Universitat Politècnica de Catalunya
E.T.S. d'Enginyers de Camins, Canals i Ports de Barcelona
in cotutelle with
Swansea University
College of Engineering
and in collaboration with SEAT S.A.

Doctoral Thesis

Static and dynamic global stiffness analysis for automotive pre-design

Author:
Fabiola Cavaliere

Supervisors:
Prof. Pedro Díez
Prof. Sergio Zlotnik
Prof. Rubén Sevilla

Collaborator:
Dr. Xabier Larrayoz

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in

Civil Engineering

part of the Marie Skłodowska-Curie ITN-EJD funded by the European Union Horizon 2020 program with grant number 764636

Barcelona, April 20, 2022

Declarations

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed: [redacted]
Date: 20/04/2022

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed: [redacted]
Date: 20/04/2022

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed: [redacted]
Date: 20/04/2022

The University's ethical procedures have been followed and, where appropriate, that ethical approval has been granted.

Signed: [redacted]
Date: 20/04/2022

"*Imagination is more important than knowledge. For knowledge is limited, whereas imagination embraces the entire world, stimulating progress, giving birth to evolution.*"

Albert Einstein

Abstract

Universitat Politècnica de Catalunya
E.T.S. d'Enginyers de Camins, Canals i Ports de Barcelona

in cotutelle with

Swansea University
College of Engineering

and in collaboration with SEAT S.A.

Doctor of Philosophy

Static and dynamic global stiffness analysis for automotive pre-design

by Fabiola Cavaliere

In order to be worldwide competitive, the automotive industry is constantly challenged to produce higher quality vehicles in the shortest time possible and with the minimum costs of production. Most of the problems with new products derive from poor quality design processes, which often leads to undesired issues in a stage where changes are extremely expensive. During the preliminary design phase, designers have to deal with complex parametric problems where material and geometric characteristics of the car components are unknown. Any change in these parameters might significantly affect the global behaviour of the car. A target which is very sensitive to small variations of the parameters is the noise and vibration response of the vehicle (NVH study), which strictly depends on its global static and dynamic stiffness. In order to find the optimal solution, a lot of configurations exploring all the possible parametric combinations need to be tested. The current state of the art in the automotive design context is still based on standard numerical simulations, which are computationally very expensive when applied to this kind of multidimensional problems. As a consequence, a limited number of configurations is usually analysed, leading to suboptimal products. An alternative is represented by reduced order method (ROM) techniques, which are based on the idea that the essential behaviour of complex systems can be accurately described by simplified low-order models.

This thesis proposes a novel extension of the proper generalized decomposition (PGD) method to optimize the design process of a car structure with respect to its global static and dynamic stiffness properties. In particular, the PGD method is coupled with the inertia relief (IR) technique and the inverse power method (IPM) to solve, respectively, the parametric static and dynamic stiffness analysis of an unconstrained car structure and extract its noise and vibrations properties. A main advantage is that, unlike many other ROM methods, the proposed approach does not require any pre-processing phase to collect prior knowledge of the solution. Moreover, the PGD solution is computed with only one offline computation and presents an explicit dependency on the introduced design variables. This allows to compute the solutions at a negligible computational cost and therefore opens the door to fast optimisation studies and real-time visualisations

of the results in a pre-defined range of parameters. A novel algebraic approach is also proposed which allows to involve both material and complex geometric parameters, such that shape optimisation studies can be performed. In addition, the method is developed in a nonintrusive format, such that an interaction with commercial software is possible, which makes it particularly interesting for industrial applications. Finally, in order to support the designers in the decision-making process, a graphical interface app is developed which allows to visualise in real-time how changes in the design variables affect pre-defined quantities of interest.

Acknowledgements

With this thesis my experience as a PhD student comes to an end. It was a real privilege to have the opportunity to be part of the ProTechTion project, funded by the European Union's EU Framework Programme for Research and Innovation Horizon 2020 (grant number 764636). A long list of people contributed to make these years amazing, and I would like to thank some of them.

First of all I would like to thank my supervisors: Prof. Pedro Díez and Prof. Sergio Zlotnik from BarcelonaTech, Prof. Rubén Sevilla from Swansea University and Dr. Xabier Larrayoz from Seat S.A.. I will miss our joint meetings. Each of them was a unique chance to learn something new in a funny and relaxed environment. Thank you for being always open to discuss my doubts, for your trust in me and for your empathy. You all contributed to my growth as a researcher and as a person. I would also like to thank Alberto Romero from Seat S.A. for the technical trainings and for generously transferring his knowledge to me. Also, I am thankful to the BETA CAE team, who helped me in using their powerful software to improve the quality of my work.

During these PhD years, I met new amazing colleagues from the ProTechTion project and from the LaCàn lab. Thank you all for the nice lunches, coffee breaks, conferences and events. Especially, I would like to thank Christina, who shared the whole path with me from the very beginning. Thank you Chris for all the fun!

Thanks to all my friends, who have always supported me from any corner of Europe. You are simply essential in my life.

Thanks to Barcelona that welcomed me and made me feel at home. Thanks for the new habits, the sun, the warmness and for all the amazing people I had the chance to meet.

A special thank to Bernat, that makes life much sunnier.

And finally, thanks to my parents and to my brother for their unconditional love and for supporting all my choices. I owe you everything I am.

Contents

List of Figures

List of Tables

List of Abbreviations

SBE	**S**imulation-**B**ased **E**ngineering
ROM	**R**educed **O**rder **M**odelling
POD	**P**roper **O**rthogonal **D**ecomposition
PGD	**P**roper **G**eneralised **D**ecomposition
CAD	**C**omputer-**A**ided **D**esign
NVH	**N**oise **V**ibration and **H**arshness
BIW	**B**ody **I**n **W**hite
FE	**F**inite **E**lement
QoI	**Q**uantity **o**f **I**nterest
ROM	**R**educed **O**rder **M**odelling
IPM	**I**nverse **P**ower **M**ethod
IR	**I**nertia **R**elief
ROM	**R**educed **O**rder **M**odelling
DV	**D**esign **V**ariable
DoE	**D**esign **O**f **E**xperiment
ETS	**E**quivalent **T**orsional **S**tiffness
GA	**G**enetic **A**lgorithm

To my family...

1 Introduction

1.1 The automotive design process: state of the art and challenges

The automotive design process represents an extremely complex and time-consuming engineering problem. A vehicle itself is a complex structure made of thousands of components and several systems which must work together under the most diverse road, traffic and weather conditions. The development process starts years before the vehicle is ready for production and the marketplace. During this time, teams of engineers from the most diverse areas of expertise have to cooperate and work in parallel in order to predict and optimise every single aspect related to the product, from its manufacturing, to its life-cycle and final disposal. On top of the inherent complexity of a car development process, modern vehicles are required to meet hundreds of targets to satisfy increasingly ambitious customers needs and governmental regulations. To make things worse, the current global market is pushing for producing new products at a faster pace. In such a competitive scenario, the only chance for the automotive companies to keep the pace is to increase the efficiency of their development processes by reducing both the time-to-market and the costs of production. It has been estimated that a poor quality of the design stage is the cause of the 85% of problems with new products which do not work as they should, take too long to bring to market, or cost too much [1]. Clearly, a good design process has a major impact on the quality of the final product. Conversely, good manufacturing cannot compensate the design flaws. The conventional design process is affected by a peculiar *paradox* [2], as shown in Fig. 1.1. The graphic shows that the knowledge about a design problem, thus the

ability to find better solutions, increases during time. As the project proceeds, the designers are forced to take decisions which, many times, cannot be modified later on. As a consequence, the increasing knowledge about the design problem remains unexploited, while the freedom to make changes decreases more and more (see Fig. 1.1).

During the 20th century, thanks to the tremendous growth of com-

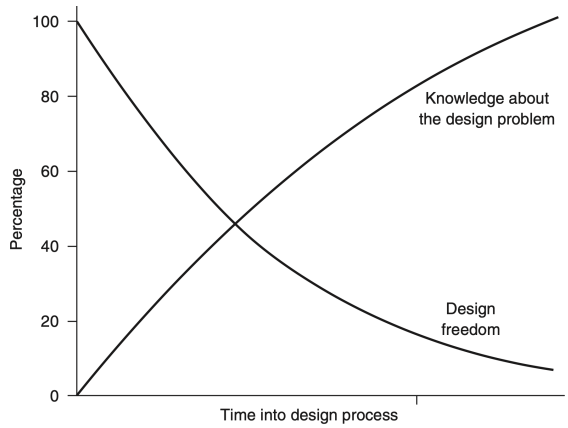


Figure 1.1: The design paradox between design knowledge and design freedom. (Source: Dieter and Schmidt [2])

puting power and the advances in computational science, a new discipline known as simulation-based engineering (SBE) [3] has been established in the industrial sector. SBE methods are able to emulate the physical behaviour of an engineering system by employing numerical methods (e.g. finite elements, finite volumes, finite differences) to solve complex mathematical models. With SBE, the industry started to substitute physical prototypes and experimental testing with virtual prototyping and computer simulations. Ideally, these numerical simulations could be used to explore the whole design space and to identify the optimal solutions already during the preliminary phase of the design process, avoiding the design paradox described above. However, the drawback of standard SBE methods is that, every time a design variable (input of the problem) is changed, the computational model has to be updated and a new numerical simulation has to be

performed in order to evaluate its performance. Consequently, as shown in Fig. 1.2, the same problem is to be solved several times with updated design variables until the optimal design is obtained and the development process can keep on. Due to the characteristic large scale of a detailed (also referred

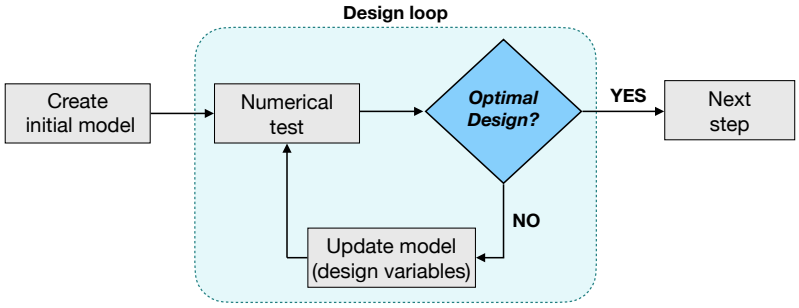


Figure 1.2: Cyclic standard design process.

as *high-fidelity* or *full-order*) vehicle model, standard numerical methods can be extremely time-consuming and computationally expensive, even if high performance computing resources, usually available in industrial contexts, are used. Considering that the number of required simulations may reach millions in order to test all the possible configurations, exploring the whole design space by performing full-order simulations is clearly not affordable. For this reason, many times designers are only able to test a limited number of configurations, mainly based on their experience and intuition, accepting that this could bring unexpected problems in a later stage, causing delays in the whole development process and leading to a suboptimal product.

In this general scenario, alternative solutions to optimise the design process are needed. One possible approach could be to increase the dimension of the problem by introducing the design parameters as extra coordinates and solve the resulting multidimensional problem. However, the issue with high-dimensional problems is that, if standard mesh-based methods are used, the computational cost explodes exponentially with the number of design variables involved. More precisely, if D is the number of variables and M is the number of nodes used to discretise each design space coordinate,

the total number of degrees of freedom scales with M^D, which becomes once again computationally unaffordable in industrial applications.
In order to overcome this computational challenge, known as *curse of dimensionality* phenomenon, a lot of research has been conducted in the last two decades in the field of reduced order modelling (ROM) techniques [4]. As suggested by the name, ROM methods aim at reducing the numerical complexity of high-dimensional problems by constructing approximated low-order models which can speed up the calculation time, while preserving the expected accuracy within a controlled error. Commonly, the approximated model, also known as *surrogate model* or *metamodel* [5, 6], is constructed in an *offline* stage. Afterwards, in the *online* phase, the model can be used to obtain multiple-query solutions for different values of the design variables or to perform optimisation studies by running real-time efficient calculations.

Usually, ROM approaches are classified into *a posteriori* and *a priori* methods. The more popular are a posteriori methods. These approaches first need to solve the full-order problem for a suitably chosen set of parameters. The obtained solutions (*snapshots*) are taken as a basis of functions describing the most relevant characteristics of the solution. The (reduced) basis is then used as a training set to solve problems corresponding to different values of the parameters, at a much lower computational cost during the online phase. Clearly, the accuracy of the solution for any new value of the parameters highly depends on the right choice of the representative set of problems (*snapshots*) used to extract the reduced basis. Some of the most popular a posteriori methods are the Krylov-based methods [7], the reduced basis method [8], and the Proper Orthogonal Decomposition (POD) technique [9].

An appealing alternative is represented by the proper generalised decomposition (PGD) method [10–12], which is instead an *a priori* approach, in the sense that it computes the reduced basis without relying on previously computed full-order solutions associated with arbitrary samples of the parametric space. This is possible thanks to the main assumption

of the method, that is to treat the parameters as extra coordinates and approximate the solution of the resulting high-dimensional problem as a sum of "rank-one" terms. Each of these "rank-one" terms is given by the product of basis functions depending explicitly on the coordinates of the problem (spatial and parametric coordinates). This compact approximated expression of the solution, also known as *computational vademecum*, is particularised, in the online phase, for any set of the design variables at a negligible computational cost and with very low computational resources. As a consequence, it can be uploaded on light computational devices (such as tablets or smartphones), such that the visualisation of the results, optimisation studies or inverse analysis can be performed in real-time.

Based on this short description of the method, it is already clear that PGD has a big potential in the design context, as it supports the designers in the decision-making already in the preliminary phase of the development process. In fact, with PGD, the standard design loop shown in Fig. 1.2 can be substituted by the linear design process of Fig. 1.3. The price to pay in order to avoid the continuous iterations is to invest some time and computational effort in an offline phase. During this stage, the parametric model is first built and then solved by means of just one offline PGD computation, providing an explicit parametric PGD solution. Thus, in the online phase, designers can explore the whole design space in real-time by visualising and post-processing the results for every possible combination of the design variables.

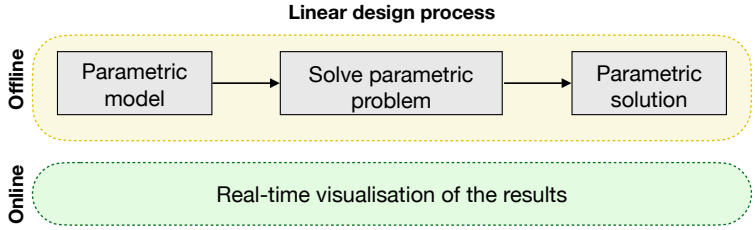


Figure 1.3: Linear design process. The problem is defined and solved in a multidimensional space, avoiding the continuous iterations typical of the conventional design loop.

The PGD method was successfully tested in the most diverse classes of problems, such as flow problems [13–18], thermal problems [19–21], solid mechanics [22–24], fracture mechanics [25, 26], geophysical problems [27, 28], elastic metamaterials and coupled magneto-mechanical problems [29, 30]. However, it still presents some limitations.

First, the application to geometrically parametrised problems remains particularly challenging. This is due to the fact that one essential requirement of the PGD method is that the input quantities of the problem (such as the stiffness and mass matrix) are to be expressed with an explicit and separated dependence on the parameters. This is clearly not an easy task, especially when complex shape parametrisations, like the one which are typical in the automotive context, are considered. Previous works that dealt with parametric shapes are usually limited to simple geometric dependence [31–35]. Other authors proposed a technique based on the idea that a parent domain can be associated to the parametric domain in order to introduce the parametric dependency on the geometry in the governing equations [32, 36]. More recently, another approach was proposed [37] in which the control points characterising the NURBS curves or surfaces used in CAD representation are defined as the geometric parameters of the problem, making an effort towards the full integration of computational geometry and analysis in the field or ROMs [38–40]. Another important drawback of the original PGD method, when compared to other *a posteriori* approaches, is the *intrusivity* of its implementation. In fact, the standard PGD requires an *intrusive* modification of the finite element (FE) formulation of the problem, as described in several works [33, 37, 41–44]. Clearly, this makes the method not suitable for an industrial context, where commercial software with inaccessible source codes is usually employed. This limitation has motivated the development of nonintrusive implementations of the PGD rationale for solid [45] and fluid [46] mechanics problems.
As it will be fully described later, this thesis proposes a *hybrid* version of the PGD method which solves both the intrusivity and complex shape optimisation issues. On one hand, it can still be considered an a priori

approach, as it does not require any previous full-order computation of the solution. On the other hand, a pre-process step is added, where the input data are only sampled (without solving the problem) in the parametric domain of interest and then expressed in the required explicit format.

With the goal of proving the big potential of PGD in the context of design optimisation, this thesis focuses on one of the several targets which must be fulfilled during the development process of a new car, that is the Noise, Vibration and Harshness (NVH) behaviour of the vehicle. Next section reviews the main challenges in the context of the standard NVH analysis and introduces the advantages of developing a PGD-based methodology to solve it.

1.2 Noise, Vibration and Harshness (NVH) analysis

In the last decades, the NVH performance of a vehicle has had an increasingly stronger impact on the customer perception of ride comfort and brand quality, becoming one of the most prioritised attributes when purchasing a new car. In order to be competitive in the global market, car manufacturers need to improve the NVH properties of their products without degrading any other (often conflicting) target such as crashworthiness, light-weight, safety, ecological impact and styling. The automotive NVH optimisation represents a multidisciplinary field of research in continuous evolution. To give some examples, it needs experimental techniques for the measurements of noise and vibrations [47], methods based on psychoacoustics to evaluate the human perception of the discomfort [48], or strategies to correlate numerical and experimental results [49]. Several solutions have been proposed in order to improve the NVH experienced by the occupants of the cabin. Most of the times, the attention is focused on the source of noise and vibrations [50]. In the past, this was mainly represented by the engine and powertrain. Today, vehicles are much quieter than several years ago, and the focus has shifted to other sources of noise, such as road excitation and tyre performance [51, 52]. Also, in the last years a lot of research has been

done to study the NVH properties of hybrid and electric vehicles [53–56].

Many times, the most effective way of improving the NVH performance is to act on the material and geometric properties of the car body structure resulting from the assembly of all the substructural components, which in car design is referred as global Body-in-White (BiW) structure, where the term *global* is used to differentiate it from the *local* components. In fact, the NVH response is particularly sensitive to changes in the design parameters, as it strictly depends on the global static and dynamic stiffness of the vehicle body structure. The dynamic properties, which provide important information about the vibrational behaviour of the BiW, are usually obtained in terms of *natural frequencies* and *mode shapes* by performing the standard FE modal analysis. The standard FE static analysis, instead, allows to evaluate the global static stiffness and extract indicators of the ride comfort. Due to the large number of configurations to be tested and the high cost of each static and dynamic simulation involved, the NVH analysis is traditionally performed at a later stage of the design process, risking to encounter late undesired issues.

As discussed in the previous section, in order to speed up the design cycle and avoid the unsustainable waste of time and resources in prototyping non-optimal products, the automotive industry is highly interested in new software [57] and advanced simulation-based methodologies able to predict the vehicle performance already during the preliminary phase of the development process [58]. Most of the research efforts of the last years concentrated on developing new techniques that build equivalent simplified FE models of the BiW characterised by beams, joints and panels, [59–61]. This allows to drastically reduce the degrees of freedom, such that the calculation time of each simulation is much less and fast optimisation studies can be performed. Of course, a pre-process step is needed where full-order simulations are performed in order to calibrate the properties of the equivalent reduced model.

The approach proposed in this work aims at avoiding the pre-process

phase, where a choice on the set of full-order simulations has to be made. Moreover, the extreme simplification of the original model with beam elements might lead to poor accuracy. For this reason, the idea of this thesis is to construct a parametric detailed FE model of the BiW structure, as the one shown in Fig. 1.4. The material properties of the car components can be defined in a parametric way. Similarly, the geometric properties can be defined parametrically using the morphing tools available in modern commercial software. Once the parametric model is built, the goal is to

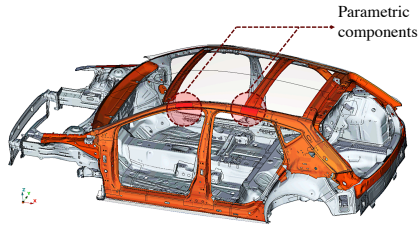


Figure 1.4: Parametric BiW structure. (*Courtesy of SEAT S.A.*)

follow a linear design process, as said in the previous section. However, solving the parametric global static and dynamic stiffness analysis of the BiW presents several challenges. The goal of this thesis is to address this challenges by developing a PGD-based strategy which can finally provide a tool to accelerate and improve the preliminary phase of the design process. The main objectives and contributions of the present work are described in next section.

1.3 Objectives and outline of the Doctoral thesis

The goal of this thesis is to propose a new PGD-based method to perform the parametric NVH study of a vehicle BiW structure characterised by material and complex shape design variables. The proposed methodology allows to efficiently explore the variation in the design space of the BiW static and dynamic global stiffnesses. In this way, the NVH performance is evaluated already in the pre-design phase of the development process.

To address the challenges posed by a parametric NVH study, this thesis presents the following contributions:

- In the static case, the PGD is coupled with the inertia relief method (PGD-IR), which is an approach widely used by the industry to perform parametric static analysis of a unconstrained BiW structures.

- In the dynamic case, the PGD is coupled with the Inverse Power Method (PGD-IPM) to develop a parametric eigensolver able to extract the dynamic properties of the BiW in terms of lowest natural frequencies and corresponding mode shapes.

- A *hybrid* version of the PGD method is developed, which preserves the *nonintrusivity* of the methodology, allowing the integration within the commercial software and permitting to deal with complex shape parametrisations.

- The proposed PGD-IR and PGD-IPM approaches are validated by performing the static and dynamic global stiffness analysis of a free-free 3D structure characterised by geometric and material parameters.

- The method is applied to an industrially representative problem to perform the NVH analysis of a parametric BiW structure.

- A multi-objective optimisation analysis is performed as a direct and straightforward exploitation of the proposed solution.

- A graphical interface app is developed which allows to visualise in real-time how changes in the design variables affect pre-defined quantities of interest (QoIs).

The thesis is structured in five chapters, including this introduction. The next chapters are organised as follows. Chapter 2 develops a computational framework for the solution of unconstrained parametric structural problems with the inertia relief (IR) and the PGD method. Also, the *hybrid* approach to deal with geometric parameters and preserve the *nonintrusivity* is detailed. The applicability and potential of the developed technique is shown using a simple linear elastic three dimensional test case with one material and one geometric parameter. Chapter 3 presents a parametric version of the inverse power method (IPM) by using the PGD rationale. With only one offline computation, the proposed PGD-IPM approach provides an analytical parametric expression of few smallest (in magnitude) eigenvalues (or natural frequency) and corresponding eigenvectors (shape modes), which contains all the possible solutions for every combination of the parameters within pre-defined ranges. The same three dimensional structure used in the static case is tested to validate the dynamic results. Chapter 4 finally applies the PGD-IR and PGD-IPM algorithms to an industrial example to perform the parametric NVH analysis of a representative BiW structure. A parametrised FE model of the BiW is built by means of a pre-processor commercial software, which allows to take into account any kind of material and geometric parameters. A comparison between the results obtained through the PGD-NVH solver and the corresponding full-order FE simulations performed by means of the MSC-Nastran commercial software is shown to validate the proposed method. In addition, an optimisation study is presented to find the optimal materials and shape properties with respect to the NVH performance. Finally, a graphical interface app is developed which allows to visualise the results in real-time. Chapter 5 summarises the most relevant conclusions and outlooks of this thesis.

1.4 Publications, conferences and awards

Articles in indexed journals:

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: Nonintrusive reduced order model for parametric solutions of inertia relief problems. In: *International Journal for Numerical Methods in*

Engineering 122.16 (2021), pp. 4270–4291.

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: Nonintrusive parametric solutions in structural dynamics. In: *Computer Methods in Applied Mechanics and Engineering* 389 (2022), p. 114336. ISSN: 0045-7825.

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: Nonintrusive parametric NVH study of a BiW structure. *Submitted* (2022).

International conference contributions:

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: A Proper Generalized Decomposition (PGD) approach with inertia relief for the solution of unconstrained parametric static problems. ADMOS (International Conference on Adaptive Modelling and Simulation). Alicante, Spain (2019). Poster.

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: A Proper Generalized Decomposition approach with Inertia Relief (PGD-IR) for the static analysis of geometrically parametrised unconstrained structures. MORTech (International Workshop on Reduced Basis, POD and PGD Model Reduction Techniques). Paris, France, (2019). Poster.

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: PGD for inertia relief problems and parametric modal analysis. WCCM (World Congress in Computational Mechanics and European Congress on Computational Methods in Applied Sciences and Engineering). Virtual congress, (2021).

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: A nonintrusive reduced order method for the NVH assessment and automotive structural dynamics. UKACM (UK Association for Computational Mechanics). Virtual congress, (2021). Extended abstract.

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: Nonintrusive proper generalized decomposition method for the design optimization of a car. ADMOS (International Conference on Adaptive Modelling and Simulation). Virtual congress, (2021).

Awards:

- Public award for the "Best 4 minutes thesis" in the Catalonian regional contest (2020).

- Mike Crisfield Prize for the best presentation - UKACM conference (2021).

- Best presentation award - ADMOS conference (2021).

2 Parametric static analysis of unconstrained structures

Unconstrained structures are widespread in the automotive, aerospace and naval industry. As is well known, due to the singularity of the stiffness matrix, conventional static analyses cannot be performed if the system undergoes rigid body motions. At the same time, imposing dummy constraints in order to make a free-body system statically determinate leads to unrealistic reaction forces and, as a consequence, an unrealistic distribution of the internal stresses. The inertia relief (IR) method represents an attractive alternative for solving unconstrained structural problems without resorting to the more expensive full dynamic analysis [62]. The main idea is to counteract the unbalanced applied loads by a set of rigid body accelerations, the latter providing body forces which are distributed over the structure in such a way that the applied forces are equilibrated and the static analysis can be performed. The technique is available into most of the commercial finite element packages and it has been widely used by the industry in different fields [63–71].

The static global stiffness analysis of a vehicle body structure is a common example that involves the computational simulation of an unconstrained configuration using the IR method. The BiW global stiffness plays a significant role in the design process of a car. As already discussed in the introduction, an important challenge of this problem is the number of parameters (e.g. geometry, material) to be considered during the design of a BiW. As any change in the material or geometrical characteristics of the car components might have considerable effects on the global behaviour of the structure, the number of simulations that are required to account for the

whole range of the involved parameters becomes prohibitive when classical numerical approaches are employed. As a consequence, the possibility to perform parametric studies, shape optimisation or inverse identification in the context of a BiW remains a challenge.

In order overcome this issue, this chapter proposes a PGD-IR method for the solution of an unconstrained structure characterised by material and/or geometric parameters. A *hybrid* algebraic version of the PGD method is developed, which allows to deal with complex shape parametrisations. The structure of this chapter is as follows. Section 2.1 presents the problem statement in terms of an elastodynamic boundary value problem. Section 2.2 briefly reviews the idea behind the IR method for a non-parametric problem. The proposed PGD-IR approach is presented in Section 2.3, where the PGD approach is proposed to solve the parametric IR problem. Also, the hybrid algebraic approach to deal with geometric parameters is detailed. In Section 3.3 one numerical examples is used to show the potential of the proposed method. Finally, Section 2.5 summarises the conclusions. The content of this chapter is based on the following publication:

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: Nonintrusive reduced order model for parametric solutions of inertia relief problems. In: *International Journal for Numerical Methods in Engineering* 122.16 (2021), pp. 4270–4291.

