

POLITECNICO DI TORINO

Corso di Laurea Magistrale
in Ingegneria Aerospaziale

Tesi di Laurea Magistrale

**Including camera-based measurements
in Frequency-based Substructuring**



Supervisors

prof. Zucca Stefano

prof. Rixen J. Daniel

Ms.Sc. Ahmed El Mahmoudi

Candidate

Quaranta Marina 263312

Aprile 2021

*"When considering the consequences of not doing the little things,
you realize there are no little things."*

Brad Stevens

Abstract

Dynamic Substructuring make it possible to obtain structures behavior starting from model of their parts.

The purpose of this study is to analyze a new methodology for Experimental Dynamic Substructuring to experimentally model a structure by using high-speed camera (HSC) measurements.

Experimental substructuring can be performed in different ways, but this thesis focuses on frequency based substructuring (FBS) and on finding an appropriate way to include photogrammetric data (camera measurements) to couple substructures.

Starting from experimental camera and accelerometers measurements performed on two substructures excited by hammer impacts, the idea of this thesis is to use information of a high number of points, combined with accelerometers information through a mixing method (System Equivalent Model Mixing) to benefit from the advantages of both methods.

High-speed cameras generate a new opportunity for substructuring because each pixel can be seen as an independent sensor, so camera measurements provide a full field representation of the interfaces.

The aim of this work is to elaborate this huge amount of data, to post process and combine them with neat accelerometers data through a software code, in order

to get a more accurate frequency response functions (FRF) to enhance substructures coupling.

A comparison between experimental results obtained by this new approach and a numerical coupling will show how high-speed camera measurement can be included in the methodology of Frequency based substructuring.

Index

Abstract	ii
Index	iv
List of figures.....	vii
Chapter 1 Introduction.....	9
1.1 Motivation.....	9
1.2 Structure of the thesis.....	11
Chapter 2 State of the Art	13
2.1 Dynamic of a substructure.....	13
2.2 Interface conditions	18
2.2.1 Compatibility	19
2.2.2 Forces equilibrium.....	20
2.3 Coupling and Decoupling – Lagrange multiplier FBS.....	22
2.3.1 Primal and Dual Assembly formulation	23
2.3.2 Structures coupling	25

2.3.3	Structures decoupling	26
2.4	System Equivalent Model Mixing	28
Chapter 3 Experimental Substructuring.....		33
3.1	Basics of Experimental FBS.....	33
3.2	Virtual Point Transformation.....	35
3.2.1	Discrete interface problem	35
3.2.2	VP formulation.....	36
3.2.3	Substructures coupling by VPT	38
3.3	Basics on measurement techniques	40
3.3.1	Data acquisition – Comparison between sensors	40
3.3.2	Displacements from digital images – High-speed camera	41
3.3.3	Gradient-based Optical Flow Methods	42
Chapter 4 Preliminary Analysis on tower structure.....		46
4.1	Tower set up	47
4.2	Results of the preliminary analysis	49
Chapter 5 Analysis on complex structure		52
5.1	Experimental setup	53
5.1.1	Structure and camera arrangement	53
5.1.2	Raw data acquisition	55
5.2	FRFs and SEMM method	63

5.3 Results.....	66
Chapter 6 Discussion and future development.....	69
Chapter 7 Summary.....	72
Appendix A	74
Extract of used code.....	74
Example of loading video and getting displacements.....	74
Animation of the video.....	76
Example of creation of FFT matrix	77
Upload camera FRF from matlab code and accelerometer FRF	78
Example of projecting accelerometers data to pixels through VPT	79
Application of SEMM method	80
Bibliography	81
Ringraziamenti.....	83

List of figures

2. 1 SEMM schematic representation.....	29
4. 1 Drawing of the preliminary test	46
4. 2 Tower set up.....	47
4. 3 Selected points for LK (left) and SOF (right) methods.....	48
4. 4 Fast Fourier Transformation of displacements obtained by SOF (above) and by LK (below).....	49
4. 5 Receptances for displacements obtained by SOF (above) and by LK (below)	50
4. 6 Comparison between camera and accelerometers data.....	51
5. 1 Drawing of the two separate substructures (left) and of the coupled structure (right).....	52
5. 2 Photograph of the experimental setup for XY plane measurements	53
5. 3 Selected Pixel for SOF - plan XY, YZ, XZ	57
5. 4 Selected Pixel for LK - plan XY, YZ, XZ	57
5. 5 Response to the third impact at the 4th location for the SOF pixels (Above) and for the LK pixels (below)	58

5. 6 Response to the first impact at the 9th location for the SOF pixels (Above) and for the LK pixels (below)	59
5. 7 Fast Fourier Transformation of responses in Y direction	61
5. 8 Fast Fourier Transformation for 1 pixels of the chessboard pattern (above) and 1 pixel of the gradient pattern (below) in y direction – impact H43	62
5. 9 Fast Fourier Transformation for 1 pixels of the chessboard pattern (above) and 1 pixel of the gradient pattern (below) in Z direction – impact H91	62
5. 10 Receptance of the 5th pixel (above) compared to acceleration of one of the accelerometer (below)	64
5. 11 Comparison of the corrupted video (7th impact, right) and the normal video (1st impact, left).....	67

Chapter 1

Introduction

In engineering, the idea of dividing a system in small subparts plays an important role in understanding the system itself. Dynamic Substructuring helps to solve structural problems in many fields, from aeronautics to mechanics, and in recent years interest in this subject has exponentially grown thanks to the development and availability of new technologies for acquire information about structures dynamic.

Nowadays, however, obtaining experimental information about substructures is not completely free from mistakes and bias errors, and every sensor used, from accelerometers to laser vibrometers, has drawbacks.

1.1 Motivation

The dynamic of a structure has been studied since it was understood that many problems arise from continuous excitement over time and from excitation at particular frequency.

Introduction

To obtain an accurate description of the phenomenon, experimental data are required, and to get them different kind of sensors are used, instruments to transform displacements and accelerations of substructures into numbers.

Accelerometers are the most common sensors but have the big disadvantage of acquire data for a limited number of points on the structure and add mass to the system.

Among the many alternative sensors that exist [1], one of them became the focus of this thesis: the high-speed camera. The reason for the choice is the novelty of the instrument, and its potentiality.

When a structure is filmed by the high-speed camera, each pixel can provide information about the structure displacement, so information about a huge number of points are provided. The camera has been a bit ignored so far because the video and the data are very heavy, and in the past, there wasn't enough computational resources to manage them.

Nowadays, with the increase in technological progress, this instrument acquires new interest from a research point of view, because it's easier to manage big amount of data with a decent computational cost.

To be able to get information on many points allows to have smaller errors and a description of the structure that is almost continuous, and this is a big advantage in dynamic substructuring, because in the coupling procedure it's important to have a full field description of the interface, to reduce errors propagation.

Considering the studies so far, usually results obtained from processing high speed camera data are quite noisy [2], so the aim of this work is to develop a proper way to reduce this noise, or to overcome it and effectively use high speed camera data to couple substructures.

1.2 Structure of the thesis

In order to do so, the purpose of this thesis is to develop a code which can combine all the information that a high-speed camera can provide with accelerometers data, that are very precise. By doing so, a good description of the whole system and especially of the interfaces is obtained, and an appropriate procedure for effectively coupling substructures can be developed.

One of the biggest advantages of this procedure is that using high speed camera and accelerometers data combined together, the final model is built without using a numerical model, and that's very important for situation where a numerical description of the system is not available.

1.2 Structure of the thesis

This work is built on two main part. The first one includes Chapter 2 and Chapter 3. Chapter 2 is about the theory of Dynamic Substructuring, it analyzes features in general and all principles behind the substructuring, from the requirements for coupling to the SEMM principles.

Chapter 3 instead is more about the experimental part of Dynamic Substructuring. It describes the practical application of Chapter 2 concepts, from the sensors that can be used to get displacements to the Virtual Point Transformation to get information about interfaces or points that are not described from physical sensors.

Then, the second part of this work concerns the actual experiments that are carried out: Chapter 4 describes preliminary tests, performed on a simple tower structure to understand the behavior of the camera and to get familiarity with the tools.

Introduction

The crucial part of the work is analyzed in Chapter 5, where a complex structure is described. Here a detailed discussion is conducted about how measurements and data are acquired and about how displacements, Fast Fourier Transformation and Frequency Response Functions are calculated. A further deepening is carried out about the SEMM method application. All the methodology and procedure are dissected, from the measurements to the post process part and the main results are reported.

In the last Chapter, all the main features and conclusion of this thesis are summed up and collected to give a perspective on the future of this preliminary study, in order to make the results accessible to continue on this track and demonstrate that high-speed camera measurements can be effectively included in the Frequency Based Substructuring methodology.

Chapter 2

State of the Art

2.1 Dynamic of a substructure¹

Dynamic Substructuring is an engineering concept to determine an accurate dynamic description of an assembly based on the dynamic properties of its sub-components, obtained from numerical modelling or from experimental measurements.

Substructuring techniques born in early 60s, to sub-dividing the mesh of a finite element model in order to reduce the computational cost of large problems, impossible to solve with the CPU power at that time.

Nowadays, DS offers many advantages over and above the computational efficiency, that led to huge interest in this field, including:

¹ The content of this section and the next four ones can also be found in a more detailed and extended form in [5], [8].

- Models of subsystems can be combining. Different components of the same project can be modeled separately and then coupled to meet interfaces requirements. During optimization, one can modify only the interested parts, and when it is not possible to test the full system one can experimentally characterize the single part.
- Articulated systems can be divided in substructures of different complexity: complex parts can be modeled more in detail than the rest, improving the analysis efficiency and speed.
- It is possible to couple substructures modeled in different domains, and to create hybrid models combining experimental models and numerical models.
- Great possibility to optimize analysis and design of systems.

The dynamic of structures can be expressed by several domains. Since all formulations are mathematically equivalent, it is possible to create hybrid models and to choose the domain that best suit the problem, according to the expected results, the computational and experimental resources and more. However, it is important to note that the formulation chosen to express the problem leads to different numerical and experimental approaches and approximation methods.

Considering a discrete and linearized problem, the most straightforward way to describe the dynamic of a substructure is to consider displacements as unknowns:

$$\mathbf{M}^{(s)}\ddot{\mathbf{u}}^{(s)} + \mathbf{C}^{(s)}\dot{\mathbf{u}}^{(s)} + \mathbf{K}^{(s)}\mathbf{u}^{(s)} = \mathbf{f}^{(s)} + \mathbf{g}^{(s)} \quad (2.1)$$

Dynamic of a substructure

Where M , C , K are respectively mass, damping and stiffness linearized matrices, the superscript (s) refers the equation to a given substructure s . f and g are the force vectors, divided by external force and interfaces forces.

This equation is obtained by an equilibrium between external forces and internal inertia, damping and elasticity forces and expresses the problem in physical and time domain, because the behavior of the substructure is expressed by the displacements at nodes over time.

Since the matrices M , C , K are usually generated by finite element software, they could be too detailed and include a huge number of DoF, which make the dynamic analysis unnecessarily complex.

To reduce the equation order, it could be useful to use as unknowns a combination of vectors of a subspace and use a reduced domain, such as modal domain.

In the modal domain, the behavior of the substructure is described by its natural vibration modes, obtained by the solution of the eigenvalue problem:

$$\left(\mathbf{K}^{(s)} - \omega_i^{(s)2} \mathbf{M}^{(s)} \right) \boldsymbol{\phi}_i^{(s)} = 0 \quad i = 1, \dots, n \quad (2.2)$$

In this equation, $\boldsymbol{\phi}_i^{(s)}$ are the eigenmodes of the structure, referred to the correspondent eigenfrequencies $\omega_i^{(s)}$. The subscript i refers to the considered mode and in the extended model there are as many eigenmodes as DoF in the system. Since the modes at lowest frequency better describe the behavior of the substructure, one can choose to represent the dynamics in an approximate way, discarding the latest modes and reducing the matrices size, considering $m < n$ eigenmodes.

In this domain, physical displacements are replaced by modal coordinates η , still in time domain, that represent the amplitudes of the modal component of the response. In matrix form:

$$\mathbf{u}^{(s)} = \mathbf{\Phi}^{(s)} \boldsymbol{\eta}^{(s)} \quad (2.3)$$

Here, $\mathbf{\Phi}^{(s)}$ is an orthogonal and mass-normalised matrix, that include the eigenmodes ordered by increasing eigenfrequency.

Substituting this change of variable in (2.1), the differential equation become:

$$\mathbf{M}_m^{(s)} \ddot{\boldsymbol{\eta}}^{(s)} + \mathbf{C}_m^{(s)} \dot{\boldsymbol{\eta}}^{(s)} + \mathbf{K}_m^{(s)} \boldsymbol{\eta}^{(s)} = \mathbf{f}_m^{(s)} + \mathbf{g}_m^{(s)} \quad (2.4)$$

The subscript m indicates the new matrices in the transformed and reduced space.

Recalling the mass and the stiffness orthogonality, this system includes m single DoF equations, so it's very efficient to solve.

There are different techniques to reduce the extended system in modal domain, but there are also other domains to reduce the computational cost, such as the State-space domain that describes the substructure behavior by a first order representation, even if the total equations are doubled.

As already mentioned above, there's not a single choice of the right domain, but several parameters in the research context that leads to choose one domain over another.

The above equations are differential equations in time, discretized in space. Sometimes it could be useful to discretized also time. Choosing finite differences

Dynamic of a substructure

methods makes the problem changes from differential equations to algebraic equations, solvable in a recursive form.

Alternatively, time dependency can also be decomposed, drawing an analogy with the modal approach, expressing the response into a combination of several components, for examples using Fourier or Laplace transformation. Fourier decomposition allows to analyze the system in the frequency domain and to study the structure's response to harmonic loads.

Applying the Discrete Fourier decomposition on the coordinates and on force vectors, considering a limited number of frequencies:

$$\mathbf{u}^{(s)}(t) = \sum_{k=-N_\omega}^{N_\omega} \bar{\mathbf{u}}^{(s)}(\omega) e^{-j\omega_k t} \bar{\mathbf{f}}^{(s)}(\omega) = \frac{1}{2\pi} \sum_{k=-N_\omega}^{N_\omega} \mathbf{f}^{(s)}(t) e^{j\omega_k t} \quad (2.5)$$

and substituting in (2.1), the dynamic equation loses its time dependence:

$$(-\omega^2 \mathbf{M}^{(s)} + j\omega \mathbf{C}^{(s)} + \mathbf{K}^{(s)}) \bar{\mathbf{u}}^{(s)} = \bar{\mathbf{f}}^{(s)} + \bar{\mathbf{g}}^{(s)} \quad (2.6)$$

The bracketed term is called *dynamic stiffness matrix* $\mathbf{Z}(\omega)$ and represents the force needed to generate a displacement on the considered DoF. If the equation is written for velocity or accelerations instead of displacements, the respective matrix is called *mechanical impedance* or *apparent mass*.

The inverse of the matrix $\mathbf{Z}(\omega)$ is more intuitive, because represents the displacement of a DoF excited by a harmonic force and is called *receptance matrix* $\mathbf{Y}(\omega)$ if the unknowns are displacements, *mobility* or *accelerance* if the unknowns are velocity or accelerations respectively.

