Technische Universität München

Ingenieurfakultät Bau Geo Umwelt

Lehrstuhl für Statik

Multiphysics Simulation and CAD-Integrated Shape Optimization in Fluid-Structure Interaction

Altuğ Emiroğlu

Vollständiger Abdruck der von der Ingenieurfakultät Bau Geo Umwelt der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender:

Priv.-Doz. Dr.-Ing. Roland Wüchner

Prüfer der Dissertation:

1. Prof. Dr.-Ing. Kai-Uwe Bletzinger
2. Prof. Dr. ir. Joris Degroote
3. Prof. Riccardo Rossi, Ph. D.

Die Dissertation wurde am 14.02.2019 bei der Technischen Universität München eingereicht und durch die Ingenieurfakultät Bau Geo Umwelt am 22.05.2019 angenommen.

Schriftenreihe des Lehrstuhls für Statik TU München

Band 39

Altuğ Emiroğlu

MULTIPHYSICS SIMULATION AND CAD-INTEGRATED SHAPE OPTIMIZATION IN FLUID-STRUCTURE INTERACTION

München 2019

Abstract

In this thesis, robust structural dynamic simulation methods for large scale prestressed structures, steady state partitioned fluid-structure interaction (FSI) analysis, CAD-based sensitivity filtering and CAD-integrated optimization methods with applications to Multidisciplinary Design Optimization (MDO) are investigated. The robust dynamic analysis methods for structures are of high importance since the accurate computation of the post-processing values, namely the stresses and strains, in time dependent problems are computationally costly. In order to reduce the computational effort, an updated linear dynamic analysis approach is introduced. Another presented achievement is the establishment of robust simulation methods for MDO in FSI. This incorporates several challenges starting from coupled simulations of the involved disciplines, sensitivity analysis for the design improvement, sensitivity filtering for achieving feasible designs while keeping the numerical models suitable for the repetitive computations of the FSI state and finally generating a manufacturable design. This thesis addresses the computational performance issues related to the solution of the steady state partitioned FSI problems by utilizing a mixture of analytical derivatives and least-squares based methods for the construction of an approximate Jacobian and employing an implicit coupling method. In addition, the necessary sensitivity filtering operations for the infeasible design eliminations in optimization problems are discussed in combination with the Mortar method. The sensitivity filtering is achieved by tying the discrete numerical models to their underlying CAD descriptions through the CAD-based Mortar Mapping method with embedded geometrical constraints that intrinsically satisfy the equality constraints. In particular, this method proves to be beneficial when MDO incorporates partitioned solution procedures for the surface coupled problems, such as FSI, by transferring the shared and the individual discipline relevant design handles to the CAD parameters, namely the control points. Finally, the advancements are integrated into the MDO workflow that is based on the Multidisciplinary Feasible (MDF) architecture, and CAD-integrated multidisciplinary optimization in FSI is achieved. The developed methods and their properties are demonstrated and elaborated on a set of problems with varying complexity. The demonstration cases are chosen in such a way that the level of complexity starts from comprehensive but small scale cases and reaches industrially relevant problems.

Zusammenfassung

In dieser Dissertation werden robuste strukturdynamische Simulationsmethoden für große, vorbelastete Strukturen, die Analyse der stationären, partitionierten Fluid-Struktur-Interaktion (FSI), CAD-basierte Sensitivitätsfilterung und CAD-integrierte Optimierungsmethoden mit Anwendungen für die multidisziplinäre Designoptimierung (MDO) untersucht. Die robuste dynamische Analyse von Strukturen ist in der Berechnung sehr essentiell da die präzise Analyse der Postprozessvariablen, beispielsweise der Spannungen und Dehnungen, in zeitabhängigen Problemen rechenintensiv sein können. Um den Rechenaufwand zu reduzieren, wird ein lineares, dynamisches Analyseverfahren mit Zustandsaktualisierung eingeführt. Ein weiteres Ergebnis dieser Arbeit ist die Entwicklung und systematische Untersuchung robuster Simulationsmethoden für die MDO mit FSI. Dies beinhaltet mehrere Herausforderungen; angefangen von der numerischen Kopplung der verwendeten Methoden, über die Sensitivitätsanalyse zur Designoptimierung, inklusive der Sensitivitätsfilterung zum Erzielen realisierbarer Designs, wobei die numerischen Modelle für die sich wiederholenden Berechnungen des FSI-Zustands erhalten bleiben, bis hin zur Anwendung dieser Ansätze zum Erzeugen eines herstellbaren Entwurfs. Es werden die Effizienzprobleme im Zusammenhang mit der partitionierten Lösung von stationären FSI-Problemen unter Verwendung einer Kombination aus analytischen Ableitungen und einem Ansatz basierend auf den kleinsten Fehlerquadraten für die Erstellung einer angenäherten Jacobi-Projektion und der Verwendung einer impliziten Kopplungsmethode erläutert. In Verbindung mit der Mortar-Methode wird die notwendige Sensitivitätsfilterung für die Beseitigung von unzulässigen Entwürfen bei Optimierungsproblemen entwickelt und diskutiert. Die Sensitivitätsfilterung wird erreicht, indem die diskreten numerischen Modelle durch das CAD-basierte Mortar-Mapping-Verfahren mit eingebetteten geometrischen Randbedingungen an die zugrundeliegenden CAD-Beschreibungen angebunden werden. Diese Methode erweist sich als vorteilhaft, wenn die MDO partitionierte Lösungsverfahren für oberflächengekoppelte Probleme wie FSI verwendet, indem die gemeinsam genutzten und die für die einzelnen Methoden relevanten Entwurfsdesigns auf die CAD-Geometrieparameter, d.h. die Kontrollpunkte der NURBS-Flächen, übertragen werden. Die methodischen Entwicklungen werden in einen MDO workflow integriert, der auf der sogenannten Multidisciplinary Feasible (MDF) Architektur basiert

und dadurch CAD-integrierte multidisziplinäre Optimierung mit FSI ermöglicht. Die vorgestellten Entwicklungen und Forschungsergebnisse werden anhand von Beispielen unterschiedlicher Komplexität demonstriert und diskutiert. Die Bandbreite reicht von analytischen Testfällen bis hin zu industriell motivierten Fragestellungen.

Acknowledgments

As I am reaching the end of an important era in my life, I feel grateful for being given the chance to be a member of the Chair of Structural Analysis.

First of all, I wish to express my gratitude to Prof. Dr.-Ing. Kai-Uwe Bletzinger for not only giving me the opportunity of being a part of this chair and to work in precious projects, such as AeroStruct and Mistralwind, which funded this research, but also supporting me with all the bureaucratical issues related to my stay in this chair and Germany. Secondly, I cannot stress the involvement and support of Priv.-Doz. Dr.-Ing. Roland Wüchner enough for taking time for discussions regarding research and external projects and broadening my horizon in thinking about the solutions to the challenges I encountered. I would like to mention that I greatefully acknowledge the funding of this research through the Federal Ministry for Economic Affairs and Energy (BMWi) [1] under contracts FKZ 20A1102A, 20A1102B, 20A1102C, 20A1102E for AeroStruct and FKZ 0325795D, 01158044 for Mistralwind.

I would like to express my appreciation to Andreas Apostolatos, who has been there not only as a friend since the beginning of our journey with the COME studies but also as a colleague during our research fellowship. Our fruitful discussions and collaboration are the main catalysts to the accomplishment of this work. I would also like to present my thanks to Shahrokh Shayegan and Reza Najian for their expertise and case setups of the fluid simulations. I would like to thank my current officemates Iñigo López Canalejo and Tobias Teschemacher for their energy and constant positive mood for enchanting our office environment. I would like to also thank all my former and current colleagues at the Chair of Structural Analysis.

Finally, I would like to send my thankfulness to my parents for supporting me at every stage of my life, which led me to this point, and also to all my friends from Munich, Turkey and all over the world, who have influences on me in any way.

Altuğ Emiroğlu
Technische Universität München
February, 2019

Contents

Imagination is more important than knowledge.
Knowledge is limited. Imagination encircles the world.

Albert Einstein

Chapter 1

Introduction

Multidisciplinary Design Optimization (MDO) has gained considerable attention in the last decades as a tool for bringing the interacting disciplines together and searching for the optimal design of the combined systems while collaboratively improving the participating disciplines' performances. The need for better designs in highly complex engineering systems gave birth to its research field with a variety of aspects from mathematical modeling to cost-effectiveness [2].

The performance of the design often relates to its physical response to the operating conditions, with which it is interacting. Thus, the accurate determination of the physical states of the designs in multidisciplinary contexts is essential, which is in today's circumstances achieved by means of computer simulations. The evaluation of the system responses in coupled settings comes with a price in many aspects including accurate mathematical modeling, efficient computerization and automation. Accurate mathematical modeling of physical systems has been well-established for many disciplines including fluid and structural mechanics. Even though efficient numerical analysis methods have been researched for individual disciplines, the efficient numerical analysis of their interactive responses is still an active research field due to intricacies in the employed methods for

their simulation methods. *Multidisciplinary Analysis* (MDA), as a research field, concerns with the collaborative behavior of coupled systems. Thus, it constitutes one of the main tools of the MDO process. One of the frequent occurrences in MDA is the *Fluid-Structure Interaction* (FSI) analysis, in particular, emerging in the engineering fields of wind energy, aerospace, biomedical industry etc. Even if the coupled responses of the constituting disciplines are left aside, the isolated behaviors of these are typically nonlinear, which involve a certain level of mathematical and computational challenge themselves. Furthermore, the interactive nature of the coupled problems adds another layer of complexity. There has been a considerable amount of research regarding this issue, and a variety of methods have been developed in order to tackle this challenge, which can be roughly grouped under monolithic, partitioned and field elimination approaches [3]. The monolithic approaches employ a simultaneous solution procedure for both of the disciplines, which brings the difficulty of describing the state variables of the subdisciplines from each others' perspectives. Furthermore, such monolithic structures are often designed to incorporate specific applications in mind, and as a result, lack the flexibility of including additional physical effects. Moreover, the preferred method employed for the numerical analysis of the coupled solution might not be suitable for at least one of the disciplines. On the other hand, researchers also adopt partitioned strategies for the simulation of coupled systems. The partitioned methods make use of the existing state of the art techniques for the coupled disciplines and utilize various iterative methods to retrieve the coupled physical state. The complexity in such coupling methods is the necessary computational effort and the stability issues related to the chosen coupling scheme. The field elimination requires the elimination of one of the differential equations by describing the primary variables from the other differential equation's point of view. Moreover, in the context of MDO, the coupled simulations are performed successively for each design evolution. Thus, regardless of the monolithic, the partitioned or field elimination approach to the coupled simulations, robustness is a crucial criterion.

The exploration of the design space in optimization problems is necessary in order to find a suitable design that satisfies the restrictions and conforms with the desired performance. For this purpose, a wide diversity of optimization algorithms have been developed, which suit different pur-

poses for the considered optimization problems. In particular, a substantial amount of MDO structures exist depending on the choice of analysis tools [4]. These optimization structures constitute many components, and the core optimization algorithm can be based on different optimization algorithms such as zero or higher order methods. The choice for the optimization algorithm depends on the definition of the performance criterion and the design handles that define the properties of the design. The design handles can be the shape as well as the parameters that define the physical behavior of the considered disciplines. In the context of shape optimization, the design can either be controlled by relatively few handles that are used to construct the shape from basic parameters or a free-form approach can be adopted where the shape itself, which is typically discretized for numerical analysis, can be the design handle itself. In the context of free-form shape optimization, it is reported that the gradient-based methods give more satisfying results [5]. The gradient-based methods require the computation of the objective function derivatives with respect to the design variables, which refers to the sensitivity analysis. There exists a wide variety of sensitivity analysis methods in the literature [6]. In the case of free-form surface-driven shape optimization, where the discrete shape is the design handle itself, adjoint methods are favored, since the complexity of the sensitivity analysis only scales with the number of performance criteria and adjoint methods result in efficient algorithms [7].

Furthermore, the shape optimization of the discrete problems often suffers from infeasible design updates of the numerical model, which do not allow the further disciplinary analysis and determination of the physical state, due to reasons stemming from the disciplinary analysis or the employed sensitivity analysis method [8]. Even if this is not the case, all of the design space might not be useful or desirable. Under these circumstances, sensitivity filtering methods are adopted for both ensuring the quality of the discrete numerical models and discarding the undesired designs [8]. The sensitivity analysis methods can roughly be grouped under implicit and explicit methods. The implicit filtering methods are part of the sensitivity analysis and have to be considered simultaneously by enforcing the sensitivity fields to satisfy an equation, for instance see [9] in the field of topology optimization. Typically, the explicit filtering methods benefit from their decoupled application from the sensitivity analysis phase. Thus, the explicit filtering methods are not bond to the sensitivity analysis stage, which

makes them applicable in combination with various sensitivity analysis tools.

The goal of the optimization is to achieve a better design which can replace the previous version. The engineering structures in this sense have to be produced, tested and finally put to operation. The definition of the product to be manufactured is often achieved utilizing *Computer Aided Design* (CAD) tools. Thus, the regeneration of the CAD models at the end of the optimization process is crucial, and the optimization processes that incorporate the CAD information highly benefit from this. There exist optimization methods that define the basic parameters of the CAD models as design handles [10]. Even though this type of optimization methods deliver a CAD model as a product, the basic parameters highly restrict the design space compared to the parameter-free or namely the node based optimization approaches [8]. On the other hand, there are methods which reconstruct a representative CAD model from the resulting discrete numerical models [11], [12]. These methods require algorithms for the identification of certain features that are necessary for building the *Boundary Representation* (BREP) of the CAD model, such as vertex and edge detection methods etc. , which are either not suitable for the regeneration of very complex discrete models or result in nonintuitive shape definitions. Furthermore, the initial CAD models are readily available, since the design of engineering structure often start with CAD models and the corresponding numerical analysis models stem from them. Thus, these models can be employed without impairing the desired free-form shape optimization tasks. In order to adopt the initial CAD model's parametrization as design variables, one needs to tie the discrete numerical model and the original CAD model. There are various proposed methods for this purpose, such as finite differencing, morphing boxes [13], *Free-Form Deformation* (FFD) [14], algorithmic differentiation [15] etc. , and they all come with their benefits and shortcomings.

CHAPTER 2 addresses the robust disciplinary simulation methods. It lays the foundations for the disciplinary analyses for the following FSI analysis. The employed methods for structural and fluid mechanics are laid out. Furthermore, the efficient coupling methods for the strongly coupled, partitioned FSI applications are extended using a combination of analytic expressions and least-squares approaches for the computation of the fluid interface Jacobian. The proposed method considers the FSI problem purely

from the structural point of view by utilizing the current state information and the iteration history for the approximation of the required tangent stiffness term for the acceleration of the iterative solution procedure. Just as importantly, this section also discusses robust and accurate dynamic analysis methods that are suitable for large scale structures. The proposed methods are then demonstrated on numerical examples in an increasing level of complexity.

CHAPTER 3 focuses on the required components for establishing the MDO process. The structural sensitivity analysis using the semianalytic, discrete adjoint method and the fluid sensitivity analysis utilizing continuous adjoint methodology are briefly presented with respective reasonings for their adoption. A free-form CAD-based parametrization method is proposed, which stems from the underlying NURBS or B-Spline description. The connection between the discrete numerical models and their underlying CAD descriptions is accomplished utilizing the principles of *Isogeometric Analysis* (IGA) and a Mortar-based formulation. Moreover, it is shown that such a procedure acts as a sensitivity filter that is relatable to the Vertex Morphing method and can be used in combination with it too, see also [16], [17]. The proposed method brings the typically nonconforming shape sensitivity fields related to the subdisciplines to a common platform and is a core component of the applied MDO process. The integration of the CAD into the MDO workflow is achieved through a software module that is based on OpenCASCADE Community Edition (OCE) and pythonOCC [18], [19], [20]. Last but not least, an MDO workflow based on the *Multidisciplinary Feasible* (MDF) architecture that can incorporate commercial scale CAD models is established and demonstrated on a wind energy application.

CHAPTER 4 rounds up the discussions and summarizes the outcomes of this work while giving an outlook and suggestions for possible future research ideas.

Chapter 2

High Fidelity Simulation Methods

This chapter introduces the governing equations and necessary notations for the structural analysis, fluid dynamics and the forthcoming fluid-structure interaction analysis as well as the improvements and the extensions to the existing methods. Firstly, the structural governing equations for both the dynamic and the steady state analysis are presented in correlation with the utilized software environment and the *Finite Element Method* (FEM). The extensions and the assumptions for the updated linear transient as well as the updated modal analyses are laid out in Sec. 2.1.6-2.1.7. Secondly, the steady state *Reynold Averaged Navier-Stokes* (RANS) equations in rotationg frames are introduced which use the *Finite Volume Method* (FVM) for the solution of the mentioned governing equations. Finally, the steady state interaction of the structures with fluids as well as the improvements for the Jacobian approximations in order to increase the convergence characteristics for the iterative solution procedures are proposed in Sec. 2.3.

2.1 Structural Analysis

This section introduces the notations and the solution methods for the structural governing equations using the FEM. The demonstration of the governing equations starts with a general point of view for a geometrically nonlinear transient analysis case mainly following the definitions in [21]. The specializations, the simplifications as well as the extensions to the existing structural analysis methods are discussed. In the scope of this work, the in-house structural analysis tool Carat++ [22] is utilized for both the transient structural analysis and the steady state solution of the FSI problems that are related to the MDO workflow, whereas Lagrange is employed for the solution of the pure FSI problems in steady state. Lagrange [23] is a software that is developed by Airbus Defence and Space (ADS) and was delivered to the Chair of Structural Analysis of TUM within the research project AeroStruct [24].

2.1.1 Governing Equations

The strong form of a general structural *Initial Boundary Value Problem* (IBVP) can be expressed as:

$$\rho\ddot{\mathbf{d}} + c\dot{\mathbf{d}} - \nabla\cdot\left(\boldsymbol{\sigma}+\boldsymbol{\sigma}_0\right) - \mathbf{b} = 0 \text{ , in } \Omega\times\mathbb{T} \text{ ,} \tag{2.1a}$$

$$\mathbf{d} = \mathbf{d}_0 \text{ , in } \Omega \text{ for } t = T_0 \text{ ,} \tag{2.1b}$$

$$\dot{\mathbf{d}} = \mathbf{v}_0 \text{ , in } \Omega \text{ for } t = T_0 \text{ ,} \tag{2.1c}$$

$$\mathbf{d} = \mathbf{0} \text{ , on } \Gamma_\mathrm{d} \subset \Gamma\times\mathbb{T} \text{ ,} \tag{2.1d}$$

$$\mathbf{t} = \bar{\mathbf{t}} \text{ , on } \Gamma_\mathrm{n} \subset \Gamma\times\mathbb{T} \text{ ,} \tag{2.1e}$$

where $\mathbf{d}$, $\boldsymbol{\sigma}$, $\boldsymbol{\sigma}_0$, $\mathbf{b}$, $\mathbf{t}$ denote the primal variables -displacements-, the deformation induced stresses, the prestresses, the body forces and the boundary tractions respectively. In addition, $\dot{\bullet}$ and $\ddot{\bullet}$ refer to the first and the second time derivatives. In an FSI setting, in addition to the applied body forces and the prescribed boundary tractions, the structure is subject to the fluid tractions $\mathbf{t}^\mathrm{f}$ on its wet surface $\Gamma_\mathrm{FSI} \subset \Gamma_\mathrm{n}$. This can be expressed as a state dependent Neumann boundary condition in the following way:

$$\mathbf{t} = \bar{\mathbf{t}} + \mathbf{t}^\mathrm{f}(\mathbf{d}) \text{ , on } \Gamma_\mathrm{FSI} \subset \Gamma_\mathrm{n}\times\mathbb{T} \text{ .} \tag{2.2}$$

The relation between the stresses and the displacements are given through the material law. In the scope of this thesis the kinematic modeling of the

structures are not restricted to a certain type. Thus, the relations are given in a global Cartesian coordinate system and specializations are presented or referenced where necessary for certain kinematic assumptions. The linear and the nonlinear Green-Lagrange strains can be written in the global Cartesian coordinate system in tensor form as:

$$\boldsymbol{\varepsilon}^{\text{lin}} = \frac{1}{2}\left(\nabla\mathbf{d} + (\nabla\mathbf{d})^T\right), \tag{2.3a}$$

$$\boldsymbol{\varepsilon}^{\text{nln}} = \frac{1}{2}\left(\nabla\mathbf{d} + (\nabla\mathbf{d})^T + \nabla\mathbf{d}\cdot\nabla\mathbf{d}\right). \tag{2.3b}$$

In the scope of this thesis, the work conjugate second Piola-Kirchhoff stresses and the Green-Lagrange strains are related through the St.Venant-Kirchhoff law as:

$$\boldsymbol{\sigma} = \boldsymbol{\mathcal{C}} : \boldsymbol{\varepsilon}, \tag{2.4}$$

where $\boldsymbol{\mathcal{C}}$ denotes the material tensor. Clearly, when linear Green-Lagrange strains are adopted, this law coincides with the Hooke's law [25].

2.1.2 Weak Form

Having the strong form of the problem, one can define the weak form of the structural governing equation, which corresponds to the virtual work by:

$$\delta\Pi = \int_\Omega \delta\mathbf{d}\cdot\left[\rho\ddot{\mathbf{d}} + c\dot{\mathbf{d}} - \nabla\cdot\left(\boldsymbol{\sigma} + \boldsymbol{\sigma}_0\right) - \mathbf{b}\right] \mathrm{d}\Omega = 0. \tag{2.5}$$

Applying integration by parts and splitting the boundary integrals into the Dirichlet and the Neumann boundaries one obtains the following:

$$\begin{aligned}\delta\Pi = &\int_\Omega \rho\,\delta\mathbf{d}\cdot\ddot{\mathbf{d}}\,\mathrm{d}\Omega + \int_\Omega c\,\delta\mathbf{d}\cdot\dot{\mathbf{d}}\,\mathrm{d}\Omega + \int_\Omega \delta\boldsymbol{\varepsilon} : \left(\boldsymbol{\sigma} + \boldsymbol{\sigma}_0\right)\mathrm{d}\Omega \\ &- \int_\Omega \delta\mathbf{d}\cdot\mathbf{b}\,\mathrm{d}\Omega - \int_{\Gamma_\mathrm{d}} \delta\mathbf{d}\cdot\mathbf{t}\,\mathrm{d}\Gamma - \int_{\Gamma_\mathrm{n}} \delta\mathbf{d}\cdot\left(\bar{\mathbf{t}} + \mathbf{t}^\mathrm{f}\right)\mathrm{d}\Gamma = 0.\end{aligned} \tag{2.6}$$

In the above equation, $\delta\mathbf{d}$ and $\delta\boldsymbol{\varepsilon}$ denote the virtual displacements and the virtual strains. Alternatively, one can define a form a, a functional l

and make use of the inner product of the L^2 space to shorten the virtual work expression:

$$a(\delta\mathbf{d},\mathbf{d}) \coloneqq \int_{\Omega} \delta\boldsymbol{\varepsilon} : \left(\boldsymbol{\sigma}+\boldsymbol{\sigma}_0\right) \mathrm{d}\Omega \,, \tag{2.7a}$$

$$l(\delta\mathbf{d}) \coloneqq \langle \delta\mathbf{d},\mathbf{b}\rangle_{0,\Omega} + \langle \delta\mathbf{d},\mathbf{t}\rangle_{0,\Gamma_\mathrm{d}} + \left\langle \delta\mathbf{d},\bar{\mathbf{t}}+\mathbf{t}^\mathrm{f}\right\rangle_{0,\Gamma_\mathrm{n}} \,, \tag{2.7b}$$

and the inner product of the L^2 space on a domain D is defined as:

$$\left\langle \boldsymbol{\psi},\boldsymbol{\varphi}\right\rangle_{0,D} = \int_D \boldsymbol{\psi}\cdot\boldsymbol{\varphi}\ \mathrm{d}D \,, \quad \forall \boldsymbol{\psi},\boldsymbol{\varphi} \in L^2(D) \,. \tag{2.8}$$

By doing so, one can define the structural IBVP as: Find $\mathbf{d}$ for all the admissible variations $\delta\mathbf{d}$ such that:

$$\left\langle \delta\mathbf{d},\rho\ddot{\mathbf{d}}\right\rangle_{0,\Omega} + \left\langle \delta\mathbf{d},c\dot{\mathbf{d}}\right\rangle_{0,\Omega} + a(\delta\mathbf{d},\mathbf{d}) = l(\delta\mathbf{d}) \,. \tag{2.9}$$

Above equation defines the weak form for the equilibrium of the linear momentum for a time dependent problem and has to be satisfied at any point in time $t \in \mathbb{T}$ for every $\delta\mathbf{d}$. The details on the equivalence of the weak and strong forms can be found in [26].

2.1.3 Spatial Semidiscretization

The discretization and the numerical solution of the structural problem is achieved using the FEM. The FEM requires the division of the domain Ω into subsets of the domain of interest and approximation of the unknowns with appropriate basis functions within each subdomain. In the context of FEM, these functions are often linear, bilinear or trilinear basis functions. For instance, the primary variable of the structural problem -the displacement field $\mathbf{d}$- is approximated as:

$$\mathbf{d}(\mathbf{X},t) = \sum_{i=1}^{m} \boldsymbol{\phi}_{(i)}(\mathbf{X})\,\hat{d}_{(i)}(t) \,, \tag{2.10}$$

where $\boldsymbol{\phi}_{(i)}$ denotes the basis functions calculated at the coordinate $\mathbf{X}$, which are related to the degrees of freedom (DOFs) $\hat{d}_{(i)}$ and m is the number of DOFs. The full set of DOFs are collected in a vector of variables:

$$\hat{\mathbf{d}} = \left[\hat{d}_{(1)} \cdots \hat{d}_{(m)}\right]^T \,. \tag{2.11}$$

Applying the appropriate discretization scheme, one can cast the Eqn. (2.6) to a discrete equation system. In addition, one needs to satisfy Eqn. (2.6) for every discrete set of $\delta\hat{\mathbf{d}}$ yielding the discrete residual form of the dynamic equilibrium:

$$\delta\Pi = \frac{\partial\Pi}{\partial\hat{d}_{(i)}}\delta\hat{d}_{(i)} = \hat{R}^{\mathrm{d}}_{(i)}\delta\hat{d}_{(i)} = 0\,, \tag{2.12}$$

where the dynamic residual $\hat{\mathbf{R}}^{\mathrm{d}}$ defined as:

$$\hat{\mathbf{R}}^{\mathrm{d}}(\hat{\mathbf{d}}) = \mathbf{M}\ddot{\hat{\mathbf{d}}} + \mathbf{C}\dot{\hat{\mathbf{d}}} + \hat{\mathbf{R}}(\hat{\mathbf{d}}) - \hat{\mathbf{F}}_{\mathrm{dyn}} = 0\,. \tag{2.13}$$

Herein, $\mathbf{M}$, $\mathbf{C}$, $\hat{\mathbf{R}}$ and $\hat{\mathbf{F}}_{\mathrm{dyn}}$ denote the mass matrix, the damping matrix, the steady state residual vector and the dynamic loading respectively. The entries of the mass matrix $\mathbf{M}$ are calculated through the following equation:

$$M_{(i,j)} = \left\langle \boldsymbol{\phi}_{(i)}, \rho\boldsymbol{\phi}_{(j)} \right\rangle_{0,\Omega}\,, \tag{2.14}$$

while the entries of the steady state residual vector $\hat{\mathbf{R}}$ are defined as:

$$\begin{aligned}\hat{R}_{(i)} = &\int_{\Omega} \frac{\partial\boldsymbol{\varepsilon}}{\partial\hat{d}_{(i)}} : \left(\boldsymbol{\sigma} + \boldsymbol{\sigma}_0\right)\,\mathrm{d}\Omega \\ &- \left\langle \boldsymbol{\phi}_{(i)}, \mathbf{b} \right\rangle_{0,\Omega} - \left\langle \boldsymbol{\phi}_{(i)}, \bar{\mathbf{t}}_{\mathrm{sta}} \right\rangle_{0,\Gamma_{\mathrm{n}}} - \left\langle \boldsymbol{\phi}_{(i)}, \mathbf{t}^{\mathrm{f}}_{\mathrm{sta}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}}\,.\end{aligned} \tag{2.15}$$

It should be noted that above equation contains only the static part of the prescribed boundary tractions. Eqn. (2.15) is going to be revisited in the following sections in order to introduce the assumptions related to the individual solution procedures. The dynamic consistent nodal force vector is computed accordingly from;

$$\hat{\mathbf{F}}_{\mathrm{dyn}} = \left\langle \boldsymbol{\phi}, \bar{\mathbf{t}}_{\mathrm{dyn}} \right\rangle_{0,\Gamma_{\mathrm{n}}}\,. \tag{2.16}$$

The computation of the damping matrix $\mathbf{C}$ is rather cumbersome. The correct definition of the damping parameters requires rigorous material testing [27]. As a simpler approach one can adopt the Rayleigh damping [28] and compute the damping matrix as a linear combination of the mass and stiffness matrices:

$$\mathbf{C} = \alpha_R\mathbf{M} + \beta_R\mathbf{K}\,, \tag{2.17}$$

where α_R and β_R denote the Rayleigh damping parameters. Following from the literature [29], for given two angular eigenfrequencies $\hat{\omega}_1$, $\hat{\omega}_2$ and related logarithmic decrements ζ_1, ζ_2, these parameters can be estimated:

$$\alpha_R = \frac{2\hat{\omega}_1\hat{\omega}_2}{\hat{\omega}_2^2 - \hat{\omega}_1^2}\left(\hat{\omega}_2\zeta_1 - \hat{\omega}_1\zeta_2\right), \tag{2.18a}$$

$$\beta_R = \frac{2\hat{\omega}_1\hat{\omega}_2}{\hat{\omega}_2^2 - \hat{\omega}_1^2}\left(-\frac{\zeta_1}{\hat{\omega}_2} + \frac{\zeta_2}{\hat{\omega}_1}\right). \tag{2.18b}$$

2.1.4 Time Discretization

In the scope of this thesis, the time integration is achieved via Newmark scheme [30] if not indicated otherwise. According to Newmark, the update rules at a discrete time step $t = t_s$ are given as:

$$\hat{\mathbf{d}}_{t_s} = \hat{\mathbf{d}}_{t_s-1} + \Delta t\,\dot{\hat{\mathbf{d}}}_{t_s-1} + \Delta t^2\left(\frac{1}{2} - \beta_N\right)\ddot{\hat{\mathbf{d}}}_{t_s-1} + (\Delta t)^2\beta_N\ddot{\hat{\mathbf{d}}}_{t_s}, \tag{2.19a}$$

$$\dot{\hat{\mathbf{d}}}_{t_s} = \dot{\hat{\mathbf{d}}}_{t_s-1} + \Delta t\left(1-\gamma_N\right)\ddot{\hat{\mathbf{d}}}_{t_s-1} + \Delta t\gamma_N\ddot{\hat{\mathbf{d}}}_{t_s}, \tag{2.19b}$$

where γ_N and β_N are the Newmark parameters which can be adjusted to maintain certain characteristics of the dynamic problem, such as accuracy, high frequency damping etc., [30]. Employing the update rules of Eqn. (2.19) in Eqn. (2.13) one obtains the space and time discretized dynamic residual:

$$\begin{aligned}\hat{\mathbf{R}}^{\mathrm{d}}_{t_s}\left(\hat{\mathbf{d}}_{t_s}\right) = &\left(\frac{1}{\beta_N(\Delta t)^2}\mathbf{M} + \frac{\gamma_N}{\beta_N\Delta t}\mathbf{C}\right)\hat{\mathbf{d}}_{t_s} + \hat{\mathbf{R}}\left(\hat{\mathbf{d}}_{t_s}\right) - \hat{\mathbf{F}}_{\mathrm{dyn}}\\ &-\left(\frac{1}{\beta_N(\Delta t)^2}\mathbf{M} + \frac{\gamma_N}{\beta_N\Delta t}\mathbf{C}\right)\hat{\mathbf{d}}_{t_s-1}\\ &-\left(\frac{1}{\beta_N\Delta t}\mathbf{M} - \frac{\beta_N-\gamma_N}{\beta_N}\mathbf{C}\right)\dot{\hat{\mathbf{d}}}_{t_s-1}\\ &-\left(\frac{1-2\beta_N}{2\beta_N}\mathbf{M} - \Delta t\,\frac{2\beta_N-\gamma_N}{2\beta_N}\mathbf{C}\right)\ddot{\hat{\mathbf{d}}}_{t_s-1}.\end{aligned} \tag{2.20}$$

2.1.5 Solution Procedure

The residual in Eqn. (2.20) defines a nonlinear system. Consequtively, an iterative solution procedure based on *Newton-Raphson* (NR) method is adopted. Thus, the linearization of Eqn. (2.20) at each time instance is necessary. Firstly, the derivative of the steady state residual vector $\hat{\mathbf{R}}$ is obtained:

$$\frac{\partial \hat{R}_{(i)}}{\partial \hat{d}_{(j)}} = K^{\mathrm{s}}_{(i,j)} = K^{\mathrm{int}}_{(i,j)} + K^{\mathrm{load}}_{(i,j)} \,, \tag{2.21}$$

where the entries of the so called internal and load tangent stiffnesses -$K^{\mathrm{int}}_{(i,j)}$ and $K^{\mathrm{load}}_{(i,j)}$ respectively- are defined as:

$$K^{\mathrm{int}}_{(i,j)} = \int_{\Omega} \frac{\partial \boldsymbol{\varepsilon}}{\partial \hat{d}_{(i)}} : \frac{\partial \boldsymbol{\sigma}}{\partial \hat{d}_{(j)}} \, \mathrm{d}\Omega + \int_{\Omega} \frac{\partial^2 \boldsymbol{\varepsilon}}{\partial \hat{d}_{(i)} \partial \hat{d}_{(j)}} : \left(\boldsymbol{\sigma} + \boldsymbol{\sigma}_0\right) \mathrm{d}\Omega \,, \tag{2.22a}$$

$$K^{\mathrm{load}}_{(i,j)} = -\left\langle \boldsymbol{\phi}_{(i)}, \frac{\partial \bar{\mathbf{t}}}{\partial \hat{d}_{(j)}} \right\rangle_{0,\Gamma_{\mathrm{n}}} - \left\langle \boldsymbol{\phi}_{(i)}, \frac{\partial \mathbf{t}^{\mathrm{f}}}{\partial \hat{d}_{(j)}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} . \tag{2.22b}$$

In Eqn. (2.22b), the second right hand side (RHS) term is of high interest. In the scope of this work, this term is going to be either approximated or neglected which results in a Quasi-Newton scheme. See Sec. 2.3 for further details.

