UNIVERSITE PIERRE & MARIE CURIE
LA SCIENCE A PARIS

ONERA

MASTER MATHEMATIQUE DE LA MODELISATION
ANALYSE NUMERIQUE ET EQUATIONS AUX DERIVEES PARTIELLES

UNITE CALCUL HAUTE PERFORMANCE

METHODES DE RECHERCHE DE VALEURS PROPRES PAR DECOMPOSITION DE DOMAINES

Mémoire de DEA

Moussa DIARRA

PROFESSEUR : François-Xavier ROUX

LABORATOIRE JACQUES LOUIS LIONS – PARIS 6
MASTER ANALYSE NUMERIQUE ET EQUATIONS AUX DERIVEES PARTIELLES
http://ann.jussieu.fr

SUBSTRUCTURING AND THE CRAIG-BAMPTON METHOD

Moussa. Diarra

June 27, 2008

Contents

1 Introduction **7**
- 1.1 Onera . . . 7
  - 1.1.1 in brief . . . 7
  - 1.1.2 High Performance Computing Unit . . . 8
- 1.2 The report problem . . . 9
  - 1.2.1 Scientific context . . . 9
  - 1.2.2 Objectives of this report . . . 10

2 Theoretical background of the elastodynamics problem **13**
- 2.1 General equation, boundary conditions and initial conditions . 13
- 2.2 the homogeneous problem . . . 14
- 2.3 Associated spectral problem . . . 15
- 2.4 Variational formulation of the spectral problem . . . 15
  - 2.4.1 construction of the variational fomulation . . . 15
  - 2.4.2 Variational formulation . . . 17
- 2.5 Associated Linear operators and some algebraic properties . . 17
  - 2.5.1 Mass operator . . . 17
  - 2.5.2 Stiffness Operator . . . 18
  - 2.5.3 Rewriting the variational formulation . . . 19
- 2.6 Basic properties of the eigenfrequencies and structural modes for a structure fixed on Γ_0 . . . 19
- 2.7 Basic properties of the eigenfrequencies and structural modes for a free structure . . . 21
  - 2.7.1 The eigenvalue problem . . . 21
  - 2.7.2 Rigid body modes (solutions for $\lambda = 0$) . . . 21
  - 2.7.3 Elastic structural modes (solutions for $\lambda \neq 0$) . . . 21
  - 2.7.4 Orthogonality between the elastic structural modes and the rigid body modes . . . 23
  - 2.7.5 Expansion of the displacement field using the rigid body modes and the elastic structural modes . . . 24
  - 2.7.6 Linear operator equation . . . 24

2.8 Matrix formulation of the generalized eigenvalue problem . . . 24

3 The generalized symmetric eigenvalue problem: a short algebraic description 27
3.1 Some definitions . . . 28
3.2 Reduction to standard form . . . 29
3.3 Shift and Invert . . . 30

4 Substructuring and the CRAIG-BAMPTON method 31
4.1 Dynamic substructuring . . . 32
4.1.1 The principle of the Schur complement method . . . 33
4.1.2 Short description of the FETI method . . . 36
4.2 Primal and dual assembly of substructures . . . 40
4.2.1 Primal assembly of substructures . . . 40
4.2.2 Dual assembly of substructures . . . 41
4.3 The Craig-Bampton Method . . . 42
4.3.1 a brief review of the Component Mode Synthesis . . . 42
4.3.2 Craig-Bampton . . . 43
4.3.3 The dual Craig-Bampton method . . . 48

5 Implementation and numerical analysis 53
5.1 The fortran implementation . . . 53
5.2 Lanczos Algorithm . . . 55
5.3 Computational procedure . . . 56
5.3.1 computation of local eigenmodes . . . 56
5.3.2 computation of global eigenmodes . . . 57
5.4 Implementation of the dual Craig-Bampton . . . 58

Aknowledgement

This report is a result of my studies at the *'Laboratoire Jacques-Louis Lions' University Paris 6*.
I am grateful to Professor François-Xavier ROUX, Head of the High Performance Computing Unit at ONERA, who directed my work and greatly encouraged me. In particular, I want to thank him for allowing me to study as a part time student given the fact that I was a full time employee of an insurance group and was often asked to complete unforeseen tasks. Above all, I want to thank him for his comprehension and also for his patience and generosity, all of which were highly motivating.
Mr ROUX was also my teacher at the university where he introduced us to **the Domain Decomposition Methods and Parallel Computation**.

In addition,I would like to thank all the staff working at the High Performance Computing Unit for their warm welcome and continuous help. Among them, I want to mention Xavier JUVIGNY and Ibrahima GUEYE who often came to meet me at the ONERA's reception despite atrocious weather conditions

Chapter 1

Introduction

The work presented in this report is a theoretical and numerical study concerning the computation of eigenmodes of a structure using a domain decomposition method.

1.1 Onera

1.1.1 in brief

ONERA, the French Aeronautics and Space Research Center is a public, scientific and technical establishment with both industrial and commercial responsibilities, it reports to the French Ministry of Defense and enjoys financial independence.

The expertise of ONERA covers all the scientific disciplines involved in aircraft, spacecraft and missile design. It makes ONERA an essential partner in the French and European aeronautics and space community. ONERA also cooperates with NASA, the US Army and the US Air Force in the area of aerodynamics and computational processes. ONERA cooperates with the Russian TsAGI and CIAM for aerodynamics and ramjet propulsion and also with Asian countries (Japan, China, Singapore).

Missions of ONERA are:

- To assist government agencies in charge of coordinating civil and military aerospace policy
- To direct and carry out aerospace research

- To design, produce and operate the resources needed to perform research and testing for manufacturers
- To make available and commercialize the results of its research and facilitate application of this research by industry, including non-aerospace sectors
- To support the French training policy for scientists and engineers

ONERA conducts research in the disciplines and techniques involved in design of an aircraft or spacecraft: aerodynamics, flight dynamics, energetics, structural strength, materials, optics and laser, acoustics, radar and electromagnetism, electronics, systems, robotics, information processing. The research is focused on federating themes and programmes, such as fluid mechanics and information processing.....

For these purposes, ONERA owns

- A range of research and industrial wind tunnels, some with capabilities unparalleled elsewhere, cover French needs and those of the major foreign manufacturers, from low subsonic to hypersonic speeds.
- a number of laboratories which use state-of-the-art techniques for developing new materials and processes, conducting experiments and making measurements designed in particular to validate mathematical models and numerical simulations.
- powerful computers such as the NEC SX-8 (28 processors for 493 Gflops and 448 Goctets of memory), ONERA excels in high-performance computations. National, European and international agreements allow its partners to benefit from these capabilities.

1.1.2 High Performance Computing Unit

High Performance Computing Unit headed by professor François-Xavier ROUX is a part of DTIM (Modeling and Information Processing Department of the Information Processing and System Branch).
The DTIM's mission is to develop information processing and modeling techniques for the following fields:

- complex information processing systems.
- image processing.

- Numerical and statistical modeling, high-performance computing (parallel and distributed).

particularly, the *High Performance Computing Unit* focuses on solvers (finite elements solvers,...) and Domain Decomposition Method (DDM) for parallel applications and algorithms; design and implementation of numerical methods or algorithms on parallel architectures. It's activities concern:

- parallel and distributed computation
- numerical methods

The research activity also include several state-of-the-art methods for solving large-scale linear systems of equations obtained from the discretization of partial differential equations and all aspects concerning efficient use of scientific computers: architectures, automatic mesh splitting, load balancing on parallel machines.....
In addition, the HPC unit has a strong background on preconditionning techniques and domain decomposition methods.

Hardware

Onera has several means of calculation to serve it's scientific departments:

- a vectorial calculator Nec with 4 SX8+ nodes, i.e. 28 processors for 493 Gflops and 448 Goctets of memory, with a 10 Toctets shared disk space. The peak power of this machine would reach 2Tflops by the end of year 2007
- a superscalar calculator Bull made up of 14 inter-connected nodes, each one including 16 processors, 1.6 GHz and 64 Go of memory, i.e. a total 1.43 Tflops peak performance, 896 Goctets of main memory and 9 Toctets of shared disk space
- a research linux cluster made up from PCs(Intel and AMD) with more than 60 nodes, 153 processors and 280 Goctets of memory

1.2 The report problem

1.2.1 Scientific context

When a struture is very large and complex, several difficulties may be encountered in it's **vibration analysis**. The number of degrees of freedom

may lead to a mathematically complicated models and make an eigensolution difficult, if not impossible (in large-scale linear systems of equations, equations can number in millions!). such a structure may also be too large for modal testing. Very often, large structural systems are built in parts by firms in various different locations (for instance, the Airbus A380 family) and it is costly to assemble all the parts for vibration testing and analysis.

So, one of the most used technique for the vibration analysis of large complex structure is the method of substructures. This method involves the subdivision of a structural system into a series of subsystems each of which are much simpler substructures. The vibration response characteristics of the overall system are then obtained by a suitable combination of substructures data, where each substructure(or component) has been analyzed independently. This method belongs to the class of domain decomposition methods.

The substructure approach is attractive because it parallels the engineering design process where large structural systems are often designed and manufactured in parts, each part being associated with a different engineering group. The individual substructures are easier to handle than the complete system and their vibration testing and analysis may proceed as independently as possible. Then a change in the design of any substructure need only modify the response data of the changed substructure.

However, since the system is large, only the first eigenmodes are computed (100 to 1000) in real life problems with the modern computing facilities because the smallest eigenvalues (or eigenfrequencies) have a mechanical significance while error increases as the frequency increases

1.2.2 Objectives of this report

The purpose of this report proposed by professor François-Xavier ROUX is to implement the Craig-Bampton method and to assess if we can go further in the spectrum (ie, if we can compute more eigenmodes than usually) at reasonable price and CPU time with respect of the error bounds (ie, limited loss of accuracy)

The second objective of this report concerns the localisation of the eigenvalues in the middle of spectrum (for the *Acoustic problem*) since the eigenfrequencies are not well separated, we will apply the shift-and-invert procedure to the Craig-Bampton reduced matrices.

Parallel implementation will be done for the Craig-Bampton method, and for simplicity in this report, we assume that the computational domain is partitionned into N_s non overlapping substructures, we will use the Message Passing Interface (MPI) to communicate between substructures which are neighbours and share common interface to ensure continuity of the problem.

This report is organized as follows: chapter 2 presents a theoretical background of the elastodynamics problem. In chapter 3 we present a short algebraic description of *the generalized symmetric eigenvalue problem*. In chapter 4, *substructuring and the Craig-Bampton method* are presented including the dual Craig-Bampton [18] a paper by Daniel Rixen (the two figures presenting vibration modes and static modes are taken from this paper). In chapter 5 we discuss the parallel implementation and numerical analysis of the method presented.

Chapter 2

Theoretical background of the elastodynamics problem

2.1 General equation, boundary conditions and initial conditions

Let Ω be a d-dimensional domain ($d = 2, 3$) with a Lipschitz boundary $\partial\Omega$. The external unit normal to $\partial\Omega$ is denoted as $\mathbf{n}$. Let $u(x,t)$ be the displacement at time t of a particle located at point x in Ω.
We consider a structure around a position of static equilibrium taken as reference configuration Ω.
For linear materials, the stress tensor $\sigma(u)$ is defined through the constitutive equations

$$\sigma_{ij}(u) = \mu(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}) + \lambda divu\delta_{ij} \quad , \quad i,j = 1,\dots,d \tag{2.1.1}$$

where $\mu > 0, \lambda \geq 0$ are the lamé coefficients and δ_{ij} is the kronecker's tensor. Lamé coefficients μ and λ are defined by the Young modulus E and the Poisson ratio ν as follows

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad \text{and} \quad \mu = \frac{E}{2(1+\nu)}$$

The symmetric stress tensor σ_{ij} is related to the linearized symmetric strain tensor ε_{ij} by the constitutive equation (Hooke's law)

$$\sigma_{ij} = C_{ijkl}\varepsilon_{kl} \quad , \tag{2.1.2}$$

in wich C_{ijkl} is the fourth-order tensor of the elastic coefficients of the material.

Strain tensor is related to displacement field $\mathbf{u}$ by

$$\varepsilon_{ij}(u) = \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}) \tag{2.1.3}$$

Then, equation (2.1.1) can be rewritten as

$$\sigma_{ij}(u) = 2\mu\varepsilon_{ij}(u) + \lambda div u\delta_{ij} \quad , \quad i,j = 1,\ldots,d \tag{2.1.4}$$

Denoting as $f(x,t)$ a given body force field applied in Ω, the elastodynamic equation is written as

$$\rho\frac{\partial^2 u_i}{\partial t^2} - \frac{\partial \sigma_{ij}}{\partial x_j}(u) = f_i(x,t) \quad \text{in } \Omega \quad , t > 0 \tag{2.1.5}$$

or equivalently

$$\rho\frac{\partial^2 u_i}{\partial t^2} - div\sigma(u) = f_i(x,t) \quad \text{in } \Omega \quad , t > 0 \tag{2.1.6}$$

in wich $\rho(x)$ is the mass density field defined on Ω. For prescribed displacement $u_D(x,t)$ on the part Γ_D of the boundary $\partial\Omega$ of Ω, we have the boundary condition

$$u(x,t) = u_D(x,t) \quad \text{on } \Gamma_D \tag{2.1.7}$$

For a given surface force $F(x,t)$ applied to the part $\Gamma = \partial\Omega\backslash\Gamma_D$, we have the boundary condition

$$\sigma_{ij}(x,t)n_j(x) = F_i(x,t) \quad \text{on } \Gamma \tag{2.1.8}$$

Finally, the above equations must be completed by the Cauchy initial conditions

$$u(x,0) = u_0(x) \qquad , \qquad \frac{\partial u}{\partial t} = v_0(x) \quad \text{in } \Omega \tag{2.1.9}$$

where u_0 is the given displacement field and v_0 is the given velocity field.

2.2 the homogeneous problem

We consider the following time boundary value problem for wich $u_D = 0$ on Γ_D, $f = 0$ in Ω and $F = 0$ on Γ. Then from Eqs. (2.1.5), (2.1.7) and (2.1.8), we deduce the boundary value problem in terms of u

$$\begin{cases} \rho\dfrac{\partial^2 u_i}{\partial t^2} - \dfrac{\partial \sigma_{ij}}{\partial x_j} = 0 \quad \text{in } \Omega \quad , t > 0 \\ u(x,t) = 0 \quad \text{on } \Gamma_D \\ \sigma_{ij}(u)n_j = 0 \quad \text{on } \Gamma \\ u(x,0) = u_0(x) \qquad , \qquad \dfrac{\partial u}{\partial t} = v_0(x) \quad \text{in } \Omega \end{cases} \tag{2.2.1}$$

in wich u_0 and v_0 are given initial conditions. Eq (2.2.1) correspond to a structure fixed on Γ_D. For a *free structure* ($\Gamma_D = \emptyset$, the second equation of (2.2.1) does not exist and the third equation (robin condition) holds on the global boundary $\Gamma = \partial\Omega$ of domain Ω.