Using $\mathbf{Y}(\omega)$, the equation can be written in the form:

$$\bar{\mathbf{u}}^{(s)} = \mathbf{Y}^{(s)}(\bar{\mathbf{f}}^{(s)} + \bar{\mathbf{g}}^{(s)}) \quad (2.7)$$

$$\text{where } \mathbf{Y}^{(s)}(\omega) = \mathbf{Z}^{(s)}(\omega)^{-1} = (-\omega^2 \mathbf{M}^{(s)} + j\omega \mathbf{C}^{(s)} + \mathbf{K}^{(s)})^{-1} \quad (2.8)$$

The elements in $\mathbf{Y}(\omega)$ are the *frequency response functions* (FRFs) and could be obtained directly from experimental test, measuring the displacements following applied forces.

Starting from the physical domain, several alternative domains can be formulated to describe the dynamic of a substructure and it is possible to shift from one domain to another. If measurements are carefully performed, both numerical and experimental approach could be efficient to determine the behavior of the substructure.

Since this work is supposed to improve experimental substructuring, and tests are based on hammer impacts, physical and frequency domains are considered from this point on.

2.2 Interface conditions

Until now, the discussion concerned a single substructure. However, DS is about achieving the behavior description of an assembly, so this work have to deal with coupling substructures.

Interface conditions

In order to do that, matching DoF on interfaces must represent the same motion (usually one direction displacements) on interfaces and must satisfy two conditions: compatibility and forces equilibrium.

First of all, the dynamic of the assembly can be described by a block matrix that includes equations for each substructure in the system:

$$\mathbf{Z}\bar{\mathbf{u}} = \bar{\mathbf{f}} + \bar{\mathbf{g}} \quad (2.9)$$

$$\mathbf{Z} = \begin{bmatrix} \mathbf{Z}^{(1)} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{Z}^{(N_{sub})} \end{bmatrix}$$
$$\bar{\mathbf{u}} = \begin{bmatrix} \bar{\mathbf{u}}^{(1)} \\ \vdots \\ \bar{\mathbf{u}}^{(N_{sub})} \end{bmatrix} \quad \bar{\mathbf{f}} = \begin{bmatrix} \bar{\mathbf{f}}^{(1)} \\ \vdots \\ \bar{\mathbf{f}}^{(N_{sub})} \end{bmatrix} \quad \bar{\mathbf{g}} = \begin{bmatrix} \bar{\mathbf{g}}^{(1)} \\ \vdots \\ \bar{\mathbf{g}}^{(N_{sub})} \end{bmatrix} \quad (2.10)$$

Since the system is still uncoupled, g forces are unknown.

2.2.1 Compatibility

The first condition, coordinate compatibility, assures that displacements of matching DoF have same value on both sides of the interfaces.

Considering the DoF on interfaces (boundary DoF), compatibility condition can be express as:

$$\bar{\mathbf{u}}_b^{(s)} - \bar{\mathbf{u}}_b^{(r)} = 0 \quad (2.11)$$

If s and r are two substructures with a common interface, subscript b refers to boundary and if the numbering of boundary DoF match across the interfaces.

To make notation as general as possible, a Boolean matrix \mathbf{B} is introduced, so the condition becomes:

$$\mathbf{B}\bar{\mathbf{u}} = 0 \quad (2.12)$$

In this way, the equation is valid for the whole system, because \mathbf{B} considers boundary DoF for each substructure and makes them match on respective interfaces.

2.2.2 Force equilibrium

The second condition is based on the actio-reactio Newton law and requires that internal force of an assembly must be zero. Since \mathbf{g} are forces between substructures, when the system is assembled their sum on interfaces must be zero:

$$\mathbf{g}_b^{(s)} + \mathbf{g}_b^{(r)} = 0 \quad (2.13)$$

$$\mathbf{g}_i^{(s)} = 0 \quad \mathbf{g}_i^{(r)} = 0 \quad (2.14)$$

Interface conditions

Also this time it's possible to introduce a Boolean matrix $\mathbf{L}^{(s)T}$ in order to make DoF correctly match, pairing DoF on interfaces and make internal forces null:

$$\mathbf{L}^T \bar{\mathbf{g}} = 0 \quad (2.15)$$

$\mathbf{L}^{(s)T}$ is a sparse localization matrix, and generalises the substructures' DoF in to a global set of DoF. This matrix is the transpose of a matrix that allow to get local DoF of each substructure from a set of global DoF:

$$\mathbf{u}^{(s)} = \mathbf{L}^{(s)} \mathbf{u}_g \quad (2.16)$$

Once defined that set of global DoF, it's important to notice a relation between \mathbf{B} and \mathbf{L} . Since the local sets derive from a unique global set, compatibility condition is automatically satisfied:

$$\mathbf{B} \mathbf{L} \mathbf{u}_g = 0 \quad \forall \mathbf{u}_g \quad (2.17)$$

From this equation appears that \mathbf{B} is the nullspace of \mathbf{L} and vice versa:

$$\begin{cases} \mathbf{L} = null(\mathbf{B}) \\ \mathbf{B}^T = null(\mathbf{L}^T) \end{cases} \quad (2.18)$$

that is a very useful feature of the system.

2.3 Coupling and Decoupling – Lagrange multiplier FBS

The dynamic of the system now can be written starting from substructures dynamic, imposing interfaces conditions, in a block matrices form:

$$\begin{cases} \mathbf{Z}\bar{\mathbf{u}} = \bar{\mathbf{f}} + \bar{\mathbf{g}} \\ \mathbf{B}\bar{\mathbf{u}} = \mathbf{0} \\ \mathbf{L}^T\bar{\mathbf{g}} = \mathbf{0} \end{cases} \quad (2.19)$$

This relation is the most general and straightforward way to express the problem, considering DoF and coupling forces specifically set for each substructure.

However, this formulation could lead to a computational expensive solution, since there are so many explicit interfaces conditions and unknowns. The problem could be easier solved with different approaches: using displacements as unknowns leads to the *primal assembly* approach. Instead, using interfaces forces as unknowns leads to the *dual assembly* approach.

The approach choice depends on the research context because each formulation can be used with different numerical and experimental techniques.

Coupling and Decoupling – Lagrange multiplier FBS

2.3.1 Primal and Dual Assembly formulation

Primal assembly formulation considers as primal unknowns the displacements of a global set of DoF, defined for the whole structure. From that, local sets of DoF are made for each substructure, using Eq. (2.16).

Hence, as mentioned above, compatibility condition is already satisfied, and the dynamic equations become:

$$\begin{cases} \mathbf{Z}\mathbf{L}\mathbf{u}_g = \bar{\mathbf{f}} + \bar{\mathbf{g}} \\ \mathbf{L}^T\bar{\mathbf{g}} = \mathbf{0} \end{cases} \quad (2.20)$$

Forces equilibrium can be included in the first equation pre-multiplying it with \mathbf{L}^T , so the last term of the equation become exactly the force equilibrium and drop out.

Now the primal assembly of the coupled problem can be expressed in the form:

$$\mathbf{Z}_g\mathbf{u}_g = \mathbf{f}_g \text{ with } \mathbf{Z}_g = \mathbf{L}^T\mathbf{Z}\mathbf{L} \text{ and } \mathbf{f}_g = \mathbf{L}^T\bar{\mathbf{f}} \quad (2.21)$$

The matrix \mathbf{Z}_g is a square matrix and is the *primal assembled impedance* for global coordinates.

Interface forces can be obtained after getting displacements, substituting them in the first line of Eq (2.20), namely:

$$\bar{\mathbf{g}} = \mathbf{Z}\mathbf{L}\mathbf{u}_g - \bar{\mathbf{f}} \quad (2.22)$$

Since this approach is written using impedance, it's suitable for situation where impedance of substructures can be obtained easily, such as from finite element modelling.

Situation where is more convenient produce admittances, dual assembly could be more efficient.

Contrary to primal assembly, *dual assembly formulation* considers as main unknowns interface forces, which are defined for each pair of interface DoFs:

$$\mathbf{g} = -\mathbf{B}^T \boldsymbol{\lambda} \quad (2.23)$$

$\boldsymbol{\lambda}$ are usually *Lagrange multipliers* and represent the intensities of the interface forces that generate coupling forces.

Recalling Eq. (2.18), namely $\mathbf{L}^T \mathbf{B}^T = \mathbf{0}$, it follows that interface equilibrium is satisfied a priori:

$$\mathbf{L}^T \mathbf{B}^T \boldsymbol{\lambda} = \mathbf{0} \quad \forall \boldsymbol{\lambda} \quad (2.24)$$

Now, equilibrium condition can be eliminated from Eq. (2.19) and the coupled problem can be expressed by the dual assembly formulation:

$$\begin{cases} \mathbf{Z}\bar{\mathbf{u}} + \mathbf{B}^T \boldsymbol{\lambda} = \bar{\mathbf{f}} \\ \mathbf{B}\boldsymbol{\lambda} = \mathbf{0} \end{cases} \Leftrightarrow \begin{bmatrix} \mathbf{Z} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{u}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{f}} \\ \mathbf{0} \end{bmatrix} \quad (2.25)$$

2.3.2 Structures coupling

Since dual assembly is often used when admittances are available, it's useful to write the admittance notation of the problem, solving displacements in function of interface forces:

$$\begin{cases} \bar{\mathbf{u}} = \mathbf{Y}(\bar{\mathbf{f}} - \mathbf{B}^T \boldsymbol{\lambda}) \\ \mathbf{B}\bar{\mathbf{u}} = \mathbf{0} \end{cases} \quad (2.26)$$

Substituting the first line in the second one, the equation become:

$$\mathbf{B}\mathbf{Y}(\bar{\mathbf{f}} - \mathbf{B}^T \boldsymbol{\lambda}) = \mathbf{0} \Leftrightarrow (\mathbf{B}\mathbf{Y}\mathbf{B}^T)\boldsymbol{\lambda} = \mathbf{B}\mathbf{Y}\bar{\mathbf{f}} \quad (2.27)$$

This formulation is called *Dual Interface Problem* and can be solved in term of interface forces $\boldsymbol{\lambda}$.

This approach is very common in experimental substructuring because the dynamic of the structure can be obtained collecting the admittance of the substructures.

When $\boldsymbol{\lambda}$ are found, the dynamic displacements of the structure can be calculated, substituting $\boldsymbol{\lambda}$ in first line of Eq. (2.26):

$$\bar{\mathbf{u}} = \mathbf{Y}\bar{\mathbf{f}} - \mathbf{Y}\mathbf{B}^T(\mathbf{B}\mathbf{Y}\mathbf{B}^T)^{-1}\mathbf{B}\mathbf{Y}\bar{\mathbf{f}} \quad (2.28)$$

Collecting the forces vector, the FRF that remains is called *dually assembled admittance*:

$$\bar{\mathbf{u}} = \tilde{\mathbf{Y}}\bar{\mathbf{f}}$$

$$\text{where } \tilde{\mathbf{Y}} = \mathbf{Y} - \mathbf{Y}\mathbf{B}^T(\mathbf{B}\mathbf{Y}\mathbf{B}^T)^{-1}\mathbf{B}\mathbf{Y} = [\mathbf{I} - \mathbf{Y}\mathbf{B}^T(\mathbf{B}\mathbf{Y}\mathbf{B}^T)^{-1}\mathbf{B}]\mathbf{Y} \quad (2.29)$$

The first term in $\tilde{\mathbf{Y}}$ is the uncoupled FRF \mathbf{Y} , the second term represents the coupling part: $\mathbf{B}\mathbf{Y}$ refers to the force uncoupled responses, and $(\mathbf{B}\mathbf{Y}\mathbf{B}^T)^{-1}\mathbf{B}\mathbf{Y}$ are the Lagrange multipliers.

Eq. (2.29) take the name of *one-line equation for Lagrange multiplier frequency based substructuring* (LM-FBS). This method allows to predict the dynamic behavior of a structure starting from the substructure's admittances, once that local DoF (divided by boundary and internal) are connected with global applied forces and global DoF set.

LM-FBS is a global problem, because take into account that interfaces forces affect the whole structure, so it considers interface DoF for each substructure. That makes some lines redundant, since DoF on matching interface are counted twice. These redundancies should be eliminated in order to make the method even more efficient.

2.3.3 Structures decoupling

The main purpose of this thesis is about test and improve a coupling approach, so it does not deal directly with structure decoupling. Nevertheless, the work will involve a SEMM method, so it is necessary to know basic principles of decoupling.

Inverse decoupling will not be treated here.

Coupling and Decoupling – Lagrange multiplier FBS

Substructure decoupling aimed to characterize the dynamic of a substructure, knowing the dynamic of the other substructures and the behavior of the whole assembly.

Considering an assembly of two substructures, A and B, the decoupling formulation is based on the fact that the response to a force f_i^B of a B's DoF when B is disassembled can be obtained by exciting the assembled system with the same force f_i^B together with another force that cancels the effect of coupling, namely the opposite force that B receives from the A interface.

To obtain this force, it can be imposed to the A substructure on A-B interface the same displacements that B would cause when the assembled system was excited.

Transforming these words in equations, considering i_A and i_B internal DoFs respectively in A and B part and the interface DoF as boundary DoF, the decoupling problem can be expressed in the form:

$$\mathbf{Z}^B \mathbf{u}^B = \mathbf{f}^B \quad (2.30)$$

$$\begin{aligned} \mathbf{Z}^{AB} \mathbf{u}^{AB} &= \mathbf{f}^{AB} + \mathbf{g}^{AB} \\ \Downarrow \\ \begin{bmatrix} \mathbf{Z}_{i_A, i_B}^{AB} & \mathbf{Z}_{i_A, b}^{AB} & \mathbf{0} \\ \mathbf{Z}_{b, i_A}^{AB} & \mathbf{Z}_{b, b}^{AB} & \mathbf{Z}_{b, i_B}^{AB} \\ \mathbf{0} & \mathbf{Z}_{i_B, b}^{AB} & \mathbf{Z}_{i_B, i_B}^{AB} \end{bmatrix} \begin{bmatrix} \mathbf{u}_{i_A}^{AB} \\ \mathbf{u}_b^{AB} \\ \mathbf{u}_{i_B}^{AB} \end{bmatrix} &= \begin{bmatrix} \mathbf{0} \\ \mathbf{f}_b^{AB} = \mathbf{f}_b^B \\ \mathbf{f}_{i_B}^{AB} = \mathbf{f}_i^B \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\lambda} \\ \mathbf{0} \end{bmatrix} \end{aligned} \quad (2.31)$$

$$\text{where } \boldsymbol{\lambda} \text{ such that } \mathbf{Z}^A \begin{bmatrix} \mathbf{u}_i^A \\ \mathbf{u}_b^A = \mathbf{u}_b^{AB} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\lambda} \end{bmatrix} \quad (2.32)$$

In order to make matching the numbering of interfaces of different substructures, Boolean matrices \mathbf{B}^A and \mathbf{B}^{AB} are introduced and the decoupling relation become:

$$\begin{bmatrix} \mathbf{Z}^{AB} & \mathbf{0} & \mathbf{B}^{AB} \\ \mathbf{0} & -\mathbf{Z}^A & \mathbf{B}^{AB} \\ \mathbf{B}^{AB} & \mathbf{B}^A & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u}^{AB} \\ \mathbf{u}^A \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{f}^B \\ \mathbf{0} \end{bmatrix} \quad (2.33)$$