2.1 Finite elements formulation of elastodynamic problems

2.1.1 Problem statement

Let us consider an open bounded domain $\Omega \subset \mathbb{R}^d$, where d is the number of spatial dimensions. The boundary of the domain is assumed to be partitioned into the the disjoint parts Γ_D and Γ_N, where Dirichlet and Neumann boundary conditions are prescribed respectively. The strong form of the elastodynamic problem using the classical Voigt notation ([72]) can

be written as

$$\begin{cases} \rho\ddot{\boldsymbol{u}} - \boldsymbol{\nabla}_S^T\boldsymbol{\sigma} = \boldsymbol{b} & \text{in} \quad \Omega \times (0,T] \\ \boldsymbol{u} = \boldsymbol{u}_D & \text{on} \quad \Gamma_D \times (0,T] \\ \boldsymbol{E}^T\boldsymbol{\sigma} = \boldsymbol{t}_N & \text{on} \quad \Gamma_N \times (0,T], \\ \boldsymbol{u} = \boldsymbol{u}_0 & \text{in} \quad \Omega \times \{0\} \\ \dot{\boldsymbol{u}} = \boldsymbol{v}_0 & \text{in} \quad \Omega \times \{0\} \end{cases} \tag{2.1}$$

where $\boldsymbol{u}$ is the displacement field, $\ddot{\boldsymbol{u}}$ denotes the acceleration, $\boldsymbol{\sigma}$ is a vector containing the extensional and shear stress components of the Cauchy stress tensor, $\boldsymbol{b}$ is a external body force vector, T is the final time of interest, $\boldsymbol{E}$ is a matrix accounting for the normal direction to the boundary, $\boldsymbol{u}_D$ and $\boldsymbol{t}_N$ are the prescribed displacement and traction vectors on the Dirichlet and Neumann boundaries respectively and $\boldsymbol{u}_0$ and $\boldsymbol{v}_0$ are the initial position and velocity respectively. In three dimensions, the matrix operator $\boldsymbol{\nabla}_S$ and the matrix $\boldsymbol{E}$ are given by

$$\begin{aligned} \boldsymbol{\nabla}_S &:= \begin{bmatrix} \partial/\partial x_1 & 0 & 0 & \partial/\partial x_2 & \partial/\partial x_3 & 0 \\ 0 & \partial/\partial x_2 & 0 & \partial/\partial x_1 & 0 & \partial/\partial x_3 \\ 0 & 0 & \partial/\partial x_3 & 0 & \partial/\partial x_1 & \partial/\partial x_2 \end{bmatrix}^T, \\ \boldsymbol{E} &:= \begin{bmatrix} n_1 & 0 & 0 & n_2 & n_3 & 0 \\ 0 & n_2 & 0 & n_1 & 0 & n_3 \\ 0 & 0 & n_3 & 0 & n_1 & n_2 \end{bmatrix}^T, \end{aligned} \tag{2.2}$$

with $\boldsymbol{n}$ being the outward unit normal vector to $\partial\Omega$. For a linear elastic material, the generalised Hooke's law expresses a linear relation between the stress vector, $\boldsymbol{\sigma}$, and the strain vector, $\boldsymbol{\varepsilon}$, namely

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}, \tag{2.3}$$

where $\boldsymbol{\varepsilon} := \boldsymbol{\nabla}_S \boldsymbol{u}$ and $\mathbf{D}$ is a symmetric positive definite matrix depending upon the Young modulus, E, and the Poisson ratio, ν. In three dimensions

$$\mathbf{D} := \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & & \\ \nu & 1-\nu & \nu & \mathbf{0}_d & \\ \nu & \nu & 1-\nu & & \\ & \mathbf{0}_d & & (1-2\nu)/2\mathbf{I}_d & \end{bmatrix}. \tag{2.4}$$

The weak formulation of the strong form of Eq. (3.1) reads as follows: given $\boldsymbol{u}_D$ on Γ_D and $\boldsymbol{t}_N$ on Γ_N, find $\boldsymbol{u} \in \mathcal{H}^1_D(\Omega) := \{\boldsymbol{w} \in \mathcal{H}^1(\Omega) \mid \boldsymbol{w} = \boldsymbol{u}_D \text{ on } \Gamma_D\}$ such that

$$\int_\Omega \rho \boldsymbol{v} \cdot \ddot{\boldsymbol{u}} d\Omega + \int_\Omega \boldsymbol{\nabla}_S \boldsymbol{v} \cdot (\mathbf{D} \boldsymbol{\nabla}_S \boldsymbol{u})\, d\Omega = \int_\Omega \boldsymbol{v} \cdot \boldsymbol{b} d\Omega + \int_{\Gamma_N} \boldsymbol{v} \cdot \boldsymbol{t}_N d\Gamma, \tag{2.5}$$

for all $\boldsymbol{v} \in \mathcal{H}^1_0(\Omega) := \{\boldsymbol{w} \in \mathcal{H}^1(\Omega) \mid \boldsymbol{w} = \mathbf{0} \text{ on } \Gamma_D\}$.

2.1.2 Spatial discretisation

A partition of the domain Ω in a set of n_{el} disjoint elements Ω_e is considered. Following the classical isoparametric framework, the approximation of the displacement field is defined in a reference element, $\widehat{\Omega}$, with local coordinates $\boldsymbol{\xi}$, as

$$\boldsymbol{u}(\boldsymbol{\xi}) \simeq \boldsymbol{u}^h(\boldsymbol{\xi}) := \sum_{j=1}^{n_{en}} \mathbf{U}_j N_j(\boldsymbol{\xi}), \tag{2.6}$$

where $\mathbf{U}_j$ are nodal values, N_j are polynomial shape functions of order p in the reference element and n_{en} is the number of nodes per element. The isoparametric mapping, given by

$$\begin{aligned} \boldsymbol{\varphi}^e : \widehat{\Omega} \subset \mathbb{R}^d &\longrightarrow \Omega_e \subset \mathbb{R}^d \\ \boldsymbol{\xi} &\longmapsto \boldsymbol{\varphi}^e(\boldsymbol{\xi}) := \sum_{j=1}^{n_{en}} \boldsymbol{x}_j^e N_j(\boldsymbol{\xi}), \end{aligned} \tag{2.7}$$

is employed to establish the relation between the reference element, $\widehat{\Omega}$, and a generic physical element, Ω_e, with nodes $\{\boldsymbol{x}_j\}_{j=1,\dots,n_{en}}$. Employing the isoparametric mapping of Eq. (2.7), the element and boundary integrals

are mapped to the reference space. By using the approximation of the displacement field given by Eq. (2.6) and selecting the space of weighting functions to be equal to the space spanned by the interpolation functions, the following system of ordinary differential equations is obtained

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}. \tag{2.8}$$

As usual in a finite element context, the global mass matrix $\mathbf{M}$, the global stiffness matrix $\mathbf{K}$ and the global forcing vector $\mathbf{F}$ are obtained by assembling the elemental contributions given by

$$\begin{aligned}
\mathbf{M}^e &= \int_{\widehat{\Omega}} \rho \boldsymbol{N}^T \boldsymbol{N} |\mathbf{J}^e_{\boldsymbol{\varphi}}| d\hat{\Omega}, \\
\mathbf{K}^e &= \int_{\widehat{\Omega}} (\mathbf{B}^e)^T \mathbf{D}^e \mathbf{B}^e |\mathbf{J}^e_{\boldsymbol{\varphi}}| d\hat{\Omega}, \\
\mathbf{F}^e &= \int_{\widehat{\Omega}} \boldsymbol{N}^T \boldsymbol{b} |\mathbf{J}^e_{\boldsymbol{\varphi}}| d\hat{\Omega} + \int_{\hat{\Gamma}} \boldsymbol{N}^T \boldsymbol{t} \|\mathbf{J}^e_{\Gamma}\| d\hat{\Gamma}.
\end{aligned} \tag{2.9}$$

In the above expressions $\mathbf{B}^e := (\mathbf{J}^e_{\boldsymbol{\varphi}})^{-1} \boldsymbol{\nabla}_S \boldsymbol{N}$ is the strain-displacement matrix, $\mathbf{J}^e_{\boldsymbol{\varphi}}$ is the Jacobian of the isoparametric mapping, $\mathbf{J}^e_{\Gamma}$ is the Jacobian of the restriction of the isoparametric mapping to an element face and the matrix $\boldsymbol{N}$, in three dimensions, is given by

$$\boldsymbol{N} := \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & \cdots & N_{n_{en}} & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 & \cdots & 0 & N_{n_{en}} & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 & \cdots & 0 & 0 & N_{n_{en}} \end{bmatrix}. \tag{2.10}$$

2.2 The inertia relief method

In this section, a short review of the IR method is presented. As already mentioned, the IR method [62] is available in many commercial FE packages and it is widely used in industry to solve unconstrained structural problems without resorting to the more expensive full dynamic analysis.

When constant unbalanced external loads are applied to an unconstrained structure, the whole system undergoes a steady-state rigid body acceleration in each free direction and, due to the mass of the system, inertial forces are

generated that deform elastically the body. The trajectory as rigid body of the system (as if it was infinitely stiff) is described by a displacement field $\mathbf{U}_r(t)$ such that $\mathbf{K}\mathbf{U}_r(t) = \mathbf{0}$, and therefore also $\mathbf{K}\dot{\mathbf{U}}_r(t) = \mathbf{K}\ddot{\mathbf{U}}_r(t) = \mathbf{0}$. The global displacement $\mathbf{U}$ has to be complemented with the elastic deformation, namely

$$\mathbf{U} = \mathbf{U}_r + \mathbf{U}_e. \tag{2.11}$$

The elastic deformation field $\mathbf{U}_e$ is important to analyse the internal stresses created by the motion and also to assess other quantities of interest.

The key idea of the IR method is to compute $\mathbf{U}_e$ solving the *static* problem

$$\mathbf{K}\mathbf{U}_e = \mathbf{F}_{\text{eq}} \tag{2.12}$$

where the forces $\mathbf{F}_{\text{eq}}$ are equilibrated, that is the resultant forces and moments are zero. Despite matrix $\mathbf{K}$ is singular, the fact that $\mathbf{F}_{\text{eq}}$ is equilibrated guaranties that system (2.12) is solvable with a family of infinite solutions, all equivalent up to a rigid body motion. Isostatic constrains (as many as rigid body modes, 3 in 2D and 6 in 3D) have to be set to compute one of these solutions (they all produce the same strains and stresses).
The inertia relief methods computes the equilibrated forces as

$$\mathbf{F}_{\text{eq}} = \mathbf{F} - \mathbf{M}\ddot{\mathbf{U}}_r \tag{2.13}$$

noting that $\ddot{\mathbf{U}}_r$ is the rigid body mode (recall that $\mathbf{K}\ddot{\mathbf{U}}_r(t) = \mathbf{0}$) such that $\mathbf{F}_{\text{eq}}$ is equilibrated. Thus, Eq. (2.12) is derived from (2.8) assuming that $\ddot{\mathbf{U}}_e = \mathbf{0}$ (which stands under a constant load, and therefore constant acceleration).

The rigid body acceleration vector $\ddot{\mathbf{U}}_r$ can be expressed as a linear combination of the (6 in 3D) rigid body modes, namely

$$\ddot{\mathbf{U}}_r = \boldsymbol{\Phi}\boldsymbol{\alpha}, \tag{2.14}$$

where each column of the *rigid body transformation matrix* $\boldsymbol{\Phi}$ corresponds to one of the n_r rigid body modes ($n_r = 6$ in three dimensions) and the

coefficient vector $\boldsymbol{\alpha}$ is seen as containing the acceleration of each of the rigid body modes. Introducing the expression of Eq. (2.14) in Eqs. (2.13) and (2.12), and pre-multiplying by $\boldsymbol{\Phi}^T$, the following equation is obtained

$$\boldsymbol{\Phi}^T\mathbf{F} - \boldsymbol{\Phi}^T\mathbf{M}\boldsymbol{\Phi}\boldsymbol{\alpha} = \mathbf{0}, \tag{2.15}$$

which guaranties that the right-hand side term in Eq. (2.12) is an equilibrated system of forces (sum of forces and sum of moments equal to zero). It is worth noting that $\boldsymbol{\Phi}^T\mathbf{K} = \mathbf{0}$ because the eigenmodes are mutually orthogonal and the eigenvalues (frequencies) associated to the rigid body modes are zero. The vector of unknown accelerations $\boldsymbol{\alpha}$ providing the equilibrated forces $\mathbf{F}_{\text{eq}}$ is therefore computed by solving the system

$$\boldsymbol{\alpha} = \left(\boldsymbol{\Phi}^T\mathbf{M}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\mathbf{F}, \tag{2.16}$$

where $\boldsymbol{\Phi}^T\mathbf{M}\boldsymbol{\Phi}$ and $\boldsymbol{\Phi}^T\mathbf{F}$ are a reduced 6×6 mass matrix and a reduced 6×1 load vector, respectively. To completely define the rigid body acceleration vector $\ddot{\mathbf{U}}_r$ in Eq. (2.14) it is only necessary to compute the rigid body modes of the structure, that is the 6 columns of matrix $\boldsymbol{\Phi}$. By definition, a rigid body mode is a displacement vector which do not cause any static force. As a consequence, a simple way to analytically compute the six (in 3D) rigid body modes $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, ..., \boldsymbol{\phi}_6]$ is to impose their definition given by the system $\mathbf{K}\boldsymbol{\Phi} = \mathbf{0}$, which corresponds to the kernel of the global stiffness matrix. To this end, the set of indices corresponding to the $n_d = d \times n_n$ degrees of freedom, with n_n being the number of mesh nodes, is partitioned into the reference set s and the remaining set l. To simplify the notation, and without loss of generality, the set s is assumed to correspond to the last n_r degrees of freedom. The system of equations to obtain the rigid body modes is then written as

$$\begin{bmatrix} \mathbf{K}_{ll} & \mathbf{K}_{ls} \\ \mathbf{K}_{sl} & \mathbf{K}_{ss} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}_l \\ \boldsymbol{\Phi}_s \end{bmatrix} = \begin{bmatrix} \mathbf{0}_l \\ \mathbf{0}_s \end{bmatrix}. \tag{2.17}$$

As $\mathbf{K}_{ll}$ is symmetric and positive definite, the degrees of freedom of the rigid body modes corresponding to the l set can be expressed in terms

of the degrees of freedom of the rigid body modes corresponding to the reference set, namely

$$\mathbf{\Phi}_l = -\mathbf{K}_{ll}^{-1}\mathbf{K}_{ls}\mathbf{\Phi}_s. \tag{2.18}$$

A natural assumption consists of choosing $\mathbf{\Phi}_s = \mathbf{I}_{n_r}$, where $\mathbf{I}_{n_r}$ denotes the identity matrix of dimension $n_r \times n_r$, so that each column of the matrix $\mathbf{\Phi}_s$ represents a unit translation or rotation in the direction of the corresponding reference degrees of freedom. With all these premises, the relative elastic displacement $\mathbf{U}_e$ in Eq. (2.12) is computed. It is worth noting that, in the IR framework, displacements are measured relative to the moving reference set of degrees of freedom s, which is subjected to a constant acceleration and undergoes infinite displacements. As a consequence, the rigid body displacement $\mathbf{U}_r$ is not of interest and can be eliminated from the solving equation. Finally, the system to be solved to compute the relative elastic displacement, which in the remainder is simply referred to as $\mathbf{U}$, reads

$$\begin{bmatrix} \mathbf{K}_{ll} & \mathbf{K}_{ls} \\ \mathbf{K}_{sl} & \mathbf{K}_{ss} \end{bmatrix} \begin{Bmatrix} \mathbf{U}_l \\ \mathbf{U}_s \end{Bmatrix} = \begin{Bmatrix} \mathbf{F}_l \\ \mathbf{F}_s \end{Bmatrix} - \begin{bmatrix} \mathbf{M}_{ll} & \mathbf{M}_{ls} \\ \mathbf{M}_{sl} & \mathbf{M}_{ss} \end{bmatrix} \mathbf{\Phi}\boldsymbol{\alpha}. \tag{2.19}$$

Imposing a zero displacement in the degrees of freedom of the s set, $\mathbf{U}_s = \mathbf{0}$, ensures that the following system is solvable and provides the required relative elastic displacement at the degrees of freedom of the l set,

$$\mathbf{K}_{ll}\mathbf{U}_l = \mathbf{F}_l - \mathbf{M}_l\mathbf{\Phi}\boldsymbol{\alpha}, \tag{2.20}$$

where $\mathbf{M}_l = [\mathbf{M}_{ll}\ \mathbf{M}_{ls}]$. The IR method can be summarised in three steps as depicted in Fig. 2.1.

2.3 The parametric inertia relief method

2.3.1 Problem definition

The IR problem is now extended to the case of an unconstrained structure characterised by parametric properties. Let us introduce a set of n_p material or geometric parameters denoted by $\boldsymbol{\mu} = [\mu_1, \mu_2, \ldots, \mu_{n_p}]^T \in \boldsymbol{\mathcal{M}} \subset \mathbb{R}^{n_p}$. The set of parametric domains $\boldsymbol{\mathcal{M}}$ is defined as the Cartesian product

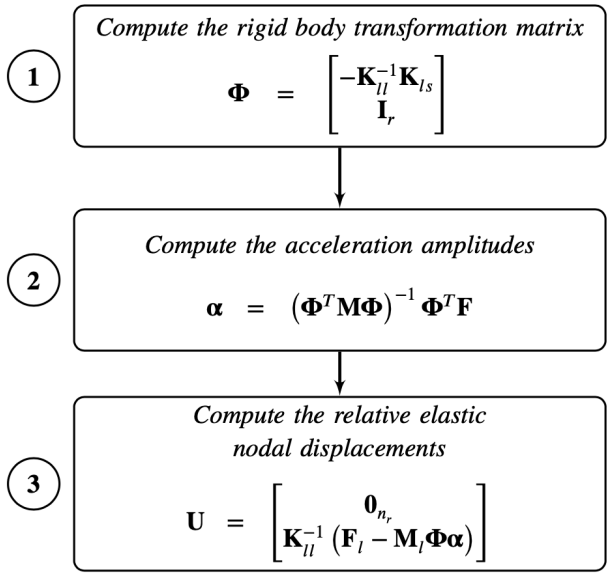


Figure 2.1: Schematic representation of the steps performed in the IR method.

of a predefined interval for each one of the parameters, namely $\boldsymbol{\mathcal{M}} := \mathcal{M}_1 \times \mathcal{M}_2 \times \cdots \times \mathcal{M}_{n_p}$, with $\mu_j \in \mathcal{M}_j$ for $j = 1, \ldots, n_p$. The semi-discrete system of Eq. (2.8) for a parametric problem can be written as

$$\mathbf{M}(\boldsymbol{\mu})\,\ddot{\mathbf{U}}(\boldsymbol{\mu}) + \mathbf{K}(\boldsymbol{\mu})\,\mathbf{U}(\boldsymbol{\mu}) = \mathbf{F}(\boldsymbol{\mu}). \tag{2.21}$$

In order to solve Eq. (2.21) with the IR method, three parametric steps have to be performed, following the rationale of the IR method described in Section 2.2 for the non-parametric case. The first step consists of computing the rigid body modes as

$$\boldsymbol{\Phi}(\boldsymbol{\mu}) = \begin{bmatrix} -\mathbf{K}_{ll}^{-1}(\boldsymbol{\mu})\,\mathbf{K}_{ls}(\boldsymbol{\mu}) \\ \mathbf{I}_r \end{bmatrix}. \tag{2.22}$$

Second, the vector of accelerations is given by

$$\boldsymbol{\alpha}(\boldsymbol{\mu}) = \left[\boldsymbol{\Phi}^T(\boldsymbol{\mu})\,\mathbf{M}(\boldsymbol{\mu})\,\boldsymbol{\Phi}(\boldsymbol{\mu})\right]^{-1}\,\boldsymbol{\Phi}^T(\boldsymbol{\mu})\,\mathbf{F}(\boldsymbol{\mu}). \tag{2.23}$$

Finally, the relative elastic displacement is computed as

$$\mathbf{U}(\boldsymbol{\mu}) = \begin{bmatrix} \mathbf{0}_{n_r} \\ \mathbf{K}_{ll}^{-1}(\boldsymbol{\mu}) \left(\mathbf{F}_l(\boldsymbol{\mu}) - \mathbf{M}_l(\boldsymbol{\mu}) \, \boldsymbol{\Phi}(\boldsymbol{\mu}) \, \boldsymbol{\alpha}(\boldsymbol{\mu}) \right) \end{bmatrix}. \tag{2.24}$$

In Eqs. (2.22) to (2.24), $\boldsymbol{\mu}$ is treated as a set of additional independent variables (or parametric coordinates), instead of problem parameters. As a consequence, the generalised solution of the three equations depends explicitly on the parameters and takes values in the multidimensional domain $\mathcal{D} = \Omega \times \boldsymbol{\mathcal{M}}$. Standard numerical methods (e.g. finite elements, finite volumes, finite differences) would require the solution of each step of the IR method in the high dimensional space $\mathcal{D}$, which is not feasible in practical problems. As already announced, the PGD is proposed as a ROM able to circumvent the *curse of dimensionality* and to provide the generalised solution of the parametric IR problem.

2.3.2 Cascade application of the encapsulated PGD approach

The goal of this section is to solve the parametric IR problem by means of the PGD technique. Following the standard PGD rationale, let us assume that the solution $\mathbf{U}(\boldsymbol{\mu})$ of Eq. (2.21) can be approximated by a linear combination of an *a priori* unknown number $N_{\mathbf{U}}$ of terms, namely

$$\mathbf{U}(\boldsymbol{\mu}) \approx \mathbf{U}^{\texttt{PGD}}(\boldsymbol{\mu}) = \sum_{i=1}^{N_{\mathbf{U}}} \beta_{\mathbf{U}}^{i} \, \mathbf{U}^{i} \prod_{j=1}^{n_p} F_{\mathbf{U}}^{j,i}(\mu_j). \tag{2.25}$$

Each term of the PGD sum i is given by the product of a spatial quantity, $\mathbf{U}^i$, defined on the discretised space Ω and a set of parametric functions $F_{\mathbf{U}}^{j,i}(\mu_j)$ depending, in a separated form, on each parameter μ_j, for $j = 1, 2, \ldots, n_p$. The spatial term, $\mathbf{U}^i$, is a vector of the size of the finite element displacement vector, whereas each parametric dimension μ_j is discretised with m_j points with coordinates $\mu_j^{p_j}$, where $p_j = 1, 2, \ldots, m_j$. The spatial and parametric quantities are usually normalised and the amplitude of each term, $\beta_{\mathbf{U}}^{i}$, indicates the relevance of the ith term to the final separated

solution.

The solver scheme behind the PGD method consists of two main steps, as outlined in Fig. 2.2. In order to compute the terms of the summation in

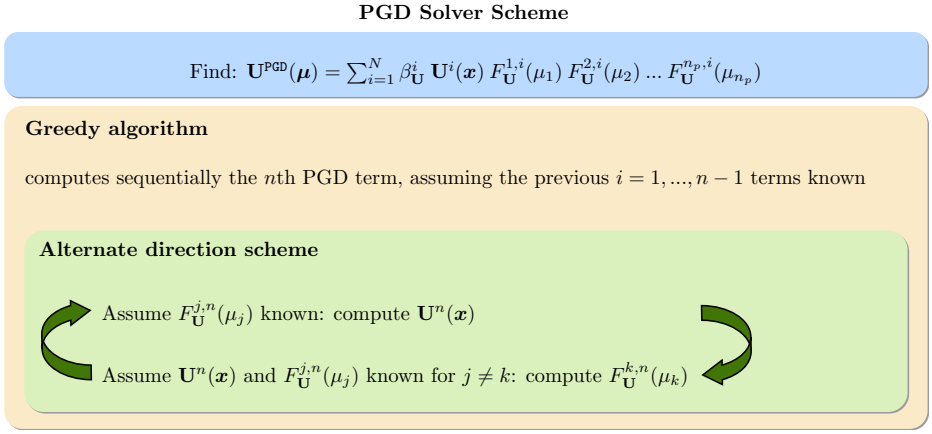


Figure 2.2: Description of the PGD solver scheme.

Eq. (2.25), a greedy approach is typically employed. Assuming that the previous $n-1$ terms are known, the greedy algorithm computes sequentially the nth term

$$\mathbf{U}^{\mathtt{PGD},n}(\boldsymbol{\mu}) = \mathbf{U}^{\mathtt{PGD},n-1}(\boldsymbol{\mu}) + \mathbf{U}^{n} \prod_{j=1}^{n_p} F_{\mathbf{U}}^{j,n}(\mu_j) \tag{2.26}$$

given by the spatial term $\mathbf{U}^n$ and the parametric terms $F_{\mathbf{U}}^{j,n}(\mu_j)$ for $j = 1, 2, \ldots, n_p$. The enrichment process automatically stops when a user-defined level of accuracy is reached, that is when the amplitude $\beta_{\mathbf{U}}^{n}$ of the last term is smaller than a user defined tolerance.

Since the unknown spatial and parametric terms $\mathbf{U}^n$ and $F_{\mathbf{U}}^{j,n}(\mu_j)$ are multiplying, the problem of computing the nth term in Eq. (2.26) is nonlinear. More precisely, it is a nonlinear least-squares problem defined to find the best rank-one approximation (meant as the product of sectional functions) of the unknown term $\mathbf{U}^n \prod_{j=1}^{n_p} F_{\mathbf{U}}^{j,n}(\mu_j)$. Commonly, the alternate direction scheme is employed as the simplest iterative solver available, which

proves to be surprisingly robust in the PGD context. The scheme consists in solving the problem separately for each unknown function, assuming that all the others are known, until a stationary solution is reached. It is worth emphasizing that despite a nonlinear problem needs to be solved to obtain each PGD term, the computational cost of the problem increases linearly with the number of introduced parameters, making the solution of high-dimensional problems affordable.

Recently, Díez at al. [73] developed the *encapsulated PGD toolbox*, which is a collection of PGD-based algorithms able to perform algebraic operations (e.g. product, division, storage, compression, solving linear system of equations, etc.) for multidimensional data represented in a discretised tensorial separated format. The main advantage of the library (freely available at `https://git.lacan.upc.edu/zlotnik/algebraicPGDtools.git`) is that each routine is *encapsulated*, meaning that it can be used as a black box. This is particularly attractive for the end user and it facilitates the interaction with commercial software.

To illustrate the idea behind the encapsulated PGD toolbox, Fig. 2.3 describes the structure of the encapsulated-PGD routine that solves parametric linear systems of equations. In a straightforward way, the same structure can be extended to other arithmetic operators. Shortly, given an algebraic linear system of equations $\mathbf{A}(\boldsymbol{\mu})\,\mathbf{x}(\boldsymbol{\mu}) = \mathbf{b}(\boldsymbol{\mu})$ depending on the set of parameters $\boldsymbol{\mu}$, the toolbox is able to return an explicit description of $\mathbf{x}(\boldsymbol{\mu})$, also called *computational vademecum*, containing the solution for every possible combination of the parameters. The only requirement to employ the encapsulated PGD approach is to pre-process the input quantities, i.e. the parametric matrix $\mathbf{A}(\boldsymbol{\mu})$ and vector $\mathbf{b}(\boldsymbol{\mu})$, such that they are expressed in a PGD separated form. Given the input data, the user only needs to employ the encapsulated PGD in the offline stage, to obtain the PGD approximation by means of the above mentioned greedy algorithm and alternate direction scheme. The output consists of the sought *computational vademecum* and, during an online stage, the solution can

be evaluated in real-time for any set of parameters at a negligible computational cost. The PGD-IR approach proposed in this work makes use of

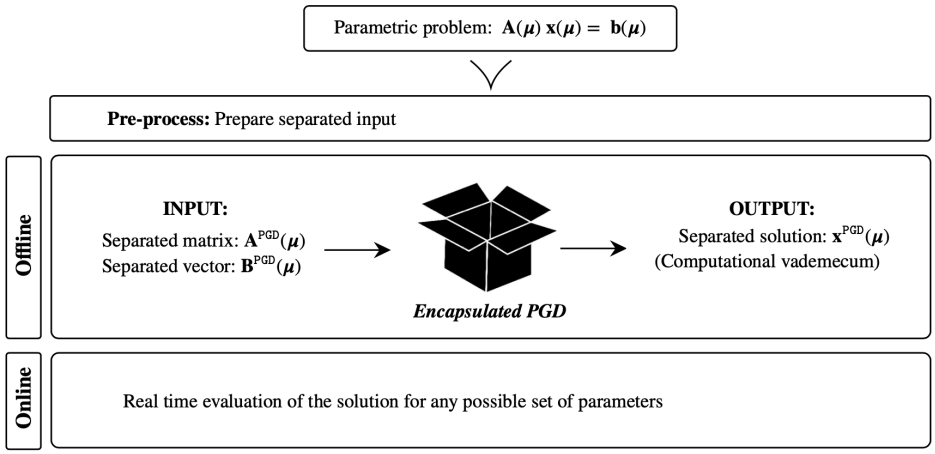


Figure 2.3: Structure of the *encapsulated-PGD* linear solver.

the encapsulated PGD toolbox. The three parametric Eqs. (2.22) to (2.24) are solved sequentially, in the sense that the solution of each equation is directly needed in the next one. Consequently, as depicted in Fig. 2.4, a *cascade* PGD scheme can be employed, in which the output of each step, obtained in a separated format by simply calling the encapsulated PGD linear solver, can be directly used as an input of the next one, until the final solution of the global problem is computed.

As already said, in order to use the toolbox, the user has to provide a separable representation of the input data. In particular, the stiffness $\mathbf{K}(\boldsymbol{\mu})$ and mass $\mathbf{M}(\boldsymbol{\mu})$ matrices must be written as

$$\mathbf{K}(\boldsymbol{\mu}) \approx \sum_{i=1}^{N_{\mathbf{K}}} \mathbf{K}^i \prod_{j=1}^{n_p} F_{\mathbf{K}}^{j,i}(\mu_j), \tag{2.27}$$

$$\mathbf{M}(\boldsymbol{\mu}) \approx \sum_{i=1}^{N_{\mathbf{M}}} \mathbf{M}^i \prod_{j=1}^{n_p} F_{\mathbf{M}}^{j,i}(\mu_j), \tag{2.28}$$

where the spatial terms are $\mathbf{K}^i \in \mathbb{R}^{n_d \times n_d}$ and $\mathbf{M}^i \in \mathbb{R}^{n_d \times n_d}$, whereas the parametric terms are $F_{\mathbf{K}}^{j,i}(\mu_j) \in \mathbb{R}^{n_j}$ and $F_{\mathbf{M}}^{j,i}(\mu_j) \in \mathbb{R}^{n_j}$, for $j =$

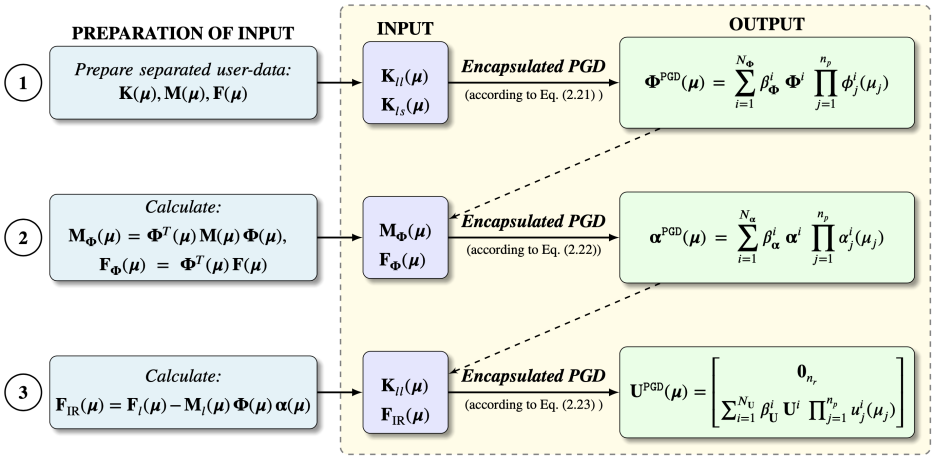


Figure 2.4: Schematic representation of the *cascade* encapsulated PGD approach for the solution of a parametric IR problem.