2.3 Associated spectral problem

We consider all the solutions $u(x,t) = u(x)exp(i\omega t)$ (ie, we take the Fourier transform) of the boundary value problem (2.2.1) where i is the imaginary complex number. The spectral problem consists in searching for (ω, u) with $u \neq 0$ satisfying the boundary value problem

$$\begin{cases} -\omega^2 \rho u_i - div\sigma(u) = 0 \quad \text{in } \Omega \quad , t > 0 \\ u = 0 \quad \text{on } \Gamma_D \\ \sigma_{ij}(u) n_j = 0 \quad \text{on } \Gamma \end{cases} \tag{2.3.1}$$

The angular frequencies ω_α in eq (2.3.1) are called *eigenfrequencies* (or natural frequencies) and u_α are the *structural modes* (or eigenmodes or mode shapes of vibration)

2.4 Variational formulation of the spectral problem

2.4.1 construction of the variational fomulation

Multiplying Eq. (2.3.1) by an arbitrary function $v \in (H^1(\Omega))^3$ (where $H^1(\Omega)$ is sobolev space, see [2]) and integrating by parts over domain Ω yields

$$-\omega^2 \int_\Omega \rho u_i v_i dx - \int_\Omega \sigma_{ij,j}(u) v_i dx = 0 \tag{2.4.1}$$

where for simplicity of manipulation, we use the tensor notation for derivative such that $a_{,j} = \partial a / \partial x_j$.
The second term on the left-hand side of Eq. (2.4.1) is transformed using the following identity

$$-\int_\Omega \sigma_{ij,j}(u) v_i dx = \int_\Omega \sigma_{ij}(u) v_{i,j} dx - \int_\Omega \left(\sigma_{ij}(u) v_i\right)_{,j} dx \tag{2.4.2}$$

The second integral on the right-hand side of Eq. (2.4.2) is transformed to an integral on $\partial\Omega$ using Stoke's formula

$$\int_\Omega \left(\sigma_{ij}(u) v_i\right)_{,j} dx = \int_{\partial\Omega} \sigma_{ij}(u) n_j v_i ds \qquad , \tag{2.4.3}$$

where ds is the surface element. From Eqs. (2.4.2) and (2.4.3), we deduce the Green's formula see [4]

$$-\int_\Omega \sigma_{ij,j}(u)v_i dx = \int_\Omega \sigma_{ij}(u)v_{i,j}dx - \int_{\partial\Omega} \sigma_{ij}(u)n_j v_i ds \qquad . \tag{2.4.4}$$

Substituting the left-hand side of Eq. (2.4.4) into Eq. (2.4.1) yields

$$-\omega^2 \int_\Omega \rho u_i v_i dx + \int_\Omega \sigma_{ij}(u)v_{i,j}dx - \int_{\partial\Omega} \sigma_{ij}(u)n_j v_i ds \qquad . \tag{2.4.5}$$

Then, we transform the second term of the left-hand side of Eq. (2.4.4) in order to get a symmetric formulation in terms of u and v. From the symmetry of the stress tensor and the definition of the strain tensor given by Eq. (2.1.3), we deduce the equalities

$$\sigma_{ij}v_{i,j} = \frac{1}{2}\sigma_{ij}(u)(v_{i,j} + v_{j,i}) = \sigma_{ij}(u)\varepsilon_{ij}(v) \qquad .$$

Substitution into Eq. (2.4.5) yields

$$-\omega^2 \int_\Omega \rho u_i v_i dx + \int_\Omega \sigma_{ij}(u)\varepsilon_{ij}(v)dx - \int_{\partial\Omega} \sigma_{ij}(u)n_j v_i ds = 0 \tag{2.4.6}$$

And finally, we introduce the *admissible function space*. we will distinguish two cases.

structure fixed onΓ_D. In this case, $\Gamma_D \neq \emptyset$ and the admissible function space satisfy the constraint on u defined by the second line of Eq. (2.3.1) and consequently is the subspace $H^1_{0,\Gamma_D}(\Omega)$ defined by

$$H^1_{0,\Gamma_D}(\Omega) = \left\{ u \in (H^1(\Omega))^3 \quad ; u = 0 \quad \text{on} \quad \Gamma_D \right\}. \tag{2.4.7}$$

Let v be any test function in $H^1_{0,\Gamma_D}(\Omega) \subset (H^1(\Omega))^3$. Since $v = 0$ on Γ_D and using the third line of Eq. (2.3.1), Eq. (2.4.5) yields

$$-\omega^2 \int_\Omega \rho u_i v_i dx + \int_\Omega \sigma_{ij}(u)\varepsilon_{ij}(v)dx = 0 \quad , \quad \forall v \in H^1_{0,\Gamma_D}(\Omega) \tag{2.4.8}$$

Free structure. In this case, $\Gamma_D = \emptyset$ and the second line of Eq. (2.3.1) does not exist. Therefore, the admissble function space of the problem is $(H^1(\Omega))^3$ (there is no constrain on u).Let v be any test function in $(H^1(\Omega))^3$. Due to the third line of Eq. (2.3.1), the integral over $\partial\Omega$ in Eq. (2.4.6) is equal to zero and Eq. (2.4.6) yields

$$-\omega^2 \int_\Omega \rho u_i v_i dx + \int_\Omega \sigma_{ij}(u)\varepsilon_{ij}(v)dx = 0 \quad , \quad \forall v \in (H^1(\Omega))^3 \tag{2.4.9}$$

2.4.2 Variational formulation

Structure fixed onΓ_D.
The variational formulation of the spectral problem defined by Eq. (2.3.1) is stated as follows.

$$\begin{cases} \text{find } (\omega^2, u) \in \mathbb{R} \times H^1_{0,\Gamma_D}(\Omega) \text{ such that} \\ \displaystyle\int_\Omega \sigma_{ij}(u)\varepsilon_{ij}(v)dx = \omega^2 \int_\Omega \rho uv dx \qquad \forall v \in H^1_{0,\Gamma_D}(\Omega) \end{cases} \tag{2.4.10}$$

For *structure fixed on* Γ_D, $\omega^2 > 0$

Free structure.
The variational formulation of the spectral problem defined by Eq. (2.3.1) without the Dirchlet condition (line 2) is stated as follows.

$$\begin{cases} \text{find } (\omega^2, u) \in \mathbb{R} \times (H^1(\Omega))^3 \text{ such that} \\ \displaystyle\int_\Omega \sigma_{ij}(u)\varepsilon_{ij}(v)dx = \omega^2 \int_\Omega \rho uv dx \qquad \forall v \in (H^1(\Omega))^3 \end{cases} \tag{2.4.11}$$

For *free structure*, $\omega^2 \geq 0$

2.5 Associated Linear operators and some algebraic properties

2.5.1 Mass operator

We introduce the bilinear form $m(u, v)$, called the structural mass bilinear form, defined by

$$m(u, v) = \int_\Omega \rho uv dx \tag{2.5.1}$$

which is symmetric because

$$m(u, v) = m(v, u) \tag{2.5.2}$$

and which is positive definite because $\rho(x) > 0$ (for all $u \neq 0$, we have $m(u, u) > 0$). This result holds for the fixed and free interface cases. We then introduce the linear operator M, called the *mass operator*, such that

$$< Mu, v >= m(u, v) \tag{2.5.3}$$

the angle brackets here denote the duality product between $(H^1(\Omega))^3$ and it's dual, since the bilinear form $m(u, v)$ is continuous on $(L^2(\Omega))^3 \times (L^2(\Omega))^3$,then M is continuous from $(H^1(\Omega))^3$ to it's dual $((H^1(\Omega))^3)'$. More mathematical tools concerning Sobolev's spaces can be found in [2].

2.5.2 Stiffness Operator

Using Eq. (2.1.2), we introduce the bilinear form $k(u, v)$, called the structural stiffness bilinear form, defined by

$$k(u,v) = \int_\Omega \sigma_{ij}(u)\varepsilon_{ij}(v)dx \tag{2.5.4}$$

or equivalently

$$k(u,v) = \int_\Omega C_{ijkl}\varepsilon_{ij}(u)\varepsilon_{ij}(v)dx \tag{2.5.5}$$

From the symmetry property $C_{ijkl} = C_{klij}$ of the tensor of elastic coefficients of the material, we deduce that bilinear form k is symmetric:

$$k(u,v) = k(v,u) \tag{2.5.6}$$

From the positive definiteness of tensor C_{ijkl}, we deduce that bilinear form k is positive semi-definite. The fixed and free cases must be distinguished.

Structure fixed onΓ_D.
Bilinear form $k(u, v)$ defined on $H^1_{0,\Gamma_D}(\Omega) \times H^1_{0,\Gamma_D}(\Omega)$ is positive definite:

$$k(u,u) > 0 \qquad \text{for all} \qquad u \neq 0 \tag{2.5.7}$$

We then introduce the linear operator K, called the stiffness operator, such that

$$< Ku, v >= k(u,v) \tag{2.5.8}$$

Free structure.
Bilinear form $k(u, v)$ defined on $(H^1(\Omega))^3 \times (H^1(\Omega))^3$ is positive semi-definite

$$k(u,u) \geq 0 \qquad \text{for all} \qquad u \neq 0 \tag{2.5.9}$$

In effect, from Eq. (2.5.4), we deduce that $k(u, u) = 0$ implies tensor $\varepsilon_{ij}(u) = 0$ due to the positive definiteness of tensor C_{ijkl}. Since Ω is a connected domain, the general solution of the system of equations $\varepsilon_{ij}(u) = 0$ is written as

$$u_{rig}(x) = a + b \times x \quad , \quad \forall x \in \Omega \tag{2.5.10}$$

in which a and b are two arbitrary constant vectors in $\mathbb{R}^3$. We denote as H^1_{rig} the subset of $(H^1(\Omega))^3$ spanned by a and b. This space is of dimension 6 and u_{rig} belongs to H^1_{rig}.
From a mechanical point of view, u_{rig} represents the *rigid body displacement field*(in the linearized theory). We then introduce the linear operator K, called as above the *stiffness operator*, such that

$$< Ku, v >= k(u,v) \tag{2.5.11}$$

Which is symmetric positive semi-definite.

2.5.3 Rewriting the variational formulation

Structure fixed onΓ_D.
Using the above notations, the variational formulation defined by Eq. (2.4.10) can be rewritten as follows.

$$\begin{cases} \text{find } (\omega^2, u) \in \mathbb{R} \times H^1_{0,\Gamma_D}(\Omega) \text{ such that} \\ k(u,v) = \omega^2 m(u,v) \qquad \forall v \in H^1_{0,\Gamma_D}(\Omega) \end{cases} \tag{2.5.12}$$

Free structure.
Similarly, the variational formulation defined by Eq. (2.4.11) can be rewritten as follows.

$$\begin{cases} \text{find } (\omega^2, u) \in \mathbb{R} \times (H^1(\Omega))^3 \text{ such that} \\ k(u,v) = \omega^2 m(u,v) \qquad \forall v \in (H^1(\Omega))^3 \end{cases} \tag{2.5.13}$$

2.6 Basic properties of the eigenfrequencies and structural modes for a structure fixed on Γ_0

The eigenvalue problem

setting $\lambda = \omega^2$, the spectral problem defined by Eq. (2.5.12) is rewritten as the following eigenvalue problem.

$$\begin{cases} \text{find } (\lambda, u) \in \mathbb{R} \times H^1_{0,\Gamma_D}(\Omega) \text{ such that} \\ k(u,v) = \lambda m(u,v) \qquad \forall v \in H^1_{0,\Gamma_D}(\Omega) \end{cases} \tag{2.6.1}$$

Remark 2.6.1 *All the variational formulations seen so far stand for $u \neq 0$ (trivial solution)*

Countable number of positive eigenvalues.

Since Ω is a bounded domain and from properties indicated in section 5 for a fixed structure (symmetry and positive definiteness of the mass and stiffness bilinear forms), it can be shown that there exists an increasing sequence of positive eigenvalues

$$0 \leq \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_\alpha \leq \ldots \tag{2.6.2}$$

The positivity of λ_α can be directly deduced From Eq. (2.6.1) by taking $u = v$ (the same way to show the ellipticity of a bilinear form see [4]). In addition, λ_α is strictly positive because $u = 0$ on Γ_D (in the case of a fixed structure, there are no rigid body modes).

Completeness of the eigenfunctions.

Let u_α be the eigenfunction associated with eigenvalue λ_α. It can be shown that the eigenfunctions $\{u_\alpha\}_{\alpha\geq 1}$ form a complete set in $H^1_{0,\Gamma_D}(\Omega)$ which means that an arbitrary function u belonging to $H^1_{0,\Gamma_D}(\Omega)$ can be expanded as

$$u = \sum_{\alpha=1}^{+\infty} q_\alpha u_\alpha \tag{2.6.3}$$

in which, $\{q_\alpha\}_\alpha$ is a sequence of real numbers.
Real numbers $\mathbf{q}_\alpha$

Orthogonality of the eigenfunctions.

It can be shown that the sequence of eigenfunctions $\{u_\alpha\}_\alpha$ satisfies the orthogonality conditions with respect to mass and stiffness,

$$m(u_\alpha, u_\beta) = \mu_\alpha \delta_{\alpha\beta} \qquad , \tag{2.6.4}$$

$$k(u_\alpha, u_\beta) = \mu_\alpha \omega_\alpha^2 \delta_{\alpha\beta} \qquad , \tag{2.6.5}$$

in which

$$\omega_\alpha = \sqrt{\lambda_\alpha} \tag{2.6.6}$$

and where μ_α is a positive real number depending on the normalization of eigenfunctions u_α, because the eigenfunctions are defined up to a multiplicative constant.
μ_α is sometimes referred as **the generalized mass**.

Remark 2.6.2 *In structural vibration, as we will see in [Chapter 4],usually $\mu_\alpha = 1$ when eigenfunctions are normalized with respect to the mass.*

Linear operator equation.