But if numbering of interfaces matches, the impedance of the uncoupled structure B can be easily found, solving Eq (2.31) in impedance form:

$$\mathbf{Y}^{(B)} = \begin{bmatrix} \mathbf{Y}_{b,b}^B & \mathbf{Y}_{b,i}^B \\ \mathbf{Y}_{i,b}^B & \mathbf{Y}_{i,i}^B \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_{b,b}^{AB} & \mathbf{Y}_{b,i_B}^{AB} \\ \mathbf{Y}_{i_B,b}^{AB} & \mathbf{Y}_{i_B,i_B}^{AB} \end{bmatrix} - \begin{bmatrix} \mathbf{Y}_{b,b}^{AB} \\ \mathbf{Y}_{i_B,b}^{AB} \end{bmatrix} (\mathbf{Y}_{b,b}^{AB} - \mathbf{Y}_{b,b}^A)^{-1} [\mathbf{Y}_{b,b}^{AB} \quad \mathbf{Y}_{b,i_B}^{AB}] \quad (2.34)$$

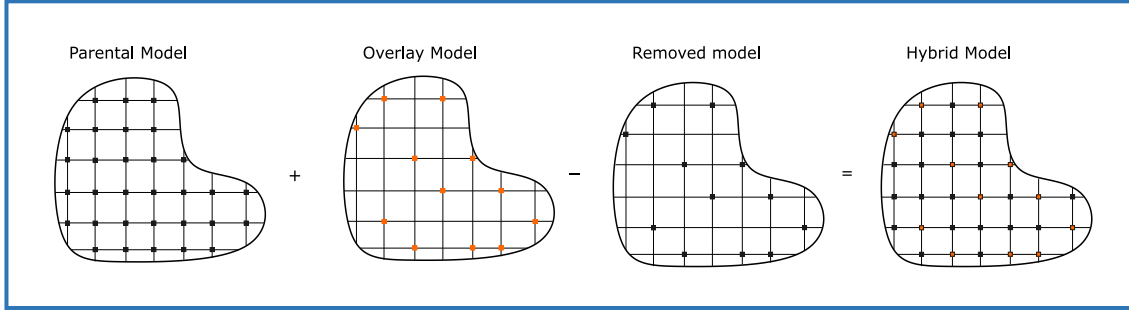
This equation shows as substructure decoupling is similar to substructure coupling, if considering A having a negative impedance.

2.4 System Equivalent Model Mixing

Now that coupling and decoupling principles are been described, system equivalent model mixing is introduced.

In DS field, models involved are imperfect: numerical models give a lot of information and are very “clean”, but they are based on assumptions and are always approximations of reality; on the other hand, experimental models

System Equivalent Model Mixing



2. 1 SEMM schematic representation

represent the reality as it is, but give information for a limited number of points and can be clouded by noise or bias errors.

For this reason, system equivalent model mixing is born: it is a quite new methods that make it possible to combine models of different nature to get an hybrid model [3]. Usually, it combines experimental and numerical model, but in this work, it will be used with different types of experimental measurements.

This method can be seen as an expansion method, because allow to extend the dynamic of a small system to a bigger set of DoF.

In SEMM there are three different models involved that lead to the final hybrid model: the parental model is the model that give the size to the final hybrid model, but its dynamic will be discarded. The removed model has the same size, or it is a subset of the parental model and is intended to remove the parental dynamic that will be replaced by the dynamic of the overlay model. Thus, the overlay model supplies the dynamic of the final hybrid model and is usually a subset of boundary DoF of the parental model.

In this thesis removed model is chosen of the same size of the parental model, in order to remove all the parental dynamic, avoiding that internal modes could affect the results and create spurious peak. Thus, considering the subscript i for

internal DoF and the subscript b for boundary ones, the three models involved are expressed in the form:

$$\mathbf{Y}^{par} \triangleq \begin{bmatrix} \mathbf{Y}_{ii} & \mathbf{Y}_{ib} \\ \mathbf{Y}_{bi} & \mathbf{Y}_{bb} \end{bmatrix}^{par} = \mathbf{Y}^{rem}; \mathbf{Y}^{ov} \triangleq [\mathbf{Y}_{bb}]^{ov}$$

This method is based on LM FBS and mathematically consists on coupling parental and overlay dynamic (through impedance or admittance) and decoupling the removed one, as shown schematically in Figure 2.1.

The equation of motion of the uncoupled model is simply [4]:

$$\mathbf{u} = \mathbf{Y}(\mathbf{f} + \mathbf{g}) \text{ where } \mathbf{Y} = \begin{bmatrix} \mathbf{Y}^{par} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{Y}^{rem} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Y}^{ov} \end{bmatrix}; \mathbf{f} = \begin{bmatrix} \mathbf{f}^{par} \\ \mathbf{f}^{rem} \\ \mathbf{f}^{ov} \end{bmatrix}; \mathbf{g} = \begin{bmatrix} \mathbf{g}^{par} \\ \mathbf{g}^{rem} \\ \mathbf{g}^{ov} \end{bmatrix} \quad (2.35)$$

The admittance of the hybrid model can be obtained in two steps [3].

First, an intermediate admittance, \mathbf{Y}^Δ , is created by decoupling the overlay model and the removed model:

$$\mathbf{Y}^\Delta = \mathbf{Y}^{rem}(-)\mathbf{Y}^{ov} \quad (2.36)$$

Recalling LM FBS for decoupling, namely Eq (2.34), this delta model can be written as:

$$\mathbf{Y}^\Delta = \mathbf{Y}^{rem} - \begin{bmatrix} \mathbf{Y}_{i,b}^{rem} \\ \mathbf{Y}_{b,b}^{rem} \end{bmatrix} (\mathbf{Y}_{b,b}^{rem} - \mathbf{Y}_{b,b}^{ov})^{-1} [\mathbf{Y}_{i,b}^{rem} \quad \mathbf{Y}_{b,b}^{rem}] \quad (2.37)$$

The second step consists in coupling the parental model with the delta model, considering that now the delta model shares all the DoF with the parental model:

System Equivalent Model Mixing

$$\mathbf{Y}^{SEMM} = \mathbf{Y}^{par} - \mathbf{Y}^{par}(\mathbf{Y}^{par} - \mathbf{Y}^{\Delta})^{-1}\mathbf{Y}^{par} \quad (2.38)$$

Substituting Eq. (2.37) in Eq. (2.38), considering that for this case $\mathbf{Y}^{par} = \mathbf{Y}^{rem}$ the equation changes in:

$$\mathbf{Y}^{SEMM} = \mathbf{Y}^{par} - \mathbf{Y}^{par} \left(\begin{bmatrix} \mathbf{Y}_{i,b}^{rem} \\ \mathbf{Y}_{b,b}^{rem} \end{bmatrix} (\mathbf{Y}_{b,b}^{rem} - \mathbf{Y}_{b,b}^{ov})^{-1} [\mathbf{Y}_{i,b}^{rem} \quad \mathbf{Y}_{b,b}^{rem}] \right)^{-1} \mathbf{Y}^{par} \quad (2.39)$$

In the end, making an inversion of the term between brackets, the SEMM admittance become:

$$\mathbf{Y}^{SEMM} = \mathbf{Y}^{par} - \mathbf{Y}^{par} \begin{bmatrix} \mathbf{Y}_{i,b}^{rem} \\ \mathbf{Y}_{b,b}^{rem} \end{bmatrix}^+ (\mathbf{Y}_{b,b}^{rem} - \mathbf{Y}_{b,b}^{ov}) [\mathbf{Y}_{i,b}^{rem} \quad \mathbf{Y}_{b,b}^{rem}]^+ \mathbf{Y}^{par} \quad (2.40)$$

In this way, the hybrid model is created avoiding the inversion of the overlay method, advantageous when the overlay model is an experimental one.

Chapter 3

Experimental Substructuring

3.1 Basics of Experimental FBS

Dynamic Substructuring has been described in the previous chapter to mathematically coupling different substructures in order to get the dynamic behavior of a system.

One of the most important advantage of Experimental substructuring is the chance to create a model without considering internal features. In numerical modelling, in fact, to create a model of the interface it is mandatory to have all the dynamic information from both internal and external parts. For experimental modelling, instead, the impedance measured on the interface is a response of the whole structure, so implicitly includes the dynamic of internal parts. [5]

However, even if the problem can be expressed in several equivalent ways, getting parameters from experimental measurements is not so straightforward.

Experimental Substructuring

Most challenges lie in the limitation about getting impedances for each DoF and collecting them in a *Function Response Matrix* to identify the system specifications [6].

In fact, measurements are never perfect, and there are always many DoF missing, so computing the inverse of the FRFs to get stiffness information increases and amplifies errors.

Moreover, a complete dynamic description for coupling should include both translational and rotational DoF, but rotational DoF are quite problematic to acquire from experimental tests [7].

Even translational DoF can be difficult to acquire if there are bolts connections, weld or continuous surfaces, but one can approximate special connections considering interface as a few discrete points (*Single Point Connection*).

Several methods have been developed to overcome the rotational DoF missing, from modal expansion to finite difference approach. One of the most known is the *equivalent multi-point connection* (EMPC), which implicitly include rotational DoF by coupling many translational DoF on interfaces. This method, however, has the big disadvantage of make the interface problem overdetermined, because involve stringent compatibility condition [8].

3.2 Virtual Point Transformation

3.2.1 Discrete interface problem

In order to alleviate errors, an alternative could be to weaken the interface compatibility condition. That is, to impose compatibility only on the DoF that are meaningful for the structure behavior, leaving the others free. In doing so, the number of compatibility equations is significantly reduced compared to the DoF on the interface [8].

A way to do it is to project all the DoF information in a subspace of DoF, using an Interface Deformation Mode filtering, thus choosing just the modes of interest. The method can be called Interface Displacement Modes filtering when includes just the rigid mode and leaves the flexible ones decoupled.

Applying IDM to a single connection point, one can transform all the translational measured DoF to a single matrix of 6 DoF (3 translations and 3 rotations) referred to a single point of selected coordinates. The same approach can be used also for measured forces.

Since the final point doesn't need real measurements, this method can also be called *Virtual Point Transformation* and the resulting 6x6 matrix includes responses of the virtual point to virtual loads [9].

3.2.2 VP formulation²

To properly coupling the elements of a system, equilibrium and compatibility conditions are required.

In general form, the compatibility condition is expressed by matching all the DoF on interfaces:

$$\mathbf{B}\mathbf{u} = \mathbf{0} \quad (3.1)$$

Considering instead just the rigid modes, discarding the flexible ones, new compatibility equations can be introduced, in function of the interface deformation modes \mathbf{q} :

$$\mathbf{u} = \mathbf{R}\mathbf{q} + \boldsymbol{\mu} \quad \mathbf{q} \in \mathbb{R}^m \quad (3.2)$$

In this equation \mathbf{R} is the $n \times m$ *mode shape matrix* and includes the IDM in the columns, with $m < n$.

$\boldsymbol{\mu}$ is a displacement residual, due to $m < n$, and includes all the flexible modes or discarded modes of the system.

To find \mathbf{q} , a pseudo-inverse of \mathbf{R} can be applied, with the introduction of a symmetric weighting matrix \mathbf{W} to minimize the residual:

$$\mathbf{R}^T \mathbf{W} \boldsymbol{\mu} = \mathbf{0} \quad (3.3)$$

² This section and the next one are better described in [7].

Virtual Point Transformation

Now, pre-multiplying the compatibility equation Eq (3.2) by $\mathbf{R}^T \mathbf{W}$, one can find displacements projected onto the IDM subspace, solving in \mathbf{q} , but also the physical displacements of the DoF involved in the filter process $\tilde{\mathbf{u}}$:

$$\mathbf{R}^T \mathbf{W} \mathbf{u} = \mathbf{R}^T \mathbf{W} \mathbf{R} \mathbf{q}$$

$$\mathbf{q} = (\mathbf{R}^T \mathbf{W} \mathbf{R})^{-1} \mathbf{R}^T \mathbf{W} \mathbf{u} \quad (3.4)$$

$$\tilde{\mathbf{u}} = \mathbf{R}(\mathbf{R}^T \mathbf{W} \mathbf{R})^{-1} \mathbf{R}^T \mathbf{W} \mathbf{u} \quad (3.5)$$

To get the residual it's enough to calculate $\mathbf{u} - \tilde{\mathbf{u}}$.

Eq (3.4) and Eq (3.5) can be also expressed in a simpler form, introducing the matrix \mathbf{T} :

$$\mathbf{q} = \mathbf{T} \mathbf{u} \text{ with } \mathbf{T} = (\mathbf{R}^T \mathbf{W} \mathbf{R})^{-1} \mathbf{R}^T \mathbf{W} \quad (3.6)$$

$$\tilde{\mathbf{u}} = \mathbf{R} \mathbf{T} \mathbf{u} \quad (3.7)$$

More information about how to get the matrices \mathbf{R} and \mathbf{W} are in [5], [9].

A similar approach can be used on interface forces, considering $m < n$ virtual loads:

$$\mathbf{m} = \mathbf{R}^T \mathbf{F} \quad (3.8)$$

Also in this case, physical forces that generate virtual loads can be found with a weighted right inverse:

$$\begin{aligned} \tilde{\mathbf{f}} &= \mathbf{F} \mathbf{f} \text{ with } \mathbf{F} \triangleq \mathbf{T}^T \mathbf{R}^T = \mathbf{R} \mathbf{T} \\ \tilde{\mathbf{f}} &= \mathbf{T}^T \mathbf{m} \text{ with } \mathbf{T}^T \triangleq \mathbf{W} \mathbf{R} (\mathbf{R}^T \mathbf{W} \mathbf{R})^{-1} \end{aligned} \quad (3.9)$$

3.2.3 Substructures coupling by VPT

Now that a reduction of forces and displacements on the interface is been applied by filtering rigid modes, one can couple the system considering just the DoF which contribute to the filtered modes, leaving the others free.

That DoF on different interfaces do not necessarily match each other. Virtual points displacements instead match each other by definition, since there are 3 translational and 3 rotational DoF and coordinates are decided a priori.

Then, the coupling equations can be written as follow:

$$\begin{cases} \mathbf{u} = \mathbf{Y}(\mathbf{f} + \mathbf{g}) = \mathbf{Y}(\mathbf{f} - \mathbf{T}_f^T \mathbf{B}^T \boldsymbol{\lambda}) \\ \mathbf{B}\mathbf{q} = \mathbf{B}\mathbf{T}_u \mathbf{u} = \mathbf{0} \end{cases} \text{ with } \mathbf{B} \triangleq [-\mathbf{I} \ \mathbf{I}] \quad (3.10)$$

Here, \mathbf{T}_u is the same \mathbf{T} of Eq (3.6) and \mathbf{T}_f is the same of Eq (3.9). $\boldsymbol{\lambda}$ are the Lagrange multiplier describing the internal forces between virtual DoF.

