the development of system identification in the frequency domain is the relatively new method of frequency mapping. In this

method, the generalized frequency called the z -transform, is a trigonometric mapping function; a complex number that has superior numerical conditioning to orthogonal polynomials without the drawback that a generalized companion matrix eigen value problem has to be solved. The frequency axis, which contains all selected frequencies of interest, stretches from f_1 until f_{N_f} and is mapped onto a half unit circle in the upper half of the complex plane. For clarity, in appendix A an elucidation between the Laplace, Fourier, and z -Transform is given. The discrete polynomial basis $\Omega_k(\omega_f)$ functions are given by:

$$\Omega_k(\omega_f) = z^k = e^{i\pi(\omega_k/N_f)}, \quad k = 1, 2, \dots, N_f \quad (3.6)$$

These discrete frequencies are used in identification methods and improve the numerical stability of the algorithms. Since these complex basis function are implicitly orthogonal with respect to the unit circle, a well conditioned Jacobian matrix \mathbf{J} is usually obtained. This will be cleared out in section 3.2.2.

Mode Shapes and Operating Deflection Shapes When a system is excited at or near one of its eigen-frequencies, the corresponding mode shape will dominate. However, there will still be a small contribution from all other modes present in this response. The measured vibration shape is called the Operational Deflection Shape (ODS). Originally the ODS is defined as the deflection of a structure at a particular frequency.

There are some subtle differences between ODS and mode shapes. First of all, ODS are measured directly while the mode shapes are obtained after post-processing these data. Secondly, the mode shapes are inherent properties of a structure, while the ODS will change if the input spectrum changes. Therefore the ODS is expressed in units, while the mode shapes are dimensionless. All mode shapes together describe the complete dynamic response of a structure. In the case that two modes are closely coupled to each other, the ODS is a linear combination of both mode shapes. In reality, all modes are contributing in the ODS and the measurement is also disturbed by noise. More details can be found in the original source [2].

Normal and Complex Mode Shapes The FEM is generally undamped and the corresponding mode shapes are real. On the other hand, because damping is always present in real-life structures, experimental modal analysis yields complex mode shapes. When the numerical and experimental modes are to be correlated either damping must be added to the FRM or only the real part of the experimental FRM compared. For VDL ETG, most systems fabricated by VDL ETG are relatively stiff and therefore contain low damping. This means only normal modes will be considered, containing so called standing waves. In this the deformations are varying sinusoidal with no phase difference. So all deformation are maximal at the same time. More details about the complex mode shapes can be found in the original source [3].

3.2.1 Right Matrix Fractional Description model

The common-denominator model is multi-variable transfer function model described by a Matrix Fraction Description (MFD) i.e. a matrix of which each element contains a fraction of two matrix polynomials. The relationship between input i ($i = 1, \dots, N_i$) and all outputs can be described by this model:

$$\hat{H}_o(\omega_f) = N_o(\Omega_j(\omega_f)) \cdot D^{-1}(\Omega_j(\omega_f)) \quad (3.7)$$

This equation can be transformed from continuous to discrete form by using discrete polynomial basis functions. In that case, the numerator row-vector polynomial ($N_o \in \mathbb{C}^{1 \times N_i}$) of output o is:

$$N_o(\omega_f) = \sum_{j=0}^n \Omega_j(\omega_f) \mathcal{B}_{oj} \quad (3.8)$$

and the denominator matrix polynomial ($D \in \mathbb{C}^{N_i \times N_i}$) is equal to:

$$D(\omega_f) = \sum_{j=0}^n \Omega_j(\omega_f) \mathcal{A}_j \quad (3.9)$$

So each transfer function can be represented with an analytical expression; a fraction of these two polynomials. These coefficients are found using a curve-fitting technique. Subsequently these coefficients are transformed into modal parameters; First the denominator polynomial D is reformulated into a generalized eigenvalue problem, resulting in $n_p N_o$ eigenvalues and corresponding right eigenvectors. And once the poles and modal participation factors are known, the numerator N is used to calculate the residues R_m using a residue estimator. These residues are then transformed into mode shapes, using Singular Value Decomposition (SVD). The benefit of fitting all FRFs of all inputs to output o is that the modal participation factors and mode shapes directly; by solving a linear least squares problem. Since the modal model is linear in the mode shapes. After completing both steps, all polynomial coefficients are transformed into modal parameters. Which again results in the modal model described by equation 3.4.

3.2.2 Identification of a Right Matrix Fractional Description model

The estimation of the polynomial coefficients is obtained by the pLSCF estimator. The working principles of this algorithm is explained in this section.

All coefficients of the numerator- and denominator polynomials are real valued. These matrices are combined into one giant matrix with has the symbol θ , the optimal solution matrix.

$$\beta_o = \begin{bmatrix} \mathcal{B}_{on_p} \\ \vdots \\ \mathcal{B}_{o1} \\ \mathcal{B}_{o0} \end{bmatrix} \in \mathbb{R}^{(n_p+1) \times N_i}, \quad \alpha = \begin{bmatrix} \mathcal{A}_{n_p} \\ \vdots \\ \mathcal{A}_1 \\ \mathcal{A}_0 \end{bmatrix} \in \mathbb{R}^{N_i(n_p+1) \times N_i} \rightarrow \theta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{N_o} \\ \alpha \end{bmatrix} \quad (3.10)$$

The size of this solution matrix is $\theta \in \mathbb{R}^{(N_i+1)(n+1) \times N_i}$. The estimation of the coefficients can be obtained by minimizing the non-linear error between the model and the measured dataset \bar{H}_o .

$$\xi_o^{NLS}(\omega_f, \theta) = N_o(\omega_f, \beta_o) D^{-1}(\omega_f, \alpha) - \bar{H}_o(\omega_f) \quad (3.11)$$

The error $e_{NLS}(\theta)$ is equal to the following Non-linear Least Squares (NLS) cost function; non-linear with respect to the parameter matrix θ :

$$e_{NLS}(\theta) = \sum_{o=1}^{N_o} \sum_{f=1}^{N_f} \text{tr} \left((\xi_o^{NLS}(\omega_f, \theta))^H \xi_o^{NLS}(\omega_f, \theta) \right) \quad (3.12)$$

where $\text{tr}(\cdot)$ is the trace operator. This boils down to the fact that the NLS error $\xi_o(\omega_f)$ is a row-vector which must be minimized.

Linear-in-the-parameters A (sub-optimal) linear Least Squares (LS) cost function can be used to approximate this NLS one. By right-multiplying $e_{NLS}(\theta)$ with $D(\omega_f, \alpha)$ the NLS cost function 3.12 becomes linear in the parameters:

$$\xi_{LS}(\theta) = W_o(\omega_f) \xi_o^{NLS} D(\omega_f, \alpha) \quad (3.13)$$

$$= W_o(\omega_f) (N_o(\omega_f, \beta_o) - \bar{H}_o(\omega_f) \cdot D(\omega_f, \alpha)) \quad (3.14)$$

$$= W_o(\omega_f) \sum_{j=0}^n (\Omega_j \cdot \mathcal{B}_{oj} - \Omega_j(\omega_f) \bar{H}_o(\omega_f) \mathcal{A}_j) \quad (3.15)$$

In this $W_o(\omega_f)$ are weighting functions, which can be used to improve the estimation. For instance, the coherence of the measurement, equation 2.3, can be used to bring forward the parts of the measured FRF which are most accurate. These equations are 'linear-in-the-parameters' for $f = 1, \dots, N_f$ and can be reformulated into matrix form:

$$\xi_o^{LS}(\theta) = \begin{Bmatrix} \xi_o^{LS}(\omega_1, \theta) \\ \xi_o^{LS}(\omega_2, \theta) \\ \vdots \\ \xi_o^{LS}(\omega_{N_f}, \theta) \end{Bmatrix} = [X_o \ Y_o] \cdot \begin{bmatrix} \beta_o \\ \alpha \end{bmatrix} = J_o \cdot \theta \quad (3.16)$$

where:

$$X_o = \begin{bmatrix} (W_o(\omega_1) [\Omega_0(\omega_1) \ \dots \ \Omega_n(\omega_1)]) \otimes I_{N_i} \\ \vdots \\ (W_o(\omega_{N_f}) [\Omega_0(\omega_{N_f}) \ \dots \ \Omega_n(\omega_{N_f})]) \otimes I_{N_i} \end{bmatrix} \in \mathbb{C}^{N_f \times (n_p+1)} \quad (3.17)$$

$$Y_o = \begin{bmatrix} -(W_o(\omega_1) [\Omega_0(\omega_1) \ \dots \ \Omega_n(\omega_1)]) \otimes \bar{H}_o(\omega_1) \\ \vdots \\ -(W_o(\omega_{N_f}) [\Omega_0(\omega_{N_f}) \ \dots \ \Omega_n(\omega_{N_f})]) \otimes \bar{H}_o(\omega_{N_f}) \end{bmatrix} \in \mathbb{C}^{N_f \times N_i(n_p+1)} \quad (3.18)$$

in which \otimes is the Kronecker product. The system can be transformed into a more compact one; the *reduced normal equations*. First of all the former two matrices are entities of the

Jacobian matrix \mathbf{J} :

$$\mathbf{J} = \begin{bmatrix} \mathbf{X}_1 & 0 & \dots & 0 & \mathbf{Y}_1 \\ 0 & \mathbf{X}_2 & \dots & 0 & \mathbf{Y}_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 \dots & \mathbf{X}_{N_o} & \mathbf{Y}_{N_o} \end{bmatrix} \quad (3.19)$$

This Jacobian matrix \mathbf{J} of this LS problem has $N_i N_o N_f$ rows and $(n_p + 1)(N_i N_o + 1)$ columns, to assure that the problem is well-conditioned the number of frequencies should be way more than the order of polynomials ($N_f \gg n_p$). This matrix, and therefore the system will have a huge size for practical implementation, since the experiments of a frame can easily require 40-60 outputs, several inputs and thousands of frequency points. This leads to a numerical 'stiff' system, which requires much computational effort.

The (Weighted) linear LS cost function, which is used to find the optimal solution:

$$\mathbf{e}_{LS}(\boldsymbol{\theta}) = \sum_{o=1}^{N_o} \text{tr} \left((\boldsymbol{\xi}_o^{LS}(\boldsymbol{\theta}))^H \cdot \boldsymbol{\xi}_o^{LS}(\boldsymbol{\theta}) \right) \quad (3.20)$$

$$= \sum_{o=1}^{N_o} \text{tr} \left(\begin{bmatrix} \boldsymbol{\beta}_o^T & \boldsymbol{\alpha}^T \end{bmatrix} \cdot \begin{bmatrix} \mathbf{R}_o & \mathbf{S}_o \\ \mathbf{S}_o^T & \mathbf{T}_o \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\beta}_o \\ \boldsymbol{\alpha} \end{bmatrix} \right) \quad (3.21)$$

$$= \text{tr}(\boldsymbol{\theta}^T \cdot \mathbf{J}^H \mathbf{J} \cdot \boldsymbol{\theta}) \quad (3.22)$$

where the matrices $\mathbf{R}_o = \mathbf{X}_o^H \mathbf{X}_o \in \mathbb{C}^{(n+1) \times (n_p+1)}$, $\mathbf{S}_o = \mathbf{X}_o^H \mathbf{Y}_o \in \mathbb{C}^{(n_p+1) \times N_i(n_p+1)}$ and $\mathbf{T}_o = \mathbf{Y}_o^H \mathbf{Y}_o \in \mathbb{C}^{N_i(n_p+1) \times N_i(n_p+1)}$ contain the derivatives $\frac{\partial \boldsymbol{\xi}_o^{LS}}{\partial \boldsymbol{\theta}}$. The entries of these matrices can be described by the following expressions:

$$[\mathbf{R}_o]_{rs} = \text{Re} \left(\sum_{k=1}^{N_f} |W_k(\omega_k)|^2 \cdot \Omega_{r-1}^H(\omega_k) \Omega_{s-1}(\omega_k) \right) \quad (3.23)$$

$$[\mathbf{S}_o]_{rj} = -\text{Re} \left(\sum_{k=1}^{N_f} |W_k(\omega_k)|^2 \mathbf{H}_o(\omega_k) \cdot \Omega_{r-1}^H(\omega_k) \Omega_{s-1}(\omega_k) \right) \quad (3.24)$$

$$[\mathbf{T}_o]_{ij} = \text{Re} \left(\sum_{k=1}^{N_f} |W_k(\omega_k)|^2 \mathbf{H}_o^H(\omega_k) \mathbf{H}_o(\omega_k) \cdot \Omega_{r-1}^H(\omega_k) \Omega_{s-1}(\omega_k) \right) \quad (3.25)$$

with $i = [(r-1)N_i + 1 : rN_i]$, $j = [(s-1)N_i + 1 : sN_i]$ for both $r, s = 1, 2, \dots, n+1$. These matrices have a predefined Toeplitz structure given by:

$$[\mathbf{R}_o]_{rs} = \sum_{k=1}^N |W_k(\omega_k)|^{-2} \cdot e^{i2\pi(s-r)f/N_f} \quad (3.26)$$

$$[\mathbf{S}_o]_{rj} = -\sum_{k=1}^N |W_k(\omega_k)|^{-2} \mathbf{H}_o(\omega_k) \cdot e^{i2\pi(s-r)f/N_f} \quad (3.27)$$

$$[\mathbf{T}_o]_{ij} = \sum_{k=1}^N |W_k(\omega_k)|^{-2} \mathbf{H}_o^H(\omega_k) \mathbf{H}_o(\omega_k) \cdot e^{i2\pi(s-r)f/N_f} \quad (3.28)$$

If this Toeplitz structure is worked out in Matlab code, the computational effort(memory requirements and computation time) can be reduced. For minimum of this cost function all

derivatives with respect to all coefficients must be equal to zero. These derivatives are combined into one giant matrix equation, which are called the *normal equations*.

$$\mathbf{J}^H \mathbf{J} \boldsymbol{\theta} = \mathbf{0} \quad (3.29)$$

$$\mathbf{M} \boldsymbol{\theta} = \begin{bmatrix} \mathbf{R}_1 & 0 & \dots & \mathbf{S}_1 \\ 0 & \mathbf{R}_2 & \dots & \mathbf{S}_2 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_1^T & \mathbf{S}_2^T & \dots & \sum_{o=1}^{N_o} \mathbf{T}_o \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_1 \\ \vdots \\ \beta_{N_o} \\ \boldsymbol{\alpha} \end{bmatrix} = \mathbf{0}, \quad \mathbf{M} \in \mathbb{C}^{N_i(n_p+1) \times N_i(n_p+1)} \quad (3.30)$$

Reduced normal equations The number of frequencies N_f can be eliminated from the size of the system of equations by reformulating the normal equations. The β coefficients can be expressed in terms of the α coefficients:

$$\beta_o = -\mathbf{R}_o^{-1} \mathbf{S}_o \cdot \boldsymbol{\alpha} \quad (3.31)$$

This results in the following, much more compact, system; the *reduced normal equations*:

$$\sum_{o=1}^{N_o} (\mathbf{T}_o - \mathbf{S}_o^H \mathbf{R}_o^{-1} \mathbf{S}_o) \boldsymbol{\alpha} \quad (3.32)$$

The parameter redundancy can be removed by imposing the following constraint: The first coefficient of the denominator polynomials is set equal to one, e.g. $\mathcal{A}_n = \mathbf{I}_{N_i}$, for which the LS solution is given by the following expression:

$$\boldsymbol{\alpha} = \begin{bmatrix} -\mathbf{M}_{[1:n_p N_i, 1:n_p N_i]}^{-1} \mathbf{M}_{[1:n_p N_i, n_p N_i+1:(n_p+1)N_i]} \\ \mathbf{I}_{N_i} \end{bmatrix} \quad (3.33)$$

This approach, which takes into account the structure of the normal equations, is much faster than solving the system of equation 3.29 directly. Once all the denominator coefficients α are found, these coefficients are substituted back into equation 3.31 to calculate the numerator coefficients β .

3.3 Poles identification

The main purpose of this step is to obtain the poles which can be used to build the Φ matrix in equation 3.43. The amount of poles used to perform the estimation can be controlled by adjusting the size of the \mathbf{M} matrix, which is defined by the number of FRFs and the order n_p of the polynomials, as to see in equation 3.29. In order to estimate the complete set of poles, it is not necessary to estimate all FRFs. This can be relevant in case of a large FRM to reduce the size \mathbf{M} , since too large matrices cause numerical instability. A selection can be made based on the accuracy of the measured FRF and the occurrence of mode shapes. For instance, the ones close to the driving point will have low SNR and also some sensors can be placed at nodes of a mode shape and therefore not all mode shapes are present in

the corresponding FRF. Once all α coefficients are estimated by the RMFD estimator, the so-called *companion matrix* can be build, which has the following structure:

$$\mathcal{A}_c = \begin{bmatrix} \mathcal{A}'_{n_p-1} & \mathcal{A}'_{n_p-2} & \dots & \mathcal{A}'_1 & \mathcal{A}'_0 \\ \mathbf{I}_{N_i} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{N_i} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{I}_{N_i} & \mathbf{0} \end{bmatrix} \quad (3.34)$$

in which the terms $\mathcal{A}'_j = \mathcal{A}_{n_p}^{-1} \mathcal{A}_j$. This matrix can be used to determine the poles and modal participation factors of the system, respectively the eigenvalues and eigenvectors of this matrix.

$$(\mathcal{A}_c - \lambda_n \mathbf{I}) \mathcal{V}_n = \mathbf{0} \quad (3.35)$$

The poles are related to the eigenfrequencies ω_r and damping ratio ζ_r :

$$\lambda_n, \lambda_n^* = -\zeta_n \omega_n \pm i \sqrt{1 - \zeta_n^2} \omega_n \quad (3.36)$$

and this set of poles, or a selection of them, is passed on to the residue estimator. In order to discriminate the meaningful poles from the mathematical ones the so called stability diagram is often used. It is a chart in which estimated poles of a variety RMFD models is plotted. The meaningful poles will show up in each model and are therefore stable. By increasing the complexity of the RMFD model the stable poles will converge to the real poles. The stability chart will help to discriminate the meaningful poles from the entire set of estimated poles. The eigenvectors in matrix \mathcal{V}_m are related to the modal participation vector L_{Nn} , by means of the following expression:

$$\mathcal{V}_n = \begin{Bmatrix} \lambda_n^{n_p-1} L_{[:,n]} \\ \lambda_n^{n_p-2} L_{[:,n]} \\ \vdots \\ \lambda_n L_{[:,n]} \\ L_{[:,n]} \end{Bmatrix} \quad (3.37)$$

So the last n rows are repeated rest of the in matrix \mathcal{V}_n . If an entire column or row of FRFs is selected, this information can be used to estimate the mode shapes directly with correct scaling.

The modal model is based on the superposition principle and therefore the system will have to following generalized mass matrix:

$$\hat{\mathbf{m}} = \Psi_{Nn}^T \mathbf{M}_N \Psi_{Nn} \quad (3.38)$$

Let \mathbf{r}_n be the influence vector which represents the displacements of the masses resulting from static application of a unit ground displacement. The influence vector induces a rigid body motion in all modes. A coefficient vector is then equal to:

$$\mathbf{\Gamma}_n = \Psi_{Nn}^T \mathbf{M}_N \mathbf{r}_n \quad (3.39)$$

The modal participation factor matrix L_{mn} for mode n is then:

$$L_{mn} = \frac{\mathbf{\Gamma}_n}{\hat{\mathbf{m}}_n} = \frac{\Psi_{Nn}^T \mathbf{M}_N \mathbf{r}_n}{\Psi_{Nn}^T \mathbf{M}_N \Psi_{Nn}} \quad (3.40)$$

and the effective modal mass $m_{eff,n}$ for mode n is:

$$m_{eff,n} = \frac{\mathbf{\Gamma}_n^2}{\hat{\mathbf{m}}_n} \quad (3.41)$$

$$L_{mn} = \frac{\Psi_{Nn}^T \mathbf{M}_N \mathbf{r}_n}{\Psi_{Nn}^T \mathbf{M}_N \Psi_{Nn}} \quad (3.42)$$

if the mode shapes are normalized w.r.t. the mass matrix $\hat{\mathbf{m}}_n$ will be equal to one. The effective modal mass provides a method for judging the significance of a vibration mode. It also can be used to indicate if the number of modes is sufficient to describe the dynamic behaviour of the complete system. For instance, the number of selected modes should be high enough such that the total effective modal mass of the model is at least 90 % of the actual mass. Such a constraint can be used to determine the number of modes which must be taken into account.