The linear operator equation corresponding to the variational formulation defined by Eq. (2.6.1) is written as

$$Ku = \lambda M u \qquad , \qquad u \in H^1_{0,\Gamma_D}(\Omega) \tag{2.6.7}$$

This problem is called **a generalized eigenvalue problem** because operator M is not the identity operator. The algebraic properties defined by Eqs. (2.6.4) and (2.6.5) can be rewritten as

$$< Mu_\alpha, u_\beta >= \mu_\alpha \delta_{\alpha\beta} \qquad , \tag{2.6.8}$$

$$< Ku_\alpha, u_\beta >= \mu_\alpha \omega_\alpha^2 \delta_{\alpha\beta} \qquad , \tag{2.6.9}$$

2.7 Basic properties of the eigenfrequencies and structural modes for a free structure

2.7.1 The eigenvalue problem

Setting $\lambda = \omega^2$, the spectral problem defined by Eq. (2.5.13)

$$\left\{ \begin{array}{l} \text{find } (\lambda, u) \in \mathbb{R} \times (H^1(\Omega))^3 \text{ such that} \\ k(u,v) = \lambda m(u,v) \qquad \forall v \in (H^1(\Omega))^3 \end{array} \right. \tag{2.7.1}$$

2.7.2 Rigid body modes (solutions for $\lambda = 0$)

we show that there exist solutions of the type $\{\lambda = 0, u \neq 0\}$.
Taking $\lambda = 0$ in Eq. (2.7.1) yields $k(u,v) = 0$ for all $v \in (H^1(\Omega))^3$. The general solution of this problem corresponds to **the rigid body displacement fields** $u_{rig} \in H^1_{rig}$. Since the dimension of H^1_{rig} is 6, then $\lambda = 0$ can be considered as *zero eigenvalue* of multiplicity 6, denoted $\lambda_{-5}, \ldots, \lambda_0$. Let $u_{-5}, \ldots, u_0$ be the corresponding eigenfunctions which are constructed such that the following orthogonality conditions with respect to the mass and stiffness bilinear forms are satisfied, for α and β in $\{-5, \ldots, 0\}$,

$$m(u_\alpha, u_\beta) = \mu_\alpha \delta_{\alpha\beta} \qquad , \tag{2.7.2}$$

$$k(u_\alpha, u_\beta) = 0 \qquad . \tag{2.7.3}$$

These eigenfunctions called the ***rigid body modes***, form a basis of $H^1_{rig} \subset (H^1(\Omega))^3$ and any rigid body displacement $u_{rig} \in H^1_{rig}$ can be expanded as

$$u_{rig} = \sum_{\alpha=-5}^{0} q_\alpha u_\alpha \tag{2.7.4}$$

2.7.3 Elastic structural modes (solutions for $\lambda \neq 0$)

We introduce H^1_{elas} of $(H^1(\Omega))^3$ corresponding to all the displacement field u_{elas} belonging to $(H^1(\Omega))^3$ and which do not belong to H^1_{rig} ($H^1_{elas} = (H^1(\Omega))^3 \backslash H^1_{rig}$). Consequently, $k(u_{elas}, v_{elas})$ defined on $H^1_{elas} \times H^1_{elas}$ is positive definite and we then have

$$k(u_{elas}, u_{elas}) > 0 \qquad , \qquad \forall u_{elas} \neq 0 \in H^1_{elas} \tag{2.7.5}$$

Eigenvalue problem restricted to H^1_{elas}

The eigenvalue problem defined by Eq. (2.7.1) and restricted to space H^1_{elas} is written as follows.

$$\begin{cases} \text{find } (\lambda \neq 0, u) \in \mathbb{R} \times H^1_{elas}(\Omega) \text{ such that} \\ k(u_{elas}, v_{elas}) = \lambda m(u_{elas}, v_{elas}) \qquad \forall v_{elas} \in H^1_{elas}(\Omega) \end{cases} \tag{2.7.6}$$

Countable number of positive eigenvalues.

We consider the solutions of Eq. (2.7.6). Since Ω is a bounded domain and due to the symmetry and positive definiteness of bilinear forms m and k on $H^1_{elas} \times H^1_{elas}$, it can be proved that there exists an increasing sequence of positive eigenvalues

$$0 < \lambda_1 \leq \lambda_2 \leq \ldots \lambda_\alpha \leq \ldots \tag{2.7.7}$$

In addition, any multiple positive eigenvalue has a finite multiplicity (which means that a multiple positive eigenvalue is repeated a finite number of times). The positivity of eigenvalues can easily be proved by taking $v_{elas} = u_{elas}$ in Eq. (2.7.6).

Completeness of the eigenfunctions corresponding to the positive eigenvalues.

Let u_α be the eigenfunction associated with eigenvalue $\lambda_\alpha > 0$. It can be shown that the eigenfunctions $\{u_\alpha\}_{\alpha \geq 1}$ form a complete set in H^1_{elas} and that an arbitrary function u_{elas} belonging to H^1_{elas} can be expanded as

$$u_{elas} = \sum_{\alpha=1}^{+\infty} q_\alpha u_\alpha \tag{2.7.8}$$

in which, $\{q_\alpha\}_\alpha$ is a sequence of real numbers (generalized coordinates as seen above).
These eigenfunctions are called **the elastic structural modes**.

Orthogonality of the eigenfunctions corresponding to the positive eigenvalues.

It can be shown that the sequence of eigenfunctions $\{u_\alpha\}_\alpha$ in H^1_{elas} corresponding to the positive eigenvalues satisfies the following orthogonality conditions with respect to the mass and stiffness bilinear forms,

$$m(u_\alpha, u_\beta) = \mu_\alpha \delta_{\alpha\beta} \qquad , \tag{2.7.9}$$

$$k(u_\alpha, u_\beta) = \mu_\alpha \omega_\alpha^2 \delta_{\alpha\beta} \qquad , \tag{2.7.10}$$

in which

$$\omega_\alpha^2 = \lambda_\alpha \tag{2.7.11}$$

and where μ_α is a positive real number depending on the normalization of eigenfunctions u_α, because the eigenfunctions are defined up to a multiplicative constant.

2.7.4 Orthogonality between the elastic structural modes and the rigid body modes

Substituting any elastic structural modes $u = u_\alpha \in H^1_{elas} \subset (H^1(\Omega))^3$ into Eq. (2.7.1) yields

$$k(u_\alpha, v) = \lambda_\alpha m(u_\alpha, v) \qquad , \quad \forall v \in (H^1(\Omega))^3 \tag{2.7.12}$$

in which $\lambda_\alpha \neq 0$. Taking $v = u_{rig} \in H^1_{rig} \subset (H^1(\Omega))^3$ in Eq. (2.7.12) yields

$$k(u_\alpha, u_{rig}) = \lambda_\alpha m(u_\alpha, u_{rig}) \qquad . \tag{2.7.13}$$

For rigid body modes(solutions for $\lambda = 0$), we deduce that

$$k(u_{rig}, v) = 0 \qquad , \qquad \forall v \in (H^1(\Omega))^3 \tag{2.7.14}$$

Taking $v = u_\alpha$ in Eq. (2.7.14) and using the symmetry of k yields

$$k(u_\alpha, u_{rig}) = 0 \qquad . \tag{2.7.15}$$

From Eqs. (2.7.13) and (2.7.15), we deduce that

$$m(u_\alpha, u_{rig}) = 0 \qquad . \tag{2.7.16}$$

Equations (2.7.14) and (2.7.15) express the orthogonality between the elastic structural modes and the rigid body modes. Substituting Eq. (2.5.10) into (2.7.15) yields

$$\int_\Omega u_\alpha(x)\rho(x)dx = 0 \qquad , \tag{2.7.17}$$

$$\int_\Omega x \times u_\alpha(x)\rho(x)dx = 0 \qquad , \tag{2.7.18}$$

Let Ω_α be the domain deduced from Ω such that $\Omega_\alpha = \{x_\alpha = x + u_\alpha(x), x \in \Omega\}$ in which u_α is any elastic structural mode. For a free structure, Eq (2.7.17) implies that the inertial center of Ω coincides with the inertial center of Ω_α.

2.7.5 Expansion of the displacement field using the rigid body modes and the elastic structural modes

From the above results, we deduce that any displacement field u in $(H^1(\Omega))^3$ has the following decomposition

$$u = u_{rig} + u_{elas} \tag{2.7.19}$$

Substituting Eqs. (2.7.4) and (2.7.8) into Eq. (2.7.19) yields

$$u = \sum_{\alpha=-5}^{+\infty} q_\alpha u_\alpha \qquad , \tag{2.7.20}$$

in which $\{q_\alpha\}_{\alpha \geq -5}$ is a sequence of real numbers.
Equation (2.7.19) can be written as the direct sum

$$H^1(\Omega))^3 = H^1_{rig} \oplus H^1_{elas} \tag{2.7.21}$$

$(H^1_{rig} \cap H^1_{elas} = \{0\})$

2.7.6 Linear operator equation

The linear operator equation corresponding to the variational formulation defined by Eq. (2.7.1) is written as

$$Ku = \lambda Mu \qquad , \qquad u \in (H^1(\Omega))^3 \tag{2.7.22}$$

and correspond to **a generalized eigenvalue problem**. The orthogonality conditions of the rigid body modes and structural modes can be rewritten as

$$< Mu_\alpha, u_\beta >= \mu_\alpha \delta_{\alpha\beta} \qquad , \tag{2.7.23}$$

$$< Ku_\alpha, u_\beta >= \mu_\alpha \omega^2_\alpha \delta_{\alpha\beta} \qquad , \tag{2.7.24}$$

in which $\alpha \geq -5$ and $\beta \geq -5$

2.8 Matrix formulation of the generalized eigenvalue problem

The finite element discretization of the three-dimensional continuous problem deduced from the eigenvalue problem defined by Eqs. (2.6.1) or (2.7.1)leads

to the following matrix equation of the generalized symmetric eigenvalue problem

$$[K]\,U = \lambda\,[M]\,U \tag{2.8.1}$$

in which $U = (U_1, \ldots, U_n)$ is the vector of the DOFs(Degrees of freedom) which are the values of the displacement field at the nodes of the finite element mesh of domain Ω. The $(n \times n)$ mass matrix $[M]$ is symmetric positive definite. For a stucture fixed on Γ_D, the $(n \times n)$ stiffness matrix $[K]$ is symmetric positive definite and then invertible. For a free structure, stiffness matrix $[K]$ is symmetric positive semi-definite and singular of rank $n - 6$.
Strong mathematical analysis of the finite element method from variational formulation can be found in [4]

Chapter 3

The generalized symmetric eigenvalue problem: a short algebraic description

The pair of matrices K, M in the following equation

$$[K]\,U = \lambda\,[M]\,U \tag{3.0.1}$$

is often referred to as a **matrix pencil** or **matrix pair**. The most natural way of defining eigenvalues of a matrix pair is to think of them as pairs (α, β) of complex numbers.
Thus, (α, β) is an eigenvalue of the pair (K, M) if by definition there is a vector u called an associated eigenvector, such that

$$\beta K u = \alpha M u \tag{3.0.2}$$

Equivalently, (α, β) is an eigenvalue if and only if

$$det(\beta K - \alpha M) = 0 \tag{3.0.3}$$

Remark 3.0.1 *:*

- *we note that the trivial pair* $(0, 0)$ *always satisfies the definition*
- *Another difficulty is that there are infinitely many pairs* (α, β) *which can be termed generalized eigenvalues to represent the same 'standard eigenvalue'. this is because we can multiply a given* (α, β) *by any complex scalar and still get an eigenvalue for the pair* (K, M)

Remark 3.0.2 *:*
The standard definition of an eigenvalue corresponds to the case where $B = I$ *and* $\beta = 1$.

3.1 Some definitions

definition:

A matrix pair (K, M) is called singular if $\beta K - \alpha M$ is singular for all α, β.
A matrix pair that is not singular is said to be regular.
singular pairs" may cause numerical difficulties.

definition:

If X and Y are two singular matrices, the pair (KYX, YMX) is said to be equivalent to the pencil (K, M).

There are three known ways out of difficulty. A popular way (used in structural engineering for instance) is to take the ratio α/β as an eigenvalue, which corresponds to selecting a particular pair $(\alpha, 1)$ in the set. When β is zero, the eigenvalue takes the value infinity and this may not be usefull in numerical point of view. A second way would be to use pairs (α, β) but scale them by some norm in in $\mathbb{C}^2$, eg so that $|\alpha|^2 + |\beta|^2 = 1$. Finaly, the third way is t denote by $< \alpha, \beta >$ the set of all pairs that satisfy (3.0.2). The eigenvalue is then a set instead of element in $\mathbb{C}^2$. This set is referred to as *"generalized eigenvalues"* (that is $\lambda =< \alpha, \beta >$).

Considering the first way, say $\alpha/\beta = \lambda$, we give some properties....
In structural engineering, the generalized eigenvalue under consideration is of the form see (2.6.7)

$$\begin{cases} \text{find } (\lambda, u) \in \mathbb{R} \quad \textit{and} \quad 0 \neq u \in V \quad \textit{such} \quad \textit{that} \\ Ku = \lambda M u \end{cases} \tag{3.1.1}$$

where V is an abstract finite-dimensional hilbert space with inner product $(.,.)$ see [Chapter2]

K and M are linear continuous symmetric operators defined on V, K is positive definite and M is positive semi-definite.

Consider the eigenvalue problem:

$$\begin{cases} \text{find } (\lambda, u) \in \mathbb{R} \quad \textit{and} \quad 0 \neq u \in V \quad \textit{such} \quad \textit{that} \\ k(u,v) = \lambda m(u,v), \qquad \forall v \in V \end{cases} \tag{3.1.2}$$

If (3.1.2) corresponds to the eigenvalue problem for a self adjoint elliptic partial differential operator, it will typically admit an infinite set of non-decreasing eigenvalues. Without loss of such generality, let the eigenvalues of (3.1.2) (counted with their multiplicity satisfy

$$0 < \lambda_1 \leq \lambda_2 \leq \dots \tag{3.1.3}$$

In particular, we note that λ_1 is the minimum of the Rayleigh quotient over V

$$\lambda_1 = \inf_{0 \neq u \in V} \frac{k(u,v)}{m(u,v)} \tag{3.1.4}$$

where **Rayleigh quotient** is the ratio:

$$\mathcal{R}(u) = \frac{k(u,v)}{m(u,v)} \tag{3.1.5}$$

3.2 Reduction to standard form

In several papers in literature, a traditional appraoch is to transform the problem (3.0.1) to be an ordinary eigenvalue problem.
For example, when M is non singular, we can transform the original system

$$Ku = \lambda Mu \tag{3.2.1}$$

into

$$M^{-1}Ky = \lambda y \qquad with \qquad u = M^{-1}y \tag{3.2.2}$$

this simply amounts to multiplying both matrices in the pair by M^{-1}, thus transforming (K, M) into the equivalent pair $(M^{-1}K, I)$.