With the LMFBS procedure, Eq (3.10) can be resolved in $\boldsymbol{\lambda}$:

$$\boldsymbol{\lambda} = (\mathbf{B}\mathbf{T}_u \mathbf{Y} \mathbf{T}_f^T \mathbf{B}^T)^{-1} \mathbf{B}\mathbf{T}_u \mathbf{Y} \mathbf{f} \quad (3.11)$$

Substituting $\boldsymbol{\lambda}$ in the first line of Eq (3.10), coupled interfaces displacements can be found:

$$\mathbf{u} = \mathbf{Y} \mathbf{f} - \mathbf{Y} \mathbf{T}_f^T \mathbf{B}^T (\mathbf{B}\mathbf{T}_u \mathbf{Y} \mathbf{T}_f^T \mathbf{B}^T)^{-1} \mathbf{B}\mathbf{T}_u \mathbf{Y} \mathbf{f} \quad (3.12)$$

Virtual Point Transformation

This equation is similar to Eq (2.28), but with the introduction of \mathbf{T}_f and \mathbf{T}_u matrices, responsible for the Virtual Point Transformation, and the \mathbf{B} matrices, that refer to the Virtual Point DoF.

These matrices can be grouped in:

$$\mathbf{B}_c = \mathbf{B}\mathbf{T}_u \quad (3.13)$$

$$\mathbf{B}_e = \mathbf{B}\mathbf{T}_f \quad (3.14)$$

Where the subscript c refers to compatibility and the subscript e refers to equilibrium condition at virtual point.

In this way the formulation become identical to Eq (2.28) and can be solved the coupling by LM FBS method, leaving the admittance matrix as it is.

Alternatively, a solution in terms of Virtual Point DoF can be obtained, substituting $\mathbf{q} = \mathbf{T}\mathbf{u}$ and $\mathbf{f} = \mathbf{T}^T\mathbf{m}$ in Eq (3.12), solving in \mathbf{q} [9]:

$$\mathbf{q} = \mathbf{T}\mathbf{Y}\mathbf{T}^T\mathbf{m} - \mathbf{T}\mathbf{Y}\mathbf{T}^T\mathbf{B}^T(\mathbf{B}\mathbf{T}\mathbf{Y}\mathbf{T}^T\mathbf{B}^T)^{-1}\mathbf{B}\mathbf{T}\mathbf{Y}\mathbf{T}^T\mathbf{m} \quad (3.15)$$

To simplify this notation, the product $\mathbf{T}\mathbf{Y}\mathbf{T}^T$ can be grouped in a *virtual point matrix* for each substructure that includes virtual point responses to virtual point loads (and so subscript qm are used):

$$\begin{aligned} \mathbf{q} &= \mathbf{Y}_{qm}\mathbf{m} - \mathbf{Y}_{qm}\mathbf{B}^T(\mathbf{B}\mathbf{Y}_{qm}\mathbf{B}^T)^{-1}\mathbf{B}\mathbf{Y}_{qm}\mathbf{m} \\ \mathbf{Y}_{qm} &= \mathbf{T}_u\mathbf{Y}\mathbf{T}_f^T \end{aligned} \quad (3.16)$$

Again, this formulation can be solved by LM-FBS algorithm.

3.3 Basics on measurement techniques

3.3.1 Data acquisition – Comparison between sensors

In experimental Substructuring, before defining and applying Substructure coupling, decoupling or VPT, dynamic features of the system are required, usually obtained by input and output measurements on the structure.

It's common to get FRFs of the structures by post processing data from impact tests. This is usually performed with an impact roving hammer to generate input and using sensors to get the output.

The impacts themselves need to satisfy requirements³, from the right location to the type of impact (that needs to be single, impulsive and with high amplitude), but what will make the difference in the post processing part are the sensors used to get output.

The most common one is acceleration transducer, i.e. accelerometer, which measures excitation response through the pressure value of a seismic mass on a piezoelectric body.

This kind of sensors are very precise, but they provide information for a limited number of locations. This leads to a discrepancy between experimental information and the huge amount of data one can get from a numerical model. Even if some expansion methods were applied, there is a risk of turning away

³ For further information about performing a good impact test see [6].

Basics on measurement techniques

from real information [10]. Moreover, these sensors add some mass to the system, which can modify the complete dynamic itself.

For these reason, non-contact methods have been developed, from the laser beam to the white light ones.

Laser Doppler falls into the first type and measures vibration velocity of surfaces. It's quite accurate but has the disadvantage of measuring just one point at a time.

Electronic Speckle Pattern interferometry (ESPI) is also a laser beam method and measures displacements (even out-of-plane) thanks to interference fringes [11].

High-speed camera method falls instead into the second category of non-contact sensors. and its biggest advantage is to get full-field optical displacements, because each pixel can be considered as a single sensor. Also, complex structures movements can be analyzed through slow-motion video.

3.3.2 Displacements from digital images – High-speed camera

Even if can be hard to deal with a huge amount of data that are usually affected by noise [1] [12], high-speed cameras have many advantages and that have significantly increased interest in getting displacements from digital images. This interest has resulted in several method developed to determine displacements.

For instance, Digital Image Correlation (DIC) consists in selecting subset in consecutive images and measuring displacements by finding the position with the highest correlation in the second image and so on for all pair of consecutive images.

Newton-Raphson algorithm is a root-finding DIC approach that calculates displacements between sequential image by finding correlation through a convergence process. It is a quite accurate method, but it's also very time consuming.

Beside DIC, there are several methods to get displacement from images, among which phase optical flow and differential optical flow methods.

The latter are based on intensity gradient of the images, calculate velocity by looking at the difference in intensity at each point between two consecutive images [9]. Since in this study displacements are obtained by Gradient based optical flow, a better description of the methods is provided.

3.3.3 Gradient-based Optical Flow Methods

The intensity of a 2D image can be stored as a grey value from 0 (black) to 255 (white) for each pixel in a 2D matrix.

One of the problems in getting displacement by this method is the *aperture problem*, that is the fact that a gray value associated to a single pixel can be equal to the value of hundreds of pixels on the same image.

To overcome it, one can consider a subset of pixel of interest instead a single one, so more information about the region of interest is available, even if the problem is still not unique. A practical way to implement this strategy is using a spackle pattern on the structure surfaces, so that the texture is not isotropic (i.e. with

Basics on measurement techniques

preferential direction) or periodic either and it becomes easier to identify the position of the region of interest [13].

Mathematical principles of this method can be analyzed after making the assumption that both pattern and light on the structure don't change during time.

Then, the 2D matrix can be considered as a function of the image coordinate $I(x, y)$. When the structure moves, intensity values trough the image coordinates move themselves, so the intensity function depends also on time $I(x, y, t)$.

A further assumption is to consider small displacements and short time intervals, so that intensity change can be approximated by a first order Taylor expansion:

$$I(x_j + \Delta x, y_k + \Delta y, t + \Delta t) = I(x_j, y_k, t) + \frac{\partial I}{\partial x} \Delta x + \frac{\partial I}{\partial y} \Delta y + \frac{\partial I}{\partial t} \Delta t \quad (3.17)$$

Due to the first assumption, looking at one point on consecutive images, the grey value doesn't change:

$$I(x_j, y_k, t) = I(x_j + \Delta x, y_k + \Delta y, t + \Delta t) \quad (3.18)$$

Substituting Eq (3.18) in Eq (3.17) the *optical flow equation* [1] is obtained:

$$\frac{\partial I}{\partial x} \Delta x + \frac{\partial I}{\partial y} \Delta y + \frac{\partial I}{\partial t} \Delta t = 0 \quad (3.19)$$

The last term represents intensity variation of a selected pixel during time (image coordinated are fixed) and can be calculated by comparing pair of consecutive images:

$$\frac{\partial I}{\partial t} \Delta t = I(x_j, y_k, t + \Delta t) - I(x_j, y_k, t) \quad (3.20)$$

Now Eq. (3.19) become:

$$\frac{\partial I}{\partial x} \Delta x + \frac{\partial I}{\partial y} \Delta y = I(x_j, y_k, t) - I(x_j, y_k, t + \Delta t) = \Delta I(x, y) \quad (3.21)$$

This equation involves two unknowns, Δx and Δy , because the intensity function I is known for each pixel, so the terms $\frac{\partial I}{\partial x}$ and $\frac{\partial I}{\partial y}$ can be computed for the whole matrix.

Since there is one equation for two unknowns, the problem is underdetermined, and that is mathematical expression of the aperture problem [11].

Two methods are been developed to overcome this underdetermination. The first one is the *Lucas – Kanade Optical Flow Method*, which consists in adding the hypothesis that a subset of pixels near to the interested pixel has the same displacement. With this assumption, Eq. (3.21) can be rewritten for each subset of pixel:

$$\begin{aligned} \frac{\partial I_1}{\partial x} \Delta x + \frac{\partial I_1}{\partial y} \Delta y &= \Delta I_1(x, y) \\ &\vdots \\ \frac{\partial I_n}{\partial x} \Delta x + \frac{\partial I_n}{\partial y} \Delta y &= \Delta I_n(x, y) \end{aligned} \quad (3.22)$$

This equation can be now solved in a least square sense [1] to get displacement in both x and y direction:

Basics on measurement techniques

$$\begin{Bmatrix} \Delta x \\ \Delta y \end{Bmatrix} = - \begin{bmatrix} \sum \left(\frac{\partial I_i}{\partial x} \right)^2 & \sum \left(\frac{\partial I_i}{\partial x} \frac{\partial I_i}{\partial y} \right) \\ \sum \frac{\partial I_i}{\partial x} \frac{\partial I_i}{\partial y} & \sum \left(\frac{\partial I_i}{\partial y} \right)^2 \end{bmatrix}^{-1} \begin{Bmatrix} \sum \frac{\partial I_i}{\partial x} \Delta I_i \\ \sum \frac{\partial I_i}{\partial y} \Delta I_i \end{Bmatrix} \quad (3.23)$$

The second method is the *Simplified Gradient Based Optical Flow Method* (SGBOF) that consist in calculate displacement in the direction of the gradient:

$$\begin{aligned} |\nabla I| \Delta s &= I(x_j, y_k, t) - I(x_j, y_k, t + \Delta t) \\ \text{where } |\nabla I| &= \sqrt{\frac{\partial I^2}{\partial x} + \frac{\partial I^2}{\partial y}} \end{aligned} \quad (3.24)$$

Δs is the unknown displacement, and for a simplified condition where Δs is perpendicular to the gradient vector , the displacements can be obtained for horizontal or vertical edges [11, 1]:

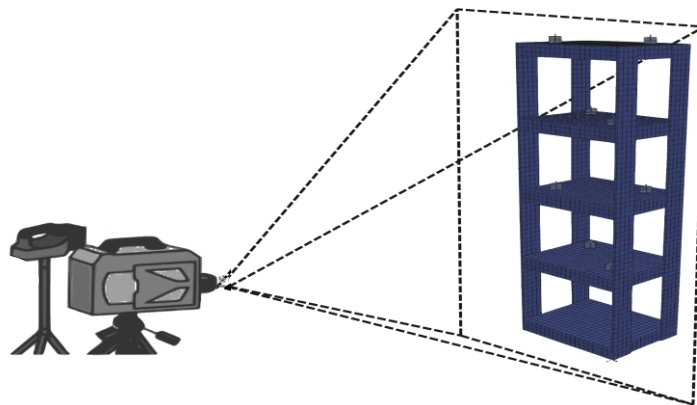
$$\begin{cases} \Delta y = -\frac{\Delta I(x, y)}{\frac{\partial I}{\partial y}}, \text{ if } \frac{\partial I}{\partial x} = 0 \text{ (horizontal edges)} \\ \Delta x = -\frac{\Delta I(x, y)}{\frac{\partial I}{\partial x}}, \text{ if } \frac{\partial I}{\partial y} = 0 \text{ (vertical edges)} \end{cases} \quad (3.25)$$

Chapter 4

Preliminary Analysis on tower structure

The final aim of this topic of research is to use HSC measurements for coupling substructures. To demonstrate the feasibility, to perform SEMM method and to properly mix camera data and accelerometers data, measurements need to be acquired for each substructure.

However, before taking substructures data, some preliminary tests are performed on a single tower structure, in order to get familiar with the equipment and make the most of the camera functions.



4. 1 Drawing of the preliminary test

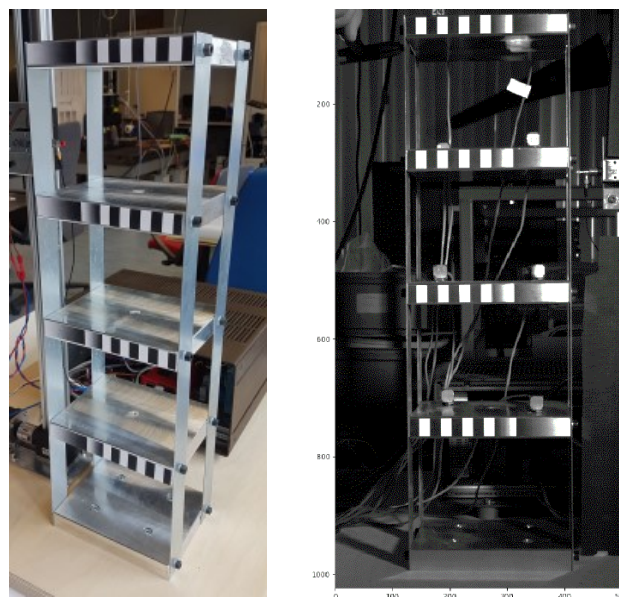
Tower set up

4.1 Tower set up

Preliminary experimental measurements are carried out on a four levels tower, which moves with a low frequency and high amplitude. As mentioned above, this test is performed to better understand how to acquire displacements from high-speed camera data, so the structure is excited with a single hammer impact on the highest level.

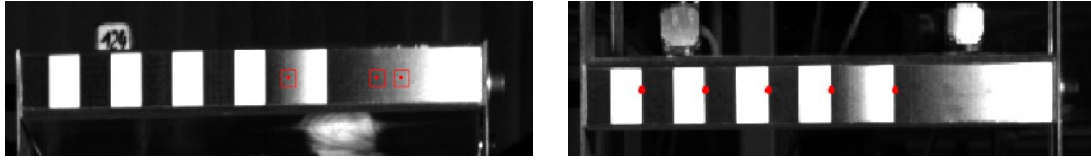
By the nature of the methods used to get displacements, thus Lucas Kanade (LK) and Simplified Optical Flow (SOF) methods, recorded surfaces need to satisfy some requirements to overcome the aperture problem.

Since the structure is made by a single material (thus a single color), stickers are applied on surfaces before filming. The stickers have vertical edges that are suitable for SOF application and a diffused gradient of color that is suitable for Lucas Kanade method.



4. 2 Tower set up

Preliminary Analysis on tower structure



4.3 Selected points for LK (left) and SOF (right) methods

In order to make a comparison between camera data and more known data, accelerometers are mounted on the structures and measurements are taken at the same time. A representation of the complete set up and the location of the impact is in Figure 4.2.

Once the video of the impact is filmed and the accelerometers data are acquired, these raw data are elaborated through a Python code which provide displacements thanks to the free image processing library Pyidi⁴ (Python Image based – Displacement – Identification). For the SOF method displacements are obtained from pixels on the vertical edges, for the LK method instead points are selected on the diffused gradient, as represented in Figure 4.3.

For each point, displacements are calculated for both direction x and direction y . Then these vectors are then manipulated and processed through the Python code to get the FFTs of the displacements first, then to get the receptances.