3.4 Residue identification

The residue matrix R_{mn} is obtained in a second step by solving another LS problem in which the modal model is used. It all boils down to the following linear algebra problem:

$$\underbrace{\mathbf{e}(\omega_f)}_{\mathbf{e}_j} = \underbrace{\left[\sum_{n=1}^{n_p} \frac{\omega_f^2}{(i\omega_f - \lambda_n)(i\omega_f - \lambda_n^*)} \right]}_{\Phi} \underbrace{\begin{Bmatrix} R_{1j} \\ R_{2j} \\ \vdots \\ R_{N_p j} \end{Bmatrix}}_{\theta} - \underbrace{\hat{\mathbf{H}}_j(\omega_f)}_{\mathbf{y}} \quad (3.43)$$

which is minimized. This Right Hand Side (RHS) of this equation resembles the following matrix equation:


$$\underbrace{\begin{bmatrix} \frac{\omega_1^2}{(i\omega_1 - \lambda_1)(i\omega_1 - \lambda_1^*)} & \cdots & \frac{\omega_1^2}{(i\omega_1 - \lambda_n)(i\omega_1 - \lambda_n^*)} \\ \vdots & \ddots & \vdots \\ \frac{\omega_{N_f}^2}{(i\omega_{N_f} - \lambda_1)(i\omega_{N_f} - \lambda_1^*)} & \cdots & \frac{\omega_{N_f}^2}{(i\omega_{N_f} - \lambda_n)(i\omega_{N_f} - \lambda_n^*)} \end{bmatrix}}_{\Phi} \underbrace{\begin{Bmatrix} R_{1j} \\ R_{2j} \\ \vdots \\ R_{N_p j} \end{Bmatrix}}_{\theta} - \underbrace{\begin{Bmatrix} \hat{\mathbf{H}}_j(\omega_1) \\ \vdots \\ \hat{\mathbf{H}}_j(\omega_{N_f}) \end{Bmatrix}}_{\mathbf{y}} \quad (3.44)$$

This error is minimized and the optimal solution can be found by calculating the pseudo inverse of ϕ matrix:

$$\theta = [\Phi^T \Phi]^{-1} \Phi^T \mathbf{y} \quad (3.45)$$

Normally, the estimated residues R are subsequently decomposed into participation factors L and mode shapes Ψ . Since the modal participation factors were already estimated by the pole estimation, this information can be used to estimate the mode shapes directly:

$$\underbrace{e(\omega_f)}_{e_j} = \underbrace{\left[\sum_{r=1}^{n_p} \frac{\omega_f^2 L_r}{(i\omega_f - \lambda_r)(i\omega_f - \lambda_r^*)} \right]}_{\Phi} \underbrace{\begin{Bmatrix} \Psi_{1j} \\ \Psi_{2j} \\ \vdots \\ \Psi_{N_{pj}} \end{Bmatrix}}_{\theta} - \underbrace{\hat{H}_j(\omega_f)}_y \quad (3.46)$$

Another advantage of this pLSCF algorithm. More information can be found in the original source [4]. If the poles are estimated correctly, this residue estimator is very accurate. This can be checked by running the matlab script `CheckResidueEstimator.m` in which the estimated FRM is identical to the simulated FRM, this can be found  `SystemIdentification`

3.5 Implementation using simulated data

Now that all steps of the system identification are treated the developed algorithm is illustrated using 4 cases; First a system identification is performed using simulated data. The FRMs of the rectangular plate and the BF are simulated using the FEM of both systems and part of these matrices is used to perform the system identification. Secondly, both mechanical systems are identified using real experimental data.

3.5.1 Rectangular plate

First a simple rectangular plate is used to check the correctness of the algorithm. The system is measured at the 25 intersection points of the measurement grid. Only displacement in z -direction is considered, resulting in a 25 Degree Of Freedom (DOF) system. By adding APDL code snippets to the modal analysis tree the required nodal information of these locations is calculated by ANSYS Workbench. The entire FRM is simulated and the first column of the model will be used for system identification. This matrix is generated using the Harmonic Response module inside ANSYS Workbench. The first 23 modes were used to simulate the FRFs in ANSYS. This is sufficient, since the highest eigenfrequency in this FRI is mode 14: 1060.90 Hz. The material is a standard steel alloy, with a Young's modulus of 200 MPA and a density of 7800 Kg/m³. The geometry of the rectangular plate is made and the same (5 × 5) grid as in figure 2.4 is used. A global mass check was done; the total weight of the FEM is 3.2932 Kg, which corresponds to the measured weight of the plate, 3.297 Kg. Each gridpoint is defined as a `NamedSelection`. On these points a harmonic force of 100 N was applied in z -direction. The first 23 modes are calculated in the Modal analysis and used to simulate the ODS. For all points a FRF was calculated, the solutions were combined into one matrix which is loaded into MATLAB.

For the sake of conciseness FRI is set to [10 : 1200] Hz with 2500 frequency points and every mode has constant damping ratio of $2 \cdot 10^{-3}$. In this FRI, 14 modes are present. The

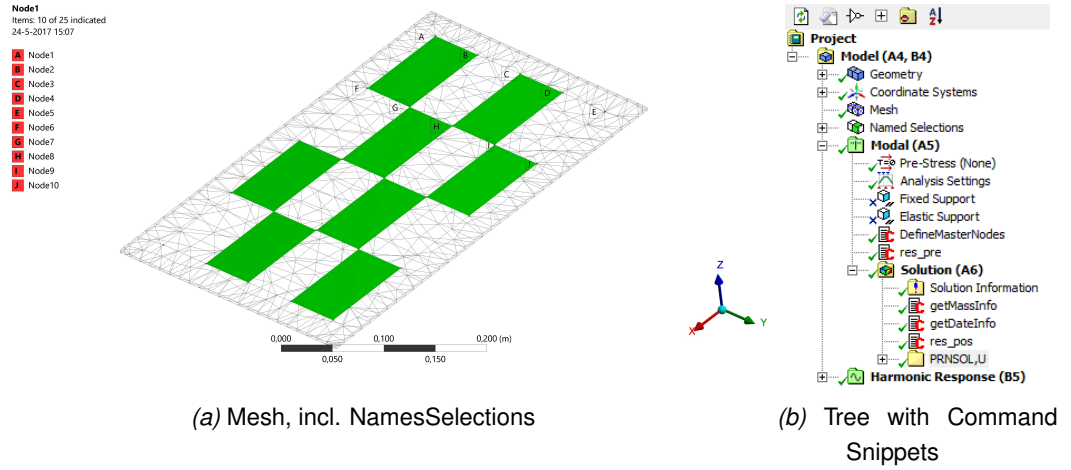
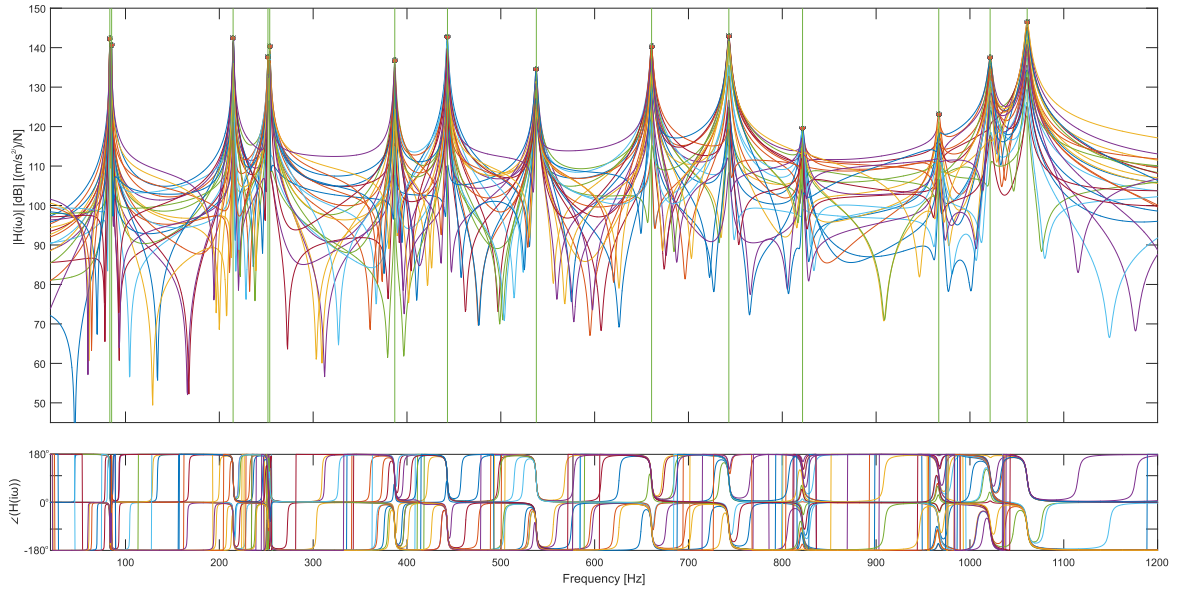


Figure 3.1: ANSYS WB 17.2: model

Figure 3.2: Simulated Frequency Response Matrix; 1st column


corresponding frequencies are listed in table 3.1 Furthermore only one column of the FRM is used for the identification. This is sufficient to determine mode shapes and adding more columns yields no further information and will only distract the reader from the essence of this chapter. Lastly, the coherence will be equal to one for the entire frequency region since the dataset is noise-free and therefore the weighting function $W_k(\omega_f)$ not applied; equal for all frequencies. With this condensed simulated dataset both algorithms are compared. All FRFs which are used for this estimation are shown in figure 3.2. Due to the low damping, the transfer functions show sharp peaks and they are free of noise, so the system identification method should be able to estimate the natural frequencies and mode shapes very accurately. The first two modes are very closely-coupled; The same holds for mode 4 and 5. The algorithm should be able to distinguish these poles to be applicable in practise. In many

Mode	Frequency[Hz]	Mode	Frequency[Hz]
1	83.16	8	537.69
2	85.04	9	660.54
3	214.88	10	742.96
4	251.94	11	821.64
5	253.88	12	966.80
6	386.91	13	1021.74
7	442.96	14	1060.89

Table 3.1: Resonance frequencies

designs some kind of symmetry is involved. Due to practical reasons, for instance manufacturability, there will never be a perfect symmetry. This results in closely-spaced peaks and it is important that the algorithm is able to handle this.

Estimated RMFD Model

The complete column is used to estimate the RMFD. The simulated FRFs are smooth functions and therefore it is relatively easy to obtain an accurate fit. The 'synthesized' polynomials are matching the 'measured' FRFs very accurately, as to see in figure 3.3; the identified poles are matching with the peaks very well while the order of the polynomials is only very low, $n_p = 2$. In order to quantify this fit equation 3.13 is calculated and divided by the amplitudes of the FRFs to make the difference relative. The average relative residual is only 0.53 %. There are also some 'numerical' poles; the ones which show no clear correspondence to a peak. For convenience, a small part of the FRM $H_{1:6,1,:}$ is shown in figure 3.3; The 'measured' FRFs and 'synthesized' FRFs are shown in the middle and bottom window respectively and the summations of both is shown in the top one. The rest of the FRM is estimated with comparable accuracy. The estimation of the RMFD using pLSCF is implemented into a Matlab function called `pLSCFestimator`, which can be found  `SystemIdentification`

The estimation process is very computationally efficient. With a CPU time of 0.86 seconds all polynomial coefficients are estimated. Also this system identification algorithm is stable and very user-friendly, since the user only has to supply the measurement FRM and the polynomial order of the estimated model. Of course, in practise the fit will be of less accuracy due to the noise present in the measured dataset, this simulated data is noise free and has low damping and for that reason the estimation is less complex. Now that the FRFs are fitted by the RMFD model, the coefficients of the α matrix can be used to construct the companion matrix and calculate the poles and corresponding modal participation factors using equation 3.35.

Pole selection The number of poles are determined by the size of the companion matrix. This is equal to the product of the order of the polynomials and the number of outputs $N_o n_p$ and therefore the number of poles will vary significantly when the order is varied for

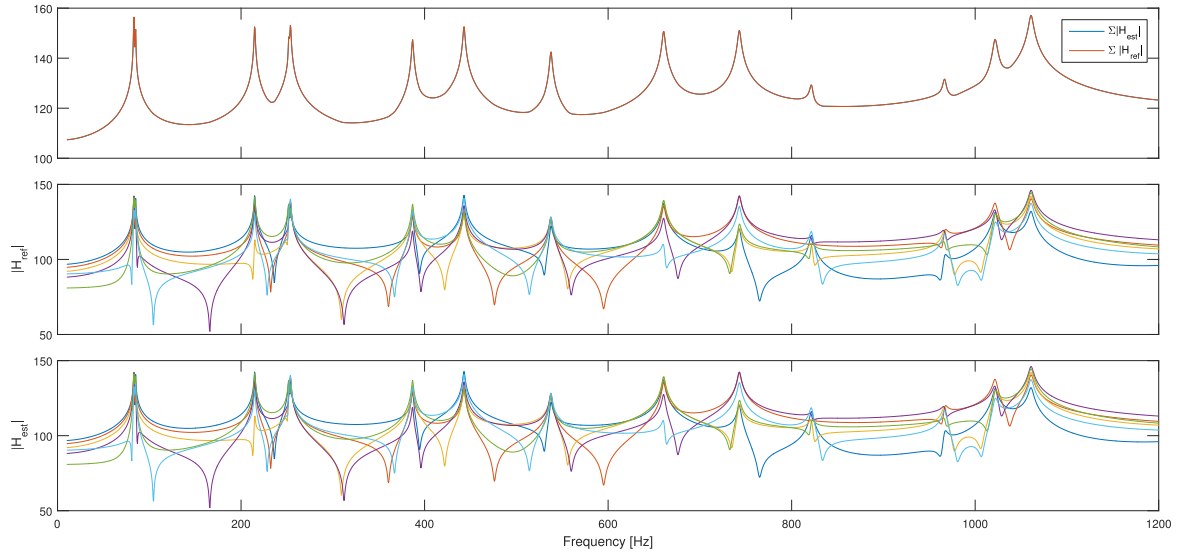


Figure 3.3: Fit of RMFD to $H_{1:6,1}$

practical FRMs. To discriminate the meaningful poles from the mathematical ones the RMFD estimation is repeated with an increasing number of poles. To increase the number of poles gently, only part of the column is fitted. First of all the majority of mathematical poles are filtered using constraints, e.g. the Right Half Plane (RHP) poles are neglected as well as the conjugates of each pole. The remaining poles of each estimation are shown in the so called stability diagram, shown in figure 3.4. In this chart the blue curve represents the cumulative of all FRFs. This clearly brings forward the resonances indicated by the red circles and vertical lines, which makes the selection of relevant poles more convenient.

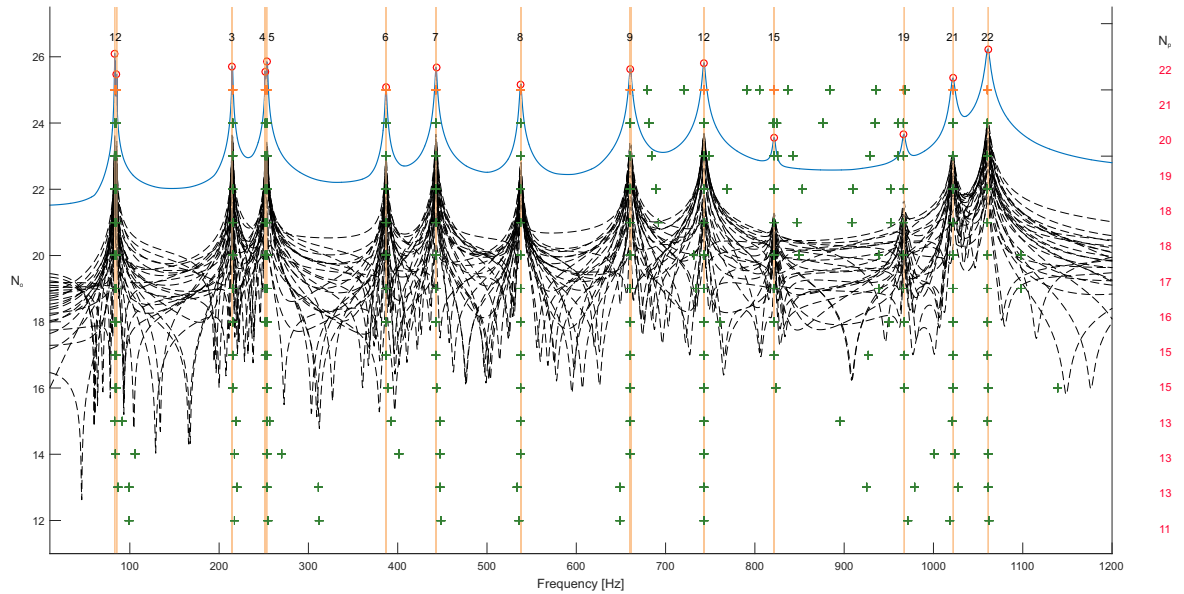


Figure 3.4: Stability diagram of simulated rectangular Plate

Some sensor locations are at of close to node of a mode shape and therefore no resonance

is observed. In order to get the most accurate estimate of all resonances, one should select only FRFs in which all resonances are measured. Furthermore, mathematical poles will differ more in value compared to the meaningful ones. This information can be used to make a good distinction. The stability diagram shows that all meaningful poles are estimated accurately when $N_o = 18$ or more. To estimate all modal participation factors the complete column is fitted, but all poles are already known from fitting only the first 18 FRFs. The selected poles are indicated by the orange markers and their indices. These poles are passed on to the residue estimator in which they are used to build the regression matrix Φ . One may wonder why this effort of applying a MDOF curve fitting technique is done when all modal parameters can also be obtained by using a simple local SDOF estimation technique like peak picking. The main reason is that these local estimates are less accurate and more time consuming for practical cases. Also for real measurements the peaks will be less obvious and then it will be quite hard to estimate the poles accurately. So for practical purposes, a global curve fitting technique like pLSCF will result in a better estimate of the poles of the system under study with less effort, since the user only has to supply a selection of the FRFs and the order of polynomials and the stability charts will indicate the mechanical poles of the system under study.

Estimated modal parameters

Once the meaningful poles are identified, they are used together with the measured FRM to estimate the residues in a LS sense, as treated in section 3.4. The result of the procedure is the modal model, shown in figure 3.5b. The Matlab function in which these steps are implemented, is called `ResidueEstimator.m`. The FRF are influenced by modes outside the FRI. This is approximated by adding upper and lower residue terms to the Φ matrix. The result is an even more accurate modal model, presented in figure 3.5a.

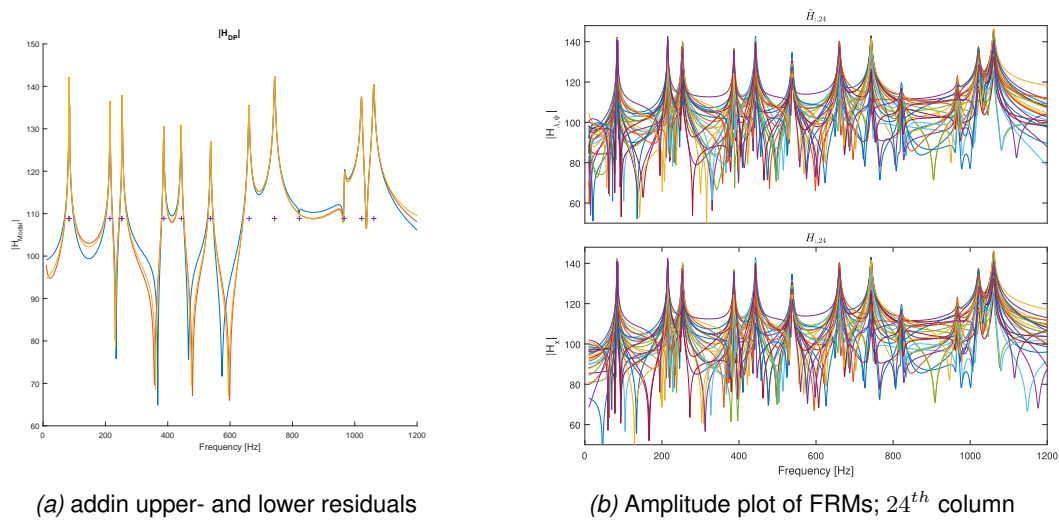


Figure 3.5: Modal Model of simulated rectangular plate

Looking to the poles in figure 3.5a one can see that the identified poles all correspond to a

peak in the FRF. The estimated resonances show the very little deviation from the analytical ones, the average deviation is only $\Delta\bar{\omega}_n = -1.52 \cdot 10^{-4} \%$. The other poles are mathematical ones and are filtered out to improve the fit. This same strategy is performed with a set of poles which still contained three mathematical then the obtained results is still accurate. The estimator gives very low residues for these mathematical poles, so they drop out. However, the best fit will always be achieved when the amount of mathematical poles is as low as possible.