$1, 2, \ldots, n_p$. Similarly, the input nodal force vector $\mathbf{F}(\boldsymbol{\mu})$ must be written as

$$\mathbf{F}(\boldsymbol{\mu}) \approx \sum_{i=1}^{N_{\mathbf{F}}} \mathbf{F}^i \prod_{j=1}^{n_p} F_{\mathbf{F}}^{j,i}(\mu_j), \tag{2.29}$$

with $\mathbf{F}^i \in \mathbb{R}^{n_d}$ and $F_{\mathbf{F}}^{j,i}(\mu_j) \in \mathbb{R}^{n_j}$. In the above expressions $N_{\mathbf{K}}$, $N_{\mathbf{M}}$ and $N_{\mathbf{F}}$ are the number of terms required to produce a separable approximation of $\mathbf{K}(\boldsymbol{\mu})$, $\mathbf{M}(\boldsymbol{\mu})$ and $\mathbf{F}(\boldsymbol{\mu})$ respectively.
Finally, in order to solve the parametric equations of the PGD-IR approach, steps 2 and 3 depicted in Fig. 2.4 require some extra operations between the parametric objects, such as products, additions or compression. These operations can be easily performed by the *encapsulated PGD toolbox*.

It is important to underline that it is not always trivial to find a separated representation of the input data, as given by equations (2.27)−(2.29), especially when geometric parameters are considered in the problem. This issue will be addressed in the next section.

Remark 1. *The first two steps of the PGD-IR shown in Fig. 2.4 are*

parametric problems only if geometric parameters are considered, because, by definition, the rigid body modes of a structure do not depend on the material properties.

2.3.3 Geometric parameters: a nonintrusive algebraic approach to separate input quantities

The extension of the proposed nonintrusive PGD framework to geometrically parametrised problems represents a challenging task. This is due to the fact that, if geometric parameters are introduced in the problem, it is not trivial to find separable representation of the input quantities.
If a closed form separated expression of the stiffness and mass matrices is sought, the weak form of the problem must be modified to account for the parametric geometry. A common approach consists of formulating the problem in a reference domain, leading to several limitations that are briefly discussed in Appendix A. The most important limitation in the context of the current work is that the implementation based on a reference domain requires access to the code, precluding its application in an industrial framework, where commercial codes are typically employed.

In this section a nonintrusive algebraic approach is proposed, which is able to deal with general geometric parametrisations. The main idea is to perform a sampling of the parametric matrices and to express them in a separated format. The approach requires the computation of the parametric matrices for different values of the geometric parameters, whilst maintaining the connectivity matrix of the FE mesh. To this end, a mesh morphing approach is adopted in this work. Every time a sampling of the parametric matrices is required, the initial (reference) mesh is deformed according to the geometric parameters and the global stiffness and mass matrices are computed. The main advantage is that this approach can be easily integrated in commercial packages that are equipped with a mesh morphing capability. Alternatively, the user can define the preferred mesh morphing approach and produce a set of meshes to be imported in the preferred FE software. It is also worth mentioning that the sampling does not require the solution of the FE system of equations, as only the global stiffness

and mass matrices are of interest for the proposed PGD-IR approach. For this reason the method can be defined as a *hybrid* version of the PGD. On one hand, it can still be considered an a priori approach, as it does not require any previous full-order computation of the solution. On the other hand, a pre-process step is added, where the input data are sampled (without solving the problem) in the parametric domain of interest and then expressed in the required separated format. Once the set of global stiffness and mass matrices is available, they are expressed in a separated format using the encapsulated PGD toolbox.

To illustrate the proposed nonintrusive approach, let us consider the stiffness matrix $\mathbf{K} \in \mathbb{R}^{n_d \times n_d}$, depending on n_p parameters $\boldsymbol{\mu} = [\mu_1, \mu_2, \ldots, \mu_{n_p}]^T \in \boldsymbol{\mathcal{M}} \subset \mathbb{R}^{n_p}$. The parametric dimension $\mu_j \in \mathcal{M}_j$, for $j = 1, 2, \ldots, n_p$, is discretised using m_j points with coordinates $\mu_j^{p_j}$, where $p_j = 1, 2, \ldots, m_j$. The full-order sampling of the parametric matrix consists of evaluating $\mathbf{K}(\boldsymbol{\mu})$ in the set of m_{tot} points used to discretise the parametric domain $\boldsymbol{\mathcal{M}} = \mathcal{M}_1 \times \mathcal{M}_2 \times \cdots \times \mathcal{M}_{n_p}$, where $m_{\text{tot}} = \prod_{j=1}^{n_p} m_j$. Each point is characterised by its sectional indices $(p_1, p_2, \ldots, p_{n_p})$, which are duly sorted by using a linear array index i such that

$$i = p_1 + (p_2 - 1)m_2 + (p_3 - 1)m_2 \times m_3 + \cdots = p_1 + \sum_{j=2}^{n_p} (p_j - 1) \prod_{l=2}^{j} m_l. \quad (2.30)$$

Note that the association between the multi index $(p_1, p_2, \ldots, p_{n_p})$ and the index i is also obtained by updating $i = i + 1$ inside n_p nested loops, with no need to use explicitly expression (2.30). Employing the association between the multi-index $(p_1, p_2, \ldots, p_{n_p})$ and the linear index i, the parametric stiffness matrix $\mathbf{K}(\boldsymbol{\mu})$ can be written as

$$\mathbf{K}(\boldsymbol{\mu}) = \sum_{p_1=1}^{m_1} \sum_{p_2=1}^{m_2} \cdots \sum_{p_{n_p}=1}^{m_{n_p}} \mathbf{K}(\mu_1^{p_1}, \mu_2^{p_2}, \ldots, \mu_{n_p}^{p_{n_p}}) F_{p_1, p_2, \ldots, p_{n_p}}(\mu_1, \mu_2, \ldots, \mu_{n_p}), \quad (2.31)$$

where $F_{p_1, p_2, \ldots, p_{n_p}}$ is such that $F_{p_1, p_2, \ldots, p_{n_p}}(\mu_1^{p_1}, \mu_2^{p_2}, \ldots, \mu_{n_p}^{p_{n_p}}) = 1$ and it is equal to zero for any other values of the discrete indices p_j. Using the

linear indexing i introduced in Eq. (2.30), Eq. (2.31) becomes

$$\mathbf{K}(\boldsymbol{\mu}) = \sum_{i=1}^{m_{\text{tot}}} \mathbf{K}^i \prod_{j=1}^{n_p} F_{\mathbf{K}}^{j,i}(\mu_j), \tag{2.32}$$

where $\mathbf{K}^i = \mathbf{K}(\mu_1^{p_1}, \mu_2^{p_2}, \ldots, \mu_{n_p}^{p_{n_p}})$, and $F_{\mathbf{K}}^{j,i}(\mu_j^{p_l}) = \delta_{p_l,p_j}$ for any $p_l = 1, 2, \ldots, m_j$, while p_j is given by i as defined in Eq. (2.30). Finally, Eq. (2.32) represents the desired separated representation of the stiffness matrix.

Depending on the number of parameters, n_p, and the number of nodes chosen to discretise the parametric domains, m_j, the separated expression of the parametric stiffness matrix might involve a large number of terms, m_{tot}. It is possible to reduce the computational cost of the following calculations by employing the PGD-compression, available in the encapsulated PGD-toolbox [74]. The idea is to perform an $\mathcal{L}_2$-projection of the expression of Eq. (2.32) to reduce the number of terms in the summation while maintaining an accurate representation of $\mathbf{K}(\boldsymbol{\mu})$. In a similar fashion, a separated representation of the mass matrix can be also obtained. As it will be shown by means of numerical examples, the main advantage of the proposed algebraic technique is its flexibility which in general allows to add an arbitrary number of geometric parameters as variables of the problem. In addition, the nonintrusive character of the proposed ROM, makes the approach proposed in this work suitable for industrial applications.

2.4 Numerical examples: parametric inertia relief with material and geometric parameters

In this section a numerical example is presented in order to illustrate the numerical properties of the proposed PGD-IR method when both material and geometric parameters are considered.

A pure torsion test case is considered for an unconstrained linear elastic 3D structure characterised by one material and one geometric parameter, that are treated as additional coordinates of the problem. For a better readability, the two variables are denoted here with different symbols,

that is $\mu \in \mathcal{M}_\mu$ and $\theta \in \mathcal{M}_\theta$ for the material and geometric parameters respectively.

As depicted in Fig. 2.5, the reference domain $\hat{\Omega}$ consists of a block with dimensions $[-L_x/2, L_x/2] \times [-L_y/2, L_y/2] \times [-L_z/2, L_z/2]$ with an inclusion given by $[-L_x/6, L_x/6] \times [-L_y/4, L_y/4] \times [-L_z/2, L_z/2]$, where $L_x = 6$, $L_y = 12$ and $L_z = 1$. The torsional load is given by two parallel forces of constant magnitude $F = 10$ acting on the positive and negative z direction respectively and applied at the points $P = (2, 4, 1/2)$ and $Q = (-2, 4, 1/2)$. Fig. 2.5 also shows the spatial discretisation employed, consisting on a regular mesh with 236 nodes and 742 linear tetrahedral elements.

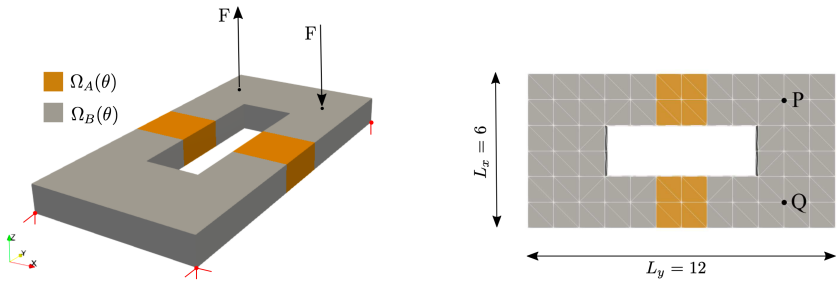


Figure 2.5: Computational domain, showing the partition into two non-overlapping subdomains $\Omega_A(\theta)$ and $\Omega_B(\theta)$ (left) and top view of the discretised computational domain showing the dimensions and the points P and Q where the forces are applied (right).

The physical domain $\Omega(\theta)$ depends upon the geometric parameter and it is split into two non-overlapping subdomains $\Omega_A(\theta)$ and $\Omega_B(\theta)$. The parametric Young's modulus E is defined as

$$E(\boldsymbol{x}, \mu) = \begin{cases} E_A(\mu) = \mu & \text{for} \quad \boldsymbol{x} \in \Omega_A(\theta), \\ E_B = 200 & \text{for} \quad \boldsymbol{x} \in \Omega_B(\theta), \end{cases} \tag{2.33}$$

where the Young modulus $E_A(\mu)$ is considered varying in the range $\mathcal{M}_\mu = [10, 410]$, and $\mathcal{M}_\mu$ is discretised with a uniform distribution of $n_\mu = 41$ points. The Poisson's ratio and the density are assumed constant in the

whole domain and taken as $\nu = 0.3$ and $\rho = 1$ respectively.
The geometrically parametrised domain $\Omega(\theta)$ is described with the Cartesian coordinates $\boldsymbol{x}$, and it is defined as the image of the reference domain $\hat{\Omega}$, with reference coordinates $\hat{\boldsymbol{x}}$, via a geometric mapping $\boldsymbol{\Psi}(\hat{\boldsymbol{x}}, \theta)$, namely

$$\begin{cases} x &= \psi_1(\hat{\boldsymbol{x}}, \theta) = \hat{x} + \theta \sin\left(\dfrac{\pi \hat{y}}{L_y}\right)\left(\hat{x} - \dfrac{L_x}{2}\right), \\ y &= \psi_2(\hat{\boldsymbol{x}}, \theta) = \hat{y}, \\ z &= \psi_3(\hat{\boldsymbol{x}}, \theta) = \hat{z}. \end{cases} \tag{2.34}$$

The parameter θ is taken to be in the interval $\mathcal{M}_\theta = [0, 0.5]$, and $\mathcal{M}_\theta$ is discretised with a uniform distribution of $n_\theta = 21$ points.

Fig. 2.6 shows the influence of the parameter θ in the geometry of the computational domain for three different values of θ. The particular value $\theta = 0$ leads to a deformed configuration that coincides with the reference configuration, i.e. the mapping of Eq. (2.34) becomes the identity. The configurations in Fig. 2.6 also show that the mapping changes the nodal coordinates of the mesh while maintaining the connectivities, as required within the current PGD framework.

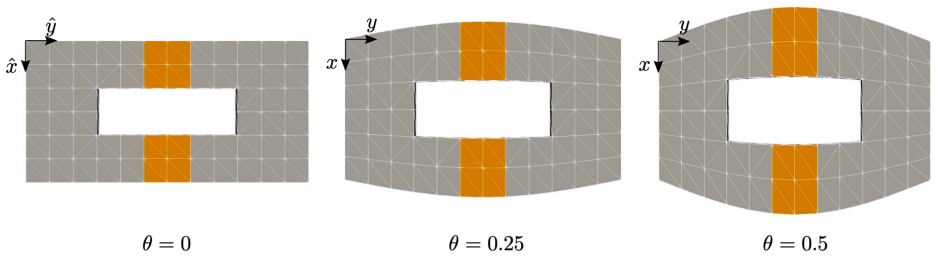


Figure 2.6: Physical domain for three different values of the geometric parameter θ.

The objective of this numerical test is to employ the proposed PGD-IR approach to obtain a *computational vademecum* able to describe the variation of the solution with respect to the material and geometric parameters.

Following the proposed PGD-IR framework, the first step consists of choosing a reference set of six degrees of freedom able to counteract the rigid body motions of the structure. Next, in order to employ the encapsulated PGD toolbox, it is necessary to define the input data (i.e. stiffness matrix, mass matrix, force vector) in a separated format. By using the linear dependence of the stiffness matrix on the Young's modulus, an analytical separable representation of the stiffness matrix with respect to μ can be easily constructed. For the geometric parameter θ, the algebraic PGD toolbox is employed, as discussed in detail in Section 2.3.3. For every nodal value of the geometric parameter $\theta^p = [\theta^1, \theta^2, \dots, \theta^{n_\theta}]^T$, the geometrically deformed mesh is generated according to the mapping of Eq. (2.34), and two stiffness-like matrices $\mathbf{K}_A(\theta^p)$ and $\mathbf{K}_B(\theta^p)$ are computed. The quantity $\mathbf{K}_A(\theta^p)$ is calculated by imposing the Young's modulus $(E_A, E_B) = (1, 0)$, thus accounting for the contribution of the finite elements belonging to the subdomain $\Omega_A(\theta^p)$ to the global stiffness matrix. Analogously, $\mathbf{K}_B(\theta^p)$ corresponds to the choice $(E_A, E_B) = (0, 1)$ and accounts for the contribution of the finite elements belonging to the subdomain $\Omega_B(\theta^p)$. Once these matrices are sampled in the parametric nodes n_θ, a separated form of the parametric global stiffness matrix is readily available, namely

$$\mathbf{K}(\mu, \theta) = E_A(\mu) \sum_{i=1}^{n_\theta} \mathbf{K}_A^i \, G_{\mathbf{K}}^i(\theta^p) + E_B \sum_{i=1}^{n_\theta} \mathbf{K}_B^i \, G_{\mathbf{K}}^i(\theta^p), \tag{2.35}$$

with $G_{\mathbf{K}}^i(\theta^p) = \delta_{p,i}$, for every $p = 1, 2, \dots, n_\theta$. Note that, from now on, the material parametric functions will be denoted by $M_\square(\mu)$, while $G_\square(\theta)$ will be used for the geometric functions, being $\square$ the quantity to which they refer.

In this example, a PGD-compression was performed, which is always advisable when the number of PGD-terms is large, and an accurate approximation of the stiffness matrix was obtained in the known PGD format

$$\mathbf{K}^{\mathtt{PGD}}(\mu, \theta) = \sum_{i=1}^{N_{\mathbf{K}}} \mathbf{K}^i \, M_{\mathbf{K}}^i(\mu) \, G_{\mathbf{K}}^i(\theta). \tag{2.36}$$

In this case, after performing compression with a tolerance $tol = 10^{-5}$,

the number of PGD terms was reduced to $N_{\mathbf{K}} = 10$. Following the same procedure, the PGD approximation of the parametric mass matrix is also obtained, namely

$$\mathbf{M}^{\texttt{PGD}}(\mu, \theta) = \sum_{i=1}^{N_{\mathbf{M}}} \mathbf{M}^i \, M^i_{\mathbf{M}}(\mu) \, G^i_{\mathbf{M}}(\theta). \tag{2.37}$$

Please note that the mass matrix is actually independent on the Young modulus, that is $M^i_{\mathbf{M}}(\mu) = 1$. However, the general expression of Eq. (2.37) is used to maintain a consistent notation for all the inputs of the PGD-IR approach. Finally, the global forcing vector is also written in the general separated form

$$\mathbf{F}^{\texttt{PGD}} = \mathbf{F} \, M_{\mathbf{F}}(\mu) \, G_{\mathbf{F}}(\theta), \tag{2.38}$$

where, again, it is worth emphasizing that $\mathbf{F}$ is the standard FE forcing vector and $M_{\mathbf{F}}(\mu) = G_{\mathbf{F}}(\theta) = 1$, because the right hand side is not dependent on the material parameter and, for the given set of forces applied to the structure is also independent on the geometric parameter.
The computation of the separated form of the stiffness and mass matrices and the forcing vector completes the pre-process required to apply the proposed PGD-IR approach. Next, the three steps of the PGD-IR approach can be sequentially completed. As detailed in Remark 1, the three steps involve a parametric problem because not only material parameters are considered but also geometric parameters. The final generalised solution can be written as

$$\mathbf{U}^{\texttt{PGD}}(\mu, \theta) = \sum_{i=1}^{N_{\mathbf{U}}} \beta_{\mathbf{U}} \, \mathbf{U}^i \, M^i_{\mathbf{U}}(\mu) \, G^i_{\mathbf{U}}(\theta). \tag{2.39}$$

It is worthy to mention that the proposed PGD-IR approach was implemented in a Matlab routine which acts as a black-box, following the philosophy of the encapsulated PGD toolbox. In fact, the routine only requires to receive the input quantities in a separated form in order to return the output in the same separated form.

Fig. 2.7 plots the evolution of the amplitude $\beta_{\mathbf{U}}$ of each PGD term. It can be observed that the amplitude rapidly decreases as the number of terms is increased. With 15 computed terms the amplitude of the last term is almost four orders of magnitude lower than the amplitude of the first one. In addition, the results show that the first four PGD terms capture the most relevant information of the generalised solution as the fifth and subsequent terms have an amplitude at least two orders of magnitude lower than the amplitude of the first term.

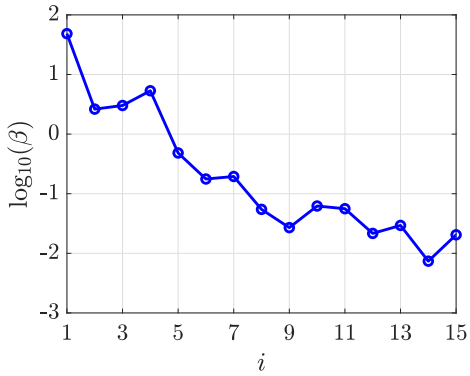
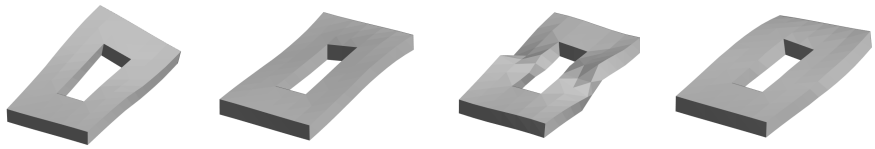


Figure 2.7: Evolution of the amplitude of the PGD terms β^i of the solution $\mathbf{U}^{\texttt{PGD}}(\mu, \theta)$ with respect to the number of PGD terms, i.

The first four normalised spatial terms are shown in Fig. 2.8, whereas the first four parametric functions are displayed in Fig. 2.9. The spatial terms provide an illustration of the deformation induced by the four most relevant modes of the generalised solution. The parametric functions corresponding to the material illustrate that the four terms have the maximum contribution to the generalised solution for $\mu = 10$. As the material property approaches the maximum value of $\mu = 410$, the third and fourth terms have less influence on the solution. Finally, the terms corresponding to the geometric parameter have a more global character, proving the extra difficulty in solving geometrically parametrised problems. In order to get a particularised solution for a chosen set of the parameters $(\bar{\mu}, \bar{\theta})$, the correspondent function values $M_{\mathbf{U}}(\bar{\mu})$ and $G_{\mathbf{U}}(\bar{\theta})$ are evaluated for each



(a) Term 1 **(b)** Term 2 **(c)** Term 3 **(d)** Term 4

Figure 2.8: First four spatial terms of the generalised solution $\mathbf{U}^{\mathtt{PGD}}(\mu, \theta)$.

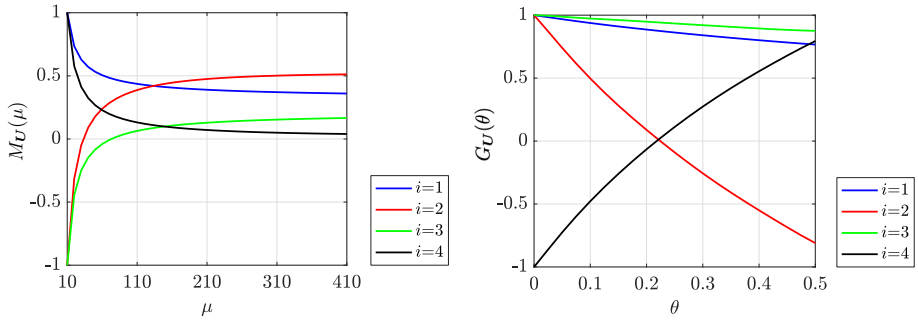


Figure 2.9: First four material $M_{\mathbf{U}}(\mu)$ and geometric $G_{\mathbf{U}}(\theta)$ terms of the generalised solution $\mathbf{U}^{\mathtt{PGD}}(\mu, \theta)$.

PGD-term i and then multiplied by the correspondent spatial term and amplitude. Fig. 2.10 shows the particularised solutions in terms of deformed configuration and equivalent von Mises stress field for nine specific sets of parameters. The dominant character of the first spatial term of Fig. 2.8 can be clearly observed, whereas the magnitude of the stress highly depends on the parametric choice. Please remember that these particularised solutions were obtained in real-time during an online post-process step.

In order to validate the PGD results, the accuracy with respect to the full-order FE computations is measured as the relative error between the PGD and FE solutions in the $\mathcal{L}_2(\Omega \times \mathcal{M}_\mu \times \mathcal{M}_\theta)$ norm, that is

$$\varepsilon_{\mathrm{PGD}} = \left(\frac{\int_{\mathcal{M}_\theta} \int_{\mathcal{M}_\mu} \int_{\Omega} \left(\mathbf{U}^{\mathtt{PGD}} - \mathbf{U}^{\mathtt{FE}}\right) \cdot \left(\mathbf{U}^{\mathtt{PGD}} - \mathbf{U}^{\mathtt{FE}}\right) d\Omega \, d\mu \, d\theta}{\int_{\mathcal{M}_\theta} \int_{\mathcal{M}_\mu} \int_{\Omega} \mathbf{U}^{\mathtt{FE}} \cdot \mathbf{U}^{\mathtt{FE}} \, d\Omega \, d\mu \, d\theta} \right)^{1/2} . \tag{2.40}$$

It is worth noting that to compute this error measure, the problem is solved

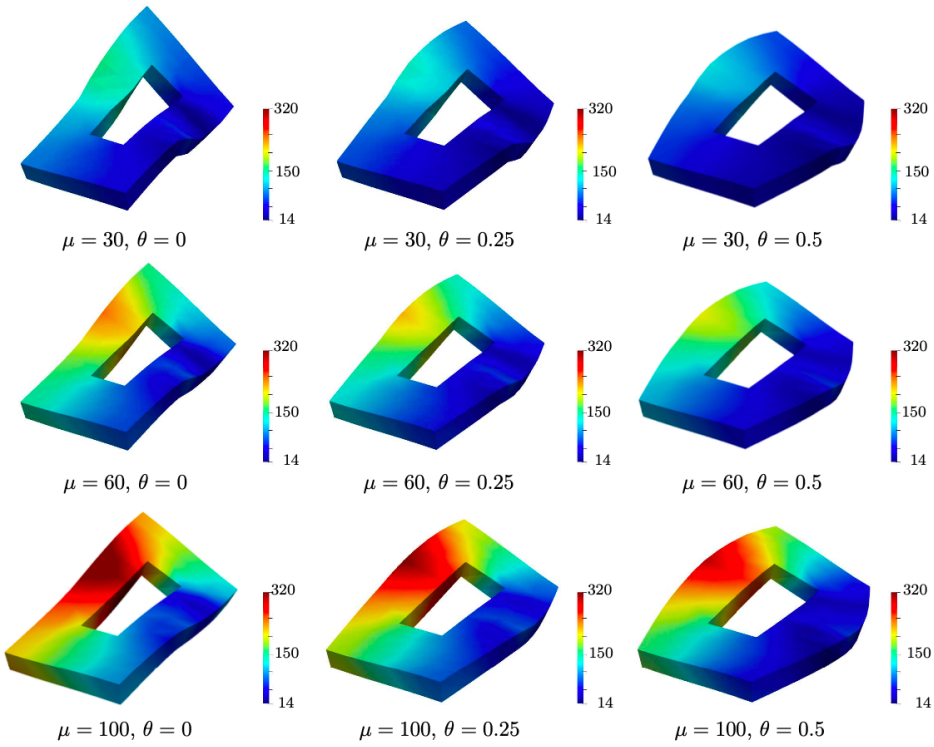


Figure 2.10: Particular cases of the generalised solution, showing the von Mises stress field, for nine choices of the parameters. The solutions are obtained in real-time after the PGD-IR is applied to compute the spatial and parametric terms.

by means of the standard FE method for each possible combination of the parameters, that is $n_\mu \times n_\theta = 21 \times 41 = 861$ FE simulations.
Fig. 2.11 shows the evolution of the relative error with respect to the number of PGD terms. As expected, the level of accuracy increases as the number of terms increases, up to a user-defined tolerance, which in this case was chosen equal to 10^{-3}. Note that the PGD solution converges to the desired tolerance with only nine PGD terms. An interesting advantage of the PGD method with respect to the standard FE method concerns the storage memory. In fact, the obtained PGD *computational vademecum* needs ~ 74 KB of storage memory versus the ~ 6650 KB needed to store all the 861 full-order FE solutions. Computational time is not particularly significant

PGD term	**i=1**	**i=2**	**i=3**	**i=4**	**i=5**	**i=6**	**i=7**	**i=8**	**i=9**
Iterations	12	27	17	14	14	18	24	17	18

Total n. of iterations = 161

Table 2.1: Total number of iterations performed by the alternate direction scheme to compute each PGD term.

in the PGD context. In fact, the main goal is to provide a method which is able to explore an arbitrary large parametric space with only one offline computation. Nevertheless, an interesting comparison is shown in Table 2.1 where the number of iterations needed by the alternate direction scheme for the computations of each PGD term is provided. As the cost of each iteration corresponds to the cost of a full-order FE simulation, the results in Table 2.1 show that the cost of the proposed PGD-IR is equivalent to 161 full-order solutions, compared to the 861 full-order computations required by the standard FE approach.

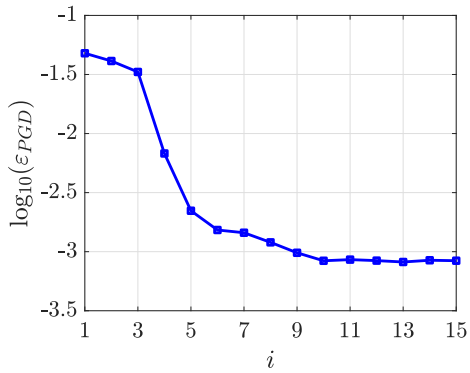


Figure 2.11: $\mathcal{L}_2(\Omega \times \mathcal{M}_\mu \times \mathcal{M}_\theta)$ norm of the difference between the PGD solution and the FE solution as a function of the number of PGD terms, i.

Finally, a major advantage of computing a PGD *computational vademecum* is the possibility to explore the design space and check, in real-time, the effects of the design parameters on a predefined quantity of interest (QoI). As an example, the relative displacement in the z direction, $\Delta \mathrm{U}_{\mathrm{PQ}}(z)$, of the points P and Q (see Fig. 2.5) is selected as QoI. The variation of the

chosen QoI in the parametric space is depicted in Fig. 2.12.

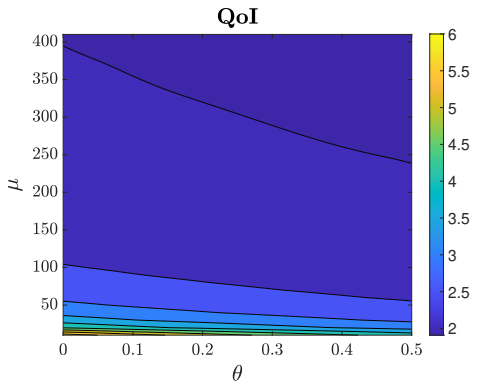


Figure 2.12: Variation of the QoI $\Delta U_{PQ}(z)$ with respect to the parameters μ and θ.

2.5 Chapter summary

A nonintrusive algebraic PGD approach combined with the IR method for the solution of unconstrained problems being characterised by material and geometric parameters has been presented. The developed solver makes use of the *encapsulated PGD toolbox* developed by Díez et al. [74], which enables to perform algebraic operations for multidimensional data and allows to solve sequentially the three parametric problems required by the IR method.

An algebraic *hybrid* PGD approach has been proposed to deal with geometric parameters by morphing a mesh generated for a reference configuration. The proposed method acts as a *black-box*, such that a nonintrusive interaction with commercial FE packages is possible.