⁴ <https://github.com/ladisk/pyidi>

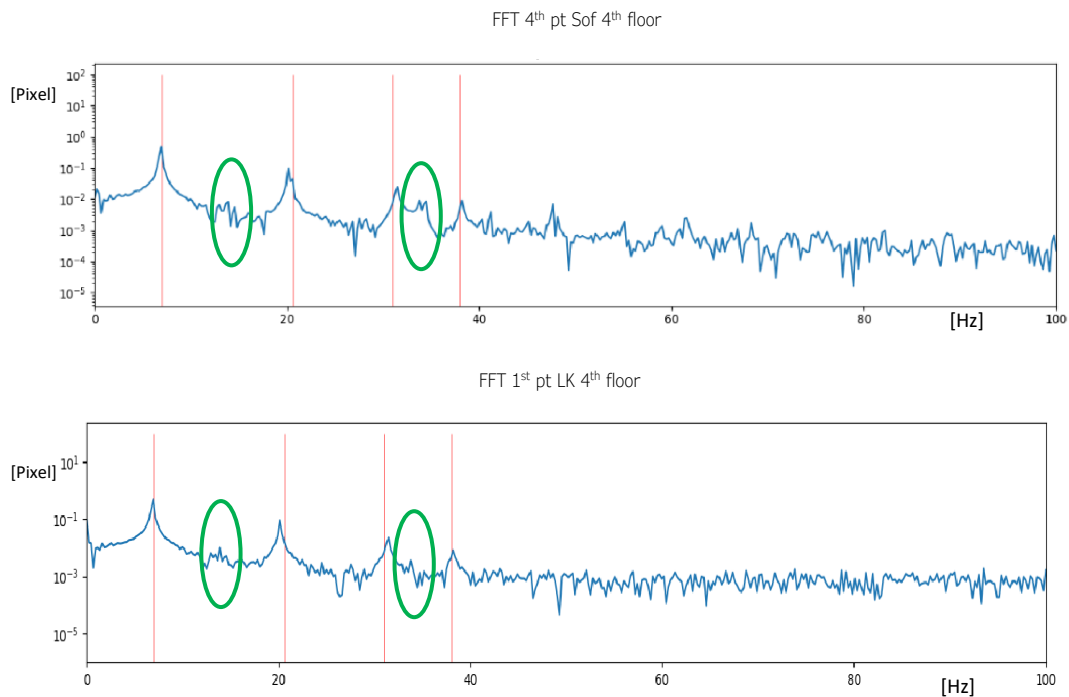
Results of the preliminary analysis

4.2 Results of the preliminary analysis

In order to evaluate differences between SOF method and LK method, the comparison should be carried out on two FFTs of the same pixel, calculated by the two methods.

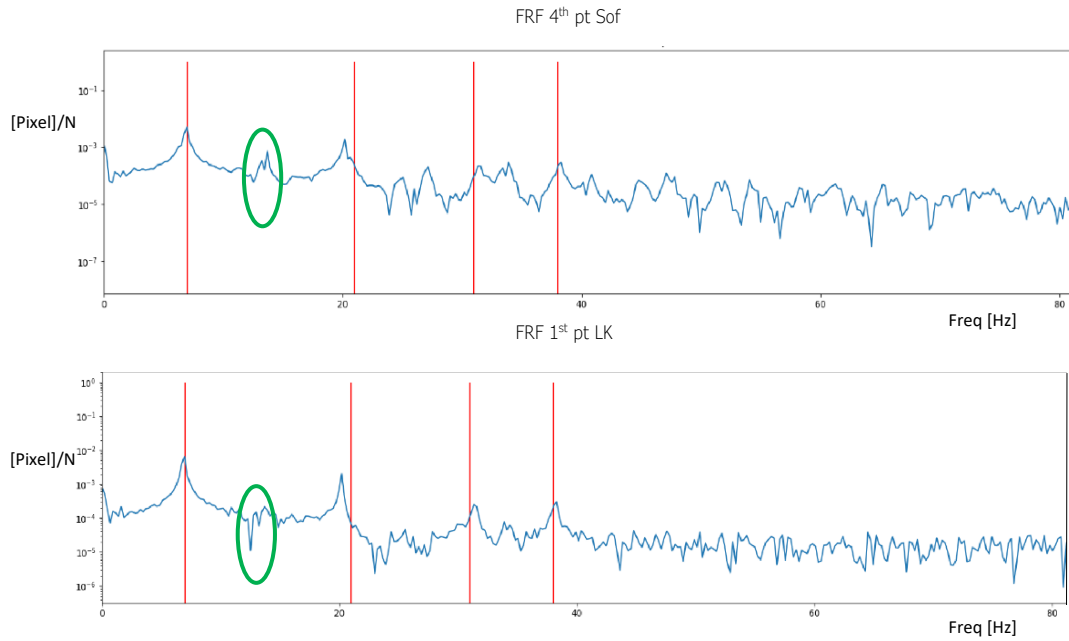
However, this is not possible due to the different approaches and different patterns of the methods. Thus, the comparison is performed on two different pixels, but as close as possible to each other: first pixel of the LK set and the fourth of the SOF set, on the same level.

To get the FRFs the force vector is necessary, but the camera can't acquire it. To overcome this problem, an artificial force vector is used, with just the first value non-zero to simulate an impulsive impact. The resulting FRFs are an



4. 4 Fast Fourier Transformation of displacements obtained by SOF (above) and by LK (below)

Preliminary Analysis on tower structure



4. 5 Receptances for displacements obtained by SOF (above) and by LK (below)

approximation but still representative of the phenomenon, as shown in Figure 4.5, where the modes of the structure are well recognizable.

In both Figure 4.4 and Figure 4.5, Lucas Kanade methods seems to recognize better the modes of the structure, avoiding some spurious peak that instead appears with the simplified optical flow method.

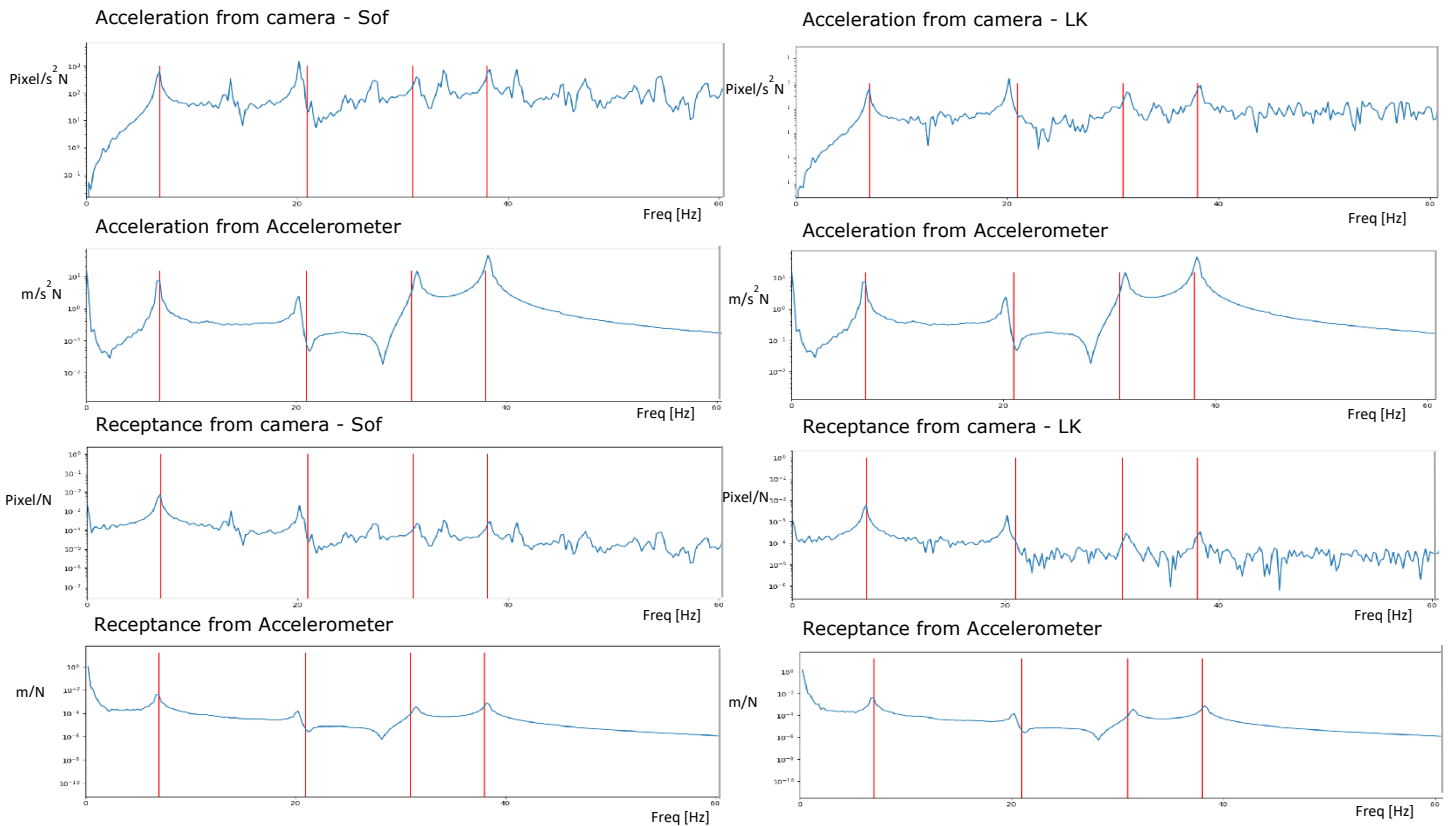
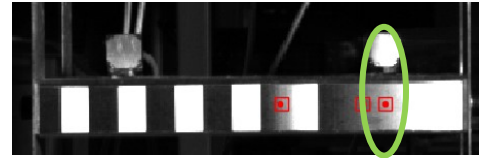
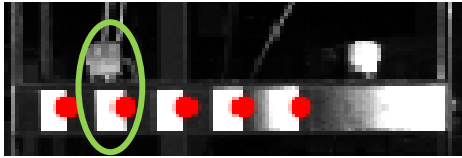
Once the camera data has been obtained, they are compared with accelerometers data. Accelerometers notoriously give more accurate results, and the acquisition system PAK independently calculates the FRFs, using the force vector directly from the hammer signal that is acquired from PAK.

Figure 4.6 represents the differences of the two methods with accelerometers data. For both methods, one pixel and the closest accelerometer have been chosen. PAK system output are accelerances, so these vectors are divided by ω^2

Results of the preliminary analysis

to compare them with camera receptances and camera receptances are multiply by ω^2 to compare them with PAK accelerances.

The amplitude scale is different because displacements are calculated in pixel as a first approximation, but it's evident how accelerometers data are cleaner and noise-free compared to the camera data. However, the vibrational modes of the structures, at least for low frequency, are still recognizable and this leads to think that the camera measurements can be effective even if noisy.



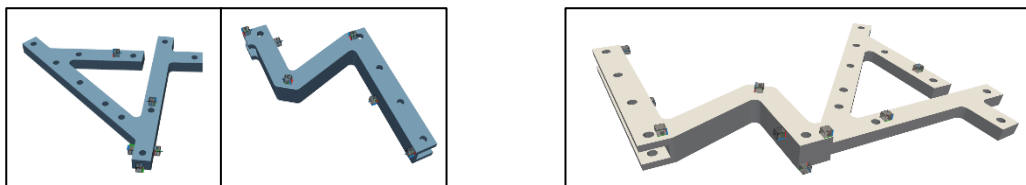
4.6 Comparison between camera and accelerometers data

Chapter 5

Analysis on complex structure

Now that preliminary measurements prove that valid results can be obtained from camera data, more complex structures can be tested and analyzed. This chapter describes the measurement campaign that has been carried out on a first substructure suitable for coupling and discusses how raw data are manipulated through Python and MATLAB codes to create the final SEMM matrix ready for coupling.

Since the final aim of this experimental study will be to couple dynamics of two substructures (displaced in Figure 5.1), first of all measurements of a single substructure are computed, starting from A structure.



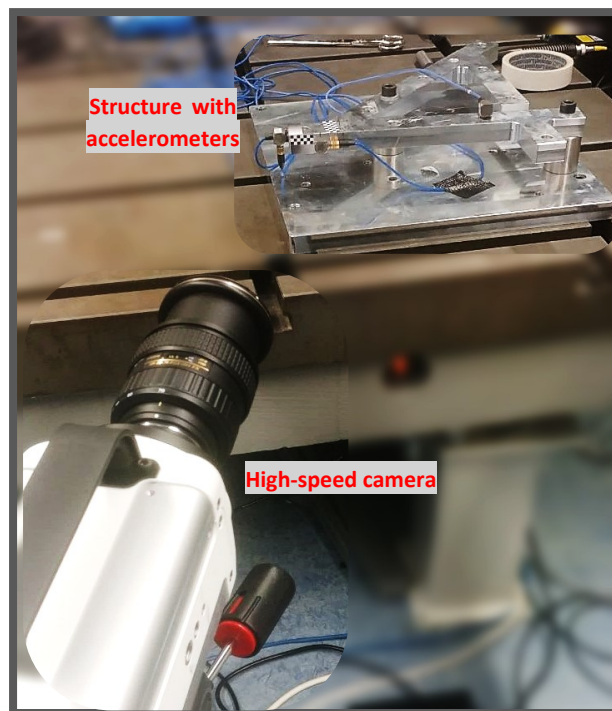
5. 1 Drawing of the two separate substructures (left) and of the coupled structure (right)

Experimental setup

5.1 Experimental setup

5.1.1 Structure and camera arrangement

Like the previous measurements, both accelerometers and camera data are carried out simultaneously, filming the structure responses to hammer impacts for each planes of coordinate XZ , YZ , XZ and acquiring acceleration and force vectors at the same time. An example for the structure setup for the YZ plane measurements is shown in Figure 5.2.



5. 2 Photograph of the experimental setup for XY plane measurements

Analysis on complex structure

On the structure 4 accelerometers are placed, connected with a PAK system to collect data. PAK system acquire data with a Sampling Rate of 102401 Hz, a Frequency Resolution of 0.25 Hz and a Block Duration of 4 s. As described in Chapter 3, three triaxial sensors are enough for switching from 9DoF to 6DoF in the VPT: these three are placed as close as possible to the interface point, to better describe interface dynamic (that is the focus in structures coupling) and to reduce the effect of flexible modes in VPT application. However, an extra accelerometer for validation is introduced, placed far from the coupling point, because more information can mitigate errors.

For the VPT, at least 6 impacts need to be performed, to determined virtual forces of the VP, but again more impacts can reduce errors due to the overdetermination of the transformation. Thus, 9 impacts are carried out around the virtual point: two in X direction, two in Y direction, two in -X direction and three in -Z direction.

The hammer is wire-linked to the PAK system to acquire force vectors, and for each impact direction the structure is hit three times in order to average the tree impacts and reduce errors (for examples due to the fact that the chosen point of impact is not hit precisely, since the structure is hit manually).

A Fastcam Nova high-speed camera is placed near to the structure, with lens as parallel as possible to the structure face to have accurate 2D displacements. The camera has been set with a resolution of 512x512 pixels and recorded at 4000 frames per second. The camera is moved parallel to a plane every time measurements for a new plane are repeated.

Experimental setup

From each video 2 seconds of data is required, because this allows to have a frequency resolution of 0.5Hz (when the data are converted to the FFT) that is compatible with accelerometers data. This is a fundamental requirement not only to compare data from different source, but also to apply SEMM method in the second phase of the work.