Secondly, following from Eqn. (2.20) and having the steady state tangent stiffness defined, one can obtain the tangent stiffness matrix for the solution of the nonlinear dynamic problem:

$$\frac{\partial \hat{R}^{\mathrm{d}}_{(i)}}{\partial \hat{d}_{(j)}} = K_{(i,j)} = K^{\mathrm{d}}_{(i,j)} + K^{\mathrm{s}}_{(i,j)} \,, \tag{2.23}$$

where the contribution of the dynamic problem $\mathbf{K}^{\mathrm{d}}$ to the tangent stiffness matrix is obtained from:

$$\mathbf{K}^{\mathrm{d}} = \left(\frac{1}{\beta_N (\Delta t)^2} \mathbf{M} + \frac{\gamma_N}{\beta_N \Delta t} \mathbf{C} \right) . \tag{2.24}$$

Finally, NR solution procedure at time step t_s and iteration n is defined as:

$$\mathbf{K}\left(\hat{\mathbf{d}}_{t_s,n}\right) \Delta \hat{\mathbf{d}}_{t_s,n} = -\hat{\mathbf{R}}^{\mathrm{d}}_{t_s}\left(\hat{\mathbf{d}}_{t_s,n}\right) . \tag{2.25}$$

The discrete DOF vector at time step t_s is then updated accordingly:

$$\hat{\mathbf{d}}_{t_s,n+1} = \hat{\mathbf{d}}_{t_s,n} + \Delta\hat{\mathbf{d}}_{t_s,n} \ . \tag{2.26}$$

2.1.6 Updated Linear Transient Analysis

In practice, it is possible that a structure is subject to a combination of static and dynamic loads as well as prestresses where the nonlinearity at the static loading phase is much prominent than that of the dynamic phase. Herein, the deformation inducing external consistent nodal forces $\hat{\mathbf{F}}_{\text{ext}}$ are decomposed into a constant and a time varying part:

$$\hat{\mathbf{F}}_{\text{ext}} = \hat{\mathbf{F}}_{\text{const}} + \hat{\mathbf{F}}_{\text{t}} \ , \tag{2.27}$$

where;

$$\hat{\mathbf{F}}_{\text{const}} = \hat{\mathbf{F}}_{\text{sta}} + \left\langle \hat{\mathbf{F}}_{\text{dyn}} \right\rangle_{\mathbb{T}} \ , \tag{2.28a}$$

$$\hat{\mathbf{F}}_{\text{t}} = \hat{\mathbf{F}}_{\text{dyn}} - \left\langle \hat{\mathbf{F}}_{\text{dyn}} \right\rangle_{\mathbb{T}} \ . \tag{2.28b}$$

In the above equations, $\langle\cdot\rangle_{\mathbb{T}}$ is understood as the mean value of a series in the time domain. It can be shown that as long as damping is present in the dynamic system, the vibrations related to the constant part of the loading $\hat{\mathbf{F}}_{\text{const}}$ converge to a steady state solution. In addition, any arbitrary periodic loading can be expressed by a Fourier-series, to which the structure responds with the same frequency content, see also [31]. This allows the assumption that the structure essentially oscillates around the equilibrium point which is defined by the loading $\hat{\mathbf{F}}_{\text{const}}$.

Herein, the complexity in the nonlinear dynamic analysis is that the above statements are valid for linear dynamics. However, if the static and the dynamic phases are distinguished by assuming a nonlinear behaviour in the initial static phase and a linear behaviour in the forthcoming dynamic phase, one can still introduce the same notions. For this purpose, the constant loading defined in Eqn. (2.28a) is assumed to give a *pseudo equilibrium point* around which the structure oscillates. By doing so, the static analysis can be performed with the nonlinear strain measures and the solution procedure defined in Sec. 2.1.5 under loading defined by $\hat{\mathbf{F}}_{\text{const}}$. This stage of the analysis can also consider the applied prestresses if necessary.

Furthermore, it is assumed that the time varying part of the loading $\hat{\mathbf{F}}_{\text{t}}$ does not cause further nonlinearities and the structural oscillations can be

linearized around the pseudo equilibrium point. This assumption enables the use of the tangent stiffness matrix at this point and performing the dynamic analysis in a linear fashion. However, the initial conditions for the dynamic phase are still computed by the load case defined by $\hat{\mathbf{F}}_{\text{sta}}$ giving $\hat{\mathbf{d}}_{\text{sta}}$, since otherwise the initial conditions for the structural IBVP as well as the dynamic loading are applied incorrectly. The solution steps of the procedure defined in Sec. 2.1.5 and Eqns. (2.20)-(2.26) have to be modified:

$$\begin{aligned}\widetilde{\hat{\mathbf{R}}}^{\text{d}}_{t_s}\left(\hat{\mathbf{d}}^{\text{d}}_{t_s}\right) = &\left(\frac{1}{\beta_N(\Delta t)^2}\mathbf{M}+\frac{\gamma_N}{\beta_N \Delta t}\mathbf{C}\right)\hat{\mathbf{d}}^{\text{d}}_{t_s}+\mathbf{K}^{\text{s}}(\hat{\mathbf{d}}_{\text{eq}})\hat{\mathbf{d}}^{\text{d}}_{t_s}-\hat{\mathbf{F}}_{\text{dyn}}\\ &-\left(\frac{1}{\beta_N(\Delta t)^2}\mathbf{M}+\frac{\gamma_N}{\beta_N \Delta t}\mathbf{C}\right)\hat{\mathbf{d}}^{\text{d}}_{t_s-1}\\ &-\left(\frac{1}{\beta_N \Delta t}\mathbf{M}-\frac{\beta_N-\gamma_N}{\beta_N}\mathbf{C}\right)\dot{\hat{\mathbf{d}}}^{\text{d}}_{t_s-1}\\ &-\left(\frac{1-2\beta_N}{2\beta_N}\mathbf{M}-\Delta t\,\frac{2\beta_N-\gamma_N}{2\beta_N}\mathbf{C}\right)\ddot{\hat{\mathbf{d}}}^{\text{d}}_{t_s-1}\,. \end{aligned} \tag{2.29}$$

In the above equation, the terms related to the static phase are ommitted since they are assumed to be in equilibrium and have no effect in the residual expression. $\hat{\mathbf{d}}_{\text{eq}}$ denotes the pseudo static equilibrium point at which the structure is in equilibrium with the external loading defined in Eqn. (2.28a). In addition, $\hat{\mathbf{d}}^{\text{d}}_{t_s}$ denotes the dynamic part of the displacement solution. This implies that the static phase and the dynamic phase of the problem are separated and the actual displacements at time step t_s are given as:

$$\hat{\mathbf{d}}_{t_s} = \hat{\mathbf{d}}_{\text{sta}} + \hat{\mathbf{d}}^{\text{d}}_{t_s}\,. \tag{2.30}$$

Moreover, the effective stiffness matrix of the dynamic problem is modified accordingly:

$$\widetilde{\mathbf{K}} = \mathbf{K}^{\text{d}} + \mathbf{K}^{\text{s}}(\hat{\mathbf{d}}_{\text{eq}})\,. \tag{2.31}$$

This can also be referred as a model updating technique where the stiffness variations due to the nonlinearities caused by the static deformations and the prestresses are incorporated in the dynamic phase through the tangent stiffness matrix $\mathbf{K}^{\text{s}}(\hat{\mathbf{d}}_{\text{eq}})$. See also [32] for the inclusion of the stress stiffening for the dynamic response of the structures.

2.1.7 Updated Modal Analysis

Modal analysis makes use of the mass and stiffness properties of a structure and computes the so-called eigenfrequencies at which a structure resonates with the corresponding deformation modes; eigenmodes. In the context of FEM, the mass and stiffness properties are given by the respective matrices and it is assumed that the time dependent displacements can be described as the superposition of chosen n eigenmodes $\hat{\boldsymbol{\Phi}}_{(i)}$ that are related to the modal degrees of freedom:

$$\hat{\mathbf{d}} = \sum_{i=1}^{n} \hat{\boldsymbol{\Phi}}_{(i)} \sin\left(\hat{\omega}_{(i)} t\right) . \tag{2.32}$$

The conventional modal analysis with FEM assumes linear strain measures for the computation of the stiffness matrix, which essentially implies that the structural oscillations occur around the equilibrium point $\hat{\mathbf{d}} = \mathbf{0}$. This results in the exclusion of the stress stiffening and the prestressing effects that are present. This is in particular not accurate for large scale structures where the structural stiffness properties are affected by the constant loading. As an alternative, one can employ the static equilibrium point $\hat{\mathbf{d}}_{\text{sta}}$. In that case, the assumption implies that the structural oscillations take place around the static equilibrium point, which is not necessarily the case since the level of the dynamic loading is neglected. On the other hand, it is proposed in Sec. 2.1.6 that the structural oscillations occur in the vicinity of the state $\hat{\mathbf{d}}_{\text{eq}}$ and correspondingly the linearization can be achieved around this equilibrium point. Thus, an updated modal analysis approach is employed by incorporating the nonlinear strain measures and computing the tangent stiffness at the equilibrium state defined by the loading $\hat{\mathbf{F}}_{\text{const}}$. Introducing Eqn. (2.32) in Eqn. (2.13), neglecting the damping matrix and dropping the external forces, since the interest is on the free vibrations, Eqn. (2.13) takes the following form:

$$-\left(\omega^*_{(i)}\right)^2 \mathbf{M} \hat{\boldsymbol{\Phi}}^*_{(i)} + \mathbf{K}^{\text{s}}(\hat{\mathbf{d}}_{\text{eq}}) \hat{\boldsymbol{\Phi}}^*_{(i)} = \mathbf{0} . \tag{2.33}$$

Above equation defines an eigenvalue problem to be solved for the *updated angular eigenfrequencies* $\omega^*_{(i)}$ and the *updated eigenmodes* $\hat{\boldsymbol{\Phi}}^*_{(i)}$. The related *updated eigenfrequencies* $f^*_{(i)}$ can be computed using a simple relation: $f^*_{(i)} = \frac{\omega^*_{(i)}}{2\pi}$.

Herein, a distinction should be made. Updated modal analysis methods already exist and reference the stress stiffening effect as well as its influence on the eigenfrequencies of the structure, but this effect is often included by making use of the actual static equilibrium point $\hat{\mathbf{d}}_{\text{sta}}$. However, when forced vibrations of a structure are to be computed by transient modal analysis, the equilibrium point, around which the structure vibrates, does not necessarily coincide with the static equilibrium point. The presented method includes the loading level of the dynamic phase into account by the second RHS term in Eqn. (2.28a). Thus, it is expected to give more accurate results.

2.2 Fluid Dynamics

This section presents the governing equations for the fluid dynamics problems that are considered within the scope of this thesis. The existing implementations within the software framework OpenFOAM® as well as the extensions of the HELYX® are extensively used. Herein, only the solved equations are presented for clarity. The detailed information on the theory and the implementation aspects can be found in [33], [34], [35], [36].

In the scope of this thesis, the interactions of the structures with incompressible Newtonian fluids are considered. The steady state RANS equations are suitable for this purpose. However, even though the time derivatives vanish due to the steady state assumption, the *Arbitrary Lagrangian-Eulerian* (ALE) version of the Navier-Stokes equations are still necessary when a rotating frame is employed. Thus, the solution of the fluid *Boundary Value Problem* (BVP) is achieved using the steady state incompressible ALE Navier-Stokes equations with the domain velocity defined by the constant angular rotation. The governing equations with its boundary conditions can be written as follows:

$$\left(\mathbf{v}-\mathbf{v}_{\Omega}\right)\cdot\nabla\mathbf{v}-\nabla\cdot\widetilde{\boldsymbol{\sigma}}^{\mathrm{f}}-\mathbf{b}=0,\ \text{in}\ \Omega\,, \tag{2.34a}$$

$$\nabla\cdot\mathbf{v}=0,\ \text{in}\ \Omega\,, \tag{2.34b}$$

$$\mathbf{v}=\bar{\mathbf{v}},\ \text{on}\ \Gamma_{\mathrm{d}}\,, \tag{2.34c}$$

$$\mathbf{t}=\bar{\mathbf{t}},\ \text{on}\ \Gamma_{\mathrm{n}}\,, \tag{2.34d}$$

with the density averaged fluid Cauchy stress tensor $\widetilde{\boldsymbol{\sigma}}^{\mathrm{f}}$, the density averaged pressure $\widetilde{p}$ and the domain velocity $\mathbf{v}_{\Omega}$ defined as:

$$\widetilde{\boldsymbol{\sigma}}^{\mathrm{f}}=-\widetilde{p}\,\mathbf{I}+\widetilde{\nu}\left(\nabla\mathbf{v}+(\nabla\mathbf{v})^{T}\right)\,, \tag{2.35a}$$

$$\widetilde{p}=\frac{p}{\rho}\,, \tag{2.35b}$$

$$\mathbf{v}_{\Omega}=\boldsymbol{\omega}\times\mathbf{r}\,. \tag{2.35c}$$

Herein, p refers to the actual pressure while $\widetilde{\nu}$ refers to the turbulent kinematic viscosity in turbulent cases or the constant kinematic viscosity in laminar cases. In the frame of this work, the turbulent kinematic viscosity is determined using the steady state version of the Spalart-Allmaras turbulence model where necessary [37]. In the scope of this work, the body

forces **b** in Eqn. (2.34a) contain the Coriolis forces when a rotating frame is of interest. The momentum and the continuity equations in Eqns. (2.34a)-(2.34b) are clearly density averaged and do not differ from the original equation system since the flow field is assumed to be incompressible and has constant density. In an FSI setting, the domain deformation should also be taken into account too. The deformations related to the boundary perturbations are computed using an appropriate technique within OpenFOAM®.

2.3 Steady State FSI Analysis

In this section, the employed solution strategies and the performance comparisons of the adopted methods for the solution of the fluid-structure interactions are presented. In the frame of this work, the goal of the FSI analysis is to obtain the equilibrium condition, which is then used for the shape optimization. The FSI analysis phase is restricted to the steady state case due to the complexities in both the transient sensitivity analysis methods and the computational effort necessary for advancing the opitimization iterations.

The typical boundary conditions for a Dirichlet-Neumann coupling between the structural and the fluid BVPs at their common interface Γ_{FSI} can be written as follows:

$$\mathbf{d}^{\text{s}} - \mathbf{d}^{\text{f}} = \mathbf{0}, \text{ on } \Gamma_{\text{FSI}} , \tag{2.36a}$$

$$\mathbf{v} = \dot{\mathbf{d}}^{\text{s}} = \mathbf{0}, \text{ on } \Gamma_{\text{FSI}} , \tag{2.36b}$$

$$\mathbf{t} + \mathbf{t}^{\text{f}} = \mathbf{0}, \text{ on } \Gamma_{\text{FSI}} . \tag{2.36c}$$

The above conditions enforce the material continuity as well as the linear momentum equilibrium across the common interface. In practice, partitioned FSI techniques are in high interest since they allow coupling of different software with their advantageous sides. For instance, in the scope of this work, the structural BVP is solved via FEM while the fluid BVP is dealt with using FVM. One can see the FSI problem from the structural point of view and treat the fluid BVP as a function that returns boundary tractions acting on the structural wet surface for a given boundary deformation pattern. This point of view facilitates the use of the equations defined in Sec. 2.1 for the solution of the FSI problem. Herein, the complexity of the FSI problem stems from the correct linearization of the highly nonlinear fluid BVP and the interested post-processing value, namely the fluid tractions $\mathbf{t}^{\text{f}}$, with respect to the boundary perturbations. Given the definition for the fluid tractions as:

$$\mathbf{t}^{\text{f}} = \mathbf{n} \cdot \boldsymbol{\sigma}^{\text{f}} , \tag{2.37}$$

the second RHS term in Eqn. (2.22b), which contains the derivative of $\mathbf{t}^{\mathrm{f}}$, can also be written as follows:

$$\left\langle \boldsymbol{\phi}_{(i)}, \frac{\mathrm{d}\mathbf{t}^{\mathrm{f}}}{\mathrm{d}\hat{d}_{(j)}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} = \left\langle \boldsymbol{\phi}_{(i)}, \frac{\partial \mathbf{n}}{\partial \hat{d}_{(j)}} \cdot \boldsymbol{\sigma}^{\mathrm{f}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} + \left\langle \boldsymbol{\phi}_{(i)}, \mathbf{n} \cdot \frac{\mathrm{d}\boldsymbol{\sigma}^{\mathrm{f}}}{\mathrm{d}\hat{d}_{(j)}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} . \tag{2.38}$$

In the above equation, the first RHS term is straightforward to compute since it contains only the derivative of the surface normal vector. On the other hand, it is important to note that the second RHS term in Eqn. (2.38) can not be computed with partial derivatives but should be computed via total derivatives, since the constituting variables $\mathbf{v}$ and p of $\boldsymbol{\sigma}^{\mathrm{f}}$ can only be determined through the solution of the fluid BVP for a given boundary displacement field.

In order to establish a clear ground for the discussions in this section, the residual form and the linearization of the structural BVP considering only the fluid tractions as external loading are briefly presented here. The residual form of the structural BVP in this case reads:

$$\hat{R}^{\mathrm{FSI}}_{(i)} = \int_{\Omega} \frac{\partial \boldsymbol{\varepsilon}}{\partial \hat{d}_{(i)}} : \left(\boldsymbol{\sigma} + \boldsymbol{\sigma}_0\right) \mathrm{d}\Omega - \left\langle \boldsymbol{\phi}_{(i)}, \mathbf{t}^{\mathrm{f}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} . \tag{2.39}$$

Moreover, in the extent of this thesis, the linear Green-Lagrange strain measure is adopted for FSI problems without the loss of generality as in Eqn. (2.3a), since the proposed methods in the following sections essentially concentrate on the approximation of the load tangent. Then, the tangent stiffness matrix for the above Eqn. (2.39) can be computed from:

$$\frac{\partial \hat{R}_{(i)}}{\partial \hat{d}_{(j)}} = K^{\mathrm{s}}_{(i,j)} = K^{\mathrm{int}}_{(i,j)} + K^{\mathrm{f}}_{(i,j)} , \tag{2.40a}$$

$$K^{\mathrm{int}}_{(i,j)} = \int_{\Omega} \frac{\partial \boldsymbol{\varepsilon}}{\partial \hat{d}_{(i)}} : \frac{\partial \boldsymbol{\sigma}}{\partial \hat{d}_{(j)}} \, \mathrm{d}\Omega , \tag{2.40b}$$

$$\begin{aligned} K^{\mathrm{f}}_{(i,j)} &= -\left\langle \boldsymbol{\phi}_{(i)}, \frac{\mathrm{d}\mathbf{t}^{\mathrm{f}}}{\mathrm{d}\hat{d}_{(j)}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} \\ &= -\left\langle \boldsymbol{\phi}_{(i)}, \frac{\partial \mathbf{n}}{\partial \hat{d}_{(j)}} \cdot \boldsymbol{\sigma}^{\mathrm{f}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} - \left\langle \boldsymbol{\phi}_{(i)}, \mathbf{n} \cdot \frac{\mathrm{d}\boldsymbol{\sigma}^{\mathrm{f}}}{\mathrm{d}\hat{d}_{(j)}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} . \end{aligned} \tag{2.40c}$$

It is important to note that even though the linear strain measures are adopted for the computation of the internal forces, which results in a linear relation between the internal forces and the displacements, the dependence of the load stiffness on the structural displacements as well as the state dependency in the residual expression remain. Thus, an appropriate iterative solution procedure for the equation system is necessary, making the NR solution procedure in Sec. 2.1.5 favourable.

Finally, the software framework for the solution of the FSI problem exploits the capabilities of the in-house open-source software EMPIRE [38]. The considered algorithms in this work follow a partitioned solution procedure and utilize EMPIRE which makes use of the MPI communication for the data transfer between the disciplinary solvers. The algorithms mentioned in the following sections are variants of the *Gauß-Seidel* (GS) method. Thus, the respective flow of the realized communication patterns can be explained as depicted in Fig. 2.1. In addition to this, the implementations of the various mapping operators in EMPIRE enable the load and motion transfer -namely the coupling variables as in Eqn. (2.36)- at the coupling interface, see also [39], [40].

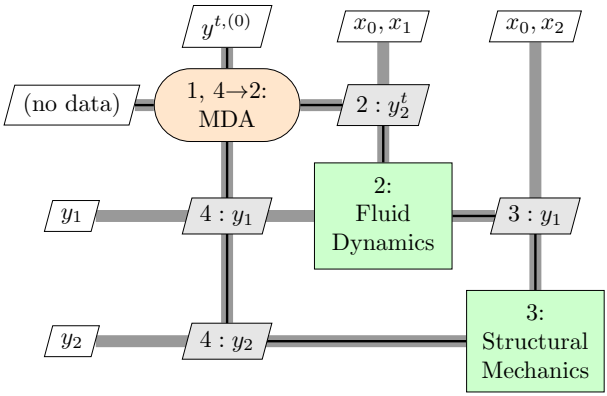


Figure 2.1: The Extended Design Structure Matrix (XDSM) [41] of the GS workflow.

2.3.1 Gauß-Seidel Method (GS)

Having the tangent stiffness matrices for the NR method defined in Sec. 2.3, one can apply simplifications on the solution procedure to obtain the GS method. For instance, in practice it is common to treat the individual solvers for the solutions of the structural and the fluid BVPs as black-box solvers. In such a case, one can not compute the contributions related to the linearization of the fluid tractions. Only the input and the output values of the individual software would be available. In a typical context, the Dirichlet-Neumann decomposition using the relations in Eqn. (2.36) are adopted. In addition to that, a relaxation factor for the applied boundary conditions can also be employed. For instance, when relaxation is applied on the Dirichlet conditions, the solution procedure reads:

$$\hat{R}^{\mathrm{FSI}}_{(i)} = \int_{\Omega} \frac{\partial \boldsymbol{\varepsilon}}{\partial \hat{d}_{(i)}} : \left(\boldsymbol{\sigma} + \boldsymbol{\sigma}_0\right) \mathrm{d}\Omega - \left\langle \boldsymbol{\phi}_{(i)}, \mathbf{t}^{\mathrm{f}}\left(\mathbf{d}_{n-1}\right) \right\rangle_{0,\Gamma_{\mathrm{FSI}}} , \tag{2.41a}$$

$$\widetilde{\frac{\partial \hat{R}_{(i)}}{\partial \hat{d}_{(j)}}} = \tilde{K}^{\mathrm{s}}_{(i,j)} = K^{\mathrm{int}}_{(i,j)} + \cancel{K^{\mathrm{f}}_{(i,j)}} , \tag{2.41b}$$

$$\Delta \hat{\mathbf{d}}_n = \left(\tilde{\mathbf{K}}^{\mathrm{s}}\right)^{-1} \hat{\mathbf{R}}^{\mathrm{FSI}} , \tag{2.41c}$$

$$\hat{\mathbf{d}}_{n+1} = \hat{\mathbf{d}}_n + \Delta \hat{\mathbf{d}}_n , \tag{2.41d}$$

$$\hat{\mathbf{d}}_{\Gamma_{\mathrm{FSI}},n+1} = \hat{\mathbf{d}}_{\Gamma_{\mathrm{FSI}},n} + \alpha_{GS} \Delta \hat{\mathbf{d}}_{\Gamma_{\mathrm{FSI}},n} . \tag{2.41e}$$

From this point of view, the GS solution procedure for the coupling of fluid-structure interaction is a Quasi-Newton method, where part of the Jacobian, namely the fluid contributions in this particular system, is neglected. Thus, the convergence rate is expected to be linear [42]. From the engineering experience, in the case of steady state FSI the relaxation parameter is not expected to improve the convergence rate when chosen $\alpha_{GS} < 1$ but only increases the stability of the fluid solution step. Additionally, one can adopt the Aitken relaxation method to estimate a relaxation factor in order to accelerate the solution procedure [43]. The GS fixed point iteration method is suitable when black-box solvers are the only available option for the solution of the FSI problem since this method does not necessitate the computation of second RHS term in Eqn. (2.22b), namely the coupling term, and only the results of the individual solvers are needed. This also facilitates the use of various efficient solution techniques for each BVP to be coupled, such as iterative or direct solution techniques.

2.3.2 Follower Load Extension to Gauß-Seidel Method (GS-FL)

In Sec. 2.3.1, the analogy of the GS method to Quasi-Newton methods is presented. In certain practical software environments, the tangent stiffness matrix of the structural problem related to its internal forces, namely the $\mathbf{K}^{\text{int}}$, is available to the user. In such a scenario, the contributions from the load stiffness can be added to the internal tangent stiffness matrix and the accuracy of the effective tangent stiffness of the FSI problem can be increased. The least complicated term that can be considered is the first RHS term of Eqn. (2.40c) since it contains only the derivative of a geometric quantity, i.e. the current surface normal $\mathbf{n}$. This term can also be considered as the follower load contribution to the left hand side (LHS) of the system in case a constant pressure load would have been applied. The information necessary for the computation of the derivative of the surface normal with respect to the nodal displacements is always available since the discretized wet surface as well as the shape functions used for the discretization are known to the user.

The surface normal $\tilde{\mathbf{n}}$ and the normalized surface normal $\mathbf{n}$ at a given surface coordinate in the current configuration reads:

$$\tilde{\mathbf{n}} = \mathbf{g}_1 \times \mathbf{g}_2 \,, \tag{2.42a}$$

$$\tilde{n} = \left\| \tilde{\mathbf{n}} \right\|_2 \,, \tag{2.42b}$$

$$\mathbf{n} = \frac{1}{\tilde{n}} \tilde{\mathbf{n}} \,, \tag{2.42c}$$

The $\mathbf{g}_\alpha$ denote the base vectors on the surface material coordinates at a deformed state $\mathbf{x} = \mathbf{X} + \mathbf{d}$. Then the first RHS term in Eqn. (2.22b) is straightforward to compute using the following relations:

$$\frac{\partial \mathbf{n}}{\partial \hat{d}_{(j)}} = \frac{1}{\tilde{n}} \frac{\partial \tilde{\mathbf{n}}}{\partial \hat{d}_{(j)}} - \frac{1}{\tilde{n}^2} \frac{\partial \tilde{n}}{\partial \hat{d}_{(j)}} \tilde{\mathbf{n}} \,. \tag{2.43}$$

where,

$$\frac{\partial \tilde{n}}{\partial \hat{d}_{(j)}} = \mathbf{n} \cdot \frac{\partial \tilde{\mathbf{n}}}{\partial \hat{d}_{(j)}} \,, \tag{2.44a}$$

$$\frac{\partial \tilde{\mathbf{n}}}{\partial \hat{d}_{(j)}} = \frac{\partial \mathbf{g}_1}{\partial \hat{d}_{(j)}} \times \mathbf{g}_2 + \mathbf{g}_1 \times \frac{\partial \mathbf{g}_2}{\partial \hat{d}_{(j)}} \,. \tag{2.44b}$$

Using the above inclusion, one can write the extended inexact Jacobian of the GS procedure in Eqn. (2.41b) as follows:

$$\widetilde{\frac{\partial \hat{R}_{(i)}}{\partial \hat{d}_{(j)}}} = \widetilde{K}^{\mathrm{s}}_{(i,j)} = K^{\mathrm{int}}_{(i,j)} + \widetilde{K}^{\mathrm{f}}_{(i,j)} \,, \tag{2.45}$$

with,

$$\widetilde{K}^{\mathrm{f}}_{(i,j)} = -\left\langle \boldsymbol{\phi}_{(i)}, \frac{\partial \mathbf{n}}{\partial \hat{d}_{(j)}} \cdot \boldsymbol{\sigma}^{\mathrm{f}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} - \cancel{\left\langle \boldsymbol{\phi}_{(i)}, \mathbf{n} \frac{\mathrm{d}\boldsymbol{\sigma}^{\mathrm{f}}}{\mathrm{d}\hat{d}_{(j)}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}}} \,. \tag{2.46}$$

The resulting inexact Jacobian is more accurate compared to the one of the GS method but still not expected to increase the convergence behaviour significantly. The significance of the terms in Eqn. (2.40c) can vary depending on the state since both the direction and the magnitude of the tractions acting on the structure depend on both terms. An advantage of the method is that the sparsity pattern of the tangent stiffness matrix remains unchanged, which maintains the necessary memory usage for the matrix storage. On the other hand, a disadvantage is the necessity of a linear solver that is capable of dealing with nonsymmetric matrices, which is a common occurrence when follower loads are to be considered and the resulting system Jacobian is nonsymmetric. In the extent of this thesis, this method is going to be abbreviated as GS-FL.

2.3.3 IBQNLS Extension to Gauß-Seidel Method (GS-IBQNLS)

The last term in the residual equation of the FSI problem in Eqn. (2.39), when formulated from the structural point of view, essentially contains the consistent nodal forces $\hat{\mathbf{f}}^{\mathrm{f}}$. Thus, the residual equation can be reformulated as follows:

$$\hat{R}^{\mathrm{FSI}}_{(i)} = \int_{\Omega} \frac{\partial \boldsymbol{\varepsilon}}{\partial \hat{d}_{(i)}} : \left(\boldsymbol{\sigma} + \boldsymbol{\sigma}_0\right) \mathrm{d}\Omega - \hat{f}^{\mathrm{f}}_{(i)} \,. \tag{2.47}$$

Then the contribution $\mathbf{K}^{\mathrm{f}}$ to the tangent stiffness matrix of the FSI problem defined in Eqn. (2.47) contains the derivatives of the consistent nodal forces $\hat{\mathbf{f}}^{\mathrm{f}}$ with respect to the nodal displacements or equivalently the nodal

positions:

$$K^{\mathrm{f}}_{(i,j)} = \frac{\mathrm{d}\hat{f}^{\mathrm{f}}_{(i)}}{\mathrm{d}\hat{x}_{(k)}} \frac{\mathrm{d}\hat{x}_{(k)}}{\mathrm{d}\hat{d}_{(j)}} = \frac{\mathrm{d}\hat{f}^{\mathrm{f}}_{(i)}}{\mathrm{d}\hat{x}_{(k)}} \delta_{(k,j)} = \frac{\mathrm{d}\hat{f}^{\mathrm{f}}_{(i)}}{\mathrm{d}\hat{x}_{(j)}} \,. \tag{2.48}$$

Since the exact computation of the term in Eqn. (2.48) is computationally inefficient, one can approximate it via an appropriate method. [44] compares a wide variety of approximation methods. Herein, a least-squares approximation is favoured following the methodology in [45], which is named as IBQNLS in [46], hence the name IBQNLS extension. The least-squares approximation requires inputs and outputs; $\hat{\mathbf{x}}$ and $\hat{\mathbf{f}}^{\mathrm{f}}$ respectively. For this purpose, one can adopt the series of fields $\hat{\mathbf{x}}_n$ and $\hat{\mathbf{f}}^{\mathrm{f}}_n$ that occur during each solution iteration n. Then the input and output modes can be defined as $\boldsymbol{\Delta}\hat{\mathbf{x}}_n = \hat{\mathbf{x}}_n - \hat{\mathbf{x}}_{n-1}$ and $\boldsymbol{\Delta}\hat{\mathbf{f}}^{\mathrm{f}}_n = \hat{\mathbf{f}}^{\mathrm{f}}_n - \hat{\mathbf{f}}^{\mathrm{f}}_{n-1}$. The input and output modes are then grouped in the following matrices:

$$\mathbf{V}_n = \left[\boldsymbol{\Delta}\hat{\mathbf{x}}_1, \boldsymbol{\Delta}\hat{\mathbf{x}}_2, \cdots, \boldsymbol{\Delta}\hat{\mathbf{x}}_{n-1}\right] , \tag{2.49a}$$

$$\mathbf{W}_n = \left[\boldsymbol{\Delta}\hat{\mathbf{f}}^{\mathrm{f}}_1, \boldsymbol{\Delta}\hat{\mathbf{f}}^{\mathrm{f}}_2, \cdots, \boldsymbol{\Delta}\hat{\mathbf{f}}^{\mathrm{f}}_{n-1}\right] . \tag{2.49b}$$

Having the above matrices defined, the approximated load and the effective tangent stiffnesses can be calculated from:

$$\widetilde{\mathbf{K}}^{\mathrm{f}}_n = -\mathbf{W}_n \left(\mathbf{V}_n^T \mathbf{V}_n\right)^{-1} \mathbf{V}_n^T \,, \tag{2.50a}$$

$$\widetilde{\mathbf{K}}^{\mathrm{s}}_n = \mathbf{K}^{\mathrm{int}} + \widetilde{\mathbf{K}}^{\mathrm{f}}_n \,. \tag{2.50b}$$

The rest of the solution procedure follows from Eqn. (2.25). The advantage of the presented method is the improving accuracy of the approximated quantities as the solution iterations advance. Accordingly, the convergence behaviour is expected to improve as more input and output modes are available. On the other hand, the resulting system Jacobian $\widetilde{\mathbf{K}}^{\mathrm{s}}$ is a non-symmetric matrix and the contributions related to $\widetilde{\mathbf{K}}^{\mathrm{f}}$ are assembled to the interface DOFs. Thus, the resulting Jacobian matrix for the structural models with large number of interface DOFs are densely populated too. Moreover, it is important to note that even if the exact Jacobian contribution would have been available, it would have similarly populated the matrix, since a perturbation on a part of the FSI interface would cause a flow perturbation on the whole fluid domain. In case of least-squares

approximation methods, it can be favourable to employ methods that approximate the inverse of the Jacobian for the fixed point iterations, such as IQNILS [46]. Within the scope of this work, this method is included for the demonstrations on the convergence behaviour of the coupling algorithms due to the similarity to the proposed method presented in Sec. 2.3.4. In the rest of this work, this method is going to be abbreviated as GS-IBQNLS where necessary.