One numerical example is used to underline the main properties of the method. The example considers an academic test case with one material and one geometric parameter. The ability to compute a *computational vademecum* is shown and the accuracy of the generalised solution is measured by comparing the PGD solution to a set of standard FE full-order solutions. It it shown that the proposed PGD-IR approach is able save

almost the 99% of storage memory, requiring only the 20% of computational time needed by the FE method to solve the problem for every possible set of parameters.

3 Parametric solutions in structural dynamics

The dynamic response of an automotive structure depends on its material and geometric properties. Changes in the design parameters can have considerable effects on the structural dynamic behaviour. For this reason, during the development process of a new system, designers have to run sensitivity analysis, such that potential problems can be identified and corrected before time, money and resources are wasted on prototyping and manufacturing a non-optimal design. The modal analysis is widely used in the automotive industry to predict the dynamic properties of a the vehicle body structure under free-vibration conditions. This consists of solving a generalised eigenvalue problem where the eigenvectors represent the natural deformation of the structure when vibrating (mode shapes), and the eigenvalues are the corresponding natural frequencies. Trying to efficiently identify how the mode shapes and natural frequencies depend upon the design parameters is still today a challenging computational task. In this work, the PGD approach has been coupled with an eigensolver technique, the inverse power method (IPM) [75], to efficiently identify how a variation of user-defined material and geometric parameters affects the dynamic response of the structure in terms of dominant eigenmodes and related natural frequencies. Similar works have been presented in the field of dynamic problems. For example, [76] and [77] proposed a POD-PGD approach to implement a real-time integration scheme able to solve the equations of solid dynamics depending on parametric initial conditions. In [78], [79] and [80], PGD is coupled with a hybrid integration scheme which combines modal and harmonic analysis to efficiently solve frequency-dependent parametric problems. Moreover, the PGD method

has been coupled with standard eigenvalue solvers, such as the Arnoldi technique [81], the IPM [82] and the direct power method [83], in order to solve single- and multi-group neutron diffusion eigenvalue problems.

The chapter is organised as follows. Section 3.1 reviews the formulation of the modal analysis in the standard non-parametric framework. In particular, the IPM is introduced as the eigensolver to be coupled with the PGD method. The description of the parametric modal analysis is given in Section 3.2. In Section 3.3, the modal analysis of the same 3D structure with one material and one geometric parameter used in the static case is considered to underline the main properties of the developed ROM. Finally, conclusions and outlooks are summarised in Section 3.4.

The content of this chapter is based on the following publication:

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: Nonintrusive parametric solutions in structural dynamics. In: *Computer Methods in Applied Mechanics and Engineering* 389 (2022), p. 114336. ISSN: 0045-7825.

3.1 Problem statement: finite element discretisation and modal analysis

In the absence of volumetric forces, the unknown displacement field $\boldsymbol{u}(\boldsymbol{x}, t)$ is to be computed for $\boldsymbol{x} \in \Omega$ and $t \in (0, T]$ such that

$$\rho\, \ddot{\boldsymbol{u}} = \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(\boldsymbol{u}), \tag{3.1}$$

where ρ stands for the density, $\ddot{\boldsymbol{u}} := \frac{\partial^2 u}{\partial t^2}$ is the acceleration, and $\boldsymbol{\sigma}$ denotes the Cauchy stress tensor. The differential equation (3.1) is to be complemented with boundary conditions and initial conditions. Typically, the boundary of the domain Ω, $\partial\Omega$, is partitioned into the disjoint parts Γ_D and Γ_N, where Dirichlet and Neumann boundary conditions are prescribed. Under unconstrained conditions, as assumed in this work, Γ_N coincides with

$\partial\Omega$ and the Neumann boundary conditions are homogeneous everywhere. The Cauchy stress tensor dependence on $\boldsymbol{u}$ is given by the constitutive law (here, the generalised Hooke's law associated with the fourth order stiffness tensor $\mathbf{C}$) and the definition of the linear strain tensor, namely

$$\boldsymbol{\sigma}(\boldsymbol{u}) = \mathbf{C} : \boldsymbol{\varepsilon}(\boldsymbol{u}), \text{ and } \boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{1}{2}\left(\boldsymbol{\nabla}\boldsymbol{u} + \boldsymbol{\nabla}\boldsymbol{u}^{\top}\right). \tag{3.2}$$

Using weighted residuals in space (integrating in Ω), problem (3.1) is re-written in the weak form

$$\underbrace{\int_{\Omega} \rho\, \boldsymbol{v}(\boldsymbol{x}) \cdot \ddot{\boldsymbol{u}}(\boldsymbol{x}, t)\, d\Omega}_{m(\ddot{\boldsymbol{u}}, \boldsymbol{v})} + \underbrace{\int_{\Omega} \boldsymbol{\varepsilon}(\boldsymbol{v}(\boldsymbol{x})) : \boldsymbol{\sigma}(\boldsymbol{u}(\boldsymbol{x}, t))\, d\Omega}_{a(\boldsymbol{u}, \boldsymbol{v})} = 0 \tag{3.3}$$

for every $\boldsymbol{v}(\boldsymbol{x})$ taking values in Ω, and at very time t. The discretised versions of the bilinear forms $m(\cdot,\cdot)$ and $a(\cdot,\cdot)$ in a finite element space are the mass matrix $\mathbf{M}$ and the stiffness matrix $\mathbf{K}$, respectively. Similarly, the discrete version of the unknown $\boldsymbol{u}(\boldsymbol{x}, t)$ is the vector of time-dependent nodal values $\mathbf{U}(t) \in \mathbb{R}^{n_d}$. Thus, the semi-discrete counterpart of Eq. (3.3) is the following linear system of ordinary differential equations

$$\mathbf{M}\ddot{\mathbf{U}}(t) + \mathbf{K}\mathbf{U}(t) = \mathbf{0}. \tag{3.4}$$

3.1.1 Modal analysis

The modal analysis solution of Eq. (3.4) assumes that the time dependence of $\mathbf{U}(t)$ is harmonic, that is

$$\mathbf{U}(t) = \cos(\omega t - \psi)\boldsymbol{\phi}, \tag{3.5}$$

where ω is the *angular velocity* and ψ the phase angle. Note that the frequency of the mode is $\frac{\omega}{2\pi}$. Hence

$$\ddot{\mathbf{U}}(t) = -\omega^2 \cos(\omega t - \alpha)\boldsymbol{\phi} = -\omega^2 \mathbf{U}(t) \tag{3.6}$$

and therefore, replacing Eq. (3.6) in Eq. (3.4), it turns out that

$$\mathbf{K}\boldsymbol{\phi} = \omega^2 \mathbf{M}\boldsymbol{\phi}. \tag{3.7}$$

Expression (3.7) is an eigenvalue problem. Since $\mathbf{M}$ is symmetric positive-definite and $\mathbf{K}$ is symmetric semi-positive, the solution of Eq. (3.7) provides eigenvalues ω_i^2 and their corresponding eigenvectors $\boldsymbol{\phi}_i$, for $i = 1, 2, \ldots, n_d$ (it is assumed they are sorted such that $\omega_1 \leq \omega_2 \leq \cdots \leq \omega_{n_d}$).
Eigenmodes $\boldsymbol{\phi}_i$, for $i = 1, 2, \ldots, n_d$, associated with eigenfrequencies ω_i, are the natural mode shapes, representing the deformation of the structure as vibrating in its ith mode. Modes associated with different frequencies are *orthogonal* with respect to both $\mathbf{M}$ and $\mathbf{K}$. That is $\boldsymbol{\phi}_i^\top \mathbf{M}\boldsymbol{\phi}_j = 0$ and $\boldsymbol{\phi}_i^\top \mathbf{K}\boldsymbol{\phi}_j = 0$ for i and j such that $\omega_i \neq \omega_j$.
Note that once the eigenmode $\boldsymbol{\phi}_i$ is available, the corresponding eigenfrequency is readily computed invoking the Rayleigh quotient:

$$\omega_i^2 = \frac{\boldsymbol{\phi}_i^\top \mathbf{K}\boldsymbol{\phi}_i}{\boldsymbol{\phi}_i^\top \mathbf{M}\boldsymbol{\phi}_i}. \tag{3.8}$$

In order to enforce unicity (up to their sign) of the eigenmodes, they are normalised with respect to the metric provided by $\mathbf{M}$. That is, $\boldsymbol{\phi}_i$ are selected such that $\boldsymbol{\phi}_i^\top \mathbf{M}\boldsymbol{\phi}_i = 1$, and consequently $\boldsymbol{\phi}_i^\top \mathbf{K}\boldsymbol{\phi}_i = \omega_i^2$, for $i = 1, 2, \ldots, n_d$.
The solution $\mathbf{U}(t)$ of Eq. (3.4) is recovered as a linear combination of the modes $\boldsymbol{\phi}_i$ multiplied by their time harmonic dependence, $\cos(\omega_i t - \psi_i)$, and their corresponding amplitude. The amplitude and the phase ψ_i of each mode are to be computed using the initial conditions.
Often, structural engineering analysis does not require obtaining the full time dependence. The fundamental modes corresponding to the lowest eigenfrequencies provide the essential information to assess the structural dynamics response. Thus, obtaining the lowest eigenfrequencies ω_i and their corresponding eigenmodes is a pertinent outcome for engineering analysis.

3.1.2 Numerical eigenvalue solver: the inverse power method

Several numerical methods are available in the literature ([84], [85], [86]) in order to solve the eigenvalue problem (3.7). Depending on the mathematical structure, the number of eigenpairs of interest and the computational cost associated to the algebraic operations, a different numerical eigensolver might be recommended. Roughly speaking, two main categories can be identified: *global* approaches, such as the QR method ([87]), that approximates all the eigenvalues, or *partial* methods which compute a smaller set of eigenvalues, such as Lanczos [88], Arnoldi [89], Davidson [90], and Jacobi-Davidson [91] methods. The simplest eigensolvers are the well-known power methods, that aim at computing the largest eigenvalue and its corresponding eigenvector (direct power method) or the lowest (inverse power method). After computing the largest (resp. lowest) eigenvalue, a deflation technique is used to remove it from the problem and the same method provides the second largest (resp. lowest). In this work we are interested only in a few lowest eigenvalues, so the IPM represents a suitable choice. Moreover, the generalisation to the parametric problem introduced in the next section suggests adopting the algorithmically simplest methodology.

Assuming $\sigma_1 > 0$, the IPM iterates approximations to eigenvector $\boldsymbol{\phi}_1$. An initial guess $\boldsymbol{\phi}_1^0$ is selected. Then, the ν-th iteration $\boldsymbol{\phi}_1^\nu$ is obtained solving the following linear system

$$\mathbf{K}\boldsymbol{\phi}_1^\nu = \mathbf{M}\boldsymbol{\phi}_1^{\nu-1}\,, \text{ for } \nu = 1, 2, 3\ldots \tag{3.9}$$

The iterative sequence is expected to converge to $\boldsymbol{\phi}_1$ and the corresponding eigenvalue ω_1^2 is obtaining computing the Rayleigh quotient, as indicated in Eq. (3.8). Note also that the obtained value of $\boldsymbol{\phi}_1$ is to be normalised by dividing by its $\mathbf{M}$-norm, that is dividing by $\sqrt{\mathbf{M}^\top \boldsymbol{\phi}_1 \mathbf{M}}$.

The subsequent eigenvectors and eigenvalues are computed using a deflation technique, which consists of removing the already computed eigenvectors from the original matrix. In practice, in a power method this can also be done at the level of the iteration (3.9), enforcing orthogonality to the already computed eigenvectors. Thus, the second smallest eigenvalue σ_2

and its corresponding eigenvector $\boldsymbol{\phi}_2$ are computed similarly, but enforcing $\mathbf{M}$-orthogonality to $\boldsymbol{\phi}_1$ (which is already available) at every iteration, that is $\boldsymbol{\phi}_1^\top \mathbf{M} \boldsymbol{\phi}_2^\nu = 0$. In the standard algorithms, this is readily done subtracting the projection of the iterated approximation, for example by means of the Gram-Schmidt-type orthogonalisation processes (see [86]), that is replacing $\boldsymbol{\phi}_2^\nu$ by

$$\boldsymbol{\phi}_2^\nu - (\boldsymbol{\phi}_1^\top \mathbf{M}\, \boldsymbol{\phi}_2^\nu)\boldsymbol{\phi}_1. \tag{3.10}$$

Here, for the sake of easing the generalisation to the parametric case analysed in the next section, an equivalent strategy is adopted, enforcing orthogonality already in the linear system to be solved at each iteration using Lagrange multipliers. Accordingly, Eq. (3.9) is to be replaced by a new $(n_d + 1) \times (n_d + 1)$ system of equations

$$\begin{bmatrix} \mathbf{K} & \mathbf{M}\boldsymbol{\phi}_1 \\ (\mathbf{M}\boldsymbol{\phi}_1)^\top & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_2^\nu \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{M}\boldsymbol{\phi}_2^{\nu-1} \\ 0 \end{bmatrix}, \tag{3.11}$$

where λ is the Lagrange multiplier, which is an instrumental unknown to be discarded as part of the result.

When the first n eigenvalues $\sigma_1, \sigma_2, \ldots, \sigma_n$ are already computed and the corresponding eigenvectors are collected in the $n_d \times n$ matrix $\boldsymbol{\Phi}_n = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_n]$, the iterative scheme to compute $\boldsymbol{\phi}_{n+1}$ boils down to solve the following $(n_d + n) \times (n_d + n)$ system of equations at each iteration

$$\begin{bmatrix} \mathbf{K} & \mathbf{M}\boldsymbol{\Phi}_n \\ (\mathbf{M}\boldsymbol{\Phi}_n)^\top & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_{n+1}^\nu \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{M}\boldsymbol{\phi}_{n+1}^{\nu-1} \\ \mathbf{0} \end{bmatrix}, \tag{3.12}$$

where $\boldsymbol{\lambda}$ is the $n \times 1$ vector of Lagrange multipliers.

An important case to take into account, which is typical in the automotive and aerospace applications, is the case of a free-free structure (with no loads and no constrains). The stiffness matrix $\mathbf{K}$ associated to this kind of systems is singular, with a six-dimensional kernel (in 3D) containing the rigid-body modes. This is to say $\sigma_i = 0$ for $i = 1, 2, \ldots, 6$ and the corresponding rigid-body modes are precisely $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_6$, collected in

the $n_d \times 6$ matrix $\mathbf{\Phi}_6$.

The computation of the 6 rigid-body modes is to be performed using a different technique concerning only matrix $\mathbf{K}$ (therefore, independent of $\mathbf{M}$), as it was described in Section 2.2 and Section 2.3.1 for the standard implementation and the parametric one, respectively.

Once the rigid-body modes $\mathbf{\Phi}_6$ are obtained, the fundamental eigenfrequency and eigenvector, σ_7 and $\boldsymbol{\phi}_7$ are computed using the iterative scheme described in Eq. (3.12). This is summarised in Algorithm 1 for a general eigenvalue ω_{n+1}^2 and eigenvector $\boldsymbol{\phi}_{n+1}$, assuming that the previous ones are available in $\mathbf{\Phi}_n$.

Algorithm 1 Inverse power method to compute $\boldsymbol{\phi}_{n+1}$ and ω_{n+1}

Input: $\mathbf{K}$, $\mathbf{M}$, $\mathbf{\Phi}_n$, tolerance ε and initial guess $\boldsymbol{\phi}^{\mathtt{old}}$ (for $\nu = 0$)

1: Normalise $\boldsymbol{\phi}^{\mathtt{old}} \leftarrow \boldsymbol{\phi}^{\mathtt{old}} / \sqrt{\boldsymbol{\phi}^{\mathtt{old}\top} \mathbf{M}\, \boldsymbol{\phi}^{\mathtt{old}}}$

2: **while** $E_\phi > \varepsilon \, \|\boldsymbol{\phi}^{\mathtt{new}}\|$ **do**

3: Solve $\begin{bmatrix} \mathbf{K} & \mathbf{M}\mathbf{\Phi}_n \\ (\mathbf{M}\mathbf{\Phi}_n)^\top & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}^{\mathtt{new}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{M}\boldsymbol{\phi}^{\mathtt{old}} \\ \mathbf{0} \end{bmatrix}$

4: Normalise $\boldsymbol{\phi}^{\mathtt{new}} \leftarrow \boldsymbol{\phi}^{\mathtt{new}} / \sqrt{\boldsymbol{\phi}^{\mathtt{new}\top} \mathbf{M}\, \boldsymbol{\phi}^{\mathtt{new}}}$

5: Compute errors $E_\phi = \|\boldsymbol{\phi}^{\mathtt{new}} - \boldsymbol{\phi}^{\mathtt{old}}\|$

6: Update $\boldsymbol{\phi}^{\mathtt{old}} \leftarrow \boldsymbol{\phi}^{\mathtt{new}}$

7: Store solution $\boldsymbol{\phi}_{n+1} = \boldsymbol{\phi}^{\mathtt{new}}$

8: Compute $\omega_{n+1} = \sqrt{\boldsymbol{\phi}_{n+1}^\top \mathbf{K}\, \boldsymbol{\phi}_{n+1}}$

Output: $\boldsymbol{\phi}_{n+1}$ & ω_{n+1}

3.2 Parametric modal analysis

As mentioned in the introduction, the goal of this work is to solve the parametric version of the problem stated in Section 3.1, and more precisely of the eigenvalue problem (3.7) arising from the modal analysis. The input data of the problem are assumed to depend on a set of n_p parameters $\boldsymbol{\mu} = [\mu_1, \mu_2, \ldots, \mu_{n_p}]^T \in \boldsymbol{\mathcal{M}} \subset \mathbb{R}^{n_p}$ describing the material properties (e.g. elastic modulus, density, etc.) and the geometric characterisation of the shape of the structure. Typically, the multidimensional parametric domain $\boldsymbol{\mathcal{M}}$ is defined as the Cartesian product of sectional intervals, for each one of the parameters, namely $\boldsymbol{\mathcal{M}} := \mathcal{M}_1 \times \mathcal{M}_2 \times \cdots \times \mathcal{M}_{n_p}$, with $\mu_j \in \mathcal{M}_j \subset \mathbb{R}$ for $j = 1, \ldots, n_p$. Based on this assumption, the parametric version of the modal analysis results in solving the eigenvalue problem (3.7) for parameter-dependent input matrices, that is $\mathbf{K}(\boldsymbol{\mu})$ and $\mathbf{M}(\boldsymbol{\mu})$. The solution also depends on the design parameters and it is given in terms of eigenvalues $\omega^2(\boldsymbol{\mu})$ and eigenvectors $\boldsymbol{\phi}(\boldsymbol{\mu})$.

3.2.1 Inverse power iteration and deflation with the encapsulated PGD toolbox

A coupling of the PGD method with the IPM is proposed to solve the parametric eigenvalue problem. The final goal is being able to compute a set of n_{eig} smallest (in magnitude) non-zero eigenvalues and the corresponding eigenmodes, both in a parametric format. From a conceptual point of view, the extension of the IPM algorithm from the non-parametric to the parametric framework is as simple as rewriting all quantities in Algorithm 1 with their parametric dependency. However, in a numerical sense, the algorithm requires several parametric algebraic operations to be performed, which is certainly a challenging task. The recently developed encapsulated PGD toolbox introduced in the previous chapter and fully described in [73], plays a key role in this work. The toolbox contains a collection of PGD-based algorithms able to perform algebraic operations between parametric objects, such as scalars, vectors and matrices depending on the parameters $\boldsymbol{\mu}$. This allows the introduction of a general methodology to solve all the parametric operations involved in the IPM algorithm, that is: 1) solving a

linear system of equations (line 3 in Algorithm 1); 2) divide a vector by a scalar (lines 1 and 4), vector-matrix-vector product (lines 1, 4 and 8); and 3) compute the square root of a scalar (parameter-dependent) quantity (lines 1, 4 and 8). All operations but the square root were already available in the toolbox. The square root routine was developed in the context of this work and added to the toolbox, following the algorithmic structure described in Appendix B.

According to the standard PGD philosophy, the encapsulated toolbox assumes a separated representation of the multidimensional solution, which is obtained using a greedy-type algorithm combined with an alternated directions scheme, as depicted in Fig. 2.2 of the previous chapter and deeply described in [73] and [92]. As a consequence, the PGD approximation of the nth eigenmode $\boldsymbol{\phi}_n(\boldsymbol{\mu})$ can be expressed as the linear combination of an a priori unknown number of terms $n_{\boldsymbol{\phi}_n}$, namely

$$\boldsymbol{\phi}_n(\boldsymbol{\mu}) \approx \boldsymbol{\phi}_n^{\texttt{PGD}}(\boldsymbol{\mu}) = \sum_{i=1}^{n_{\boldsymbol{\phi}_n}} \beta_{\boldsymbol{\phi}_n}^i \, \boldsymbol{\phi}_n^i \prod_{j=1}^{n_p} F_{\boldsymbol{\phi}_n}^{j,i}(\mu_j). \tag{3.13}$$

Every ith term of this sum is given by the product of a series of functions, each one depending separately on one of the problem parameters. More precisely, $\boldsymbol{\phi}_n^i$ refers to the spatial dimension and $F_{\boldsymbol{\phi}_n}^{j,i}(\mu_j)$ corresponds to the set of parametric functions depending separately on each parameter μ_j, for $j = 1, 2, \ldots, n_p$. If spatial and parametric terms are normalised, a factor or amplitude $\beta_{\boldsymbol{\phi}_n}^i$ appears, that indicates the relevance of the ith term of the sum to the final separated solution.

Analogously, the PGD approximation of the angular frequency $\omega_n(\boldsymbol{\mu})$ reads

$$\omega_n(\boldsymbol{\mu}) \approx \omega_n^{\texttt{PGD}}(\boldsymbol{\mu}) = \sum_{i=1}^{n_{\omega_n}} \beta_{\omega_n}^i \prod_{j=1}^{n_p} F_{\omega_n}^{j,i}(\mu_j). \tag{3.14}$$

Note that, being ω_n a scalar, its spatial part becomes β_{ω_n}. In order to perform parametric operations by means of the encapsulated toolbox, the input data of the routines must also be provided in the separated form. In this case, a pre-process step is necessary to define the parametric stiffness

and mass matrices as

$$\mathbf{K}(\boldsymbol{\mu}) = \sum_{i=1}^{n_{\mathbf{K}}} \mathbf{K}^i \prod_{j=1}^{n_p} F_{\mathbf{K}}^{j,i}(\mu_j) \quad \text{and} \quad \mathbf{M}(\boldsymbol{\mu}) = \sum_{i=1}^{n_{\mathbf{M}}} \mathbf{M}^i \prod_{j=1}^{n_p} F_{\mathbf{M}}^{j,i}(\mu_j). \quad (3.15)$$

If an exact analytical separated expression exist for the two matrices, the amplitudes do not appear in (3.15) because no normalisation of the PGD terms is needed.

As already discussed in Sec. 2.3.3 of the previous chapter, it is not always trivial to find a separated analytical representation of the input data, especially when geometric parameters are considered in the problem. Furthermore, assuming that a separated expression can be found, standard a priori PGD methods require a modification of the weak form of the problem. Such an "intrusive" approach is practically not employable in the industrial context, where commercial FE software is used, i.e. source codes are typically not accessible. For this reason, the *hybrid* and *nonintrusive* algebraic approach proposed in the previous chapter and presented in [93] is employed also in this dynamic study. The main idea is to add a pre-process step to the method. From the computational point of view, this process consists in assembling (without solving the problem) the input matrices for every possible combination of the parametric values in the discretized domain and then express them in the separated format. If each introduced parametric dimension $\mu_j \in \mathcal{M}_j$ for $j = 1, 2, \ldots, n_p$ is discretised using m_j nodal values, then the full-order sampling of the parametric matrices consists of evaluating $\mathbf{K}(\boldsymbol{\mu})$ and $\mathbf{M}(\boldsymbol{\mu})$ in the set of m_{tot} points used to discretise the parametric domain $\boldsymbol{\mathcal{M}} := \mathcal{M}_1 \times \mathcal{M}_2 \times \cdots \times \mathcal{M}_{n_p}$, where $m_{tot} = \prod_{j=1}^{n_p} m_j$. It is worth noting that this technique preserves efficiency, since the computational cost for the matrices assembly is usually small compared to the cost of a full-order dynamic simulation. For the simple example tested in this work, the time needed to assemble the input matrices for one configuration is four times cheaper than solving the corresponding dynamic problem. If more complex examples and finer meshes are considered, the cost of the assembling is expected to be negligible compared to the full-order computation. In addition, the sampling of the matrices can be done in parallel. Once the matrices are sampled, they can be

expressed as in Eq. (3.15). The number of PGD terms needed to describe the two matrices can be reduced to a much smaller number than m_{tot} by performing a PGD-based data compression [92]. In fact, every time a PGD operation (such as product, sum, difference) leads to a large number of terms, data compression is advisable to reduce the number of PGD terms whilst maintaining a desired level of accuracy.

If the structure is unconstrained, the parametric matrix $\boldsymbol{\Phi}_6(\boldsymbol{\mu})$ containing the first six (in 3D) rigid-body modes must be also given as separated input data to the IPM. As explained in detail in the previous chapter (Sections 2.2-2.3), it can be computed in a separated format as the kernel of the stiffness matrix. These rigid-body modes depend on the design variables only if geometric parameters are involved in the problem because, by definition, the rigid body modes of a structure do not depend on the material properties.

Once the initial input of the problem is available, a "cascade" application of the PGD method can be performed, in the sense that the output of each parametric algebraic operation, obtained by calling an encapsulated PGD routine, can be directly used as the input of the next operation, until the global solution is obtained. The parametric input data are computed as separated multidimensional tensors, which we indicate with the superscript $\square^{\texttt{PGD}}$. An overloading of the arithmetic operators allows to use the standard Matlab symbols to call the algebraic operations contained in the encapsulated library, which makes the method highly user-friendly. Table 3.1 summarises some of the algebraic operations available in the encapsulated PGD library and the corresponding Matlab symbols.

Figure 3.1 shows a pseudo-code describing the algorithmic aspects of the method when the first non-zero n_{eig} smallest eigenpairs of an unconstrained structures are sought.

Analogously to Eq. (3.12), the parametric IPM consists of iteratively solving the following system of equations:

$$\begin{bmatrix} \mathbf{K}(\boldsymbol{\mu}) & \mathbf{M}(\boldsymbol{\mu})\boldsymbol{\Phi}_n(\boldsymbol{\mu}) \\ (\mathbf{M}(\boldsymbol{\mu})\boldsymbol{\Phi}_n(\boldsymbol{\mu}))^\top & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_{n+1}(\boldsymbol{\mu}) \\ \boldsymbol{\lambda}(\boldsymbol{\mu}) \end{bmatrix} = \begin{bmatrix} \mathbf{M}(\boldsymbol{\mu})\boldsymbol{\phi}_{n+1}(\boldsymbol{\mu}) \\ \mathbf{0} \end{bmatrix}. \tag{3.16}$$

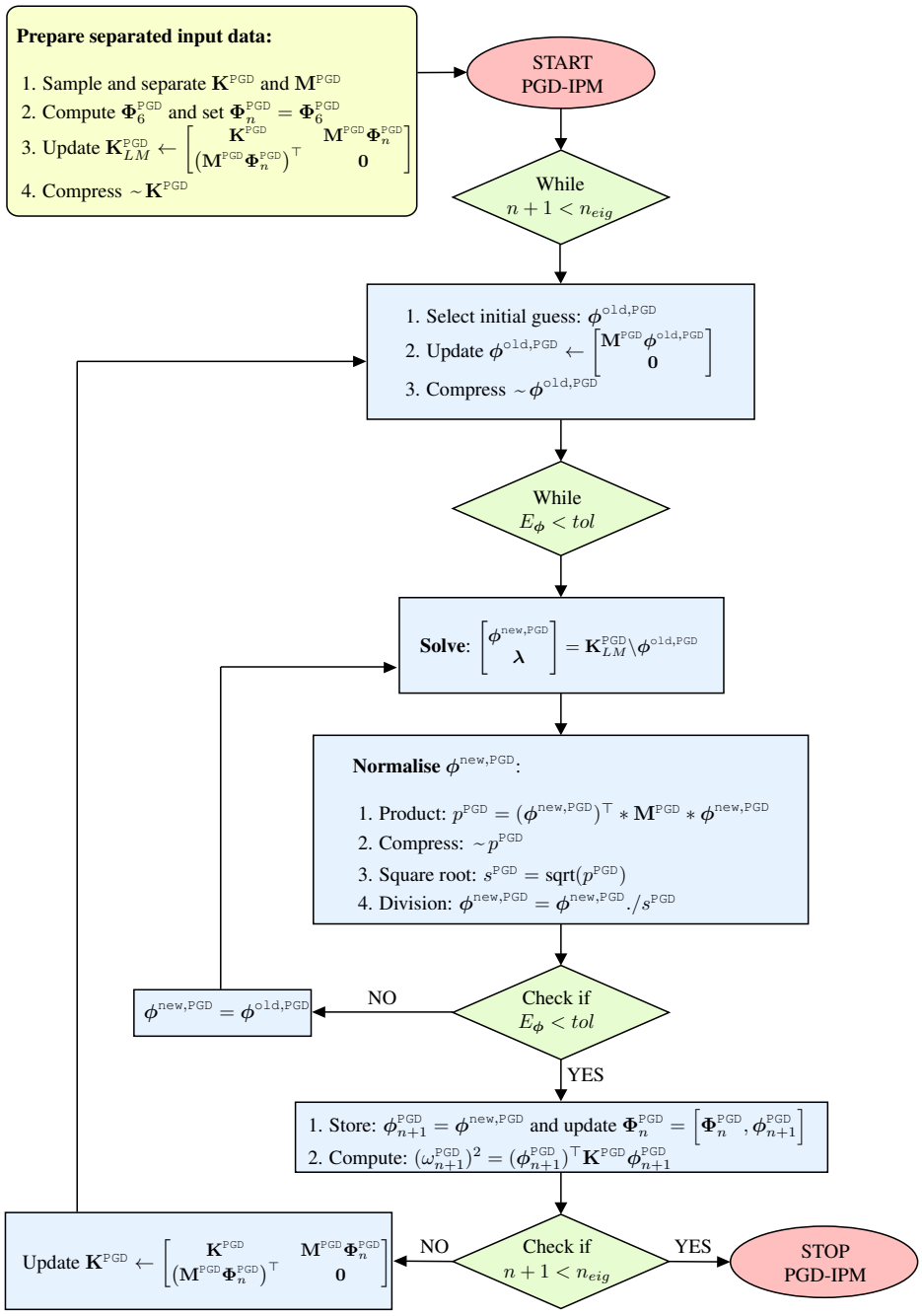


Figure 3.1: Pseudo-code of the PGD-IPM method.