4 seconds of data would be even better, because it would give more time to the structure to vibrate and the risk of excluding some data would be reduced. However, this was not possible due to archiving reasons: each second recorded is about 2Gb, so increasing data to 4 seconds (that would mean increasing video duration to about 5 seconds) would mean increasing to about 10Gb the space needed to store each video. Considering that the total expected video for the structure is 60 (7 impacts for XZ and YZ plans, 6 for the XY plan, each repeated three times), 600Gb of data would be unmanageable.

The structure is illuminated by an additional source of light and as the preliminary structure two different patterns (here a chessboard pattern and a speckle pattern) are applied on faces to facilitate data collection both with Lukas Kanade and Simplified Optical Flow methods.

5.1.2 Raw data acquisition

The experiment is carried out by executing one hammer at a time, recorded by the high-speed camera. Once done, repeated and filmed the hits for the first impact location, PAK system automatically processes three impacts and acquire

Analysis on complex structure

the averaged impact data. Then, the procedure is repeated for the next location and so on until all 27 hammers are performed, recording in PAK all the 9 impacts.

The average between the videos of the three hammer per location is computed in the post processing phase.

The high- speed camera film all the hits, except the ones in the faces being filmed, because only hand and hammer would be framed, and no displacement can be computed. Therefore, for XZ plan two impacts in -Y direction, for YZ plan two impacts in -X direction and for XY plan three impacts in -Z direction are excluded.

The hammer is connected with the PAK system, but isn't related with the camera. Camera recording start with a manual trigger, that is pressed just before impact. This leads to a lack of synchronization and therefore to an excess of frames that are removed in the post processing phase, discarding data in the displacement vectors related to the time before the impact. Thus, to get 2 seconds of data, about 3 seconds of recorded file is required.

After the measurement campaign, the videos are processed through a python code, written to work on one plan at a time.

First, some pixels are selected to calculate their displacements. Ideally, the more points are selected, the more information on the structure are available. On a practical level, choose a high number of pixels make the points very close to each other and that leads to information very similar to each other with a high computational cost.

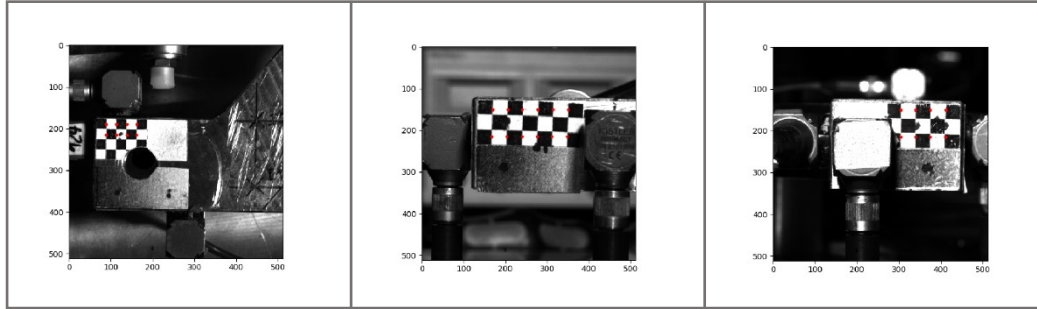
Experimental setup

In this work, 16 pixels are selected in the XY and XZ plans, 24 pixels are selected in the YZ plane. For each plan, half of them relate to the chessboard pattern for the SOF method, the others relate to the speckle pattern for the LK method.

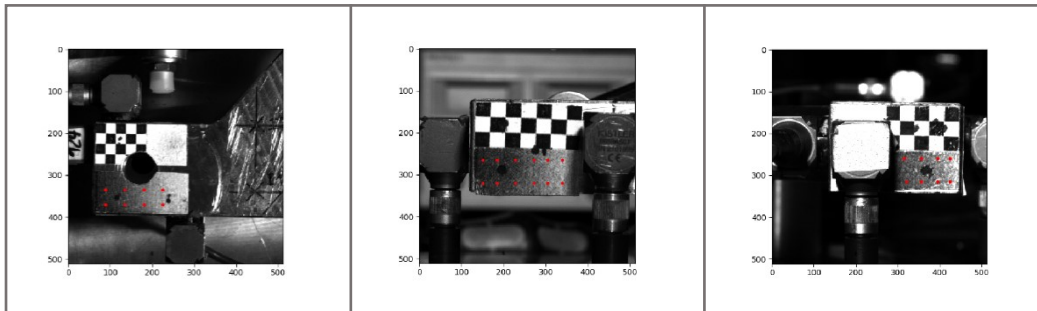
Selected pixels for each plane are represented in Figure 5.3 and Figure 5.4.

After selecting the points, displacements are computed in the directions included in the video frames, thus with both Lucas Kanade and Simplified Optical Flow methods, using the Pyidi library (for code details see Appendix).

Figure 5.5 and Figure 5.6 show an example of displacements for YZ plan, because are represented displacements in Y and Z direction for 12 pixels for each method. The title of figures related to the direction, the method and the impact: H43, in fact, refers to the third impact of fourth location (thus in Y direction) and H91 refers to the first impact of the ninth location (in -Z direction).

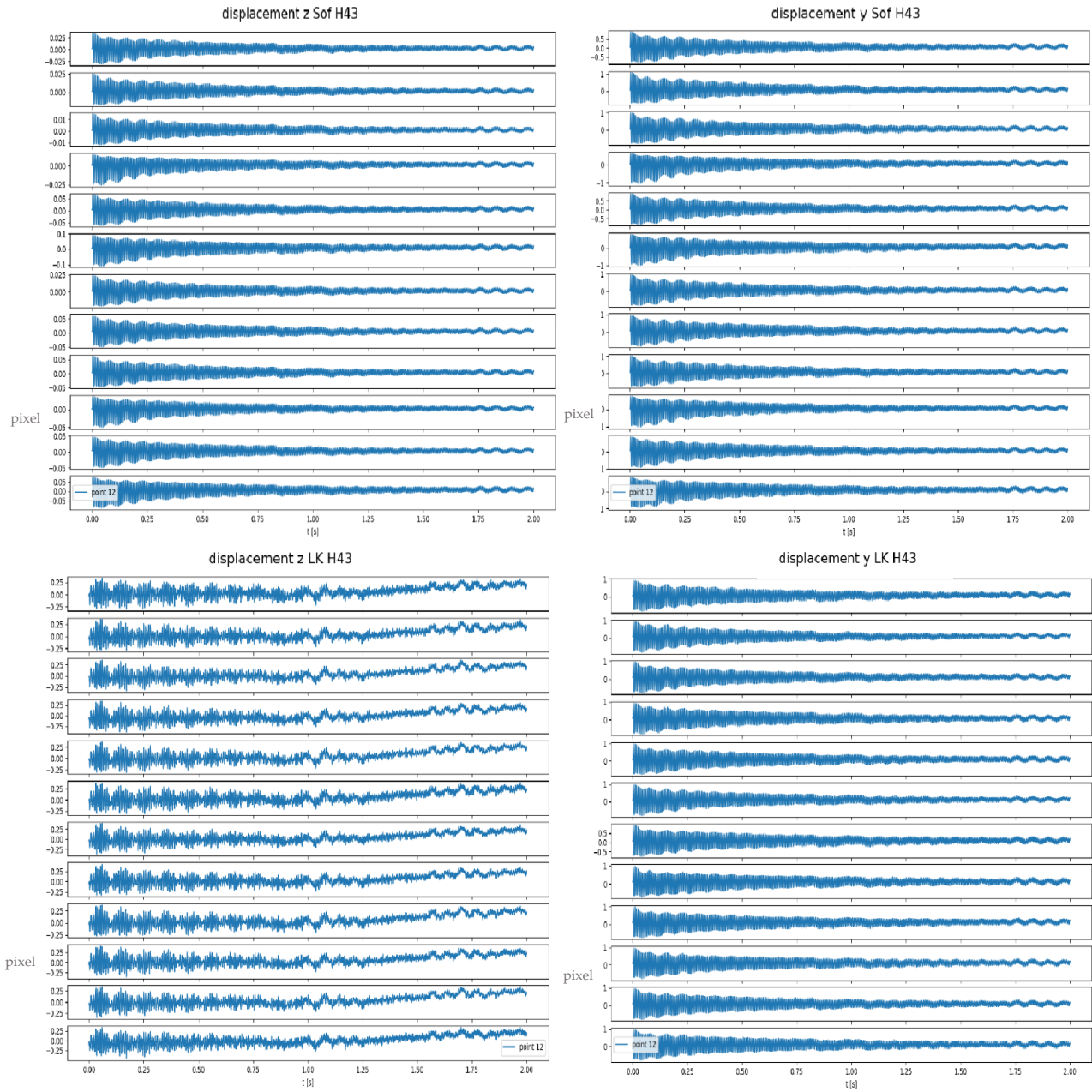


5. 3 Selected Pixel for SOF - plan XY, YZ, XZ



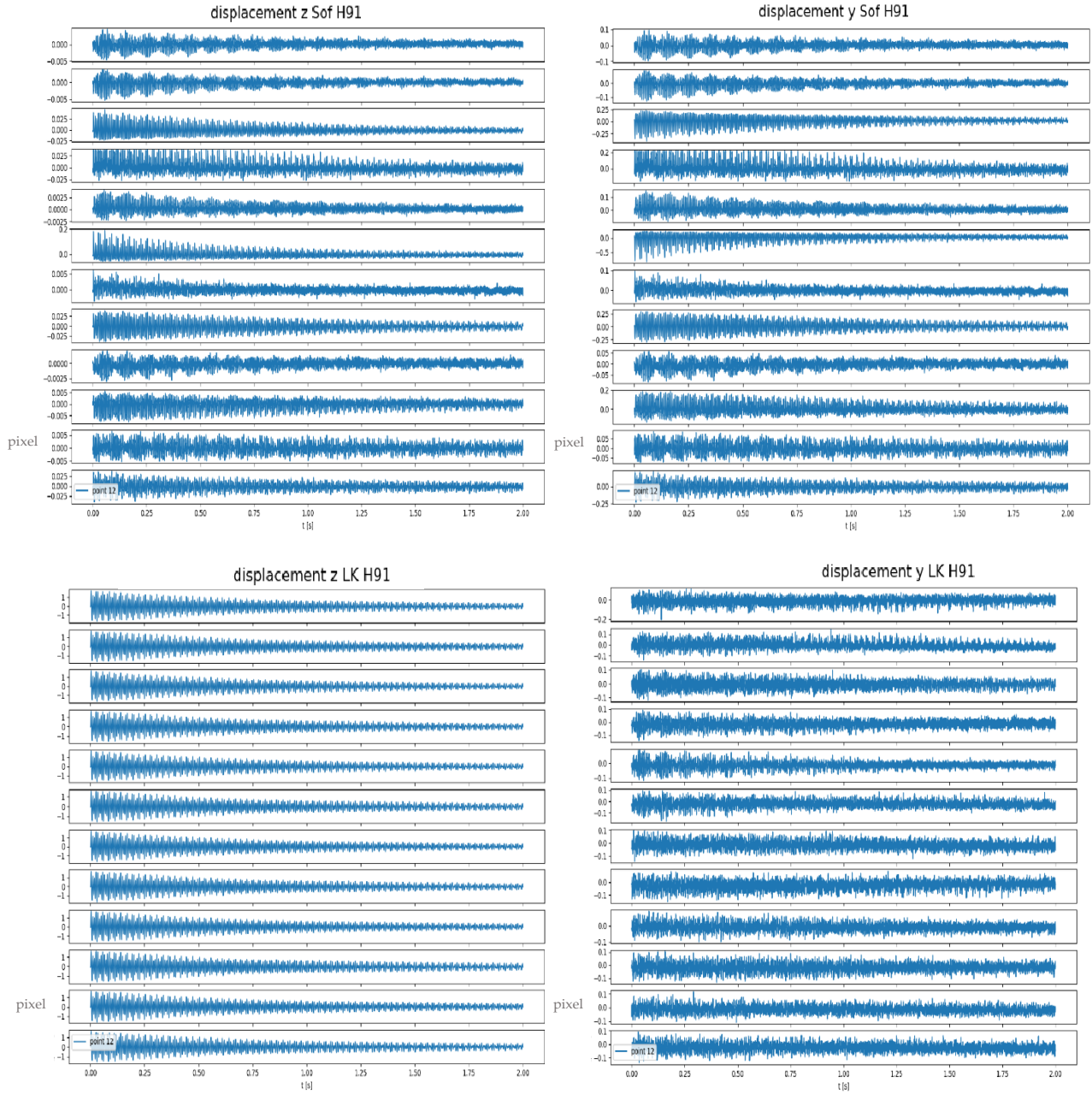
5. 4 Selected Pixel for LK - plan XY, YZ, XZ

Analysis on complex structure



5. 5 Response to the third impact at the 4th location for the SOF pixels (Above) and for the LK pixels (below)

Experimental setup



5. 6 Response to the first impact at the 9th location for the SOF pixels (Above) and for the LK pixels (below)

Analysis on complex structure

In these figures the goodness of the method is shown: for the impact in Y direction, where the hit was quite strong, both method recognize maximum displacement in y direction (about 1 pixel) and minimum displacement in z direction (about 0.25 pixel for LK method and 0.05pixel for SOF). The effect of damped vibration is evident in the y displacement graphs, where two characteristic modes of the structure coexist. Instead, in the z direction obtained with LK method for the 4th location, multiple characteristic modes overlap and intersect each other, but this is consistent with the phenomenon because the force is minimal and doesn't excite a particular mode.

Similar comments can be made for impact in – Z direction, but since this hit was weaker than the previous, in the graphs appears that one method is better than the other: SOF method doesn't recognize properly the direction of the force, in fact displacement is even bigger in y direction (0.25 pixel, versus 0.05 pixel in z direction) and there is no prevailing mode. LK method, instead, give a good description of the phenomenon, with a damped displacement in z direction (up to 1 pixel) that shows a characteristic mode that prevail and a more chaotic and small displacement in y direction (about 0.2 pixel) where no main mode exists.

Once displacements are stored in vectors, NumPy⁵ free library is used to get the Fast Fourier Transformation of each displacement vector (see Appendix), even if some of them are quite coarse, because the not excited directions have very chaotic displacements.