When K and M are both hermitian and in addition, B is positive definite, a better alternative may be to exploitthe Cholesky factorization of M. If $M = LL^T$, we get after multiplying from the left by L^{-1} and from the right by L^{-T}, the standard eigenvalue problem

$$L^{-1}KL^{-1}y = \lambda y \tag{3.2.3}$$

In some papers, the inverted eigenvalue problem is considered,

$$Mu = \lambda^{-1}Ku \tag{3.2.4}$$

the reason is that, for the problems in hand, the smallest eigenvalues are not well separated, thus the reciprocals of the wanted eigenvalues are well separated.

3.3 Shift and Invert

The shift and invert spectral transformation is used to enhance convergence to a desired portion of the spectrum. If $(\lambda,\ u)$ is an eigenpair for $(K,\ M)$ and $\sigma \neq \lambda$ then

$$(K - \sigma M)^{-1} M u = \frac{1}{\lambda - \sigma} u \tag{3.3.1}$$

which can be rewritten

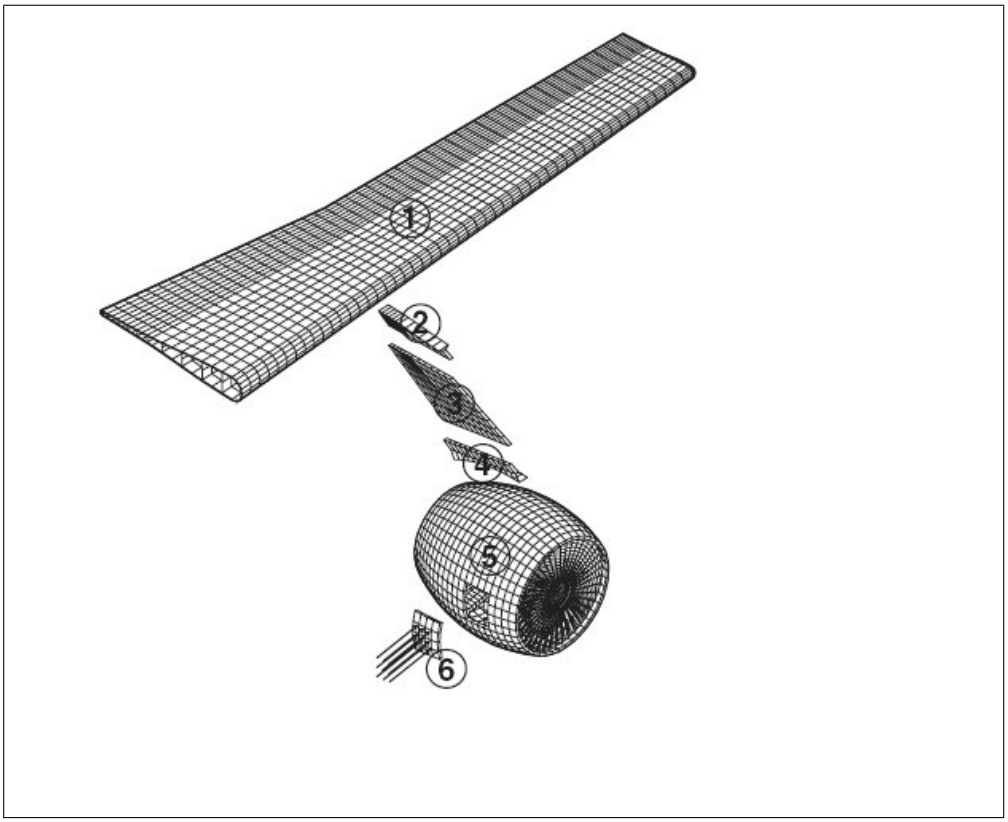
$$Cx = \nu x \tag{3.3.2}$$

where $C = (A - \sigma B)^{-1} B$ and $\nu = \frac{1}{\lambda - \sigma}$.

This transformation is effective for finding eigenvalues near σ since the *nev* eigenvalues ν_j of C that are largest in magnitude correspondond to the *nev* eigenvalues λ_j of the original problem that are nearest to the shift σ in absolute value.These transformed eigenvalues of largest magnitude are precisely the eigenvalues that are easy to compute with iterative methods. Once they are found, they may be transformed back to eigenvalues of the original problem.

Chapter 4

Substructuring and the CRAIG-BAMPTON method



4.1 Dynamic substructuring

When a large complex structural system must be analysed for it's response to dynamic excitation, the whole discretization of the system can lead to very costly analysis, since the stiffness and mass matrices K and M in a finite element model

$$K_h x = \lambda M_h x, \tag{4.1.1}$$

of a structure under consideration are usually very large.

So the substructuring methods consist in decomposing the global domain Ω into several subdomains $\Omega_i (\Omega = \bigcup_{i=1}^{N_s} \Omega_i)$ and ensure the continuity through interfaces between the subdomains. The interface between two subdomains Ω_i and Ω_j will be noted $\Gamma_{ij} = \partial\Omega_i \bigcap \partial\Omega_j$
In this report, for simplicity, we assume that Ω is a d-dimensional domain $(d = 2, 3)$ with a lipschitz boundary $\partial\Omega$ whose outer unit normal direction is denoted $\mathbf{n}$. we also assume that Ω is partitionned into non-overlapping subdomains.

In practice, the original problem is refomulated upon each subdomain Ω_i, yielding a family of sub-problems of reduced size that are coupled one to another through the value of the unkowns at subdomains interfaces.
The continuity at the interface is one of the crucial points of the domain decomposition methods, because the solution obtained on a particular subdomain Ω_i must satisfy the continuity conditions.

Very often, the interface coupling is removed at the expense of introducing an iterative process among subdomains, yielding at each step independant sub-problems (of lower complexity) upon subdomains, which can be efficiently faced by a multi-processor system. There are several papers in the litterature concerning iterative substucturing method

To illustrate in structural mechanics, the domain decomposition method split the unknowns into internal unknowns and boundary unknowns. The boundary unknowns belong to two or more subdomains while the internal unknowns belong to only one subdomain.

The domain decomposition methods can be classified as the primal and the dual methods. The primal methods means that the original unknowns are used during all computation (eg: **the Schur complement method**, applied

into mechanical problem, the nodal displacement are the unknowns of the original problem as well as in the condensed problem). On the other hand, the dual methods [11] [chapter 5] defines the Lagrange multipliers which are the forces λ_i to be enforced on domain boundaries to ensure displacement continuity accross subdomains.

We emphasize that, substructuring could also be understood in a purely algebraic sense (ie it is not obtained by a domain decomposition of a real structure).

Remark 4.1.1 *The interface problem can be defined in terms of the Steklov-Poincaré operator [17]*

4.1.1 The principle of the Schur complement method

Consider a coercive partial differential equation like the simple Poisson problem

$$\left\{ \begin{array}{rcll} -\Delta u & = & f & \text{in } \Omega \\ u & = & 0 & \text{on } \partial\Omega \end{array} \right. \tag{4.1.2}$$

The variationnal form of equation (4.1.2) is

$$\left\{ \begin{array}{l} \text{find } u \in H_0^1(\Omega) \text{ such that} \\ \displaystyle\int_\Omega \nabla u \nabla v = \int_\Omega f v \qquad \forall v \in H_0^1 \end{array} \right. \tag{4.1.3}$$

the discretization of equation (4.1.3) using a finite element approximation leads to an algebraic system of equations:

$$Kx = b, \tag{4.1.4}$$

where the stiffness matrix K is a sparse symmetric positive definite matrix. If we consider the computational domain as a global mesh, the natural way is to split the computational domain Ω into N_s subdomains $\Omega_i^{(s)}, i = 1, ...N_s$. For each substructure, we partition the unknowns into internal degrees of freedom (d.o.f) designated by the subscript i, and interface boundary degrees of freedom designated by the subscript b.The substructure static equations

of equilibrium can be written in matrix form as:

$$\begin{bmatrix} K_{ii}^{(1)} & 0 & \dots & 0 & K_{ib}^{(1)} L_b^{(1)} \\ 0 & \ddots & & & \vdots \\ \vdots & & \ddots & & \vdots \\ 0 & & & K_{ii}^{(N_s)} & K_{ib}^{(N_s)} L_b^{(N_s)} \\ L_b^{(1)^T} K_{ib}^{(1)^T} & \dots & \dots & L_b^{(N_s)^T} K_{ib}^{(N_s)^T} & \sum_{s=1}^{N_s} L_b^{(s)^T} K_{bb}^{(s)} L_b^{(s)} \end{bmatrix} \begin{bmatrix} x_i^{(1)} \\ \vdots \\ \vdots \\ x_i^{(N_s)} \\ x_b \end{bmatrix} = \begin{bmatrix} b_i^{(1)} \\ \vdots \\ \vdots \\ b_i^{(N_s)} \\ b_b \end{bmatrix} \tag{4.1.5}$$

where

$$b_b = \sum_{s=1}^{N_s} L_b^{(s)} b_b^{(s)}$$

and x_b is the complete vector of interface d.o.f

$$x^{(s)} = \begin{bmatrix} x_i^{(s)} \\ x_b^{(s)} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & L_b^{(s)} \end{bmatrix} \begin{bmatrix} x_i^{(s)} \\ x_b \end{bmatrix} \tag{4.1.6}$$

$L_b^{(s)}$ is the localisation operator to substructure (s), which extracts from a given vector those d.o.f which are attached to the interface of substructure (s).
Applying a static condensation to the substructure internal d.o.f transforms Eqs (4.1.5) into the following **interface** problem:

$$\sum_{s=1}^{N_s} \left(L_b^{(s)^T} K_{bb}^{(s)} L_b^{(s)} - L_b^{(s)^T} K_{ib}^{(s)^T} K_{ii}^{(s)^{-1}} K_{ib}^{(s)} L_b^{(s)} \right) x_b = b_b - \sum_{s=1}^{N_s} \left(L_b^{(s)^T} K_{ib}^{(s)^T} K_{ii}^{(s)^{-1}} b_i^{(s)} \right) \tag{4.1.7}$$

which can be rewritten

$$S x_b = C \tag{4.1.8}$$

where

$$S = \sum_{s=1}^{N_s} S^{(s)} \; S^{(s)} = L_b^{(s)^T} K_{bb}^{(s)} L_b^{(s)} - L_b^{(s)^T} K_{ib}^{(s)^T} K_{ii}^{(s)^{-1}} K_{ib}^{(s)} L_b^{(s)}$$

and

$$C = b_b - \sum_{s=1}^{N_s} L_b^{(s)^T} K_{ib}^{(s)^T} K_{ii}^{(s)^{-1}} b_i^{(s)}$$

Each substructure matrix $S^{(s)}$ is the local Schur complement, which correspond to a local condensation operator.
The assembling of the Schur complement

$$S = \sum_{s=1}^{N_s} S^{(s)}$$

distroys the original sparsity of the stiffness matrix K. However, we don't need to assemble explicitely S for computing S times a vector. The linear system (4.1.8) is solved iteratively, using a Preconditionned gradient algorithm [20].

remark on the generalized condensation

for the generalized eigenvalue problem,

$$Kx = \lambda Mx, \tag{4.1.9}$$

where $K \in \mathbb{R}^{(n,n)}$ and the mass matrix $M \in \mathbb{R}^{(n,n)}$ are real symmetric and positive definite (x is the modal displacement and λ the square of the angular frequency). once again if we partition the unknowns of the global structure into internal degrees of freedom and the interface boundary degrees of freedom:

$$\begin{bmatrix} K_{ii} & K_{ib} \\ K_{ib}^T & K_{bb} \end{bmatrix} \begin{bmatrix} x_i \\ x_b \end{bmatrix} = \lambda \begin{bmatrix} M_{ii} & M_{ib} \\ M_{ib}^T & M_{bb} \end{bmatrix} \begin{bmatrix} x_i \\ x_b \end{bmatrix} \tag{4.1.10}$$

from the first row of (4.1.10), one obtains the exact transformation relation

$$x_i = -\left(K_{ii} - \lambda M_{ii}\right)^{-1} \left(K_{ib} - \lambda M_{ib}\right) x_b \tag{4.1.11}$$

between the internal unknowns and the boundary unknowns. Substituting in the second row of eqn (4.1.10) yields the exactly condensed eigenvalue problem

$$\left\{-K_{bb} + \lambda M_{bb} + \left(K_{ib}^T - \lambda M_{ib}^T\right) \left(K_{ii} - \lambda M_{ii}\right)^{-1} \left(K_{ib} - \lambda M_{ib}\right)\right\} x_b = 0 \tag{4.1.12}$$

We have the following result for Schur complement matrix

Theorem 4.1.2 *If h denotes the mesh size of the finite element problem, the condition number in the L^2 norm of the Schur complement matrix S governing the interface problem ($Sx_b = C$)grows asymptotycally as $O(1/h)$.*

Note that the conditionning number of the global stiffness matrix K varies asymptotically as $O(1/h^2)$.

Remark 4.1.3 *Preconditionning of the Schur complement is discussed in [11]*
Theoretical study of the preconditionning of the Schur complement matrix and the link with the preconditionning of the global matrix can be found in [17]

4.1.2 Short description of the FETI method

The FETI method [11] is based on introducing the Lagrange multiplier of the continuity condition on interfaces between subdomains. In the case of linear elasticity equations, the Lagrange multiplier is equal to the field of interaction forces between subdomains.
In each subdomain, Ω_i, the local displacement field is solution of the linear elasticity equations with external loadings and boundary conditions inherited from the complete problem, and imposed forces (Neumann boundary conditions) on the interfaces with other subdomains.
Note λ the discrete flux on the interface $\Gamma_{ij} = \partial\Omega_i \bigcap \partial\Omega_j$

$$\frac{\partial u_i}{\partial n_i} = \lambda = -\frac{\partial u_j}{\partial n_j} \tag{4.1.13}$$

The local Neumann problems can be written

$$\left\{ \begin{array}{rcl} -\Delta u_i & = & F_i \\ \frac{\partial u_i}{\partial n_i} & = & \lambda \end{array} \right. \tag{4.1.14}$$

With a finite element discretization, this leads to the following set of equations:

$$K_i x_i = y_i + B_i^t \lambda \tag{4.1.15}$$

where K_i is the stiffness matrix, x_i the displacement field,B_i a signed boolean matrix associated with the discrete trace operator, and λ the Lagrange multiplier.
The discrete continuity condition along the interfaces is written as follows

$$\sum_{i=1}^{N_s} B_i x_i = 0 \tag{4.1.16}$$

where the signed discrete trace matrices B_i are such that if subdomains Ω_i and Ω_j are connected by the interface Γ_{ij}, then restriction of equation (4.1.16) on Γ_{ij} is: $u_i - u_j = 0$.