Table 3.1: Some of the algebraic operations available in the encapsulated PGD toolbox. Matlab symbols can be used in a standard way.

Matlab symbol	**Encapsulate PGD routine**
$\mathbf{K}^{\texttt{PGD}} \backslash \boldsymbol{f}^{\texttt{PGD}}$	Solve linear system
$\boldsymbol{u}^{\texttt{PGD}} \pm \boldsymbol{v}^{\texttt{PGD}}$	Sum (or difference)
$\mathbf{K}^{\texttt{PGD}} * \boldsymbol{u}^{\texttt{PGD}}$	Product
$\boldsymbol{u}^{\texttt{PGD}} ./ \boldsymbol{v}^{\texttt{PGD}}$	Division
$(\boldsymbol{u}^{\texttt{PGD}})'$	Transpose
sqrt$(\boldsymbol{u}^{\texttt{PGD}})$	Square root
$\sim \boldsymbol{u}^{\texttt{PGD}}$	Compression
$\left[\boldsymbol{u}^{\texttt{PGD}}, \boldsymbol{v}^{\texttt{PGD}}\right]$ and $\left[\boldsymbol{u}^{\texttt{PGD}}; \boldsymbol{v}^{\texttt{PGD}}\right]$	Arrays concatenation

If the first non-zero eigenmode $\boldsymbol{\phi}_7(\boldsymbol{\mu})$ is sought, then the matrix of already known mode shapes coincides with the matrix of rigid body modes, i.e. $\boldsymbol{\Phi}_n(\boldsymbol{\mu}) = \boldsymbol{\Phi}_6(\boldsymbol{\mu})$. For every new computed eigenmode $\boldsymbol{\phi}_{n+1}$, the matrix is updated as $\boldsymbol{\Phi}_n(\boldsymbol{\mu}) = [\boldsymbol{\Phi}_n(\boldsymbol{\mu}), \boldsymbol{\phi}_{n+1}]$. The Lagrange multipliers $\boldsymbol{\lambda}(\boldsymbol{\mu})$ are introduced in order to get unicity by enforcing mass-orthogonality to the set of already known eigenvectors. It is important to point out that, according to the author's knowledge, this deflation technique has not been proposed before in the context of the standard IPM. This is due to the fact that in a non-parametric context other more efficient methods can be used, as discussed in Section 3.1.1. Nevertheless, in this novel extension of the eigenvalue problem to the parametric framework, the Lagrange multipliers technique proved to be the most efficient. Alternative strategies were also tested. For example, Felippa et al. [94] proposed a modification of the stiffness matrix as $\mathbf{K} + \boldsymbol{\Phi}_6 \boldsymbol{\Phi}_6^\top$, whose eigenvalues are identical to those of $\mathbf{K}$ but the eigenvalues associated to the rigid body modes are raised to unity. This overcomes the stiffness singularity but leads to full matrices, which is not advisable, especially in the parametric

format. Furthermore, extra operations should be performed in order to normalise the obtained eigenvectors at every iteration, e.g. by means of the Gram-Schmidt orthogonalisation. The Lagrange multipliers technique proposed here solves the singularity issue and compute an orthonormal set of eigenvectors at the same time, reducing to the minimum the number of algebraic operations needed in the parametric case.

As it is shown in the pseudo-code, a PGD guess vector $\boldsymbol{\phi}^{\texttt{old}}$ needs to be prepared every time a new eigenmode is sought. Then the extended system in Eq. (3.16) is iteratively solved by calling the corresponding encapsulated routine until convergence is reached, that is when a quantity $E_{\boldsymbol{\phi}}$ is smaller than a user-defined tolerance. Here $E_{\boldsymbol{\phi}}$ is defined as:

$$E_{\boldsymbol{\phi}} = \frac{\left| \sum_{i=1}^{n_{\boldsymbol{\phi}^{\texttt{new}}}} \beta^i_{\boldsymbol{\phi}^{\texttt{new}}} - \sum_{j=1}^{n_{\boldsymbol{\phi}^{\texttt{old}}}} \beta^j_{\boldsymbol{\phi}^{\texttt{old}}} \right|}{\sum_{j=1}^{n_{\boldsymbol{\phi}^{\texttt{old}}}} \beta^j_{\boldsymbol{\phi}^{\texttt{old}}}}, \tag{3.17}$$

where $\beta^i_{\boldsymbol{\phi}^{\texttt{new}}}$ and $\beta^j_{\boldsymbol{\phi}^{\texttt{old}}}$ represent, respectively, the amplitudes of the PGD terms describing $\boldsymbol{\phi}^{\texttt{new}}$ and $\boldsymbol{\phi}^{\texttt{old}}$ after being normalised. The normalisation is performed every time a new eigenvector $\boldsymbol{\phi}^{\texttt{new}}$ is computed, by dividing it by its $\mathbf{M}$-norm, namely $\sqrt{(\boldsymbol{\phi}^{\texttt{new,PGD}})^\top \mathbf{M}^{\texttt{PGD}} \boldsymbol{\phi}^{\texttt{new,PGD}}}$, where *sqrt* and $*$ are the Matlab symbols used to perform the encapsulated square root and product between PGD objects. Note that, in a parametric format, the normalisation requires four algebraic operations: a product, a compression, a square root and a division.

Finally, once convergence is reached, the sought eigenvector $\boldsymbol{\phi}^{\texttt{PGD}}_{n+1}$ can be stored and the corresponding eigenvalue can be calculated according to the Rayleigh quotient $(\omega^{\texttt{PGD}}_{n+1})^2 = (\boldsymbol{\phi}^{\texttt{PGD}}_{n+1})^\top \mathbf{K}^{\texttt{PGD}} \boldsymbol{\phi}^{\texttt{PGD}}_{n+1}$. The same procedure is repeated until the desired number n_{eig} of eigenpairs is obtained.

It is worthy to emphasise that the described algorithm is solved by means of just one offline computation. The resulting eigenpairs represent the compact version of all the possible solutions for every value of the parameters. Once this offline stage is completed, the obtained *computational vademecum* can

be used for optimisation studies, or it could be uploaded on light computing devices such as a tablet, where the designers could visualise in real-time how the global response of the structure would change with a variation of the parameters.

3.3 Numerical examples: parametric inverse power method with material and geometric parameters

A numerical example is presented in order to show the properties of the proposed method. The same linear elastic 3D solid structure characterised by one material and one geometric parameter presented in Sec. 2.4 is employed. The two variables, which are treated as additional coordinates of the problem, are denoted with $\mu \in \mathcal{M}_\mu$ and $\theta \in \mathcal{M}_\theta$ for the material and geometric parameters, respectively.

The objective of this numerical test is to explore how changing the introduced design parameters affect the dynamic response of the structure. This can be done by performing a parametric modal analysis of the structure by means of the proposed PGD-IPM eigensolver. As explained in the previous section, the first essential step consists of defining the input data (i.e. stiffness and mass matrices) in a separated format.

This step was already performed in the previous chapter for the same structure. As a consequence, the separated stiffness and mass matrix obtained in Sec. 2.4 (Eqs. (2.36) and (2.37)) were used also in this example. Once the separated representation of the input data is available, the proposed method can be finally employed following the steps shown in Fig. 3.1. In this example, the goal is to compute the first three mode shapes $\boldsymbol{\phi}_n(\mu, \theta)$, with $n = 7, 8, 9$, corresponding to the smallest non-zero eigenvalues. For each nth mode, the generalised PGD solution reads:

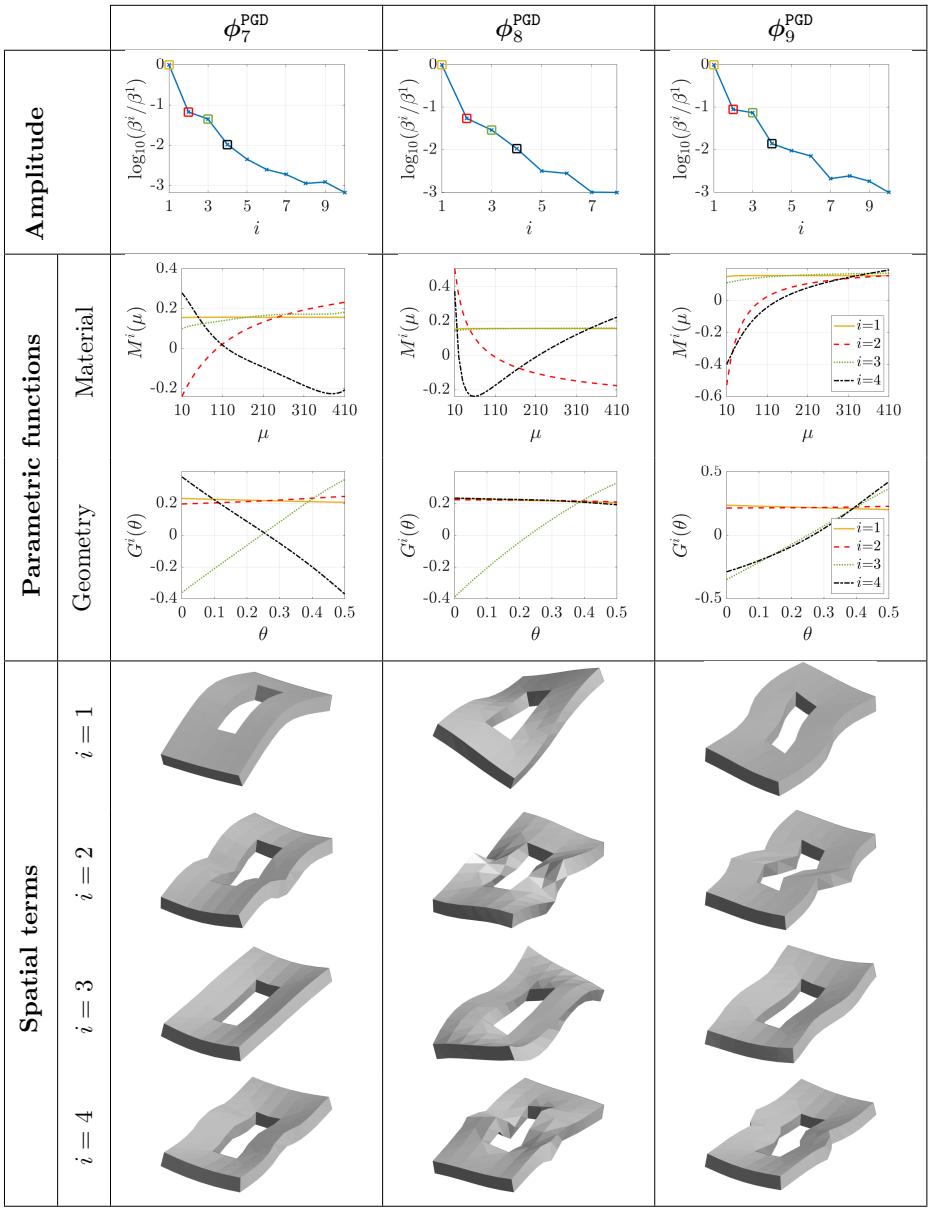
$$\boldsymbol{\phi}_n(\mu, \theta)^{\texttt{PGD}} = \sum_{i=1}^{N_{\boldsymbol{\phi}}} \beta^i \boldsymbol{\phi}_n^i M^i(\mu) G^i(\theta). \tag{3.18}$$

Fig. 3.2 plots the generalised results in terms of amplitude, parametric and spatial terms for the three modes $\boldsymbol{\phi}_7^{\texttt{PGD}}, \boldsymbol{\phi}_8^{\texttt{PGD}}, \boldsymbol{\phi}_9^{\texttt{PGD}}$. As expected, the amplitudes β^i rapidly decrease as the number of PGD terms increase. Using a tolerance of 10^{-3} to stop the enrichment of the PGD solution, a maximum number of 10 terms is needed to get an accurate representation of the three parametric mode shapes. In addition, the results show that the first four PGD terms capture the most relevant information of the generalised solution, as the fifth and subsequent terms have an amplitude at least two orders of magnitude lower than the amplitude of the first PGD term. For this reason, the first four parametric and spatial terms are also shown in Fig. 3.2. For each mode shape, same colours are used to depict the parametric functions and amplitudes related to the same ith term. The parametric functions seem to show a higher influence of the material parameter then the geometric one on the final response. The spatial modes provide an illustration of the deformation induced by the four most relevant terms of the generalised solution. Those illustrations already allow to identify the type of mode shapes. In fact, the first mode $\boldsymbol{\phi}_7^{\texttt{PGD}}$ can be identified as a flexional mode, the second $\boldsymbol{\phi}_8^{\texttt{PGD}}$ as torsional, while the third $\boldsymbol{\phi}_9^{\texttt{PGD}}$ mode shows a shear-type deformation along the longest edge of the structure.

In order to get a particularised solution for a chosen set of the parameters $(\bar{\mu}, \bar{\theta})$, the corresponding function values $M^i(\bar{\mu})$ and $G^i(\bar{\theta})$ are evaluated for each PGD-term i and then multiplied by the corresponding spatial term and amplitude of the desired mode shape. This procedure can be easily performed in a post-process step, providing real-time results for any combination of the parameters.

As already known, once the parametric mode shapes are available, the corresponding three eigenvalues can be simply computed by means of the Rayleigh quotient. In order to validate the PGD results, the accuracy with respect to the full-order FE computations is measured as the relative error between the PGD and FE eigenvalue solutions in the $\mathcal{L}_2(\mathcal{M}_\mu \times \mathcal{M}_\theta)$ norm,

Figure 3.2: PGD generalised solution for the first three shape modes $\boldsymbol{\phi}_7^{\texttt{PGD}}, \boldsymbol{\phi}_8^{\texttt{PGD}}$ and $\boldsymbol{\phi}_9^{\texttt{PGD}}$. For each mode, the evolution of the amplitude β^i of the PGD terms , the first four parametric functions and first four spatial terms are shown. Same colour is assigned to amplitudes values and corresponding parametric functions.



that is

$$\|\varepsilon_{\text{PGD}}\|_2 = \left(\frac{\int_{\mathcal{M}_\theta} \int_{\mathcal{M}_\mu} \left(\omega^{\texttt{PGD}} - \omega^{\texttt{FE}}\right)^2 \, d\mu \, d\theta}{\int_{\mathcal{M}_\theta} \int_{\mathcal{M}_\mu} (\omega^{\texttt{FE}})^2 \, d\mu \, d\theta} \right)^{1/2} . \tag{3.19}$$

Also the maximum error is calculated as the $\mathcal{L}_\infty(\mathcal{M}_\mu \times \mathcal{M}_\theta)$ norm:

$$\|\varepsilon_{\text{PGD}}\|_\infty = \max_{\mu \in \mathcal{M}_\mu, \theta \in \mathcal{M}_\theta} \left(\frac{\left|\omega^{\texttt{PGD}} - \omega^{\texttt{FE}}\right|}{\omega^{\texttt{FE}}} \right) . \tag{3.20}$$

Table 3.2 reports the calculated $\mathcal{L}_2$ and $\mathcal{L}_\infty$ errors for the three computed eigenvalues, proving that a high level of accuracy can be obtained by using the proposed PGD-IPM method. It is worth noting that to compute this error measure, the problem had to be solved by means of the standard FE method for each possible combination of the parameters, that is $n_\mu \times n_\theta = 21 \times 41 = 861$ FE simulations. It is important to underline that the main goal in the PGD context is not to reduce the computational cost, but to provide a method which is able to explore an arbitrary large parametric space with only one offline computation. Another important advantage concerns the storage memory. In fact, the obtained PGD computational vademecum needs ∼182 KB of storage memory for the three eigenmodes versus the ∼14900 KB needed to store 861 full-order FE solutions for each of the three eigenmodes.

Table 3.2: Accuracy of the PGD results with respect to the full-order FE computations measured as the relative error between the PGD and FE solutions in the $\mathcal{L}_2$ and $\mathcal{L}_\infty$ norm according to Eqs. (4.4) and (4.5).

	$\omega_7^{PGD}(\mu,\theta)$	$\omega_8^{PGD}(\mu,\theta)$	$\omega_9^{PGD}(\mu,\theta)$
$\|\varepsilon_{\text{PGD}}\|_2$	9.18×10^{-4}	1.00×10^{-3}	1.00×10^{-3}
$\|\varepsilon_{\text{PGD}}\|_\infty$	4.20×10^{-3}	3.56×10^{-3}	1.15×10^{-1}

To conclude, another important property of the method is shown, which is the possibility to explore the design space and check, in real-time, the effects of the design parameters on a predefined QoI. In this case, the frequency

associated to each mode shape is chosen as QoI, which is computed as:

$$f_n^{\texttt{PGD}}(\mu,\theta) = \frac{\omega_n^{\texttt{PGD}}(\mu,\theta)}{2\pi}. \tag{3.21}$$

The variation of the three frequencies $f_7^{\texttt{PGD}}$, $f_8^{\texttt{PGD}}$ and $f_9^{\texttt{PGD}}$ in the parametric space is depicted in Fig. 3.3. This results clearly show that the geometric parameters has less influence on the frequency, especially for values of μ smaller than the constant Young's modulus E_0 in the remaining domain. This conclusion could be used to make decisions during the design process.

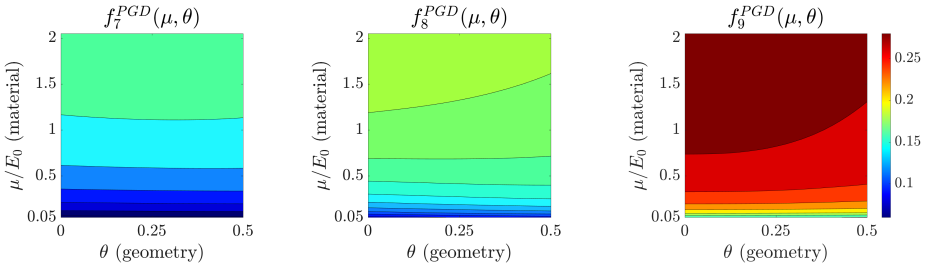


Figure 3.3: Variation of the first three smallest natural frequencies f_7, f_8 and f_9 with respect to the parameters μ and θ

3.4 Chapter summary

This work proposes a nonintrusive algebraic PGD approach combined with the inverse power method (PGD-IPM) to perform the parametric modal analysis of unconstrained structures being characterised by material and geometric parameters. The developed eigensolver uses the library of PGD-based routines implemented by Díez et al. [73] to sequentially perform algebraic operations between parametric objects in a black-box format. In addition, a new algorithm was developed and added to the encapsulated library, which computes the square root of a parametric quantity.

A Lagrange multipliers deflation technique is proposed to overcome singularity issues in the case of unconstrained structures and to compute a multiple set of smallest natural frequencies and corresponding mode shapes.

One numerical example is tested to compute the desired parametric solutions. In particular, an academic test case with one material and one geometric parameter is proposed to show the properties of the PGD solution. The variation of the natural frequencies and mode shapes in the parametric domain is calculated with only one offline computation and the accuracy of the generalised solution is measured by comparing the PGD solution to the whole set of standard FE full-order solutions. The solution for specific sets of parameters are computed in real-time during a post-process step at the negligible cost of a linear combination. With a small effort, the generalised solutions generated by employing the developed technique could be uploaded on portable devices (such as tablets) such that designers could evaluate in real-time the impact of certain parameters on the global response of the structure.

4 Industrial application: parametric NVH study of a BiW structure

The main goal of this chapter is to validate the feasibility of the proposed method in an industrial context. More precisely, the developed PGD-IR and PGD-IPM methods are finally applied to solve the parametric NVH study of a simplified, but realistic, BiW structure. The nonintrusivity of the method is finally proved by showing its capability of interacting with a commercial FE software.

The main novelty is the construction of a parametrised FE model of the BiW by using morphing and optimisation tools, such that complex shape parametrisations can be easily handled in an industrial context. Moreover, the potential of the proposed method in the post-process phase is shown. First, a multi-objective optimisation study is presented in order to find a set of optimal design configurations with respect to the NVH performance. Finally, the ultimate goal is to upload the results of the proposed PGD-NVH study on light computational devices, supporting the designers with the decision-making. For this purpose, a graphical interface app is developed which allows to visualise in real-time how changes in the design variables affect the global response of the BiW.

The remainder of the chapter is structured as follows: Section 4.1 reviews the standard static and dynamic global stiffness analysis of BiW structure, that is the reference full-order problem. In Section 4.2, the problems are redefined in the corresponding parametric framework. In

Section 4.3, the method is finally applied to perform the NVH study of a BiW structure with material and geometric parameters. In particular, three phases are described: the pre-process which concerns the parametrisation of the model and the preparation of the parametric input data; the offline computation, which solves the parametric static and dynamic problem by means of the PGD-IR and PGD-IPM algorithms and discusses the results; finally, the post-process is presented which consists of an optimisation study and the development of the graphical interface app for the real-time visualisation of the results. Section 4.4 gives the final conclusions.

The content of this chapter is based on the following publication:

- **Cavaliere, F.**, Zlotnik, S., Sevilla, R., Larráyoz, X., & Díez, P: Nonintrusive parametric NVH study of a BiW structure. *Submitted.* (2022).

4.1 Review of the standard NVH analysis

As mentioned at the beginning of this thesis, in real applications, the noise and vibration properties of a BiW structure are usually evaluated by performing standard static and dynamic FE analyses. The dynamic study, which consists of extracting the lowest natural frequencies and corresponding vibrational modes by means of the standard modal analysis, allows to identify and optimise the first torsional and bending modes, which are good indicators of the BiW vibrational behaviour. The ride comfort properties, instead, are related to the static torsional and bending global stiffnesses of the BiW structure. For the sake of simplicity, this thesis focuses on the evaluation of the *torsional* static and dynamic properties. Nevertheless, the extension to the bending stiffness or other quantities of interests is straightforward.

The theoretical background of the static ans dynamic global stiffness analyses have been exhaustively discussed in Chapter 2 and Chapter 3, respectively. Shortly, the IR has been introduced as the method typically

employed in the industrial context to perform the static analysis of unconstrained structures, which is how the BiW is represented during the NVH study. Moreover, the IPM algorithm has been indicated as the easiest eigensolver method for the computation of the lowest natural frequencies and corresponding vibrational modes. A global overview of the two approaches in the non-parametric format is given in Fig. 4.1. As recalled by this scheme, the two studies have one first step in common, which is the computation of the rigid body modes $\mathbf{\Phi}$ as the kernel of the global stiffness matrix $\mathbf{K}$. Then, the two analysis can proceed independently until the final solutions are obtained. More precisely, the IR method involves two more steps to finally compute the desired displacement field $\mathbf{U}$. The IPM algorithm, instead, needs to iteratively solve the eigenvalue problem until the first n_{eig} vibrational modes $\boldsymbol{\phi}_i$ with the corresponding eigenfrequencies ω_i (with $i = 1, ..., n_{eig}$) are obtained.
It is important to recall that the general scheme of Fig. 4.1 refers to the static and dynamic analysis of one specific design configuration. This means that, by following the standard procedure, the two algorithms must be solved several times for every time that the design variables are modified. Thanks to the PGD-NVH multidimensional solver introduced in this work, the design problem can be defined in its parametric version and solved with just one offline computation, as described in the next section.

4.2 PGD solver for the parametric NVH analysis

From a conceptual point of view, the extension of the IR and IPM algorithms from the non-parametric to the parametric framework is as simple as rewriting all the quantities outlined in Fig. 4.1 with their parametric dependency. The set of n_p parameters describing the material properties (e.g. elastic modulus, density, etc.) and the geometric characterisation of the BiW shape is defined as $\boldsymbol{\mu} = [\mu_1, \mu_2, \dots, \mu_{n_p}]^\top \in \boldsymbol{\mathcal{M}} \subset \mathbb{R}^{n_p}$. Each parameter μ_j belongs to a predefined interval $\mathcal{M}_j$, such that the multidimensional parametric domain $\boldsymbol{\mathcal{M}}$ is defined as the Cartesian product of the sectional intervals, namely $\boldsymbol{\mathcal{M}} := \mathcal{M}_1 \times \mathcal{M}_2 \times \cdots \times \mathcal{M}_{n_p}$, with $\mu_j \in \mathcal{M}_j \subset \mathbb{R}$ for $j = 1, \dots, n_p$. Since the parameters are treated as

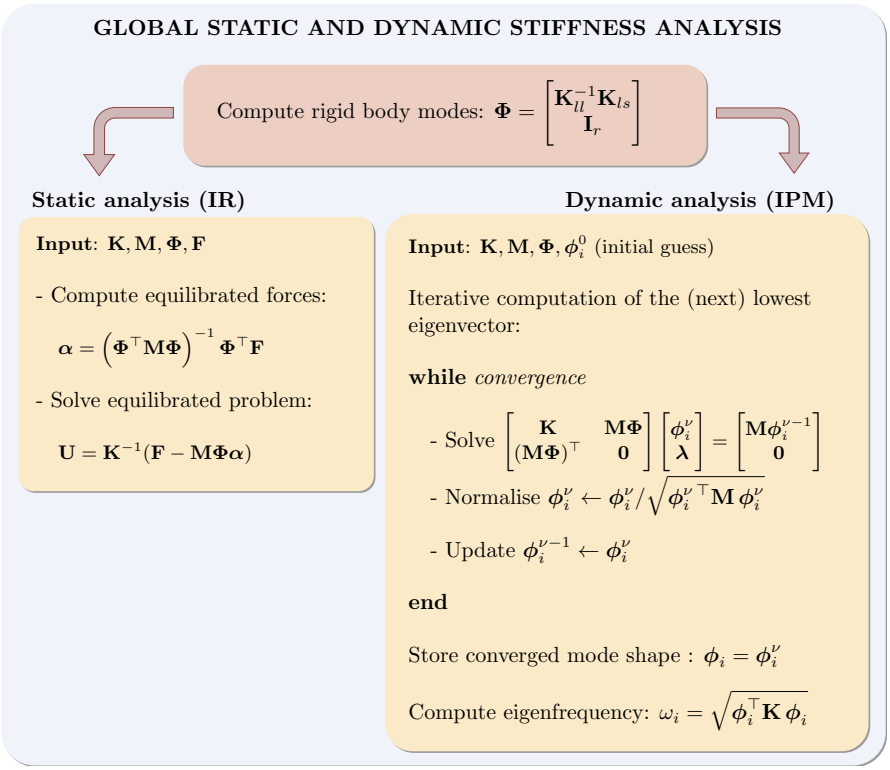


Figure 4.1: Overview of the NVH study in the non-parametric framework. The static analysis is performed by means of the IR method. The IPM algorithm is shown for the computation of the lowest (next) non-zero eigenvector $\boldsymbol{\phi}_i$, assuming that the matrix $\boldsymbol{\Phi}$ of already computed modes is available. Conceptually, if all the input and output are expressed with their parametric dependency, the parametric version of the algorithm is identical.

extra coordinates, the parametric input data, $\mathbf{K}(\boldsymbol{\mu})$ and $\mathbf{M}(\boldsymbol{\mu})$, and the generalised solutions of the IR and IPM algorithms, $\mathbf{U}(\boldsymbol{\mu})$ and $\boldsymbol{\phi}_i(\boldsymbol{\mu})$, are defined in the multidimensional domain $\mathcal{D} = \Omega \times \boldsymbol{\mathcal{M}}$, being Ω the spatial domain and $\boldsymbol{\mathcal{M}}$ the parametric one.

The resulting multidimensional problems can be finally solved by means of the PGD-IR and PGD-IPM methods introduced in the previous chapters. Fig. 4.3 shows a global flowchart in order to recall the main steps of the

two methods. In summary, during the first step, the input quantities of the problem (stiffness and mass matrix) are defined in the well known PGD separated format, which represents the essential requirement of the whole method. Due to the difficulty of finding a separated expression of the input data for complex geometric parametrisation, the *hybrid* and *nonintrusive* algebraic approach described in Sec. 2.3.3 is used.
Once the input stiffness and mass matrices $\mathbf{K}^{\mathtt{PGD}}(\boldsymbol{\mu})$ and $\mathbf{M}^{\mathtt{PGD}}(\boldsymbol{\mu})$ are available, a cascade of algebraic operations between parametric objects (i.e. scalar, vectors, matrices) is solved. This can be done by means of the of PGD-based routines contained in the encapsulated PGD toolbox introduced in the previous chapters. Fig. 4.2 recalls that the big advantage of the toolbox is that it works as a black box. This means that the user just needs to define the parametric input data and call the algorithm that performs the desired operation, which automatically gives back the parametric solution. This is a really appealing feature for its use in an industrial context. In fact, the user would not necessarily need to know what is behind the PGD routine, but he would just need a tutorial on how to use it. The parametric operations involved in the PGD-NVH solver are

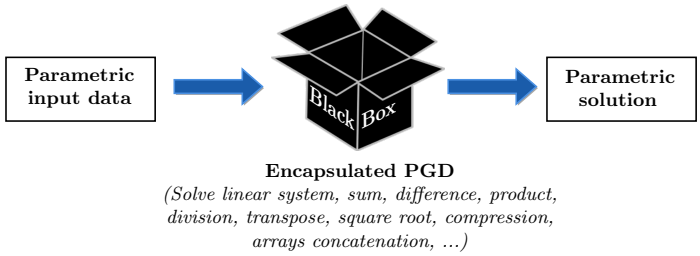


Figure 4.2: Structure of the encapsulated PGD toolbox to perform parametric algebraic operations.

specified in Fig. 4.3, such that the reader can understand the complexity and the potential of the toolbox. It is important to mention that the linear solver, the division, the square root and the compression PGD operations need two kind of tolerances to be set up. One tolerance for the iterative solver scheme, and the other to stop the enrichment of the PGD terms. The compression, which is performed every time the number of PGD terms undergoes a substantial increment (e.g. after products), usually needs

stricter tolerances, such that the accuracy is not compromised.