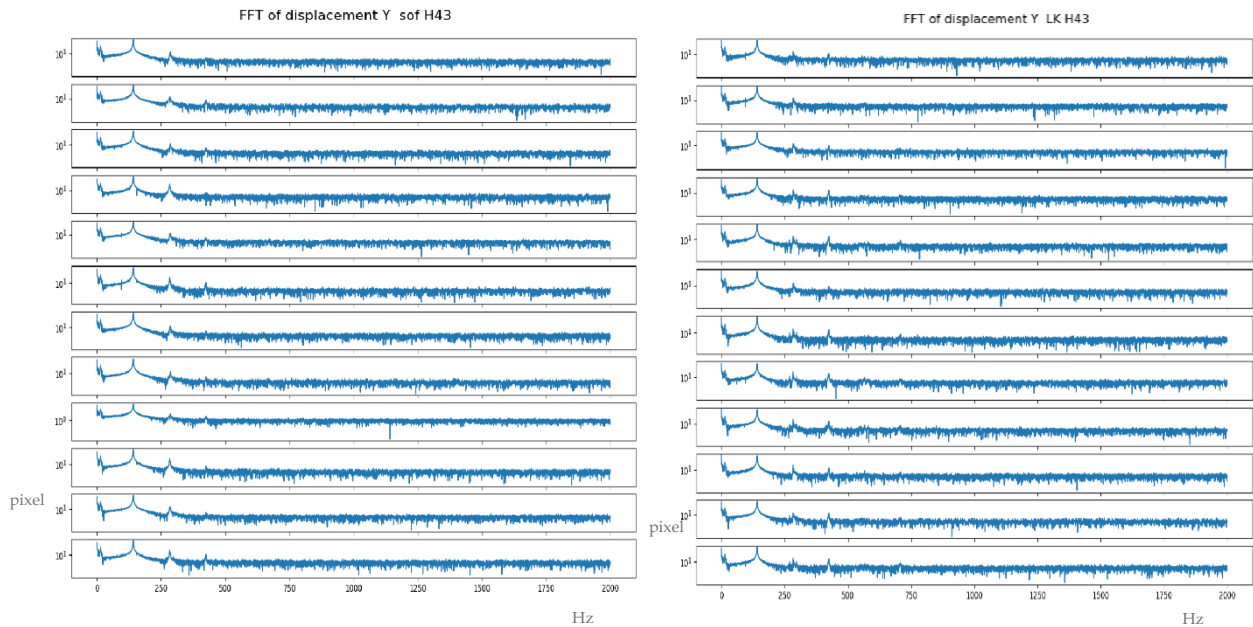
⁵ <https://numpy.org/>

Experimental setup

Figure 5.7 shows an example of the FFT for vectors that related to the responses in y direction to the impact “H43”, as already explained. Y direction is chosen because it is the excitation direction. Since the pixels are pretty close to each other, FFTs are also quite similar to each other and, at least for a low frequency range, vibrational modes of the structure are well identified.

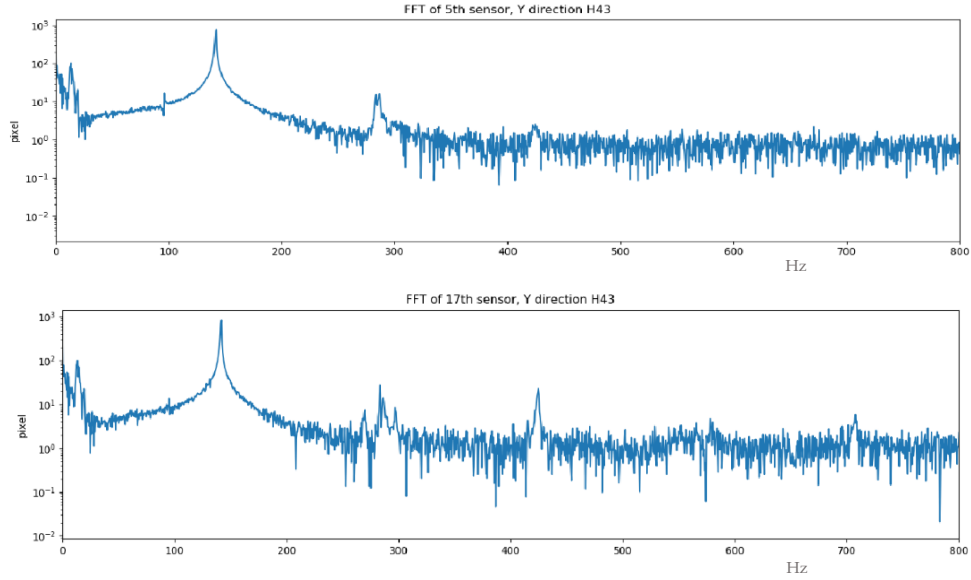
As mentioned in the previous section, FFTs are computed up to 2000Hz, but the characteristic modes are recognizable just in a low frequency range.

Going more into detail, Figure 5.8 show the FFTs for one pixel in the chessboard pattern and one pixel in the gradient pattern in the same direction and for the same impact in a smaller range of frequency.

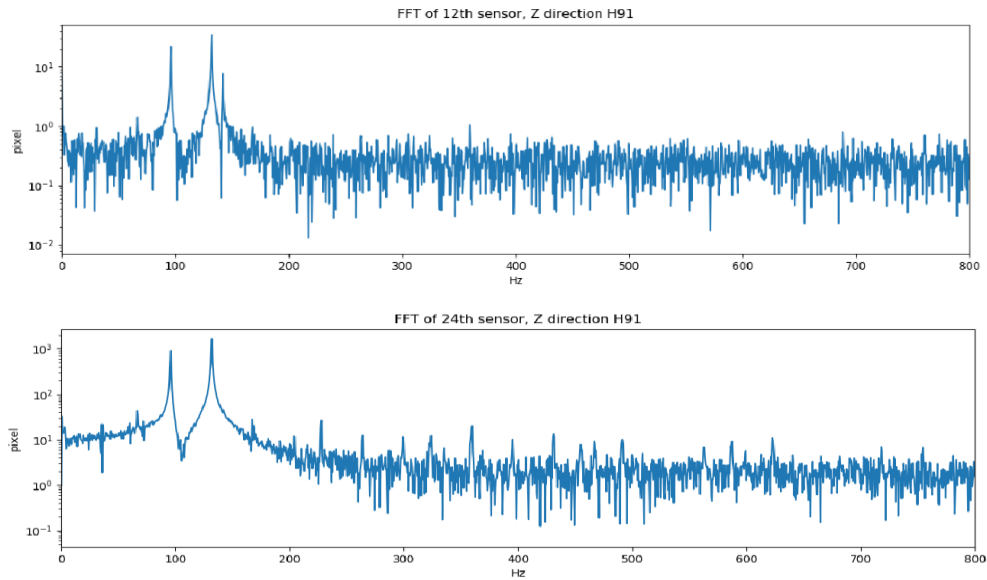


5.7 Fast Fourier Transformation of responses in Y direction

Analysis on complex structure



5. 8 Fast Fourier Transformation for 1 pixels of the chessboard pattern (above) and 1 pixel of the gradient pattern (below) in y direction – impact H43



5. 9 Fast Fourier Transformation for 1 pixels of the chessboard pattern (above) and 1 pixel of the gradient pattern (below) in Z direction – impact H91

FRFs and SEMM method

For the displacements obtained with the LK method, in the FFT is easier to identify peaks, in fact the one around 450Hz is much more visible and there also some peaks that could represent peaks around 700Hz. Instead for the first FFT, even the peak around 450Hz is confused, and after that the response is too noisy.

These comments don't apply to Figure 5.9, where other two pixels are compared, but in the response to the last impact. The graphs are quite similar, even if the first one (related to displacements with SOF method) is noisier, in both of them it is hard to get peaks after 200-300 Hz, even though in the second one they can be hypothesized.

This is probably due to the constraints of the structure and because the force of the last impact is not strong enough to excite modes in a higher frequency range.

Now that FFTs are computed, vectors are manipulated to get a form suitable for a MATLAB code, in order to get FRFs.

5.2 FRFs and SEMM method

FRF of the structure is important to determine the vibrational behavior of the system. To get the FRFs for each DoF, force vectors are required, but the high-speed camera cannot acquire them. To overcome this problem, the force vectors are extracted from the PAK system, that acquires FFT of both input (force) and output (accelerometers DoF) to process and calculate FRFs.

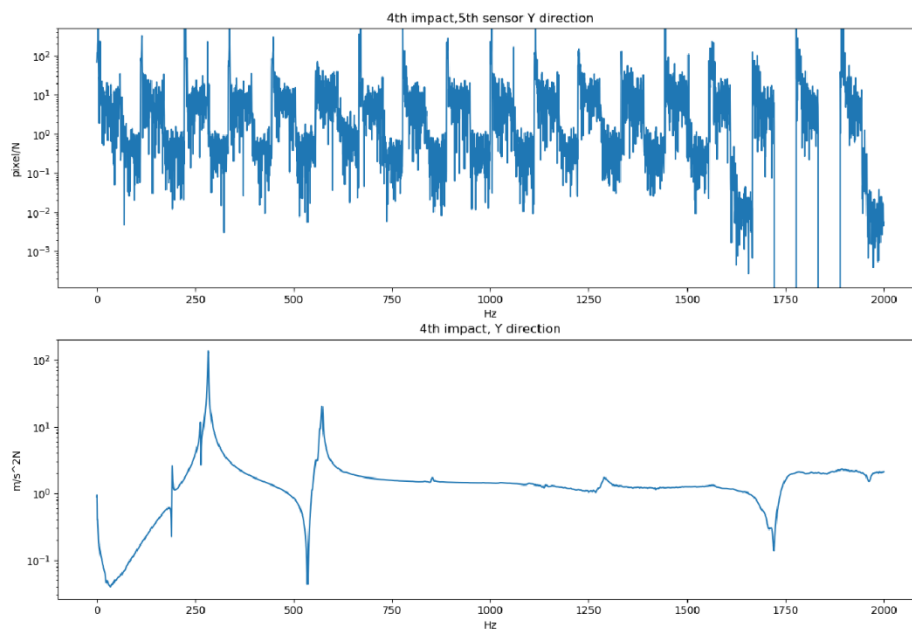
Analysis on complex structure

Since FRF is obtained from a division of the input FRF by the output FRF, the force vectors shall not only match the associated response, but also be compatible in terms of matrix size to the camera FFT vectors.

To compute this division then, force vectors must be compatible and it is useful to have chosen the same frequency resolution: now it is possible to simply delete some data from the vectors so that they contain the same number of elements and are of the same frequency range of 2000 Hz with the same frequency resolution of 0.5Hz.

Once the vectors are comparable, a MATLAB code is used to obtain the FRFs and the graphs obtained are compared with the accelerometer values.

As Figure 5.10 shows, unfortunately the FRFs are so noisy that characteristic modes are not recognizable even in a low frequency range. Some errors in the code could be caused, but not only.



5. 10 Receptance of the 5th pixel (above) compared to acceleration of one of the accelerometers (below)

FRFs and SEMM method

This could be due both to the very low excitation forces, and to the fact that the force vectors obtained by the PAK system have undergone a certain windowing, so they are already partially processed, while the data obtained by the camera are without windowing.

However, the application of the SEMM method is possible, because, as explained in chapter 3, in this method the dynamic of the final hybrid system is provided by the overlay model, and in this work the overlay matrix is formed by accelerometers information, that are very clean and precise, so the final result could be acceptable anyway.

Since the SEMM method involves the integration of two different matrix that describe the same structure, before applying the method an intermediate step is required.

In fact, the overlay model must be a subset of the parental model and must share the interface DoF. However, accelerometers describe the behavior of the structure in some point that are not part of the pixel sets. So, two point for each filmed plane are chosen to become the DoF for the overlay matrix. These points need to include the accelerometers dynamic, thus a VPT is applied. The transformation allows to project the dynamic of the four accelerometers in one point, that is not virtual in this case, but it is one of the selected pixels. VPT is repeated for each pixel used to create the overlay matrix describe the same dynamic of the accelerometers but has a subset of the parental model as DoF (see Appendix).

Now the parental model matrix is formed by the FRFs of each pixel of each plan, and the overlay model matrix consists of information of the accelerometers transported in two pixel per plane.

5.3 Results

Thanks to the open-source-package pyFBS⁶, that is the same that allow to apply VPT, SEMM method can be computed on the two models, with the setting “fully extend” because, as already explained, this thesis apply the method considering as removal model all the dynamic of the parental model and not only the common DoF.

This operation leads to the final SEMM hybrid matrix, that describe the dynamic of the whole structure (see Appendix).

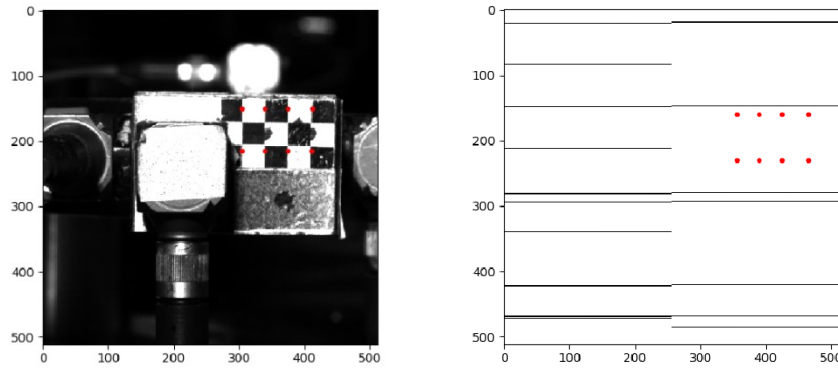
Before repeating the whole experiment for the second structure and proceed with the coupling, the VPT of the whole matrix is applied towards the virtual point of connection between the two structures, in order to make the coupling between structures effective, that properly describes the dynamics of the entire system.

As explained in the previous section, the hybrid SEMM matrix is created after calculating FRF of all pixels on the three plans and after having carried out six VPT from the accelerometers to the selected pixels.

Unfortunately, in XZ plan something went wrong with the video’s storage, because some of the videos are corrupted even if the size is apparently correct (around 6Gb) the representation is not the right one. In fact, when pixels are

⁶ <https://pyfbs.readthedocs.io/en/master/#>

Results



5. 11 Comparison of the corrupted video (7th impact, right) and the normal video (1st impact, left)

selected in each video to get displacements, the frame presents some black and white line instead of a photograph of the structure, as shown in Figure 5.11.

This error has been found only in this plan for video of the impacts 7,8,9.

Obviously, from these videos it was not possible to calculate the displacements, but this was not completely invalidating, because this work involves the use of pixels as a parental model, thus using the camera data to get information of a high number of points, but there is no minimum number of pixels to consider. So, these corrupted data are simply discarded from the procedure.

Another consideration to make is about the results obtained with the package Pyidi, because displacements are determined in pixel.

However, calculating a proportion of the number of pixels in the image (with the image resolution data) and the actual dimension of the structure, it turns out that 1mm is registered by $6 \div 7$ pixels. This means that, if the maximum displacement is 1 pixel, that would correspond to a displacement of $\frac{1}{6 \div 7} = 0.14 \div 0.16$ mm.

The problem is that physically the maximum displacements of the structure far exceed the value of 1 pixel. So, using these methods to calculate displacements

Analysis on complex structure

may result in deletion of data that are important to identify the dynamic of the structure.

The use of high-speed camera for the analysis of the structures dynamic is still in a preliminary phase, so this work has been carried out in an effort to solve the obstacles that have arisen one by one, from the right setting in the camera, to the determination of the displacements.

At the end, a SEMM matrix has been calculated, ready for a coupling procedure that can be used in dynamic substructuring.

Chapter 6

Discussion and future development

The use of high-speed camera to get dynamic information about structures is still a new field in evolution.

The foregoing discussion has attempted to demonstrate that high-speed camera measurements can be used in substructure coupling.

Without claiming that the work is not done in 2020, the global pandemic situation influenced this work, especially in the time elapsed between one measurements campaign and the other but also, even minimally, in the tools and instruments available.

The hypotheses were tested with data covering accelerometers data, clear but limited in numbers, and camera data, a huge amount of data but very noisy.