Now that the residues R_{mn} are estimated, the mode shapes are determined using SVD analysis. In this analysis the residue matrix is decomposed:

$$R_{mn} = U_n S_n V_n^T = \Psi_{mn} \cdot L_{mn}^T \quad (3.47)$$

in this the matrix V is equal to unity, the diagonal matrix S describes the modal participation factors L_{mn} and the matrix U equals the mode shape Ψ_{mn} . More specifically, the matrix U is symmetric and the first row or column of this matrix described the mode shape Ψ_{mn} . This is done for all modes and the results are shown in figure 3.6 for the first 12 modes.

As to see in the equation 3.47 the mode shapes can also be estimated directly when a full column is fitted by the pLSCF. The vectors of R_m , and Ψ_{mn} are equivalent up to a scalar, so if one is only interested in the shape this decomposition is not necessary. The modal participation factor indicates the relative influence of the corresponding mode in the total displacements

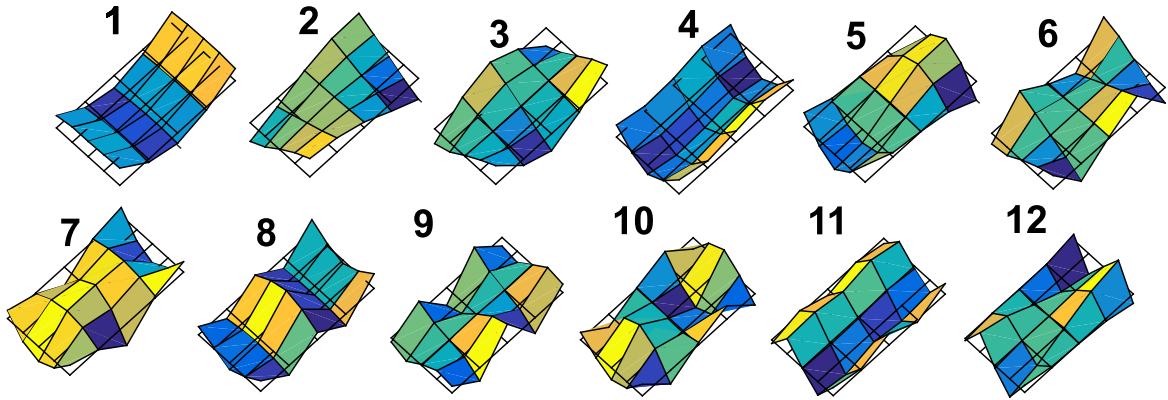


Figure 3.6: Estimates mode shapes of rectangular plate

The found shapes shown nearly perfect correspondence to the ones calculated in the modal analysis of the FEM build in ANSYS Workbench, the first 3 modes are shown in figure 3.7. The shape is already known from the residues, but the modal participation factor can be used to scale all estimates mode shapes. The amplitudes of both estimated and simulated FRMs are calculated at each resonance. The relative error between both is calculated by division of the difference by the simulated amplitudes. The average error was 0.24 % with a maximum of 1.1 % for the 6th resonance. The average error is of all other resonances is only 0.17 %. So if the measurements are noise free, the algorithm is able to consistently estimate the entire set of modal parameters with very high accuracy.

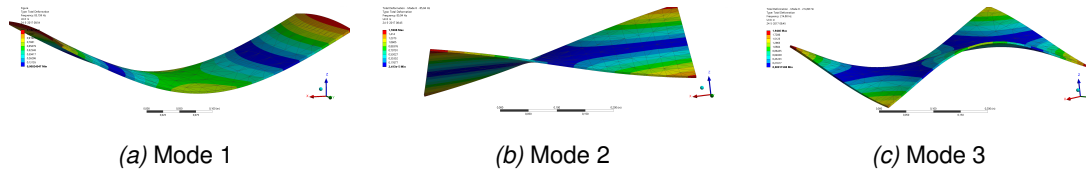


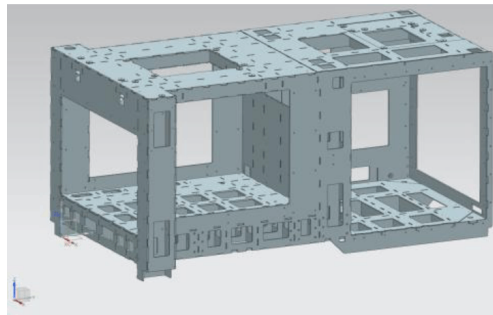
Figure 3.7: LTA 2 Bottom Frame

3.5.2 Bottom Frame

The developed system identification algorithm is applied to one of the systems which are fabricated by VDL ETG; the BF. The internal stiffness of the BF is of major importance for the dynamical performance and thus position accuracy of the LTA2. In 2013 a design verification study was performed on this particular system and the experimental data and FEM of this study is used to illustrate the practical application of the developed algorithm. For more information about the BF; the pdf file of the report of this study can be found in [VDL ETG](#).



(a) Measured system

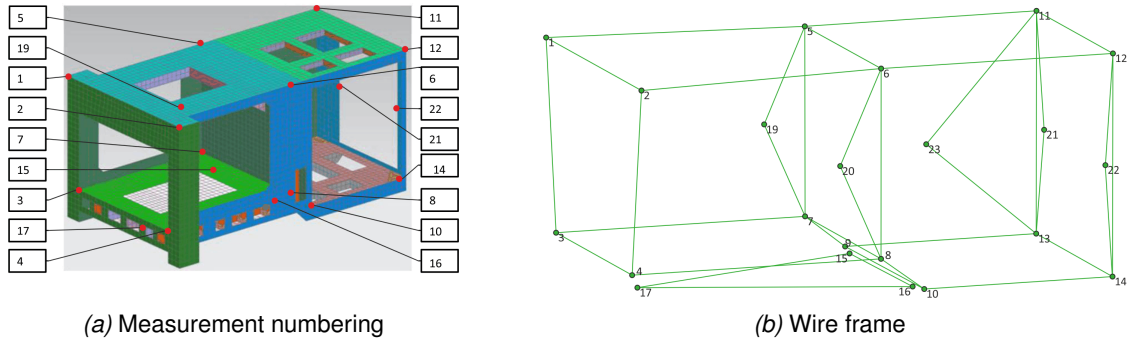


(b) NX model

Figure 3.8: LTA 2 Bottom Frame

In the FEM all 22 measurement locations are defined. To maintain consistent numbering; namely all uneven number on one side, the index 18 is absent in the system. To visualize the mode shapes a wire frame is made, shown in figure 3.9b.

First of all one entire column of the FRM is simulated using NX Nastran. More precisely, the first n analytical resonance frequencies $[\omega_n^a]$ and the corresponding nodal solutions of all 'measured' nodes $[\Psi_{mn}^a]$ are calculated by the program. Furthermore a constant damping ratio of $\zeta = 0.01$ is applied to all modes to get more realistic FRFs. The real measurements often show more damping, the peaks are less clear then in the FRM of the rectangular plate. These modal matrices are then used to calculate the state space model, using the following

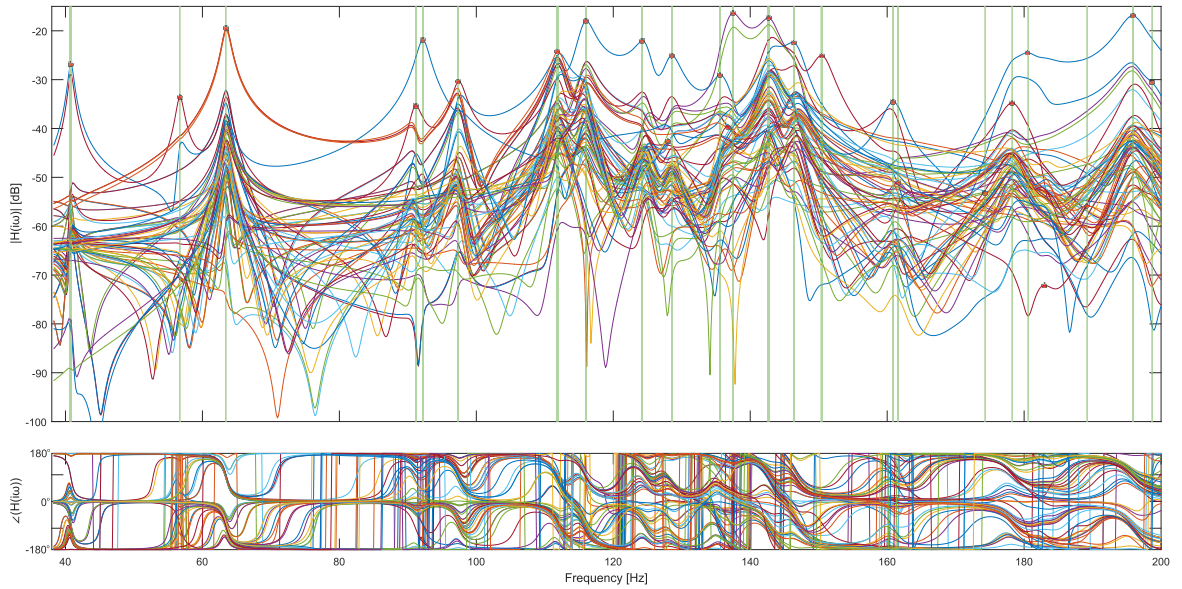
**Figure 3.9:** Measurement locations

equations:

$$A = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\Lambda_n^2 & -2\zeta\Lambda_n \end{bmatrix}, \quad C = \begin{bmatrix} [\Psi_{mn}] & \mathbf{0} \end{bmatrix} \quad (3.48)$$

$$B = \begin{bmatrix} \mathbf{0} \\ [\Psi_{mn}]^T \end{bmatrix} \quad \text{and} \quad D = \begin{bmatrix} \mathbf{0} \end{bmatrix} \quad (3.49)$$


For the simulated FRM, node 1 is chosen the excitation point. The resulting matrix H_{BF}^a contains 69×3 , inputs are impact force applied to node 1 in all three Cartesian directions, outputs are accelerations of all 22 nodes in all three Cartesian directions. The FRFs of node 17 are copied to node 18, to get a full matrix with corresponding indices.

**Figure 3.10:** Simulated Frequency Response Matrix; 1st column


This matrix is used to perform the system identification, in which the copied data is neglected. It is important to notice that if there is any copied data, the system will become singular. In practise two measurements will never be exactly the same so then this will be a non-issue. This FRM has a more realistic amount of DOFs, this involves that a selection

Mode	Frequency	Mode	Frequency	Mode	Frequency	Mode	Frequency
1	40.79	8	115.89	15	142.90	22	178.37
2	56.71	9	124.69	16	146.69	23	180.17
3	63.49	10	127.29	17	150.57	24	182.73
4	91.06	11	128.45	18	158.35	25	183.90
5	92.20	12	135.68	19	160.86	26	188.47
6	97.43	13	137.01	20	162.33	27	195.95
7	111.99	14	137.31	21	172.52	28	198.58

Table 3.2: Resonance frequencies [Hz]

of the DOFs is used for the estimation of the poles of the complete modal model. All used scripts to obtain the FRM can be found in  VDL_ETG/NXnastran.

Estimated RMFD model

The whole purpose of this curve fitting technique is to identify the poles of the system under study, which resemble the denominator terms of the modal model. The (Weighted) linear LS cost function, equation 3.20 is minimized to find the optimal set of polynomial coefficients. The α matrix is subsequently used to estimate the poles of the system. This means that it is not necessary to use all 66 FRFs for the estimation of the poles since all resonances are visible in most of the FRFs. Some mode shapes are more active in one DOF than another and for that reason fitting multiple FRFs results in better estimates. But this does not include that the entire column of the FRM is required, since one would have an enormous amount of poles to fit, namely equal to the product $n_p N_o$ and this means that no clear stability charts can be made. The best result is by fitting the RMFD on a set of FRFs in which all modes are present significantly. So the FRFs corresponding to the most active DOFs are preferred over the ones which contain one or more zeros. A selection of the entire column of the BF FRM is used to estimate the RMFD model. The error vector is summed and divided by the sum of the FRFs, resulting in a relative error of $3.52 \cdot 10^{-9}$. The fit could be improved even more, when the mathematical poles are filtered out. For convenience, a small part of the FRMs $H_{1:6,1,:}$ is shown in figure 3.11. The 'measured' FRFs and 'synthesized' FRFs are shown in the middle and bottom window respectively and the summation of both is shown in the top one. The rest of the FRM is estimated with comparable accuracy. The estimation of the RMFD using pLSCF is implemented into the Matlab code pLSCF.m shown  VDL_ETG/SystemIdentification/pLSCF.

The estimation process is very computationally efficient. With a CPU time of 0.25 seconds all polynomial coefficients are estimated, from which all natural frequencies and mode shapes can be calculated. Also this system identification algorithm is stable and very user-friendly, since the user only has to supply a selection of the simulated FRM and the polynomial order of the estimated model. The selection contains the 27 most active DOFs; this was determined by summing the frequency content of each FRF.

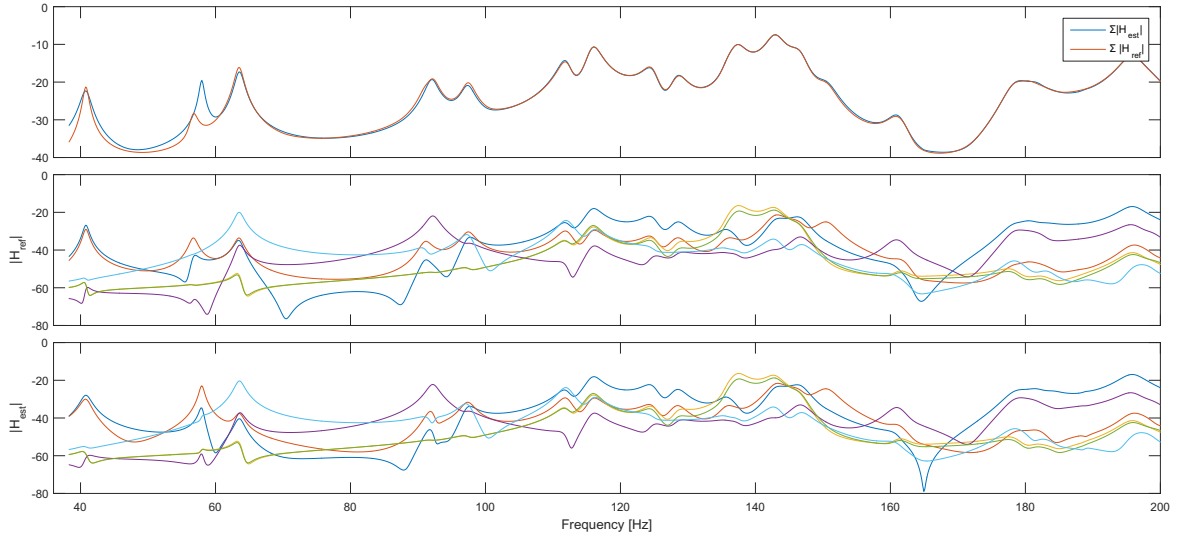


Figure 3.11: Fit of RMFD to $H_{1:6,1}$

Pole selection The poles are selected using the stability chart shown in figure 3.12. The order of the polynomials is set to $n_p = 2$ and in total 27 FRFs are fitted. The second resonance, at approximately 57 Hz is the only resonance which is not represented in the fitted modal model. The number of FRFs was increased until the resonance was identified, but then the other estimates were estimated with less accuracy. This resonance is a so called local mode, it only involves a small part of the DOFs, namely DOF 22 and 42. Therefore it is not part of the common denominator, the pole is not inactive in all other FRFs. All other resonances are identified by a stable pole, since they are global mode shapes. The selected poles are indicated by the orange markers.

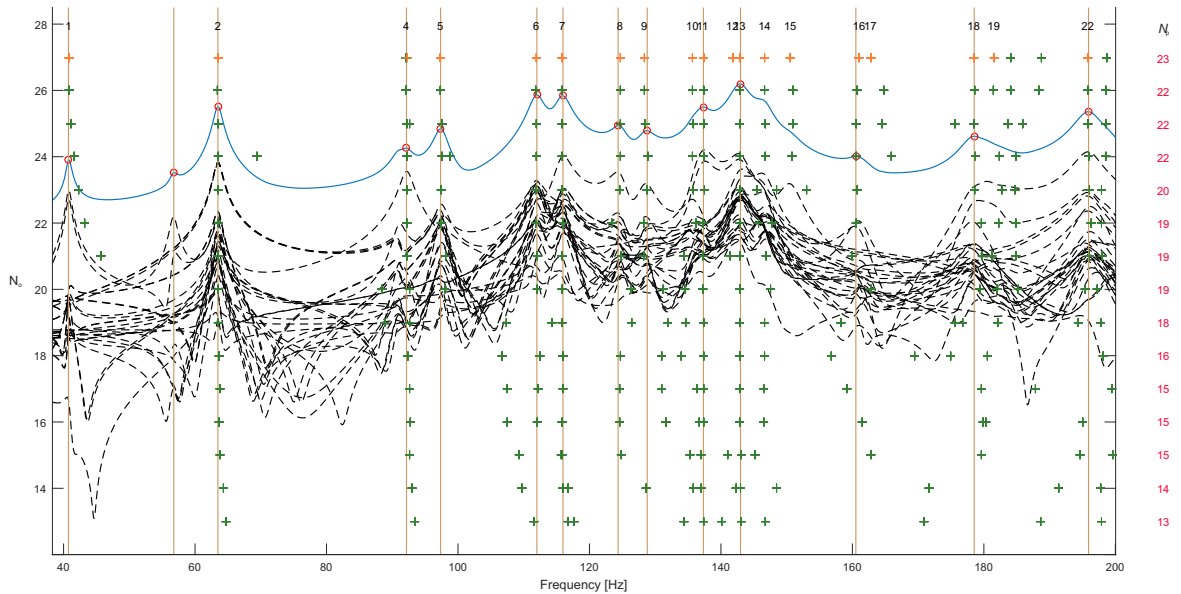


Figure 3.12: Stability diagram of simulated BF

Increasing the amount of FRFs does not result in better estimates, since the system will

overloaded by the amount of mechanical poles. The number of poles used for the estimation is limited. This depends on the complexity of the FRFs. E.g., for a FRF which shows 10 resonances approximately 10 complex pole pairs can be identified. Therefore the number of poles should be approximately 20 to 30 to get a good fit. If the number of poles of the RMFD is increased significantly further the α matrix will become close to singular which results in an inaccurate fit. The system has too many unknowns to solve the set of equations accurately. This can also be seen when looking to the condition number, which is the ratio between the highest and lowest eigenvalue. A high condition number will result in a

Estimated modal parameters

The set of selected poles is entered into the mode shape estimator, which is able to estimate the complete FRM. So the fact that the RMFD model estimator is not able to estimate the complete FRM is not an issue, since the whole purpose of this step, the distillation of poles, can also be performed based on a subset of the FRM. The estimated modal matrices of this model are indicated by $\tilde{\Psi}_{mn}^a$ and $\tilde{\Lambda}_n^e$

First the estimated mode shapes $\tilde{\Psi}_{mn}^e$ are compared to the FEM mode shapes Ψ_{Nn}^a graph-

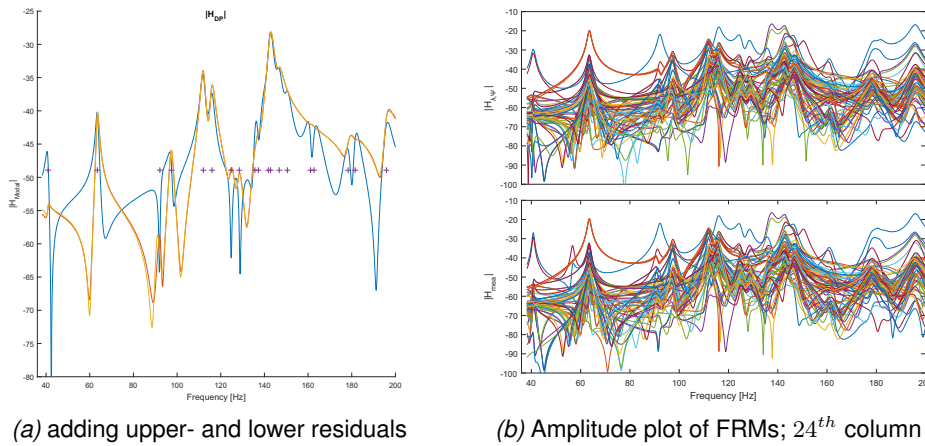


Figure 3.13: Modal Model of simulated BF

ically. Visual information is very helpful to obtain insight in the structural dynamics of a system, it tells way more than numerical results. As to see in figure 3.14 the wire frame gives a clear indication of the FEM mode shape. The wire frame visualisation of the measured mode shape can be used to interpret resonances, which can be very helpful to gain insight in the dynamic behaviour of the system under study. For instance, looking at the mode shape shown in figure 3.14 one can clearly see that this resonance is mainly determined by bending at one of the corners of the frame in vertical direction. Furthermore a script was developed to produce short animations of the mode shapes. To quantify the correspondence between estimated and analytical models correlation analysis is performed and all results will be treated in Chapter 4.