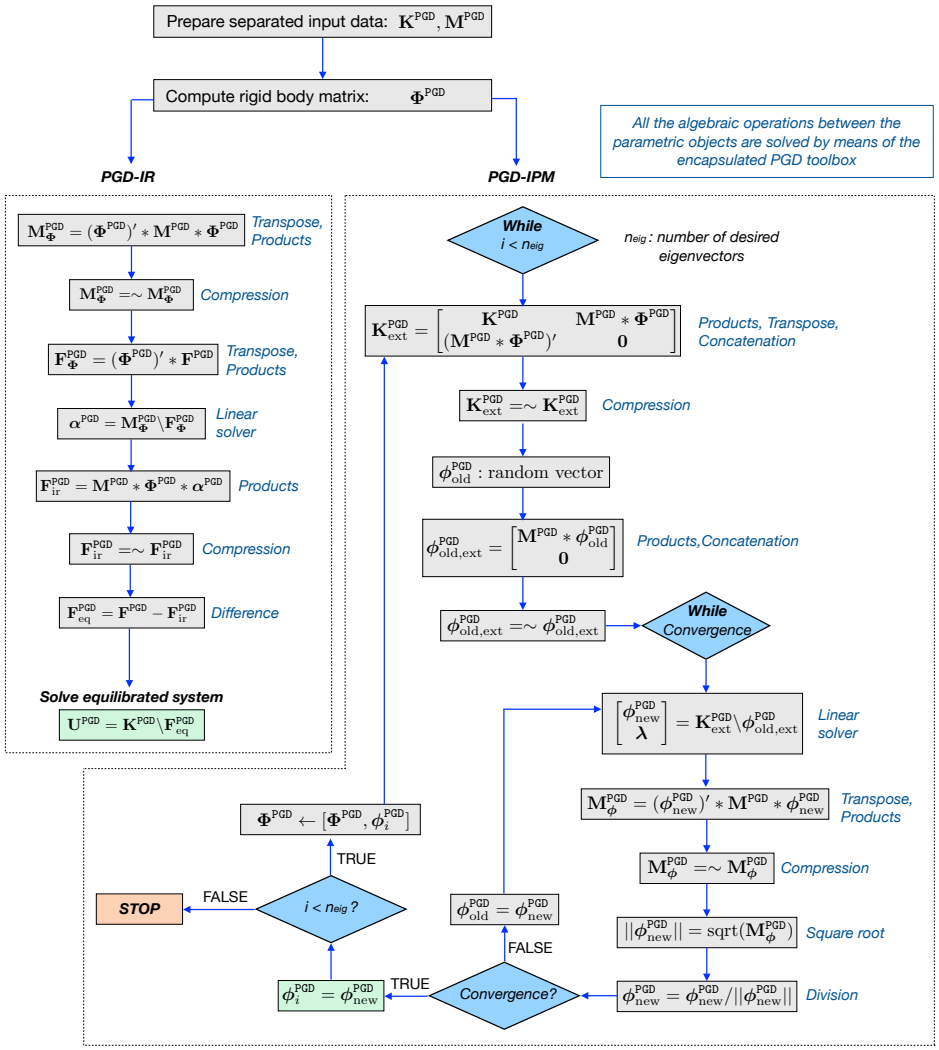


Figure 4.3: Global overview of the parametric NVH algorithms.

Once all the steps are performed, the final parametric solutions $\mathbf{U}^{\text{PGD}}(\boldsymbol{\mu})$ and $\boldsymbol{\phi}_i^{\text{PGD}}(\boldsymbol{\mu})$ can be stored and used in the post-process phase. From Fig. 4.3, it can be observed that the PGD-IR algorithm has a much simpler structure than the PGD-IPM. In fact, the IPM solver needs several iterations to be performed until the solution convergences to the eigenmode. Moreover, this has to be repeated for every new eigenmode of interest. Due to the level of

approximation introduced every time a PGD operation is performed, the user could expect a poor accuracy of the final solution. This is disproved in next section, where the proposed parametric NVH algorithm is finally applied to a realistic industrial case of study, showing successful results for both the static and dynamic analyses.

4.3 Numerical application: parametric NVH study of a BiW structure

The proposed method is finally tested on a simplified BiW structure. Fig. 4.4 (left picture) shows the geometry and the mesh discretisation of the model, which is formed by $3,819$ nodes, each one characterised by six degrees of freedom (three translations and three rotations). Isoparametric triangular and quadrilateral elements based on the Mindlin-Reissner shell formulation (CTRIA3 and CQUAD4 respectively in MSC-Nastran) are used to discretise the model.

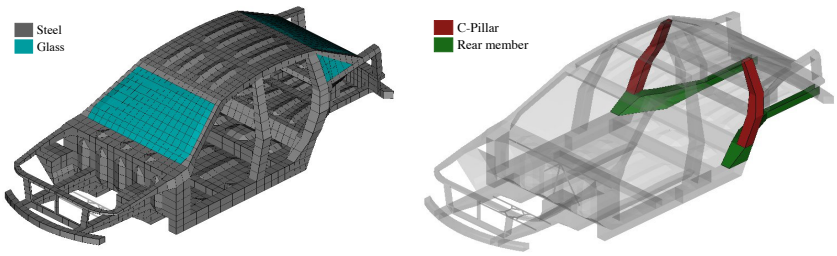


Figure 4.4: Geometry and mesh properties of the BiW structure used for the static and dynamic global stiffness analysis (left). The two car components highlighted (right) are characterised by parametric properties, that is the thickness and the cross section of each one of the components.

All the car components are characterised by isotropic linear elastic materials (MAT1 in MSC-Nastran) with properties described in Table 4.1. In this example, four parameters are introduced as extra coordinates of the problem, which are the thickness and the cross sections of the C-pillars and the rear long members shown in the right picture of Fig. 4.4. Note

Table 4.1: Material properties of the BiW structure.

Material	**Young's modulus**	**Poisson's ratio**	**Density**
Steel	210 GPa	0.29	7820 kg/m^3
Glass	90 GPa	0.14	2480 kg/m^3

that, although from a physical point of view the thickness is meant as a geometric parameter, in the shell element formulation it is treated as a material property. The cross section clearly represents a geometric design variable.
The proposed PGD-NVH methodology consists of three main phases:

1. **Pre-process:** A FE model of the BiW is constructed and parametrised such that the input data of the multidimensional problem can be sampled and expressed in a parametric format;

2. **Offline computation:** The static and dynamic global stiffness analysis are performed in an offline stage by means of one computation which uses the encapsulated PGD toolbox to get the parametric results;

3. **Post-process:** The parametric solution can be used for several purposes, such as efficient optimisation studies and real-time evaluation of the results.

The three steps are described in the following sections.

4.3.1 Pre-process: parametrisation of the BiW model

The pre-process starts with the preparation of a parametrised FE model of the BiW structure. In this work, this is done by means of the ANSA CAE pre-processor software, which contains a powerful *Optimisation Task Tool* able to organize the set-up of an optimisation study. The first step

consists of defining the design variables (DVs) and their ranges of variation. In particular, the DVs of the problem are denoted by $\boldsymbol{\mu} = [\mu_1, \mu_2, \mu_3, \mu_4]$, where μ_1 and μ_2 represent the cross sections of the C-pillar and the rear long member, while μ_3 and μ_4 are the thickness of the same components. The first two geometric parameters are defined by means of a morphing tool available in ANSA, which allows to manage the shape changes. More precisely, the position of the nodes changes without changing the element connectivity. In order to do that, each BiW component affected by the change is selected and subdivided into an *action area* and a *transition area* as depicted in Fig. 4.5. The action area is actively affected by the

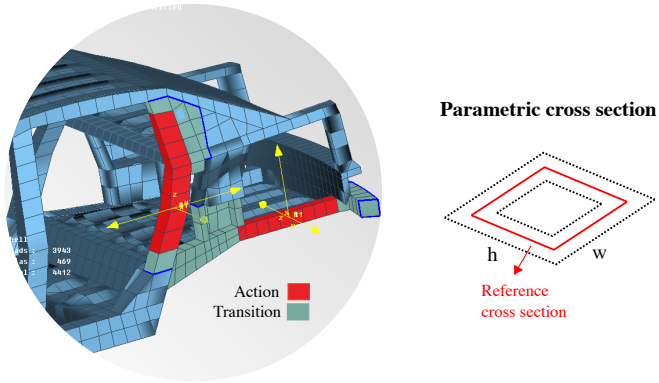


Figure 4.5: Definition of the geometric design variables in Beta-Ansa. The cross section of the components changes in the action area, the modification is smoothly absorbed through a transition area (left). The reference cross section changes its height and width while keeping the shape (right).

change in the cross section, while the transition part is used to smooth the deformation. A reference cross section with variable height and width of the action area can be chosen to guide the geometry variation along the whole component. In this example, the shape of the section is preserved, which means that the width and height vary together in a user-defined range. The reference undeformed cross section is associated to a current value of the design variable equal to 0. The cross section of the C-pillar is assumed to vary its height and width in a user-defined range of $\mathcal{M}_1 = [-20, 20]$

mm. Analogously, the rear long member cross section is defined such that the correspondent design variable μ_2 varies in the range $\mathcal{M}_2 = [-10, 10]$ mm. The material variables are parametrised by selecting the property of interest in the optimisation tool and defining the ranges of variations. In this case, the variables μ_3 and μ_4, representing the thickness of the two parametric BiW components, are defined in the ranges $\mathcal{M}_3 = [1.4, 1.8]$ mm and $\mathcal{M}_4 = [0.5, 1.3]$ mm, respectively. Table 4.2 summarises the DVs definition. Each of the four parametric spaces is discretised by means of

Table 4.2: Design variables

	IDs	Component	Type	Current Value	Min	Max	Step
Geometry	μ_1	C-pillar	h, w	0	-20	20	5
	μ_2	Rear long member	h, w	0	-10	10	2.5
Material	μ_3	C-pillar	t	1.4	1.0	1.8	0.1
	μ_4	Rear long member	t	0.9	0.5	1.3	0.1

*h: section height; w: section width; t: element thickness (values are in millimetres).

nine equidistant nodes, which means that the total number of parametric combinations is given by $m_{tot} = 9^4 = 6,561$ different configurations. The next step requires a sampling of the input data (mass and stiffness matrices) for each combination. In order to do that, a list with all the 6,561 parametric combinations can be uploaded into the ANSA optimisation tool. Then, a Design of Experiments (DoE) study can be set up which automatically generates the input files in the format of the desired commercial software, which in this work is MSC-Nastran. By using a special Nastran language (DMAP), all the input files generated by the DoE study are run such that the mass and stiffness matrices are just assembled and stored, without solving any static or dynamic problem. The stored files are then uploaded into Matlab and expressed in the PGD format as explained in Sec. 2.3.3.

In order to finalise the pre-process phase, a data compression is performed to reduce the number of PGD terms. In this example, after performing the PGD compression imposing an accuracy of 10^{-8}, the number of terms needed to approximate the stiffness $\mathbf{K}^{\texttt{PGD}}(\boldsymbol{\mu})$ and mass $\mathbf{M}^{\texttt{PGD}}(\boldsymbol{\mu})$ matrices reduces respectively to $n_{\mathbf{K}} = 66$ and $n_{\mathbf{M}} = 20$, instead of the initial 6,561 terms.

4.3.2 Offline computation: nonintrusive PGD-NVH solver

This section finally shows the results of the proposed nonintrusive PGD-NVH solver. The first common step of the the PGD-IR and PGD-IPM algorithms is represented by the computation of the parametric rigid body matrix. As explained in the previous chapters, the rigid body modes can be computed as the kernel of the stiffness matrix $\mathbf{K}^{\texttt{PGD}}(\boldsymbol{\mu})$. In order to do that, the first step consists of choosing a set s of reference degrees of freedom that represents a set of isostatic constraints. Once the set of total n_d is partitioned into the s and the left l-set of degrees of freedom, the encapsulated PGD toolbox can be employed to solve the parametric linear system of equations introduced in Eq. (2.22) and obtain the PGD rigid body matrix $\boldsymbol{\Phi}^{\texttt{PGD}}(\boldsymbol{\mu})$ as kernel of the global stiffness matrix $\mathbf{K}^{\texttt{PGD}}(\boldsymbol{\mu})$. Finally, the parametric static and dynamic solutions can be computed.

Static analysis

The set up of the standard FE analysis to evaluate the global static torsional stiffness of a BiW structure is shown in Fig. 4.6 (left picture). The test consists of loading the BiW model with a couple of parallel and opposite forces applied at the front and rear shock towers of the car frame, such that the resulting torsional moment is equal to 1 Nm. The QoI of this problem is the equivalent torsional stiffness (ETS), which is calculated as a function of the front and back twisting rotations of the car body when the constant torque is applied, namely

$$\text{ETS} = \frac{1}{\alpha_{\text{AB}} + \alpha_{\text{CD}}} \times \frac{\pi}{180}. \tag{4.1}$$

Here the two twisting angles α_{AB} and α_{CD} represented in Fig. 4.6 (right picture) are given by the following relative vertical displacements

$$\alpha_{\text{AB}} = \frac{\left|u_z(A)\right| + \left|u_z(B)\right|}{L_{\text{AB}}}, \qquad \alpha_{\text{CD}} = \frac{\left|u_z(C)\right| + \left|u_z(D)\right|}{L_{\text{CD}}}, \tag{4.2}$$

where $u_z(P)$ denotes the displacement in the vertical z direction at point P and L_{PQ} denotes the distance between the points P and Q.

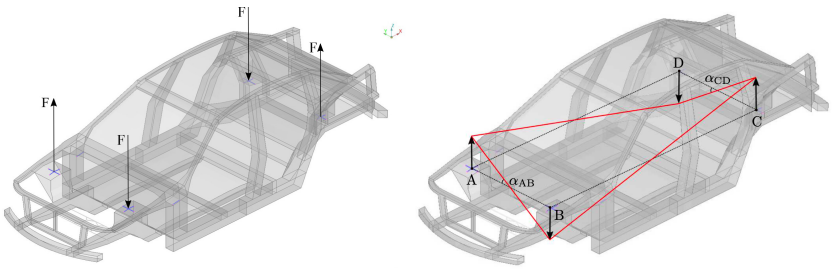


Figure 4.6: Load conditions used for the static global torsional stiffness analysis of the BiW structure (left). Illustration of the angles used to compute the ETS in Eq. (4.1) (right).

Now that the QoI has been defined, let us describe the steps of the PGD-IR algorithm. As shown in Fig. 4.3, once the rigid body matrix $\mathbf{\Phi}^{\texttt{PGD}}(\boldsymbol{\mu})$ has been computed, the remaining steps of the PGD-IR algorithm are performed by employing the encapsulated PGD routines. The final results is the parametric displacement vector:

$$\mathbf{U}^{\texttt{PGD}}(\boldsymbol{\mu}) = \sum_{i=1}^{N} \beta^i_{\mathbf{U}}\, \mathbf{U}^i\, u^i_1(\mu_1)\, u^i_2(\mu_2)\, u^i_3(\mu_3)\, u^i_4(\mu_4), \tag{4.3}$$

which contains all the solutions for every possible combination of the the four parameters. In this example, $N = 75$ terms were necessary in order to get the approximated solution. In particular, the solution was considered sufficiently accurate when the amplitude β^N of the last calculated PGD term was four order of magnitudes smaller than the amplitude β^1 associated to the first PGD term. In order to better understand the structure of a PGD solution, Fig. 4.7 shows the first three terms of the sum in Eq. (4.3).

The amplitude factors β^i, which are marked with a box for each term, are a measure of how much term i contributes to the final solution. The spatial terms $\mathbf{U}^i$ provide an illustration of the deformation induced by each PGD term. This deformation is scaled by the parametric functions, which take a different value depending on the chosen set of parameters $[\bar{\mu_1}, \bar{\mu_2}, \bar{\mu_3}, \bar{\mu_4}]$. Once the parametric solution $\mathbf{U}^{\texttt{PGD}}(\boldsymbol{\mu})$ is available, it can be particularised in real-time for any combination of the parameters and the ETS can be easily calculated according to Eq. (4.1).

In order to validate the proposed method, a comparison between the PGD solution and the standard FE solution was performed. Note that, for this simple example it is feasible to compute the FE solution at every parametric point. This allows to test the accuracy of the proposed PGD method. However, if the number of parameters increases, the FE approach becomes unfeasible, while PGD is still a viable option. For this purpose, the full-order problem was solved for all the 6,561 combinations by running the linear solver of MSC-Nastran in combination with the inertia relief option to circumvent the singularity of the stiffness matrix. As shown in Fig. 4.8, the results obtained with the PGD method are in perfect agreement with MSC-Nastran. The accuracy of the PGD with respect to the full-order computations is measured as the relative error between the PGD and Nastran ETS solutions in the $\mathcal{L}_2(\mathcal{M}_1 \times \mathcal{M}_2 \times \mathcal{M}_3 \times \mathcal{M}_4)$ norm, that is

$$\|\varepsilon_{\text{ETS}}\|_2 = \sqrt{\frac{\int_{\mathcal{M}_1}\int_{\mathcal{M}_2}\int_{\mathcal{M}_3}\int_{\mathcal{M}_4}\left(\text{ETS}^{\texttt{PGD}} - \text{ETS}^{\texttt{Nastran}}\right)^2 \, d\mu_1\, d\mu_2\, d\mu_3\, d\mu_4}{\int_{\mathcal{M}_1}\int_{\mathcal{M}_2}\int_{\mathcal{M}_3}\int_{\mathcal{M}_4}(\text{ETS}^{\texttt{Nastran}})^2 \, d\mu_1\, d\mu_2\, d\mu_3\, d\mu_4}}, \tag{4.4}$$

and in terms of the maximum error, calculated as the $\mathcal{L}_\infty(\mathcal{M}_1 \times \mathcal{M}_2 \times \mathcal{M}_3 \times \mathcal{M}_4)$ norm:

$$\|\varepsilon_{\text{ETS}}\|_\infty = \max_{\substack{\mu_1 \in \mathcal{M}_1, \mu_2 \in \mathcal{M}_2, \\ \mu_3 \in \mathcal{M}_3, \mu_4 \in \mathcal{M}_4}} \left(\frac{\left|\text{ETS}^{\texttt{PGD}} - \text{ETS}^{\texttt{Nastran}}\right|}{\text{ETS}^{\texttt{Nastran}}} \right), \tag{4.5}$$

leading to a maximum error equal to 5×10^{-3} and the $\mathcal{L}_2$-norm error equal to 8.83×10^{-4}. Finally, in order to better understand how changes in the

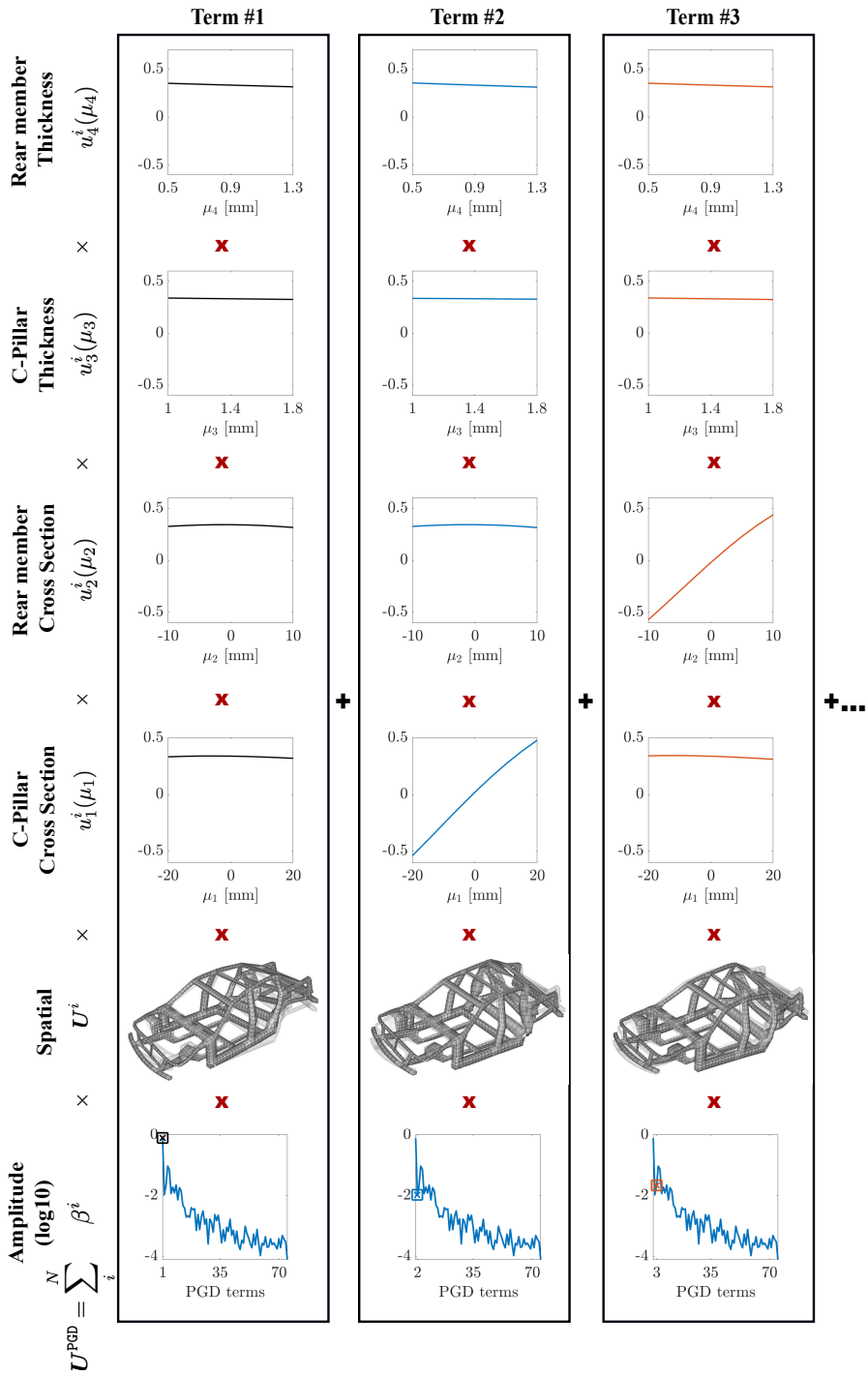


Figure 4.7: First three term of the generalised solution $\mathbf{U}^{\mathtt{PGD}}(\boldsymbol{\mu})$ of the parametric static analysis.

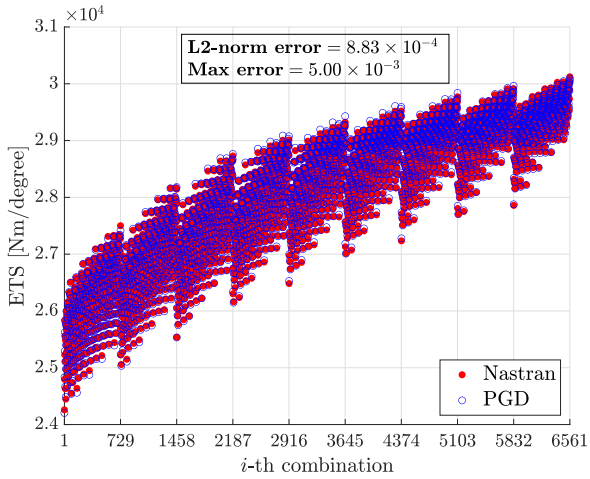


Figure 4.8: Equivalent torsional stiffness $ETS^{\texttt{PGD}}(\boldsymbol{\mu})$ with respect to the full-order FE solutions for every possible combination of the parameters.

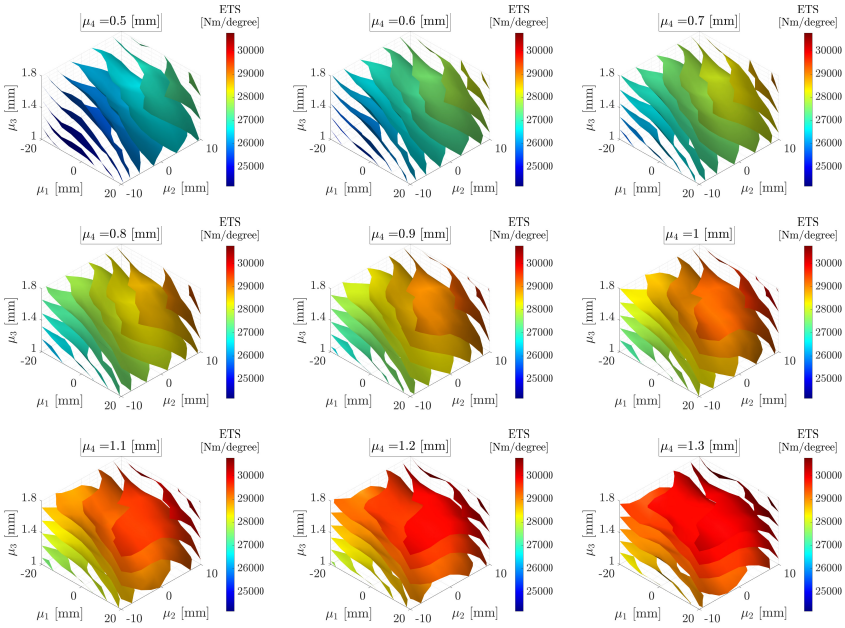


Figure 4.9: Isosurfaces showing the variation of ETS with respect to the parameters μ_1, μ_2 and μ_3. Each plot refers to one specific value of parameter μ_4.

defined design variables affect the static response of the car, Fig. 4.9 shows the variation of the ETS with respect to the four parameters in terms of isosurfaces. In particular, each plot shows the variation in the Cartesian space defined by the first three parameters (μ_1, μ_2, μ_3) when the forth parameter μ_4 is fixed. The plots show a substantial range of variation in the ETS, which is around $6,000$ Nm/degree. Moreover, it can be observed that the fourth parameter μ_4, corresponding to the thickness of the rear long members, has the biggest influence on the ETS variation. When μ_4 increases, the isosurfaces tend to be parallel to the (μ_1, μ_2) space, meaning that the variation due to changes in the cross sections gets reduced with the thickness increment.

Dynamic analysis

The mode shapes associated to the three smallest non-zero frequencies are computed in order to identify the first torsional mode of the BiW structure under analysis. In order to do that, the PGD-IPM eigensolver is used. The parametric input data of the problem is represented by the stiffness and mass matrices already sampled and expressed in the PGD format during the pre-process phase. Due to the unconstrained configuration, the matrix of rigid body modes $\mathbf{\Phi}^{\texttt{PGD}}(\boldsymbol{\mu})$ is essential for the computation of the subsequent non-rigid eigenmodes. As shown in Fig. 4.3, a parametric guess vector is chosen and the IPM system of equations is iteratively solved by employing the encapsulated PGD toolbox. At every iteration, the obtained parametric eigenvector is normalised to enforce unicity. The iteration stops when convergence is reached, that is when a quantity $E_{\boldsymbol{\phi}}$ is smaller than a user-defined tolerance. Here $E_{\boldsymbol{\phi}}$ is defined as:

$$E_{\boldsymbol{\phi}} = \frac{\left| \sum_{i=1}^{n_{\boldsymbol{\phi}^{\text{new}}}} \beta^i_{\boldsymbol{\phi}^{\text{new}}} - \sum_{j=1}^{n_{\boldsymbol{\phi}^{\text{old}}}} \beta^j_{\boldsymbol{\phi}^{\text{old}}} \right|}{\sum_{j=1}^{n_{\boldsymbol{\phi}^{\text{new}}}} \beta^j_{\boldsymbol{\phi}^{\text{old}}}}, \tag{4.6}$$

where $\beta^i_{\boldsymbol{\phi}^{\text{new}}}$ and $\beta^j_{\boldsymbol{\phi}^{\text{old}}}$ represent, respectively, the amplitudes of the PGD terms describing two eigenmodes obtained by two consecutive iterations. Once convergence is reached, the sought ith eigenvector is obtained in the PGD format:

$$\boldsymbol{\phi}_i^{\texttt{PGD}}(\boldsymbol{\mu}) = \sum_{i=1}^{N_{\boldsymbol{\phi}}} \beta^i_{\boldsymbol{\phi}}\, \boldsymbol{\phi}^i\, \phi_1^i(\mu_1)\, \phi_2^i(\mu_2)\, \phi_3^i(\mu_3)\, \phi_4^i(\mu_4). \tag{4.7}$$

The corresponding eigenfrequency can be calculated according to the Rayleigh quotient

$$\omega_i^{\texttt{PGD}}(\boldsymbol{\mu}) = \sqrt{(\boldsymbol{\phi}_i^{\texttt{PGD}})^\top \mathbf{K}^{\texttt{PGD}} \boldsymbol{\phi}_i^{\texttt{PGD}}}. \tag{4.8}$$

For this simplified BiW structure, it can be verified that the first and third non-rigid eigenvectors represent two different kind of bending modes, independently on the combination of the parameters. Similarly, the second non-rigid eigenvector always represents a torsional mode, which is the one of interest. Of course, when more complex models are analysed, the order of the modes can easily change with the parameters, so their identification would represent an important task.
Once the sought smallest torsional eigenvalue $\omega_t^{\texttt{PGD}}(\boldsymbol{\mu})$ is available, the corresponding torsional natural frequency $f_t^{\texttt{PGD}}(\boldsymbol{\mu})$ can be easily calculated as:

$$f_t^{\texttt{PGD}}(\boldsymbol{\mu}) = \frac{\omega_t^{\texttt{PGD}}(\boldsymbol{\mu})}{2\pi}. \tag{4.9}$$

As it was done in the static case, a comparison between the PGD solution and the standard FE solution of the modal analysis is studied. In this case, a full-order real eigenvalue analysis was performed in MSC-Nastran for all the 6,561 parametric combinations. Once again, as shown in Fig. 4.10, the results are in perfect agreement. The relative errors are calculated as in Eqs. 4.4 and 4.5 by substituting ETS with the frequency, leading to a $\mathcal{L}_2$-norm error equal to 1.86×10^{-4} and a maximum relative error of 1.70×10^{-3}. The plot also shows how small variations in the material

and geometric parameters of the two components can lead to variations of the torsional frequency in the range of 2-3 Hz, which might change the perception of vibration for the occupants of the vehicle.