Among the results, one of the crucial aspects is the comparison between the two methods to get displacements from the video: simplified optical flow and Lucas Kanade methods. The experiments lead to the conclusion that LK method is more efficient and precise in determination of displacements. The LK computational

cost is higher, and it's more complex and delicate than the simplified optical flow, because the pattern can't be too coarse or too fine. For this reason, one can make a preliminary test with the simplified optical flow first, because it's easier to use, but on balance with the LK method, the characteristic modes are more recognizable even at higher frequency, that is a very important factor for describing the dynamic behavior of structures.

Moreover, by summing up the work, it is also found that this method is still quite rough, and to get accurate results the structure needs to be constrained so as to have a high vibrational amplitude, but also needs to be excited by a strong force. In fact, one of the reasons that may have misled from modes identification at high frequency can be the too low force level (from 70 N up to 150 N in this work).

A suggestion for future development of this research is to use bigger force, even an order of magnitude greater.

An additional comment can be made on the main structure experiment: the force vectors used came from the PAK system. This means that windowing has been computed, because the PAK system process vectors to reduce noises. The camera displacement vectors, instead, are without windowing, so some of the noise is included. In future research, windowing on output vector should be applied, so some unwanted values can be discarded.

Furthermore, a different point of view for the parental model should be mentioned. In fact, in this thesis the FRF has been determined from the FFT and the force vectors, and then the SEMM method is applied to the FRF as it is.

However, another way can be followed. After computing the FRF, comparing it with the accelerometers data, the characteristic vibrational modes of the structure should be the same, at least the ones in a low frequency range. A useful

Results

intermediate step could be to use all the poles from the accelerometers, even the ones in a high frequency range, to reconstruct the FRF of the camera data, for example using the free package pyEMA⁷, in order to identify into the noise, the poles of the structure in a high frequency range.

In this way the FRF would be complete and consistent as a parental model and the application of the SEMM method would lead to better results.

All in all, this thesis represents a starting point for this topic of research, even if a complete coupling has not been provided, some solution for common mistakes are proposed, and some alternative for create an appropriate method are suggested.

This study serves as a window to an understanding of the process of high-speed camera measurements in dynamic substructuring, because it express small limits and great potential for future applications.

⁷ <https://pypi.org/project/pyEMA/>

Chapter 7

Summary

This work finds its collocation in the context of dynamic substructuring.

The motivation that drives it is the desire to go deeper into the use of high-speed camera as a tool to improve the coupling between substructures, thanks to the potential and the ability of acquire information about a large number of points on the structures.

In order to do so, the state of the art of dynamic substructuring and of high-speed camera tools have been deepened, to better understand how to approach to experimental substructuring and how to use the available resources.

This thesis is based on frequency formulation of the dynamic, because the elements in $\mathbf{Y}(\omega)$ can be obtained directly from experimental test.

Coupling and decoupling are the final purpose of the study, but they are also the basis of SEMM method, which is applied to combine camera data with accelerometer data, using the LM FBS to decouple the “dirty” dynamic of a model (camera data model), to keep just the DoF information and then couple the tidy dynamic of a smaller set of DoF (accelerometers model) to get the dynamic of the final hybrid model.

Results

A brief description of the VPT is provided because it will be applied several times during the test: VPT consist in impose compatibility only on the DoF that are important for the structure behavior, and project the dynamic from a set of DoF to a single virtual point of chosen coordinates.

Before starting experimental tests, a description of the method to get information from images is provided. Both methods involved in this thesis, Lucas Kanade and Simplified Optical Flow are based on gradient intensity and the Taylor expansion for displacements. LK method considers that a set of pixels around the point of interest moves as the point itself, the SOF instead considers situation where the displacement is perpendicular to null gradient zone.

Then, the main part of the thesis is about experimental tests. First preliminary tests show how camera data can recognize the characteristic modes of the structures, even if the graphs are noisier than accelerometers ones.

The tests on the complex structure gave more complex results. In fact, SOF struggle to recognize the modes, even in a low frequency range, and each method has FRFs that are far from being like the accelerometer ones.

In the work, in addition to going forward with a description of the application of the VPT and the SEMM method, all possible reasons are analyzed that led to those graphs. From this starting point, new ideas for future improvements are proposed, from increasing the force intensity to the windowing to the camera output and many more.

All these suggestions demonstrate that this tool is full of potential and that there are still many ways to go that can prove how well high-speed camera can be included in dynamic substructuring.

Appendix A

Extract of used code

Example of loading video and getting displacements

```
#loading video and information about the camera

mraw11, info11 = pyMRAW.load_video(data11)
video11 = pyidi.pyIDI(data11)

#set pixels for the SOF method

points1 = [[150,169],[150,206],[150,240],[150,277],[150,314],[150,350],
           [215,167],[215,203],[215,240],[215,276],[215,311],[215,348]]
video11.set_points(points=points1)
video11.show_points()

#set method and get displacements

video11.set_method(method='sof')
kin11_sof=video11.get_displacements()

#save or load displacements

np.save( 'kin11YZ_sof.npy',kin11_sof, allow_pickle=True)
# kin11YZ_sof = np.load('kin11YZ_sof.npy', allow_pickle=True)
```

Extract of used code

```
#set pixels for LK method

points2 = [[265,150],[265,185],[265,227],[265,272],[265,304],[265,340],
           [320,150],[320,185],[320,227],[320,272],[320,304],[320,340]]
video11.set_points(points=points2)
video11.show_points()

#set method and get displacements

methLK11=pyidi.methods._lucas_kanade.LucasKanade(video11)
methLK11.calculate_displacements(video11)
kin11YZ_lk = methLK11.displacements

#save or load displacements

np.save('kin11YZ_lk.npy',kin11YZ_lk, allow_pickle=True)
# kin11YZ_lk = np.load('kin11YZ_lk.npy', allow_pickle=True)
```

Animation of the video

```
import matplotlib.animation as animation

def ipynb_animate(video, fps=50):
    #matplotlib notebook
    fig = plt.figure(figsize=(10, 10))
    im = plt.imshow(video[0], cmap='gray', animated=True)
    def updatefig(i):
        im.set_array(video[i])
        return im,
    ani = animation.FuncAnimation(fig, updatefig, blit=True,
frames=video.shape[0], interval=1000/fps)
    return ani

#ani = ipynb_animate(mraw11[80:90,:,:])
```

Extract of used code

Example of creation of FFT matrix

```
frequency_cam = np.fft.rfftfreq(n_disp1YZ, dt_disp)

FFT_cam_raw=np.empty([84,12,4001])

FFTdispY_H11_sof = np.fft.rfft(kin11YZ_sof_cut[:, :, 1], n=n_disp1YZ, axis=1)
FFT_cam_raw[0][:][:]= FFTdispY_H11_sof
FFTdispZ_H11_sof = np.fft.rfft(kin11YZ_sof_cut[:, :, 0], n=n_disp1YZ, axis=1)
FFT_cam_raw[1][:][:]= FFTdispZ_H11_sof
.
.
.
FFTdispY_H11_lk = np.fft.rfft(kin11YZ_lk_cut[:, :, 1], n=n_disp1YZ, axis=1)
FFT_cam_raw[42][:][:]= FFTdispY_H11_lk
FFTdispZ_H11_lk = np.fft.rfft(kin11YZ_lk_cut[:, :, 0], n=n_disp1YZ, axis=1)
FFT_cam_raw[43][:][:]= FFTdispZ_H11_lk
```

Upload camera FRF from MATLAB code and accelerometer FRF

```
#Camera frf

from_matlab = loadmat('DATA/uff/Uf_Test3/A_test_XY/FRF_cam.mat')
freq_cam = loadmat('DATA/uff/Uf_Test3/A_test_XY/freq_cam.mat')

freq=np.array(freq_cam['freq_cam'])
FRF_cam=np.array(from_matlab['FRF_cam'])

#GET ACCELEROMETERS DATA"
uff_file_data_xy=
r"DATA\uff\Uf_Test3\A_test_XY\A_test_XY_FRF.analysis.uf"
uff_file_output_xy=
r"DATA\uff\Uf_Test3\A_test_XY\A_test_XY_FRF.analysis_S.uf"
uff_file_input_xy=
r"DATA\uff\Uf_Test3\A_test_XY\A_test_XY_FRF.analysis_H.uf"
```


Extract of used code

Example of projecting accelerometers data to pixels through VPT

```
#on XZ plane, accelerometers data are projected on the 1st and 16th
pixel

xlsx_acc = r"DATA\xlsx\sensor_impact_location_orientation.xlsx"
df_chn_ov = pd.read_excel(xlsx_acc, sheet_name='ChannelOverlay')
df_chn_par = pd.read_excel(xlsx_acc, sheet_name='ChannelParental')
df_imp_acc = pd.read_excel(xlsx_acc, sheet_name='Impact')
df_chn_acc = pd.read_excel(xlsx_acc, sheet_name='Channel')
df_vpref = pd.read_excel(xlsx_acc, sheet_name='VP_RefChannel')
Hov = np.transpose(Hov, (1, 0, 2))

#VPT to the first pixel of the accelerometers data

df_vp_1xz = pd.read_excel(xlsx_acc, sheet_name='VP_Channel1xz')
vpt_1xz = pyFBS.VPT(df_chn_acc, df_imp_acc, df_vp_1xz, df_vpref)
vpt_1xz.apply_VPT(sensor_pyema.freq, Hov)
Y_qf_1xz = vpt_1xz.Tu @ Hov
Hoverlay_final[:, 0, :] = Y_qf_1xz[:, 0, :]
Hoverlay_final[:, 1, :] = Y_qf_1xz[:, 2, :]

#VPT to the second pixel

df_vp_2xz = pd.read_excel(xlsx_acc, sheet_name='VP_Channel2xz')
vpt_2xz = pyFBS.VPT(df_chn_acc, df_imp_acc, df_vp_2xz, df_vpref)
vpt_2xz.apply_VPT(sensor_pyema.freq, Hov)
Y_qf_2xz = vpt_2xz.Tu @ Hov
Hoverlay_final[:, 2, :] = Y_qf_2xz[:, 0, :]
Hoverlay_final[:, 3, :] = Y_qf_2xz[:, 2, :]
```

Application of SEMM method

```
Y_SEMM=pyFBS.SEMM(Hparental_final, Hoverlay_final,  
    df_chn_num=df_chn_par, df_imp_num=df_imp_acc,  
    df_chn_exp=df_chn_ov, df_imp_exp=df_imp_acc,  
    SEMM_type='fully-extend')
```

Bibliography

- [1] T. F. C. Berninger, *Experimental Vibration Analysis*, 2019.
- [2] J. Javh, M. Brumat, J. Slavic e M. Boltezar, «A high-speed camera measurement set-up for deflection shape analysis».
- [3] S. W. Klaassen, M. V. van der Seijs e D. de Klerk, «System equivalent model mixing,» *Mech. Systm. Signal Process.*, n. 105, pp. 90-112, 2018.
- [4] T. Bregar, K. Zaletelj, G. Čepon, J. Slavič e M. Boltežar, «Full-fiel FRF estimation from noisy high - speed -camera data using a dynamic substructuring approach,» *Mechanical System and Signal Processing*, 2020.
- [5] M. v. d. Seijs, "Experimental Dynamic Substructuring Analysis and Design Strategieess for Vehicle Development," 2016.
- [6] L. Mulder, *Applied Frequency - Domain System Identification in the Field of Experimental Modal Analysis*, 2017.
- [7] T. Bregar, A. El Mahmoudi, G. Čepon, D. J. Rixen and M. Boltežare, "Performance of the Expanded Virtual Point Transformation on a Complex Test Structure, Experimental Techniques," 2020. [Online]. Available: <https://doi.org/10.1007/s40799-020-00398-1>.

- [8] M. Allen, D. Rixen, P. Tiso, R. Mayes, M. van der Seijs e T. Abrahamsson, *Substructuring in Engineering Dynamics - Emerging Numerical and Experimental Techniques*, Springer, 2018.
- [9] M. V. van der Seijs, D. D. van den Bosch, D. J. Rixen e D. de Klerk, *An improved Methodology for the Virtual Point Transformation of measured frequency Response Functions in Dynamic Substructuring*, 2013.
- [10] J. Baqersad, P. Poozesh, C. Niezrecki e P. Avitabile, «Photogrammetry and Optical Methods in Structural Dynamics - A Review,» *Mechanical System and Signal Processing*, vol. 86, pp. 17-34, February 2016.
- [11] J. Javh, J. Slavic e M. Boltezar, «The Subpixel Resolution of Optical-Flow-Based Modal Aalysis,» *Mechanical System and Signal Processing*, vol. Vol. 88, pp. p. 89-99, 2017.
- [12] J. Javh, J. Slavic e M. Boltezar, «Hih Frequency Modal Identification on Noisy High-Speed Camera Data,» vol. 98, pp. 344-351, 2018.
- [13] J. -. J. Orteu, M. A. Sutton e H. W. Schreier, *Image Correlation for Shape, Motion and Deformation Measurements*, Springer, 2009.
- [14] J. G. Chen, N. Wadhwa, Y. -. J. Cha, F. Durand e W. T. Freeman, «Modal identification of simple structrures with high-speed video using motion magnificaion,» *Journal of Sound and Vlbraion*, vol. 345, pp. 58-71, 2015.

Ringraziamenti

Prima di chiudere questa tesi e, con la laurea, questo capitolo della mia vita, voglio sprecare alcune righe per qualche ringraziamento.

Innanzitutto, per il mio relatore Stefano Zucca, per aver assecondato la mia voglia di partire nonostante la pandemia, per avermi fornito quest'opportunità e per avermi seguito durante il lavoro.

In secondo luogo, vorrei ringraziare il professor Rixen, per la possibilità di approfondire un campo di ricerca che non avrei potuto approfondire altrove.

Non manco ovviamente di ringraziare Ahmed, che ogni settimana senza mai mancare un appuntamento ha seguito ogni passaggio della ricerca, correggendo e aiutandomi a superare ogni ostacolo tecnico. Non avrei imparato tutto quello che ho imparato senza un supporto così puntuale.

È stato un anno complicato, e ancora più duri per me sono stati questi mesi all'estero, tanto voluti quanto sofferti. Ho dedicato il 2020 al lavoro, tecnico e personale, ed ora sono qui a raccoglierne i frutti. Ho imparato che un successo è tale quando oltre all'obiettivo raggiunto ci sono le persone che ami e c'è la persona che sei diventata per raggiungerlo. Non sarebbe stato lo stesso senza alcune persone.

Un ringraziamento ai miei genitori, il cui aiuto economico è solo l'ultima goccia della brocca di supporto da cui ho potuto attingere ogni momento e in ogni forma. A Paola, Giuseppe e Valerio che, ciascuno a loro proprio, mi hanno sempre spinto a credere nel mio valore.

A Otta, Ele e Isa, porti sicuri per quando il mare è stato in tempesta, ma anche per passeggiate soleggiate, seppur queste siano solo un ricordo per ora.

All'Avvocato, Rai, Ale, Nonnoleo, Lu, Salvo e Masca, la cosa più importante che il Poli mi abbia lasciato e con cui, nonostante io sia letteralmente di un altro genere, mi sono sempre sentita nel posto giusto.

Ad Adri che ha raccolto tutti i miei santo cielo e li ha trasformati in una risata.

Grazie a tutte le persone che, in un modo o nell'altro, mi sono state vicine e hanno reso questa laurea non solo un obiettivo, ma un successo.

Marina Quaranta