2.3.4 Follower Load and IBQNLS Extensions to Gauß-Seidel Method (GS-FL-IBQNLS)

In Sec. 2.3.2, the follower load extension to the inexact Jacobian of the GS method is presented and the possibility of employing a least-squares approach for the approximation of the whole load stiffness term is discussed in Sec. 2.3.3. In addition to the follower load term, the relatively more complicated part related to the derivative of the fluid stresses with respect to the nodal displacements, second RHS term in Eqn. (2.40c), can be approximated via the least-squares approach similar to Sec. 2.3.3.

The inputs and outputs for the approximation of the second RHS term in Eqn. (2.40c) are defined as $\hat{\mathbf{d}}$ and $\boldsymbol{\sigma}^{\mathrm{f}}$ while the input-output modes are defined as $\Delta\hat{\mathbf{x}}_n = \hat{\mathbf{x}}_n - \hat{\mathbf{x}}_{n-1}$ and $\Delta\boldsymbol{\sigma}^{\mathrm{f}}_n = \boldsymbol{\sigma}^{\mathrm{f}}_n - \boldsymbol{\sigma}^{\mathrm{f}}_{n-1}$ respectively. It is important to note that the stress tensors are computed at the respective Gauß points, which are at coupling surface element centroids of the fluid FVM discretization. Herein, two important modifications are made in order to reduce the computational effort and the storage requirements. Firstly, the outputs $\boldsymbol{\sigma}^{\mathrm{f}}_n$ are stored using a Voigt notation since the fluid stress tensors are symmetric by definition, see Eqn. (2.35a). Secondly, the output modes $\Delta\boldsymbol{\sigma}^{\mathrm{f}}_n$ are premultiplied with the current surface normals $\mathbf{n}$ and preintegrated for the generation of the matrix $\mathbf{W}_n$ yielding vector fields defined at nodes similar to the $\Delta\hat{\mathbf{f}}^{\mathrm{f}}_n$ of Sec. 2.3.3. The entry k of the modified output modes $\mathbf{W}$ at iteration n is computed in the following way:

$$\mathbf{W}_{n,(k)} = \left\langle \boldsymbol{\phi}, \mathbf{n} \cdot \Delta\boldsymbol{\sigma}^{\mathrm{f}}_n \right\rangle_{0,\Gamma_{\mathrm{FSI}}} . \tag{2.51}$$

The above definition of the matrix for the output modes eases the assembly and reduces the memory usage during the computations necessary for the approximated Jacobian. The matrix $\mathbf{V}_n$ that contains the input modes

remains unchanged. Having the matrices defined, the inexact Jacobian of the system, or namely the tangent stiffness, with the approximated load stiffness can be computed from:

$$\widetilde{\mathbf{K}}_n^{\mathrm{f}} = -\left\langle \boldsymbol{\phi}, \frac{\partial \mathbf{n}}{\partial \hat{\mathbf{d}}} \cdot \boldsymbol{\sigma}^{\mathrm{f}} \right\rangle_{0,\Gamma_{\mathrm{FSI}}} - \mathbf{W}_n \left(\mathbf{V}_n^T \mathbf{V}_n \right)^{-1} \mathbf{V}_n^T \,, \tag{2.52a}$$

$$\widetilde{\mathbf{K}}_n^{\mathrm{s}} = \mathbf{K}^{\mathrm{int}} + \widetilde{\mathbf{K}}_n^{\mathrm{f}} \,. \tag{2.52b}$$

The approximation procedure defined in this section is expected to contain properties of the both methods presented in Sec. 2.3.2 and Sec. 2.3.3. The accuracy of the approximated tangent is higher since more terms are computed exactly. While keeping the exact geometric contributions from the load stiffness, the solution procedure is expected to have an improving convergence behaviour due to the improving accuracy in the approximated part of the system Jacobian. As expected, the assembled system Jacobian is similarly populated on the interface DOFs and constitutes a nonsymmetric matrix as in Sec. 2.3.3. In the extent of this work, this method is going to be referred as GS-FL-IBQNLS.

2.4 Results and Discussion on the Dynamic Analysis

This section consists of the demonstrations for the developed dynamic analysis methods in an order of increasing complexity. Foremost, the results of the updated linear transient and the updated modal analyses are presented. For a simpler but a comprehensive example, the dynamic analysis of a SDOF nonlinear spring-mass system is considered. In addition, the performance of the methods are also investigated on a beam model. Furthermore, the developed methods are applied to a real-life scenario which also contains the results regarding the project Mistralwind [47] in combination with experimental results that facilitated the validation of the implemented methods.

2.4.1 Nonlinear Spring - Mass System

For clarity, the method is presented on a nonlinear spring-mass system with the spring stiffness containing a constant and a state dependent part:

$$k(d) = k_{\text{const}} + d^2 \,. \tag{2.53}$$

Accordingly the internal force of the spring and the steady state residual are:

$$F_{\text{int}}(d) = k(d)\,d = k_{\text{const}} d + d^3 \,, \tag{2.54}$$

$$R(d) = F_{\text{int}}(d) - F_{\text{sta}} \,. \tag{2.55}$$

Moreover, the static tangent stiffness can be calculated from:

$$\frac{\partial R}{\partial d} = k^{\text{s}}(d) = k_{\text{const}} + 3d^2 \,. \tag{2.56}$$

Following from Eqn. (2.53), Eqn. (2.54) and Eqn. (2.56), the state dependent spring stiffness, internal force and the tangent stiffnesses of the nonlinear spring are plotted in Fig. 2.2. Having the steady state residual and the tangent stiffness defined, the dynamic residual and the stiffness of the dynamic problem for a *Backward Difference scheme* (BDF) [48] can be computed from:

$$R^{\text{d}}_{t_s} = m\ddot{d} + c\dot{d} + R(d) - F_{\text{dyn}} \,, \tag{2.57}$$

$$\frac{\partial R^{\text{d}}}{\partial d} = K_{t_s} = \frac{m}{\Delta t^2} + \frac{c}{\Delta t} + k^{\text{s}}\left(d_{t_s}\right) \,, \tag{2.58}$$

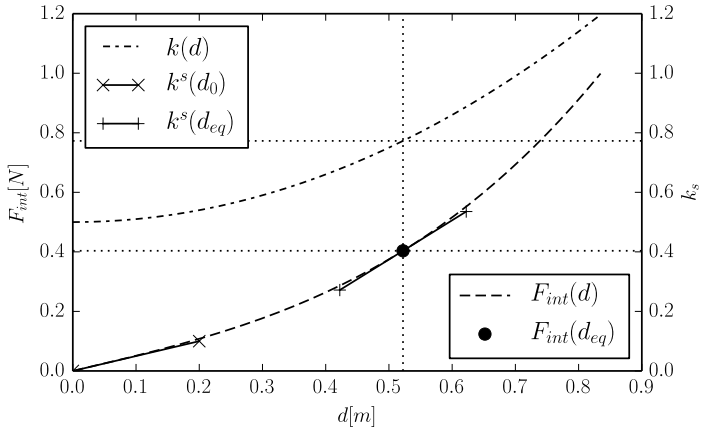


Figure 2.2: The internal force-displacement curve for the nonlinear spring with $k_{\text{const}} = 0.5$ kg/s^2 and applied pseudo equilibrium load $F_{\text{eq}} = 0.404$ N.

where m and c denote the mass and the damping parameter of the problem. In addition, the dynamic residual and the effective stiffness for an initially nonlinear but linear in transient regime analysis with the assumed structural stiffness at the pseudo equilibrium point is obtained as:

$$R^{\text{d}}_{t_s} = m\ddot{d} + c\dot{d} + k^{\text{s}}\left(d_{e\text{q}}\right)d - F_{\text{dyn}}\,, \tag{2.59}$$

$$\frac{\partial R^{\text{d}}}{\partial d} = K_{t_s} = \frac{m}{\Delta t^2} + \frac{c}{\Delta t} + k^{\text{s}}\left(d_{\text{eq}}\right)\,. \tag{2.60}$$

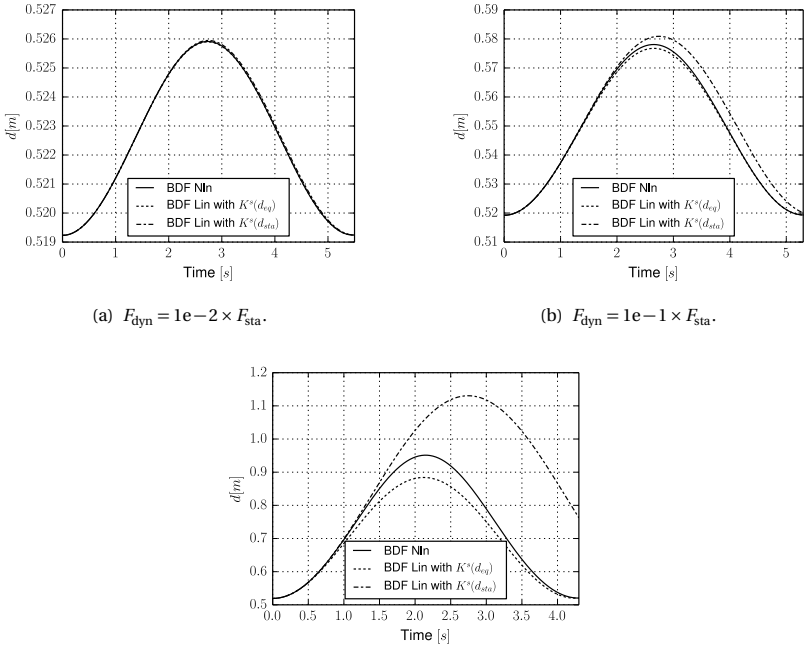
Finally, the simplest case of assuming statically nonlinear but linear in transient regime gives the following residual and effective stiffness expressions:

$$R^{\text{d}}_{t_s} = m\ddot{d} + c\dot{d} + k^{\text{s}}\left(d_{\text{sta}}\right)d - F_{\text{dyn}}\,, \tag{2.61}$$

$$\frac{\partial R^{\text{d}}}{\partial d} = K_{t_s} = \frac{m}{\Delta t^2} + \frac{c}{\Delta t} + k^{\text{s}}\left(d_{\text{sta}}\right)\,. \tag{2.62}$$

The transient simulations with an initial static loading for the chosen case parameters; time step size $\Delta t = 1\text{e}{-3}$ s, mass $m = 1.0$ kg, damping $c = 0.0$

kg/s, constant stiffness $k_{\text{const}} = 0.5$ kg/s^2 and static loading $F_{\text{sta}} = 0.4$ N, are performed and the comparisons are presented in Fig. 2.3. The fully nonlinear case can be considered as the numerical reference while the remaining are the approximations of the case.



(a) $F_{\text{dyn}} = 1\text{e}{-2} \times F_{\text{sta}}$.

(b) $F_{\text{dyn}} = 1\text{e}{-1} \times F_{\text{sta}}$.

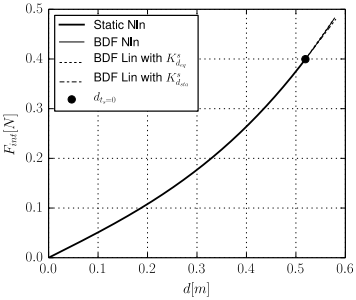
(c) $F_{\text{dyn}} = F_{\text{sta}}$.

Figure 2.3: The displacement-time series of the nonlinear spring-mass system with the chosen parameters for varying dynamic loading conditions.

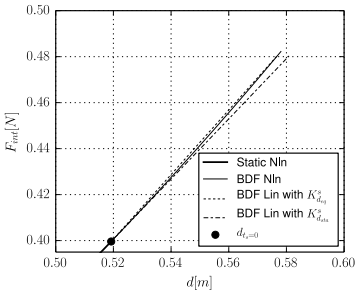
The figure Fig. 2.3 reveals how the method performs under several dynamic load cases with increasing ratio with respect to the static loading. As the dynamic load becomes less prominent, the displacement solutions of the updated linear transient method converge to the nonlinear transient solution, since the stiffness approximation becomes more accurate. In case of employing the tangent stiffness of the static equilibrium point, the

lack of accuracies in both the maximum displacement and the period of the vibrations are clearly evident. On the other hand, when the tangent stiffness of the pseudo equilibrium point is adopted, the period of the vibrations is kept close to the nonlinear case while the maximum displacement slightly differs from the reference case. The reason lies clearly in the employed assumption for the constant static tangent stiffness matrix, see Fig. 2.4. The tangent stiffness of the static equilibrium point is only valid at the beginning of the transient phase, which neglects the varying stiffness properties. As a result both the maximum ocurring displacements and the vibration period is estimated incorrectly. On the other hand, adopting the tangent stiffness matrix of the pseudo equilibrium point implies an "averaged" value of the stiffness throughout the simulation resulting in a more accurate estimation of the vibration period. However, due to the higher tangent stiffness of the pseudo equilibrium point, the resulting displacements for a corresponding loading is less than that of the fully nonlinear case. In a general case, the latter can either overestimate or underestimate the maximum or minimum displacements, since the stiffening or softening behaviour depends on the design of the structure. In Fig. 2.4(b) and Fig. 2.4(d), one can clearly observe that the displacements follow the chosen tangent stiffness direction after the initial nonlinear static analysis. In addition, the performance of the updated modal analysis method can be discussed. The reference equilibrium point $\hat{\mathbf{d}}_{\text{ref}}$ can be chosen either as the static equilibrium $\hat{\mathbf{d}}_{\text{sta}}$ or the pseudo equilibrium $\hat{\mathbf{d}}_{\text{eq}}$ which is defined by the load case in Eqn. (2.28a). As a reference, one can apply *Discrete Fourier Transformation* (DFT) to the displacement series of the nonlinear transient analysis. The displacement solution is generated by computing the transient response of the system using the nonlinear BDF scheme. The comparisons of the modal and the DFT analyses for various load cases are presented in Table 2.1.

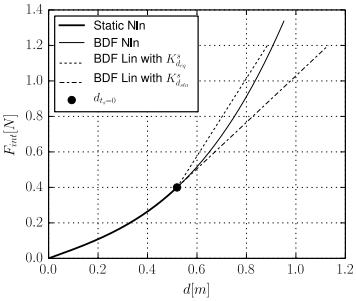
The results in Table 2.1 reveal that the measured eigenfrequencies of forced vibrations differ depending on the load case in the transient regime and assuming that the vibrations occur around the static equilibrium point is not enough while the transient loading might become more prominent compared to the static loading. The proposed method allows the adjustment of the stiffness properties depending on the transient load case. This facilitates the accurate computation of the frequencies under forced vibrations. This is in particular important in the design phase of structures



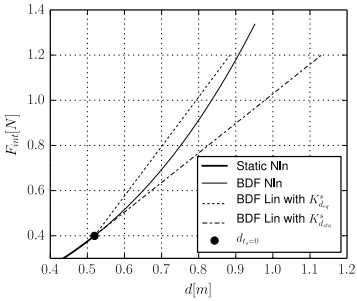
(a) The static and the dynamic phases for $F_{\text{dyn}} = 1\text{e}{-1} \times F_{\text{sta}}$.



(b) The close up view of the dynamic phase for $F_{\text{dyn}} = 1\text{e}{-1} \times F_{\text{sta}}$.



(c) The static and the dynamic phases for $F_{\text{dyn}} = F_{\text{sta}}$.



(d) The close up view of the dynamic phase for $F_{\text{dyn}} = F_{\text{sta}}$.

Figure 2.4: The internal force-displacement curves of the nonlinear spring-mass system for varying dynamic loading conditions. The curves are restricted to the maximum displacements that occur during each respective simulation.

	Impulse	$F_{\text{dyn}} = 1\text{e}{-2} \times F_{\text{sta}}$	$F_{\text{dyn}} = 1\text{e}{-1} \times F_{\text{sta}}$
DFT Nln BDF	0.1821	0.1827	0.1887
f_{lin}	0.1125	0.1125	0.1125
$f_{\text{upd}}(d_{\text{sta}})$	0.1821	0.1821	0.1821
$f_{\text{upd}}(d_{\text{eq}})$	0.1821	0.1827	0.1885

Table 2.1: The comparison of the computed eigenfrequencies.

which operate under known loading conditions. In case a structure is designed to operate in tight limit conditions, its frequency response under the operational loads is crucial. Moreover, if the measurements on the real-world structures are to be used as a reference to validate the quality of the built FEM models, the proposed method facilitates ensuring the higher accuracy of the numerical computations compared to the conventional modal analysis.

2.4.2 Steel Rod

In order to verify the assumptions and the eigenfrequency computation methods introduced in Sec. 2.1.6 and Sec. 2.1.7, a simple beam model with a circular cross section is chosen. A numerical reference case is prepared using a Newmark time integration scheme for geometrically nonlinear structures. The results of the transient analysis for geometrically linear structures, effects of the initial nonlinearity on the almost linear deformations in the transient regime as well as conventional and updated eigenfrequency analyses are compared against the numerical reference case. The accuracy comparison of the transient simulations are achieved by tracking the tip displacements of the beam. The tip displacements are then analysed for their eigenfrequency content using DFT. The computed eigenfrequencies from the applied DFT on the results of the nonlinear Newmark time integration scheme also form the reference values for the comparisons to be made.

The beam model considered in this section is a steel rod with a circular cross section. The length of the rod is 1 m and the diameter is chosen to be 10 cm. The FE-Modeling of the structure is achieved by 10 beam elements based on the Timoschenko beam theory that follows the implementation

with respect to [22], [49], [50]. The material and element properties of the modeling are listed in Table 2.2. The shear correction factor is computed following the literature [51]. The static loading on the structure of inter-

Parameter	**Value**	[**Units**]
E	2e+11	$[N/m^2]$
ν	0.26	[–]
ρ	7800.0	$[kg/m^2]$
A	7.853981634e−3	$[m^2]$
I_{yy}, I_{zz}	4.908738521e−6	$[m^2]$
κ_y, κ_z	1.11	[–]

Table 2.2: The material and element parameters for the FE-Modeling.

est is an axial compressive loading of 1 MN at the tip. The conventional and the updated eigenfrequency analyses of the structure under the mentioned load case yield the presented eigenfrequencies in Table 2.3. The eigenmodes and the differences of regular and updated eigenmodes of the structure are presented in Fig. 2.5 and Fig. 2.6 respectively.

Eigenmode	**Eigenfrequencies**, f	**Updated Eigenfrequencies** , f^*
1st Bending	7.045266e+1	5.505889e+1
2nd Bending	4.277413e+2	4.242713e+2
3rd Bending	1.142314e+3	1.201759e+3

Table 2.3: The computed eigenfrequencies.

The free vibrations under the given prestressing condition can be investigated by applying an impulse load on the structure and performing a transient analysis. In order to capture the pronounced eigenfrequencies in the transient analysis, one can choose the time step such that the period of the highest interested eigenfrequency is sufficiently resolved. In this case, the time step size is chosen such that first three bending modes are represented correctly. The chosen time step size for the transient simulations is

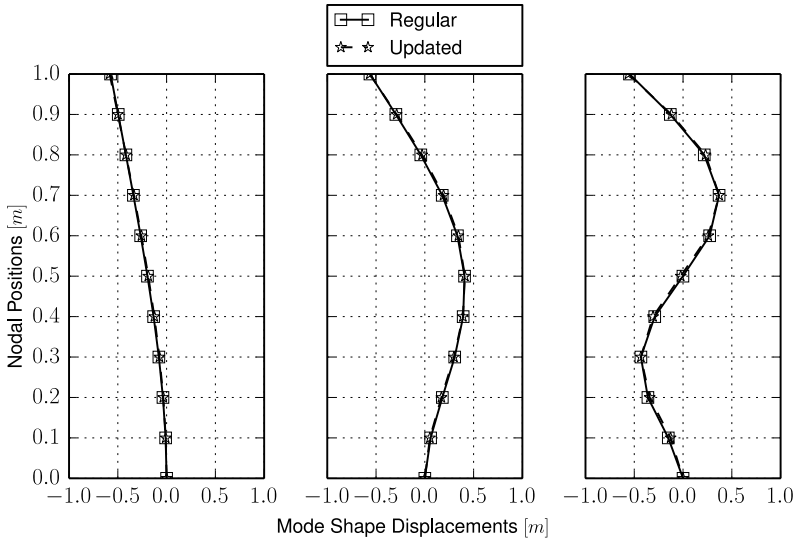


Figure 2.5: Computed eigenmodes. Each eigenmode is normalized by its norm.

$\Delta t = 1\text{e}{-5}$ s and the simulation is run for 1 second. The displacement plots can be observed in Fig. 2.7(a). The effect of the constant axial loading on the structure is evident. The axial compressive loading causes a softening of the structure and reduces the first eigenfrequency and extends the corresponding period. In addition to this, it can be observed that the maximum displacements are also increased. This effect would be particularly critical in case of a fatigue analysis of the structure, since both the resulting maximum and minimum stresses and the number of stressing cycles in a given interval are modified. In this distinct load case, the impulse loading does not cause geometrically nonlinear deformations of the structure. Thus, the nonlinear Newmark and linear Newmark schemes yield coincident results. On the other hand, when the initial nonlinearity of the structure is neglected, the error in the displacements grow over time. See Fig. 2.7(b).

The numerical verification of the updated eigenfrequency analysis can be achieved through the DFT application on the tip displacements of different

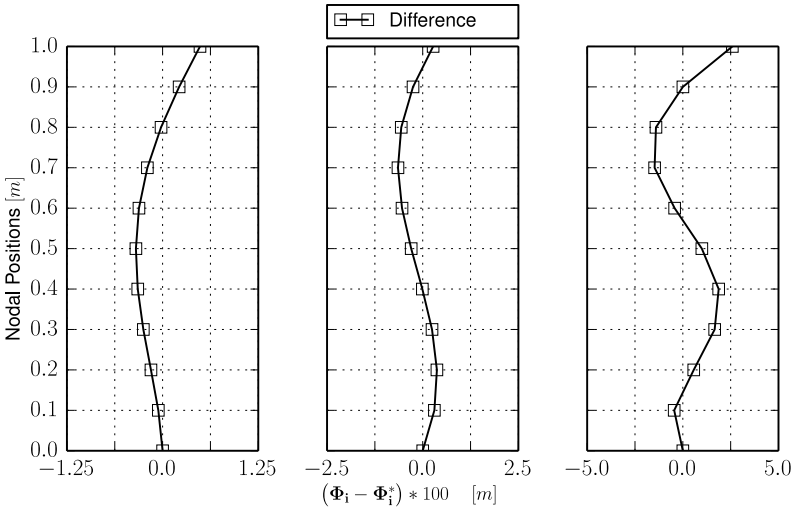


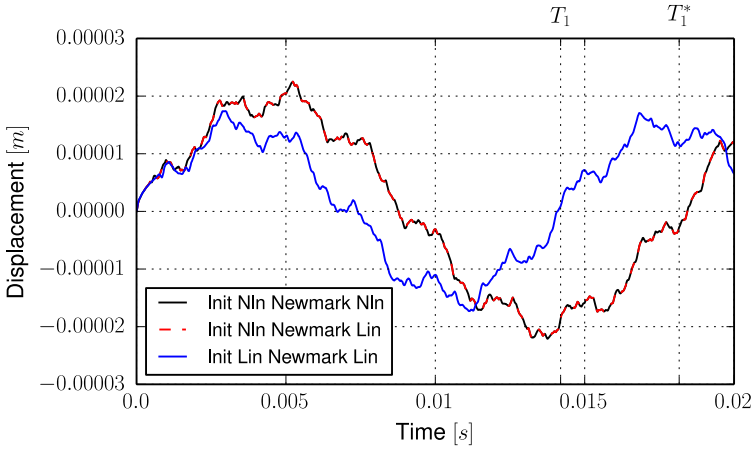
Figure 2.6: Differences in computed eigenmodes.

analyses. It can well be observed that the eigenfrequency content of the initially geometrically nonlinear but geometrically linear in the transient analysis matches exactly with the one of the nonlinear Newmark scheme. Moreover, it is also verified that the updated eigenfrequency analysis complies with these results in all of the considered eigenfrequencies. Conversely, while the eigenfrequencies of the linear Newmark scheme follows the conventional eigenfrequency analysis, the discrepancy and the shift with respect to the updated one are clearly visible in all of the investigated eigenfrequencies.

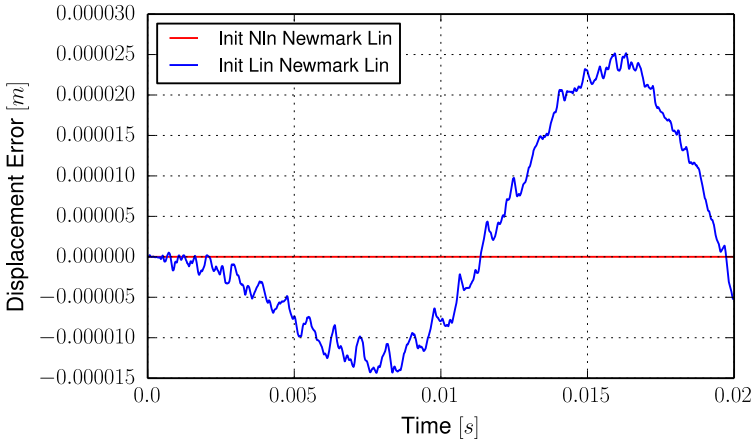
In addition to the free vibrations, the forced vibrations of the model are investigated. Unlike the impulse load in the free vibration study, a horizontal load with a magnitude of $100000N$ starting from $t = 0.00002$s is applied at the tip and kept constant until the end of the simulation. Three different transient simulations with the same procedures as the free vibration case are run for 1 second real time. Time step size is chosen according to the eigenfrequency analysis as $\Delta t = 1\text{e}{-5}$ s. The occurring displacements and the errors in displacements with respect to the nonlinear Newmark scheme

are plotted in Fig. 2.9(a) and Fig. 2.9(b) respectively. The first glance on the displacement plots reveals the importance of the initial geometrically nonlinear simulation. Even if the nonlinearity in the transient regime is neglected, the simulation yields complying results with a fully nonlinear transient simulation. This is mainly due to the prominence of the static loading's effects on the structural stiffness. The linearity assumption introduced in the transient regime does not impair the quality of the results compared to a fully nonlinear transient analysis even though the maximum horizontal displacements reach approximately 11.5% of the beam length. In contrast, fully linear analysis yield maximum horizontal displacements of approximately 6.5% of the beam length and is much lower than that of the reference analysis case. In addition to that, the frequency content of the response to a forced vibration are also investigated. It can well be observed that the DFT analysis of the occurring displacements yield the same eigenfrequencies as the free vibration case and the adopted eigenfrequency analysis method, see also Fig. 2.8 and Fig. 2.10.

The accuracy in the displacements-time series and the matching frequency content of the forced vibrations pave the way for the use of the updated eigenmodes in other analysis methods; such as modal transient analysis and fatigue hot-spot detection.

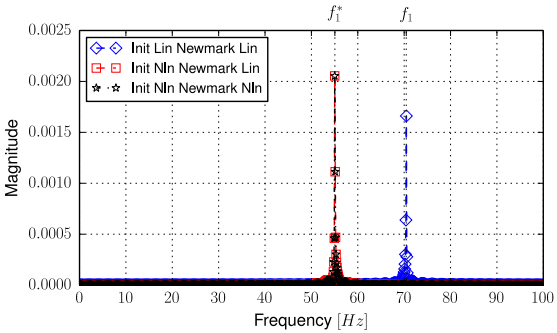


(a) Horizontal displacements at beam tip for different simulation methods (T_1 = 0.0142 s, T_1^* = 0.0182 s).

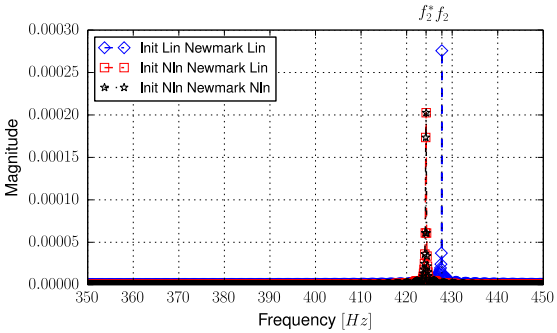


(b) Error in horizontal displacements with respect to the Newmark nonlinear time integration scheme.

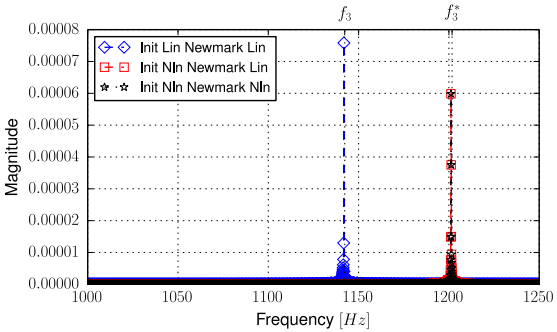
Figure 2.7: The comparison of the computed beam tip displacements for impulse loading.



(a) First eigenfrequencies ($f_1 = 70.45$ Hz, $f_1^* = 55.06$ Hz).

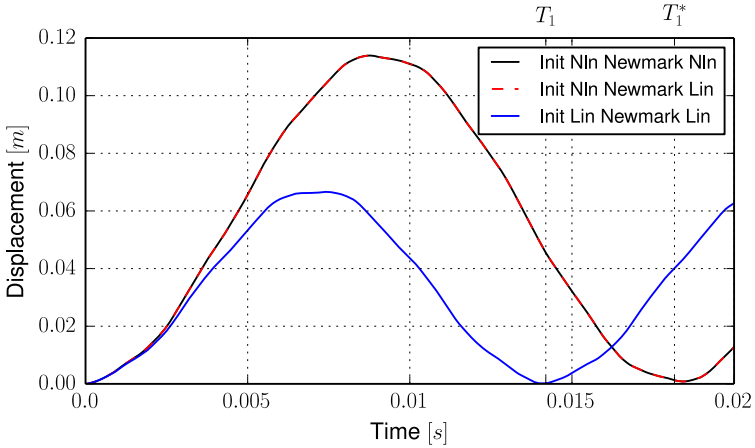


(b) Second eigenfrequencies ($f_2 = 427.74$ Hz, $f_2^* = 424.27$ Hz).

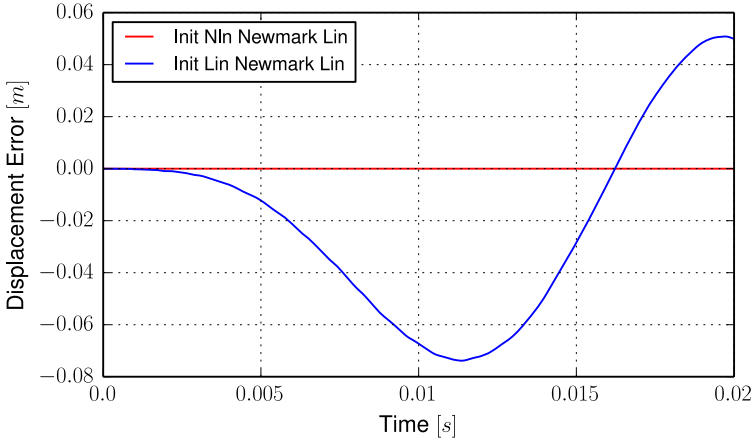


(c) Third eigenfrequencies ($f_3 = 1142.31$ Hz, $f_3^* = 1201.76$ Hz).

Figure 2.8: Comparison of the computed eigenfrequencies by applying DFT on the displacement results of free vibrations and eigenfrequency analyses.