The estimation using the simulated FRM shows promising results, it proves that if the reso-

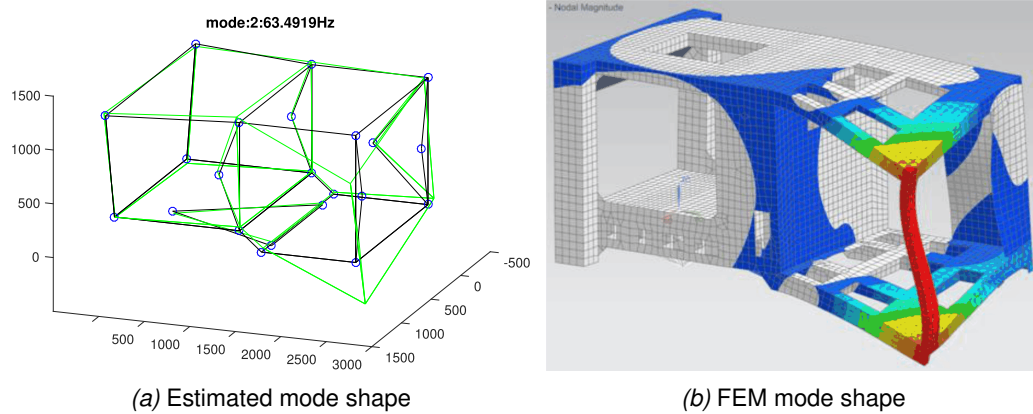


Figure 3.14: mode 2: 63.5 Hz

nances are significantly present in the measurements, the developed algorithm will distillate the modal parameters accurately. The amplitudes both estimated and simulated FRMs are calculated at each resonance. The relative error between both is calculated by division of the difference by the simulated amplitudes. The average error was 2.1 % with a maximum of 9.9 % for the first pole. The first and third pole are estimated relatively bad, the mean error of all other resonances is only 1.2 %. So if the measurements are noise free, the algorithm is able to consistently estimate a large DOF system with very high accuracy.

3.6 Application to an experimental structure

Subsequently, the real measurements on both systems are used. The presence of noise results in a FRM which is harder to fit by the modal model. However, in practise measured FRFs will always be contaminated with noise and not all modes will be excited due to compliance of the structure. The developed algorithm must be able to cope with these imperfections and estimate the majority of the modal parameters with sufficient accuracy. The coming sections will describe the performance of the algorithm in the specific cases.

3.6.1 rectangular Plate

Now that the simulated FRMs have convincingly proved that the algorithm is able to identify a complete modal model of a mechanical system using a full column of noiseless data, the same procedure is applied to real experimental data. The measured FRM, which was the result of the data acquisition phase, shown in figure 2.5 is estimated using the same pLSCF method, described in section 3.2.2. The low frequent suspension and inaccuracy of the accelerometers at low frequencies are filtered out by increasing the starting frequency of FRI to 36 Hz. Still substantial below the first resonance of the system, 81 Hz respectively. With this small adjustment the identification process is performed similar as in the simulated case.

Estimated RMFD model

In this FRM the resonance peaks are still narrow which makes relatively easy to identify. However, these FRFs contain noise and therefore it is harder for the algorithm to determine all the mechanical poles accurately. This results relatively higher amount of mathematical poles are estimated, therefore the polynomial order is increased, $n_p = 6$. Due to the presence of noise the FRFs are less smooth, as to see in middle window of figure 3.15. Also the low frequency peaks are estimated less accurate since they show too much damping. Since the FRFs are very noisy at these low frequencies and therefore the measured signal will not be representing the real dynamic behaviour of the system under study. The lower residual term is not able to represent this behaviour fully. This effect is reduced by increasing the lower frequency limit slightly. The correspondence between the rest of the synthesized and measured FRFs is still very accurate, as to see in the top window.

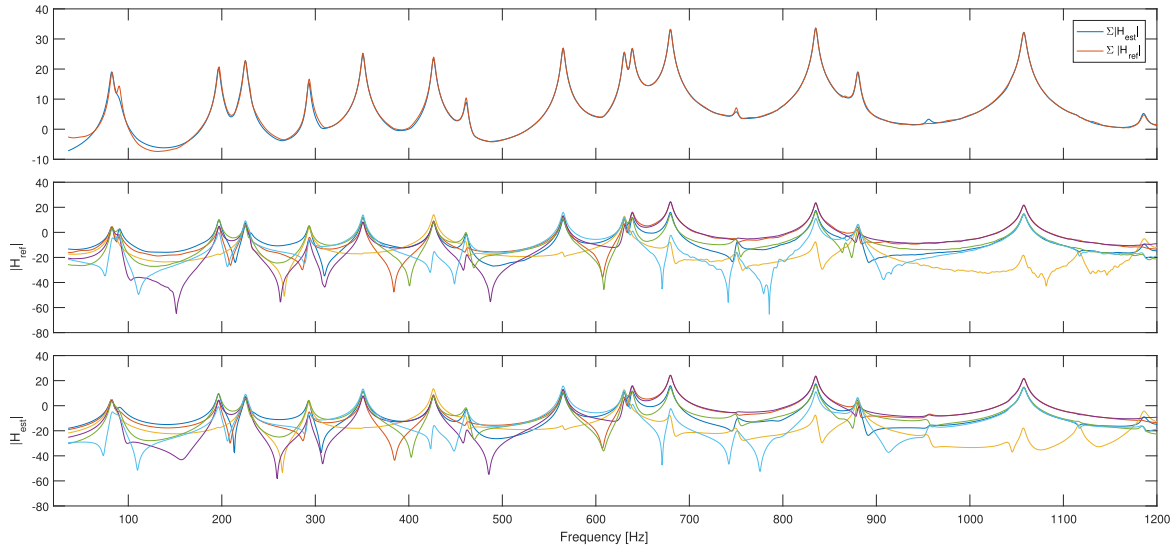


Figure 3.15: Fit of RMFD model to the measurements

The most important objective, the estimation of the peaks, is still very well succeeded. In between the resonances some deviations show up, although their contribution is very little realizing that they are plotted in decibel scale. The error vector is summed and divided by the sum of the FRFs, resulting in a relative error of $9.78 \cdot 10^{-4}$, still very close to zero.

Pole selection This RMFD model is estimated multiple times in which part of the measured FRFs is fitted. The procedure started by fitting the first 17 FRFs and is continued until the full column was fitted. The stability chart of these results are shown in figure 3.16. Again a selection of the poles of the RMFD model which estimates the full column are passed on to the residue estimator. The selected poles are indicated by the orange markers. The selection is based on the peaks of the blue curve and the stability of each pole throughout the different estimations.

As the second mode clearly contained too much damping, the real part of this poles is adjusted to obtain a better fit. This is done since there was just one pole which showed

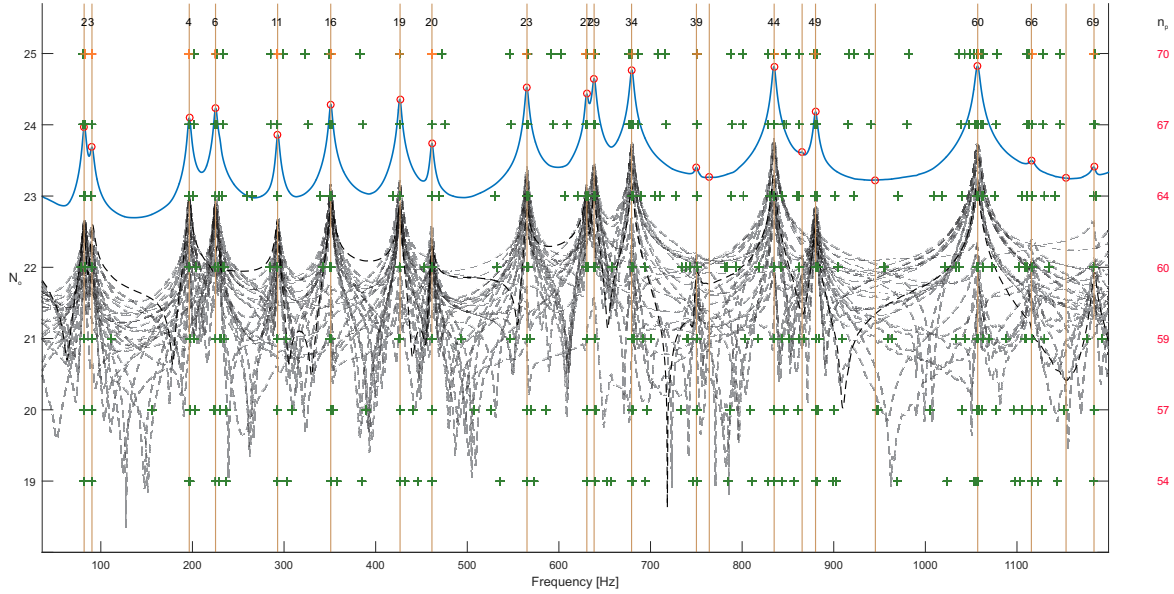


Figure 3.16: Stability diagram of measured rectangular Plate

significant discrepancy. If multiple poles are estimated badly one must compare multiple sets of selected poles and use this to improve the estimation iteratively. More investigation about this iterative process is required to develop a more convenient procedure to adjust the damping of the poles. For this case the fit accuracy is sufficient to proceed with the next step of the system identification process.

Estimated modal model

The identified poles are passed on to the residue estimator. The driving point FRF is shown in figure 3.17a and it shows that the majority of the peaks are estimated accurately. The addition of the lower- and upperresiduals improves the fit even more. As to see in figure 3.17b the complete set of FRFs is estimated accurately at the resonances, in between some discrepancies are observed. The modal parameters are consistently estimated and these modal matrices can be used for further analysis, e.g. for validation purposes.

For convenience, the absolute values of both measured and estimated FRMs are compared. A matrix H_{comp} is build in which the amplitude vectors of both datasets are stacked in alternating order. The red and green corners indicate the measured and estimated values respectively.

The average difference between measured and estimated FRMs is 5.4 % with a maximum of 22.2 % for the 17th pole. The 14th pole is estimated relatively bad, the mean error of all other resonances is only 4.4 %. This difference is not entirely caused by the estimator error since also noise is inside the system which cannot be described by the modal model.

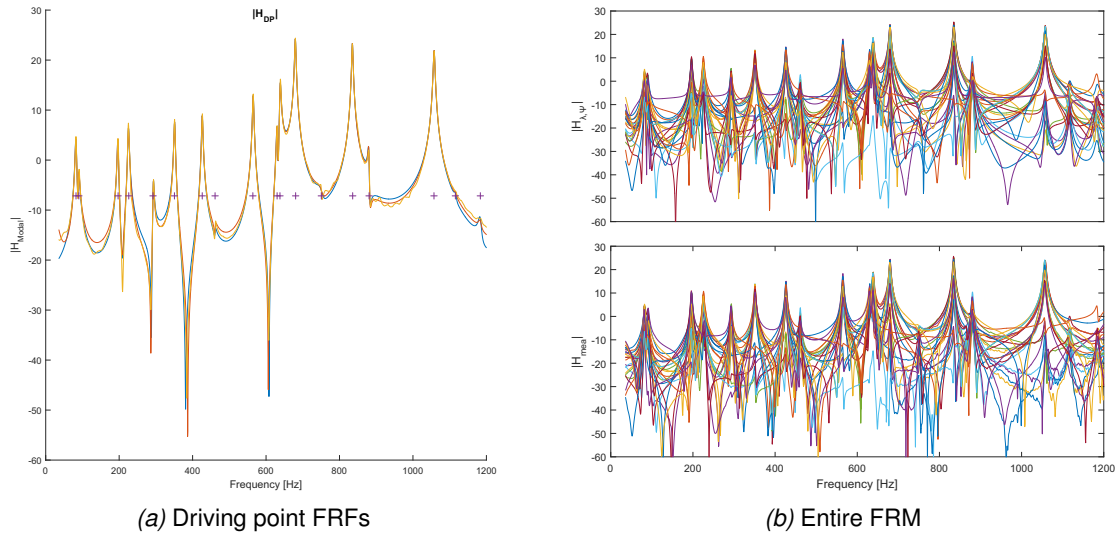


Figure 3.17: Fit estimated FRFs

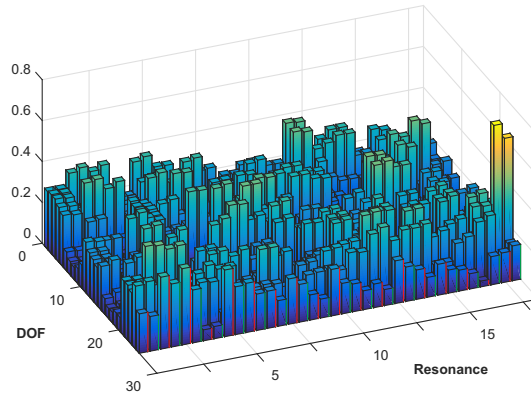



Figure 3.18: Comparison FRMs at resonances

3.6.2 Bottom frame

In practise a system will be made out of many parts from different materials and also part of the system can be complaint such that not all parts of the system are excited. This all will result in a more complex FRM. Part of these effects can be recognized in the measurements of the experiments executed on the BF. The peaks are less clear compared to former cases and also some modes are only present is a small selection of the DOFs. This dataset is very illustrative of any mechanical system fabricated at VDL ETG.

Vibration experiments are performed on one of the fabricated BFs. The system was excited in x and z direction. In this case only the x direction is considered, so only the first 22 measurement DOFs is measured. part of the FRM is measured, namely $H_{BF}^x([1 : 17, 19 : 23], 1)$ and $H_{BF}^x(46 + [1 : 17, 19 : 23], 47)$. This matrix is used to perform the system identification, in which the copied data is neglected. All used scripts can be found in  VDL_ETG/NXnastran. This is the final test case for the identification algorithm, using real measurement data measured on one of the products of VDL ETG. Only part of the outputs was measured, due to

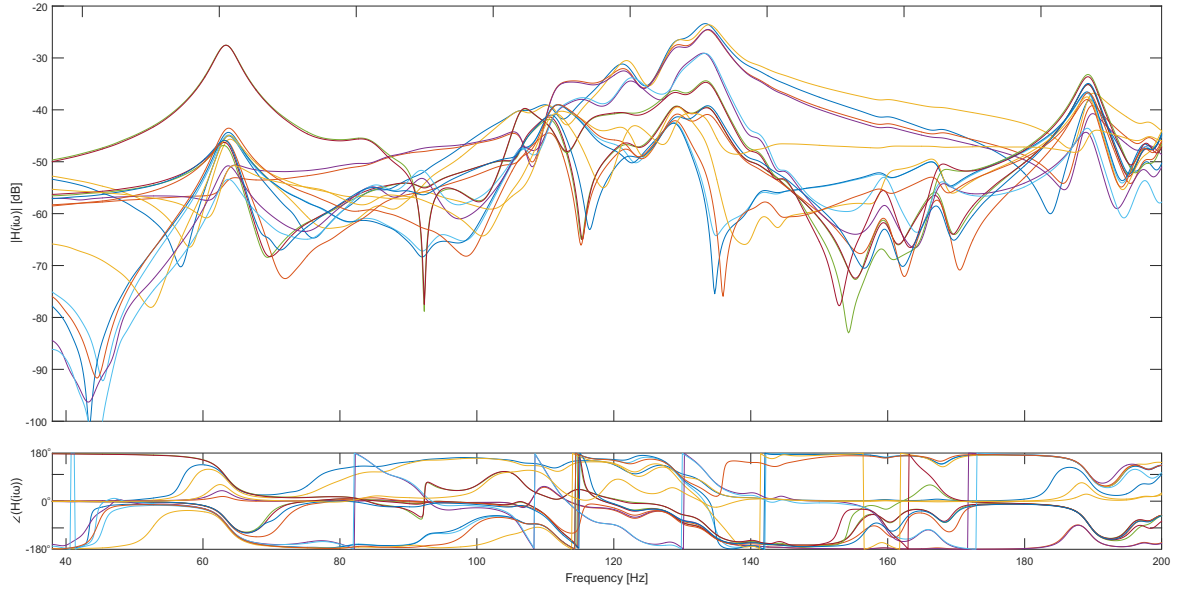


Figure 3.19: Measured Frequency Response Matrix; $\tilde{H}_{1:23,1}$

lack of time. The measurements show good correspondence to the FEM, as described in the report of the design verification, present in folder `VDL_ETG/NXnastran`. Corresponding to a measurement in which the system is excited in x direction and all 23 nodes accelerations in x -direction were measured. The second measurement involved excitation at node 1 and the accelerations of first 17 nodes were measured, all in z -direction. So two incomplete columns are measured, this dataset is used to illustrate a practical implementation of the developed algorithm.

Estimated RMFD model

The FRFs are estimated the pLSCF estimator which results in a RMFD model which can be used to identify the poles of the system and subsequently the residue estimator is used to calculate the corresponding mode shapes. Since the FRM H_{BF}^a contains many FRFs, only of the first part of the estimated FRFs is shown in figure 3.21, but the rest of the estimated FRFs show similar correspondence. All used scripts can be found in `SystemIdentification`. The estimated FRFs is still very accurate, as to see in the top window.

Pole identification The poles are less clear, as to see in figure 3.21.

In order to get such a good fit the user has to consider the size of the FRM to be estimated. All available measured FRFs are used and the polynomial order n_p was determined after couple of iterations. Increasing the order any further gives no improvement of the fit. It is hard to identify the origin of this issue exactly, since it involves the complexity of the FRM, the number of frequencies n_f and the order of the polynomials n_p . More research about this

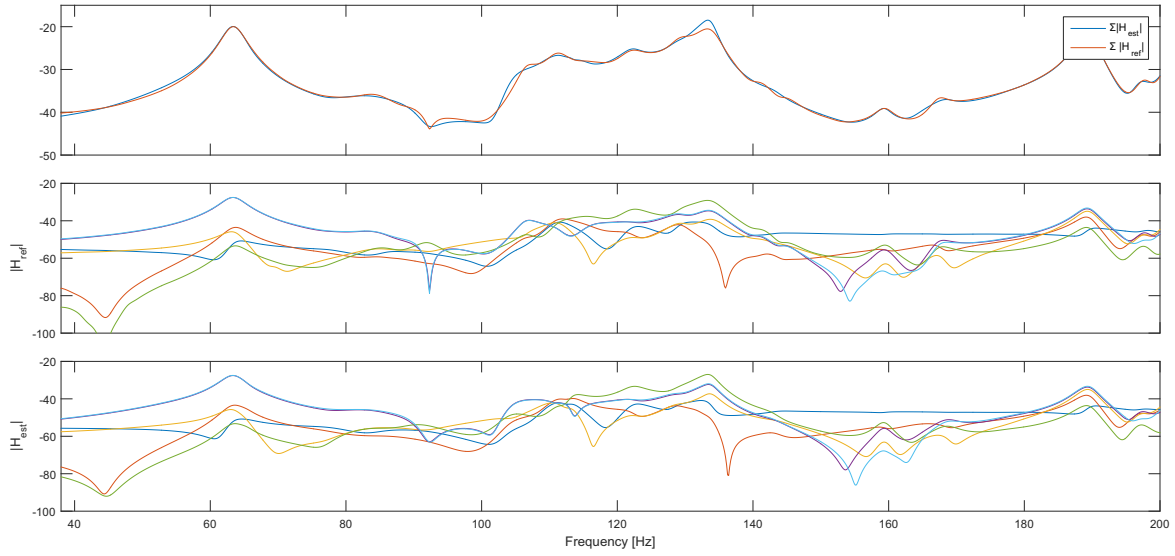


Figure 3.20: Fit of RMFD model to the BF measurements

issue would be required, but this is outside the scope of this study. As to see in figure 3.21 there are modes which only have little influence, so they are hard to identify. For that reason also poles which are not close to a peak are selected.

Estimated modal model

Both measurements are fitted separately since different input directions are used. In figure 3.22a the driving points FRFs of these models are plotted together with analytical FRF.

The estimated modal matrices of this model are indicated by Ψ_{mn}^x and Λ_n^x in which the superscript x refers to *experimental*. The first 4 dominant resonances corresponds to the simulated model very well. Since the measurements are only in one direction, the found mode shape also only describe one DOF of the nodes. However, the knowledge of the FEM can be used to expand the measured DOFs to all DOFs of the wireframe, this will be treated in section 4.3.1.