Remark 4.1.4 *$L_b^{(s)}$ (seen in previous paragraph): is the localisation operator that relates the local boundary d.o.f to the global interface d.o.f*
$B^{(s)}$: is a signed boolean matrix which localizes a substructure quantity to the substructure interface $\Gamma^{(s)}$ see [11][chapter 5].

In most subdomains, local problems (4.1.15) are ill posed, because only Neumann boundary conditions are imposed.
So, we note that the solution λ of (4.1.15) with continuity condition (4.1.16) is unique only up to an additive vector of $ker(B^t)$.
The space of Lagrange multipliers is therefore choosen as $range(B)$. We will also use a matrix built from the null space elements of K, ie columns of this matrix form a basis of the kernel of K (Rigid body motions in the case of elasticity). We note this matrix R
Then equation (4.1.15)is equivalent to:

$$\left\{ \begin{array}{ccc} x_i & = & K_i^+[y_i + B_i^t\lambda_i] + R_i\alpha_i \\ R_i^t[y_i + B_i^t\lambda_i] & = & 0 \end{array} \right. \tag{4.1.17}$$

The first equation means that the solution of the problem is defined as the sum of a particular solution computed using the pseudo inverse of K_i plus an element of the kernel. The second equation means that the right-hand side of equation (4.1.15) must be in the image space of K_i (the solution of (4.1.15) exists if and only if $b_i + B_i^t\lambda \in range(K_i)$); this constraint will lead us to the introduction of a projection P.
substituting x_i given by equation (4.1.17) in the continuity condition (4.1.16) gives:

$$\sum_i B_i K_i^+ B_i^t + \sum_i B_i R_i \alpha_i = -\sum_i B_i K_i^+ b_i \tag{4.1.18}$$

Interface equation (4.1.18) With the constraint on λ set by the second equation of (4.1.17), leads to the global condensed interface problem satisfied by λ and α the vector of coefficients of the kernel components:

$$\left[\begin{array}{cc} F & -G \\ -G^t & 0 \end{array} \right] \left[\begin{array}{c} \lambda \\ \alpha \end{array} \right] = \left[\begin{array}{c} d \\ c \end{array} \right] \tag{4.1.19}$$

With:

- $F = \sum B_i K_i^+ B_i^t$, dual Schur Complement matrix (or Interface Flexibility)

- $G\alpha = \sum B_i R_i \alpha_i$, jump of rigid body motions defined by α_i in Ω_i
- $(G^t\lambda)_i = R_i^t B_i^t \lambda$,
- $d = -\sum B_i k_i^+ b_i$
- $c = -R_i^t b_i$

The condensed interface problem (4.1.19) is an hybrid system. It's solution λ satisfies the following orthogonality condition:

$$\mu^t F \lambda = \mu^t d, \quad \forall \mu / G^t \mu = 0 \tag{4.1.20}$$

Given λ_0 satisfying the second equation of (4.1.17), then $G^t(\lambda - \lambda_0) = 0$. So, if P is is any projector in the kernel of G^t, then equation (4.1.20) entails that λ is the solution of the following projected problem:

$$P^t F P(\lambda - \lambda_0) = P^t(d - F\lambda_0) \tag{4.1.21}$$

The FETI method is based on the solution of the pojected condensed interface problem (4.1.20) via a conjugate gradient algorithm, using the orthogonal projector in the kernel of G^t

Solution of the Condensed Interface Problem

The number of constraints of the hybrid condensed interface problem (4.1.19) is the total number of rigid body modes.As this number is low, the projector associated with this constraint can be explicitly computed:

$$P = I - QG(G^tQG)^{-1}G^t \tag{4.1.22}$$

Q is a symmetric, positive definite matrix which induces an inner product on the space of Lagrange multipliers.There are different good choices for Q. In most cases it is sufficient to use $Q = I$. However, for problems with jumps in the coefficients, one has to make a more elaborate choice.
For simplicity in this report, we will assume that $Q \equiv I$:

$$P = I - G(G^tG)^{-1}G^t \tag{4.1.23}$$

The computation of the product by projector P requires products by G and G^t and the solution of systems with form:

$$(G^tG)\alpha = G^t g \tag{4.1.24}$$

The product by G^t can be performed independently in each subdomain, the product by G requires exchanging data through interfaces between neighbouring subdomains. Both products can be easily performed in parallel in a message passing programming environment.
Parallelizing the solution of problem (4.1.24) is more challenging because of it's global implicit nature. To avoid the construction of matrix G^tG), this problem can be solved by the conjugate gradient algorithm. Then only products by G and G^t that can be performed in parallel are required.
Applying the projected conjugate gradient algorithm to the condensed interface problem (4.1.19), requires the following two main steps.

1. Given an approximation value λ^p, compute the particular solution of the local Neumann problem in each subdomain:

$$x_i^{p+} = K_i^+[b_i + B_i^t\lambda^p] \tag{4.1.25}$$

and compute the jump of the local displacement fields along interfaces between subdomains that is the gradient of the condensed interface problem:

$$g^p = \sum_i B_i x_i^{p+} \tag{4.1.26}$$

2. Compute the projected gradient, Pg^p given by formula:

$$Pg^p = g^p + G\alpha^p with (G^tG)\alpha^p = -G^tg^p \tag{4.1.27}$$

The projection step consists of,in fact, in computing the rigid body motion coefficients α_i that minimize the jump of the complete displacement fields given by:

$$x_i^p = x_i^{p+} + R_i\alpha_i \tag{4.1.28}$$

This minimization is performed in the sense that the jump of the complete displacement fields x_i^p, which is in fact the projected gradient, is orthogonal to the traces of all the local rigid body modes:

$$Pg^p = \sum_i B_i x_i^{p+} + \sum_i B_i R_i \alpha_i = \sum_i B_i x_i^p \tag{4.1.29}$$

$$(G\beta)^t Pg^p = 0 \quad \forall\beta \Leftrightarrow (B_iR_i)^t Pg^p = 0 \quad \forall i \tag{4.1.30}$$

This projection phase consists of solving a global coarse problem associated with the rigid body coefficients.

Remark 4.1.5 *The most basic FETI preconditionner, is of the form:*

$$M^{-1} = BKB^t \tag{4.1.31}$$

Further details and a developpement including mathematical analysis of the FETI method can be found in [11][chapter 5].
We have the following important conditionning result

Theorem 4.1.6 *If the elements and the substructures have regular shape and size (the diameters are respectively h and H), the FETI method applied to the poisson or plane strain/stress elasticity problem has a condition number that is bounded by :*

$$\kappa \leq \frac{H}{h} \tag{4.1.32}$$

A performance comparison between the dual FETI method and the primal methods can be found in [11][chapter 6].

4.2 Primal and dual assembly of substructures

The discretized expression of a linearized structural dynamics problem can be written in matrix form

$$\omega^2 M^{(s)} u^{(s)} + K^{(s)} u^{(s)} = f^{(s)} + g^{(s)}, s = 1, \ldots, N_s \tag{4.2.1}$$

for each substructure.
In the remaining of this section $\omega^2 = \lambda$ will be included in mass matrix M, we also keep the external force f which will be set to zero if needed (for the modal analysis). $g^{(s)}$, the internal forces on the interfaces between substructures that ensure compatibility ($g^{(s)}$ cancel out when assembled on the interface)

4.2.1 Primal assembly of substructures

The assembly of (4.2.1) is done in similar way like the stiffness matrix of the simple Poisson problem and leads to

$$Mu + Ku = f \tag{4.2.2}$$

where

$$\begin{bmatrix} M_{ii}^{(1)} & 0 & \dots & 0 & M_{ib}^{(1)} L_b^{(1)} \\ 0 & \ddots & & & \vdots \\ \vdots & & \ddots & & \vdots \\ 0 & & & M_{ii}^{(N_s)} & M_{ib}^{(N_s)} L_b^{(N_s)} \\ L_b^{(1)^T} M_{ib}^{(1)^T} & \dots & \dots & L_b^{(N_s)^T} M_{ib}^{(N_s)^T} & \sum_{s=1}^{N_s} L_b^{(s)^T} M_{bb}^{(s)} L_b^{(s)} \end{bmatrix}, \begin{bmatrix} u_i^{(1)} \\ \vdots \\ \vdots \\ u_i^{(N_s)} \\ u_b \end{bmatrix}$$

$$\begin{bmatrix} K_{ii}^{(1)} & 0 & \dots & 0 & K_{ib}^{(1)} L_b^{(1)} \\ 0 & \ddots & & & \vdots \\ \vdots & & \ddots & & \vdots \\ 0 & & & K_{ii}^{(N_s)} & K_{ib}^{(N_s)} L_b^{(N_s)} \\ L_b^{(1)^T} K_{ib}^{(1)^T} & \dots & \dots & L_b^{(N_s)^T} K_{ib}^{(N_s)^T} & \sum_{s=1}^{N_s} L_b^{(s)^T} K_{bb}^{(s)} L_b^{(s)} \end{bmatrix}, \begin{bmatrix} f_i^{(1)} \\ \vdots \\ \vdots \\ f_i^{(N_s)} \\ f_b \end{bmatrix}$$

4.2.2 Dual assembly of substructures

The dual assembly is similar to the one we have seen in FETI method

$$\begin{cases} \omega^2 M^{(s)} U^{(s)} + K^{(s)} U^{(s)} + \begin{bmatrix} b^{(s)^T} \lambda \\ 0 \end{bmatrix} = F^{(s)} \\ \sum_{i=1}^{N_s} b^{(s)} U^{(s)} = 0 \end{cases} \tag{4.2.3}$$

where B_i is the discrete trace operator which give the restriction on Ω of a boundary defined field on Ω_i. Comparing with (4.2.1), $b^{(s)^T}\lambda$ represent the interconnecting forces between substructures(flux in mechanics of structures).
We consider the block-diagonal notations

$$M = \begin{bmatrix} M^{(1)} & & 0 \\ & \ddots & \\ 0 & & M^{(N_s)} \end{bmatrix}, K = \begin{bmatrix} K^{(1)} & & 0 \\ & \ddots & \\ 0 & & K^{(N_s)} \end{bmatrix},$$

$$u = \begin{bmatrix} u^{(1)} \\ \vdots \\ u^{(N_s)} \end{bmatrix}, f = \begin{bmatrix} f^{(1)} \\ \vdots \\ f^{(N_s)} \end{bmatrix}$$

$$B = \left[B^{(1)} \dots B^{(N_s)}\right], B^{(s)^T} = \begin{bmatrix} b^{(s)^T} \\ 0 \end{bmatrix}$$

The sets of Eqs (4.2.3) can the be written in block form

$$\begin{bmatrix} M & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ \lambda \end{bmatrix} + \begin{bmatrix} K & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} u \\ \lambda \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} \tag{4.2.4}$$

The dual assembled system (4.2.4) is equivalent to the assembled system (4.2.2) since they express the same local equilibrium and enforce the same interface compatibility.

4.3 The Craig-Bampton Method

4.3.1 a brief review of the Component Mode Synthesis

Component mode synthesis (CMS) method is a substructure coupling method. That is a complex structure (e.g., systems of automotive components, or Shuttle Orbiter Vehicle with attached payloads) is divided into a number of substructures or components see(citer figure) and then assembling a reduced order model of entire structure. The term **Component Mode** is used to signify that the substructures also named components are represented (in the sense of Ritz's vectors). These assumed modes may be classified into several classes:

- **Normal modes:** Component normal modes are eigenvectors and may be classified according to the interface boundary conditions specified for the component:fixed-interface normal modes, free-interface normal modes, hybrid interface normal modes or loaded-interface normal modes. normal modes are obtained from the solution of the generalized eigenvalue problem

$$\left[K_{ii}^{(s)} - \lambda M_{ii}^{(s)}\right] \{\Phi_i\} = 0, s = 1, 2, \dots, N_s \tag{4.3.1}$$

- **Constraint modes:** A constraint mode is defined as the static deformation of a structure when a unit displacement is applied to one coordinate of specified set of constrained coordinates while the remaining d.o.f of the structure are force free.
  Rigid body modes are actually a special case of constraint modes.

- **Attachement modes:** An attachement mode is defined as the component displacement vector due to a single unit force applied at one of the coordinates of a given attached set.

- ...

4.3.2 Craig-Bampton

Craig-Bampton method [6] is based on constraint-modes and fixed-interface modes, that is, the displacement transformation for this method employs a combination of fixed-interface normal modes and interface constraint modes. The substructured system (4.2.2) of the dynamic equations indicates that

$$M_{ii}^{(s)}u_i^{(s)} + K_{ii}^{(s)}u_i^{(s)} = f_i^{(s)} - K_{ib}^{(s)}u_b^{(s)} - M_{ib}^{(s)}u_b^{(s)} \tag{4.3.2}$$

which indicates that every substructure can be considered as being exited by internal forces and imposed displacements on its interface boundary.

first of all, we show that the dynamic behavior of a given subsystem requires two types of information:

- the static boundary modes
- the subsystem eigenmodes in clamped boundary configuration

$$\begin{bmatrix} K_{11} & K_{12} \\ K_{12}^T & K_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} - \omega^2 \begin{bmatrix} M_{11} & M_{12} \\ M_{12}^T & M_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ f_2 \end{bmatrix} \tag{4.3.3}$$

f_2 represents the external loads and/or the boundary reactions see (fig 4.1). The first equation in (4.3.3) gives

$$u_1 = -(K_{11} - \omega^2 M_{11})^{-1}(K_{12} - \omega^2 M_{12})u_2 \tag{4.3.4}$$

yielding the relationship

$$\left\{(K_{11} - \omega^2 M_{11}) - (K_{12}^T - \omega^2 M_{12}^T)(K_{11}^T - \omega^2 M_{11}^T)^{-1}(K_{12} - \omega^2 M_{12})\right\} u_2 = f_2 \tag{4.3.5}$$

one notes that, the coefficient of u_2 in (4.3.5) admits as poles the zeros of

$$det(K_{11} - \omega^2 M_{11}) = 0 \tag{4.3.6}$$

which correspond to the eigenfrequencies of the subsystem clamped on its boundary degrees of freedom u_2 fixed. Therefore matrix

$$(K_{11} - \omega^2 M_{11})^{-1} \tag{4.3.7}$$

(which is sometimes referred as admittance matrix, the inverse of impedance matrix)
is expanded in terms of eigensolutions of the subsystem clamped on it's interface. Let us introduce $\tilde{\omega}_r$ and $\tilde{\theta}_r$ the eigenfrequencies and eigenmodes of the subsystem

$$det(K_{11} - \omega^2 M_{11})\tilde{\theta} = 0 \tag{4.3.8}$$

They are numbered in the following order

$$\begin{cases} 0 < \tilde{\omega}_1^2 \leq \ldots \leq \tilde{\omega}_n^2 \\ \tilde{\theta}_{(1)}, \ldots, \tilde{\theta}_{(n)} \end{cases} \tag{4.3.9}$$

with the orthonormality properties

$$\tilde{\theta}_i M_{11} \tilde{\theta}_j = \delta_{ij} \tag{4.3.10}$$

one obtains spectral expansion by solving the forced harmonic response for the subsystem clamped on its boundary:

$$(K_{11} - \omega^2 M_{11})\theta = \eta \tag{4.3.11}$$

expanding the solution vector in terms of eigenmodes:

$$\theta = \sum_{i=1}^{n_1} \alpha_i \tilde{\theta}_{(i)} \tag{4.3.12}$$

we get

$$\sum_{i=1}^{n_1} \alpha_i (K_{11} - \omega^2 M_{11})\tilde{\theta}_{(i)} = \eta \tag{4.3.13}$$

making use of orthogonality

$$\alpha_i = \frac{\tilde{\theta}_{(i)}^T}{\tilde{\omega}_i^2 - \omega^2} \tag{4.3.14}$$

we obtain

$$\theta = \frac{\tilde{\theta}_{(i)} \tilde{\theta}_{(i)}^T}{\tilde{\omega}_i^2 - \omega^2} \eta \tag{4.3.15}$$

and

$$(K_{11} - \omega^2 M_{11})^{-1} = \frac{\tilde{\theta}_{(i)} \tilde{\theta}_{(i)}^T}{\tilde{\omega}_i^2 - \omega^2} \tag{4.3.16}$$

in particular, one deduces

$$K_{11}^{-1} = \frac{\tilde{\theta}_{(i)} \tilde{\theta}_{(i)}^T}{\tilde{\omega}_i^2} \tag{4.3.17}$$

successive transformations lead to

$$(K_{11} - \omega^2 M_{11})^{-1} = K_{11}^{-1} + \omega^2 \sum_{i=1}^{n_1} \frac{\tilde{\theta}_{(i)} \tilde{\theta}_{(i)}^T}{\tilde{\omega}_i^4 (\tilde{\omega}_i^2 - \omega^2)} \tag{4.3.18}$$

or equivalently (by making use of the modal expansion of M_{11}

$$(K_{11} - \omega^2 M_{11})^{-1} = K_{11}^{-1} + \omega^2 K_{11}^{-1} M_{11} K_{11}^{-1} + \omega^4 \sum_{i=1}^{n_1} \frac{\tilde{\theta}_{(i)} \tilde{\theta}_{(i)}^T}{\tilde{\omega}_i^4 (\tilde{\omega}_i^2 - \omega^2)} \tag{4.3.19}$$

and then, we compute the reduced matrix

$$Z_{22}^* = K_{22} - \omega^2 M_{22} - (K_{21} - \omega^2 M_{21})(K_{11} - \omega^2 M_{11})^{-1}(K_{12} - \omega^2 M_{12}) \tag{4.3.20}$$

and we will keep only the terms of orders 1 and 2 in ω^2.
Finally, we get the following expression:

$$\begin{aligned} Z_{22}^* \;=\; & K_{22} - K_{21} K_{11}^{-1} K_{12} \\ & - \omega^2 \left[M_{22} + M_{21} K_{11}^{-1} K_{12} - K_{21} K_{11}^{-1} M_{12} + K_{21} K_{11}^{-1} M_{11} K_{11}^{-1} K_{12} \right] \\ & + \omega^4 \sum_{i=1}^{n_1} \frac{1}{(\tilde{\omega}_i^2 - \omega^2)} \left\{ - \frac{K_{21} \tilde{\theta}_{(i)} \tilde{\theta}_{(i)}^T K_{12}}{\tilde{\omega}_i^4} + \frac{M_{21} \tilde{\theta}_{(i)} \tilde{\theta}_{(i)}^T K_{12}}{\tilde{\omega}_i^2} + \frac{K_{21} \tilde{\theta}_{(i)} \tilde{\theta}_{(i)}^T M_{12}}{\tilde{\omega}_i^2} \right\} \\ & + \omega^4 \sum_{i=1}^{n_1} \frac{1}{(\tilde{\omega}_i^2 - \omega^2)} \left\{ - M_{21} \tilde{\theta}_{(i)} \tilde{\theta}_{(i)}^T M_{12} \right\} \end{aligned} \tag{4.3.21}$$

In (4.3.21) we observe that:

- the first term
$$K_{22}^* = K_{22} - K_{21} K_{11}^{-1} K_{12} \tag{4.3.22}$$
represents the stiffness of the subsystem statically condensed on it's boundary. It is obtained by setting $\omega^2 = 0$ in (4.3.4)

- the second term
$$M_{22}^* = M_{22} + M_{21} K_{11}^{-1} K_{12} - K_{21} K_{11}^{-1} M_{12} + K_{21} K_{11}^{-1} M_{11} K_{11}^{-1} K_{12} \tag{4.3.23}$$
represent the mass of the subsystem statically condensed on it's boundary.

- the last two terms represent the contribution of the subsystem eigenmodes. To each eigenfrequency there corresponds a pole of the reduced matrix, and the associated term is spectrally expanded in terms of the subsystem eigenmodes in clamped boundary configuration.

Then, the expression (4.3.20) demonstrates that the appropriate representation of the dynamic behavior of a given subsystem requires two types of information:

- the static boundary modes represented by the static condensation
- the subsystem eigenmodes in clamped boundary configuration

Remark 4.3.1 *In literature, the frequency dependant matrix*

$$Z(\omega^2) = K - \omega^2 M \tag{4.3.24}$$

is sometimes referred as ***the mechanical impedance matrix*** *while its inverse is referred as* ***the admittance matrix***

This provide the principle of the *Craig and Bampton method.*

Then, the internal d.o.f can be represented as

$$u_i^{(s)} = u_{i,stat}^{(s)} + \Phi^{(s)}\eta^{(s)} \tag{4.3.25}$$

where $u_{i,stat}^{(s)}$ represents the quasi-static solution

$$u_{i,stat}^{(s)} = {K_{ii}^{(s)}}^{-1}\left(-K_{ib}^{(s)}u_b^{(s)} + f_i^{(s)}\right) \tag{4.3.26}$$

and where $\Phi^{(s)}$ is a matrix of eigenmodes associated to $M_{ii}^{(s)}$ and $K_{ii}^{(s)}$ (in real life problems, $\Phi^{(s)}$ contains only the first $n_\phi^{(s)}$ free vibration modes of the substructure clamped on it's interface, (see fig 4.1). These modes are solutions of

$$\left(K_{ii}^{(s)} - \lambda M_{ii}^{(s)}\right)\phi_k^{(s)} = 0 \qquad k = 1, \ldots, n_\phi^{(s)} \tag{4.3.27}$$

The equation (4.3.25) defines an aproximation basis such that

$$u = \begin{bmatrix} u_i^{(1)} \\ \ldots \\ u_i^{(N_s)} \\ u_b \end{bmatrix} \simeq \underbrace{\begin{bmatrix} \Phi^{(1)} & 0 & \ldots & \Psi^{(1)}L_b^{(s)} \\ 0 & \ldots & & \vdots \\ & & \Phi^{(N_s)} & \\ 0 & \ldots & 0 & I \end{bmatrix}}_{T_{CB}} \begin{bmatrix} \eta^{(1)} \\ \ldots \\ \eta^{(N_s)} \\ u_b \end{bmatrix} + \begin{bmatrix} {K_{ii}^{(1)}}^{-1} f_i^{(1)} \\ \vdots \\ {K_{ii}^{(N_s)}}^{-1} f_i^{(N_s)} \\ 0 \end{bmatrix} \tag{4.3.28}$$

where T_{CB} is the Craig-Bampton reduction matrix and where $\Psi^{(s)} = {K_{ii}^{(s)}}^{-1}K_{ib}$ are static modes representing the static deformation in a substructure when

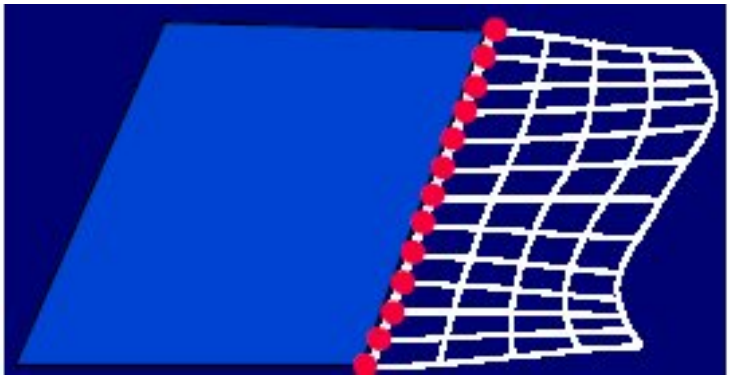


Figure 4.1: Vibration modes (fixed interface)

a unit displacement is applied on a boundary unit d.o.f (see fig 4.2) and finaly $\eta^{(s)}$ are referred as **the generalized coordinates** in literature.
Using the reduction basis defined by (4.3.28), the initial dynamic problem (4.2.1) can be transformed in the classical Craig-Bampton reduced form.

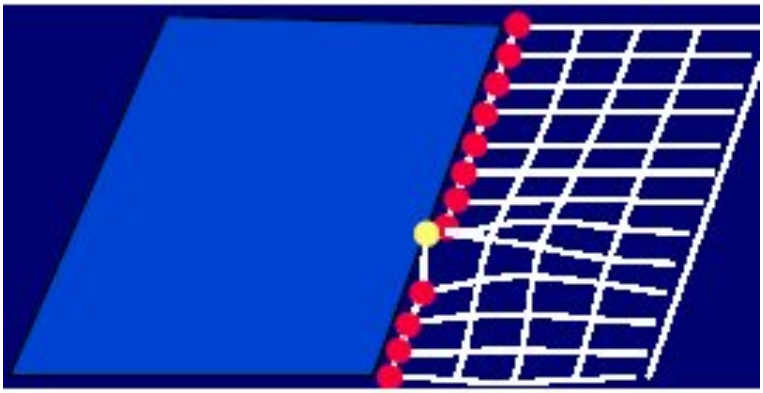


Figure 4.2: Static modes

When normalized with respect to the mass matrix M_{ii}, the modal matrix satisfy in each substructure

$$\Phi^T M_{ii} \Phi = I, \quad \Phi^T K_{ii} \Phi = \Lambda = diag(\omega_j^2) \tag{4.3.29}$$

Applying the reduction procedure (4.3.28), one obtains the standard Craig-

Bampton reduced matrices

$$\overline{K}_{CB} = T_{CB}^T K T_{CB} = \begin{bmatrix} \Lambda^{(1)} & & & 0 \\ & \ddots & & \\ & & \Lambda^{(N_s)} & \\ 0 & & & Sbb \end{bmatrix} \tag{4.3.30}$$

and

$$\overline{M}_{CB} = T_{CB}^T M T_{CB} = \begin{bmatrix} I & & & \widetilde{M}_{ib}^{(1)} \\ & \ddots & & \vdots \\ & & I & \widetilde{M}_{ib}^{(N_s)} \\ \widetilde{M}_{ib}^{(1)^T} & \dots & \widetilde{M}_{ib}^{(N_s)^T} & \widetilde{M}_{bb} \end{bmatrix} \tag{4.3.31}$$

where

$$S_{bb} = \sum_{s=1}^{N_s} (L_b^{(s)^T} \Psi^{(s)^T} K_{bb}^{(s)} \Psi^{(s)} L_b^{(s)}) \tag{4.3.32}$$

$$\widetilde{M}_{ib}^{(1)} = \Phi^{(1)^T} (\sum_{s=1}^{N_s} M_{1s} \Psi^{(s)} L_b^{(s)} + M_{1b}) \tag{4.3.33}$$

$$\widetilde{M}_{ib}^{(N_s)} = \Phi^{(N_s)^T} (\sum_{s=1}^{N_s} M_{N_s s} \Psi^{(s)} L_b^{(s)} + M_{N_s b}) \tag{4.3.34}$$

$$\widetilde{M}_{bb} = \sum_{k=1}^{N_s} \{ L_b^{(k)^T} \Psi^{(k)^T} (\sum_{s=1}^{N_s} M_{ks} \Psi^{(s)} L_b^{(s)} + M_{kb}) + M_{kb}^T \Psi^{(k)} L_b^{(k)} \} + M_{bb} \tag{4.3.35}$$

4.3.3 The dual Craig-Bampton method

Free interface modes as reduction basis

If we consider the dual assembled problem (4.2.3) or (4.2.4), every substructure can be seen as being excited through interface connection forces (Lagrange's multipliers). The local dynamical behavior can therefore be expressed in terms of normal modes (eigenmodes) of the entire local matrices, hence in terms of free vibration modes of the substructures with free interfaces, and in terms of local static solutions

$$u^{(s)} = u_{stat}^{(s)} + \sum_{r=1}^{n^{(s)}-m^{(s)}} \theta_r^{(s)} \eta_r^{(s)} \tag{4.3.36}$$

where

$$u_{stat}^{(s)} = -K^{(s)^\dagger} B^{(s)^T} \lambda + \sum_{i=1}^{m^{(s)}} R_i^{(s)} \alpha_i^{(s)} \tag{4.3.37}$$

with the following definitions:

- $K^{(s)^\dagger}$ is the inverse of $K^{(s)}$ when there are enough boundary conditions to prevent the substructure from floating when it's interface with neighboring domains is free. If a substructure is floating $K^{(s)^\dagger}$ is the generalized inverse of $K^{(s)}$.
- $R^{(s)}$ is the matrix having as column the $m^{(s)}$ corresponding rigid modes when substructure is floating.
- $\alpha_i^{(s)}$ are amplitudes of the local rigid body modes.
- $\eta_r^{(s)}$ are the amplitudes of the local eigenmodes.