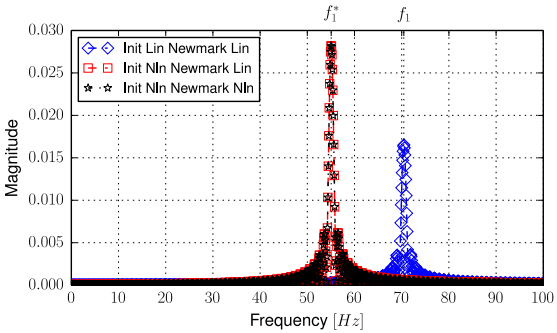


(a) Horizontal displacements at beam tip for different simulation methods (T_1 = 0.0142 s, T_1^* = 0.0182 s).

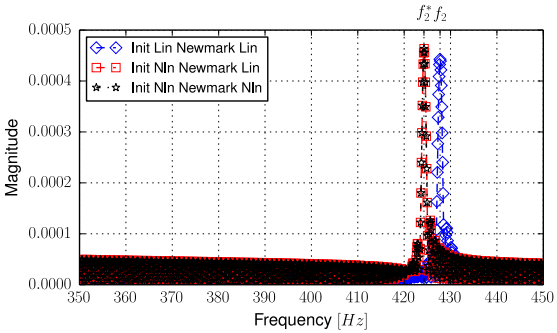


(b) Error in horizontal displacements with respect to the Newmark nonlinear time integration scheme.

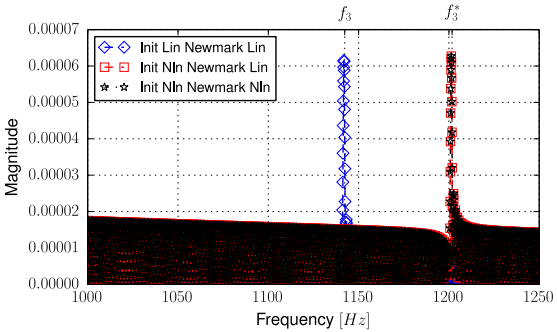
Figure 2.9: The comparison of the computed beam tip displacements for impulse loading.



(a) First eigenfrequencies ($f_1 = 70.45$ Hz, $f_1^* = 55.06$ Hz).



(b) Second eigenfrequencies ($f_2 = 427.74$ Hz, $f_2^* = 424.27$ Hz).



(c) Third eigenfrequencies ($f_3 = 1142.31$ Hz, $f_3^* = 1201.76$ Hz).

Figure 2.10: Comparison of the computed eigenfrequencies by applying DFT on the displacement results of forced vibrations and eigenfrequency analyses.

2.4.3 Hybrid (Steel-Concrete) Wind Turbine Tower (Project Mistralwind)

The supporting structures of the wind turbines are designed and built for an intended lifetime of 20 years. In the near future, a vast number of the firstly built wind turbine towers are going to come to the end of their designed lifespan but the actual remaining lifetime is ambiguous since the assumptions stemming from the design considerations might not be accurate for each particular wind turbine supporting structure. The design assumptions can either overestimate or underestimate the real conditions that occur within the service life of the supporting structures. The project Mistralwind [47] concerns with the estimation of the current state and the remaining lifetime of the load bearing structures. In this context, the supporting structure refers to the tower structure that is a steel-concrete hybrid construction. Within the consortium of the project, the tasks related to the Chair of Structural Analysis (Statik-TUM) can be summarized as the development of a suitable FEM model of the load bearing structure, the development and the implementation of robust simulation methods that can be applied to large scale structures and the fatigue hot-spot detection. In this section, the CAD and the FEM modeling of the provided hybrid tower is firstly presented. Then the validation of the FEM model by means of the measurement data is accomplished. The constructed model is later on utilized for the simulation methods described previously in Sec. 2.1.

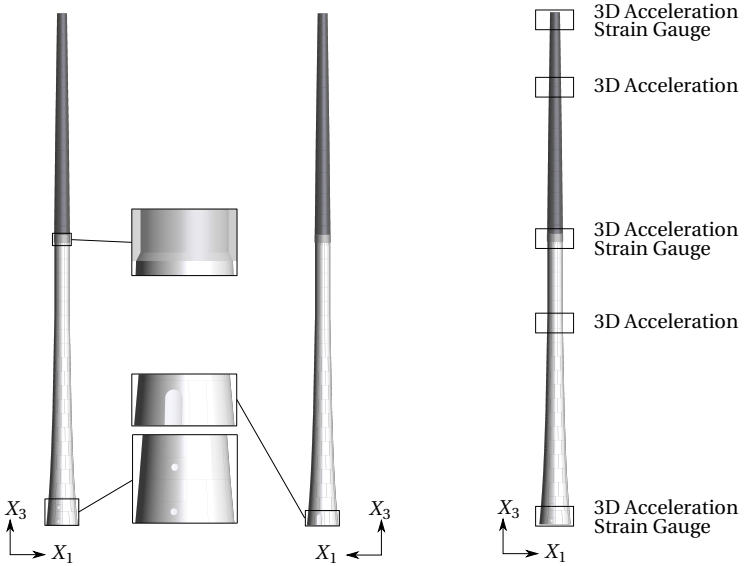
Modeling

The tower structure that is provided within the frame of the project was designed and built by the project partner Max Bögl Wind AG. The structure consists of two tubular shafts which are made of high-performance concrete and steel sections. A special adapter connection is used between the concrete and the steel shafts. This part is also used as an anchor for the external prestressing cables that are spanned between the fundament and the adapter. The prestressing cables ensure that the concrete shaft remains under compressive loading. The technical drawings as well as the material properties of the tower are delivered by the owner party Max Bögl Wind AG [52] under the non-disclosure agreement of the respective project partners. Thus, the modeling of the tower is presented without giving details on the dimensioning and the material properties.

Firstly, a CAD model based on the technical drawings of the provided tower structure is constructed. Due to the relatively thin wall thickness of the hollow structure, shell elements based on the Reissner-Mindlin kinematics are preferred for the FEM modeling and a corresponding mid-surface modeling for the CAD model. The CAD and FEM modelings of the regular sections of the tower are straight-forward. However, a rather more detailed approach is necessary for the adapter piece since it both acts as a connection part between the concrete and the steel shafts and an anchor for the external prestressing cables. The FEM modeling of this piece using the volume elements requires a vast amount of elements, i.e. several times the number of elements necessary for the rest of the tower structure, due to the detailed features. In order to reduce the computational effort, a shell modeling of this piece is also favoured. Thus, the adapter piece is modelled in such a way that the loads are transferred correctly between the concrete and the steel shafts and the mass properties are correctly represented. In addition, the connection between the various components at this location are achieved by utilizing *rigid body elements* (RBE). The modeling of the prestressing cables is another challenge. The form of the cables depends on the prestressing force, the self weight and the tower structure as they touch the concrete shaft's inner walls. This requires a form-finding step for the cables and a contact based modeling which brings high nonlinearity and complexity in the solution procedures. Instead, a simplification is made by applying the given final force acting on the prestressing cables as a distributed load around the adapter piece. Next, the rotor mass is attached at the tip of the tower using RBEs with the provided mass properties. Unfortunately, the rotational inertia properties of the turbine blades are not available and are not modelled. Finally, as the ground material properties are not given, a fixed support is applied at the tower root. In case the actual ground material is not as stiff as assumed, this causes an increase in the computed eigenfrequencies and errors in the computed transient response. On the other hand, on-site observations indicated that the ground material is stiff enough to assume a fixed support for the fundament-tower connection, refer also to [53] for the ground material type eigenfrequency relations. Finally, the material modeling of the concrete segments that are comprised in the concrete shaft is achieved by the orthotropic material law due to the various volume fractions of reinforcing steel in radial, circumferential and the tower axial directions. Another point to be mentioned is that each concrete segment

of the tower is split into 8 additional subsections in the circumferential direction for further localized analysis. The CAD modeling of the tower after the introduced assumptions is depicted in Fig. 2.11(a). Further details in the modeling considerations can be found in the final project report of Mistralwind-Workpackage 5 [54].

Furthermore, within the frame of the project, the structure is instrumented with a variety of sensors that capture the dynamic characteristics. The instrumentation of the tower and the post-processing of the collected data are carried out by the Chair of Non-destrucrive Testing (ZfP-TUM). The collected data is then used for the validation of the generated FEM model. The corresponding sensor placements are depicted in Fig. 2.11(b).



(a) The CAD modeling of the tower with the structural details.

(b) The sensor placement.

Figure 2.11: The CAD modeling of the tower with the structural details.

Validation

First of all, the validation of the constructed FEM model is to be accomplished by means of the collected sensor data. The sensor data gives accurate estimation of the dynamic characteristics such as eigenfrequencies. In addition, the measurement data indicates that the dynamic part of the displacements under the operational loads remain less than 0.025% of the tower height, which enables the assumption that the dynamic part of the loading does not cause significant geometric nonlinearities and the proposed method in Sec. 2.1.7 is appropriate for the modal analysis. Thus, a modal analysis is carried out using both the conventional linear modal analysis and the proposed updated modal analysis methods. The results of the considered methods and the comparison to the measurement data listed in Table 2.4 reveal that the updated modal analysis method greatly increases the accuracy in the eigenfrequency computation compared to the conventional modal analysis method. The conventional modal analysis yields eigenfrequencies with approximately 9% error in the first bending frequency, whereas the updated modal analysis method yields eigenfrequencies with less than 1% error. The transient simulations of fairly long time periods require a robust simulation method that enables accurate es-

Operational Modal Analysis [55]		**Conventional Modal Analysis**		**Updated Modal Analysis**	
Eigenmode	$f\ [Hz]$	Modeshape	$f\ [Hz]$	Modeshape	$f^*\ [Hz]$
Fore-Aft 1	0.274	Door $\parallel$ 1	0.299	Door $\parallel$ 1	0.276
Side-Side 1	0.277	Door $\perp$ 1	0.299	Door $\perp$ 1	0.277
Fore-Aft 2a Fore-Aft 2b	1.063 1.184	Door $\parallel$ 2	1.201	Door $\parallel$ 2	1.148
Side-Side 2	1.100	Door $\perp$ 2	1.207	Door $\perp$ 2	1.154
Fore-Aft 3	3.410	Door $\parallel$ 3	3.565	Door $\parallel$ 3	3.524
Side-Side 3	3.333	Door $\perp$ 3	3.580	Door $\perp$ 3	3.540

Table 2.4: The comparison of the measured and the computed eigenfrequencies. The directions of the bending modes are denoted by $\parallel$ parallel to the door and $\perp$ perpendicular to the door direction respectively.

timations of the stressing states. In this particular case, where the structure consists of a vast amount of DOFs, the nonlinear transient simulations are not favourable due to the necessary computational effort. Instead, a modified linear transient analysis method as described in Sec. 2.1.6 is sought.

Fatigue Hot-Spot Detection

Another task in the frame of Mistralwind is the hot-spot detection for the following fatigue analysis that is performed by the Chair of Materials Science and Testing (CBM-TUM). The established FEM model consists of vast amount of elements which generate a high amount of data as a result of the transient simulations that should be fed into the fatigue analysis. The fatigue analysis of every individual element is in this sense both computationally inefficient and requires high data storage capacity. Instead, the critical regions for the fatigue analysis should be detected and the corresponding elements should be analysed for the applied load cases. Firstly, it should be noted that the transient simulations are carried out making use of the updated linear transient analysis method described in Sec. 2.1.6. Furthermore, since the static and the dynamic phases are split and even though the static phase is computed by means of nonlinear strain measures, the stress states of these can also be considered as a linear combination of the static stresses and the additional dynamic stresses:

$$\boldsymbol{\sigma}\left(\hat{\mathbf{d}}_{t_s}\right) = \boldsymbol{\sigma}\left(\hat{\mathbf{d}}_{\mathrm{sta}}\right) + \boldsymbol{\sigma}\left(\hat{\mathbf{d}}^{\mathrm{d}}_{t_s}\right) . \tag{2.63}$$

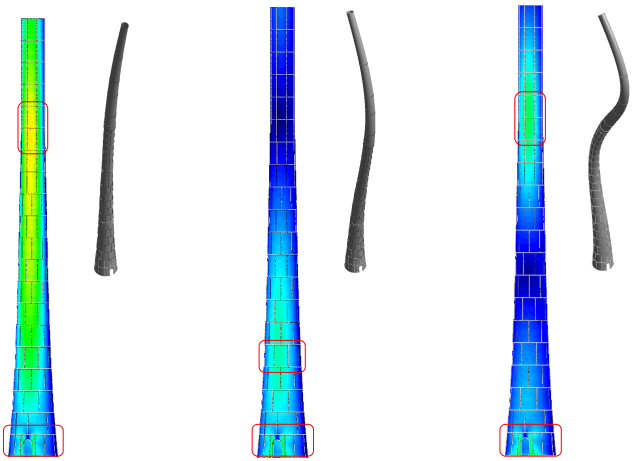
In addition, the assumption for the linear transient phase enables the decomposition of the stresses as a linear combination of the eigenmodes. Introducing Eqn. (2.32) in Eqn. (2.63) one arrives at the following equation for the transient stress states:

$$\hat{\mathbf{d}}^{\mathrm{d}}_{t_s} = \sum_{i=1}^{n} \alpha_{(i),t_s} \hat{\boldsymbol{\Phi}}_{(i)} , \tag{2.64a}$$

$$\boldsymbol{\sigma}\left(\hat{\mathbf{d}}_{t_s}\right) = \boldsymbol{\sigma}\left(\hat{\mathbf{d}}_{\mathrm{sta}}\right) + \sum_{i=1}^{n} \alpha_{(i),t_s} \boldsymbol{\sigma}\left(\hat{\boldsymbol{\Phi}}_{(i)}\right) . \tag{2.64b}$$

The last term in Eqn. (2.64b) allows the investigation of the modal stress states for the detection of the hot-spots. In addition, the prominence and

the accuracy of the modal analysis method proposed in Sec. 2.1.7 are previously revealed. Furthermore, it is clear that the corresponding eigenmodes are also differing from the eigenmodes related to the conventional modal analysis. This observation is also made in Sec. 2.4.2 and in Fig. 2.6. Thus, the modal analysis is carried out by making use of the updated modal analysis method. In order to investigate the modal stress states, the eigenmodes are applied as a non-zero Dirichlet condition on the FEM model and the von Mises stresses are studied. The respective stress states of the first, the third and the fifth bending modes are illustrated in Fig. 2.12. The second, the forth and the sixth are excluded in the plots for brevity since they are related to similar deformations but in a perpendicular direction. The study of the stress states related to the prominent eigenmodes reveal



(a) Door || 1 (b) Door || 2 (c) Door || 3

Figure 2.12: von Mises stresses of the concrete shaft bending modes and the corresponding tower eigenmodes. The identified tower sections for the fatigue analysis are indicated with the red boxes.

the highest stress exposure positions. Thus, the concrete segments that are related to the highest exposures are identified as fatigue hot-spots. Some of the fatigue hot-spots and their von-Mises equivalent stress distributions are plotted in Fig. 2.12.

2.5 Results and Discussion on Steady State FSI Analysis

In this section, the investigations on the considered coupling methods for FSI analysis are presented. The demonstration cases are selected in such a way that an increasing complexity is achieved. Foremost, an analytical FSI problem is presented where the numerical solution using the higher order solution schemes can be applied. Subsequently, a structural problem is considered where the loading condition resembles the fluid stresses acting on the structural wet surface. In addition to the comprehensive examples, a practical case is presented where the FSI of a 90° pipe is thoroughly investigated in a comparative manner using the developed and the existing methods. Last but not least, the FSI analysis of the CX-100 wind turbine blade is performed.

2.5.1 Couette Flow - Spring Attached Rigid Plate in FSI

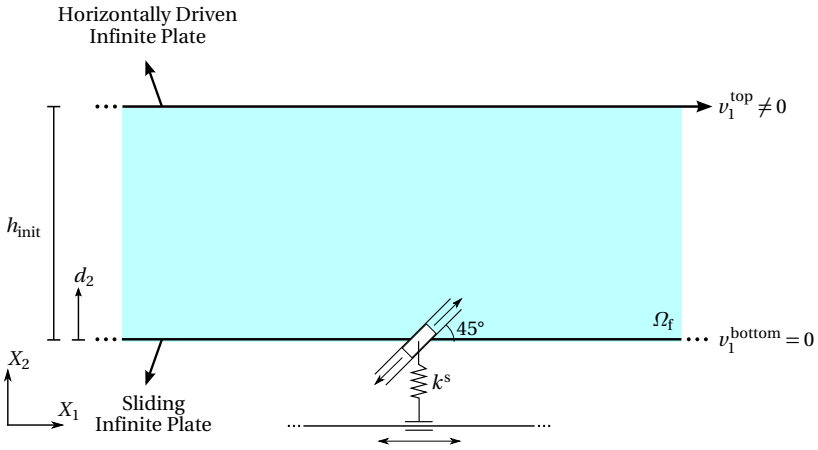


Figure 2.13: Couette flow — Spring Attached Rigid Plate FSI problem setting for the analytical derivation.

In order to demonstrate the mentioned methods in Sec. 2.3, an analytical FSI problem is generated. Fig. 2.13 depicts the problem setup. The problem consists of a horizontally sliding infinite plate with a prescribed velocity

on top and another infinite plate at the bottom that is attached to a spring which is constrained by a horizontally sliding support at its lower end and a 45° inclined support at its upper end. Between the two mentioned plates lies a Newtonian fluid at rest. The fluid flow is induced by the prescribed velocity of the upper plate and applies a shear stress on the lower plate.

Analytical Solution

Even though analytical solutions for the interaction of Taylor-Couette flow and shell structures exist [56], an analytical solution for the presented FSI case is not found in the literature. Thus, the analytical solution for the interaction of the Couette flow and the spring attached rigid plate is derived. In the literature it is also stated that the pressure gradient vanishes for Couette flow too, see [57]. The governing steady state Navier Stokes equations reduce to a diffusion problem in 1D:

$$\widetilde{\nu}\frac{\partial^2 v_1}{\partial X_2^2} = 0 \text{ , in } \Omega_\mathrm{f}, \tag{2.65a}$$

$$v_1\left(h_\mathrm{init} - d\right) = v_1^\mathrm{top} \text{ ,} \tag{2.65b}$$

$$v_1(0) = 0 \text{ .} \tag{2.65c}$$

The solution of the above ordinary differential equation (ODE) results in a linear velocity profile given by the following equation:

$$v_1\left(X_2\right) = \frac{v_1^\mathrm{top}\, X_2}{h_\mathrm{init} - d} \text{ .} \tag{2.66}$$

In the above equations the index of the displacement variable d_2 is dropped since the solution does not depend on the horizontal displacements. A further simplification is introduced by setting the kinematic viscosity $\widetilde{\nu} = 1$. The resultant tractions then can be expressed in terms of the velocity gradient in the vertical direction:

$$t^\mathrm{f} = \frac{v_1^\mathrm{top}}{h_\mathrm{init} - d} \text{ .} \tag{2.67}$$

It is assumed that the springs are implemented continuously along the lower plate. The structural tractions that result from the distributed spring

system is given by:

$$t^{\mathrm{s}} = -k^{\mathrm{s}}\, d\ . \tag{2.68}$$

The posed problem is in equilibrium only if the tractions acting on the interface are in equilibrium. The residual equation of the given system can be written as follows:

$$R = t^{\mathrm{s}} + t^{\mathrm{f}} \tag{2.69a}$$

$$= -k^{\mathrm{s}} d + \frac{v_1^{\mathrm{top}}}{h_{\mathrm{init}} - d} = 0\ . \tag{2.69b}$$

Another assumption that is introduced at this point is no-overlapping and no-penetration condition of the lower and upper plates. Assuming that the vertical displacement of the lower plate complies with this assumption, i.e. $d < h_{\mathrm{init}}$, the problem can be rearranged to form the following polynomial:

$$k^{\mathrm{s}} d^2 - k^{\mathrm{s}} h_{\mathrm{init}} d + v_1^{\mathrm{top}} = 0\ . \tag{2.70}$$

Thus, the roots of the problem are;

$$d_{1,2} = \frac{k^{\mathrm{s}} h_{\mathrm{init}} \pm \sqrt{(k^{\mathrm{s}})^2 h_{\mathrm{init}}^2 - 4k^{\mathrm{s}} v_1^{\mathrm{top}}}}{2k^{\mathrm{s}}}\ . \tag{2.71}$$

It is clear that the problem would only have a unique solution when the following holds $(k^{\mathrm{s}})^2 h_{\mathrm{init}}^2 - 4k^{\mathrm{s}} v_1^{\mathrm{top}} = 0$ with multiple roots. Furthermore, the following is a necessary condition for sensible solutions to exist at all: $(k^{\mathrm{s}})^2 h_{\mathrm{init}}^2 \geq 4k^{\mathrm{s}} v_1^{\mathrm{top}}$. Even in this simplified problem it is easy to observe that the FSI problems do not necessarily have unique solutions, since even though the solved subdisciplines are linear, the post processing variables that are coupled at the interface can depend on each other. For this particular case, the fluid problem is linear on its primary variable but the post processing variable traction is not linearly dependent on the interface displacement. In a general setting, the converged result would depend on the imposed initial guess for the variable d and on the employed numerical solution procedure. In this specific case, when two of the problem parameters are fixed the third parameter can be chosen as given below

such that a unique solution exists:

$$k^{\mathrm{s}} = \frac{4\, v_1^{\mathrm{top}}}{h_{\mathrm{init}}^2}\,, \tag{2.72a}$$

$$v_1^{\mathrm{top}} = \frac{k^{\mathrm{s}} h_{\mathrm{init}}^2}{4}\,, \tag{2.72b}$$

$$h_{\mathrm{init}} = 2\sqrt{\frac{v_1^{\mathrm{top}}}{k^{\mathrm{s}}}}\,. \tag{2.72c}$$

Numerical Solution Using Newton-Raphson Method

Given the residual equation in Eqn. (2.69a), one can derive the necessary terms for the NR solution procedure. The linearization of the residual around a given solution d_n reads:

$$R\left(d_n + \Delta d_n\right) = R_n + \left.\frac{\partial R_n}{\partial d}\right|_{d=d_n} \Delta d_n + \mathrm{HOT}\,, \tag{2.73}$$

where;

$$R_n = t_n^{\mathrm{s}} + t_n^{\mathrm{f}} = -k^{\mathrm{s}} d_n + \frac{v_1^{\mathrm{top}}}{h_{\mathrm{init}} - d_n}\,, \tag{2.74a}$$

$$\left.\frac{\partial R_n}{\partial d}\right|_{d=d_n} = -k^{\mathrm{s}} + \frac{v_1^{\mathrm{top}}}{\left(h_{\mathrm{init}} - d_n\right)^2}\,, \tag{2.74b}$$

and the computation of the state update is achieved through;

$$\Delta d_n = -\left(\left.\frac{\partial R_n}{\partial d}\right|_{d=d_n}\right)^{-1} R_n\,. \tag{2.75}$$

Numerical Solution Using Gauß-Seidel Method

As explained in Sec. 2.3.1 GS fixed point iterations can also be used for the solution of the given nonlinear problem. In this case, the solutions of the structural and the fluid problems should be performed in a sequential manner. As there are not any external loads other than fluid tractions acting on the spring system, the solution procedure should start with the

computation of the fluid loads. Thus, the following solution sequence is adopted:

$$t_{n+1}^{\mathrm{s}} = t_{n}^{\mathrm{f}} , \tag{2.76a}$$

$$d_{n+1} = \frac{t_{n+1}^{\mathrm{s}}}{k^{\mathrm{s}}} , \tag{2.76b}$$

$$t_{n+1}^{\mathrm{f}} = \frac{v_1^{\mathrm{top}}}{h_{\mathrm{init}} - d_{n+1}} . \tag{2.76c}$$

Furthermore, one can formulate the fixed point iterations in a residual way such that the solution procedure resembles the NR method. Following from Eqn. (2.74);

$$\widetilde{\left.\frac{\partial R_n}{\partial d}\right|_{d=d_n}} = -k^{\mathrm{s}} + \frac{v_1^{\mathrm{top}}}{\left(h_{\mathrm{init}} - d_n\right)^2} , \tag{2.77}$$

and the state update;

$$\Delta d_n = -\left(\widetilde{\left.\frac{\partial R_n}{\partial d}\right|_{d=d_n}} \right)^{-1} R_n . \tag{2.78}$$

Numerical Solution Using GS-IBQNLS

In order to demonstrate the performance of the least-squares family methods GS-IBQNLS is chosen. Moreover, due to the rigid lower plate assumption, the geometric derivatives are exactly zero and the GS-FL-IBQNLS method is not expected to perform better in this particular case. Following from Sec. 2.3.3 and Eqn. (2.74), the solution procedure can be expressed by the approximated fluid interface Jacobian replacing the exact one as the following:

$$\widetilde{\left.\frac{\partial R_n}{\partial d}\right|_{d=d_n}} = -k^{\mathrm{s}} + \widetilde{\frac{\partial t_n^{\mathrm{f}}}{\partial d}} , \tag{2.79}$$

and the state update;

$$\Delta d_n = -\left(\widetilde{\left.\frac{\partial R_n}{\partial d}\right|_{d=d_n}} \right)^{-1} R_n . \tag{2.80}$$

Performance Comparisons

The investigations on the accuracy and performance comparisons of the presented algorithms in this section are performed on two sets of problem parameters. Keeping the conditions defined in Eqn. (2.72) in mind, one set of chosen parameters result in a unique solution case and the other set of case parameters pose a nonunique solution. The chosen case parameters, the corresponding analytical solutions and the residual evolutions with respect to the interface displacement are presented in Table 2.5, Table 2.6 and Fig. 2.14 respectively.

	k^s	h_{init}	v_1^{top}
Unique Solution	1.0	1.0	0.25
Nonunique Solution	1.0	1.0	0.05

Table 2.5: The unique and nonunique solution problem parameters.

	d_1	d_2
Unique Solution	0.5	0.5
Nonunique Solution	0.0527864045	0.9472135955

Table 2.6: The unique and nonunique analytical solutions.

The evolutions of the relative interface residuals as well as the relative displacement errors for both cases are plotted in Fig. 2.15-2.16. For the case of unique solution, it should be noted that the quadratic polynomial exhibits one root with multiplicity of 2. Thus, the convergence rate of the conventional NR method remains linear, see also [58]. Another aspect that can be observed in this case is the remaining error for the GS method. This points out the necessity for a full or an approximated Jacobian in order to achieve firstly a convergence and secondly an accurate result at all in this specific case.

In case of the nonunique solution parameters, it can be noted that a quadratic convergence for the NR method can be achieved since the multiplicity of the roots are 1 and the initial guess is in the vicinity of a root. In

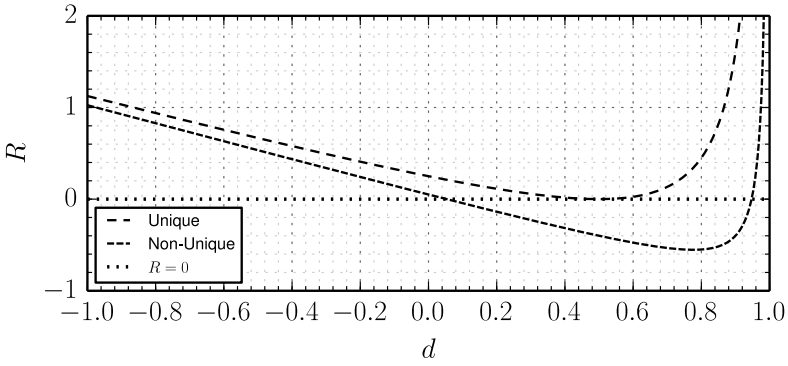
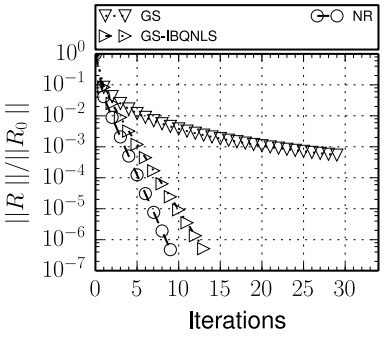
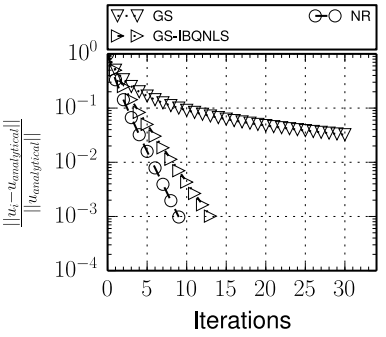


Figure 2.14: Residuals vs. displacements for the unique and nonunique solution problem parameters.

case the full Jacobian from a fluid analysis is not present, one can approximate it via GS-IBQNLS method and obtain a much better convergence rate than the fixed-point methods in general. GS-IBQNLS uses multiple layers of iteration history in order to compute an approximate Jacobian. On the other hand, since there is only one DOF for the solution of the system, use of many layers of history easily results in a singular matrix for $\mathbf{V}_n^T \mathbf{V}_n$, which is one of the drawbacks of the least-squares methods in general. In this case, the iteration history is limited to the last iteration. For larger practical cases the properties of the input modes can be analysed and the modes that cause a singularity in the system can be eliminated, e.g. using QR-decomposition [46].

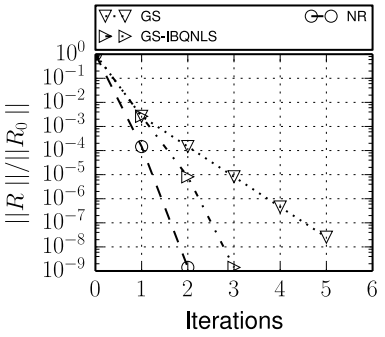


(a) Evolution of the residual.

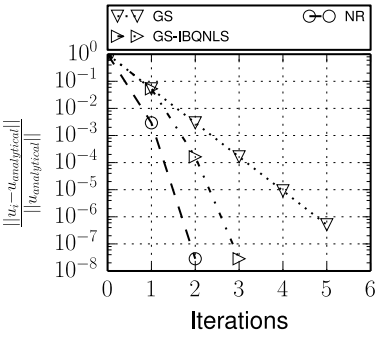


(b) Evolution of the relative displacement error with respect to the analytical solution.

Figure 2.15: Solution steps of the analytical FSI problem with the unique solution parameters.



(a) Evolution of the residual.



(b) Evolution of the relative displacement error with respect to the analytical solution.

Figure 2.16: Solution steps of the analytical FSI problem with the nonunique soluion parameters.

2.5.2 Ring with State Dependent Loads

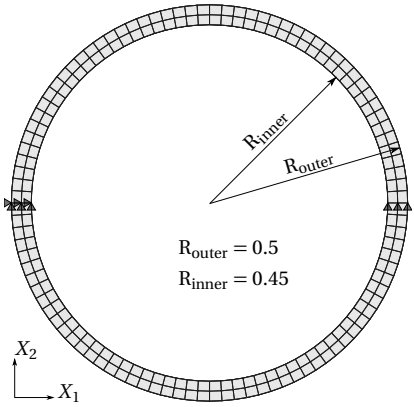


Figure 2.17: Ring case with state dependent pressure loads.

The model case depicted in Fig. 2.17 considers a state dependent loading where not only the local loading direction depends on the surface orientation but also its magnitude. Thus, the problem behaves nonlinear. The structure is modelled by linear hexahedral elements while the boundary conditions are applied by a simple support and a sliding support at the respective positions as depicted in Fig. 2.17. The material is assumed to be linear elastic isotropic, which has the parameters listed in Table 2.7.

E	10	$[MPa]$
ν	0.3	$[-]$

Table 2.7: The structural material parameters.

Finally, the discrete state dependent traction vectors at respective surface Gauß points are defined as:

$$\bar{\mathbf{t}} = \mathbf{n} \cdot \boldsymbol{\sigma}^* \text{ on } \Gamma_{\mathrm{n}} \,, \tag{2.81}$$

where the state dependent stress tensor $\boldsymbol{\sigma}^*$ defined as:

$$\boldsymbol{\sigma}^* = c \left(\mathbf{d} \cdot \tilde{\mathbf{n}} \right) \mathbf{I} \,. \tag{2.82}$$

c	$1.0 \quad [MPa]$
$\mathbf{d}$	$[1,0,0]^T$

Table 2.8: The load case parameters.

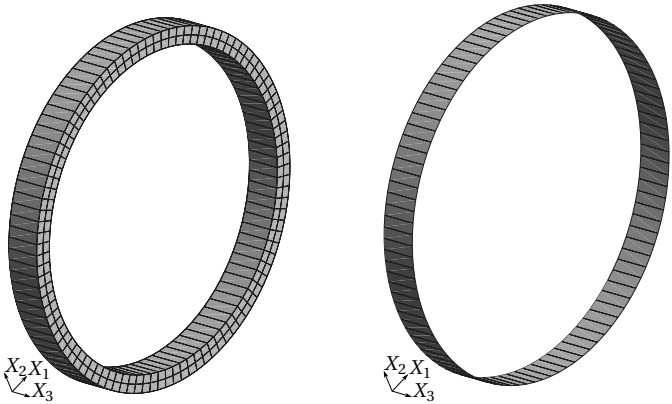
Here c, $\mathbf{d}$, and $\mathbf{I}$ denote a constant scaling factor, a constant predefined direction and the identity tensor respectively. The parameters regarding the loading condition are listed in Table 2.8. In Sec. 2.3.3-2.3.4 the fluid loading acting on the wet surface of the structure is considered as an unknown function which depends on the state. There it is assumed that this function is either not accessible or it is too costly to compute its derivatives with respect to the state variables. By making use of such a definition as in Eqn. (2.82), the stress tensor varies depending on the state, of which both the exact and the approximated LHS contributions can be computed. Thus, it allows a performance comparison of the solution procedures ranging from higher order methods such as NR to lower order methods such as GS. Furthermore, the effect of the neglected and approximated terms can also be investigated. Fig. 2.18(a) and Fig. 2.18(b) depict the discretization of the structural model as well as the Neumann boundary, namely the wet surface in an FSI setting.