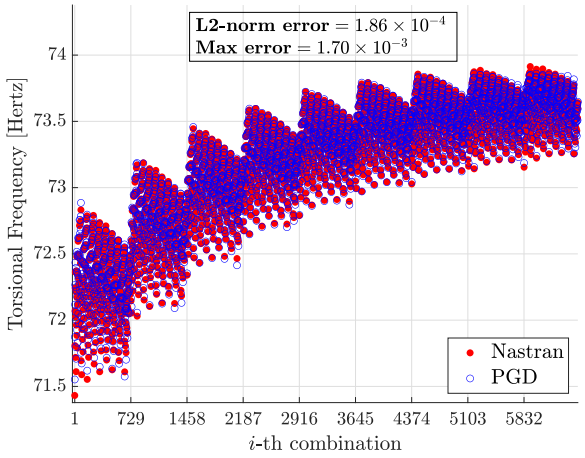


Figure 4.10: Torsional frequency $f_t^{\texttt{PGD}}(\boldsymbol{\mu})$ with respect to the full-order FE solutions for every possible combination of the parameters.

Finally, the variation of the torsional frequency is also depicted in terms of isosurface in Fig. 4.11. Each plots refers to a fixed value of μ_4 and shows the variation in the 3D cartesian space defined by (μ_1, μ_2, μ_3). Also in this case, the thickness of the rear long member (described by μ_4) proves to have the biggest influence on the QoI. The vertical character of the isosurfaces suggests that, differently from the static case, the thickness of the C-pillar (corresponding to μ_3) represents the less influencing parameter. It is worthy to emphasise that the described algorithm is solved by means of just one offline computation. The resulting eigenpairs represent the compact version of all the possible solutions for every value of the parameters.

4.3.3 Post-process: optimisation study and real-time visualisation

One of the most interesting features of the PGD method is that, once the offline process is finished, obtaining the solution for a given set of

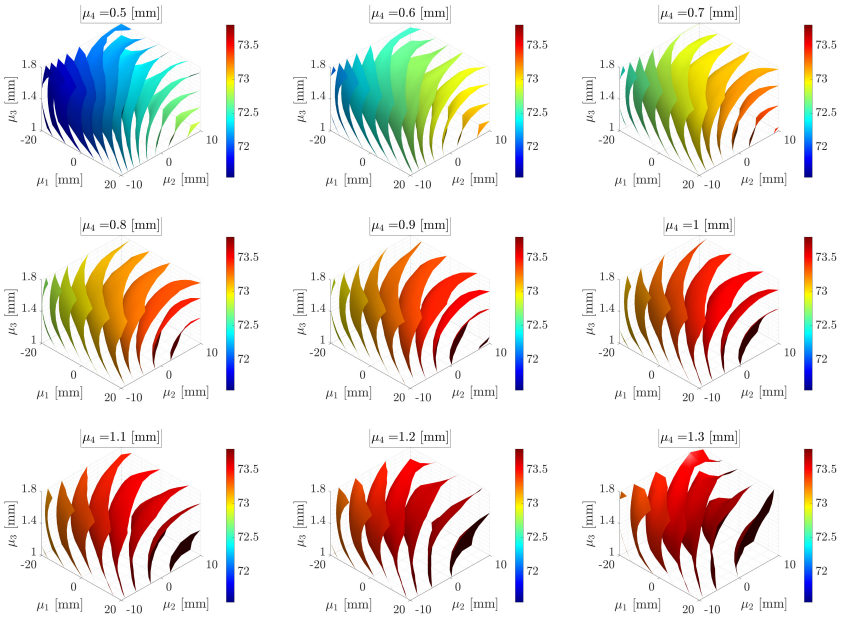


Figure 4.11: Isosurfaces showing the variation of the smallest torsional natural frequency f_2 with respect to the parameters μ_1, μ_2 and μ_3. Each plot refers to one specific value of parameter μ_4.

parametric values takes 10^{-4} seconds. This opens the possibility to perform efficient optimisation studies and visualise the results in real-time.
In order to show the potential of the proposed methodology, a multi-objective optimisation analysis is performed. The goal is to find the optimal combinations of the parameters such that the ETS and torsional frequency are maximised while the mass of the two parametric car components is minimised. Three objective functions are defined as:

$$\begin{cases} g_1(\boldsymbol{\mu}) &= M(\mu_1, \mu_2, \mu_3, \mu_4), \\ g_2(\boldsymbol{\mu}) &= \mathrm{ETS}(\mu_1, \mu_2, \mu_3, \mu_4), \\ g_3(\boldsymbol{\mu}) &= f_t(\mu_1, \mu_2, \mu_3, \mu_4). \end{cases} \tag{4.10}$$

where $M(\mu_1, \mu_2, \mu_3, \mu_4)$ represents the total mass of the C-pillars and the rear long members, depending on their variable geometries and thickness. Clearly, this quantity is strictly related to the production cost. $\text{ETS}(\mu_1, \mu_2, \mu_3, \mu_4)$ is the parametric output obtained by means of the proposed PGD-IR algorithm and $f_t(\mu_1, \mu_2, \mu_3, \mu_4)$ is parametric torsional frequency calculated by the PGD-IPM eigensolver. The explicit dependency of the three functions on the parameters permits to easily compute the Pareto front of the multiple objective functions by means of a genetic algorithm (GA).

Methods like the GA gained a lot of popularity in the last years in the context of optimisation studies. It belongs to the class of *gradient-free* optimisation techniques, which represents an alternative to the more classical *gradient-based* methods. In both cases, the user has to provide the models to evaluate the objective and constraint functions for any given set of design variables. The main difference between the two approaches is that gradient-based algorithms require to compute, when possible, the gradient of the objective and constraint functions. An advantage of gradient-based method is that they usually converge more efficiently to the optimal solution, as the gradients contain richer information about the function behaviour. The gradient-free algorithms, instead, are easier to set up and do not require continuity of the functions [95]. Recently, several works have been presented which use gradient-free method in the context of optimisation problems based on reduced order models [96–98].
The GA algorithm is part of a class of emerging gradient-free techniques, which are known as *evolutionary optimisation algorithms* ([99]). These approaches are a type of artificial intelligence methods inspired by optimisation processes that occur in nature. In particular, GAs are population based methods. The optimization starts with a set of design points (the population), which is then repeatedly modified at each iteration. At each step, the genetic algorithm randomly selects individuals from the current population and uses them to produce the next generation of individuals. Over successive generations, the population evolves toward an optimal solution.

In this work, the GA available in the Global Optimisation Toolbox released by Matlab is used to perform the optimisation study and obtain a Pareto front. The obtained Pareto points are shown in Fig. 4.12. According to the definition, they represent a trade-off between the objective functions, meaning that each point is considered optimal if no objective can be improved without sacrificing at least one other objective. The cloud of sampled points in the plot represents the mass and ETS coordinates corresponding to each of the 6,561 parametric combinations considered initially. It is clear that the optimisation study allows to drastically reduce the number of configurations which would be considered by the designers in the final decision-making process. Note that, in this example, the Pareto front was computed by assigning the same weight to both objective functions. Nevertheless, it is straightforward to obtain other fronts if the user wants to put more emphasis on one of the objective functions.

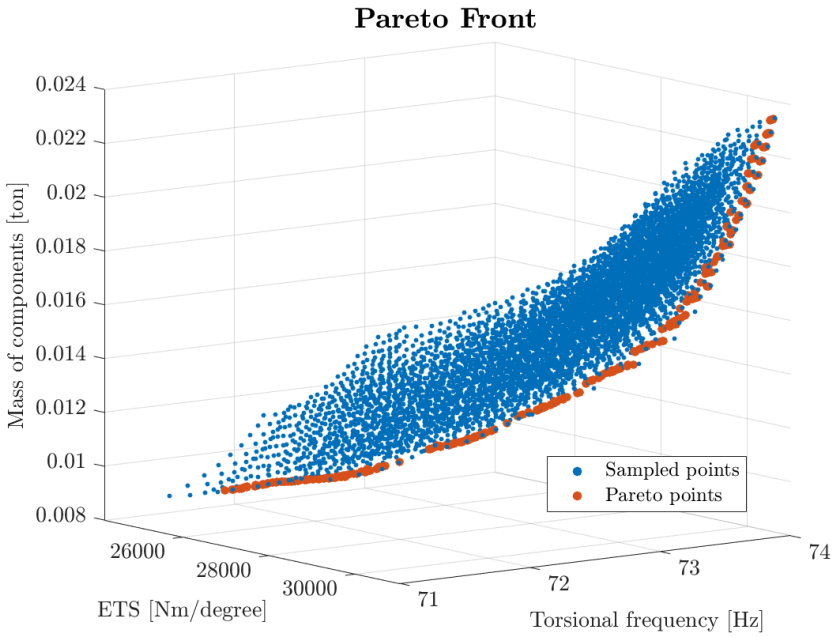


Figure 4.12: Multi-objective optimisation showing the Pareto front as a function of the objectives.

The ultimate goal of this work is the development of a computational tool which can be accessed by the designers to visualise the effect of the parametric changes on the global response of the BiW in real-time. At this purpose, a standalone desktop app was built by using the Matlab App Designer software. Fig. 4.13 shows the developed graphical user interface. An interactive version of the Pareto front depicted in Figs. 4.12 is shown in the app. The sliders in the red box of Fig. 4.13 allow to adjust the set of parameters to one of the 6,561 combinations given by the discretised parametric space. The corresponding point is simultaneously visualised in the Pareto Front plot, while the numerical values of the parameters, together with the mass and QoI coordinates, appear in the tables on the bottom right. Since the optimal solutions obtained through the multi-objective optimisation do not necessarily coincide with one of the 6,561 combinations represented by the cloud of points, an extra box is added which allows to scroll through a list of all the Pareto points (blue box in Fig. 4.13). Also in this case, the point information is reflected in the Pareto front plot and the numerical values are updated in the corresponding tables. If the user is interested in visualising all the Pareto points, it can be done by simply clicking on the "Display all optimal points" button. Finally, the static and dynamic deformation of the BiW corresponding to the selected parametric choice is updated in real-time in the two left top plots.
For a better understanding, the reader can finally refer to Fig. 4.14 to have a global overview of the proposed PGD-NVH method.

4.4 Chapter summary

A parametric global static and dynamic stiffness analysis is performed by means of a PGD-based methodology in order to optimise the NVH performance of a BiW structure characterised by material and geometric design variables. In particular two material and two shape design variables are studied, which correspond to the thickness and cross sections of the C-pillar and rear long member components. A coupling of the PGD with the IR method, which allows the static analysis of the BiW in its unconstrained configuration, is used to compute the parametric static equivalent torsional

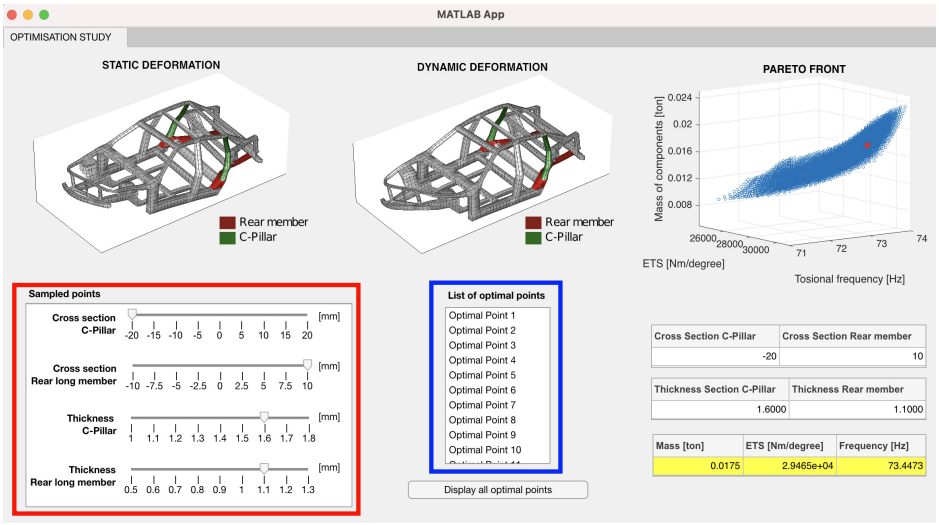


Figure 4.13: Graphical interface of the developed PGD-NVH app for the static and dynamic analyses.

stiffness (ETS) of the BiW. The parametric lowest torsional frequency under free vibration condition is obtained, instead, by employing a coupled PGD-IPM (inverse power method) eigensolver. A parametrised FE model formed by shell elements is built in the ANSA CAE pre-processor commercial software, which allows to deal with complex shape changes and prepare the input data of the problem in the separated PGD format. During an offline computation, the proposed PGD-NVH solver is executed in an in-house Matlab environment, acting as a *black-box*, such that a nonintrusive interaction with commercial FE packages is possible. Finally, a post-process of the obtained parametric results was presented. The accuracy of the method was measured by comparing the two quantities of interest (ETS and torsional frequency) with the corresponding full-order results, which resulted into a maximum relative error in the order of 10^{-3}. It is important to mention that the PGD results were obtained by performing only one offline computation for the static and dynamic problems and then particularising the results for any parametric combination in real-time. On the contrary, a total of 13,122 full-order simulations (6,561 for the static and 6,561 for the dynamic case) were needed to sample the results by means of standard methods. Thanks to explicit dependency of the QoIs on

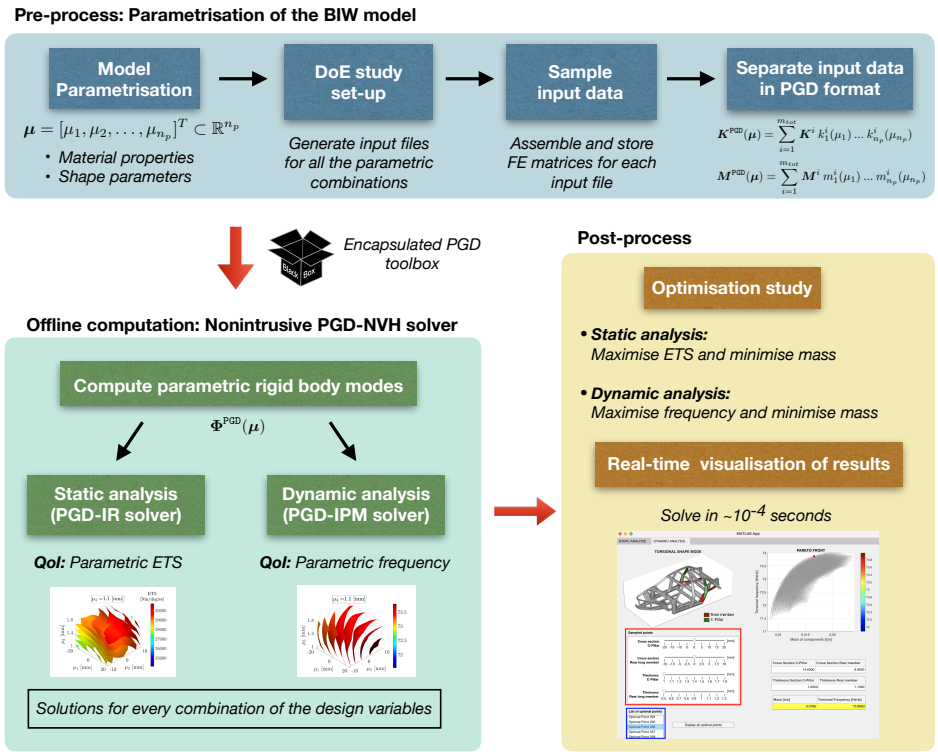


Figure 4.14: Overview of the proposed PGD-NVH method.

the parameters, a multi-objective optimisation study was performed by a genetic algorithm implemented in Matlab. The study allowed to identify a set of optimal Pareto points which drastically reduced the combination of design variables to take into account in the final decision-making process. Finally, a graphical interface app was developed by using the Matlab App Designer software, providing an interactive visualisation of the results, such that the designers can check in real-time the effects of variables on the global static and dynamic behaviour of the BiW structure. The developed app is just an example of the potential of this method. In fact, the information contained in the app could be modified and adapted to the needs of the specific problem, representing the kind of support that the industry urgently needs to optimise the development process.

5 Conclusions and future developments

5.1 Conclusions

This thesis is one of the 15 projects implemented in the context of the ProTechTion (Industrial decision-making on complex **Pro**duction **Tech**nologies supported by simula**Tion**-based engineering) programme, funded by the European Union's EU Framework Programme for Research and Innovation Horizon 2020.
All the projects are supported by a consortium of eight renowned universities and twelve industrial partners from all over Europe. In particular, this thesis was supervised by the Universitat Politècnica de Catalunya (UPC) and Swansea University (SU), in collaboration with SEAT S.A. as industrial partner. The overall objective of ProTechTion is to bridge the gap between the full potential of multi-disciplinary SBE technology and the methodologies available in industrial environments to perform complex production processes.

This thesis is dedicated to the development of a new methodology to improve the design optimisation process of an automotive structure with respect to its noise and vibration performance, which is one of the most complex targets to accomplish during the vehicle development process. In order to achieve the proposed objective, a ROM technique was developed which allows to perform the linear static and dynamic global stiffness analysis of automotive structures characterised by *parametric* material and geometric properties, which represent the design variables typically unknowns during the design process.

The proposed technique, based on the PGD reduced order method, allows to efficiently explore an arbitrary large design space and perform real time sensitivity analysis with respect to the different design parameters. The key idea behind the method is to invest an initial affordable computational effort, already during the preliminary phase of the design process, to built an accurate surrogate model of the problem. The obtained reduced model is then used to perform efficient optimisation studies and real-time evaluations, which serve as support to designers during the intricate decision-making process.
Several challenges were addressed throughout the chapters of this thesis to achieve the objectives announced in Section 1.3. As a summary, the main contributions of each chapter are presented in this final section.

- ***Chapter 2: Parametric static analysis of an unconstrained structure.***
  This chapter introduces the problem related to the static analysis of an unconstrained structure, which in the industrial context is usually solved by means of the IR problem. After a review or IR in the non-parametric case, an extension of the problem to the parametric format is presented. The PGD method is introduced as the most appealing ROM for the resolution of the resulting multidimensional problem. More specifically, the encapsulated PGD toolbox is employed to sequentially solve the parametric algebraic operations involved in the multidimensional version of the IR algorithm. A *hybrid* version of the PGD method is developed, which allows to deal with complex shape parametrisations, overcoming one of the main limitation of the standard PGD method. An academic numerical example finally shows the properties of the developed PGD-IR method. In particular, the structure of the parametric PGD solution is described and compared with the corresponding full-order FE results. Moreover, the variation of a specific QoI in the design space is explored during a real time post-process step.

- ***Chapter 3: Parametric solutions in structural dynamic.***
  In this chapter, the PGD method is coupled with the IPM algorithm in order to perform the modal analysis of a 3D structure characterised by material and geometric parameters. A review of the standard IPM method is carried out to show the structure of the iterative algorithm. As it is well known, the method allows to extract the smallest (in magnitude) natural frequencies and corresponding vibrational mode shapes, which represent the typical QoIs in structural dynamic problems. The coupled PGD-IPM is developed following the rationale already introduced in Chapter 2. A new deflation technique is developed to compute subsequent parametric eigenmodes. The encapsulated PGD toolbox is a key asset to perform several parametric algebraic operations included in the algorithm. Moreover, a new PGD-based routine is derived and implemented to calculate the square root of a parametric object, which is essential in the PGD-IPM framework. The same 3D structure used in the static problem is tested to validate the proposed method. The parametric results in terms of the three smallest natural frequencies and corresponding mode shapes were presented and discussed. The comparison with the corresponding full-order FE solutions confirm the accuracy of the PGD solution and highlighted the advantages of the proposed method.

- ***Chapter 4: Industrial application.***
  In this chapter, the developed methodology is finally applied to an industrial case of study, which is the NVH analysis of a simplified BIW structure. The main QoIs of a standard NVH study are reviewed, which basically consist of evaluating the static and dynamic global stiffness of the body structure in order to extract specific indicators of the noise and vibration properties. The parametric definition of the problem is presented and the PGD-IR and PGD-IPM algorithms, described in the two previous chapters, are unified in a global algorithm. The problem is finally solved in three main steps: a pre-process step; one offline computation; a post-process step. The

pre-process phase includes the construction of a parametrised FE model of the BIW structure by means of a morphing and optimisation tool provided by a commercial software. The hybrid PGD approach is used to assemble the input quantities (without solving the problem) by means of the commercial software. Then, the sampled matrices are imported in the in-house code and expressed in the PGD format. Afterwards, with one offline computation, the static and dynamic solutions were obtained in the compact parametric PGD format, which contains the solutions for every possible combination of the design variables. Finally, the obtained computational vademecum is used to perform an efficient multi-objective optimisation study by means of a gradient-free genetic algorithm. The study aims at finding a set of optimal design configurations that optimise the static and dynamic indicators while minimising the mass (thus the cost) of the parametric car components. Finally, a graphical interface app is developed for the real-time visualisation of the results. The app could be uploaded on light computational devices (such as smartphones and tables) such that designers could use it during the preliminary design process in order to explore the design space in real time a take decisions.

5.2 Future developments

The proposed method opens new perspectives for future research. Some ideas are described below.

- ***Increase the number of design variables.***
  Real industrial design problems are usually characterised by a high number of design variables. In this work, a maximum of four parameters were considered, which already represents a good result if compared to the majority of PGD applications available in the literature. It is worthy to mention that, conceptually, the PGD does not present any limitation in terms of number of parameters to be included. However, the iterative scheme, used to solve the nonlinear problem of finding the best rank-one approximation of each PGD

term, might have convergence problems when the number of parameters increases. An interesting future work would be to investigate how to solve this issue.

- ***Compute a Separated Response Surface (SRS) of the PGD input data.***
  As stated in Chapter 4, the proposed nonintrusive and hybrid PGD approach requires a sampling of the input data in the whole design space, that is assembling the input data for every possible combination of the design variables. Each assembly has a negligible computational cost if compared to the cost of full-order static and dynamic simulation. Moreover, this task can be easily parallelised. However, sampling a multidimensional space with an uniform distribution in every parametric dimension is not optimal, especially in view of an increment in the number of design variables involved. To improve the pre-process phase of the proposed method, alternative statistical sampling methods can be employed, such as the Latin hypercube [100] or Halton sequence [101] methods, which allow to generate a near-random sample of parameter values from a multidimensional distribution. Once the sampling in the quasi-random set of representative parametric combinations is available, the idea is to compute a Separated Response Surface (SRS) which defines the required separated approximation of the input data. As introduced in [102] in the context of crashworthiness uncertainty quantification to approximate the solution (and not the input data), the algorithm employed to compute the SRS is based on the philosophy of the least-squares PGD approximation. In the context of this thesis, some preliminary work has been set up in order to test the PGD-SRS sampling method. Although this has already shown interesting properties, the obtained results have been considered not sufficiently accurate for the analysed case of study. The reason might be connected to some choices required by the algorithm in order to find the separated representation. First, a penalty factor might be defined in order to

enforce the smoothness of the approximated solution, which is not automatically guaranteed otherwise. Moreover, a scaling of the data points is needed in order to assign the same weight to the components of the input matrix (or vector) depending on design variables which might belong to different order of magnitudes. It is reasonable to think that, with a small extra effort in calibrating these quantities, the method could reach the expected accuracy, allowing to further alleviate the pre-process phase of the proposed method.

- ***Translation into Python programming language.***
  As it is well known, Python is gaining increasing popularity in every industrial sector. As a main advantage with respect to Matlab, Python is a free, open-source software which attracts a much wider community than commercial closed-source software. As already mentioned, the ultimate goal of this work is to provide a graphical interface app which allows to visualise the results in real-time and support the designers, independently on the language of implementation of the PGD code. However, owning the tools and the expertise to adapt the proposed PGD method to different cases of study represents the real objective. As a consequence, in order to make the proposed methodology more appealing for an industrial application, a translation of the developed Matlab code into the preferred Python platform is planned.

- ***Apply the method to other cases of study.***
  This work has presented a general methodology for the linear static and dynamic analysis of a parametric engineering structure. The automotive industry is just one of the possible areas of applications. A natural extension to other fields such as aeronautic, aerospace or civil engineering is easy to imagine. Moreover, especially in the context of the modal analysis, a wide range of other apparently non-related applications might be approached by the proposed PGD-based strategy. To give some examples, the modal analysis is employed in fields like food process engineering to investigate the quality of fruits

[103, 104], in the design of musical instruments like violins [105], or to test ultrasonic medical instruments [106]. In this perspective, the proposed technique has all the characteristics to have a strong impact on a potentially wide scale.

A Analytical approach to separate input quantities

The analytical technique follows the standard isoparametric concept widely used in FE formulations. This implies that a mapping function $\Psi(\theta)$ which transforms the reference domain Ω into the geometrically parametrised domain $\Omega(\theta)$ has to be defined, such that

$$\Psi(\theta) : \Omega \rightarrow \Omega(\theta) \tag{A.1}$$

$$\mathbf{X} \mapsto \mathbf{x} = \Psi(\mathbf{X}, \theta), \tag{A.2}$$

where $\mathbf{X}$ represents the coordinate system associated to the reference domain Ω, while $\mathbf{x}$ describes the modified domain $\Omega(\theta)$. According to the standard procedure, in order to transform the integrals involved in the weak formulation from the parametrized domain to the reference one, the Jacobian matrix $\mathbf{J}_\Psi(\theta) = \partial\mathbf{x}/\partial\mathbf{X}$ associated to the mapping $\Psi(\theta)$ has to be introduced. Then, the discretized definition of the stiffness and mass matrices at the reference element level Ω_e (already defined in Sec. 2.1.2) becomes

$$\mathbf{K}^e = \int_{\Omega_e} \mathbf{B}^\mathsf{T} \mathbf{J}_\Psi(\theta)^{-\mathsf{T}} \boldsymbol{\mathcal{C}}\, \mathbf{J}_\Psi(\theta)^{-1} \mathbf{B}\, \det(\mathbf{J}_\Psi(\theta))\, \mathrm{d}\Omega, \tag{A.3}$$

$$\mathbf{M}^e = \int_{\Omega_e} \mathbf{N}^\mathsf{T} \mathbf{N}\, \det(\mathbf{J}_\Psi(\theta))\, \mathrm{d}\Omega. \tag{A.4}$$

The modification of the stiffness matrix formulation caused by the introduction of the inverse of the Jacobian matrix $\mathbf{J}_\Psi^{-1}$, leads to the first important limitation of the analytical method. In fact, it is well known that even if the mapping function and correspondent Jacobian can be written

in a separated form, the inverse matrix $\mathbf{J}_{\Psi}^{-1}(\theta)$ is in general not separable due to the presence of $\det(\mathbf{J}_{\Psi}(\theta))$ at the denominator. As a consequence, an explicit dependency of the stiffness matrix on the geometric parameter cannot be found and other methods should be employed to find a separated expression of it. This difficulty is discussed in detail in [107]. An alternative mixed formulation has also been recently considered to circumvent this difficulty in a discontinuous Galerkin framework [108].

B PGD square root

This section presents the algorithm devised to implement the square root operation in the encapsulated PGD toolbox.

Recall that PGD combines a greedy strategy (computing sequentially the rank-one terms) and, for each rank-one problem, an iterative alternated directions scheme looping in all the parametric dimensions (for $\gamma = 1, 2, \dots, n_p$) assuming that all the sectional information is known for any other sectional dimension $j \neq \gamma$. The core of the algorithm is summarised in the expression used to iterate (how to compute the sectional dimension γ, assuming that the rest of the sectional dimensions are known). The goal of this section is to present this expression for the square root operation. In this particular case, the expression for the first term ($i = 1$) is different than for the subsequent terms ($i = 2, 3, \dots$).

The input is a scalar parametric value $X(\boldsymbol{\mu})$ expressed in a separated format

$$X(\boldsymbol{\mu}) = \sum_{i=1}^{n_X} \beta_X^i \prod_{j=1}^{n_p} F_X^{j,i}(\mu_j), \tag{B.1}$$

which is assumed to be nonnegative for every value of $\boldsymbol{\mu}$. Note that, in accordance with the previous developments, the amplitude β_X^i of each term arises from a normalisation of the *parametric sectional* modes $F_X^{j,i}(\mu_j)$.

The aim is to compute $Y(\boldsymbol{\mu}) = \sqrt{X(\boldsymbol{\mu})}$, that is $Y(\boldsymbol{\mu})$, nonnegative and such that $Y(\boldsymbol{\mu})^2 = X(\boldsymbol{\mu})$. The solution is written in the form of a sum of n_Y rank-one terms, namely

$$Y(\boldsymbol{\mu}) = \sum_{i=1}^{n_Y} \beta_Y^i \prod_{j=1}^{n_p} F_Y^{j,i}(\mu_j). \tag{B.2}$$

First rank-one term ($i = 1$).