An approximation is then constructed by considering only the first $n_\theta^{(s)}$ free modes $\theta^{(s)}$ in the expansion. If we denote $\Theta^{(s)}$ the matrix containing these $n_\theta^{(s)}$ modes,

$$u^{(s)} \simeq -K^{(s)^\dagger} B^{(s)^T} \lambda + R^{(s)} \alpha^{(s)} + \Theta^{(s)} \eta^{(s)} \tag{4.3.38}$$

From this approximation, we can observe that, part of the solution in the subspace of $\Theta^{(s)}$ is also included in $K^{(s)^\dagger} B^{(s)^T} \lambda$ since the Singular Value decomposition of the generalized inverse has the classical spectral expansion

$$K^{(s)^\dagger} = \sum_{r=1}^{n^{(s)}-m^{(s)}} \frac{\theta_r^{(s)} \theta_r^{(s)^T}}{\omega_r^{(s)^2}} \tag{4.3.39}$$

we use here $\omega_r^{(s)^2}$ to avoid confusion between eigenvalues and the Lagrange's multipliers.Further details concerning Moore-Penrose generalized inverse for matrices can be found in [9]
Hence, the approximation (4.3.38) can be equivalently written as

$$u^{(s)} = -G_{res}^{(s)} B^{(s)^T} \lambda + R^{(s)} \alpha^{(s)} + \Theta^{(s)} \eta^{(s)} \tag{4.3.40}$$

where

$$G_{res}^{(s)} = \sum_{r=n_\theta^{(s)}+1}^{n^{(s)}-m^{(s)}} \frac{\theta_r^{(s)} \theta_r^{(s)^T}}{\omega_r^{(s)^2}} = K^{(s)^\dagger} - \sum_{r=1}^{n_\theta^{(s)}} \frac{\theta_r^{(s)} \theta_r^{(s)^T}}{\omega_r^{(s)^2}} \tag{4.3.41}$$

$G_{res}^{(s)}$ is **the residual local flexibility matrix** and has the following properties

- $G_{res}^{(s)} = G_{res}^{(s)^T}$
- $G_{res}^{(s)^T} K^{(s)} G_{res}^{(s)} = G_{res}^{(s)}$
- $\Theta^{(s)^T} K^{(s)} G_{res}^{(s)} = 0$
- $\Theta^{(s)^T} M^{(s)} G_{res}^{(s)} = 0$
- $R^{(s)^T} M^{(s)} G_{res}^{(s)} = 0$

The equation (4.3.36) defines an aproximation basis such that

$$\begin{bmatrix} u \\ \lambda \end{bmatrix} \simeq \underbrace{\begin{bmatrix} R^{(1)} & \Theta^{(1)} & 0 & & 0 & -G_{res}^{(1)} B^{(1)^T} \\ 0 & \ddots & \ddots & & & \vdots \\ & & & R^{(N_s)} & \Theta^{(N_s)} & -G_{res}^{(N_s)} B^{(N_s)^T} \\ 0 & & \dots & & 0 & I \end{bmatrix}}_{T_{dual}} \begin{bmatrix} \alpha^{(1)} \\ \eta^{(1)} \\ \dots \\ \alpha^{(N_s)} \\ \eta^{(N_s)} \\ \lambda \end{bmatrix} \tag{4.3.42}$$

Using approximation (4.3.42) in the dual approximation (4.2.4) and assuming that the rigid and elastic modes are orthonormalized with respect to $M^{(s)}$ ($\Phi^T M \Phi = I$), one obtains the hybrid system

$$\tilde{M} \begin{bmatrix} \vdots \\ \alpha^{(s)} \\ \eta^{(s)} \\ \vdots \\ \lambda \end{bmatrix} + \tilde{K} \begin{bmatrix} \vdots \\ \alpha^{(s)} \\ \eta^{(s)} \\ \vdots \\ \lambda \end{bmatrix} = T_{dual}^T f \tag{4.3.43}$$

with the reduced hybrid matrices

$$\tilde{M} = T_{dual}^T \begin{bmatrix} M & 0 \\ 0 & 0 \end{bmatrix} T_{dual} = \begin{bmatrix} I & 0 \\ 0 & M_{res} \end{bmatrix} \tag{4.3.44}$$

and

$$\tilde{K} = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & \Lambda^{(1)} \end{bmatrix} & & 0 & & \begin{bmatrix} R^{(1)^T} B^{(1)^T} \\ \Theta^{(1)^T} B^{(1)^T} \end{bmatrix} \\ & \ddots & & & \vdots \\ 0 & & & \begin{bmatrix} 0 & 0 \\ 0 & \Lambda^{(N_s)} \end{bmatrix} & \begin{bmatrix} R^{(N_s)^T} B^{(N_s)^T} \\ \Theta^{(N_s)^T} B^{(N_s)^T} \end{bmatrix} \\ \begin{bmatrix} B^{(1)} R^{(1)} & B^{(1)} \Theta^{(1)} \end{bmatrix} & \dots & \begin{bmatrix} B^{(N_s)} R^{(N_s)} & B^{(N_s)} \Theta^{(N_s)} \end{bmatrix} & & -F_{res} \end{bmatrix}, \tag{4.3.45}$$

where

$$F_{res} = \sum_{s=1}^{N_s} B^{(s)} G_{res}^{(s)} B^{(s)^T} \quad \text{and}$$

$$M_{res} = \sum_{s=1}^{N_s} B^{(s)} G_{res}^{(s)} M^{(s)} G_{res}^{(s)} B^{(s)^T}$$

Remark 4.3.2 *According to [18], the cost for building the reduced matrices (4.3.44) and (4.3.45) is comparable to the cost of building the classical Craig-Bampton matrices.*

Chapter 5

Implementation and numerical analysis

5.1 The fortran implementation

The design goal for this implementation was to compute eigenvalues using Craig-Bampton method and to assess how far we can go in the spectrum. The aim is to compute more eigenvalues than what is usually done in structural dynamics.
This will be achieved at some points at the cost of computation time, communication time and memory consumption.
As parallelization architecture, the MPI (Message Passing Interface) [20] is employed. The MPI is a portable interface with several implementations which are available for a wide range of architectures. Each processor in a distributed memory parallel computer has local memory and local adress space. The Message Passing model assumes a set of processes that have only local memory but are able to communicate with other processes by sending and receiving messages. Data transfer from the local memory of one processor to the local memory of another one requires operations to be performed by both processors. Thus it matches the hardware of most of parallel super computers as well as workstation networks.
MPI is a large specification consisting of over 100 functions and offers many basic and advanced features including point to point communications, collective communications..... MPI's prime goals are to provide standardization, source code compatibility, high performance and to allow efficient implementation across a range of architectures and offering real portability between HPC (high Performance Computing) platforms.
Many generic versions of MPI are available free of charge (MPICH, LAM,...)

and some useful papers can be found at http://www.idriss.fr
If needed, MPI allows intermixing different programming langages such as C/C++ and Fortran.
MPI is used in the implementation for the interface data transfer between the subdomains.
In fact, the physical domain is divided into N_s non-overlapping subdomains $\Omega_1, \Omega_2, \ldots, \Omega_{N_s}$ and the problem on the global domain is replaced by solving iteratively a condensed problem on the subdomains.
Each subdomain is allocated to a processor which knows only the data corresponding to it's subdomain and information about neighboring subdomains.

For this report, a FETI-solver and a Lanczos routine was provided by my supervisor. We have also used ARPACK to compute local eigenmodes
The FETI code provided, includes MPI interface and several routines which will help us to implement what we have seen in chapter 4.
Among others, we can cite the following useful steps:

- calculation of the pseudo-inverse on the floating subdomains ($K^{(s)^\dagger}$)
- calculation of the rigid body modes on the floating subdomains ($R^{(s)}$)
- handling the interior and interface nodes for each subdomain to enable transfer from a subdomain variable to the global variable and back ($B^{(s)}$)
- Matrix vector products with G and G^T
- calculation of G^TG and $(G^TG)^{-1}$ for use in the projection operator
- Conjugate projected gradient with re-orthogonalization and preconditionning
- Calculation of the initial guess for the Lagrange multipliers (λ_0)
- calculation of the right hand side for the Conjugate projected gradient iterations
- calculation of the solution on each subdomain, given the values of α and λ

Below, a short description of Lanczos Procedure

5.2 Lanczos Algorithm

The Lanczos Algorithm for extracting the smallest eigenpairs of a system is an iterative algorithm based on inverse power methods. It is an algorithm for computing an orthogonal basis of a Krylov subspace.
Krylov subspace $\mathcal{K}_r$ ([20]) is a subspace of the form

$$\mathcal{K}_r = span\{v_0, K^{-1}Mv_0, \ldots, (K^{-1}M)^{r-1}v_0\} \tag{5.2.1}$$

The main iteration of the algorithm can be described by the following recursion
At the p-th iteration:

$$w = e(p)v_{p+1} = K^{-1}Mv_p - e(p-1)v_{p-1} - d(p)v_p \tag{5.2.2}$$

where $d(p)$ and $e(p)$ are selected in such a way that, v_{p+1} is M-orthogonal to v_p and v_{p-1}:

$$d(p) = v_p^T MK^{-1}Mv_p \qquad and \qquad e(p-1) = v_p^T MK^{-1}Mv_{p-1} \tag{5.2.3}$$

Then, (5.2.2) can be expressed in matrix form as follows:

$$K^{-1}MV_p = V_pT_p + S, \tag{5.2.4}$$

with

$$V_p = [v_0 \cdots v_p] \quad and \quad S = [0 \cdots e(p)v_{p+1}], \tag{5.2.5}$$

Where T_p is a tridiagonal matrix, which is the orthogonal projection of K onto the Krylov subspace (actually, T_p is the Hessenberg matrix H_p obtained from Arnoldi's methods when K is symmetric (see [20] for more details about Arnoldi's methods)):

$$T_p = \begin{pmatrix} d(1) & e(1) & 0 & 0 & \cdots & 0 & 0 \\ e(1) & d(2) & e(2) & 0 & \cdots & 0 & 0 \\ 0 & e(2) & d(3) & e(3) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & d(p-1) & e(p-1) \\ 0 & 0 & 0 & 0 & \cdots & e(p-1) & d(p) \end{pmatrix} \tag{5.2.6}$$

The application of the Rayleigh-Ritz procedure to the standard form of the initial problem, by a projection into the subspace generated by the Lanczos vectors $q = V_pz$, leads to the reduced eigenvalue problem:

$$T_pz = \frac{1}{\omega^2}z \tag{5.2.7}$$

The most CPU time is spent within the following operations:

- Solution of the large-scale linear system with K^{-1}
- M-orthogonalization of v_p
- computation of matrix-vector products with M

5.3 Computational procedure

The original problem is taken from ARC3D software. We've had two data files (for two subdomain case) containing the stiffness matrix stored as sparse matrix, the mass matrix which was a diagonal matrix, type of equations, information about neighboring subdomains, number and list of interface equations. Dirichlet conditions were already included in the problem, so there were no list of Dirichet nodes in the data files.

5.3.1 computation of local eigenmodes

As we have seen in chapter 4, in each subdomain, the following local problem is solved with Lanczos

$$K^{(s)}u^{(s)} = \omega^{(s)^2} M^{(s)}u^{(s)} \tag{5.3.1}$$

we kept the following error estimate ϵ to verify the results of the local eigenvalue problem to get the nomal modes of each substructure. We've found that $\epsilon \leq 10^{-16}$

$$\epsilon = \frac{\|Ku - \lambda Mu\|}{|\lambda|\|M\|\|u\|} \leq O(10^{-16}) \tag{5.3.2}$$

where $\lambda = \omega^2$.
We have also successfully verified the theoretical results presented in chapter 2 and chapter 4

$$\Phi^T M_{ii}\Phi = I, \quad \Phi^T K_{ii}\Phi = \Lambda = diag(\omega_j^2) \tag{5.3.3}$$

where $\Phi^{(s)}$ is the matrix containing local eigenvectors and $\Lambda^{(s)}$ is a diagonal matrix containing the local eigenvalues.
However, these results were not so obvious. Indeed, if a subdomain contains at least one rigid body mode (ie $A\%kern\_dim > 0$) then, at each iteration of Lanczos(or ARPACK) we must do a projection of the kernel before every matrix-vector product with the mass matrix M.
Practically, in Lanczos Algorithm, at the p-th iteration, before each dot product $d(p) = (K^{-1}Mv_p, Mv_p)$ and $e(p-1) = (Mw_p, w_p)^{\frac{1}{2}}$ (where w_p is defined in equation 5.2.2), we have to perform the following steps:

- test if there are rigid body modes in the subdomain (ie $A\%kern\_dim > 0$)
- then call in order the following subroutines: **dmtxv, dfull_fwbw and dmxvsub** available in FETI code
- perform the following dot products (M, v_p) **or** (M, w_p)
- call again the subroutines **dmtxv, dfull_fwbw and dmxvsub**

Inside Lanczos procedure, we have used the subroutine **local_matrix_solve** to solve $K^{-1}Mv$

5.3.2 computation of global eigenmodes

In order to verify the results we will obtain with Craig-Bampton method, we have first computed eigenmodes of the entire problem. Since the datas for the problem are separated in different files (a subdomain is allocated to a processor and only this processor knows the data corresponding to it's subdomain and information about the neighboring subdomains), the way to compute the eigenvalues and eigenvectors of the entire problem is to do the interface assembly to connect the subdomains.
Practically, all parallelizable tasks are done on each processor with internal degrees of freedom, and to work with the global vector (which include the contribution of all interface data), we have used the following subroutines available in FETI code: **globalsumr, interface_assemb and interface_average**.
The global vector has the following form for N_s subdomains

$$\begin{bmatrix} u_i^{(1)} \\ \vdots \\ \vdots \\ u_i^{(N_s)} \\ u_b = \sum_{s=1}^{N_s} u_b^{(s)} \end{bmatrix} \tag{5.3.4}$$

To verify the results of the global eigenvalue problem. We've found that $\epsilon \leq 10^{-13}$

$$\epsilon = \frac{\|Ku - \lambda Mu\|}{|\lambda|\|M\|\|u\|} \leq O(10^{-13}) \tag{5.3.5}$$

where $\lambda = \omega^2$.
Inside lanczos procedure, we have used the following FETI subroutine **SPD_Feti_1LM** to solve $K^{-1}Mv$