Incorporating Eqn. (2.82) into Eqn. (2.40c), the contribution of the load stiffness to the tangent stiffness matrix at each NR iteration reads:

$$K^{\text{load}}_{(i,j)} = \left\langle \boldsymbol{\phi}_{(i)}, \frac{\partial \mathbf{n}}{\partial \hat{d}_{(j)}} \cdot \boldsymbol{\sigma}^* \right\rangle_{0,\Gamma_{\text{FSI}}} + \left\langle \boldsymbol{\phi}_{(i)}, \mathbf{n} \cdot \frac{\partial \boldsymbol{\sigma}^*}{\partial \hat{d}_{(j)}} \right\rangle_{0,\Gamma_{\text{FSI}}}, \tag{2.83}$$

and both terms can be computed exactly.

The convergence behaviours of the compared algorithms are illustrated in Fig. 2.19. As expected, the NR solution outperforms the other possible methods and exhibits quadratic convergence towards the solution. On the other side of the spectrum lies the GS method which shows linear convergence reducing the relative residual a level of magnitude at each solution iteration. This case can be considered as the worst possible convergence scenario if the solution converges at all since the driving part of the problem Jacobian is neglected. Another interesting result to point out is the effect of term exclusions from the exact LHS contributions from



(a) Perspective view. (b) Loading surface.

Figure 2.18: The perspective view of the ring case with state dependent loads.

the tangent stiffness matrix or in other words inclusion of terms to the GS method that can well be computed in an FSI scenario. In this particular case, the structural stiffness matrix as well as the geometric definition of the discrete model are accessible. Thus, the derivatives related to the geometric quantities, e.g. the derivatives of the surface normals, with respect to the state variables are straightforward to compute. The plot cited as GS-FL in Fig. 2.19 shows the improvement of the convergence behaviour compared to GS when the follower load stiffness is included in the tangent stiffness matrix. The inclusion of this term improves the convergence behaviour by an order of magnitude in the relative residual by the 3rd iteration and this improvement remains approximately constant in the further iterations. Furthermore, the comparison of the plots cited as GS-IBQNLS and GS-FL-IBQNLS is worth mentioning. Since GS-FL-IBQNLS relates the contribution to the current state and the stiffness contribution is partially already available from the first iteration on, the convergence of this method exhibits already a better behaviour from the first iteration. As the iterations advance, the improvement evolves to be even more visible. Even though both methods satisfy the convergence criterion at the same iteration, it can well be observed that GS-FL-IBQNLS results in a lower residual. Out

of the Jacobian approximation techniques, GS-FL-IBQNLS is as expected the best performing method among all but as a price to be paid, all the iteration history of $\boldsymbol{\sigma}^*$ is stored for the Jacobian approximation.

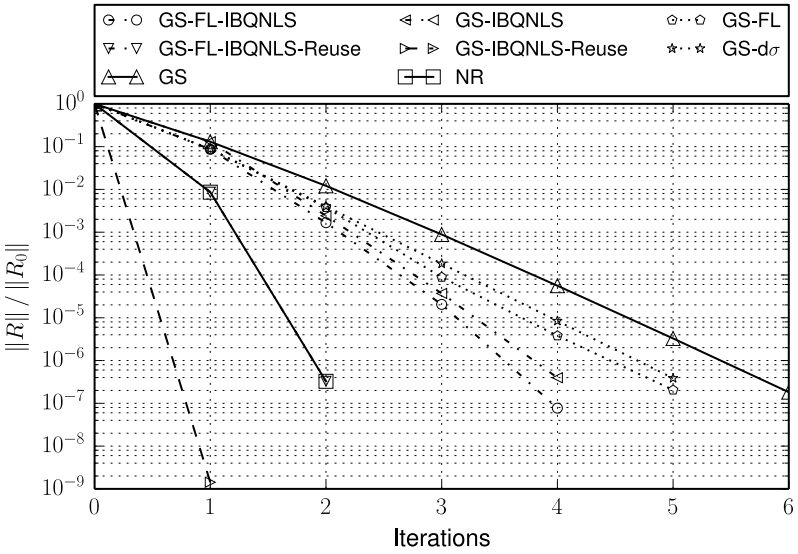
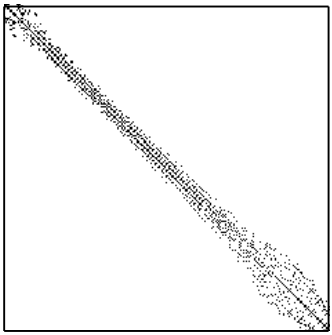


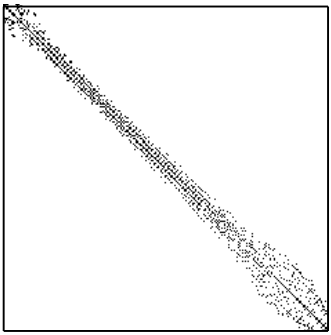
Figure 2.19: Relative residual convergence behaviours of the compared algorithms.

Another aspect to discuss is the solution procedure of the linear system of equations. All the methods compared in this section but the GS exhibit some form of non-symmetry in the resulting tangent stiffness matrix due to the nonsymmetric and non-conservative nature of the state dependent load definition. Thus, they are not appropriate in the case of a sole availability of a linear solver that is suitable for sparse symmetric matrices. If a direct solver is to be used, then the sparseness of the matrix is a favourable feature. In Fig. 2.20 the sparsity patterns of the compared methods are shown. At a first glance, the sparseness of the GS, GS-FL, GS-dσ and NR can clearly be observed and they exhibit the most favourable structure while the sparseness of the three latter methods only slightly differ from the symmetric structure of the GS method. disGS-IBQNLS and GS-FL-IBQNLS

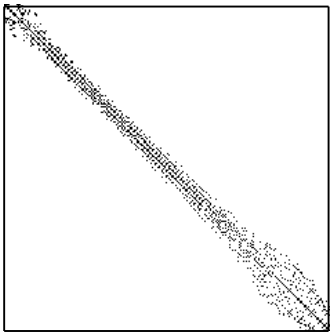
methods can be distinguished from the other methods by their nearly fully populated matrix structures. They cause a coupling between all the interface DOFs. While these methods can be considered expensive in the sense of solution time for the linear equation system due to their nonsymmetric and populated structures, they demonstrate the highest improvement in the convergence behaviour as presented in Fig. 2.19. In case of a partitioned FSI setting, the computation of the fluid solution with orders of magnitude higher number of DOFs can overshadow the computational time necessary for a structural simulation. Thus, the increased solution time of approximated Jacobian techniques should not be considered as an obstructing aspect.



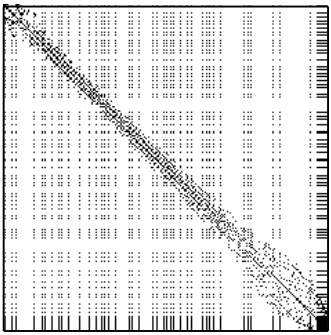
(a) GS.



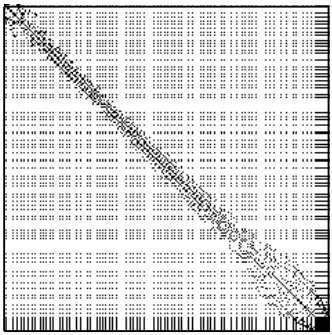
(b) GS-FL.



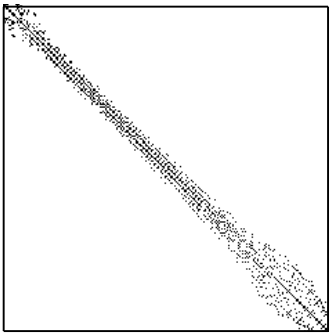
(c) GS-dσ.



(d) GS-IBQNLS.



(e) GS-FL-IBQNLS.



(f) NR.

Figure 2.20: Sparsity patterns for the compared methods with improving convergence behaviour.

2.5.3 90° Bent Pipe in FSI

In order to investigate the properties of the chosen methods, a more practical FSI case is designed. The case geometry and the parameters are chosen such that a strongly coupled FSI problem occurs where the changes in fluid forces due to the changing displacement states significantly affect the convergence behaviour.

The designed case consists of a highly viscous honey-like fluid flowing through a 90° bent pipe and the inlet conditions are chosen such that laminar flow assumptions can be introduced, see Fig. 2.22. The bend is chosen to be the interaction surface and it is designed as a flexible rubber-like material which can be considered as a shell. The straight pipes are assumed to be rigid. The case geometry and the dimensions are depicted in Fig. 2.21 and the material properties for the fluid and the structure are given in Table 2.9.

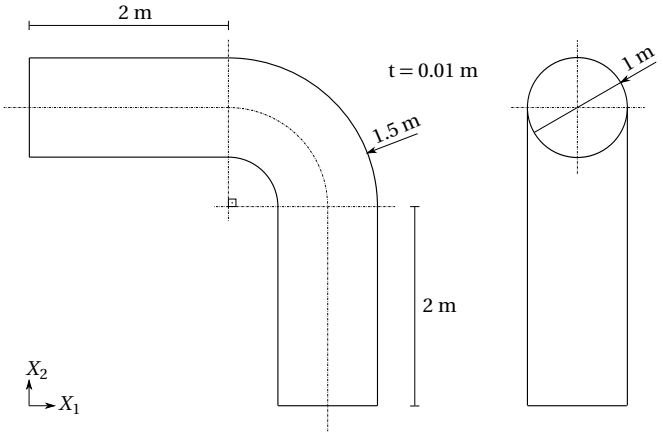


Figure 2.21: Dimensions of the 90° bent pipe FSI case.

Fluid	$\rho = 1400$	$[kg/m^3]$	$\mu = 10$	$[Pa \cdot s]$
Structure	$\nu = 0.0$	$[-]$	$E = 25$	$[MPa]$

Table 2.9: The fluid and the structural material parameters.

Boundary	Velocity	Pressure
Inlet	Developed laminar flow	$(\mathbf{n}\cdot\nabla)p=0$
Outlet	$(\mathbf{n}\cdot\nabla)\mathbf{v}=\mathbf{0}$	$p=0$
Wall	$\mathbf{v}=\mathbf{0}$	$(\mathbf{n}\cdot\nabla)p=0$

Table 2.10: The fluid boundary conditions.

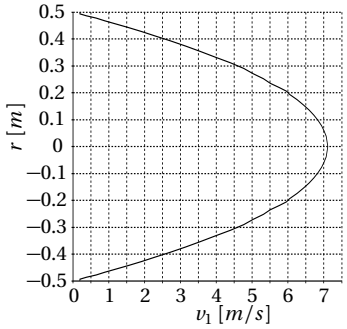
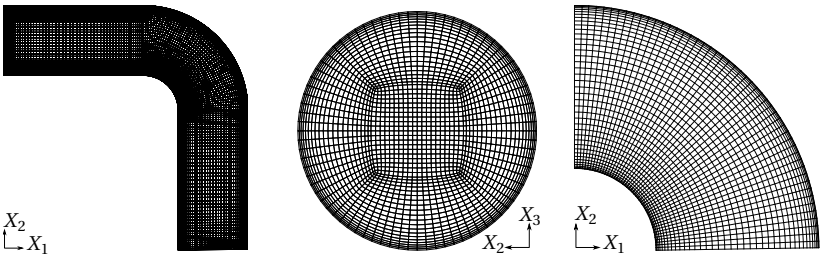


Figure 2.22: The prescribed velocity profile at the inlet.



(a) The axial discretization. (b) The inlet discretization. (c) The structural domain discretization.

Figure 2.23: The CFD as well as CSM discretizations.

The boundary conditions for the fluid case are listed in Table 2.10. The discretizations of the fluid and the structural domains are presented in Fig. 2.23. The structural solution is carried out by FEM. In order to avoid the possible numerical errors that can occur due to the mapping between

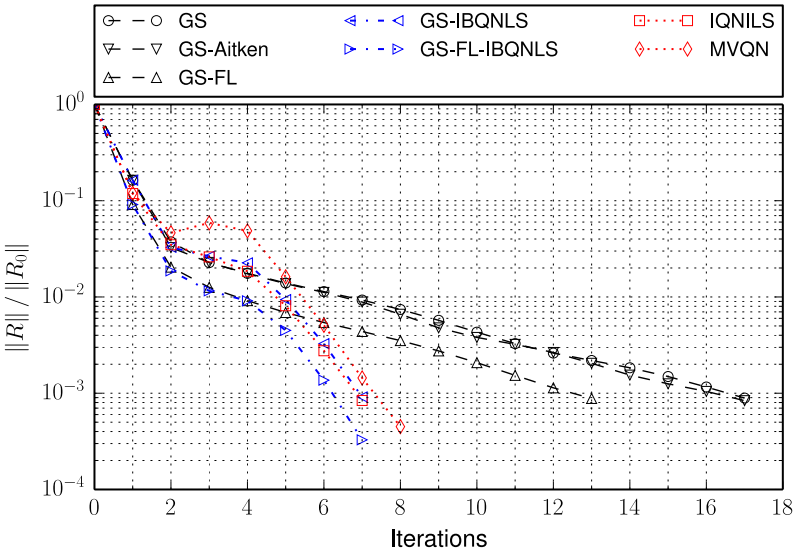


Figure 2.24: Relative residual convergence behaviour of the compared algorithms.

dissimilar grids, matching discretizations are employed at the coupling interface. Quadrilateral shell elements based on Reissner-Mindlin kinematics are used for the discretization of the geometrically linear system as seen in Fig. 2.23(c). A clamped boundary condition is applied by constraining the displacement and the rotational degrees of freedom at the free-edges of the structural model. The designed case is run with a convergence criterion on the relative force residual and the convergence is reached when the defined residual reduces below $1e-3$.

Fig. 2.24 presents the convergence behaviours of the methods to be investigated. As expected the GS method performs worst and exhibits poor convergence behaviour. An interesting outcome of this comparison is the performance of the GS method when the Aitken relaxation is adopted. It is commonly expected that GS performs better when Aitken relaxation is utilized. The poor performance can be related to the design of the Aitken relaxation method since it was originally developed for the acceleration

of a series for single variable [59]. On the contrary, in the existence of a multi-variable interface field, the method utilizes rather their norms and the norms of the changes in the interface values instead of the change in a single variable. Thus, the information on the interface reduces to a single variable which does not represent the behaviour of the system. Moreover, the method utilizes only the interface fields of the last two and the current iterations, restricting the use of the historical information. The improvement of the method denoted by GS-FL in comparison to the GS is again worth mentioning, in particular at the early stages of the analysis. The effect of the follower load related load stiffness inclusion in the tangent stiffness matrix already improves the convergence behaviour from the first iteration and in this particular case acts similar to the GS-FL-IBQNLS method. Since the GS-FL-IBQNLS method includes this contribution, it can well be noted that the inclusion of the geometry related load stiffness is highly important at the first iterations if not more than the contribution related to the variation of the flow variables as mentioned in Sec. 2.3.4. This can well be observed when the GS-IBQNLS and GS-FL-IBQNLS methods are compared. As presented in Eqn. (2.50a) and Eqn. (2.52a), the approximation of the load stiffness term in GS-FL-IBQNLS is more accurate than GS-IBQNLS, which incorporates exactly the additional term of GS-FL. Due to the higher accuracy of the load stiffness contribution approximation for the tangent stiffness matrix, the GS-FL-IBQNLS method exhibits better convergence behaviour than the GS-IBQNLS method. Even though the set residual criterion is reached at the same number of iterations for both methods, the performance gain in the first iterations remain almost unchanged. In addition, IQNILS and MVQN methods are also included in the comparisons. Since the convergence criterion is defined on the force residual, these algorithms are also employed by defining the interface residual as the output variable. It can be observed that IQNILS behaves very similar to GS-IBQNLS, since the approximated quantities are essentially the same. Finally, it can be concluded that the comparison of all the least-squares based methods exhibit similar convergence characteristics once a suitable Jacobian is built.

Another outcome of the previous comparison is the combination possibility of the existing methods. For instance, the GS-FL method exhibits similar convergence behaviour with GS-FL-IBQNLS at the first stages of the analysis. This indicates that until a suitable approximation for the derivative of

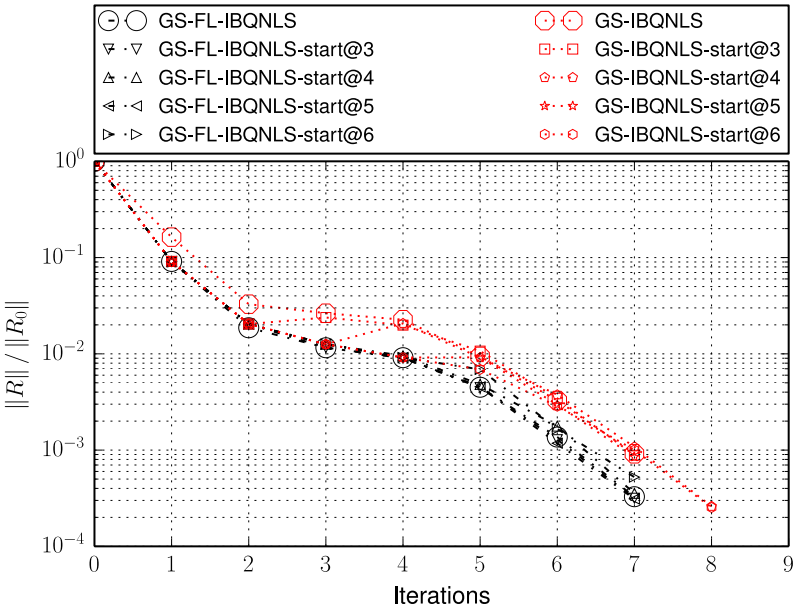


Figure 2.25: Relative residual convergence behaviour of the compared algorithms.

the fluid stress tensor is built, the accuracy of the load stiffness is dominated by the derivative of the geometric quantities. Moreover, as discussed in Sec. 2.3.2, the sparsity pattern of the GS-FL is advantageous compared to the GS-IBQNLS family of methods which significantly reduces the solution time of the linear equation system and the memory usage for the system matrix. In order to exploit this, the analysis is started with the GS-FL method and the GS-IBQNLS family of methods are switched on after a certain number of iterations. The iteration history is stored in the same fashion as mentioned in Sec. 2.3.4. The performances of the methods are presented in Fig. 2.25. It can be said that regardless of when the methods are switched on, they tend to behave similar to as if they were initiated from the very first iteration. This implies that the method, which exhibits the best performance, should be preferred for such a procedure.

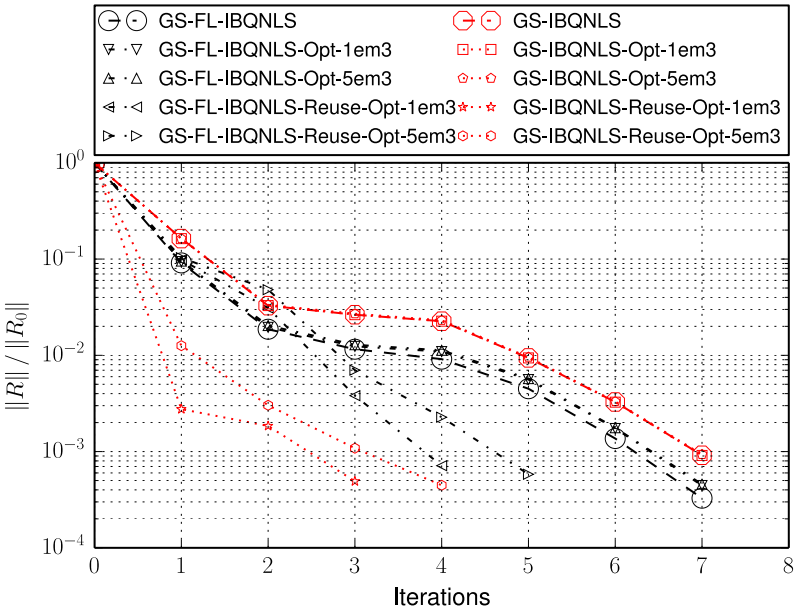


Figure 2.26: Relative residual convergence behaviour of the compared algorithms.

The goal of the FSI analysis in the scope of this work is to find the equilibrium condition for the consequent optimization task. The advantage of the least-squares family of methods are already revealed in the previous showcases and comparisons. This gives the motivation to reuse the information from previous FSI analyses in the consequent optimization iterations and build a reduced order model that can improve the convergence behaviour of the FSI analyses in each optimization iteration. In order to facilitate the iteration histories in the coming optimization iterations, a suitable storage scheme should be used. The relativity information would be lost, when only difference of the states are stored. Instead, the values at the given states should be stored to keep the relativity information available. For instance, instead of the nodal displacements, the nodal coordinates at each FSI and optimization iteration are stored. Correspondingly, the

forces for the GS-IBQNLS and the stress tensor components for the GS-FL-IBQNLS are also stored at the respective states. As a demonstration case, an optimization problem is considered where the radius of the flexible bend is chosen as the design variable. Two shape update cases are chosen as -1 mm and -5 mm for this purpose. The performed FSI analyses are presented in Fig. 2.26. It can be observed that the convergence characteristics of the problem are similar before and after the shape updates are performed, if the iteration history is not utilized. On the other hand, when the historical information is reused in the new FSI analysis of the updated design, a substantial improvement in the convergence behaviour can be observed. By introducing a shape update and modifying the problem, the possible displacement states are also modified due to the altered stiffness properties and initial guesses. Thus, the performance of the algorithms deteriorates as the shape updates become larger in magnitude. Even though the possible states to occur in the new problems are different, the improvement in the convergence behaviour can be observed, since the previously occurred states are still reasonable and can be related to the updated problem.

2.5.4 CX-100 Wind Turbine Blade in FSI

In this section, the steady state FSI analysis of the CX-100 horizontal axis wind turbine (HAWT) blade is presented. The goal of this case setup is essentially to accomplish the shape optimization of the blade structure at the equilibrium state between the interacting fluid and structure domains. Thus, an initial steady state FSI analysis is performed such that the consecutive optimization routines can be applied. In the following sections, the modeling, the discretization of the fluid and the structural domains as well as the accomplished FSI analysis are discussed.

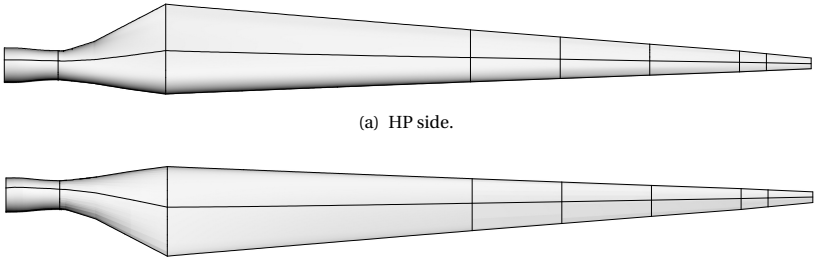
Modeling

As a start, the CAD model of the blade wet surface is available in the supplementary files of the meshing software Pointwise®'s webinar[1]. The defined interaction surface of the original CAD model is illustrated in Fig. 2.27. The generated hybrid mesh by following the webinar consists of 7.5 million hexahedral and tetrahedral cells. In addition to the discretization of the fluid domain, the defined rotating zone and the applied boundary conditions are depicted in Fig. 2.28, whereas Fig. 2.29 depicts the discretization of the fluid interaction surface, see also Table 2.11 for the applied boundary conditions at the respective boundaries.

Boundary	**Velocity**	**Pressure**
Blade	No slip	$(\mathbf{n}\cdot\nabla)p=0$
Inlet	$v_2 = 10$ m/s	$(\mathbf{n}\cdot\nabla)p=0$
Outlet	$(\mathbf{n}\cdot\nabla)\mathbf{v}=\mathbf{0}$	$p=0$
Farfield	$(\mathbf{n}\cdot\nabla)\mathbf{v}=\mathbf{0}$	$(\mathbf{n}\cdot\nabla)p=0$
Sides	Cyclic	Cyclic

Table 2.11: The boundary conditions of the fluid domain.

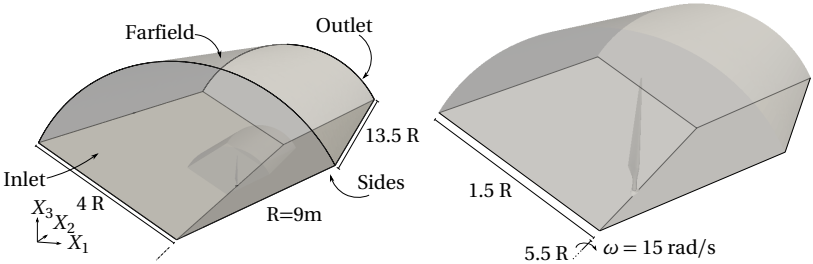
[1] Hybrid Meshing for a Horizontal Axis Wind Turbine `http://www.pointwise.com/webcasts/2011-12/Hybrid-Meshing-Horizontal-Axis-Wind-Turbine.html`



(a) HP side.

(b) LP side.

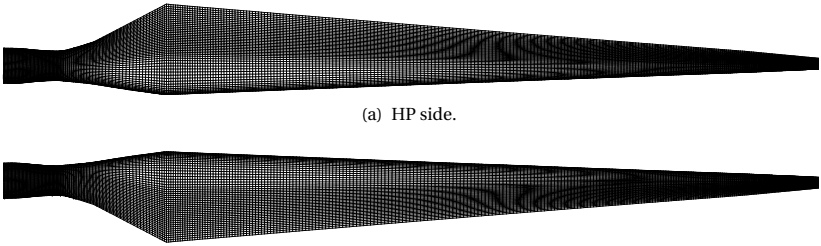
Figure 2.27: The original CAD model of the CX-100 blade wet surface.



(a) The fluid domain.

(b) The CX-100 blade and the rotating zone.

Figure 2.28: The fluid domain dimensions and the placement of the CX-100 blade.

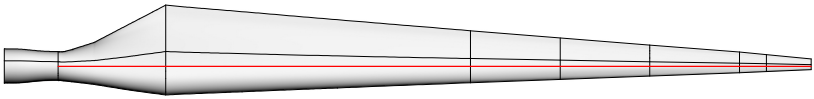


(a) HP side.

(b) LP side.

Figure 2.29: The FVM boundary discretization of the CX-100 fluid FSI interface.

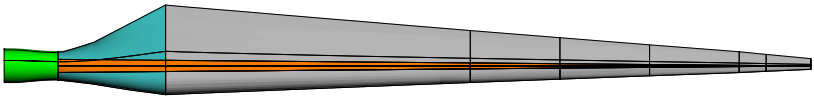
However, since the provided CAD model is meant to represent the fluid domain boundary, this model consists of the blade skin and the hub only. In addition to the blade skin, the structural stiffness of the blade is increased by a spar that runs along its length. Thus, a spar geometry is added to the existing CAD model which forms also the master model for the optimization tasks in Sec. 3.4.3, see Fig. 2.30 for the implemented spar position. It should be noted that even though there exists a number of literature on the design and the production of the CX-100 blade, the exact dimensions are either not available or the retrieved CAD model does not match the provided dimensions, see also [60], [61], [62]. Thus, the constructed model can be regarded as an approximate model which serves the demonstration purposes within this work. In addition to the implemented spar, the material zones for the definition of the composite material layers are necessary. In order to achieve this, the CAD model is split into a number of zones -namely patches. The structural discretization is achieved by approximately 11200 bilinear quadrilateral and linear triangular finite elements. Due to the thin structure of the blade and the utilized bilinear and linear discretizations of the structural domain, the shell elements based on the Reissner-Mindlin kinematics are utilized for the FEM modeling. The material behaviour is modeled by making use of the orthotropic material law which also incorporates multiple layers. The used material properties and the laminate stacking of the respective zones are listed in Table 2.12 and Table 2.13 respectively. The defined material zones and the corresponding structural FEM discretization are depicted in Fig. 2.31 and Fig. 2.32 respectively. Finally, the Dirichlet constraints are applied at the DOFs that lie at the free edge of the blade root.



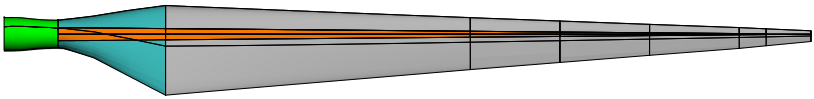
(a) HP side.

(b) LP side.

Figure 2.30: The master CAD model of the CX-100 blade with the implemented spar (red).

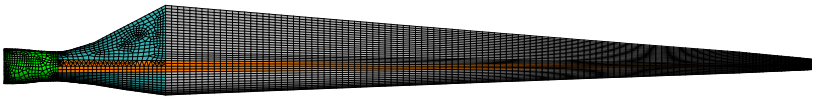


(a) HP side.

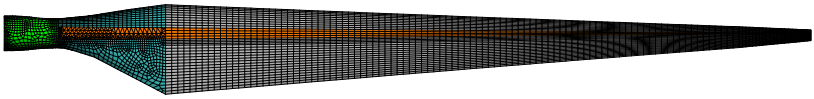


(b) LP side.

Figure 2.31: The modified CAD model of the CX-100 blade for the structural modeling. The root (green), the spar cap (orange), the HP/LP (grey) and the root-blade transition (blue) are colored for distinction.



(a) HP side.



(b) LP side.

Figure 2.32: The FEM discretization of the CX-100 structural FSI interface.

#	**Materials**	$E_1\,[GPa]$	$E_2\,[GPa]$	$G_{12}\,[GPa]$	ν
1	Gel Coat	3.44	3.44	1.38	0.3
2	Random Mat	7.58	7.58	4	0.3
3	End-grain balsa	0.12	0.12	0.02	0.3
4	DBM 1708 (± 45° fiberglass)	9.58	9.58	6.89	0.39
5	DBM 1208 (± 45° fiberglass)	9.58	9.58	6.89	0.39
6	C520 (0 ° fiberglass)	37.3	7.6	6.89	0.31
7	Carbon-fiberglass triaxial fabric	84.1	8.76	4.38	0.21

Table 2.12: The used materials and their properties for the material stacking of laminates [60].

Zone	**Material Stacking**	**Thickness** $[mm]$	**Zone**	**Material Stacking**	**Thickness** $[mm]$
HP skin LP skin Root-Blade Tip	3	6.35	Spar cap	1	0.13
	1	0.13		2	0.38
	2	0.38		4	0.89
	4	0.89		5	0.56
	5	0.56		7	0.63
Zone	**Material Stacking**	**Thickness** $[mm]$	**Zone**	**Material Stacking**	**Thickness** $[mm]$
Root	6	1.32	Spar	4	0.89
	6	0.66			
	1	0.13		7	50
	2	0.38			
	4	0.89		4	0.89
	5	0.56			

Table 2.13: The laminate material stackings of the defined material zones. Refer to table Table 2.12 for the material numberings. The stackings are ordered from bottom to top under the guidance of [61], [62].

Simulation

The FE simulation of the structural domain is achieved making use of the in-house FE software Carat++, whereas the fluid governing equations are solved using the OpenFOAM® framework. An MRF zone is defined in a subdomain of the fluid domain as illustrated in Fig. 2.28(b). This allows the simulation of the rotating wind turbine blade without actually rotating the domain itself through the defined constant angular velocity in the subdomain and incorporating the respective body forces in the fluid governing equations. Furhermore, the Spalart-Allmaras turbulence model is utilized. The load and the motion transfer between the dissimilar discretizations are achieved by making use of the Mortar Mapping implementation in the EMPIRE framework. The EMPIRE framework also facilitates the interface variable transfer between the disciplinary solvers. The extensions mentioned in Sec. 2.3 to the GS procedure are not possible to apply in this case since the available matrix classes for the geometrically linear static analysis within the software Carat++ are strictly symmetric. Thus, the nonsymmetric nature of the load tangents are not suitable for application in this demonstration case. Thus, as a coupling method the GS procedure with the Aitken relaxation is employed where the initial relaxation factor is set as; $\alpha_{\text{init}} = 0.5$. The relaxation in this case essentially helps maintaining the discretization quality in the fluid domain and the stability of the fluid simulations rather than improving the convergence behaviour of the coupling iterations. The FSI iterations are carried out for 4 iterations until a relative residual on the interface displacements are reduced below 2.5e−3. The relative residual at the iteration n is computed as follows:

$$R_n^{\text{FSI}} = \frac{\left\| \hat{\mathbf{d}}_{\Gamma_{\text{FSI}},n} - \hat{\mathbf{d}}_{\Gamma_{\text{FSI}},n-1} \right\|}{\left\| \hat{\mathbf{d}}_{\Gamma_{\text{FSI}},0} \right\|} \,. \tag{2.84}$$

The displacement contour plots and the comparison of the initial configuration against the deformed state are illustrated in Fig. 2.33 and Fig. 2.34 respectively.