Estimated modal model

The set of selected poles and the entire column of the measured FRM passed on to the residue estimator to build the regression matrix Φ and the RHS y . As to see in figure 3.17a the addition of the low and high frequent residual terms significantly increases the fit of modal model and therefore the estimated residuals will be more accurate. Also one should note that shown FRFs are plotted in decibels. Looking at the complete estimated FRM in figure 3.17b one can see that this is achieved for all FRFs with comparable accuracy. In between peaks differences are obtained but close to the resonances all amplitudes of the FRFs are estimated correctly within an average error margin of 4.6 % and a maximal error of 9.1 %, provided that the two worst estimated resonances are left out. In this case the 5th and the 13th resonances contain a too much damping, the real part is estimated too large.

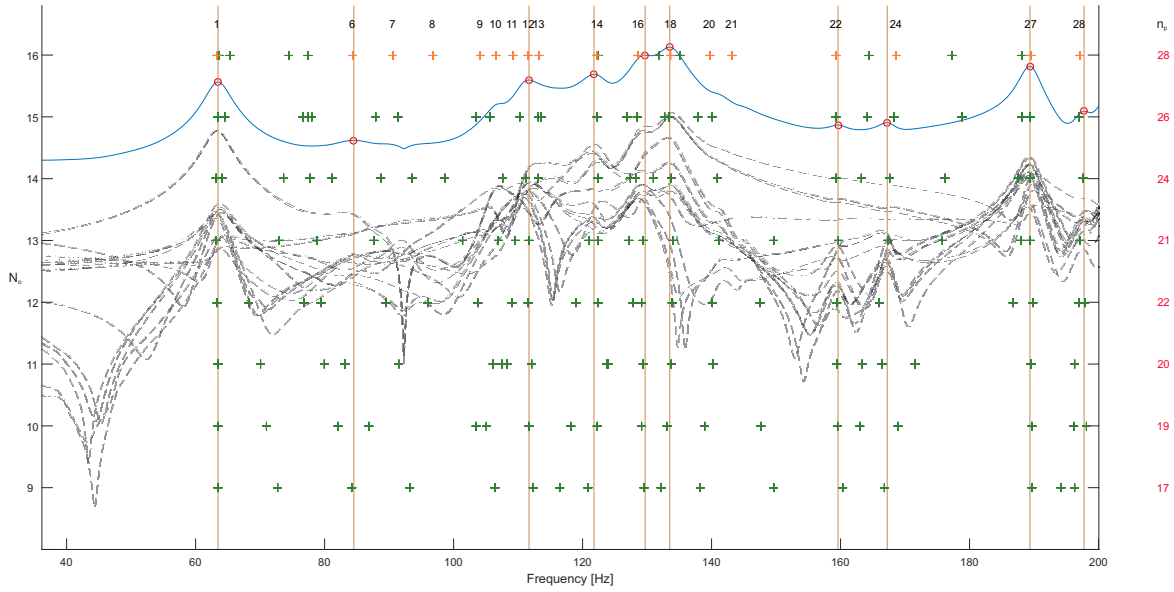
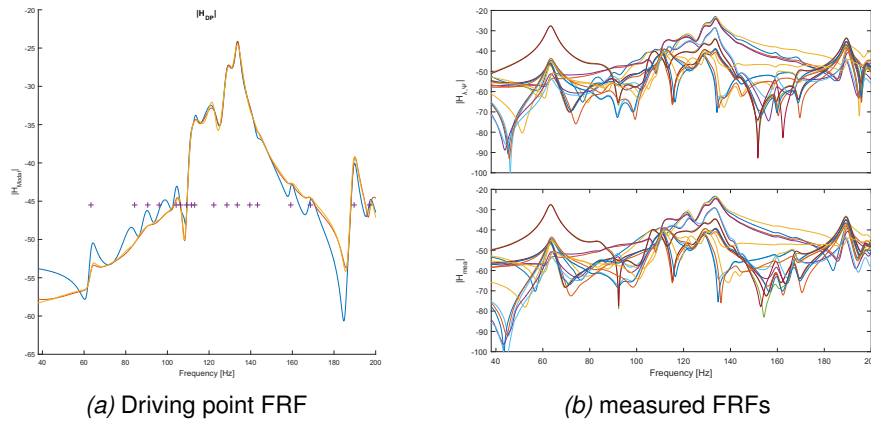


Figure 3.21: Stability diagram of BF

Figure 3.22: estimated $\tilde{H}_{BF}^x([1 : 23], 1)$

An attempt was made to adjust these two poles iteratively, but no consistent improvement was found. More research to this issue should be done, for this study the error is accepted. The corresponding estimated mode shapes are inaccurate and should not be used in further analysis. The majority of the poles is estimated with high accuracy and this can also be seen when comparing the both FRMs in figure 3.17 close to these frequencies; approximately 92 and 157 Hz. At those frequencies the amplitudes differ the most, for the rest of the resonances a very satisfactory fit is found.

For convenience, the absolute values of both FRMs for each resonance frequency are plotted in the three-dimensional bar graph in figure 3.23. The red and green lines indicate the synthesized and measured FRM respectively. All vectors in this figure are normalized to get a comparable scale. From this one can conclude that at most resonances the measured response is accurately described by the estimated modal model, which is the main objective

of performing an experimental analysis.

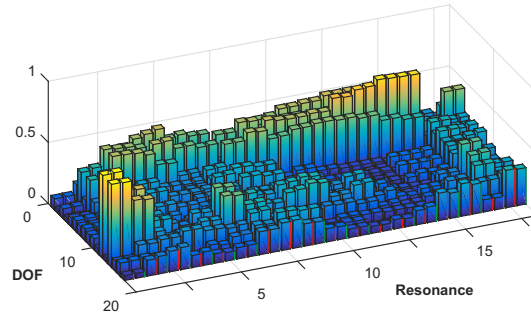


Figure 3.23: Comparison FRMs at resonances

The average difference between measured and estimated FRMs is 5.4 % with a maximum of 22.2 % for the 17th pole. The 14th pole is estimated relatively bad, the mean error of all other resonances is only 4.4 %. This difference is not entirely caused by the estimator error since also noise is inside the system.

3.7 Conclusion

The system identification procedure is now treated fully. Both systems, the rectangular plate and the BF, using simulated and measured FRMs are estimated and the developed algorithm is able to identify the poles and subsequently the mode shapes in short CPU time and with minimal user input. The developed algorithm is user-friendly, since the only input asked from the user is to select the FRI and the order n_p of the polynomials. One has to note that the RMFD estimator, more specifically the matrix M , can run into singularity problems if the number of poles is too large. The amount of poles can be increased gradually by selecting a subset of the measured FRFs. On this subset a RMFD model is fitted and this model is then used to calculate the poles. By gradually increasing the amount of selected FRFs multiple sets of poles are found and presented in the so called stability diagram. This diagram can then be used to identify all mechanical poles of the system. This procedure works best if the FRFs corresponding to the most active DOF are selected.

Once the poles are identified they are passed on to the residue estimator in which they are used to build the regression matrix ϕ . This estimator is very stable and therefore the complete FRM is fitted by the modal model. Once all modal parameters are estimated this information is saved as a Matlab data file `.mat` which can be used for the correlation analysis, treated in Chapter 4.

System Correlation

This chapter contains the correlation analysis, which is the final phase of the Experimental Modal Analysis. The modal model which is identified system will be compared with an analytical dataset and the correlation analysis is done to identify if and where any deviations occur. This insight can be used to improve the FEM or to validate the system with a reference model to verify that the system meets its specifications. Each of the tools presented below can help the engineer to gain insight in the cause of the discrepancies that exist between FRMs.

4.1 Correlation metrics

Every estimated set of modal parameters can be compared the reference dataset. Therefore no knowledge about the system matrices is required and it can be a first step to investigate correspondence between models. More details about these metrics can be found in the original source [5]. All metrics will be illustrated by using the simulated and estimated data which were presented in section 3.5.1. All used metrics are also calculated for the BF and these results are shown in appendix B and used scripts can be found in `SystemCorrelation`. This comparison can be done by comparing the resonance frequencies, the mode shapes and by using entire FRFs.

The Frequency Response Assurance Criterion (FRAC) [6] The use of FRFs for correlation instead of eigenvectors is attractive because experimental FRFs are measured directly, therefore they are more easily obtainable than eigenvectors. The eigenvectors require significant post measurement analysis of the FRM(system identification). The FRAC is defined by the following expression:

$$\text{FRAC}_{io} = \frac{\left| \{H_{io}^a(\omega_i)\}^T \{H_{io}^x(\omega_i)\}^* \right|^2}{\left(\{H_{io}^a(\omega_i)\}^T \{H_{io}^a(\omega_i)\}^* \right) \left(\{H_{io}^x(\omega_i)\}^T \{H_{io}^x(\omega_i)\}^* \right)} \quad (4.1)$$

in this $H_{io}^a(\omega_i)$ and $H_{io}^x(\omega_i)$ refer to the FRF corresponding to input DOF i and output DOF o of FRM a and x respectively; In practice most of times a will be the 'analytical' dataset and

x will be the 'experimental' dataset. The FRFs can be compared over the full FRI, a part of this range or only at the resonance frequencies. For this case the full FRI is used. The resulting matrix indicates which DOFs are correlated. The matrix is evaluated for using the simulated and estimated FRMs, shown in figure 4.1a. All maximums are at the diagonal, indicating that all DOFs are corresponding. All diagonal terms are very close to one, which indicates that the estimated FRFs are estimated with high accuracy. The higher the value of the diagonal terms, the more both FRFs correspond.

The Modal Assurance Criterion (MAC) The modal parameters of the estimated modal model can also be compared. MAC shows the degree of correlation between two vectors and it is computed by the following expression:

$$MAC_{pq} = \frac{\left| \{\Psi_p^a\}^T \{\Psi_q^{e*}\} \right|^2}{\left(\{\Psi_p^a\}^T \{\Psi_p^{a*}\} \right) \left(\{\Psi_q^e\}^T \{\Psi_q^{e*}\} \right)} \quad (4.2)$$

in which Ψ_p^{a*} and Ψ_q^{e*} are the p^{th} and q^{th} column of the analytical and experimental mode shape matrices respectively. The resulting matrix indicates which modes are correlated and therefore identifies the mode pairs of in the two datasets. The matrix is evaluated for using the simulated and estimated mode shape matrices, shown in figure 4.1b. The maximums are at the diagonal terms, which indicates that all mode shapes correspond. Since mode shapes are orthogonal, the off-diagonal terms should be very low. The more this matrix approaches the unity matrix, that better the mode shapes are estimated.

The Modal Scale Factor (MSF) Once all modes are paired, their sizes and orientations can be scaled using the MSF. It is calculated using the following equation:

$$MSF_q = \frac{\{\Psi_q^a\}^T \{\Psi_q^{e*}\}}{\{\Psi_q^a\}^T \{\Psi_q^{a*}\}} \quad (4.3)$$

In which Ψ_q is the q^{th} mode pair.

By carefully applying these three checks both datasets are equally oriented in terms of DOFs, mode shapes and directions. The matrices shown in figure 4.1 show high values at the diagonal, indicating the estimated model is very well correlated to the simulated model.

The COordinate Modal Assurance Criterion (COMAC) Now that both modal matrices have the correct scale and orientation the COMAC can be calculated to indicate the correspondence between the mode shapes of two datasets in terms of the DOFs. This is done by the following expression:

$$COMAC_q = \frac{\sum_{r=1}^{N_m} |\Psi_{qr}^a \Psi_{qr}^{e*}|^2}{\sum_{r=1}^{N_m} \Psi_{qr}^a \Psi_{qr}^{a*} \sum_{r=1}^{N_m} \Psi_{qr}^e \Psi_{qr}^{e*}} \quad (4.4)$$

in which Ψ_{qr}^a and Ψ_{qr}^e are the q^{th} DOF of the r^{th} mode shape. The expression is very compatible to the MAC, only this time the mode shapes are summed for each DOF. This indicates

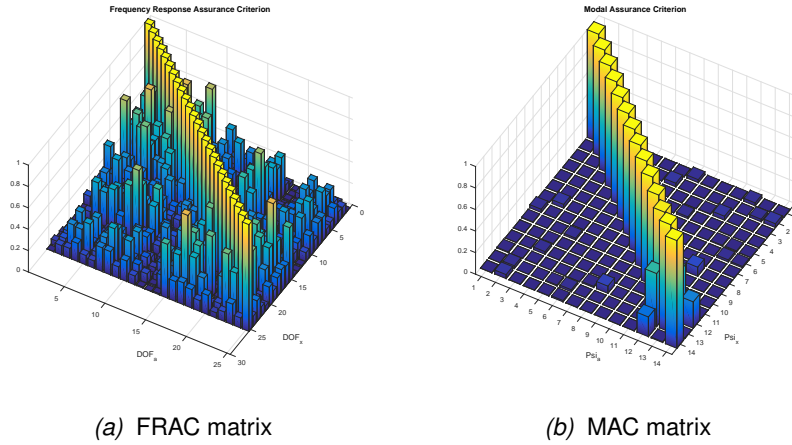


Figure 4.1: FRAC and MAC matrices evaluated for estimation case 1

which DOF is deviating the most, which can give the engineer insight in the origin of the deviation without need to calculate the structural matrices of the system. For this case the following result is obtained:

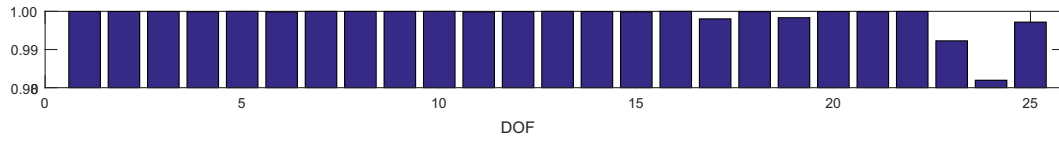


Figure 4.2: COMAC values for estimation case 1

From which one can conclude the estimation is almost perfect, the worst estimation is involved with mainly DOF 23 and 24 is less than 2%, so all DOFs are corresponding accurately.

The Enhanced COordinate Modal Assurance Criterion (ECOMAC) In practise experimental modal vectors imply the potential problem of calibration scaling errors and/or sensor orientation mistakes. In this metric the nominator part is equal to the COMAC. The terms in the denominator part of the COMAC normalize the correlation coefficient and therefore every error in scaling or orientation will not be recognized. The ECOMAC was developed to extend the COMAC computation to be more aware of typical experimental errors that occur in defining modal vectors such as sensor scaling mistakes and sensor orientation errors.

$$ECOMAC_q = \frac{\sum_{r=1}^{N_m} |\Psi_{qr}^a \Psi_{qr}^e|^2}{2N_m} \quad (4.5)$$

The result is shown in figure 4.3a, this also indicates that the last three DOFs are deviation the most. So this metric can be used to quickly check if the COMAC is evaluated correctly.

The absolute difference between mode shapes Finally, absolute differences between both mode shape matrices are calculated. The resulting matrix is shown in figure 4.3b and

clearly illustrates that the 10th mode shape is estimated the worst and that primarily last three DOFs are involved in this difference. From which one can conclude the estimation is almost

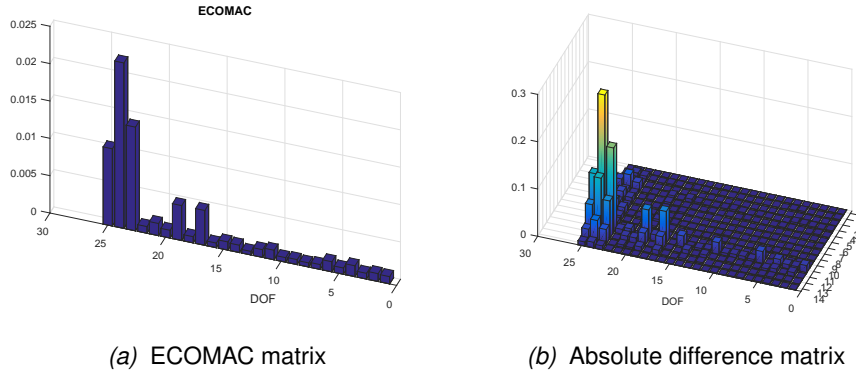


Figure 4.3: Correlation matrices evaluated for estimation case 1

perfect, the worst estimation is involved with mainly DOF 23 and 24.

4.1.1 Conclusion

The correlation metrics indicate that the estimated modal model very accurately describes the dynamic behaviour of the system under study. These metrics can also be used to show if and where in the system deviations occur. There is a variety of these assurance criteria and by even weighting can be applied to focus on the most relevant frequencies. All these metrics can help the engineer to identify any deviations between systems and this information in combination with a decent engineering judgement can be used to draw conclusions about the cause of these deviations. Furthermore in the case for a design validation study of a fabricated product, these metrics can be used to quantify how well the fabricated product meets its specifications.

4.2 Obtaining the analytical modal model

ANSYS is used to calculate full space matrices, including the global stiffness and mass matrix of the system under study. In appendix C a manual is written in which the entire procedure is explained. These matrices can have very huge sizes, since a mesh of FEM model can easily contain thousands of nodes and these nodes can have multiple DOFs. Only the mode shapes corresponding to the resonances which are in the frequency region of interest are relevant. First of all, the global mass and stiffness calculated by the modal analysis in ANSYS are exported to a '*jobname.full*' file, in which *jobname* is the user specified name of the ANSYS session. An ADPL script is used to extract the global mass and stiffness matrix into a Harwell-Boeing format(*Kmat.matrix* and *Mmat.matrix*) which are then imported into Matlab.

These matrices are saved in a particular format, to reduce computational effort, therefore

the imported matrices have to be rewritten to their original form. Because of the large size (typically 10^6 elements) of structural matrices and the fact that elements are connected to a small set of nodes, resulting in many zero elements, these matrices are converted to sparse matrices. Also only the lower triangular part of the matrices is saved, that is sufficient since global matrices are symmetric. When all of these rewriting steps are considered, the first 20 solutions (6RBM+14FBM) are calculated using the Matlab command $[\Psi_{Nn}, \Omega_{Nn}^2] = \text{eigs}(K_N, M_N, 20, 'sm')$. Lastly one has to note that the global matrices are reordered to minimize bandwidth, which also minimizes computational effort. This process is called *internal ordering*. No boundary conditions are applied, so the BCS ordering has no influence, more info about the ordering can be found in the ANSYS Parametric Design Language Guide¹. To reorder the matrices into the same numbering as the nodes, the so called *user ordering*, the PRNSOL,U command is used to export the first 14 flexible body modes calculated by the ANSYS modal analysis, resulting in the mode shape matrix Ψ_{ANSYS} in which the node numbers are known. The sum of the absolute values of each row of both mode shape matrices Ψ are calculated and sorted in ascending order from which the corresponding indices of both matrices can be found.

If all of these rewriting steps are considered, the original full global matrices $[K_{Nn}]$ and $[M_{Nn}]$ are found, of this the indices indicate the sizes of both matrices. The reordering is checked by solving the eigenvalue problem again using the reordered global matrices. The found eigensolutions are equal to the mode shape vectors Ψ_{ANSYS} .

4.2.1 Mass normalisation

To correlate measurements with analytical models, both modal parameters must be scaled to the same size. First off all the orthogonality of the mode shapes is used to assure that both mode shape matrices Ψ are ordered in the same sequence. Therefore the modal matrices are calculated.

An N DOF system, contains N natural frequencies and also N corresponding mode shapes. Only the set of n resonances are of interest, since they are inside the FRI. For the steel plate case these are the first 14 flexible body modes. Only this part of the mode shape matrix considered in the correlation analysis.

$$\{\Psi\}_r^T [M] \{\Psi\}_r = [m_r] \quad \text{and} \quad \{\Psi\}_r^T [K] \{\Psi\}_r = [k_r], \quad r = 1, 2, \dots, n \quad (4.6)$$

In which the matrices $[m_r]$ and $[k_r]$, called the modal mass and modal stiffness matrix, are diagonal matrices. This implies that the equations of motion are fully decoupled. The scaling of the mode shape vectors is arbitrary. To get a unique, consistent solution the vectors are normalized such that the modal mass matrix is equals to the unity matrix. As a consequence, the diagonal elements of the modal stiffness matrix equal the natural frequencies squared.

$$[m_r] = [I], \quad [k_r] = [\Omega_n^2] = \text{diag}[\omega_1^2, \omega_2^2, \dots, \omega_n^2] \quad (4.7)$$

The mode shapes which are calculated in ANSYS are mass normalized by default.

¹<http://148.204.81.206/Ansys/150/ANSYS%20Parametric%20DesignLanguage%20Guide.pdf>: 4.4

4.3 Model Reduction/Expansion

In order to compare the analytical model with experimental models, the size of both systems have to match. In practice a small set of locations will be measured and FEM models can easily contain thousands of nodes.