The standard greedy approach in the PGD consists of computing the rank-one terms sequentially. Let's denote the first rank-one term, prior to the normalisation of the sectional modes that bring out amplitudes by

$$\tilde{Y}(\boldsymbol{\mu}) = \prod_{j=1}^{n_p} F_{\tilde{Y}}^{j}(\mu_j). \tag{B.3}$$

The idea is to find $\tilde{Y}$ such that

$$(\tilde{Y})^2 = X. \tag{B.4}$$

An alternated directions strategy is adopted that consists of computing the sectional mode $F_{\tilde{Y}}^{\gamma}(\mu_\gamma)$ assuming that the rest of the modes $F_{\tilde{Y}}^{j}(\mu_j)$ are known for $j \neq \gamma$, and looping for $\gamma = 1, 2, \ldots, n_p$. Given γ, the problem is solved multiplying by a weighting function

$$\delta\tilde{Y}(\boldsymbol{\mu}) = \delta F^{\gamma}(\mu_\gamma) \prod_{j\neq\gamma} F_{\tilde{Y}}^{j}(\mu_j) \tag{B.5}$$

and integrating in all parametric dimensions but γ, that is

$$\int \cdots \int_{\mu_j \neq \mu_\gamma} \left(\tilde{Y}\right)^2 \delta\tilde{Y}(\boldsymbol{\mu}) \prod_{j\neq\gamma} d\mu_j = \int \cdots \int_{\mu_j \neq \mu_\gamma} X \delta\tilde{Y}(\boldsymbol{\mu}) \prod_{j\neq\gamma} d\mu_j \tag{B.6}$$

Using expressions (B.3) and (B.5) in (B.6) results in

$$\begin{aligned} &\underbrace{\left[\prod_{j\neq\gamma} \int_{\mu_j} (F_{\tilde{Y}}^{j}(\mu_j))^3 d\mu_j\right]}_{=:\beta_\star} (F_{\tilde{Y}}^{\gamma}(\mu_\gamma))^2 \delta F^{\gamma}(\mu_\gamma) = \\ &\underbrace{\sum_{i=1}^{n_X} \beta_X^i \left[\prod_{j\neq\gamma} \int_{\mu_j} F_X^{j,i}(\mu_j) F_{\tilde{Y}}^{j}(\mu_j)) d\mu_j\right] F_X^{i,\gamma}(\mu_\gamma)}_{=:R(\mu_\gamma)} \delta F^{\gamma}(\mu_\gamma) \end{aligned} \tag{B.7}$$

for all $\delta F^{\gamma}(\mu_\gamma)$. Note that scalar $\beta_\star$ and function $R(\mu_\gamma)$ introduced in (B.7) are computable at this stage of the alternated directions algorithm. Thus, the resulting sectional mode $F_{\tilde{Y}}^{\gamma}(\mu_\gamma)$ is updated in this iteration using the

following expression:

$$F^{\gamma}_{\tilde{Y}}(\mu_\gamma) = \sqrt{R(\mu_\gamma)/\beta_\star}, \tag{B.8}$$

which defines the core of the PGD square root algorithm (for the first term, $i = 1$).

Subsequent rank-one terms ($i = 2, 3, \dots$).

Now, assume that expression (B.2) for Y is obtained for some n_Y and it has to be enhanced adding the $n_Y + 1$ term. This rank-one term is denoted, prior to normalisation of the sectional contributions, by

$$\Delta Y(\boldsymbol{\mu}) = \prod_{j=1}^{n_p} F^j_{\Delta Y}(\mu_j). \tag{B.9}$$

The algorithm proposed is based in the fact that in the PGD greedy algorithm, the first modes produce a fair approximation of $\sqrt{X(\boldsymbol{\mu})}$ and therefore, in order to compute the next term, a linearisation of the equation is sufficient to improve the approximation. The following approximated equation for ΔY results from neglecting the quadratic term $(\Delta Y)^2$ in front of $Y\Delta Y$, namely

$$(Y + \Delta Y)^2 = X \quad \text{results in} \quad Y\Delta Y \approx \frac{1}{2}(X - Y^2). \tag{B.10}$$

Once again, the unknowns $F^j_{\Delta Y}$, for $j = 1, 2, \dots, n_p$ are computed with the standard alternate directions strategy, that is assuming that $F^j_{\Delta Y}$ are known for $j \neq \gamma$, multiplying by a variation δY, and integrating in all sectional dimensions but γ, analogously to (B.6). The variation δY reads

$$\delta Y(\boldsymbol{\mu}) = \delta F^{\gamma}(\mu_\gamma) \prod_{j \neq \gamma} F^j_{\Delta Y}(\mu_j). \tag{B.11}$$

Thus, in $F^{\gamma}_{\Delta Y}$ is computed such that, for all $\delta F^{\gamma}(\mu_\gamma)$

$$\sum_{i=1}^{n_Y} \beta_Y^i \overbrace{\left[\prod_{j\neq\gamma}\left[\int_{\mu_j} F_Y^{j,i}(\mu_j)\left(F^j_{\Delta Y}(\mu_j)\right)^2 d\mu_j\right]\right]}^{=:\beta^i_\star} F^{\gamma}_{\Delta Y}(\mu_\gamma)F_Y^{i,\gamma}(\mu_\gamma)\delta F^{\gamma}(\mu_\gamma) = \underbrace{\left[\int\cdots\int_{\mu_j\neq\mu_\gamma}\frac{1}{2}(X-Y^2)\left(\prod_{j\neq\gamma}F^j_{\Delta Y}(\mu_j)\,d\mu_j\right)\right]}_{=:R(\mu_\gamma)}\delta F^{\gamma}(\mu_\gamma) \quad \text{(B.12)}$$

Thus, the expression defining the core of the alternated directions algorithm for the subsequent terms $i = 2, 3, \ldots$ (and analogous to (B.8) for $i = 1$) reads

$$F^{\gamma}_{\Delta Y}(\mu_\gamma) = R(\mu_\gamma)/\left(\sum_{i=1}^{n_Y}\beta_Y^i\beta^i_\star F_Y^{i,\gamma}(\mu_\gamma)\right). \quad \text{(B.13)}$$

:

Bibliography

[1] D. Ullman. *The Mechanical Design Process*. McGraw-Hill series in mechanical engineering. McGraw-Hill Education, 2010. ISBN: 9780072975741.

[2] George Ellwood Dieter, Linda C Schmidt, et al. *Engineering design*. McGraw-Hill Higher Education Boston, 2009.

[3] Sharon C Glotzer. *International assessment of research and development in simulation-based engineering and science*. World Scientific, 2011.

[4] Alfio Quarteroni, Gianluigi Rozza, et al. *Reduced order methods for modeling and computational reduction*. Vol. 9. Springer, 2014.

[5] Alexander I.J. Forrester and Andy J. Keane. "Recent advances in surrogate-based optimization". In: *Progress in Aerospace Sciences* 45.1-3 (2009), pp. 50–79.

[6] T.W. Simpson, J.D. Poplinski, P. N. Koch, and J.K. Allen. "Metamodels for Computer-based Engineering Design: Survey and recommendations". In: *Engineering with Computers* 17.2 (2001), pp. 129–150.

[7] Roland W. Freund. "Model reduction methods based on Krylov subspaces". In: *Acta Numerica* 12 (2003), pp. 267–319.

[8] G. Rozza, D. B. P. Huynh, and A. T. Patera. "Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations". In: *Archives of Computational Methods in Engineering* 15.3 (2007), pp. 1–47.

[9] Anindya Chatterjee. "An introduction to the proper orthogonal decomposition". In: *Current science* (2000), pp. 808–817.

[10] A. Ammar, B. Mokdad, F. Chinesta, and R. Keunings. "A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids". In: *Journal of Non-Newtonian Fluid Mechanics* 139.3 (2006), pp. 153–176.

[11] Francisco Chinesta, Pierre Ladeveze, and Elías Cueto. "A Short Review on Model Order Reduction Based on Proper Generalized Decomposition". In: *Archives of Computational Methods in Engineering* 18.4 (2011), pp. 395–404.

[12] Francisco Chinesta, Roland Keunings, and Adrien Leygue. *The Proper Generalized Decomposition for advanced numerical simulations*. Springer International Publishing, 2014.

[13] A. Dumon, C. Allery, and A. Ammar. "Proper general decomposition (PGD) for the resolution of Navier–Stokes equations". In: *Journal of Computational Physics* 230.4 (2011), pp. 1387–1407.

[14] C. Leblond and C. Allery. "A priori space–time separated representation for the reduced order modeling of low Reynolds number flows". In: *Computer Methods in Applied Mechanics and Engineering* 274 (2014), pp. 264–288.

[15] Rubén Ibáñez, Emmanuelle Abisset-Chavanne, Francisco Chinesta, and Antonio Huerta. "Simulating squeeze flows in multiaxial laminates: towards fully 3D mixed formulations". In: *International Journal of Material Forming* 10.5 (2016), pp. 653–669.

[16] Pedro Díez, Sergio Zlotnik, and Antonio Huerta. "Generalized parametric solutions in Stokes flow". In: *Computer Methods in Applied Mechanics and Engineering* 326 (2017), pp. 223–240.

[17] R. García-Blanco, D. Borzacchiello, F. Chinesta, and P. Diez. "Monitoring a PGD solver for parametric power flow problems with goal-oriented error assessment". In: *International Journal for Numerical Methods in Engineering* 111.6 (2017), pp. 529–552.

[18] Matteo Giacomini, Luca Borchini, Ruben Sevilla, and Antonio Huerta. "Separated response surfaces for flows in parametrised domains: comparison of a priori and a posteriori PGD algorithms". In: *arXiv preprint arXiv:2009.02176* (2020).

[19] Ch. Ghnatios, F. Masson, A. Huerta, A. Leygue, E. Cueto, and F. Chinesta. "Proper Generalized Decomposition based dynamic data-driven control of thermal processes". In: *Computer Methods in Applied Mechanics and Engineering* 213-216 (2012), pp. 29–41.

[20] José V. Aguado, Antonio Huerta, Francisco Chinesta, and Elías Cueto. "Real-time monitoring of thermal processes by reduced-order modeling". In: *International Journal for Numerical Methods in Engineering* 102.5 (2014), pp. 991–1017.

[21] Antonio Huerta, Enrique Nadal, and Francisco Chinesta. "Proper generalized decomposition solutions within a domain decomposition strategy". In: *International Journal for Numerical Methods in Engineering* 113.13 (2018), pp. 1972–1994.

[22] J. P. Moitinho de Almeida. "A basis for bounding the errors of proper generalised decomposition solutions in solid mechanics". In: *International Journal for Numerical Methods in Engineering* 94.10 (2013), pp. 961–984.

[23] Jonatha Reis, J. P. Moitinho de Almeida, Pedro Díez, and Sergio Zlotnik. "Error estimation for proper generalized decomposition solutions: A dual approach". In: *International Journal for Numerical Methods in Engineering* 121.23 (2020), pp. 5275–5294.

[24] Jonatha Reis, J. P. Moitinho de Almeida, Pedro Díez, and Sergio Zlotnik. "Error estimation for proper generalized decomposition solutions: Dual analysis and adaptivity for quantities of interest". In: *International Journal for Numerical Methods in Engineering* 122.3 (2020), pp. 752–776.

[25] Eugenio Giner, Brice Bognet, Juan J. Ródenas, Adrien Leygue, F. Javier Fuenmayor, and Francisco Chinesta. "The Proper Generalized Decomposition (PGD) as a numerical procedure to solve 3D cracked

plates in linear elastic fracture mechanics". In: *International Journal of Solids and Structures* 50.10 (2013), pp. 1710–1720.

[26] Hasini Garikapati, Sergio Zlotnik, Pedro Díez, Clemens V. Verhoosel, and E. Harald van Brummelen. "A Proper Generalized Decomposition (PGD) approach to crack propagation in brittle materials: with application to random field material properties". In: *Computational Mechanics* 65.2 (2019), pp. 451–473.

[27] Sergio Zlotnik, Pedro Díez, David Modesto, and Antonio Huerta. "Proper generalized decomposition of a geometrically parametrized heat problem with geophysical applications". In: *International Journal for Numerical Methods in Engineering* 103.10 (2015), pp. 737–758.

[28] Marianna Signorini, Sergio Zlotnik, and Pedro Díez. "Proper generalized decomposition solution of the parameterized Helmholtz problem: application to inverse geophysical problems". In: *International Journal for Numerical Methods in Engineering* 109.8 (2016), pp. 1085–1102.

[29] Alberto Sibileau, Alberto García-González, Ferdinando Auricchio, Simone Morganti, and Pedro Díez. "Explicit parametric solutions of lattice structures with proper generalized decomposition (PGD)". In: *Computational Mechanics* 62.4 (2018), pp. 871–891.

[30] G. Barroso, M. Seoane, A.J. Gil, P.D. Ledger, M. Mallett, and A. Huerta. "A staggered high-dimensional Proper Generalised Decomposition for coupled magneto-mechanical problems with application to MRI scanners". In: *Computer Methods in Applied Mechanics and Engineering* 370 (2020), p. 113271.

[31] Francisco Chinesta, Adrien Leygue, Felipe Bordeu, Jose Vicente Aguado, Elías Cueto, David González, Iciar Alfaro, Amine Ammar, and Antonio Huerta. "PGD-based computational vademecum for efficient design, optimization and control". In: *Archives of Computational Methods in Engineering* 20.1 (2013), pp. 31–59.

[32] Adrien Leygue and Erwan Verron. "A first step towards the use of proper general decomposition method for structural optimization". In: *Archives of Computational Methods in Engineering* 17.4 (2010), pp. 465–472.

[33] B. Bognet, F. Bordeu, F. Chinesta, A. Leygue, and A. Poitou. "Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity". In: *Computer Methods in Applied Mechanics and Engineering* 201-204 (2012), pp. 1–12.

[34] Thomas Heuzé, Adrien Leygue, and Guillaume Racineux. "Parametric modeling of an electromagnetic compression device with the proper generalized decomposition". In: *International Journal of Material Forming* 9.1 (2016), pp. 101–113.

[35] Amaury Courard, David Néron, Pierre Ladevèze, and Ludovic Ballere. "Integration of PGD-virtual charts into an engineering design process". In: *Computational Mechanics* 57.4 (2016), pp. 637–651.

[36] Sergio Zlotnik, Pedro Díez, David Modesto, and Antonio Huerta. "Proper generalized decomposition of a geometrically parametrized heat problem with geophysical applications". In: *International Journal for Numerical Methods in Engineering* 103.10 (2015), pp. 737–758.

[37] Ruben Sevilla, Sergio Zlotnik, and Antonio Huerta. "Solution of geometrically parametrised problems within a CAD environment via model order reduction". In: *Computer Methods in Applied Mechanics and Engineering* 358 (2020), p. 112631.

[38] Fehmi Cirak, Michael Ortiz, and Peter Schröder. "Subdivision surfaces: a new paradigm for thin-shell finite-element analysis". In: *International Journal for Numerical Methods in Engineering* 47.12 (2000), pp. 2039–2072.

[39] Ruben Sevilla, Sonia Fernández-Méndez, and Antonio Huerta. "NURBS-Enhanced Finite Element Method (NEFEM)". In: *Archives of Computational Methods in Engineering* 18.4 (2011), pp. 441–484.

[40] Ruben Sevilla and Antonio Huerta. "HDG-NEFEM with Degree Adaptivity for Stokes Flows". In: *Journal of Scientific Computing* 77.3 (2018), pp. 1953–1980.

[41] A. Leygue and E. Verron. "A First Step Towards the Use of Proper General Decomposition Method for Structural Optimization". In: *Archives of Computational Methods in Engineering* 17.4 (2010), pp. 465–472.

[42] Thomas Heuzé, Adrien Leygue, and Guillaume Racineux. "Parametric modeling of an electromagnetic compression device with the proper generalized decomposition". In: *International Journal of Material Forming* 9.1 (2015), pp. 101–113.

[43] Amaury Courard, David Néron, Pierre Ladevèze, and Ludovic Ballere. "Integration of PGD-virtual charts into an engineering design process". In: *Computational Mechanics* 57.4 (2015), pp. 637–651.

[44] L. Chamoin and H.P. Thai. "Certified real-time shape optimization using isogeometric analysis, PGD model reduction, and a posteriori error estimation". In: *International Journal for Numerical Methods in Engineering* 119.3 (2019), pp. 151–176.

[45] Xi Zou, Michele Conti, Pedro Díez, and Ferdinando Auricchio. "A nonintrusive proper generalized decomposition scheme with application in biomechanics". In: *International Journal for Numerical Methods in Engineering* 113.2 (2018), pp. 230–251.

[46] Vasileios Tsiolakis, Matteo Giacomini, Ruben Sevilla, Carsten Othmer, and Antonio Huerta. "Nonintrusive proper generalised decomposition for parametrised incompressible flow problems in OpenFOAM". In: *Computer physics communications* 249 (2020), p. 107013.

[47] Maria Antonietta Panza. "A Review of Experimental Techniques for NVH Analysis on a Commercial Vehicle". In: *Energy Procedia* 82 (2015), pp. 1017–1023.

[48] M. J. Griffin. “Discomfort from feeling vehicle vibration”. In: *Vehicle System Dynamics* 45.7-8 (2007), pp. 679–698.

[49] N.A.Z. Abdullah, M.S.M. Sani, M.M. Rahman, and I. Zaman. “Correlation of numerical and experimental analysis for dynamic behaviour of a body-in-white (BIW) structure”. In: *MATEC Web of Conferences* 90 (2016). Ed. by S.A. Che Ghani, W.A. Wan Hamzah, and A. Alias, p. 01020.

[50] Mohamad S Qatu. “Recent research on vehicle noise and vibration”. In: *International Journal of Vehicle Noise and Vibration* 8.4 (2012), pp. 289–301.

[51] Manfred Bäcker, Axel Gallrein, and Michael Roller. “Noise, vibration, harshness model of a rotating tyre”. In: *Vehicle System Dynamics* 54.4 (2016), pp. 474–491.

[52] Stefan Uhlar, Florian Heyder, and Thomas König. “Assessment of two physical tyre models in relation to their NVH performance up to 300 Hz”. In: *Vehicle System Dynamics* 59.3 (2019), pp. 331–351.

[53] Gino Paganelli, Yann Guezennec, and Giorgio Rizzoni. “Optimizing control strategy for hybrid fuel cell vehicle”. In: *SAE Transactions* (2002), pp. 398–406.

[54] S. Kermani, S. Delprat, T.M. Guerra, R. Trigui, and B. Jeanneret. “Predictive energy management for hybrid vehicle”. In: *Control Engineering Practice* 20.4 (2012), pp. 408–420.

[55] Clara Marina Martinez, Xiaosong Hu, Dongpu Cao, Efstathios Velenis, Bo Gao, and Matthias Wellers. “Energy management in plug-in hybrid electric vehicles: Recent progress and a connected vehicles perspective”. In: *IEEE Transactions on Vehicular Technology* 66.6 (2016), pp. 4534–4549.

[56] Yechen Qin, Xiaolin Tang, Tong Jia, Ziwen Duan, Jieming Zhang, Yinong Li, and Ling Zheng. “Noise and vibration suppression in hybrid electric vehicles: State of the art and challenges”. In: *Renewable and Sustainable Energy Reviews* 124 (2020), p. 109782.

[57] J De Cuyper, M Furmann, D Kading, and M Gubitosa. "Vehicle dynamics with LMS virtual. lab motion". In: *Vehicle System Dynamics* 45.S1 (2007), pp. 199–206.

[58] Jos Jans, Katrien Wyckaert, Marc Brughmans, Michael Kienert, Herman Van der Auweraer, Stijn Donders, and R Hadjit. *Reducing Body Development Time by Integrating NVH and Durability Analysis from the Start*. Tech. rep. SAE Technical Paper, 2006.

[59] Huan Qin, Zijian Liu, Yu Liu, and Haolong Zhong. "An object-oriented MATLAB toolbox for automotive body conceptual design using distributed parallel optimization". In: *Advances in Engineering Software* 106 (2017), pp. 19–32.

[60] D. Mundo, R. Hadjit, S. Donders, M. Brughmans, P. Mas, and W. Desmet. "Simplified modelling of joints and beam-like structures for BIW optimization in a concept phase of the vehicle design process". In: *Finite Elements in Analysis and Design* 45.6-7 (2009), pp. 456–462.

[61] H Van der Auweraer, S Donders, R Hadjit, M Brughmans, P Mas, and J Jans. "New approaches enabling NVH analysis to lead design in body development". In: *Proc. EIS NVH Symposium "New Technologies and Approaches in NVH", Coventry, UK*. 2005.

[62] J Jaap Wijker. *Mechanical vibrations in spacecraft design*. Springer Science & Business Media, 2004.

[63] Raymond L Bisplinghoff, Holt Ashley, and Robert L Halfman. "Aeroelasticity, Addison-Weslej'Publishing Company". In: *Reading, Mass* (1955), p. 527.

[64] Mark F Nelson and Joseph A Wolf Jr. "The use of inertia relief to estimate impact loads". In: *SAE Transactions* (1977), pp. 2237–2243.

[65] EY Kuo and SG Kelkar. *Vehicle body structure durability analysis*. Tech. rep. SAE Technical Paper, 1995.

[66] Alan R Barnett, Timothy W Widrick, and Damian R Ludwiczak. "Closed-form static analysis with inertia relief and displacement-dependent loads using a MSC/NASTRAN DMAP Alter". In: (1995).

[67] S Baskar. "Door structural slam durability inertia relief approach". In: *SAE transactions* (1998), pp. 2407–2415.

[68] Majid Anvari and Babak Beigi. *Automotive Body Fatigue Analysis–Inertia Relief or Transient Dynamics?* Tech. rep. SAE Technical Paper, 1999.

[69] Narayanan Pagaldipti and Yaw-Kang Shyy. "Influence of inertia relief on optimal designs". In: *Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference. 2004, pp. 616–621.

[70] Lin Liao. "A study of inertia relief analysis". In: *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 19th AIAA/ASME/AHS Adaptive Structures Conference 13t*. 2011, p. 2002.

[71] He Pengqiu and Qin Sun. "Modified inertia relief method based on accurate inertia loads". In: *AIAA Journal* 55.8 (2017), pp. 2848–2852.

[72] Fish Jacob and Belytschko Ted. *A first course in finite elements*. Wiley, 2007.

[73] P. Díez, S. Zlotnik, A. García-González, and A. Huerta. "Encapsulated PGD Algebraic Toolbox Operating with High-Dimensional Data". In: *Archives of Computational Methods in Engineering* 27.4 (2019), pp. 1321–1336.

[74] Pedro Díez, Sergio Zlotnik, Alberto García-González, and Antonio Huerta. "Encapsulated PGD algebraic toolbox operating with high-dimensional data". In: *Archives of computational methods in engineering* (2019), pp. 1–16.

[75] F. Cavaliere, S. Zlotnik, R. Sevilla, X. Larrayoz, and P. Díez. "Non-intrusive parametric solutions in structural dynamics". In: *Computer Methods in Applied Mechanics and Engineering* 389 (2022), p. 114336. ISSN: 0045-7825.

[76] Carlos Quesada, Icíar Alfaro, David González, Elías Cueto, and Francisco Chinesta. "PGD-Based Model Reduction for Surgery Simulation: Solid Dynamics and Contact Detection". In: *Biomedical Simulation*. Springer International Publishing, 2014, pp. 193–202.

[77] David González, Elias Cueto, and Francisco Chinesta. "Real-time direct integration of reduced solid dynamics equations". In: *International Journal for Numerical Methods in Engineering* 99.9 (2014), pp. 633–653.

[78] Claudia Germoso, Jose V. Aguado, Alberto Fraile, Enrique Alarcon, and Francisco Chinesta. "Efficient PGD-based dynamic calculation of non-linear soil behavior". In: *Comptes Rendus Mécanique* 344.1 (2016), pp. 24–41.

[79] Muhammad Haris Malik, Domenico Borzacchiello, Jose Vicente Aguado, and Francisco Chinesta. "Advanced parametric space-frequency separated representations in structural dynamics: A harmonic–modal hybrid approach". In: *Comptes Rendus Mécanique* 346.7 (2018), pp. 590–602.

[80] Giacomo Quaranta, Clara Argerich Martin, Ruben Ibañez, Jean Louis Duval, Elias Cueto, and Francisco Chinesta. "From linear to nonlinear PGD-based parametric structural dynamics". In: *Comptes Rendus Mécanique* 347.5 (2019), pp. 445–454.

[81] S. González-Pintor, D. Ginestar, and G. Verdú. "Using proper generalized decomposition to compute the dominant mode of a nuclear reactor". In: *Mathematical and Computer Modelling* 57.7-8 (2013), pp. 1807–1815.

[82] Jaron P Senecal. *Efficient Coupling Algorithms and Reduced-Order Methods for High-Fidelity Multiphysics Simulations of Nuclear Reactors*. Rensselaer Polytechnic Institute, 2018.

[83] Zachary M. Prince and Jean C. Ragusa. "Application of proper generalized decomposition to multigroup neutron diffusion eigenvalue calculations". In: *Progress in Nuclear Energy* 121 (2020), p. 103232.

[84] Zhaojun Bai, James Demmel, Jack Dongarra, Axel Ruhe, and Henk van der Vorst. *Templates for the solution of algebraic eigenvalue problems: a practical guide*. SIAM, 2000.

[85] Gene H Golub and Charles F Van Loan. *Matrix computations*. Vol. 3. JHU press, 2013.

[86] Alfio Quarteroni, Riccardo Sacco, and Fausto Saleri. *Numerical Mathematics*. Springer New York, 2007.

[87] J. G. F. Francis. "The QR Transformation A Unitary Analogue to the LR Transformation–Part 1". In: *The Computer Journal* 4.3 (1961), pp. 265–271.

[88] Cornelius Lanczos. "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators". In: (1950).

[89] Walter Edwin Arnoldi. "The principle of minimized iterations in the solution of the matrix eigenvalue problem". In: *Quarterly of applied mathematics* 9.1 (1951), pp. 17–29.

[90] E.J Davison and S.H Wang. "Properties and calculation of transmission zeros of linear multivariable systems". In: *Automatica* 10.6 (1974), pp. 643–658.

[91] Gerard L. G. Sleijpen and Henk A. Van der Vorst. "A Jacobi–Davidson Iteration Method for Linear Eigenvalue Problems". In: *SIAM Journal on Matrix Analysis and Applications* 17.2 (1996), pp. 401–425.

[92] Pedro Díez, Sergio Zlotnik, Alberto García-González, and Antonio Huerta. "Algebraic PGD for tensor separation and compression: An algorithmic approach". In: *Comptes Rendus Mécanique* 346.7 (2018), pp. 501–514.

[93] Fabiola Cavaliere, Sergio Zlotnik, Ruben Sevilla, Xabier Larráyoz, and Pedro Díez. "Nonintrusive reduced order model for parametric solutions of inertia relief problems". In: *International Journal for Numerical Methods in Engineering* 122.16 (2021), pp. 4270–4291.

[94] C.A. Felippa, K.C. Park, and M.R. Justino Filho. "The construction of free–free flexibility matrices as generalized stiffness inverses". In: *Computers & Structures* 68.4 (1998), pp. 411–418.

[95] Joaquim R. R. A. Martins and Andrew Ning. *Engineering Design Optimization*. Cambridge University Press, 2022. ISBN: 9781108833417.

[96] KC Giannakoglou, DI Papadimitriou, and IC Kampolis. "Aerodynamic shape design using evolutionary algorithms and new gradient-assisted metamodels". In: *Computer methods in applied mechanics and engineering* 195.44-47 (2006), pp. 6312–6329.

[97] Sean Walton, Oubay Hassan, and Kenneth Morgan. "Selected Engineering Applications of Gradient Free Optimisation Using Cuckoo Search and Proper Orthogonal Decomposition". In: *Archives of Computational Methods in Engineering* 20.2 (2013), pp. 123–154.

[98] Li Zhou Li, Jin Jiu Li, Jun Zhang, Kuan Lu, and Mei Ni Yuan. "Aerodynamic shape optimization by continually moving ROM". In: *Aerospace Science and Technology* 99 (2020), p. 105729.

[99] Dan Simon. *Evolutionary optimization algorithms*. John Wiley & Sons, 2013.

[100] Michael D McKay, Richard J Beckman, and William J Conover. "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code". In: *Technometrics* 42.1 (2000), pp. 55–61.

[101] John H Halton. "Algorithm 247: Radical-inverse quasi-random point sequence". In: *Communications of the ACM* 7.12 (1964), pp. 701–702.

[102] Marc Rocas, Alberto García-González, Sergio Zlotnik, Xabier Larráyoz, and Pedro Díez. "Nonintrusive uncertainty quantification for automotive crash problems with VPS/Pamcrash". In: *Finite Elements in Analysis and Design* 193 (2021), p. 103556.

[103] G Van Woensel, E Verdonck, and J De Baerdemaeker. "Measuring the mechanical properties of apple tissue using modal analysis". In: *Journal of Food Process Engineering* 10.3 (1988), pp. 151–163.

[104] Wen Zhang, Di Cui, Zihao Liu, and Yibin Ying. "Analysis of pear dynamic characteristics based on harmonic response". In: *Transactions of the ASABE* 59.6 (2016), pp. 1905–1913.

[105] G.A. Knott, Y.S. Shin, and M. Chargin. "A modal analysis of the violin". In: *Finite Elements in Analysis and Design* 5.3 (1989), pp. 269–279.

[106] Pyae Phyo Aung and Grigoryev Yuri Vsevolodovich. "Waveguide modal analysis of the ultrasonic medical instrument". In: *Journal of Physics: Conference Series* 1479.1 (2020), p. 012121.

[107] Ruben Sevilla, Sergio Zlotnik, and Antonio Huerta. "Solution of geometrically parametrised problems within a CAD environment via model order reduction". In: *Computer methods in applied mechanics and engineering* 358 (2020), p. 112631.

[108] Ruben Sevilla, Luca Borchini, Matteo Giacomini, and Antonio Huerta. "Hybridisable discontinuous Galerkin solution of geometrically parametrised Stokes flows". In: *arXiv preprint arXiv:2006.11846* (2020).