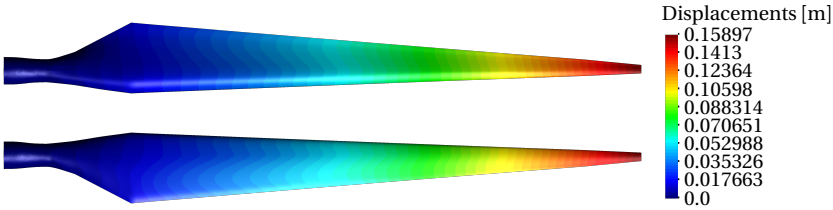


Figure 2.33: The displacement contours on HP (top) and LP (bottom) sides.

Figure 2.34: The initial (grey) and the deformed (red) states.

Chapter 3

High Fidelity Optimization Methods

This chapter consists of the required components for establishing a CAD-integrated optimization workflow for shape optimization tasks of structures in FSI environment. The disciplinary analysis methods are discussed in Chp. 2. Firstly, the employed disciplinary sensitivity analysis methods are briefly presented with the reasoning on specific choices. Secondly, the sensitivity filtering methods are discussed which is one of the main contributions of this work and this chapter. A sensitivity filtering method based on the Mortar Mapping method is introduced and its relation to the Vertex Morphing method is revealed in Sec. 3.2.2. The proposed sensitivity filtering method is then used for the integration of the CAD geometries into the surface driven shape optimization workflows, which makes the method favourable without impairing the quality of the optimization results, see Sec. 3.2.3-3.2.4 for details. Finally, the results with the combination of the existing, the implemented and the developed methods are presented with the provided discussions on particular aspects.

3.1 Sensitivity Analysis

In this section, the utilized sensitivity analysis methods for the optimization applications in the extent of this work are briefly presented. The optimization tasks in this thesis consist of a high number of design variables whereas the number of objective functions are relatively low. This makes the adjoint methods fit for the sensitivity analysis phase of the optimization procedures. The adjoint methods can be roughly categorized as discrete and continuous adjoint sensitivity analyses which can be applied to various disciplines, [63], [64]. Each category possesses advantages and disadvantages regarding their solution methods and discretization aspects. In this work, the semianalytic discrete sensitivity analysis for the structural objective functions and the continuous sensitivity analysis for the fluid objective functions are favoured, see also [5], [7], [65], [66] for further reading. The reasoning and the governing equations are presented in each respective section. In addition, it is worth mentioning that the coupled sensitivity analysis for FSI applications are excluded since this work concentrates on the setup of the optimization workflow and the CAD integration rather than the development of coupled sensitivity analysis methods. For further reading regarding the coupled sensitivity analysis in FSI, one can refer to [67].

3.1.1 Semianalytic Discrete Adjoint Sensitivity Analysis for Structures

In this work, the structural governing equations, which are necessary for the optimization tasks, are solved using the FEM by utilizing the in-house structural analysis tool Carat++. Moreover, due to the linear Green-Lagrange strains assumption, the structural discrete equation system is linear. Since the governing equations are solved using FEM, the use of the discrete adjoint sensitivity analysis is favourable due to the availability of the stiffness matrix of the FEM model already at the analysis phase. In addition, typically the problems considered within this work form symmetric, linear equation systems which results in the equivalence of the primal and the adjoint equation systems. The existing implementations in Carat++ offers the possibility for the computation of the sensitivity fields for certain objective functions using the discrete adjoint method, see also [7].

The discrete adjoint sensitivity analysis acts on the discrete equation systems rather than the continuous forms of the governing equations. The governing equations can be introduced as a constraint to the optimization problem and the objective function J can be augmented with this constraint using the Lagrange multipliers -namely the adjoint variables. The Lagrangian of the optimization problem reads:

$$\mathcal{L} = J + \hat{\boldsymbol{\lambda}}_{\mathrm{s}}^{T} \hat{\mathbf{R}} \,, \tag{3.1}$$

where the $\hat{\boldsymbol{\lambda}}_{\mathrm{s}}$ and the $\hat{\mathbf{R}}$ denote the Lagrange multipliers and the discrete residuals of the structural governing equations as in Eqn. (2.15), which depend on both the structural discrete state variables $\hat{\mathbf{d}}$ and the design variables $\hat{\mathbf{s}}$. The sensitivity analysis requires the derivative of the Lagrangian with respect to the design variables:

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}\hat{\mathbf{s}}} = \frac{\partial J}{\partial \hat{\mathbf{s}}} + \frac{\partial J}{\partial \hat{\mathbf{d}}} \frac{\mathrm{d}\hat{\mathbf{d}}}{\mathrm{d}\hat{\mathbf{s}}} + \hat{\boldsymbol{\lambda}}_{\mathrm{s}}^{T} \left[\frac{\partial \hat{\mathbf{R}}}{\partial \hat{\mathbf{s}}} + \frac{\partial \hat{\mathbf{R}}}{\partial \hat{\mathbf{d}}} \frac{\mathrm{d}\hat{\mathbf{d}}}{\mathrm{d}\hat{\mathbf{s}}} \right] = 0 \,. \tag{3.2}$$

The derivatives of the state variables with respect to design variables are costly to compute. Thus, the above equation Eqn. (3.2) can be rearranged which gives the adjoint methodology:

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}\hat{\mathbf{s}}} = \frac{\partial J}{\partial \hat{\mathbf{s}}} + \hat{\boldsymbol{\lambda}}_{\mathrm{s}}^{T} \frac{\partial \hat{\mathbf{R}}}{\partial \hat{\mathbf{s}}} + \underbrace{\left[\frac{\partial J}{\partial \hat{\mathbf{d}}} + \hat{\boldsymbol{\lambda}}_{\mathrm{s}}^{T} \frac{\partial \hat{\mathbf{R}}}{\partial \hat{\mathbf{d}}} \right]}_{\substack{\text{Adjoint Equation} \\ \text{System}}} \frac{\mathrm{d}\hat{\mathbf{d}}}{\mathrm{d}\hat{\mathbf{s}}} = 0 \,. \tag{3.3}$$

Enforcing the Lagrange multipliers to satisfy the adjoint equation system, which is indicated in Eqn. (3.3), leaves the computation of the sensitivities with simple partial derivatives. The adjoint equation system and the remaining sensitivity equation are as follows:

$$\frac{\partial \hat{\mathbf{R}}}{\partial \hat{\mathbf{d}}}^{T} \hat{\boldsymbol{\lambda}}_{\mathrm{s}} = -\frac{\partial J}{\partial \hat{\mathbf{d}}}^{T} \,, \tag{3.4a}$$

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}\hat{\mathbf{s}}} = \frac{\partial J}{\partial \hat{\mathbf{s}}} + \hat{\boldsymbol{\lambda}}_{\mathrm{s}}^{T} \frac{\partial \hat{\mathbf{R}}}{\partial \hat{\mathbf{s}}} \,. \tag{3.4b}$$

In addition, the residual term in Eqn. (3.4b) as well as its partial derivative for the geometrically linear structural problems can be written explicitly

as:

$$\hat{\mathbf{R}} = \mathbf{K}\hat{\mathbf{d}} - \hat{\mathbf{F}} \,, \tag{3.5}$$

$$\frac{\partial \hat{\mathbf{R}}}{\partial \hat{\mathbf{s}}} = \frac{\partial \mathbf{K}}{\partial \hat{\mathbf{s}}}\hat{\mathbf{d}} - \frac{\partial \hat{\mathbf{F}}}{\partial \hat{\mathbf{s}}} \,. \tag{3.6}$$

In some cases, the last term related to the partial derivatives of the forces in Eqn. (3.6) could be neglected or it effectively vanishes. It is worth mentioning that the methods described in Sec. 2.3.3-2.3.4 can be used to approximate this term, too. By plugging Eqn. (3.5) in Eqn. (3.4a), one obtains the following for the adjoint equation system:

$$\mathbf{K}^T \hat{\boldsymbol{\lambda}}_\mathrm{s} = -\frac{\partial J}{\partial \hat{\mathbf{d}}}^T \,. \tag{3.7}$$

For the case of linear structural problems, the following holds for the stiffness matrix $\mathbf{K}^T = \mathbf{K}$. Thus, the use of the FEM in combination with discrete adjoint sensitivity analysis is advantageous for such problems due to the availability of the exact stiffness matrix already at the primal solution phase.

The remaining term to be computed in Eqn. (3.4b) is the partial derivative of the stiffness matrix. This can well be achieved using a finite differencing scheme. The possible drawbacks and corrections for the errors in the finite differencing method are addressed in [68]. The existing implementation in Carat++ covers this gap. It should be noted that throughout this work a forward differencing step is used for the computation of this term in order to reduce the computational effort necessary for the sensitivity analysis.

3.1.2 Continuous Adjoint Sensitivity Analysis for Fluids

The fluid governing equations are solved using the FVM. Unlike the FEM method, commonly the governing equations are solved using iterative solution procedures which employ partitioned solution schemes for the fluid state variables $\mathbf{v}$ and p, namely the velocities and the pressures, when the FVM is adopted. This results in the unavailability of the system Jacobian for the governing equations. In this case, the continuous adjoint sensitivity analysis is favourable. In the scope of this work, the adjoint equation systems are solved using OpenFOAM® and the extensions of HELYX®.

The continuous adjoint sensitivity analysis methodology stems from enforcing the governing equations as a constraint to the optimization problem in a continuous sense. Thus, unlike the discrete adjoint methodology, the constraint is enforced in an integral sense. The Lagrangian of the problem reads:

$$\mathcal{L} = J + \int_{\Omega} \boldsymbol{\lambda}_{\mathrm{f}} \cdot \boldsymbol{\mathcal{R}}^{\mathrm{f}} \, \mathrm{d}\Omega \, . \tag{3.8}$$

In Eqn. (3.8), $\boldsymbol{\lambda}_{\mathrm{f}}$ denotes the vector of Lagrange multipliers related both to the adjoint velocities and the adjoint pressures while $\boldsymbol{\mathcal{R}}^{\mathrm{f}}$ collects the momentum and continuity equations in residual form in a vector. In a general case where the flow is in turbulent regime, the Lagrangian should be extended with the residual form of employed turbulence model equations too. In the scope of this work, the variation of the turbulent viscosity with respect to the design variables is neglected, which results in the common frozen turbulence assumption [69].

The continuous adjoint methodology follows from applying the stationary condition for the Lagrangian and setting its variation with respect to design variables to zero:

$$\begin{aligned} \frac{\mathrm{d}\mathcal{L}}{\mathrm{d}\mathbf{s}} = & \frac{\partial J}{\partial \mathbf{s}} + \frac{\partial J}{\partial \mathbf{v}} \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}\mathbf{s}} + \frac{\partial J}{\partial p} \frac{\mathrm{d}p}{\mathrm{d}\mathbf{s}} \\ & + \int_{\Omega} \boldsymbol{\lambda}_{\mathrm{f}} \cdot \frac{\partial \boldsymbol{\mathcal{R}}^{\mathrm{f}}}{\partial \mathbf{v}} \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}\mathbf{s}} \, \mathrm{d}\Omega + \int_{\Omega} \boldsymbol{\lambda}_{\mathrm{f}} \cdot \frac{\partial \boldsymbol{\mathcal{R}}^{\mathrm{f}}}{\partial p} \frac{\mathrm{d}p}{\mathrm{d}\mathbf{s}} \, \mathrm{d}\Omega \\ & + \int_{\Omega} \boldsymbol{\lambda}_{\mathrm{f}} \cdot \boldsymbol{\mathcal{R}}^{\mathrm{f}} \frac{\mathrm{d}(\mathrm{d}\Omega)}{\mathrm{d}\mathbf{s}} \overset{!}{=} 0 \, . \end{aligned} \tag{3.9}$$

The last term in Eqn. (3.9) is essentially related to the variation of the volume of interest with respect to to the design variables. In an ideal case where the primal equation systems are solved to satisfy $\boldsymbol{\mathcal{R}}^{\mathrm{f}} = \mathbf{0}$, this term effectively vanishes. Nevertheless, when a numerical solution procedure is adopted, this is not always the case. The problems related to the computation of this term are indicated in [70]. The existing implementation that is utilized in this work neglects this term since it is the common practice, [71], [72].

The rest of the derivation follows from transferring the partial derivatives of the governing equations onto the adjoint variables by making use of

integration by parts and setting the terms that multiply the total derivatives of the state variables with respect to the design variables to zero, namely the $\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}\mathbf{s}}$ and $\frac{\mathrm{d}p}{\mathrm{d}\mathbf{s}}$. Enforcing this condition results in the adjoint equation systems for the steady state incompressible Navier-Stokes equations with the frozen turbulence assumption. In addition, the boundary conditions of the primal problem are applied on the resulting boundary integrals, giving the adjoint boundary conditions for a specific set of primal boundary conditions. The derivation of the adjoint systems are omitted here and only the resulting equation systems are presented, see [73], [35] for the derivations. The adjoint equation systems read:

$$-\boldsymbol{\nabla}\boldsymbol{\lambda}_{\mathbf{v}}\cdot\mathbf{v}-\left(\mathbf{v}-\mathbf{v}_{\Omega}\right)\cdot\boldsymbol{\nabla}\boldsymbol{\lambda}_{\mathbf{v}}-\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}^{a}-\boldsymbol{\omega}\times\boldsymbol{\lambda}_{\mathbf{v}}+\frac{\partial J}{\partial\mathbf{v}}=0\,, \tag{3.10a}$$

$$\boldsymbol{\nabla}\cdot\boldsymbol{\lambda}_{\mathbf{v}}=\frac{\partial J}{\partial p}\,, \tag{3.10b}$$

where the so-called adjoint stress tensor defined as:

$$\boldsymbol{\sigma}^{a}=-\lambda_{p}\,\mathbf{I}+\widetilde{\nu}\left(\boldsymbol{\nabla}\boldsymbol{\lambda}_{\mathbf{v}}+\boldsymbol{\nabla}\boldsymbol{\lambda}_{\mathbf{v}}^{T}\right) \tag{3.11}$$

The consequences of the continuous adjoint methodology is twofold:

1. Regardless of the discretization scheme used for the primal problem, the adjoint equation system can be discretized using various discretization schemes, which either sustain the accuracy or increase stability of the solution process. This is in particular beneficial when the adjoint system is rather unstable as reported in [66].

2. The choice of the discretization scheme and/or solution procedure for the adjoint system affects the retrieved sensitivities. In case a suitable method is not chosen, the adjoint equation system does not necessarily correspond to the primal problem, which might result in slightly inaccurate sensitivity fields or not identical sensitivity fields compared to the discrete approach, see also [5] and [64].

3.2 Sensitivity Filtering

This section concerns with the utilization of the Mortar Mapping method as a sensitivity filtering operator and its properties as well as its relation to the Vertex Morphing method. In addition to that, it is going to be shown that both methods have their advantages and they can be combined to bring the benefits of both. The developed method facilitates the CAD integration into the optimization workflows whose instruments are mentioned later on in Sec. 3.3. In the scope of this thesis, the Vertex Morphing implementation following [8], [74] are used whereas the Mortar Mapping implementations in EMPIRE are utilized [38], [39], [40].

3.2.1 Vertex Morphing Method

For clarity and a definition of a unified notation with the Mortar Mapping, the Vertex Morphing method is briefly presented here following the definitions in [8]. The shape optimization problem can be stated as follows;

$$\begin{aligned} \min_{\mathbf{s}} \quad & J(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{u}) \\ s.t. \quad & \mathbf{R}_i(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{u}) = 0; \quad i = 1, \dots, k \\ & g_j(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{u}) \leq 0; \quad j = 1, \dots, m \end{aligned} \tag{3.12}$$

where J, R_i, g_j are the objective function the, state equations in residual form and the constraint functions respectively. The variables $\boldsymbol{\Theta}$, $\mathbf{z}$, $\mathbf{s}$, $\mathbf{u}$ denote the surface coordinates, the geometry, the design control field and the state variables. Additionally, the material coordinates $\boldsymbol{\theta}$, that are attached to the surface, are introduced. The state variables are a function of surface coordinates, the geometry and the design control fields; $\mathbf{u}(\boldsymbol{\Theta}, \mathbf{z}(\boldsymbol{\Theta}, \mathbf{s}))$. The surface coordinate $\boldsymbol{\Theta}(\boldsymbol{\theta})$ is defined to be a function of the material coordinates. The variables $\mathbf{z}(\boldsymbol{\Theta}(\boldsymbol{\theta}))$ and $\mathbf{s}(\boldsymbol{\Theta}(\boldsymbol{\theta}))$ are also functions of the material coordinates through the relation with the surface coordinates.

The Vertex Morphing Method defines the relation between the surface geometry $\mathbf{z}$ and the design control field $\mathbf{s}$ in the following form:

$$\mathbf{z}(\boldsymbol{\Theta}_0) = \int_{\Gamma_{\text{VM}}} F\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}_0, r\right) \mathbf{s}(\boldsymbol{\Theta}) \, \mathrm{d}\Gamma \, . \tag{3.13}$$

F denotes the (forward) filter function with a center of its support at $\boldsymbol{\Theta}_0$ and influencing the portion of the domain Γ_{VM} with radius r. The choices

of the filter function and the filter radius control the generated surface geometry via controlling the curvature. The filter function should satisfy the following property:

$$\int_{\Gamma_{\mathrm{VM}}} F\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}_0, r\right) \mathrm{d}\Gamma = 1 \,. \tag{3.14}$$

It should be noted that this condition rises the need for the modification of filter functions so that the sensitivity field is not scaled or the unit integration property is ensured close to the domain edges [8]. Additionally, the backward filter function is related to the forward filter function as:

$$A\left(\boldsymbol{\Theta}_0, \boldsymbol{\Theta}, r\right) = F\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}_0, r\right) \,. \tag{3.15}$$

In case the following condition holds, the filter function is said to be symmetric or self-adjoint:

$$F\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}_0, r\right) = F\left(\boldsymbol{\Theta}_0, \boldsymbol{\Theta}, r\right) = A\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}_0, r\right) \,. \tag{3.16}$$

In context of the shape optimization, the increments of the shape updates are also necessary to be defined. Similar to the Eqn. (3.13), the shape variable updates are related through the filter function:

$$\Delta\mathbf{z} = \int_{\Gamma_{\mathrm{VM}}} F \Delta\mathbf{s} \, \mathrm{d}\Gamma \,. \tag{3.17}$$

In addition, making use of Eqn. (3.13), the derivative of the shape with respect to the control field can be defined:

$$\frac{\mathrm{d}\mathbf{z}\left(\boldsymbol{\Theta}_0\right)}{\mathrm{d}\mathbf{s}\left(\boldsymbol{\Theta}_1\right)} = F\left(\boldsymbol{\Theta}_0, \boldsymbol{\Theta}_1, r\right) \,. \tag{3.18}$$

Since the optimization is driven on the control space, the derivative of the objective function with respect to the control field should be computed. Using the relations defined in Eqn. (3.15) and Eqn. (3.18):

$$\begin{aligned} \frac{\mathrm{d}J}{\mathrm{d}\mathbf{s}} &= \int_{\Gamma_{\mathrm{VM}}} \frac{\mathrm{d}J}{\mathrm{d}\mathbf{z}\left(\boldsymbol{\Theta}_1\right)} \frac{\mathrm{d}\mathbf{z}\left(\boldsymbol{\Theta}_1\right)}{\mathrm{d}\mathbf{s}} \, \mathrm{d}\Gamma \\ &= \int_{\Gamma_{\mathrm{VM}}} \frac{\mathrm{d}J}{\mathrm{d}\mathbf{z}\left(\boldsymbol{\Theta}_1\right)} F\left(\boldsymbol{\Theta}_1, \boldsymbol{\Theta}, r\right) \, \mathrm{d}\Gamma = \int_{\Gamma_{\mathrm{VM}}} A\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}_1, r\right) \frac{\mathrm{d}J}{\mathrm{d}\mathbf{z}\left(\boldsymbol{\Theta}_1\right)} \, \mathrm{d}\Gamma \,. \end{aligned} \tag{3.19}$$

In order to carry out numerical analysis on the defined fields, the discretizations of the defined relations are necessary. The design control field can be discretized with the shape functions related to the design handles. Incorporating the shape functions for discretization in Eqn. (3.13) yields:

$$\begin{aligned}\hat{z}_{(i)} &= \int_{\Gamma_{\mathrm{VM}}} F_{(i)} N_{(j)} \hat{s}_{(j)} \, \mathrm{d}\Gamma = \int_{\Gamma_{\mathrm{VM}}} F_{(i)} N_{(j)} \, \mathrm{d}\Gamma \hat{s}_{(j)} \\ &= B_{(i,j)} \hat{s}_{(j)} \,, \end{aligned} \tag{3.20}$$

where the defined transformation operator $B_{(i,j)}$ can also be related to the derivative of the shape with respect to the control field in combination with Eqn. (3.18):

$$\frac{\mathrm{d}\hat{z}_{(i)}}{\mathrm{d}\hat{s}_{(j)}} = \int_{\Gamma_{\mathrm{VM}}} F_{(i)} N_{(j)} \, \mathrm{d}\Gamma = B_{(i,j)} \,. \tag{3.21}$$

Incorporating Eqn. (3.21) in Eqn. (3.22), it is possible to relate the derivatives of the objective function with respect to design control parameters and the derivatives of the objective function with respect to shape:

$$\frac{\mathrm{d}J}{\mathrm{d}\hat{s}_{(j)}} = \int_{\Gamma_{\mathrm{VM}}} \frac{\mathrm{d}J}{\mathrm{d}\hat{\mathbf{z}}} \frac{\mathrm{d}\hat{\mathbf{z}}}{\mathrm{d}\hat{s}_{(j)}} \, \mathrm{d}\Gamma = \int_{\Gamma_{\mathrm{VM}}} N_{(j)} F_{(i)} \frac{\mathrm{d}J}{\mathrm{d}\hat{z}_{(i)}} \, \mathrm{d}\Gamma = B_{(j,i)} \frac{\mathrm{d}J}{\mathrm{d}\hat{z}_{(i)}} \,. \tag{3.22}$$

One can easily observe from Eqn. (3.22) that the parameter updates on the control field $\hat{\mathbf{s}}$ are mapped forward using the $\mathbf{B}$ operator onto the shape field $\hat{\mathbf{z}}$ and the sensitivity derivatives of the objective function with respect to the control field are mapped backward using the operator $\mathbf{B}^{\mathrm{T}}$. The operator $\mathbf{B}$ is symmetric only when specific requirements are met for the choice of the filter function [8]. Moreover, it is worth mentioning that when the filter function is defined as a Dirac delta function, the filtering property is cancelled and the operator $\mathbf{B}$ falls back to the identity matrix $\mathbf{I}$. This property is exploited for the selective use of the Vertex Morphing Method in Sec. 3.2.4.

Shape Update Rules and the Optimality Condition

The investigation of the shape update rules and the optimality condition follows from the Taylor series expansion of the objective function:

$$\tilde{J} = J + \nabla_{\hat{\mathbf{s}}} J^{\mathrm{T}} \Delta\hat{\mathbf{s}} + \frac{1}{2} \Delta\hat{\mathbf{s}}^{\mathrm{T}} \nabla^2_{\hat{\mathbf{s}}} J \Delta\hat{\mathbf{s}}$$

$$= J + \nabla_{\hat{\mathbf{z}}} J^{\mathrm{T}} \mathbf{B} \Delta\hat{\mathbf{s}} + \frac{1}{2} \Delta\hat{\mathbf{s}}^{\mathrm{T}} \mathbf{B}^{\mathrm{T}} \nabla^2_{\hat{\mathbf{z}}} J \, \mathbf{B} \Delta\hat{\mathbf{s}} \, . \tag{3.23}$$

Introducing the Hessian $\mathbf{H}$, the derived transformation rules in Eqn. (3.23) and the stationary condition one can derive the shape update rules:

$$\begin{aligned} &\nabla_{\hat{\mathbf{s}}} J = \mathbf{B}^{\mathrm{T}} \nabla_{\hat{\mathbf{z}}} J + \mathbf{B}^{\mathrm{T}} \mathbf{H}_{\hat{\mathbf{z}}} \mathbf{B} \Delta\hat{\mathbf{s}} = 0 \, , \\ &\Delta\hat{\mathbf{z}} = -\mathbf{B} \, \mathbf{H}_{\hat{\mathbf{s}}}^{-1} \mathbf{B}^{\mathrm{T}} \nabla_{\hat{\mathbf{z}}} J \, . \end{aligned} \tag{3.24}$$

In addition, introducing the step length factor α and $\mathbf{M} = \mathbf{H}_{\hat{\mathbf{s}}}^{-1}$ yields:

$$\Delta\hat{\mathbf{z}} = -\alpha \, \mathbf{B} \, \mathbf{M} \, \mathbf{B}^{\mathrm{T}} \nabla_{\hat{\mathbf{z}}} J \, . \tag{3.25}$$

In the above equation, one can replace $\mathbf{M}$ with the identity matrix and result in the steepest descent method, [8]. Obviously, the sensitivity field is effected by the mapping to the control field and back with the transformation matrix $\mathbf{B}$.

Moreover, following from Eqns. (3.20)-(3.22) and making use of the transformations between the fields $\hat{\mathbf{s}}$ and $\hat{\mathbf{z}}$:

$$\begin{aligned} \Delta\hat{\mathbf{z}} &= -\mathbf{B} \left(\mathbf{B}^{-1} \mathbf{H}_{\hat{\mathbf{z}}}^{-1} \mathbf{B}^{-\mathrm{T}} \right) \mathbf{B}^{\mathrm{T}} \nabla_{\hat{\mathbf{z}}} J \\ &= -\mathbf{H}_{\hat{\mathbf{z}}}^{-1} \nabla_{\hat{\mathbf{z}}} J \, . \end{aligned} \tag{3.26}$$

The Eqn. (3.26) reveals that as long as the chosen filter function results in an invertible $\mathbf{B}$ operator, the transformation of the optimization problem to a control space does not modify the local or global optima of the original problem.

3.2.2 The Mortar Mapping Method and Filtering Properties

In this section, the relation between the Vertex Morphing filter operator and the mapping operator of the Mortar Mapping method is discussed. Mortar methods can be used to enforce equality constraints or equality of the fields on dissimilar discretizations of the same geometric entities in a weak sense. In such a case, a transformation of a field $\boldsymbol{\chi}$ defined on a surface with discretization $\hat{\mathbf{z}}$ can be written in terms of discretization $\hat{\mathbf{p}}$ using Mortar methods [39], [75].

Consider an optimization problem similar to Eqn. (3.12) and introduce an equality constraint for the sensitivity and shape update fields defined on

the different discretizations of $\hat{\mathbf{z}}$ and $\hat{\mathbf{p}}$:

$$\begin{aligned} \min_{\mathbf{p}} \quad & J(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{u}) \\ s.t. \quad & \mathbf{R}_i(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{u}) = 0; \quad i = 1, \ldots, k \\ & g_0^{\Gamma}(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{p}, \mathbf{u}) = 0 \\ & g_j(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{u}) \leq 0; \quad j = 1, \ldots, m \end{aligned} \tag{3.27}$$

where the introduced constraint g_0 is:

$$\boldsymbol{\chi}_{\mathrm{p}} - \boldsymbol{\chi}_{\mathrm{z}} = 0 \quad \text{on} \quad \Gamma\,. \tag{3.28}$$

In the above equation $\boldsymbol{\chi}$ field can be the sensitivity field as well as the shape update field. Making use of the Lagrange multipliers approach for this constraint, one can extend the Lagrangian with Eqn. (3.28):

$$\widetilde{\mathcal{L}} = \mathcal{L} + \int_{\Gamma} \boldsymbol{\lambda} \cdot \left(\boldsymbol{\chi}_{\mathrm{p}} - \boldsymbol{\chi}_{\mathrm{z}} \right) \mathrm{d}\Gamma\,. \tag{3.29}$$

In the above equation $\mathcal{L}$ consists of the objective function, state equations $\mathbf{R}_i$ as constraints and the other constraints g_j imposed with an appropriate method of choice. Enforcing the equality constraint in Eqn. (3.29) requires the solution of the Lagrange multipliers and increases the number of variables to solve for. Instead, the Mortar method can be employed in order to avoid the solution of the Lagrange multipliers field. The Mortar method stems from the shape function choice for the Lagrange multiplier field's discretization to coincide with one of the fields to enforce the equality condition. The chosen discretization is then called "the master field", whereas the remainder is referred as "the slave field" [76]. For instance, if the master field is chosen as $\boldsymbol{\chi}_{\mathrm{p}}$, Eqn. (3.29) in discrete form reads:

$$\widetilde{\mathcal{L}} = \mathcal{L} + \int_{\Gamma} \hat{\boldsymbol{\lambda}}^{\mathrm{T}} \left(\mathbf{N}_{\mathrm{p}}^{\mathrm{T}} \mathbf{N}_{\mathrm{p}} \hat{\boldsymbol{\chi}}_{\mathrm{p}} - \mathbf{N}_{\mathrm{p}}^{\mathrm{T}} \mathbf{N}_{\mathrm{z}} \hat{\boldsymbol{\chi}}_{\mathrm{z}} \right) \mathrm{d}\Gamma\,, \tag{3.30}$$

where the $\mathbf{N}_{\mathrm{z}}$ and $\mathbf{N}_{\mathrm{p}}$ refer to the shape function matrices related to the discretizations of the fields defined on $\hat{\mathbf{z}}$ and $\hat{\mathbf{p}}$. Having Eqn. (3.30) and making use of the fact that the variation of the integral term should vanish for any discrete set of $\hat{\boldsymbol{\lambda}}$, the discrete field $\hat{\boldsymbol{\chi}}_{\mathrm{p}}$ can be obtained as follows:

$$\hat{\boldsymbol{\chi}}_{\mathrm{p}} = \mathbf{C}_{\mathrm{pp}}^{-1} \mathbf{C}_{\mathrm{pz}} \hat{\boldsymbol{\chi}}_{\mathrm{z}} \tag{3.31a}$$

$$= \mathbf{B}_{\mathrm{pz}} \hat{\boldsymbol{\chi}}_{\mathrm{z}}\,. \tag{3.31b}$$

where the defined matrices $\mathbf{C}_{\mathrm{pp}}$, $\mathbf{C}_{\mathrm{pz}}$ and $\mathbf{B}_{\mathrm{pz}}$ are;

$$\mathbf{C}_{\mathrm{pp}} = \int_{\Gamma} \mathbf{N}_{\mathrm{p}}^{\mathrm{T}} \mathbf{N}_{\mathrm{p}} \, \mathrm{d}\Gamma \,, \tag{3.32a}$$

$$\mathbf{C}_{\mathrm{pz}} = \int_{\Gamma} \mathbf{N}_{\mathrm{p}}^{\mathrm{T}} \mathbf{N}_{\mathrm{z}} \, \mathrm{d}\Gamma \,, \tag{3.32b}$$

$$\mathbf{B}_{\mathrm{pz}} = \mathbf{C}_{\mathrm{pp}}^{-1} \mathbf{C}_{\mathrm{pz}} \,. \tag{3.32c}$$

By doing so, one defines the transformation operator $\mathbf{B}_{\mathrm{pz}}$ from $\hat{\boldsymbol{\chi}}_{\mathrm{z}}$ to $\hat{\boldsymbol{\chi}}_{\mathrm{p}}$. In the context of Mortar Mapping, mentioned operator is also called a consistent mapping operator which preserves a constant field when transformed between different discretizations and computes a weak equivalent of a discrete field on a dissimilar discretization. Moreover, the reverse mapping operator from $\hat{\boldsymbol{\chi}}_{\mathrm{p}}$ to $\hat{\boldsymbol{\chi}}_{\mathrm{z}}$ can be defined by choosing the discretization of the Lagrange multipliers field to coincide with the discretization of $\hat{\boldsymbol{\chi}}_{\mathrm{z}}$ and following the procedure in Eqns. (3.30)-(3.32), which yields the following:

$$\hat{\boldsymbol{\chi}}_{\mathrm{z}} = \mathbf{B}_{\mathrm{zp}} \hat{\boldsymbol{\chi}}_{\mathrm{p}} \tag{3.33}$$

The explicit computation of the transformation matrix is a computationally costly operation since n_{z}^{DOF} linear equation solutions for the computation of $\mathbf{B}_{\mathrm{pz}}$ and n_{p}^{DOF} linear equation solutions for the computation of $\mathbf{B}_{\mathrm{zp}}$ would be necessary. Instead, the so called mass matrices $\mathbf{C}_{\mathrm{pp}}$ or $\mathbf{C}_{\mathrm{zz}}$ are factorized only once and the RHS is obtained through a relatively cheap matrix vector multiplication every time a field is to be mapped.