5.4 Implementation of the dual Craig-Bampton

We have focused on the dual Craig-Bampton method proposed by Daniel Rixen [18].
As we have seen in chapter 4, assembling the model reduction of each components leads to the following reduced eigenproblem:

$$\widetilde{K}_{CB} z = \omega^2 \widetilde{M}_{CB} z \tag{5.4.1}$$

where

$$\widetilde{K}_{CB} = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & \Lambda^{(1)} \end{bmatrix} & & 0 & \begin{bmatrix} R^{(1)^T} B^{(1)^T} \\ \Theta^{(1)^T} B^{(1)^T} \end{bmatrix} \\ & \ddots & & \vdots \\ 0 & & \begin{bmatrix} 0 & 0 \\ 0 & \Lambda^{(N_s)} \end{bmatrix} & \begin{bmatrix} R^{(N_s)^T} B^{(N_s)^T} \\ \Theta^{(N_s)^T} B^{(N_s)^T} \end{bmatrix} \\ \begin{bmatrix} B^{(1)} R^{(1)} & B^{(1)} \Theta^{(1)} \end{bmatrix} & \dots & \begin{bmatrix} B^{(N_s)} R^{(N_s)} & B^{(N_s)} \Theta^{(N_s)} \end{bmatrix} & -F_{res} \end{bmatrix},$$

with

$$F_{res} = \sum_{s=1}^{N_s} B^{(s)} G_{res}^{(s)} B^{(s)^T}$$

and

$$\widetilde{M}_{CB} = \begin{bmatrix} \ddots & & & 0 \\ & \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} & & \\ & & \ddots & \\ 0 & & & M_{res} \end{bmatrix},$$

with

$$M_{res} = \sum_{s=1}^{N_s} B^{(s)} G_{res}^{(s)} M^{(s)} G_{res}^{(s)} B^{(s)^T}$$

$G_{res}^{(s)}$, the residual local flexibility matrix is defined in equation (4.3.41). Then equations (5.4.1) can be rewritten during the iteration as

$$\begin{bmatrix} \ddots & & 0 & \vdots \\ & \begin{bmatrix} 0 & 0 \\ 0 & \Lambda^{(s)} \end{bmatrix} & & \begin{bmatrix} R^{(s)^T} B^{(s)^T} \\ \Theta^{(s)^T} B^{(s)^T} \end{bmatrix} \\ 0 & & \ddots & \vdots \\ \dots & \begin{bmatrix} B^{(s)} R^{(s)} & B^{(s)} \Theta^{(s)} \end{bmatrix} & \dots & -F_{res} \end{bmatrix} \begin{bmatrix} \vdots \\ \alpha^{(s)} \\ \eta^{(s)} \\ \vdots \\ \lambda \end{bmatrix}_{k+1} = \begin{bmatrix} \vdots \\ \alpha_k^{(s)} \\ \eta_k^{(s)} \\ \vdots \\ M_{res} \lambda_k \end{bmatrix} \tag{5.4.2}$$

From equations (5.4.2), we see that we don't need to assemble explicitly the reduced matrix $\widetilde{K}_{CB}$. In fact we can cancel out the equations corresponding to $\Lambda^{(s)}$, that is

$$\eta_{k+1}^{(s)} = \Lambda^{(s)^{-1}}(\eta_k^{(s)} - \Theta^{(s)^T} B^{(s)^T} \lambda_{k+1}) \tag{5.4.3}$$

Substituting equation (5.4.3) in (5.4.2) leads to the following system:

$$\begin{bmatrix} & 0 & & \begin{bmatrix} \vdots \\ R^{(s)^T} B^{(s)^T} \\ \vdots \end{bmatrix} \\ \begin{bmatrix} \dots & B^{(s)} R^{(s)} & \dots \end{bmatrix} & & -F_I \end{bmatrix} \begin{bmatrix} \vdots \\ \alpha^{(s)} \\ \vdots \\ \lambda \end{bmatrix}_{k+1} = \begin{bmatrix} \vdots \\ \alpha_k^{(s)} \\ \vdots \\ M_{2k} \end{bmatrix} \tag{5.4.4}$$

Where F_I is the interface flexibility

$$F_I = \sum_{s=1}^{N_s} B^{(s)} K^{(s)^\dagger} B^{(s)^T} \tag{5.4.5}$$

and M_{2k} is defined as

$$M_{2k} = M_{res}\lambda_k - \sum_{s=1}^{N_s} B^{(s)} \Theta^{(s)} \Lambda^{(s)^{-1}} \eta_k^{(s)} \tag{5.4.6}$$

Solving the system (5.4.4) corresponds to the FETI method.
To compute the eigenvalues of the reduced system, we apply Lanczos algorithm to the system (5.4.2) and then at each iteration we solve it first for $\eta_k^{(s)}$ in equation (5.4.3) and we apply FETI to (5.4.4).

In fortran code, we have computed in separates subroutines the following matrix-vector products $F_{res}\lambda$, $F_I\lambda$ and $M_{res}\lambda$ alongside the interface, so we can use them when needed in Lanczos procedure or for verification purpose. Equations below are computed from the right to the left respecting the arithmetic priorities:

$$F_I\lambda = \sum_{s=1}^{N_s} B^{(s)} K^{(s)^\dagger} B^{(s)^T} \lambda \tag{5.4.7}$$

$$F_{res}\lambda = \sum_{s=1}^{N_s} B^{(s)} (K^{(s)^\dagger} - \Theta^{(s)} \Lambda^{(s)^{-1}} \Theta^{(s)^T}) B^{(s)^T} \lambda \tag{5.4.8}$$

$$M_{res}\lambda = \sum_{s=1}^{N_s} B^{(s)}(K^{(s)^\dagger} - \Theta^{(s)}\Lambda^{(s)^{-1}}\Theta^{(s)^T})M^{(s)}(K^{(s)^\dagger} - \Theta^{(s)}\Lambda^{(s)^{-1}}\Theta^{(s)^T})B^{(s)^T}\lambda \tag{5.4.9}$$

Note that as we have seen in chapter 4, $G_{res} = K^{(s)^\dagger} - \Theta^{(s)}\Lambda^{(s)^{-1}}\Theta^{(s)^T}$ is symmetric.
Also note that the notation in this report is slightly different from paper by Daniel Rixen. We have used $\Lambda^{(s)}$ instead of $\Sigma^{(s)^2}$ but actually it's the same thing: the diagonal matrix of the square of eigenfrequencies which are eigenvalues.
We deal with the signed Boolean matrices $B^{(s)}$ with subroutines available in FETI code:

- $B^{(s)^T}\lambda$ is computed with subroutine **probit**
- $\sum_{s=1}^{N_s} B^{(s)}$ by a vector is computed with subroutine **interface_jump**

performing Lanczos on the reduced system

The size of the original problem is drastically reduced in each subdomain. Only the dimension of the interface remains the same. The dimension of the reduced solution vector (right hand side) is the sum of the dimension of the interface, dimension of the kernel if it exists (rigid body modes) and the dimension of the retained normal modes (for exemple, if we decide to get only the 30th first eigenmodes, then the dimension of retained normal modes will be 30).
To apply Lanczos, at every step, we will cancel out the equations corresponding to the normal modes in (5.4.2) before we call FETI to solve (5.4.4). Then the dimension of the right hand side given to FETI will be the dimension of the rigid body modes (if they exist) and the dimension of the interface. So we have slightly modified the subroutine **SPD_FETI_1LM** in order to pass directly the right hand side parameter in two parts: $\alpha^{(s)}$ and M_{2k} defined in equation (5.4.6).
And then, when FETI has solved the problem (5.4.4), we build again the solution of the reduced problem by computing the sub-vector $\eta_k^{(s)}$. So at the k-th iteration, the solution of the reduced problem in the subdomain (s) is

$$solution_k^{(s)} = (\alpha_k^{(s)}, \eta_k^{(s)}, \lambda_k^{(s)})^t \tag{5.4.10}$$

And we continue with the next iteration until the stopping criterion or max_iter is met.

comparing the results with global eigenvalues

we were able to compute some eigensolutions of the reduced system but they were very different from the results obtained previously for the global problem. This apply particularly when there are a rigid body modes in one or more subdomains.
We have tried to eliminate the rigid body modes by augmenting the diagonal of the original problem by 10%. Then, the eigenvalues obtained by both methods (global eigenvalue and Dual Craig-Bampton) were very close : $O(10^{12})$.
My advisor has also given me a sparse matrix package (including splitmesh and parallel iterative) to build a simple *Laplace problem* to test the code. The problem remains the same when there are a rigid body modes in one or more subdomains.
Practically, our Lanczos algorithm is coded in such a way that, from a certain iteration number (for example 2 times the needed eigenmodes), for all the next iteration until the stopping criterion is met, we do the following tasks:

- call LAPACK routine **dsterf.f** to compute the eigenvalues of the tridiagonal matrix T_p defined in (5.2.6).
  dsterf.f returns $INFO = 0$ when everything is OK

- test if the stopping criterion is met and then call LAPACK routine **dpteqr.f** to compute again the eigenvalues and eigenvectors of the tridiagonal matrix T_p.
  dpteqr.f returns $INFO = 0$ when everything is OK

For the calculation of the local and global eigensolutions, **dsterf** and **dpteqr** have given us exactly the same eigenvalues and have returned $INFO = 0$.
For the calculation of the eigensolutions of the reduced system by Craig-Bampton method, **dsterf** has worked well at every iteration and has returned $INFO = 0$. However, **dpteqr.f** has failed to return $INFO = 0$ (there were some negative eigenvalues).
The main difference between the reduced problem and the global problem (or local problems) is that, the reduced system (5.4.2) is not definite and is not symmetric.
Below is a short description of **dpteqr**, the entire description can be found on internet

```
      SUBROUTINE DPTEQR( COMPZ, N, D, E, Z, LDZ, WORK, INFO )
*
* -- LAPACK routine (version 3.1) --
*   Univ. of Tennessee, Univ. of California Berkeley and NAG Ltd..
*   November 2006
*
*  Purpose
*  =======
*
*  DPTEQR computes all eigenvalues and, optionally, eigenvectors
*  of a symmetric positive definite tridiagonal matrix by first
*  factoring the matrix using DPTTRF, and then calling DBDSQR to
*  compute the singular values of the bidiagonal factor.
*
*  Arguments
*  =========
*
*  INFO  (output) INTEGER
*        = 0:  successful exit.
*        < 0:  if INFO = -i, the i-th argument had an illegal value.
*        > 0:  if INFO = i, and i is:
*              <= N  the Cholesky factorization of the matrix could
*                 not be performed because the i-th principal minor
*                      was not positive definite.
*              > N   the SVD algorithm failed to converge;
*                    if INFO = N+i, i off-diagonal elements of the
*                    bidiagonal factor did not converge to zero.
*
*  =====================================================================
```

In our case, we've got $0 < INFO < N$ (ie the i-th principal minor was not positive definite). N is the dimension of the solution vector. We were able to change the value of $INFO$ by changing the stopping criterion of Lanczos algorithm.Numerically, we've found that $1 \leq INFO \leq 7$.
We have also noticed that, the following parameters of FETI code could modify the value of $INFO$: **eps_pivot, eps_null**

Conclusion

The goal of this report was to compute eigenvalues of a large problem and to assess how far we can go in the spectrum of the generalized eigenvalue problem using Craig-Bampton reduction method on a high performance parallel architecture. The first part of the study consisted of a bibliographical research, we have presented first a theoretical background of the elastodynamic problem, the generalized eigenvalue problem, a short description of FETI and the Craig-Bampton method, one of the most versatile and efficient *Component Mode Synthesis*(CMS) method.
The second part consisted of implementing the calculation of the local eigenvalues first and then building of the reduced system in order to implement the dual Craig-Bampton method.
The calculation of local eigenmodes worked very well.
For the reduced problem, we have seen that we don't need to assemble explicitly the reduced system. But, we have also noticed paticularly in presence of rigid body modes in one or more subdomains that, the eigensolutions of the reduced system were very different from the eigensolutions of the original problem. When there are no rigid body modes at all, the results were close but not exactly the same value.

Bibliography

[1] Frédéric Bourquin. *Synthèse Modale et Analyse Numérique des Multi-structures Elastiques*. PhD thesis, Université Paris 6, 1991.

[2] H. Brézis. *Analyse Fonctionnelle: Théorie et applications*. Masson, Paris, 1983.

[3] D. Buisine. Mécanique des structures. In *Cours de Licence*, number 1. Laboratoire de Mécanique de Lille.

[4] Bernardi C., Maday Y., and Rapetti F. *Discrétisations variationnelles de problèmes aux limites elliptiques*. Springer, 2004.

[5] F. Chatelain. *Eigenvalue of Matrices*. Wiley, 1993.

[6] R. Craig and M. Bampton. Coupling of substructures for dynamic analysis. *AIAA Journal*, 6:1313–1319, 1968.

[7] J.M. Cros. Parallel modal synthesis methods in structural dynamics. *Contemporary Mathematics*, 218, 1998.

[8] B. Despres. Analyse numerique matricielle. In *Cours de DEA*. Laboratoire Jacques-Louis Lions.

[9] C. Farhat and D. Rixen. *Encyclopedia of vibration volume 2, pp. 710-720*. Academic Press, New-York, 2002.

[10] C. Farhat and F-X. Roux. A method of finite element tearing and interconnecting and it's parallel solution algorithm. *Internat. J. Numer. Methods Engrg*, 32:1205–1227, 1991.

[11] C. Farhat and F-X. Roux. Implicit parallel processing in structural mechanics. In North-Holland, editor, *Computational Mechanics Advances*, volume 2. IACM, june 1994.

[12] G H. Golub and C F. Van Loan. *Matrix Computation*. Johns Hopkins University Press, 1983.

[13] Nasa Technichal Handbook. Dynamic environmental criteria. *NASA-HDBK-7005*, 2001.

[14] P. Lascaux and R. Théodor. *Analyse numérique matricielle appliquée à l'art de l'ingénieur*. Masson, Paris, 1994.

[15] P. Le Tallec. Domain decomposition method in computational mechanics. In North-Holland, editor, *Computational Mechanics Advances*, volume 1. IACM, 1994.

[16] R. Ohayon and C. SOIZE. *Structural Acoustics and Vibration*. Academic Press, 1998.

[17] A. Quarteroni and A. Vally. *Domain Decomposition Methods for Partial Differential Equation*. Oxford University Press,New-york, 1999.

[18] D-J. Rixen. Dual approach to couple dynamical submodels. In *Advances is simulation Techniques for Applied Dynamics*, Halle(Saale) Germany, 2005.

[19] K. Rothe and H. Voss. Improved dynamic substructuring on distributed memory mimd-computers, 1993.

[20] F.-X. Roux. Parallel solvers based on domain decomposition. In *Cours de DEA*. ONERA.

[21] Y Saad. *Numerical Methods For Large Eigenvalue Problems*. Halstead Press, New York, 1992.

[22] Y. Saad. *Numerical methods for large eigenvalue problems Equation*. Manchester University Press,Manchester UK, 1992.

[23] Y Saad. *Iterative Methods for Sparse Linear Systems, 2nd edition*. SIAM, Philadelpha, PA, 2003.

[24] J H. Wilkinson. *The Algebraic Eigenvalue Problem*. Clarendon Press, Oxford, 1965.