In ANSYS the locations of the sensors can be defined as *NamedSelection* resulting in a mesh with nodes at those locations. An APDL script is developed to obtain the corresponding node numbers. With this information, the DOFs can be distinguished into measured and unmeasured DOFs, the so called *master* and *slave* coordinates. Therefore the generalized eigen value problem can be rewritten in the following form:

$$\left[-[\Omega_n^2] \begin{bmatrix} M^{mm} & M^{ms} \\ M^{sm} & M^{ss} \end{bmatrix} + \begin{bmatrix} K^{mm} & K^{ms} \\ K^{sm} & K^{ss} \end{bmatrix} \right] \begin{bmatrix} \Psi_{mn} \\ \Psi_{sn} \end{bmatrix} = [\mathbf{0}] \quad (4.8)$$

in which all selected mode shapes are stored in the columns of the matrix Ψ . For the steel plate, the number of master and slave DOFs is 25 and 9537 respectively. The master DOFs are ordered in the same ordering as the *NamedSelections*, all slave DOFs are ordered by ascending node number. The analytical model contains information of all nodes and can be used to expand the experimental test vectors to the full space of the FEM. Alternatively, the full space matrices can be reduced to the size of the measured system, using the same information. There are many techniques to reduce and expand modal models. In this study the technique called SEREP² is used. The SEREP produces exactly the same dynamic properties in the reduced model as those of the full space model. There are alternative approaches, but for this study only this procedure is used, since comparing different expansion/reduction techniques is done in numerous other studies such as [7]. From this the SEREP procedure is found the most promising technique.

System Equivalent Reduction Expansion Process (SEREP)

As the name suggests, this technique is used for either the reduction of the finite element structural matrices or for the expansion of the measured experimental modal vectors. The SEREP modal transformation relies on the partitioning of the modal equations representing the entire set of FE DOFs relative to the modal DOFs. The SEREP technique utilizes the mode shapes from a full finite element solution to map to the limited set of master DOF. SEREP is intended to perform an accurate mapping matrix for the transformation.

First of all the SEREP reduction is used to reduce the number of degrees of freedom in an analytical model to reduce computing time while attempting to preserve the full DOF characteristics. The full set of DOFs is expressed in terms of the master coordinates using a transformation matrix $[T]$ which is determined by the pseudo-inverse of the master part Ψ_{mn} of the selected mode shapes. For SEREP, the number of master DOFs N_{mn} must be greater than or equal to the number of modes N_{Ψ} contained in the analytical modal matrix. If the number of selected mode shapes is equal to the number of master DOFs the

²<https://sem.org/wp-content/uploads/2016/01/sem.org-IMAC-XIV-14th-Int-14-36-1-SEREP-Expansion.pdf>

expanded mode shape vectors are exact. In the standard SEREP N_Ψ is equal to the number of experimental mode shapes.

$$\{X_{Nn}\} = \begin{Bmatrix} X_{mn} \\ X_{sn} \end{Bmatrix} = [T] \{X_{mn}\}, \quad [T] = [\Psi_{Nn}] [\Psi_{mn}]^\dagger \quad (4.9)$$

If N_Ψ is less than N_{mn} the original experimental vectors are smoothed out in a least-squares sense to best fit the test data, so original measured data is also changed. This can be beneficial to reduce noise, but there is also risk to consistency issues. To clarify this concept, in appendix D the SEREP technique is applied to a small 10 DOF lumped system using 4 master degrees of freedom.

The reduced system will be used in to perform error localisation. After various error matrices are calculated, these results can be expanded to the full space. Now that the SEREP technique is treated, systems of different sizes can be compared by levelling the sizes of the matrices. Not only FEM with measurements, but also different measured models. For instance, if in future investigation response of a system are measured in an optical way, many more outputs can be measured. In that case the SEREP technique can then also be used to expand and reduce the system matrices.

4.3.1 Reducing the analytical model

Obtaining reduced analytical structural matrices using global matrices

All nodes are expressed in terms of the master degrees of freedom by means of the SEREP proses. This same transformation matrix can be used to reduce the full space matrices, resulting in the reduced structural matrices $[K_m]$ and $[M_m]$.

$$[K_m] = [T]^T [K_N] [T], \quad [M_m] = [T]^T [M_N] [T] \quad (4.10)$$

These reduced matrices represent the same dynamic behaviour, while their size is significantly smaller compared to the full space structural matrices. In this example, 25×25 compared to 9612×9612 . As to see in figure 4.4, the modal matrices remain exactly the same.

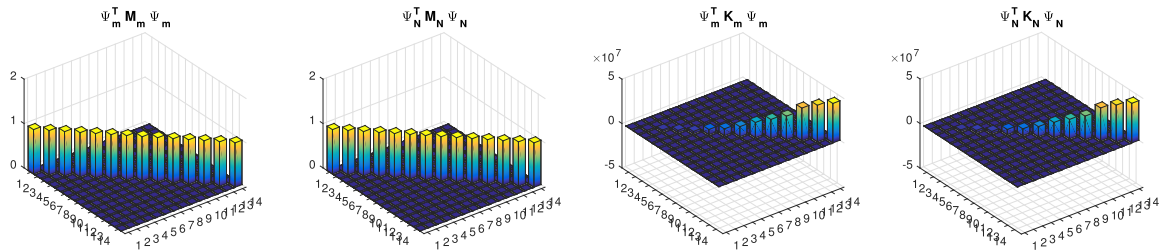


Figure 4.4: Modal matrices in terms of reduced and full space


One has to note small index n is left out of notation for readability reasons. The mode shape matrix Ψ_{mn} is the part of the full space matrix Ψ_{Nn} and it is not equal to the eigensolution of $\text{eig}([K_m], [M_m])$ since this is a different eigenvalue problem.

Obtaining reduced analytical structural matrices using Ψ_{mn} only


If one is only interested in the reduced matrices, it is also possible to determine them directly; without knowledge of the full space matrices. Therefore, only the master part of the mode shapes Ψ_{mn} is required. This can be explained by recalling the definition of the transformation matrix $[T] = [\Psi_{Nn}] [\Psi_{mn}]^\dagger$, using this to rewrite equation 4.10 yields the following expression, in which the modal matrices can be recognized

$$[K_m] = [T]^T [M_N] [T] = [\Psi_{mn}]^{\dagger T} [\Psi_{Nn}]^T [K_N] [\Psi_{Nn}] [\Psi_{mn}]^\dagger = [\Psi_{mn}]^{\dagger T} [\Omega_n] [\Psi_{mn}]^\dagger \quad (4.11)$$

$$[M_m] = [T]^T [M_N] [T] = [\Psi_{mn}]^{\dagger T} [\Psi_{Nn}]^T [M_N] [\Psi_{Nn}] [\Psi_{mn}]^\dagger = [\Psi_{mn}]^{\dagger T} [\Psi_{mn}]^\dagger \quad (4.12)$$

So the full space system matrices are not required in order to compute the reduced system matrices. This saves a lot of computational effort. The downside is that the mapping to the FEM is not known directly, but this can be achieved using the SEREP expansion procedure. The Matlab script in which this is implemented can be found in  ANSYS/GetAnalyticalModel. For large FEM this can be very useful, since only the nodal solution at the measured locations have to be calculated.

4.3.2 Orthogonality checks

The FEM can be verified using measured experimental modal data to verify the adequacy of the model. The structural matrices can be combined with the experimental and analytical mode shapes to perform orthogonality checks. The Matlab script in which this is implemented can be found in  SystemCorrelation/OrthogonalityChecks. Furthermore the correctness of the reduction/expansion of models can be verified using the so-called Pseudo-Orthogonality Check (POC), which must be exactly the same for reduced and full space systems. As already explained in paragraph 4.2.1, the eigensolutions of the FEM are fully decoupled, which implies that they are orthogonal. By replacing one of the analytical mode shape matrices in equation 4.6 with the experimental one, the so called POC is calculated. This result can be used to indicate how well the measured mode shapes are correlated to the analytical mode shapes, with consideration of the system matrices. The structural matrices scale the differences and therefore erroneous estimated modes show up based on their importance to the full dynamic behaviour of the system. This can give more insight in the important deviations between models. The results in the following expressions:

$$\text{POC}_m = [\Psi_{mn}^a]^T [M_r] [\Psi_{mn}^x] = [\Psi_{Nn}^a]^T [M_N] [\Psi_{Nn}^x] \quad (4.13)$$

$$\text{POC}_m = [\Psi_{mn}^x]^T [M_r] [\Psi_{mn}^a] = [\Psi_{Nn}^x]^T [M_N] [\Psi_{Nn}^a] \quad (4.14)$$

in which the superscripts a and x indicate the *analytical* and *experimental* mode shapes. The same principle can be applied using the stiffness matrix to the POC_k . These POCs can also be calculated without the use of structural matrices. Only the master coordinates of mode shape vectors Ψ_{mn} are required: To prove this statement, one must take a closer look

to the components which constitute the POC .

$$POC_m = [\Psi_{mn}^a]^T [M_r] [\Psi_{mn}^x] = [\Psi_{mn}^a]^T \left([T]^T [M_N] [T] \right) [\Psi_{mn}^x] \quad (4.15)$$

$$= [\Psi_{mn}^a]^T \left([\Psi_{mn}^a]^\dagger [\Psi_{Nn}^a]^T [M_N] [\Psi_{Nn}^a] [\Psi_{mn}^a]^\dagger \right) [\Psi_{mn}^x] \quad (4.16)$$

In this expression the expanded experimental modal matrix, $\Psi_{Nn}^x = [\Psi_{Nn}^a] [\Psi_m^a]^\dagger [\Psi_{mn}^x]$ can be recognized. Furthermore the first two matrices drop out since the matrix multiplication of the analytical master coordinates matrix $[\Psi_{mn}^a]$ with its pseudo-inverse equals the unity matrix. Substituting this information to rewrite equation 4.15 into a more compact form yields:

$$POC_m = [I] [\Psi_{Nn}^a]^T [M_N] \Psi_{Nn}^x = [\Psi_{Nn}^a]^T [M_N] [\Psi_{Nn}^x] \quad (4.17)$$

from which is proven that the POC s calculated in terms of reduced or full space will give the same results. Obviously, calculating in reduced space will have significant computational benefits.

These POC s can also be calculated without using any of structural matrices. Therefore the POC is written in full space coordinates. By decomposing the expanded experimental modal matrix, Ψ_{Nn}^x , into reduced experimental coordinates the mass orthogonality relationship can be recognized, which equals the unity matrix and therefore drops out of the equation:

$$POC_m = [\Psi_{Nn}^a]^T [M_N] [\Psi_{Nn}^x] = [\Psi_{Nn}^a]^T [M_N] [\Psi_{Nn}^a] [\Psi_{mn}^a]^\dagger [\Psi_{mn}^x] \quad (4.18)$$

$$= [\Psi_{mn}^a]^\dagger [\Psi_{mn}^x] \quad (4.19)$$

similar step can be executed to get the stiffness POC_K which will be equal to the following expression:

$$POC_k = [\Psi_{mn}^a]^\dagger [\Psi_{mn}^x] [\Omega_n^2] \quad (4.20)$$

So the POC_k is obtained by post-multiplying the POC_m with the diagonal matrix $[\Omega_n^2]$. Therefore this matrix will only be scaled, but it will not contain new information.

Application to estimated models

Now that the theory is treated, the POCs are evaluated using the estimated and analytical models of both the steel plate and the BF. As to see in figure 4.5a it is clear that the biggest deviation originates from the 12th modepair. It is very comparable to the MAC matrix shown in figure 4.1a, but now the off-diagonal terms are scaled to a smaller value, since the mass contribution of the off-diagonal terms is also significantly lower. So the deviation is indicated with more significance. In case of more and bigger deviations this improvement will be even more, so the effort of including the structural matrices helps to recognise of most significant deviations.

The same POC matrices are calculated for the estimation BF, for convenience these results are shown in appendix E

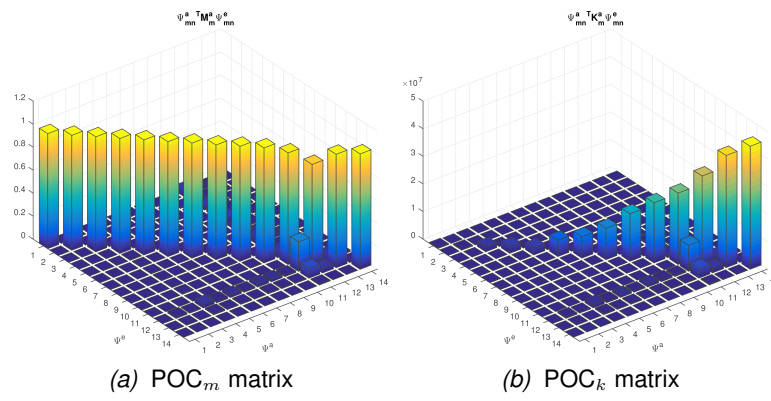


Figure 4.5: POC matrices evaluated for estimation case 1

Conclusions and recommendations

5.1 Conclusions

In this thesis a powerful toolbox is developed which provides the Structural Dynamics competence at VDL ETG with Matlab scripts to perform a full EMA. The Matlab scripts are build in a GUI to perform the impact measurements and combine them in a complete FRM.

Furthermore a stable, user-friendly, efficient curve fitting technique is found in the pLSCF estimator, which accurately fits an RMFD model to the measured FRFs.

The estimated RMFD models result in clear stabilisation diagrams, from which the poles of the modal model are identified. The user only has to select the most relevant FRFs and the order of the polynomials. The poles are then used to estimate the modal model. If the poles are perfectly identified, the estimated modal model is also.

A collection of the most applicable correlation metrics is implemented in Matlab scripts. This collection can be used either to verify the quality of the fit or to compare models. All this information is relevant to acquire insight in the dynamic behaviour of the fabricated systems.

5.2 Recommendations

In this thesis treated all fundamental steps of the EMA and all treated theory is implemented in Matlab Scripts resulting in a useful toolbox for the Structural Dynamics competence of VDL ETG. For the data acquisition a GUI is made to make the data acquisition more convenient. To develop this EMA toolbox even further the following recommendations are presented:

1. Computational effort can be saved by implementing the Toeplitz notation of the matrices R_o , S_o and T_o in Matlab.
2. Error localisation can be used to identify updating parameters of the structural matrices of the FEM of the system under study.
3. Calculation sensitivities of the FEM w.r.t. these updating parameters
4. Use this information to update the FEM using an FRF based optimization technique

Looking at the R_o , S_o and T_o matrices, one can clearly see that these matrices show Toeplitz behaviour. By making use of the build-in Matlab function `toeplitz` computational effort can be saved. For this thesis experienced CPU time was very short, in order of seconds, so no significant time will be saved. For larger systems involving larger datasets this becomes relevant.

An attempt was made to develop tools(matlab scripts) which could localize errors using the structural matrices calculated by the FEM packages, but no real satisfactory results were obtained. For lumped system cases some methods worked but for practical implementation more investigation must be performed to identify the most relevant parameters of the FEM. In this thesis all precautions are already worked out, involving the methods to export the structural matrices and to reduce and expand these matrices to level the size of both measured and analytical systems. So this thesis can be used a good starting point for further investigation in this topic.

Nowadays many FEM packages contain features to export the sensitivities. This was not treated in this thesis, since no clear error localisation was developed. If this issue is solved, the sensitivities can be used to update the structural matrices of the FEM.

Bibliography

- [1] W. H. S. L. P. Sas, *Modal Analysis Theory and Testing*.
- [2] B. Schwarz. (1999) Introduction to operating deflection shapes. [Online]. Available: <http://www.systemplus.co.jp/support/data/techpaper/mescope/tech/29.pdf>
- [3] G. Kerschen. (2009) Experimental modal analysis. [Online]. Available: http://www.ltas-vis.ulg.ac.be/cmsms/uploads/File/Mvibr_notes.pdf
- [4] B. Cauberghe, "Applied frequency-domain system identification in the field of experimental and operational modal analysis," 2004. [Online]. Available: <https://pdfs.semanticscholar.org/13a8/870be66e79b3f00b140762625dc1b5f41a98.pdf>
- [5] R. J. Allemang. The modal assurance criterion twenty years of use and abuse. [Online]. Available: <http://www.sandv.com/downloads/0308alle.pdf>
- [6] W. Heylen, "correlation considerations part 5 (degree of freedom correlation techniques)," in *Proceedings of the 16th International Modal Analysis Conference*, 1998.
- [7] P. L. John O'Callahan. (2016) Serep expansion. [Online]. Available: <https://sem.org/wp-content/uploads/2016/01/sem.org-IMAC-XIV-14th-Int-14-36-1-SEREP-Expansion.pdf>
- [8] K. J. Keesman, *System Identification: an introduction*, L. S.-V. L. Limited., Ed. Springer, 2011. [Online]. Available: <http://www.springer.com/gp/book/9780857295217>

Laplace, Fourier and Z-Transforms

This appendix originates from the following textbook [8].

Laplace Transform The Laplace transform is one of the best-known and most widely used integral transforms. It is commonly used to produce an easily solvable algebraic equation from an ordinary differential equation. Furthermore, the Laplace transform is often interpreted as a transformation from the time domain, in which inputs and outputs are functions of time, to the frequency domain, where the same inputs and outputs are functions of complex angular frequency, or radians per unit time. For LTI systems, the Laplace transform provides an alternative functional description that often simplifies the analysis of the behaviour of the system. The most commonly applied Laplace transform is defined as

$$\mathcal{L}[f(t)] \equiv F(s) := \int_0^{\infty} f(t)e^{-st} dt \quad (\text{A.1})$$

It is a linear operator on a function $f(t)$ (original) with real argument t that transforms it to a function $F(s)$ (image) with complex argument s . The Laplace transform has the useful property that relationships and operations over the originals $f(t)$ correspond to simpler relationships and operations over the images $F(s)$.

Fourier Transform The Fourier transform shows a close similarity to the Laplace transform. The continuous Fourier transform is equivalent to evaluating the bilateral Laplace transform with complex argument $s = i\omega$, with ω in rad/s. The result of a Fourier transformation of a real-valued function ($f(t)$) is often called the frequency domain representation of the original function. In particular, it describes which frequencies are present in the original function. There are several common conventions for defining the Fourier transform of an integrable function $f(t)$. In this book, with angular frequency $\omega = 2\pi z$ in rad/s and frequency z in Hertz, we use:

$$\mathcal{F}[f(t)] \equiv F(\omega) := \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad (\text{A.2})$$

for every real number of t . The most important property for further use in this book is illustrated by the following example.

Example A.1 *The Fourier transform:* Fourier transformation of the convolution integral $y(t) \int_{-\infty}^t g(t - \tau)u(\tau)\delta\tau$

$$Y(\omega) = G(\omega)U(\omega) \quad (\text{A.3})$$

which, as in the case of the Laplace transform, defines an algebraic relationship between transformed output signal $Y(\omega)$ and transformed input signal $U(\omega)$. For discrete signals the Discrete Time Fourier Transform is used, in which the integral is replaced by a summation:

$$F_N(\omega) = \frac{1}{\sqrt{N}} \sum_{t=1}^N f(t)e^{-i\omega t} \quad (\text{A.4})$$

where $\omega = 2\pi k/N$, $k = 1, 2, \dots, N$. N/k is the period associated with the specific frequency ω_k . The absolute square value of $F(\omega_k)$, $|F(2\pi k/N)|^2$, is a measure of the energy contribution of this frequency to the energy of the signal. The plot of values of $|F(\omega)|^2$ as a function of ω is called the periodogram of the signal $f(t)$.

z-Transform The z -transform converts a discrete time-domain signal, which in general is a sequence of real numbers, into a complex frequency domain representation. The z -transform is like a discrete equivalent of the Laplace transform. The unilateral or one-sided z -transform is simply the Laplace transform of an ideally sampled signal after the substitution $z = e^{sT_s}$, with T_s the sampling interval. The z -transform can also be seen as a generalization of the Discrete Fourier transform (DFT), where the DFT can be found by evaluating the z -transform $F(z)$ at $z = e^{i\omega}$. The two-sided z -transform of a discrete-time signal $f(t)$ is the function $F(z)$ defined as

$$\mathcal{L}[f(t)] \equiv F(z) := \sum_{t=-\infty}^{\infty} f(t)z^{-t} \quad (\text{A.5})$$

where $t \in \mathcal{Z}$, and z is, in general, a complex number. In this book, and basically for causal signals, the unilateral z -transform is used as well and is given by

$$\mathcal{L}[f(t)] \equiv F(z) := \sum_{t=0}^{\infty} f(t)z^{-t} \quad (\text{A.6})$$

Again, a very relevant property of the z -transform is illustrated in the following.

Example A.2 *The z-transform:* z -transformation of the convolution sum

$$y(t) = \sum_{k=0}^{t} g(t - k)u(k) \quad (\text{A.7})$$

with $y(t)$, $u(t)$, and $g(t)$ discrete-time functions, gives

$$Y(z) = G(z)U(z) \quad (\text{A.8})$$

which defines a similar relationship between transformed output signal $Y(z)$ and transformed input signal $U(z)$, as in the case of Laplace or Fourier transformation.