Relation to the Vertex Morphing method

Having the Mortar transformation operators defined for the sensitivity and the shape update fields, one can investigate the effect of introducing the shape control field $\hat{\mathbf{p}}$ as a separate discretization of the design field $\hat{\mathbf{z}}$. One has;

$$\frac{\mathrm{d}J}{\mathrm{d}\hat{\mathbf{p}}} = \mathbf{B}_{\mathrm{pz}} \frac{\mathrm{d}J}{\mathrm{d}\hat{\mathbf{z}}} \,, \tag{3.34a}$$

$$\Delta\hat{\mathbf{z}} = \mathbf{B}_{\mathrm{zp}} \Delta\hat{\mathbf{p}} \,. \tag{3.34b}$$

Similar to Eqn. (3.23), one can investigate the optimality condition by using Taylor series expansion for the objective function with respect to field $\hat{\mathbf{p}}$:

$$\begin{aligned}\tilde{J} &= J + \nabla_{\hat{\mathbf{p}}} J^{\mathrm{T}} \Delta\hat{\mathbf{p}} + \frac{1}{2} \Delta\hat{\mathbf{p}}^{\mathrm{T}} \nabla^2_{\hat{\mathbf{p}}} J \Delta\hat{\mathbf{p}} \\ &= J + \nabla_{\hat{\mathbf{z}}} J^{\mathrm{T}} \mathbf{B}^{\mathrm{T}}_{\mathrm{pz}} \Delta\hat{\mathbf{p}} + \frac{1}{2} \Delta\hat{\mathbf{p}}^{\mathrm{T}} \nabla^2_{\hat{\mathbf{p}}} J \Delta\hat{\mathbf{p}} \,. \end{aligned} \tag{3.35}$$

Making use of the stationary condition and introducing the following equality; $\mathbf{M} = \left(\mathbf{H}_{\hat{\mathbf{p}}}\right)^{-1}$, one has:

$$\Delta\hat{\mathbf{z}} = -\mathbf{B}_{\mathrm{zp}} \mathbf{M} \mathbf{B}_{\mathrm{pz}} \nabla_{\hat{\mathbf{z}}} J \,. \tag{3.36}$$

The similarity between the Eqn. (3.36) and the Eqn. (3.25) is obvious. In addition, in order to employ the steepest descent algorithm for the optimization, one can adopt $\mathbf{M} = \mathbf{I}$ and introduce the step length factor α:

$$\Delta\hat{\mathbf{z}} = -\alpha \mathbf{B}_{\mathrm{zp}} \mathbf{B}_{\mathrm{pz}} \nabla_{\hat{\mathbf{z}}} J \,. \tag{3.37}$$

One can adopt the Mortar Mapping method to transfer the design variables from the design space to the design control space by mapping the computed sensitivity field on the discrete design space $\hat{\mathbf{z}}$ with the mapping operator $\mathbf{B}_{\mathrm{pz}}$, perform an optimization step on the control space and map the design update back onto the design space using the $\mathbf{B}_{\mathrm{zp}}$ itself. The mapping operations cause a change in the search direction similar to the Vertex Morphing method. In principle, the design updates are forced to follow a pattern that can be expressed by the discretization of the field $\hat{\mathbf{p}}$ and adopts the properties of this field, e.g. smoothness, curvature, etc.

The necessity and the use of this method are prominent when the design surface is discretized dissimilarly and the discrete sensitivity fields are computed on non-matching grids, e.g. in case of surface coupled MDO problems where the design surface is shared by the constituting subdisciplines. Moreover, as discussed in the next section, this type of mapping operations can be exploited in order to regenerate CAD models as a by-product of optimization routines.

3.2.3 CAD-Based Mortar Mapping Method and Embedded Constraint Handling

The underlying parametrization of the CAD model can be utilized as a basis for the variable transformation and as a shape control field. In this study, the NURBS shape functions of the CAD model are favoured for their wide applicability in CAD programs as well as in numerical computational fields such as isogeometric analysis [77]. Moreover, unlike the finite element discretizations, where the nodal degrees of freedom can be coupled strongly, multi-patch NURBS geometries often have non-matching discretizations along their coupling interfaces. In practice, it is observed that this aspect causes discontinuous control field deformations and their derivatives - namely the rotations- across the patch interfaces since the DOF of each individual patch would have been decoupled in the coupling matrices. The continuity across the patch interfaces can well be enforced by introducing them as constraints to the optimization problem in addition to the Mortar constraint as follows:

$$\begin{aligned} \min_{\mathbf{p}} \quad & J(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{u}) \\ s.t. \quad & \mathbf{R}_i(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{u}) = 0; \qquad i = 1, \ldots, m \\ & g_0^{\Gamma}\left(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{p}, \mathbf{u}\right) = 0 \\ & g_j^{\gamma_{qr}}\left(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{p}, \mathbf{u}\right) = 0; \quad j = 1, \ldots, n \\ & g_k(\boldsymbol{\Theta}, \mathbf{z}, \mathbf{u}) \leq 0; \qquad k = 1, \ldots, l \end{aligned} \tag{3.38}$$

The introduced constraint $g_j^{\gamma_{qr}}$ enforces the continuity condition across the interface γ between the patches q and r of the CAD design control space:

$$\boldsymbol{\chi}_{\mathrm{p}}^{\gamma_q} - \boldsymbol{\chi}_{\mathrm{p}}^{\gamma_r} = 0\,. \tag{3.39}$$

This constraint can as well be enforced by making use of the Lagrange multipliers similar to Eqn. (3.29):

$$\mathcal{L}^* = \widetilde{\mathcal{L}} + \int_{\gamma_{qr}} \boldsymbol{\lambda}_{qr} \cdot \left(\boldsymbol{\chi}_{\mathrm{p}}^{\gamma_q} - \boldsymbol{\chi}_{\mathrm{p}}^{\gamma_r}\right) \mathrm{d}\gamma\,. \tag{3.40}$$

In order to incorporate the newly introduced constraints into the optimization problem and simultaneously achieve transformation of the optimization problem to the control space one can again make use of the

Mortar method for the DOFs related to the patch boundary deformations. Unlike the methodology used previously by constructing the RHS by a matrix-vector multiplication and performing only one solution step, one would need to construct the transformation matrix by performing n_q^{DOF} and n_r^{DOF} solves respectively at each patch interface level, rearrange $\mathbf{C}_{\text{pp}}$ and assemble the corresponding coupling contributions, see also [78] for the assembly procedures. In order to avoid the costly operations, one can approximate the Lagrange multipliers field as in [79]:

$$\boldsymbol{\lambda}_{qr} \cong \frac{\alpha_{\gamma_{qr}}}{2}\left(\boldsymbol{\chi}_{\text{p}}^{\gamma_q} - \boldsymbol{\chi}_{\text{p}}^{\gamma_r}\right), \tag{3.41}$$

which results in a quadratic penalty formulation for the patch coupling constraints:

$$\mathcal{L}^* = \widetilde{\mathcal{L}} + \frac{\alpha_{\gamma_{qr}}}{2}\int_{\gamma_{qr}} \left(\boldsymbol{\chi}_{\text{p}}^{\gamma_q} - \boldsymbol{\chi}_{\text{p}}^{\gamma_r}\right)\cdot\left(\boldsymbol{\chi}_{\text{p}}^{\gamma_q} - \boldsymbol{\chi}_{\text{p}}^{\gamma_r}\right) \mathrm{d}\gamma\,. \tag{3.42}$$

Here only the continuity of the fields on the shape control space are considered for brevity. Similarly, one can also enforce the C^1 continuity across the patch interfaces as done in [80]. Moreover, in order to avoid a mixed formulation with both Lagrange multipliers and penalty methods, one can similarly approximate the Lagrange multipliers field for the constraint g_0^{Γ}. It can be shown that as long as the penalty parameter is chosen accordingly, i.e. $\alpha_{\Gamma} \neq 0$, such an approximation for the condition g_0^{Γ} leads to the same formulation as in the Mortar method since it can well be considered as a norm minimization of the $L^2(\Gamma)$ space, see also [81] for the norm minimization point of view.

Finally, the Lagrangian of the optimization problem reads;

$$\begin{aligned}\mathcal{L}^* = \mathcal{L} &+ \frac{\alpha_{\Gamma}}{2}\int_{\Gamma}\left(\boldsymbol{\chi}_{\text{p}} - \boldsymbol{\chi}_{\text{z}}\right)\cdot\left(\boldsymbol{\chi}_{\text{p}} - \boldsymbol{\chi}_{\text{z}}\right) \mathrm{d}\Gamma \\ &+ \frac{\alpha_{\gamma_{qr}}}{2}\int_{\gamma_{qr}} \left(\boldsymbol{\chi}_{\text{p}}^{\gamma_q} - \boldsymbol{\chi}_{\text{p}}^{\gamma_r}\right)\cdot\left(\boldsymbol{\chi}_{\text{p}}^{\gamma_q} - \boldsymbol{\chi}_{\text{p}}^{\gamma_r}\right) \mathrm{d}\gamma\,.\end{aligned} \tag{3.43}$$

Since a staggered solution of the optimization problem is adopted, the variation of the integral terms can be set to zero separate from the variation of the Lagrangian $\mathcal{L}$. Herein, a staggered solution is understood as the sensitivity analysis with respect to the discrete design $\hat{\mathbf{z}}$, the consecutive

sensitivity filtering operation and finally the shape update computation. Setting the variation of the integral terms to zero yields;

$$\alpha_\Gamma \int_\Gamma \left(\delta\boldsymbol{\chi}_\mathrm{p} - \delta\boldsymbol{\chi}_\mathrm{z}\right)\cdot\left(\boldsymbol{\chi}_\mathrm{p} - \boldsymbol{\chi}_\mathrm{z}\right)\,\mathrm{d}\Gamma + \alpha_{\gamma_{qr}} \int_{\gamma_{qr}} \left(\delta\boldsymbol{\chi}_\mathrm{p}^{\gamma_q} - \delta\boldsymbol{\chi}_\mathrm{p}^{\gamma_r}\right)\cdot\left(\boldsymbol{\chi}_\mathrm{p}^{\gamma_q} - \boldsymbol{\chi}_\mathrm{p}^{\gamma_r}\right)\,\mathrm{d}\gamma = 0\,. \tag{3.44}$$

Introducing the shape function matrices for the respective discretizations of the fields and making use of the fact that the equations should hold for all the admissible variations, two discrete equation systems are found:

$$\left[\alpha_\Gamma \mathbf{C}_\mathrm{pp} + \alpha_{\gamma_{qr}} \mathbf{C}_\mathrm{pp}^{\gamma_{qr}}\right]\hat{\boldsymbol{\chi}}_\mathrm{p} = \alpha_\Gamma \mathbf{C}_\mathrm{pz}\hat{\boldsymbol{\chi}}_\mathrm{z}\,, \tag{3.45a}$$

$$\alpha_\Gamma \mathbf{C}_\mathrm{zz}\hat{\boldsymbol{\chi}}_\mathrm{p} = \alpha_\Gamma \mathbf{C}_\mathrm{zp}\hat{\boldsymbol{\chi}}_\mathrm{p}\,. \tag{3.45b}$$

By doing so, one can express the relation between the fields $\boldsymbol{\chi}_\mathrm{z}$ and $\boldsymbol{\chi}_\mathrm{p}$:

$$\hat{\boldsymbol{\chi}}_\mathrm{p} = \mathbf{B}_\mathrm{pz}\hat{\boldsymbol{\chi}}_\mathrm{z}\,, \tag{3.46a}$$

$$\hat{\boldsymbol{\chi}}_\mathrm{z} = \mathbf{B}_\mathrm{zp}\hat{\boldsymbol{\chi}}_\mathrm{p}\,. \tag{3.46b}$$

From Eqn. (3.45a) it is clear that the value of the penalty parameter α_Γ can be treated as a scaling factor for the equations and can be set to 1. The transformation matrices can be defined as follows;

$$\mathbf{B}_\mathrm{pz} = \left[\mathbf{C}_\mathrm{pp} + \alpha_{\gamma_{qr}} \mathbf{C}_\mathrm{pp}^{\gamma_{qr}}\right]^{-1} \mathbf{C}_\mathrm{pz}\,, \tag{3.47a}$$

$$\mathbf{B}_\mathrm{zp} = \mathbf{C}_\mathrm{zz}^{-1}\mathbf{C}_\mathrm{zp}\,. \tag{3.47b}$$

It can be concluded that the quadratic penalty formulation for the domain leads to the minimization of the norm defined on $L^2(\Gamma)$, since the penalty parameter α_Γ can eventually be eliminated from the system. Eqn. (3.47a) enhances the coupling matrix with the additional constraints defined for the patch interfaces. The advantage of such a formulation is the ease in the application of the patch coupling constraints. Even though the quadratic penalty formulation for optimization problems leads to a violation of the constraints by definition [68], the CAD geometries are in practice rarely water tight along coupled patch boundaries and with the correct selection of the penalty factors, the constraint violation does not exceed the

desired tolerance. Moreover, a linear penalty approach results the constraint violation related contribution to appear on the RHS of Eqn. (3.45a). This necessitates an iterative solution procedure which is not desirable. In addition, [81] defines a method for an automatic selection of the penalty parameters for both the field itself and its derivative, which is also the existing implementation in EMPIRE.

Having the definitions in this section, one can extend the formulation for the application of the geometric equality constraints. In addition to the patch coupling constraints, one can define equality constraints for both a portion of the domain boundary γ_q and a portion of the domain Γ_q. The Lagrangian in this case reads:

$$\begin{aligned}\mathcal{L} = J &+ \frac{\alpha_\Gamma}{2}\int_\Gamma \left(\boldsymbol{\chi}_\mathrm{p} - \boldsymbol{\chi}_\mathrm{z}\right)\cdot\left(\boldsymbol{\chi}_\mathrm{p} - \boldsymbol{\chi}_\mathrm{z}\right)\,\mathrm{d}\Gamma \\ &+ \frac{\alpha_{\gamma_{qr}}}{2}\int_{\gamma_{qr}} \left(\boldsymbol{\chi}_\mathrm{p}^{\gamma_q} - \boldsymbol{\chi}_\mathrm{p}^{\gamma_r}\right)\cdot\left(\boldsymbol{\chi}_\mathrm{p}^{\gamma_q} - \boldsymbol{\chi}_\mathrm{p}^{\gamma_r}\right)\,\mathrm{d}\gamma \\ &+ \frac{\alpha^\mathrm{c}_{\gamma_q}}{2}\int_{\gamma_q} \boldsymbol{\chi}_\mathrm{p}^{\gamma_q}\cdot\boldsymbol{\chi}_\mathrm{p}^{\gamma_q}\,\mathrm{d}\gamma + \frac{\alpha^\mathrm{c}_{\Gamma_q}}{2}\int_{\Gamma_q} \boldsymbol{\chi}_\mathrm{p}^{\Gamma_q}\cdot\boldsymbol{\chi}_\mathrm{p}^{\Gamma_q}\,\mathrm{d}\Gamma\,. \end{aligned} \tag{3.48}$$

Applying the same procedure as in Eqns. (3.44)-(3.47), one obtains the following transformation matrices:

$$\mathbf{B}^*_\mathrm{pz} = \left[\mathbf{C}_\mathrm{pp} + \alpha_{\gamma_{qr}}\mathbf{C}^{\gamma_{qr}}_\mathrm{pp} + \alpha^\mathrm{c}_{\gamma_q}\mathbf{C}^{\gamma_q}_\mathrm{pp} + \alpha^\mathrm{c}_{\Gamma_q}\mathbf{C}^{\Gamma_q}_\mathrm{pp}\right]^{-1}\mathbf{C}_\mathrm{pz}\,, \tag{3.49a}$$

$$\mathbf{B}_\mathrm{zp} = \mathbf{C}^{-1}_\mathrm{zz}\mathbf{C}_\mathrm{zp}\,. \tag{3.49b}$$

Eqn. (3.49a) defines an enhanced transformation matrix $\mathbf{B}^*_\mathrm{pz}$ from $\hat{\boldsymbol{\chi}}_\mathrm{z}$ to $\hat{\boldsymbol{\chi}}_\mathrm{p}$, which not only acts as a sensitivity filter but also embeds the equality constraints of the optimization problem to the filtering operator. The shape update rules and the optimality condition which are discussed for the Mortar Mapping method in Sec. 3.2.2 still hold since the methodology for the shape update computation does not differ from the one of the general Mortar Mapping method. It is important to note that neither patch coupling conditions nor the equality constraints of the optimization problem need to be applied on the design field, since they are already considered in the control field. Thus, the shape updates follow the constrained shape control field.

Definition of the filtering operator in such a manner paves the way for two possible uses:

1. One can apply the mapping operators individually by firstly mapping the design field sensitivities onto the shape control field sensitivities and have the possibility to drive the optimization process directly on the CAD model. Then, the forward mapping operator is used for updating the discrete numerical analysis model. This ensures the conformity of the CAD model and the discrete numerical model by the instantaneous updates of the CAD surface throughout the optimization process.

2. Alternatively, the mapping operators can be used consecutively to retrieve the filtered sensitivity field and the optimization can be driven on the discrete numerical model.

3.2.4 Combined Use of Vertex Morphing and Mortar Mapping Methods

Generating the CAD model of the optimized shape is of high interest in engineering applications. In order to achieve this, one needs to generate the connection between the desired shape updates and the underlying CAD description of the model. This need can be satisfied by making use of the defined Mortar operators. On the other hand, driving the optimization on an a priori defined CAD description restricts the design space since the design handles allow only certain type of deformations and causes a filtering effect as mentioned in Sec. 3.2.3 which might not be desirable for exploring the design space. In order to overcome this problem and simultaneously generate the CAD model, one can adopt both filtering techniques for node based optimization and the methods for tying the discrete model to the CAD model.

The Vertex Morphing method and the Mortar Mapping method with the embedded constraint handling operators are previously defined. One can fuse the advantages of both methods into a combined operator by a corresponding choice of a filter function for the Vertex Morphing method, appropriate discretization for the CAD model and mentioned penalty parameters for the Mortar Mapping method. Since the NURBS shape functions are assumed for the discretization of the surface, one can introduce

an adjustable or in this context adequately fine discretization of the CAD model such that it can represent the occurring shape updates.

Following from Eqn. (3.25) that defines the shape update rule for the Vertex Morphing method, one can introduce a selection of choices for the operator $\mathbf{M}$. For instance, introducing $\mathbf{M} = \mathbf{B}^{-1}\mathbf{B}_{\mathrm{zp}}\mathbf{B}_{\mathrm{pz}}^{*}\mathbf{B}$ one obtains the following:

$$\begin{aligned}\Delta\hat{\mathbf{z}} &= -\alpha\,\mathbf{B}\,\mathbf{M}\,\mathbf{B}^{T}\nabla_{\hat{\mathbf{z}}}J \\ &= -\alpha\,\mathbf{B}\,\mathbf{B}^{-1}\mathbf{B}_{\mathrm{zp}}\mathbf{B}_{\mathrm{pz}}^{*}\mathbf{B}\,\mathbf{B}^{T}\nabla_{\hat{\mathbf{z}}}J \\ &= -\alpha\,\mathbf{B}_{\mathrm{zp}}\mathbf{B}_{\mathrm{pz}}^{*}\mathbf{B}\,\mathbf{B}^{T}\nabla_{\hat{\mathbf{z}}}J\ . \end{aligned} \tag{3.50}$$

Above equation performs the backward and forward filtering of the sensitivity field with the Vertex Morphing method and computes an initial design update. The computed design update is mapped onto the CAD description of the underlying geometry and updates the CAD model. If any equality constraints are defined, they can be enforced through the mapping operation onto the CAD geometry. In order to ensure the updates of the CAD model and the simulation model, the CAD shape updates are mapped back onto the discrete numerical model.

This type of formulation has the advantage of selectively applying the node based sensitivity filtering and updating the CAD model simultaneously. If the filter function for the Vertex Morphing method is chosen as Dirac delta function, the operator $\mathbf{B}$ results in an identity matrix $\mathbf{I}$ as mentioned in Sec. 3.2.1 and the filtering property cancels out. Nevertheless, the filtering effect of the underlying CAD model remains. In case the penalty parameter for the Mortar method is chosen as $\alpha_{\Gamma} = 0$, then the updating of the CAD model as well as its filtering properties cancel out. The same holds for the geometric equality constraints of the optimization problem. When the penalty parameter for the Mortar method is chosen as $\alpha_{\Gamma} = 1$ and the penalty parameters for the constraints are chosen $\alpha_{(*)}^{\mathrm{c}} = 0$, then the Vertex Morphing filtering and the CAD updates are held while the application of the geometric constraints are cancelled out. See Table 3.1 for different variants and the resulting shape update rule.

Vertex Morphing	CAD Update	Geometrical Constraint	Shape Update Rule
$F \neq \delta$	$\alpha_\Gamma = 0$	$\alpha^c_{\gamma_q} = 0$	$\Delta\hat{\mathbf{z}} = -\alpha \mathbf{B}\mathbf{B}^T \nabla_{\hat{\mathbf{z}}} J$
$F = \delta$	$\alpha_\Gamma = 1$	$\alpha^c_{\gamma_q} \neq 0$	$\Delta\hat{\mathbf{z}} = -\alpha \mathbf{B}_{\mathrm{zp}} \mathbf{B}^*_{\mathrm{pz}} \nabla_{\hat{\mathbf{z}}} J$
$F \neq \delta$	$\alpha_\Gamma = 1$	$\alpha^c_{\gamma_q} = 0$	$\Delta\hat{\mathbf{z}} = -\alpha \mathbf{B}_{\mathrm{zp}} \mathbf{B}_{\mathrm{pz}} \mathbf{B}\mathbf{B}^T \nabla_{\hat{\mathbf{z}}} J$
$F \neq \delta$	$\alpha_\Gamma = 1$	$\alpha^c_{\gamma_q} \neq 0$	$\Delta\hat{\mathbf{z}} = -\alpha \mathbf{B}_{\mathrm{zp}} \mathbf{B}^*_{\mathrm{pz}} \mathbf{B}\mathbf{B}^T \nabla_{\hat{\mathbf{z}}} J$

Table 3.1: Variants for using Vertex Morphing and CAD update in a combined manner.

3.3 MDO Framework and CAD integration

The shape optimization of the structures in interaction with fluids is a complicated task that brings many tools for simulation and optimization together. Chp. 2 summarized the developed methods for the FSI analysis that is necessary prior to the optimization tasks in order to obtain the equilibrium condition. Sec. 3.1 collects the sensitivity analysis methods that is used in the scope of this work and their advantages. Sec. 3.2 explores explicit sensitivity filtering methods and the possibility of combined use when the regeneration of the CAD surfaces as a by-product of the optimization task is necessary. This section finally brings the explored and developed methods together to form a MDO framework, which performs the shape optimization of structures that are coupled in an FSI environment.

3.3.1 MDO Framework

In order to construct a framework for the optimization tasks in this work, one needs to meet certain decisions depending on many criteria. One of the many is the software that establishes the connection between the various components. The in-house open-source software EMPIRE offers a great possibility for setting up the communication patterns as well as achieving the coupling between the disciplinary solvers. Moreover, both the classical FEM-FEM as well as FEM-IGA Mortar Mapping implementations are available within EMPIRE which are made available as a library during the code compilation.

Secondly, another choice to be made is the software for the computation of the disciplinary states. In order to achieve an FSI analysis, one needs to couple the software that perform the structural analysis and the fluid dynamics simulations. In the scope of this thesis, the shape optimization tasks, which require both the computation of the state variables and the shape sensitivities, are performed making use of Carat++. The existing implementations in Carat++ contains the advantageous features mentioned in Sec. 3.1.1. Furthermore, as mentioned in Sec. 3.1.2, the computation of the state variables as well as the shape sensitivities that are related to the fluid dynamics are achieved making use of the OpenFOAM® with the developments from HELYX®. It should be once more mentioned that the scope of this work is not the development of the coupled sensitivity analysis but the set-up of a workflow that is flexibly bringing the available software together and finally tying the CAD and the numerical models. Thus, the coupled sensitivity analysis is not considered in the developments.

Having the simulation software defined, one needs to find a suitable workflow for the MDO tasks that bring the valuable contributions of the selected software together in a flexible way, such that the components can be replaced without a significant effort. [4] discusses many architectures and communication patterns for MDO tasks. The workflow to be established needs to satisfy the requirements to perform an FSI analysis -or in more general sense MDA-, sensitivity analysis on individual disciplines, apply sensitivity filtering with the Vertex Morphing and/or Mortar Mapping method and generate the respective CAD representation of the optimized shapes. For this purpose *Multi-Disciplinary Feasible* (MDF) architecture offers a good starting point. However, the variants of the MDF defined in [4] do not fully satisfy the needs of the optimization task defined in the scope of this work. Thus, a new architecture is established. The designed architecture computes the strongly coupled response of the individual disciplines. The significance of the design stems from the combination of the shape sensitivities of individual disciplines. Typically, the design of structures in FSI environment contain portions on both the coupling interface and other structural features such as stiffeners, spars etc. Thus, the design variables are distributed among all the portions of the structural domain. While the design of the coupling interface is affected by both structural and fluid sensitivities, the rest of the design is only affected by the structural sensitivity fields. In order to fuse them together, the under-

lying CAD parametrization is chosen as a basis for the optimization. The shape sensitivities are computed for each individual discipline giving design updates, then they are mapped onto the control points of the NURBS description of the CAD geometry. This enables updating the design of both disciplines and driving the optimization on the CAD parameters. As an outcome of the considerations and choices made in this section, the resulting optimization workflow is presented in Fig. 3.1.

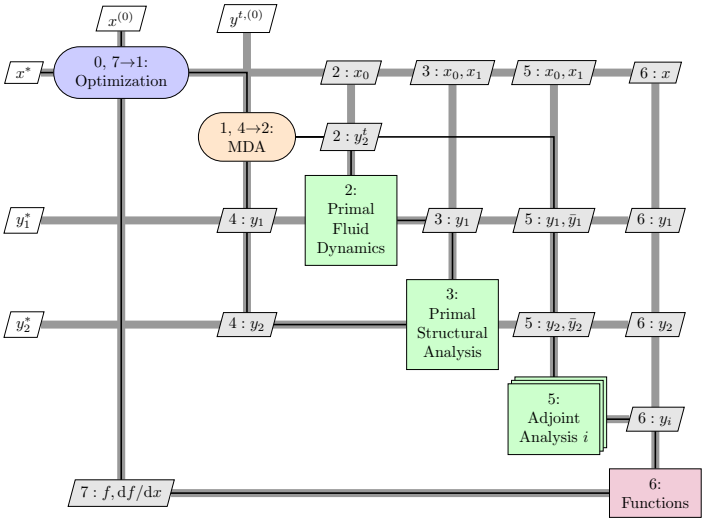


Figure 3.1: The Extended Design Structure Matrix (XDSM) [41] of the implemented MDO workflow.

3.3.2 Integration of the CAD into the MDO Framework

Integration of CAD into the optimization workflows is a crucial issue when a product at the end of an optimization analysis is sought. Moreover, even if sensitivity filtering and/or appropriate mesh deformation techniques are used, the resulting numerical model might suffer from inappropriate or insufficient discretizations for the simulations to be performed. In that case a regeneration of the discrete numerical model would be nec-

essary. Moreover, the optimized numerical models should be converted into appropriate CAD formats to make them applicable in the production processes. This rises the need for an updated CAD model of the resulting discrete model. In practice, one needs to work with commercial CAD formats, such as IGES, STEP, etc. , which can be exported from or imported into CAD programs. Such commercial formats use a variety of complexity in the shape definitions, such as circle, sphere, NURBS, B-splines, etc. , in order to both accurately describe the designed shapes and reduce the data size to be exported. One might need to either increase the flexibility of the design by modifying the description of the primitive shapes to a higher complexity so that the resulting optimized shape has a more sophisticated shape than the original or reduce the flexibility of the resulting shape by restricting the type of deformations it undergoes throughout the simulation since it might be necessary to maintain some design features. In addition to this, even if the given CAD data supports the necessary requirements, it would still be necessary to introduce it to the optimization workflow such that the connection between the discrete numerical model and the underlying CAD data is established. In this section, the integration of the CAD data into the optimization workflow is presented.

Processing commercial CAD formats and generating computational models require a robust CAD Kernel. OpenCASCADE offers a great opportunity for this purpose and its community edition OCE is an open-source software library that supports commercial CAD formats. In addition to this, pythonOCC offers the possibility to incorporate OpenCASCADE into other software environments in a flexible way thanks to its python interface. It makes code handling and visualization straightforward. Thanks to the interactive user interface of pythonOCC, this combination allows working with commercial formats, preprocessing generic CAD geometries for optimization tasks e.g. defining constraints, and visualizing the results as well as exporting optimized CAD models.

The implemented module utilizes the capabilities of OpenCASCADE to convert arbitrary definitions of the CAD models to trimmed multi-patch NURBS or B-Spline descriptions since it is the followed choice of implementation for the CAD-based Mortar method that is used for the generation of the transformation operators. After the models are converted to NURBS or B-Spline based descriptions, they are ready to be fed into the desired workflow. At this stage the implementation offers two possibilites:

1. The CAD model can be fed into the EMPIRE workflow which manages the connections between the different clients and also establishes the mapping operators internally.

2. The mapping operators can be generated within the layer that utilizes the pythonOCC functionalities thanks to the external mapping library which is generated during the compilation of EMPIRE.

This allows a flexible usage of the implemented software client and facilitates the incorporation of the CAD models into the optimization workflow by both a graphical user interface (GUI) and a software interface that automatically connects to the workflow. As a result, generation of CAD models as a by-product of the optimization process is enabled.

3.3.3 CAD Module

The GUI is implemented in such a way that a variety of CAD formats can be handled. The functionalities of the implemented GUI named *CAD Module* are listed below:

- **Import/Export**: The import and export features facilitate the use of commercial CAD formats. The current state of the implementation supports the import and the export operations of IGES and STEP formats which are widely supported by the CAD programs. In addition, the geometries given in the newly emerging JSON format can also be imported using this feature, see [82] for further reading on the JSON format.

- **Shape Operations**:

  - **Sew**: The input file can consist of separate shapes that are to be put together. This command brings separate shapes into a compound while making it possible to find out the connections between each shape.
  - **Convert Shapes to NURBS**: The CAD makes use of various shape definitions. The implementation, which is available in EMPIRE for the generation of the mapping operators, is based on the B-Spline and NURBS definitions. Thus, the shapes are converted to NURBS descriptions using this command.

  - **Detect Free Edges**: In various shape optimization tasks, parts of the geometry's edges are constrained due to different reasons, e.g. production or assembly considerations. This command creates a topology map of the sawn geometry and finds out the free edges of the geometry.
  - **Detect Shared Edges**: As mentioned earlier in Sec. 3.2.3, the mapping operations for the generation of the CAD models do not necessarily produce matching deformation fields at the patch boundaries. In order to establish the patch continuity conditions, the shared edges are detected using this command making use of a topology map of the sawn geometry.
  - **Refinement**: In some cases, the underlying CAD descriptions consist of shapes that restrict the free form shape changes. In such a scenario, a refinement step can be applied by modifying the polynomial degrees or the knot vectors of the NURBS geometries.
  - **Apply Results**: The finalized optimization results can be applied to the underlying CAD description and visualized within the implemented GUI.

- **EMPIRE**: The commands under this sublisting consist of the functionalities that are necessary for the integration of the CAD into the optimization workflow.

  - **Connect**: The connection to the prepared EMPIRE workflow is established.
  - **Send Shapes**: The preprocessed shapes are sent over the MPI communication functionalities of EMPIRE and consecutively the respective data structure is constructed.
  - **Add Dirichlet Conditions**: The applied Dirichlet conditions are prepared by finding out the parent patches and made ready to be sent to the EMPIRE data structure.
  - **Add Continuity Conditions**: The applied patch continuity conditions are prepared by finding out the parent patch couples and made ready to be sent to the EMPIRE data structure.
  - **Send Dirichlet Conditions**: The prepared list of Dirichlet conditions are communicated to EMPIRE.

  - **Send Continuity Conditions**: The prepared list of patch continuity conditions are communicated to EMPIRE.
  - **Disconnect**: After the data structure within EMPIRE is set, the client can be disconnected from EMPIRE if the further communication is unnecessary.

- **Mapping**: As an alternative to adding the client to the optimization workflow and simultaneously generating CAD models, one can generate the mapping operators between the IGA and FE discretizations of the surfaces in order to regenerate the CAD models of the optimized discrete numerical models. This is achieved by the prepared external mapping library that is generated during the compile time of EMPIRE.
  - **Set FE-Discretization**: Imports an FE discretization of a surface in a GiD format and generates a mesh within the data structure of the mapping library.
  - **Set IGA-Discretization**: After the preprocessing steps are completed, the resulting NURBS shape is made ready for use within the mapping library.
  - **Define Dirichlet Conditions**: The applied Dirichlet conditions of the optimization problem are set using this command.
  - **Define Coupling Conditions**: The applied coupling conditions are set in the mapping library data structure.
  - **Generate FEM-FEM Mapper**: The Mortar Mapping operators between two previously defined FE discretizations are generated.
  - **Generate IGA-FEM Mapper**: The Mortar Mapping operators between an IGA and an FEM discretizations of the same surface is generated using this command.
  - **Consistent Mapping**: After the mapping operators are created, the transfer of the fields between dissimilar discretizations can be achieved.
  - **Conservative Mapping**: In addition to the consistent mapping technique, the conservative mapping method can also be used for the fields that are essentially of integrated types, e.g. force.

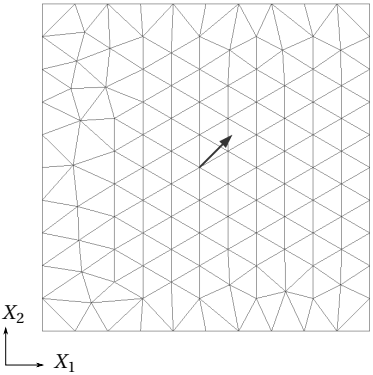
3.4 Results and Discussion

This section brings the discussed simulation and optimization methods throughout this thesis. The demonstration cases are ordered in an increasing level of complexity for the comprehensive presentation of the applied methods. Firstly, the application of the sensitivity filtering using the Mortar Mapping method is presented. Sec. 3.4.1 demonstrates this method in a very simple manner using a FEM-FEM Mortar Mapping method. Another demonstration case using the Mortar Mapping method but this time with the CAD-integrated optimization workflow is accomplished on a strain energy minimization of a 90° bent pipe structure. This example also discusses the use of the combined Vertex Morphing and Mortar Mapping operators. Finally, the shape optimization of the CX-100 wind turbine blade in FSI with the implemented CAD-integrated optimization workflow is presented. This example brings the discussed methodologies together while highlighting the importance of the CAD integration into the optimization workflow.