Correlation metrics of identified systems

The FRAC [6] The use of FRFs for correlation instead of eigenvectors is attractive because experimental FRFs are measured directly, therefore they are more easily obtainable than eigenvectors. The eigenvectors require significant post measurement analysis of the FRM(system identification). The FRAC is defined by the following expression:

$$FRAC_{io} = \frac{\left| \{H_{io}^a(\omega_i)\}^T \{H_{io}^x(\omega_i)\}^* \right|^2}{\left(\{H_{io}^a(\omega_i)\}^T \{H_{io}^a(\omega_i)\}^* \right) \left(\{H_{io}^x(\omega_i)\}^T \{H_{io}^x(\omega_i)\}^* \right)} \quad (B.1)$$

in this $H_{io}^a(\omega_i)$ and $H_{io}^x(\omega_i)$ refer to the FRF corresponding to input DOF i and output DOF o of FRM a and x respectively; In practice most of times a will be the 'analytical' dataset and x will be the 'experimental' dataset. The FRFs can be compared over the full FRI, a part of this range or only at the resonance frequencies. The resulting matrix indicates which DOFs are correlated. The matrix is evaluated for using the simulated and estimated FRMs, shown in figure 4.1a.

The MAC The modal parameters of the estimated modal model can also be compared. MAC shows the degree of correlation between two vectors and it is computed by the following expression:

$$MAC_{cdr} = \frac{\left| \{\Psi_{cr}\}^T \{\Psi_{dr}^*\} \right|^2}{\left(\{\Psi_{cr}\}^T \{\Psi_{cr}^*\} \right) \left(\{\Psi_{dr}\}^T \{\Psi_{dr}^*\} \right)} \quad (B.2)$$

The resulting matrix indicates which modes are correlated and therefore identifies the mode pairs of in the two datasets. The matrix is evaluated for using the simulated and estimated mode shape matrices, shown in figure 4.1b.

The MSF Once all modes are paired, their sizes and orientations can be equalized using the MSF. It is calculated using the following equation:

$$MSF_{cdr} = \frac{\left| \{\Psi_{cr}\}^T \{\Psi_{dr}^*\} \right|}{\left(\{\Psi_{cr}^*\}^H \{\Psi_{cr}^*\} \right)} \quad (B.3)$$

By carefully applying these three checks both datasets are equally oriented in terms of DOFs, mode shapes and directions. The matrices shown in figure B.1 show high values at the diagonal, indicating the estimated model is very well correlated to the simulated model.

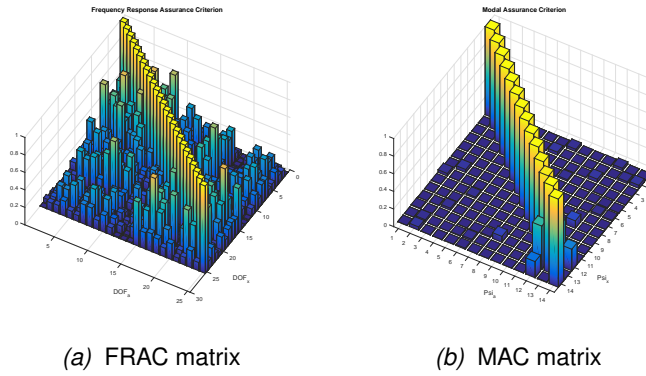


Figure B.1: FRAC and MAC matrices evaluated for estimation case 1

The COMAC Now that both modal matrices are correctly scaled and oriented, the COMAC can be calculated to indicate the correspondence between the mode shapes of two datasets in terms of the DOFs. This indicates which DOF is deviating the most, which can give the engineer insight in the origin of the deviation without need to calculate the structural matrices of the system. For this case the following result is obtained:

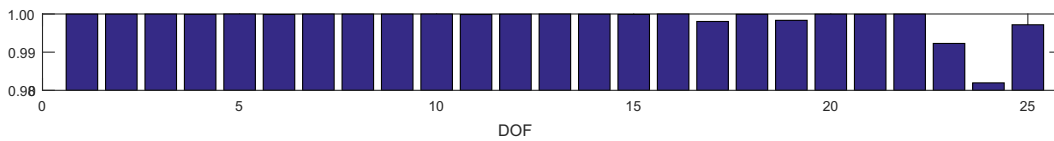


Figure B.2: COMAC values for estimation case 1

From which one can conclude the estimation is almost perfect, the worst estimation is involved with mainly DOF 23 and 25.


Deducing analytical models from ANSYS FEM


In this appendix the procedure to obtain a full space analytical model of a FEM is explained step by step. To assure the obtained model is consistent with the modal analysis performed in ANSYS there are some import aspects to note, following this manual will assure this is done correctly, in case of a FEM model in which no boundary conditions are applied.

The procedure enhances the following steps:

1. **Export the structural matrices $[K_N]$ & $[M_N]$ to .matrix binary files, using the Harwell Boeing(HB) format**
2. **Export the mode shape matrix $[\Phi_N^a]$ to .eig text file**
3. **Import all data into Matlab**
4. **Relocate the DOFs structural matrices to the user defined ordering.**

For convenience, all used scripts are attached at the end of this manual.

Using ANSYS Workbench Modal Analysis To illustrate the procedure, the steel plate model is used. This model is build using ANSYS Workbench 17.2 Mechanical. The complete case study can be found in  ANSYS/ANSYS_CaseStudy

The system is measured at the 25 intersection points of the measurement grid. To make sure that each point is described by a node of the mesh, these locations are defined as *NamedSelections*. By adding APDL code snippets to the modal analysis tree, see figure C.1b, the required nodal information of these locations is written to a Matlab file, containing the deformations and corresponding node numbers. As the name suggests, the measured nodes are defined by the snippet '*DefineMasterNodes*'. All required pre- and post-processing is done by '*res_pre*' and '*res_pos*' respectively. All these code snippets can be found in  ANSYS/GetEigenVectors.

After solving the modal analysis a matlabfile, named '*SessionName*' + the date on which the file is created, will be written in the folder of the corresponding session, in this case

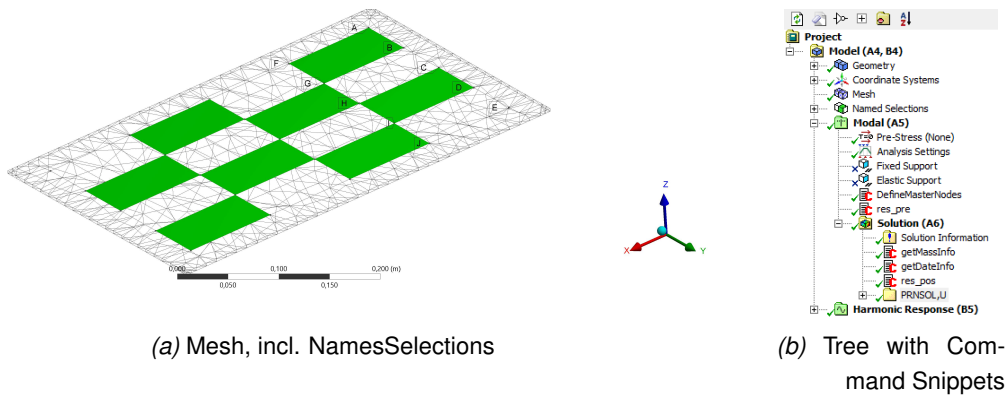


Figure C.1: ANSYS WB 17.2: model

ANSYS_CASESTUDY_PLATE_files/dp0/SYS/MECH. After this is done the deducing procedure can be executed.

1. Export the structural matrices $[K_N]$ & $[M_N]$ to .matrix binary files, using the Harwell Boeing(HB) format

The structural matrices are exported using the binary file dumping processor AUX2 by running the following script:

```
1 \aux2 -- open aux2 environment
2 hbmata,STIFFNESS,MATRIX,,ASCII,stiff,no,no
3 hbmata,MASS,MATRIX,,ASCII,mass,no,no
```

Implementation of this script in ANSYS 17.2 returns a warning message that the command hbmata is unknown. however, the ASCII-files are exported correctly so one can neglect this warning. Alternatively, the utility menu can be used using: UM→File→List→BinaryFiles.

2. Export the mode shape matrix $[\Phi_N^a]$ to .eig text file

When the eigenvalue problem has been solved, the following script will write the ASCII-le 'file.eig' which all nodal solution are printed using PRNSOL, U. In this case the file contains the first 14 FB modes, all modes which are inside the frequency region of interest.

```
1 /post1 -- open post post processor
2 /output,,eig ! -- create the output .eig file
3 *do,i,7,20 ! -- set up do loop
4 set,,i prnsol,dof ! -- set ith eigensolution
5 prnsol,dof ! -- show ith eigensolution
6 /out -- write ith eigensolution to .eig file
7 *enddo
```


Reason to start with the 7th eigensolution is that the first 6 modes are RB modes and therefore they are irrelevant. Now all required matrices are exported into a format which is compatible to Matlab. Furthermore some text `.txt` file are printed to check if the imported matrices correspond to what is expected. These scripts can be found in ANSYS Parametric Design Language Guide¹, this is not necessary but can be used to verify the produced matrices.

Name	Date modified	Type	Size	Name	Date modified	Type	Size
ds	9-5-2017 15:23	DAT File	517 KB	K	1-5-2017 08:41	Text Document	111 KB
ds_file.stat	9-5-2017 21:31	STAT File	1 KB	Kmat	9-5-2017 21:37	MATRIX File	12.564 KB
ds_file	9-5-2017 21:29	XML Document	1 KB	MatML	9-5-2017 15:23	XML Document	22 KB
file	9-5-2017 15:23	ANSYS v172 .db File	3.392 KB	menust.tmp	9-5-2017 21:29	TMP File	1 KB
file.dmp	24-4-2017 13:06	DMP File	0 KB	Mmat	9-5-2017 21:38	MATRIX File	4.665 KB
file.DSP	9-5-2017 15:23	DSP File	2 KB	Phi	30-4-2017 19:55	Text Document	6.552 KB
file.eig	9-5-2017 21:36	EIG File	3.909 KB	PhiTKPhi	24-4-2017 14:37	Text Document	17 KB
file.esav	9-5-2017 15:23	ESAV File	448 KB	PhiTMPPhi	24-4-2017 14:37	Text Document	17 KB
file.full	9-5-2017 15:23	FULL File	5.888 KB	PhiTPhi	30-4-2017 19:55	Text Document	17 KB
file.lock	9-5-2017 21:29	LOCK File	0 KB	PlateModalAnalysis_09May2017	9-5-2017 15:23	MATLAB Code	100 KB
file	9-5-2017 21:38	Text Document	1 KB	PlateModalAnalysis_24Apr2017	24-4-2017 13:34	MATLAB Code	100 KB
file.mode	9-5-2017 15:23	MODE File	2.432 KB	solve.out	9-5-2017 15:23	OUT File	387 KB
file.page	9-5-2017 21:29	PAGE File	0 KB	SYS	10-1-2017 15:03	WinRAR ZIP archive	1 KB
file.rst	9-5-2017 15:23	RST File	36.736 KB				

Figure C.2: Folder of the corresponding session

If all steps are succeeded the ANSYS folder should contain the following files, as shown in figure C.2. All this data will now be imported into Matlab, the ANSYS program is not used for the rest of the procedure.

3. Import all data into Matlab

Every time ANSYS solves the session, the working directory is cleaned up and removed. To prevent that exported matrices are deleted, all relevant files are copied to `ANSYS/GetAnalyticalModel/ANSYS_Data`, shown in figure C.3. All used functions can be found on the upper level.

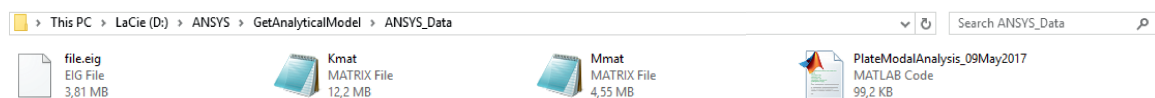


Figure C.3: Folder containing all exported data

First of all the matlab file is runned, to obtain the measured nodes and their initial coordinates. Subsequently the structural matrices are converted to sparse matrices using the function `hb_to_msm`². The format minimizes memory size, since only non-zeros elements are described. Furthermore the matrices are symmetric, so only the lower triangular part is required to describe the entire matrix. In matlab, the full matrices are reconstructed again. Subsequently, the printed nodal solutions are imported using the function `importU`, a Matlab

¹<http://148.204.81.206/Ansys/150/ANSYS%20Parametric%20DesignLanguage%20Guide.pdf>: 4.7

²Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

generated import function. The result is one giant matrix U which is then converted to the usual mode shape vector format.

4. Relocate the DOFs structural matrices to the user defined ordering.

Next step is to solve the eigenvalue problem using the imported structural matrices, again the RBMs are deleted from the eigensolutions. Calculating the modal mass and stiffness matrix, yields to the same matrices as in equation 4.6. The DOFs are internally ordered and have to be mapped into the user defined ordering in which the DOFs are linked to the nodes numbers. To reconstruct this mapping, the absolute values of each column is summed for both matrices and sorted in ascending order, as to see in figure C.4 in which the last column represents the summation of all other columns. Now a unique match is found between both matrices, from which the mapping is determined. The structural matrices are reordered according to the user definition and this results in eigensolution which are identical to the ones printed by ANSYS.

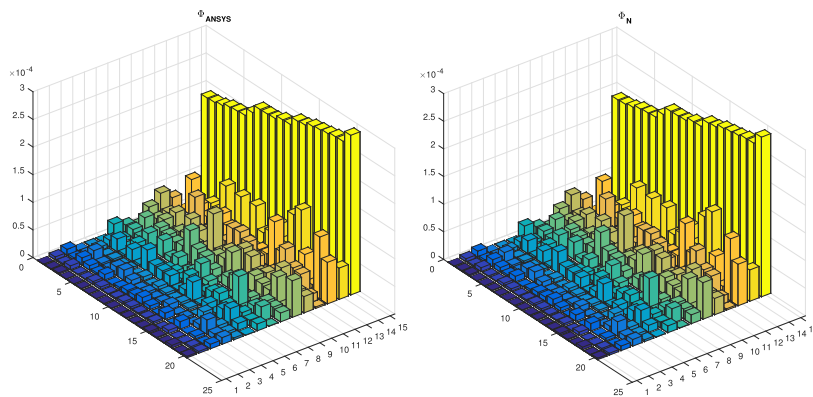


Figure C.4: Comparing modeshape matrices

There must be some APDL code possible in which the matrices are exported in user defined ordering directly, but this was not found in any of the ANSYS help files. In future research this would be an improvement to obtain the analytical full space model in a more convenient manner.

C.1 Matlab Code

```
1 clc; clear all; close all;
2 addpath([pwd, '\ANSYS_Data']); % Add ANSYS_Data Folder
3
4 % Import ANSYS matrices
5 ImportFlag = false;
6 if ImportFlag == 1
7
```

```

8 PlateModalAnalysis_09May2017% get nodal information
9 Sys.N_m = Nodeandn(:,1);% collect indices of master nodes
10 Sys.X_m = Nodeandn(:,2:end);% collect nodal coordinates
11
12 clearvars -except Sys % Clear all irrelevant nodal information
13
14 K_sp = hb_to_msm('Kmat.matrix');% Sparse stiffness matrix
15 M_sp = hb_to_msm('Mmat.matrix');% Sparse stiffness matrix
16
17 [U_raw] = importU('file.eig');
18 Ui = find(isnan(U_raw(:,1))==0);U = U_raw(Ui,:);
19
20 else
21     load AnsysMatrices;
22 end
23
24 % Construct structural matrices
25 K_N = K_sp + tril(K_sp,-1);% Global stiffness matrix
26 M_N = M_sp + tril(M_sp,-1);% Global mass matrix
27 scrsz = get(groot,'ScreenSize');
28 figure('Name','imported and reconstructed matrix')
29 set(gcf,'Position',[1 scrsz(4)/21 scrsz(3)/4 scrsz(4)/4])
30 subplot(121)
31 bar3(K_sp(1:100,1:100))
32 subplot(122)
33 bar3(K_N(1:100,1:100))
34
35 clear K_sp M_sp % clear irrelevant parameters
36
37 % Construct mode shape matrix Phi_ANSYS
38 N_dof= length(K_N); N_nds = N_dof/size(Sys.X_m,2);
39 N_mds= length(U)/N_nds;
40
41 PHI_a = U(:,2:end)'; Phi_ANSYS = zeros(N_dof,N_mds);
42
43 for i = 1:N_mds
44     i_mn = (i-1)*N_nds+1; i_mx = i*N_nds; Phi = PHI_a(:,i_mn:i_mx);
45     Phi_ANSYS(:,i) = Phi(:); %
46 end
47
48 % Solve generalized eigenvalue problem using global structural matrices
49 [Phi_N,Omega_N] = eigs(K_N,M_N,20,'sm');
50 Phi_N = fliplr(Phi_N(:,1:14)); Omega_N = Omega_N(1:14,1:14);
51 Omega_N = flipud(Omega_N); % horizontal flip
52 Omega_N = fliplr(Omega_N); % vertical flip
53 Wn = real(sqrt(diag(Omega_N)))/2/pi;% eigen-frequencies [Hz]
54
55 scrsz = get(groot,'ScreenSize');
56 figure('Name','Orthogonality check')
57 set(gcf,'Position',[1 scrsz(4)/21+scrsz(4)/4 scrsz(3)/4 scrsz(4)/4])
58 subplot(121)

```

```

59 bar3(Phi_N'*K_N*Phi_N)
60 title('\Phi_N^T M_N \Phi_N')
61 subplot(122)
62 bar3(Phi_N'*M_N*Phi_N)
63 title('\Phi_N^T K_N \Phi_N')
64
65 % reorder global matrices to user defined DOFs;
66 [SPhi_ANSYS,i_SPhi_ANSYS] = sort(sum(abs(Phi_ANSYS),2));% sum all rows, ...
    sort ascending order
67 [SPhi_N,i_SPhi_N] = sort(sum(abs(Phi_N),2));% sum all rows, sort ascending ...
    order
68
69 figure('Name','Phi comparison')
70 set(gcf,'Position',[1 scrsz(4)/21+scrsz(4)/4 scrsz(3)/4 scrsz(4)/4])
71 subplot(121)
72 bar3([abs(Phi_ANSYS(i_SPhi_ANSYS(1:150),:)),...
    sum(abs(Phi_ANSYS(i_SPhi_ANSYS(1:150),:)),2)])
73 title('\Phi_{ANSYS}')
74 subplot(122)
75 bar3([abs(Phi_N(i_SPhi_N(1:150),:)),sum(abs(Phi_N(i_SPhi_N(1:150),:)),2)])
76 title('\Phi_N')
77
78
79 [~,i_SANSYS_n] = ismember([1:9612]',i_SPhi_ANSYS);% indices to reoder sum ...
    ind. to asc. order
80
81 figure('Name','Phi comparison 2')
82 set(gcf,'Position',[1 scrsz(4)/21+scrsz(4)/4 scrsz(3)/4 scrsz(4)/4])
83 subplot(121)
84 bar3([abs(Phi_ANSYS(1:150,:))])
85 title('\Phi_{ANSYS}')
86 subplot(122)
87 bar3([abs(Phi_N(i_SPhi_N(i_SANSYS_n(1:150))),:))])
88 title('\Phi_N')
89
90 i_ud = i_SPhi_N(i_SANSYS_n);% mapping from internal ordering to user ...
    defined ordering
91 K_N = K_N(i_ud,i_ud);% reorder global stiffness matrix
92 M_N = M_N(i_ud,i_ud);% reorder global mass matrix
93 [Phi_N,Omega_N] = eigs(K_N,M_N,20,'sm');% find first 20 eigensolutions
94 Phi_N = fliplr(Phi_N(:,1:14)); Omega_N = Omega_N(1:14,1:14);% delete Rigid ...
    Body Modes
95 Omega_N = flipud(Omega_N); % horizontal flip
96 Omega_N = fliplr(Omega_N); % vertical flip
97 Wn = real(sqrt(diag(Omega_N)))/2/pi;% eigen-frequencies [Hz]
98
99 figure('Name','Orthogonality check udo')
100 set(gcf,'Position',[1 scrsz(4)/21+3*scrsz(4)/4 scrsz(3)/4 scrsz(4)/4])
101 subplot(121)
102 bar3(Phi_N'*K_N*Phi_N)
103 title('\Phi_N^T M_N \Phi_N')
104 subplot(122)

```

```
105 bar3(Phi_N'*M_N*Phi_N)
106 title('\Phi_N^T K_N \Phi_N')
107
108 figure
109 set(gcf, 'Position', [1 scrsz(4)/21+2*scrsz(4)/4 scrsz(3)/4 scrsz(4)/4])
110 bar3([SPhi_ANSYS(1:450), SPhi_N(1:450)])
111 figure
112 set(gcf, 'Position', [1 scrsz(4)/21+2*scrsz(4)/4 scrsz(3)/4 scrsz(4)/4])
113 bar3([SPhi_ANSYS(1:450), SPhi_N(1:450)])
114
115 figure('Name', 'Phi comparison')
116 set(gcf, 'Position', [1 scrsz(4)/21+scrsz(4)/4 scrsz(3)/4 scrsz(4)/4])
117 subplot(121)
118 bar3(abs(Phi_ANSYS((1:150),:)))
119 title('\Phi_{ANSYS}')
120 subplot(122)
121 bar3(abs(Phi_N((1:150),:)))
122 title('\Phi_N')
123
124 save AnalyticalModel K_N M_N Phi_ANSYS Phi_N Sys Omega_N % Save results ...
    for file
125 clearvars -except K_N M_N Phi_ANSYS Phi_N Sys Omega_N % Clear all ...
    irrelevant nodal information
```


The SEREP technique

In this appendix the concept of SEREP expansion and reduction is illustrated by means of a academic 10 DOF lumped mass system. All steps can be found in the matlab script attached to this appendix. The lumped system is described by the structural matrices, which are both square symmetric 10×10 matrices. Solving the generalized eigen value problem results in the full space model parameters Ω_N^2 and Φ_N , containing 10 eigenvalues and corresponding eigenvectors.

First step in the process is to reorder the system in terms of master and slave DOFs. In this example 6 master DOFs are chosen in arbitrary order. This is done because the node ordering by the NamedSelections in ANSYS is also non-ascending. The slave DOFs are ordered in ascending order. This results in the following system:

$$X_N = [x_6 \ x_9 \ x_7 \ x_1 \mid x_2 \ x_3 \ x_4 \ x_5 \ x_8 \ x_{10}] \quad (D.1)$$

The full space structural matrices are reordered to this same sequence. Furthermore 4 mode shape are selected for the SEREP process, in this case mode shape numbers 8, 7, 1 and 2 are used. This results in the following analytical mode shape matrix:

$$[\Psi_N] = \left[\frac{\Psi_m}{\Psi_s} \right] = \begin{bmatrix} \Psi_{6,8} & \Psi_{6,7} & \Psi_{6,1} & \Psi_{6,2} \\ \vdots & \vdots & \vdots & \vdots \\ \Psi_{1,8} & \Psi_{1,7} & \Psi_{1,1} & \Psi_{1,2} \\ \hline \Psi_{2,8} & \Psi_{2,7} & \Psi_{2,1} & \Psi_{2,2} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \quad (D.2)$$

The master part of the mode shape matrix Φ_m is square in this case. The transformation matrix T is calculated and used to calculate the reduced structural matrices K_r and M_r . To check if the dynamic characteristics are maintained, the modal matrices of both full and reduced space are calculated and shown in figure D.1. The modal stiffnesses corresponding to the selected eigensolutions are maintained. The reduced matrices are used to calculate the reduced mode shapes. Solving this small system yields 4 eigensolutions, corresponding to the ones which were selected. These reduced mode shapes can be expanded to full space using the same transformation matrix T . As to see in figure D.2 the selected eigen-solution are reconstructed by the expansion. Both matrices show a clear match; they are

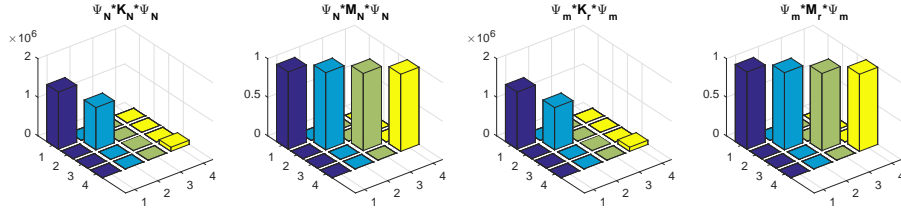


Figure D.1: Modal matrices in terms of full and reduced matrices, square Φ_m

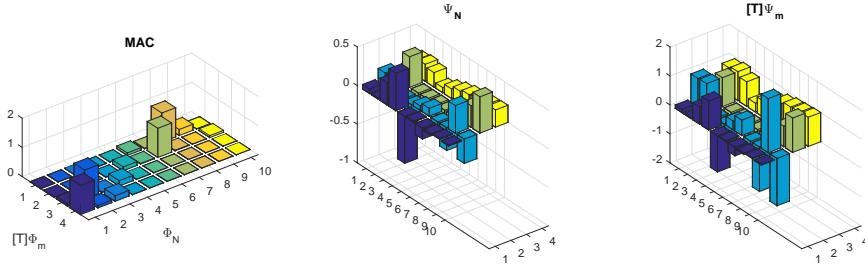


Figure D.2: Mode shape matrices in terms of full and reduced matrices, square Φ_m

identical up to a scaling constant and some modes are reverted. These differences can be solved easily using the modal scale factor and by demanding each first element of a vector must be positive.

So if the master part of the mode shape matrix Φ_m is square the expansion/reduction step is exact. This can also be seen in the transformation matrix; the top part $(\Psi_m(\Psi_m^T \Psi_m)^{-1} \Psi_m')$ will be equal to the identity matrix. In most cases only the first few modes are of interest, resulting in a top part which has significantly more rows than columns. To same steps are performed, but now the results will be smoothed out. In this example only mode shape 8,7 and 1 are selected. The resulting modal matrices become: The resulting mode shapes show

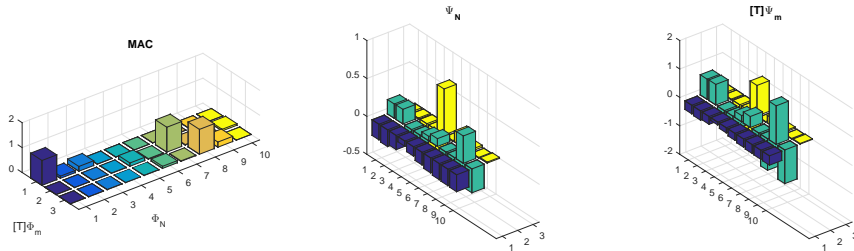


Figure D.3: Mode shape matrices in terms of full and reduced matrices, non-square Φ_m

no difference with the 3 corresponding ones. However, when looking to the top part of the transformation matrix one giant difference is observed: the top part is no longer equal to the identity matrix and shows rank deficiency ($\text{Rank}(\Psi_m(\Psi_m^T \Psi_m)^{-1} \Psi_m') = 3$). Therefore the not linear independent any more; the fourth equation is a linear combination of the other

equations. Due to the fact that the reduced mode shapes are exact this make no difference. In case that these measured mode shapes are contaminated with noise, this expansion/reduction process

D.1 Matlab Code

```

1  clc; clear all; close all
2
3  ma = [ 1 1 2 3 6 5 4 2 1 1];
4  ka = [0.4 0.48 0.60 1.2 2.2 1.6 1.32 1.0 0.8 0.68]*10^6;
5
6  Ma = diag(ma);    Ka = zeros(size(Ma));
7  Ka(1,1) = Ka(1,1) + ka(1);% stiffness 1
8  Ka(1:2,1:2) = Ka(1:2,1:2)+[ka(2) -ka(2);-ka(2) ka(2)];% stiffness 2
9  Ka(2:3,2:3) = Ka(2:3,2:3)+[ka(3) -ka(3);-ka(3) ka(3)];% stiffness 3
10 Ka(3:4,3:4) = Ka(3:4,3:4)+[ka(4) -ka(4);-ka(4) ka(4)];% stiffness 4
11 Ka(4:5,4:5) = Ka(4:5,4:5)+[ka(5) -ka(5);-ka(5) ka(5)];% stiffness 5
12 Ka(5:6,5:6) = Ka(5:6,5:6)+[ka(6) -ka(6);-ka(6) ka(6)];% stiffness 6
13 Ka(6:7,6:7) = Ka(6:7,6:7)+[ka(7) -ka(7);-ka(7) ka(7)];% stiffness 7
14 Ka(7:8,7:8) = Ka(7:8,7:8)+[ka(8) -ka(8);-ka(8) ka(8)];% stiffness 8
15 Ka(8:9,8:9) = Ka(8:9,8:9)+[ka(9) -ka(9);-ka(9) ka(9)];% stiffness 9
16 Ka(9:10,9:10) = Ka(9:10,9:10)+[ka(10) -ka(10);-ka(10) ka(10)];% stiffness 10
17
18 [Psia,Wna] = eig(Ka,Ma); % Analytical modal parameters
19
20 % MASTER DEGREES OF FREEDOM:
21 im = [6 9 4 1]; is = [2 3 5 7 8 10]; % indices of master and slave coord.
22 % ---- number of selected modes equal to number of master DOFs -----
23 in = [8 7 1 2];% selected mode shapes
24 Psim = [Psia(im,in)]; % master DOFs of selected mode shapes
25 Psis = Psia; Psis = Psis(is,in);% slave DOFs of selected mode shapes
26 Psir = [Psim;Psis];% reordered mode shape matrix
27
28 Kar = Ka([im,is],[im,is]);% reordered stiffness matrix
29 Mar = Ma([im,is],[im,is]);% reordered mass matrix
30
31 Uag = (Psim'*Psim)^-1*Psim'; % generalized inverse
32 T = Psir*Uag;% Transformation matrix
33 T_top = Psim*Uag% top part of transformation matrix
34 rank(T_top)% check if all mode shapes are lin. independent
35 [Psi_R,Wn_R] = eig(Kar,Mar);% reordered analytical model parameters
36 Kred = T'*Kar*T;% reduced stiffness matrix
37 Mred = T'*Mar*T;% reduced mass matrix
38
39 Korthog_f = Psir'*Kar*Psir;% modal stiffness matrix(full space)
40 Morthog_f = Psir'*Mar*Psir;% modal mass matrix(full space)
41 Korthog_r = Psim'*Kred*Psim;% modal stiffness matrix(reduced space)
42 Morthog_r = Psim'*Mred*Psim;% modal mass matrix(reduced space)

```

```

43
44 figure('Name','Orthogonalities')
45 subplot(141)
46 bar3(Korthog_f); title('\Psi_N*K_N*\Psi_N')
47 Hm(1) = subplot(142);
48 bar3(Morthog_f); title('\Psi_N*M_N*\Psi_N'); zlim([0 1])
49 subplot(143)
50 bar3(Korthog_r); title('\Psi_m*K_r*\Psi_m')
51 hm(2) = subplot(144);
52 bar3(Morthog_r); title('\Psi_m*M_r*\Psi_m'); zlim([0 1])
53
54 [Phir, Wnr] = eig(Kred,Mred);% solve reduced system
55
56 figure('Name','reduced matrices')
57 subplot(121)
58 bar3(diag([Wna(8,8),Wna(7,7),Wna(2,2),Wna(1,1)]))
59 subplot(122)
60 bar3(Wnr)
61
62 Phi_f = T*Phir;% Expand reduced eigensolution to full space
63
64 MAC = zeros(size(Phi_f,2),size(Psi_R,2));
65 for i = 1:size(Phi_f,2)
66     for j = 1:size(Psi_R,2)
67         MAC(i,j) = (Phi_f(:,i)'*Psi_R(:,j))^2/...
68             ((Phi_f(:,i)'*Phi_f(:,i))*(Psi_R(:,j)'*Psi_R(:,j)));
69     end
70 end
71 figure('Name','MAC')
72 subplot(131)
73 bar3(MAC); title('MAC'); xlabel('\Phi_N'); ylabel('[T]\Phi_m')
74
75 subplot(132)
76 bar3(Psi_R(:, [8 7 2 1])*diag([1 1 1 -1]))
77 title('\Psi_N')
78 subplot(133)
79 bar3(Phi_f(:, :)*diag([-1 -1 1 1]))
80 title('[T]\Psi_m')
81 % ---- number of selected modes less than number of master DOFs -----
82 % MASTER DEGREES OF FREEDOM:
83 im = [6 9 4 1]; is = [2 3 5 7 8 10]; % indices of master and slave coord.
84 in = [8 7 1];% selected mode shapes
85 Psim = [Psia(im,in)];% master DOFs of selected mode shapes
86 Psis = Psia; Psis = Psis(is,in);% slave DOFs of selected mode shapes
87 Psir = [Psim;Psis];% reordered mode shape matrix
88
89 Kar = Ka([im,is],[im,is]);% reordered stiffness matrix
90 Mar = Ma([im,is],[im,is]);% reordered mass matrix
91
92 Uag = (Psim'*Psim)^-1*Psim'; % generalized inverse
93 clear T

```

```

94 T = Psir*Uag;% Transformation matrix
95 T_top = Psim*Uag% top part of Tranformation matrix
96 rank(T_top)% check if all mode shapes are lin. independent
97 [Psi_R,Wn_R] = eig(Kar,Mar);% reordered analytical model parameters
98 Kred = T'*Kar*T;% reduced stiffness matrix
99 Mred = T'*Mar*T;% reduced mass matrix
100
101 Korthog_f = Psir'*Kar*Psir;% modal stiffness matrix(full space)
102 Morthog_f = Psir'*Mar*Psir;% modal mass matrix(full space)
103 Korthog_r = Psim'*Kred*Psim;% modal stiffness matrix(reduced space)
104 Morthog_r = Psim'*Mred*Psim;% modal mass matrix(reduced space)
105
106 figure('Name','Orthogonalities, nphi<nm')
107 subplot(141)
108 bar3(Korthog_f); title('\Psi_N*K_N*\Psi_N')
109 hm(1) = subplot(142);
110 bar3(Morthog_f); title('\Psi_N*M_N*\Psi_N'); zlim([0 1])
111 subplot(143)
112 bar3(Korthog_r); title('\Psi_m*K_r*\Psi_m')
113 hm(2) = subplot(144);
114 bar3(Morthog_r); title('\Psi_m*M_r*\Psi_m'); zlim([0 1])
115
116 [Phir, Wnr] = eig(Kred,Mred);% solve reduced system
117
118 figure('Name','reduced matrices, nphi<nm')
119 subplot(121)
120 bar3(diag([Wna(8,8),Wna(7,7),Wna(1,1)]))
121 subplot(122)
122 bar3(Wnr)
123
124 Phi_f = T*Phir;% Expand reduced eigensolution to full space
125
126 MAC = zeros(size(Phi_f,2)-1,size(Psi_R,2));
127 for i = 1:size(Phi_f,2)-1
128     for j = 1:size(Psi_R,2)
129         MAC(i,j) = (Phi_f(:,i)'*Psi_R(:,j))^2/...
130             ((Phi_f(:,i)'*Phi_f(:,i))*(Psi_R(:,j)'*Psi_R(:,j)));
131     end
132 end
133 figure('Name','MAC, nphi<nm')
134 subplot(131)
135 bar3(MAC); title('MAC'); xlabel('\Phi_N'); ylabel('[T]\Phi_m')
136
137 subplot(132)
138 bar3(Psi_R(:,[1 7 8])*diag([1 1 -1]))
139 title('\Psi_N')
140 subplot(133)
141 bar3(Phi_f(:,1:3)*diag([-1 -1 1]))
142 title('[T]\Psi_m')

```


Correlation analysis evaluated for the BF

In this appendix the POC is evaluated using the analytical and estimated modal parameters of the BF, which were a result of section 3.5.2. The Matlab script in which the correlation analysis is evaluated can be found in `SystemCorrelation/BaseFrame`. The system contains 66 DOFs corresponding to the 22 sensor locations which are all measured in 3D. The first 28 modes, which were selected to simulate the FRM, are compared in the analysis.

FRAC and MAC matrices The corresponding FRFs are correlated using the FRAC and as to see in figure E.1a, the majority of the FRFs show clear correspondence. The diagonal is very close to 1 for the majority of the FRFs. The lowest correlation coefficient is equal to 0.9695 and the average value of the diagonal is 0.9988. So the modal model fits the analytical FRM with very high accuracy. The 28 mode shapes are compared using the

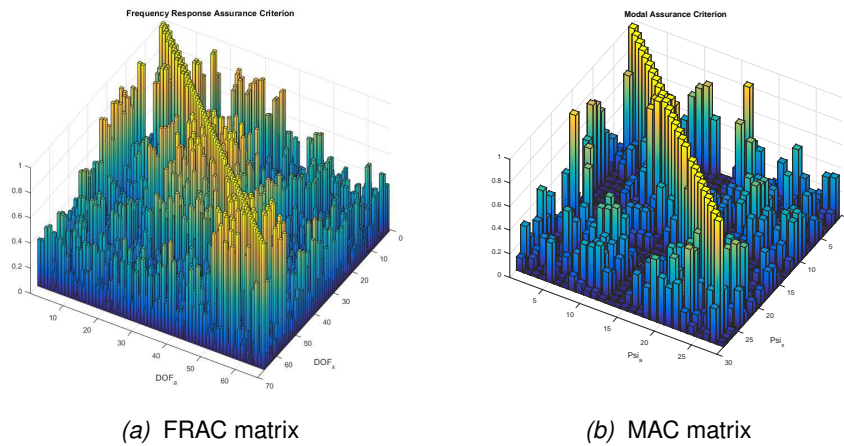


Figure E.1: FRAC matrices evaluated for estimation case 2

MAC and this matrix also approaches the unity matrix. The 28th mode is not present in the estimated since this resonance frequency(1247 Hz) is outside the FRI. Looking at the MAC

matrix shown in figure E.1b the following conclusions can be drawn about the quality of the fitted mode shapes; the 18th mode pair shows the lowest correlation value, which is equal to 0.699. The average value of the diagonal terms is equal to 0.9541. Furthermore mode shapes 9, 10 and 11 are very comparable, corresponding to resonance frequencies 783, 799 and 807 Hz respectively, but they all show the highest correlation at the diagonal. These two matrices confirm that the estimated modal model accurately represents the 'measured' dataset of the BF.

COMAC All coordinates show high COMAC values, the lowest value corresponding to DOF 44 is 0.7935 and the average COMAC value is equal to 0.9857. If the last mode is left out of the calculations the average is even higher, 0.9933 respectively.

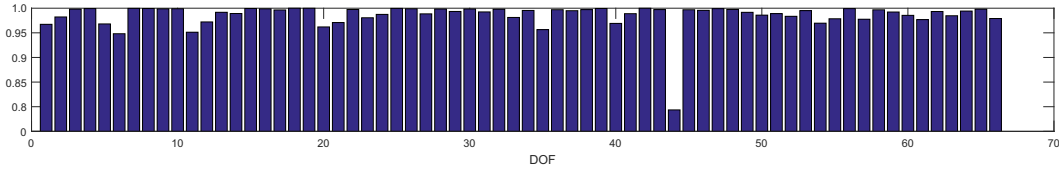


Figure E.2: COMAC values for estimation case 1

This confirms that the mode shape are estimated accurately for all DOFs.

ECOMAC and absolute difference The ECOMAC, shown in figure E.3a indicates that DOF 22 show the greatest deviations. Corresponding to node 22 in X-direction. That is also indicated by the

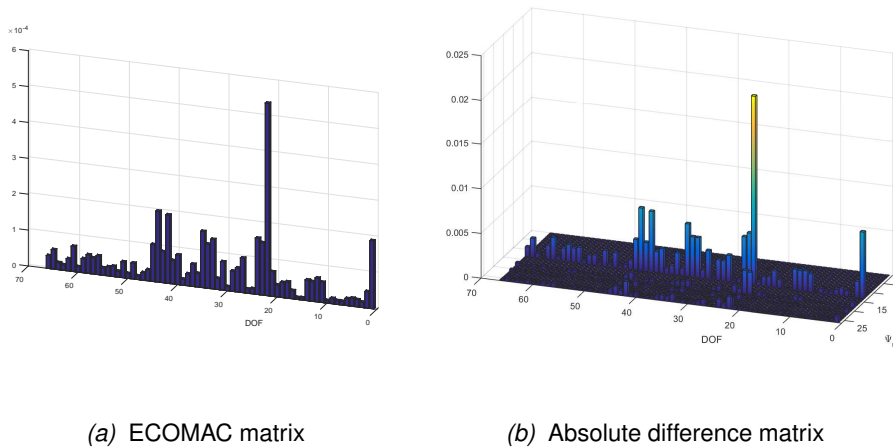


Figure E.3: Correlation matrices evaluated for estimation case 2

The POCs were not evaluated, since the global structural matrices are unknown. The FEM is built in NX Nastran and time was found to export these matrices to Matlab. For

further investigation the extration of these matrices must be investigated.

E.0.1 Conclusions

The correlation metrics clearly indicate that the estimated modal model represents the 'measured' FRM. So if the measurements are done correctly, the algorithm will be able to correctly identify the modal parameters of these FRFs.