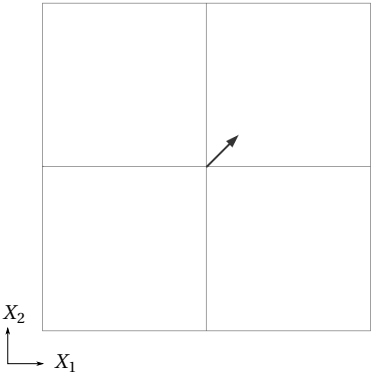
3.4.1 Sensitivity Filtering on a Planar Domain

For a simple demonstration of the Mortar based filtering method, a planar domain can be adopted following the example in [16]. A single artificial sensitivity vector $[0.1, 0.1, 0.0]^T$ in the middle of the planar domain is used for the presentation of the method when a steepest descent algorithm is employed. The discretization of the design field is achieved with linear triangular elements whereas the control field is discretized with bilinear quadrilaterals. The coarse quadrilaterals enable the filtering property. The discretizations of the fields and the sensitivity field are shown in Fig. 3.2(a) and Fig. 3.2(b).

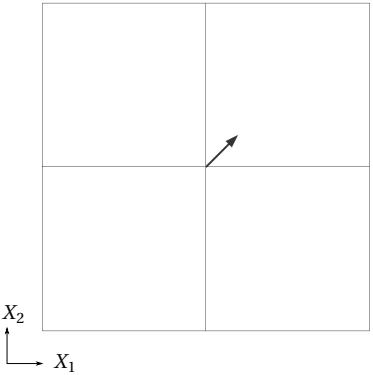
Since the shape functions that discretize the control field extend over a number of design elements, the computed design update field on the control field affects a wide range of design elements. As a result, sensitivity and design filtering are achieved by increasing the continuities of these fields across the design element edges to the control field's shape functions' continuities. In other words, the sensitivity or the design update field defined on the C^z continuous design space is projected onto the C^s continuous control space. Eventually, possible distortions of the design mesh are also avoided.



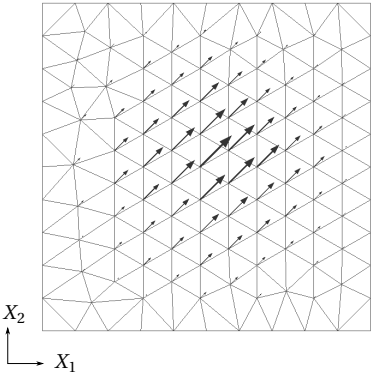
(a) A hypothetical sensitivity field defined on the design field.



(b) The corresponding sensitivity field on the control field.



(c) The design update on the control field.



(d) The filtered design update on the design field.

Figure 3.2: The sensitivity filtering operations with Mortar Mapping method and the resulting design update field on a planar domain. Adapted by permission from Springer Nature Customer Service Centre GmbH: Springer Notes on Numerical Fluid Mechanics and Multidisciplinary Design [16], (2018).

3.4.2 CAD-Integrated Optimization of a 90° Bent Pipe

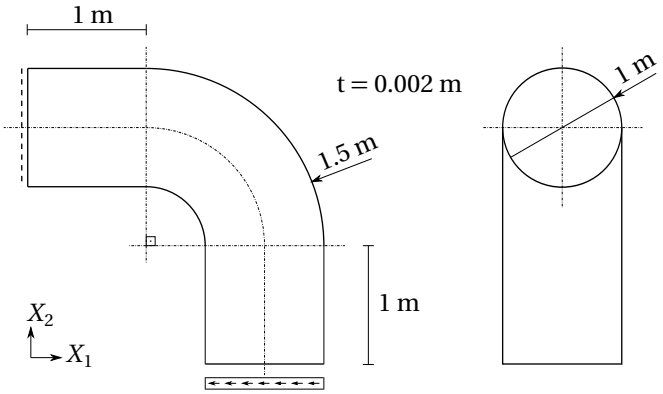


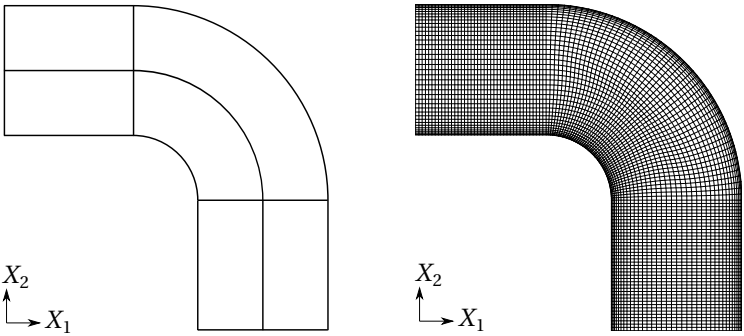
Figure 3.3: Dimensions of the 90° bent pipe optimization case.

Investigations on the filtering properties of the proposed methods and the combination of mentioned filtering methods in Sec. 3.2 are presented on a model optimization problem. The optimization case is the strain energy minimization of a tip loaded 90° bent pipe structure. The dimensions of the case are presented in Fig. 3.3. The initial geometry is modelled with 12 untrimmed patches using bi-quadratic B-Spline shape functions and an element per parametric direction, see Fig. 3.4(a). The finite element discretization is achieved using bi-linear quadrilateral shell elements that are based on the Reissner-Mindlin kinematics. The structure is simply supported at one of its free edges and loaded on the other as depicted in Fig. 3.3. The loading is distributed equally at the 80 discrete nodes of the free edge as 1 N. The shape sensitivities are computed using Carat++ and the optimization is achieved making use of the well-known steepest descent method with a step size factor defined as follows:

$$\alpha = \hat{\alpha} \left\| \nabla_{\hat{\mathbf{z}}} J \right\|_{\infty}^{-1} \quad \text{with} \quad \hat{\alpha} = 0.01. \tag{3.51}$$

The CAD modeling as well as the finite element discretization of the considered structure are depicted in Fig. 3.4.

Firstly, the filter radius' effect for the Vertex Morphing method is investigated. The filter function is chosen to be a hat function and the opti-



(a) CAD geometry. (b) FEM discretization.

Figure 3.4: The design and the discretization of the 90° bent pipe problem.

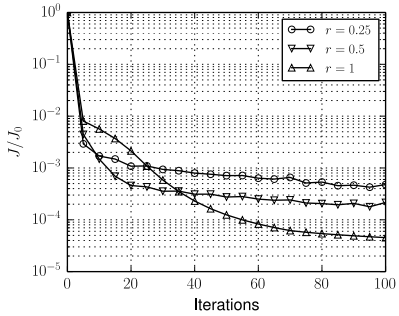
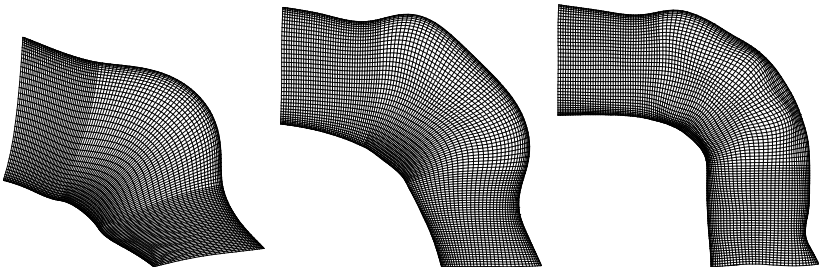


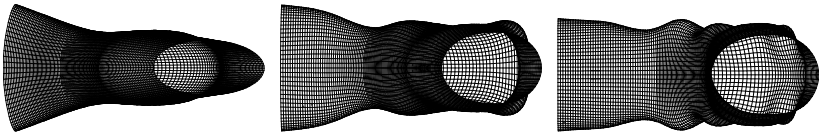
Figure 3.5: The filter radius study for the Vertex Morphing method.

mization problem is run for three different filter radii; $r = 0.25$, $r = 0.5$, $r = 1.0$. The objective evolutions throughout the optimization iterations are plotted in Fig. 3.5. It can be observed that the filter radius choice effects the found local minimum, since the search direction is modified by the filtering operation as discussed in [8]. As a result, more detailed shape features can be obtained by reducing the filter radius as can be observed in Fig. 3.6-3.8.



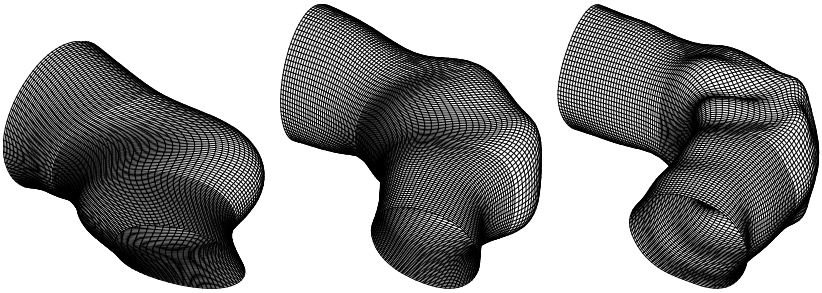
(a) $r = 1\ m$. (b) $r = 0.5\ m$. (c) $r = 0.25\ m$.

Figure 3.6: Optimized shapes using Vertex Morphing method in XY-plane view for varying filter radii after 100 optimization steps.



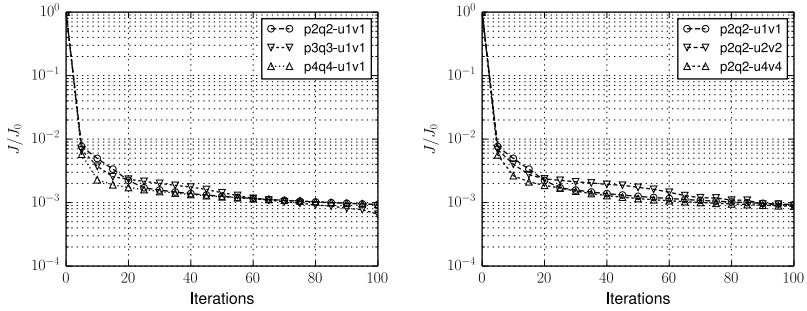
(a) $r = 1\ m$. (b) $r = 0.5\ m$. (c) $r = 0.25\ m$.

Figure 3.7: Optimized shapes using Vertex Morphing method in XZ-plane view for varying filter radii after 100 optimization steps.



(a) $r = 1\ m$. (b) $r = 0.5\ m$. (c) $r = 0.25\ m$.

Figure 3.8: Optimized shapes using Vertex Morphing method in perspective view for varying filter radii after 100 optimization steps.



(a) The p-refinement study.

(b) The h-refinement study.

Figure 3.9: The refinement study for the CAD-based Mortar Method.

As a second comparison case, the sensitivity filtering is achieved making use of the CAD-based Mortar Mapping technique. An interesting aspect of the CAD-based Mortar Mapping technique is the polynomial order choice for the NURBS basis functions as well as the element sizes that are related to the knot spans. In order to reveal the effects of different choices, p-refinement and h-refinement studies are performed. For the elimination of the effects related to shape function spans due to the knot locations and multiplicities and since the shape functions propagate across p+1 knots, one element per patch parametric space is chosen during the p-refinement study. During the p-refinement study, uniform degree elevations in both parametric directions are applied on each patch. Patch coupling constraints are applied for all the cases with penalty factors for displacement $\alpha^{\mathbf{d}}_{\gamma_{qr}} = 1e4$ and for rotation $\alpha^{\boldsymbol{\omega}}_{\gamma_{qr}} = 1e8$. By doing so, the Mortar transformation matrices are enhanced with the patch coupling contributions through the penalty formulation. The objective evolution of the optimization problem for the p-refinement and the h-refinement studies are presented in Fig. 3.9. The resulting shapes of the p-refinement and h-refinement studies are presented in Fig. 3.12-3.14 and Fig. 3.15-3.17 respectively. It can well be observed that more local features can be obtained as the NURBS geometry is refined. This effect can be related to the filter function as well as filter radius choice of the Vertex Morphing method. In

contrast to the distinct control over the filter function and radius choice of the Vertex Morphing method, the behaviour of the p- and h-refinements for the generation of the CAD-based Mortar operators are rather vague and a direct relation can not be established. However, it can be noted that the span of the NURBS basis functions can be controlled via their polynomial degrees as well as knot vectors. The knot multiplicities can be chosen such that the span of the basis functions are restricted to certain lengths. In the h-refinement study, it is favoured that the shape function spans are not strictly constrained and at least C^1 is maintained across the knot parametric locations. In addition, it should be noted that the C^{p-m} as well as G^{p-m} continuity are maintained across the knot parametric positions, where p and m denote the polynomial degree and the knot multiplicity respectively. The continuities across the patch interfaces are restricted to the imposed condition, which in the current implementation correspond to the displacement (G^0) and/or the rotation (G^1) continuity.

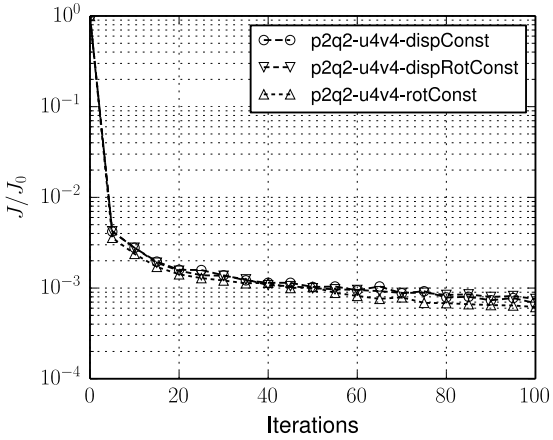


Figure 3.10: The relative objective evolutions for the constrained optimization.

Furthermore, the effect of the embedded constraint handling for the case of CAD-based Mortar Mapping method is investigated. As described in Sec. 3.2.3, the geometric equality constraints such as displacement or rotational restrictions on defined portions of the geometry can be embedded

into the filtering operator. This allows the simultaneous handling of the constraints while transferring the optimization problem to a control field based on the control points of the NURBS geometry and filtering the design updates. The constraints on displacements, rotations and both displacements and rotations are applied on the same optimization problem with same NURBS discretizations for the CAD-based Mortar Mapping method and the objective evolutions are presented in Fig. 3.10. In addition, the resulting shapes are presented in Fig. 3.18-3.20. The application of the constraints do not only affect the generated features at the constrained edges but also propagate into the rest of the domain as far as the influenced basis functions extend.

A glance at the figures of the optimization results reveals two important aspects of the method. The advantageous aspect of the method is the generated smooth CAD geometries, see Fig. 3.14, Fig. 3.17 and Fig. 3.20. Through the choice of the refinement level for the NURBS surface description one can control the resulting curvature as well as the locality of the generated features and the resulting shape effectively represents a CAD model. On the other hand, it can be observed that the mesh quality is not necessarily maintained during the optimization iterations since the implemented mapping operators do not restrict the shape updates to be in surface normal directions. Despite the fact that the method offers the possibility to regenerate the numerical model by remeshing, since the updated CAD model is available at any point of the optimization procedure, it is still favourable to maintain a certain mesh quality in order to avoid remeshing which would necessitate a check on the mesh quality and the reconstruction of the mapping operators.

As discussed in Sec. 3.2.3, one can adopt the advantageous sides of both Vertex Morphing and the CAD-based Mortar Mapping methods. While the Vertex Morphing method has few parameters to adjust, such as the filter function and its effective radius, and is able to maintain a decent mesh quality as observed in Fig. 3.6-3.8, CAD-based Mortar Mapping method generates the CAD models at any given point of the optimization procedure. As revealed in Table 3.1 the use of both methods apply a combined filtering effect when the filtering requirements are met for each of the methods:

1. The filter radius for the Vertex Morphing method should be large enough such that a reasonable portion of the domain is included in the integral of Eqn. (3.13).
2. The employed shape functions for the discretization of the Mortar Mapping method's master surface should span across several elements such that the contributions from those portions of the domain are collected in the integral of Eqn. (3.29).

From above conditions, one can conclude that the sensitivity filtering can be achieved by the Vertex Morphing method by choosing a suitable filter function and radius while the Mortar method's filtering effect can be cancelled out by choosing a relatively fine discretization for the master surface definition and, in this particular case where NURBS shape functions are employed, the optimization problem can still be driven on the CAD-based parameters, i.e. the control points. The advantages of this approach are threefold:

1. The CAD representation of the discrete numerical model is always available and can be facilitated for remeshing purposes if needed.
2. The discrete numerical model is a product of the CAD model at any optimization iteration since the forward mapping of the shape updates from the CAD model to the discrete numerical model ensures the equality of the shape update fields.
3. Compared to the a posteriori CAD reconstruction techniques additional refinement of the CAD description is not necessary.

In order to present the combined usage of both methods, the optimization of the presented problem is achieved for two different cases where the Vertex Morphing method, using a hat filter function with filter radius $r = 0.25$ and $r = 0.5$ for the respective cases, and the CAD-based Mortar Mapping method, utilizing a bi-quintic base functions and 32 elements in each respective parametric direction, are combined. The objective evolution is presented in Fig. 3.11 while the generated shapes are depicted in Fig. 3.21-3.23. It can clearly be observed that the objective evolutions and the generated shapes of the combined usage follows the results of the pure Vertex Morphing method. This verifies the introduced method and the conditions that are defined in this section for the combined usage.

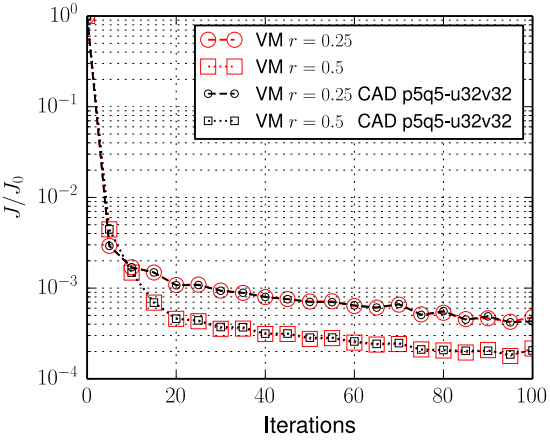
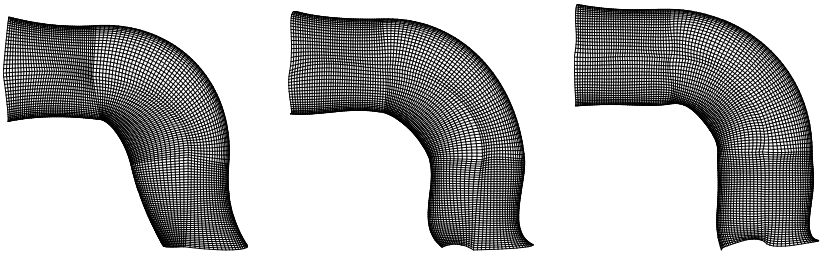
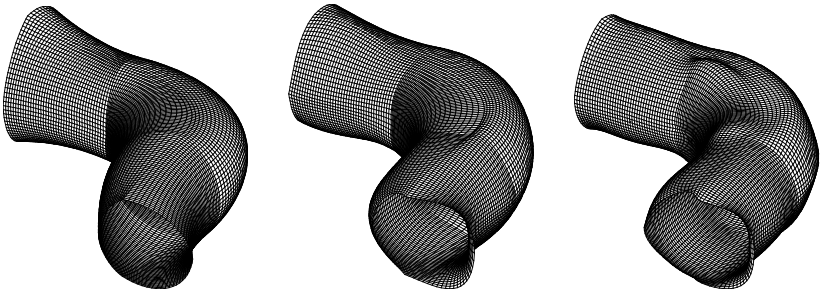


Figure 3.11: The comparison of objective evolutions for pure Vertex Morphing method and the combination of Vertex Morphing and CAD-based Mortar Mapping methods.



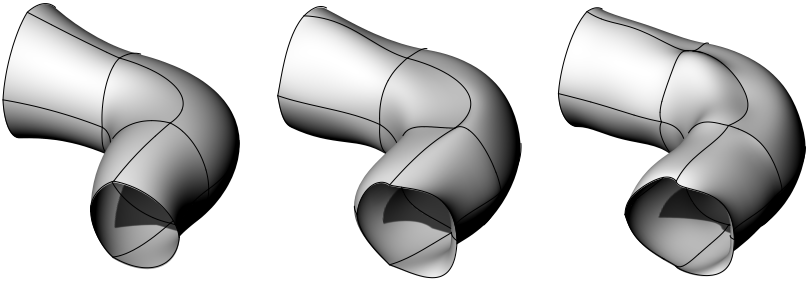
(a) $p=2, q=2, u=1, v=1$. (b) $p=3, q=3, u=1, v=1$. (c) $p=4, q=4, u=1, v=1$.

Figure 3.12: Optimized shapes using CAD-based Mortar Mapping method in XY-plane view for the p-refinement study.



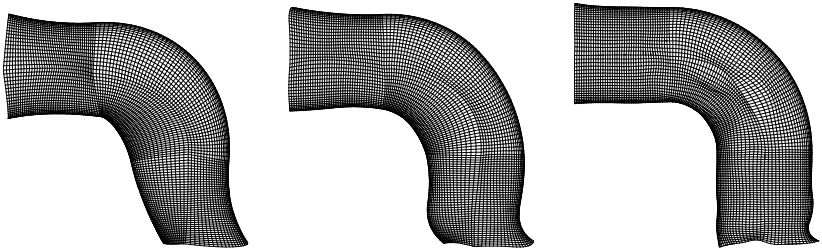
(a) $p=2, q=2, u=1, v=1$. (b) $p=3, q=3, u=1, v=1$. (c) $p=4, q=4, u=1, v=1$.

Figure 3.13: Optimized shapes using CAD-based Mortar Mapping method in perspective view for the p-refinement study.



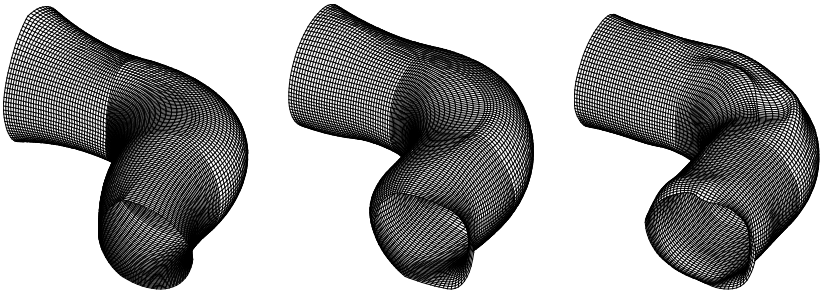
(a) $p=2, q=2, u=1, v=1$. (b) $p=3, q=3, u=1, v=1$. (c) $p=4, q=4, u=1, v=1$.

Figure 3.14: Optimized CAD geometries using CAD-based Mortar Mapping method in perspective view for the p-refinement study.



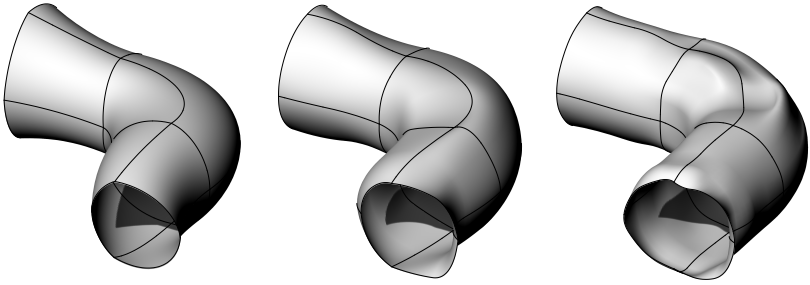
(a) $p = 2$, $q = 2$, $u = 1$, $v = 1$. (b) $p = 2$, $q = 2$, $u = 2$, $v = 2$. (c) $p = 2$, $q = 2$, $u = 4$, $v = 4$.

Figure 3.15: Optimized shapes using CAD-based Mortar Mapping method in XY-plane view for the h-refinement study.



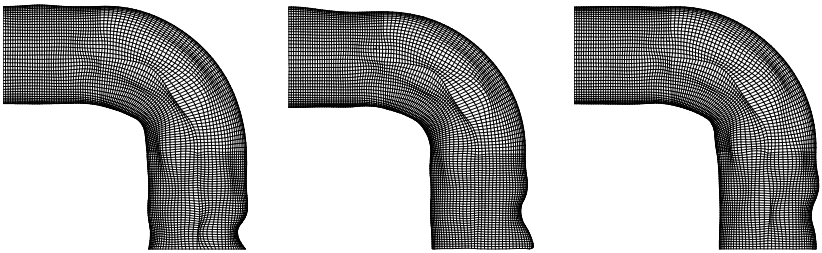
(a) $p = 2$, $q = 2$, $u = 1$, $v = 1$. (b) $p = 2$, $q = 2$, $u = 2$, $v = 2$. (c) $p = 2$, $q = 2$, $u = 4$, $v = 4$.

Figure 3.16: Optimized shapes using CAD-based Mortar Mapping method in perspective view for the h-refinement study.



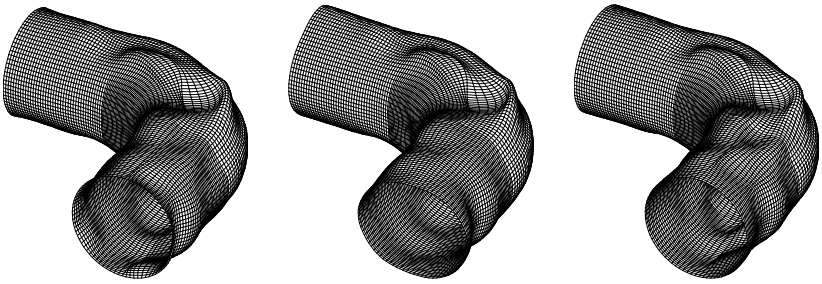
(a) $p = 2$, $q = 2$, $u = 1$, $v = 1$. (b) $p = 2$, $q = 2$, $u = 2$, $v = 2$. (c) $p = 2$, $q = 2$, $u = 4$, $v = 4$.

Figure 3.17: Optimized CAD geometries using CAD-based Mortar Mapping method in perspective view for the h-refinement study.



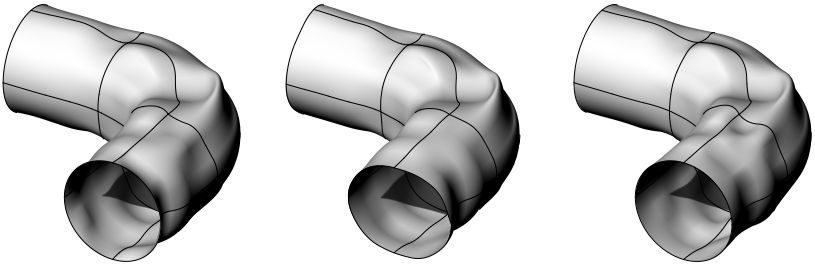
(a) Displacement constrained. (b) Rotation constrained. (c) Displacement and rotation constrained.

Figure 3.18: Optimized shapes using CAD-based Mortar Mapping method in XY-plane view with geometric constraints at free edges.



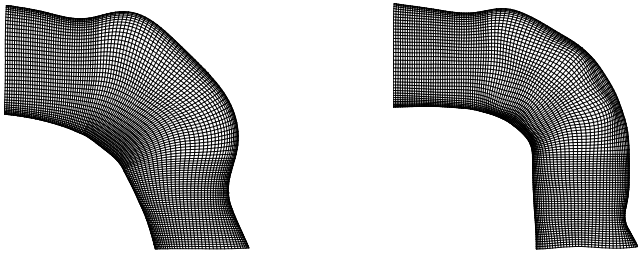
(a) Displacement constrained. (b) Rotation constrained. (c) Displacement and rotation constrained.

Figure 3.19: Optimized shapes using CAD-based Mortar Mapping method with geometric constraints at free edges.



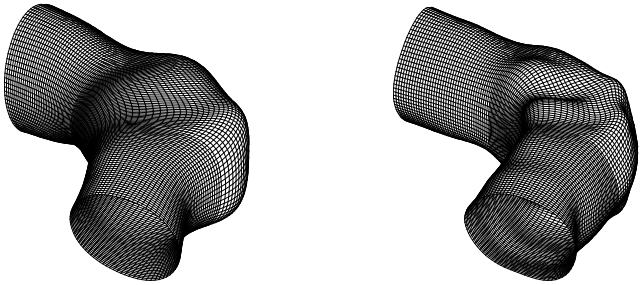
(a) $p = 2$, $q = 2$, $u = 4$, $v = 4$, displacement contstrained. (b) $p = 2$, $q = 2$, $u = 4$, $v = 4$, rotation contstrained. (c) $p = 2$, $q = 2$, $u = 4$, $v = 4$, displacement and rotation constrained.

Figure 3.20: Optimized CAD geometries using CAD-based Mortar Mapping method with geometric constraints at free edges.



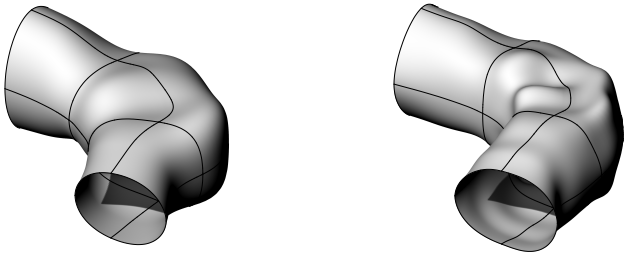
(a) $r = 0.5$. (b) $r = 0.25$.

Figure 3.21: Optimized shapes using the combination of the Vertex Morphing and the CAD-based Mortar Mapping methods in XY-plane view.



(a) $r = 0.5$. (b) $r = 0.25$.

Figure 3.22: Optimized shapes using the combination of the Vertex Morphing and the CAD-based Mortar Mapping methods.



(a) $r = 0.5$, $p = 5$, $q = 5$, $u = 32$, $v = 32$.

(b) $r = 0.25$, $p = 5$, $q = 5$, $u = 32$, $v = 32$.

Figure 3.23: Optimized CAD geometries using the combination of the Vertex Morphing and the CAD-based Mortar Mapping methods.

3.4.3 CAD-Integrated MDO of CX-100 Wind Turbine Blade

The FSI analysis of the CX-100 wind turbine blade and the employed methods are presented in Sec. 2.5.4. In this section, the components of the methods related to the optimization that are presented in this thesis are brought together in an industrial scale optimization case.

The optimization problem considered in this section is the minimization of the strain energy J_{SE} and maximization of the torque J_{T} acting on the rotor. The constructed objective function is a weighted sum of mentioned objectives. The objective function can be expressed as follows:

$$J = \alpha_{\mathrm{SE}} J_{\mathrm{SE}} - \alpha_{\mathrm{T}} J_{\mathrm{T}} \,. \tag{3.52}$$

It should be noted that the torque maximization and the strain energy minimization objectives are contradicting goals since increasing torque alone results in increasing displacements and increases the strain energy. Thus, a careful selection of the weighting parameters are necessary to achieve improvements in both of the objectives. Herein, out of a series of simulations, the results for the following weighting parameters are presented; $\alpha_{\mathrm{SE}} = 0.99$, $\alpha_{\mathrm{T}} = 0.01$.

The disciplinary sensitivity analysis is carried out at the FSI equilibrium state with the mentioned methods in Sec. 3.1. Consequently, the sensitivity filtering is achieved making use of the Vertex Morphing method using a Gaussian filter function with a radius of 0.12 m on the respective discretizations of the sub-disciplines. It is noteworthy to mention that the optimization variables are not common for the sub-disciplines and the objective functions. The structural design variables consist of the blade wet surface and the internal spar, whereas the fluid design variables contain only the blade wet surface. Moreover, even though the blade wet skin is defined as an optimization surface for both of the sub-disciplines and the objective functions, their discretizations for the underlying numerical schemes are dissimilar. In such an optimization scenario, the necessity of utilizing the CAD model is prominent, since it acts as a superset of all the optimization variables. The CAD model behaves as a master surface that drives the full set of optimization variables. The maximum step size for the shape updates is chosen as 1 mm by scaling the shape updates with the infinity norm of the shape update vector field. Finally, the optimization workflow is achieved as illustrated in Fig. 3.1. The optimization problem

is run for 11 iterations until the fluid domain mesh deformation does not allow accurate results.

The objective evolutions of the combined and the individual objectives are presented in Fig. 3.24. It could be observed that with the correct selection of the weighting parameters, both of the objectives can be improved. In addition, to that Fig. 3.26 depicts the changes in the blade profiles at the selected locations, see also Fig. 3.25 for the selected locations. The torque is increased by both adjusting the angle of attack at the sections *B*, *C* and *D*, and by increasing the area at the section *E*, while the strain energy is mainly minimized by stiffening the blade root around section *A*. A consequence of employing the underlying CAD model parameters as the optimization variables can be observed in this case. The spars are modelled by making use of linear NURBS basis functions in the blade thickness direction. This allows only shape updates that keep the spar straight or in other words filters out the curved spar designs. By doing so, the blade skin can evolve freely throughout the optimization iterations but the spar is kept straight as a design decision and a design feature to be maintained as mentioned in Sec. 3.3.2.

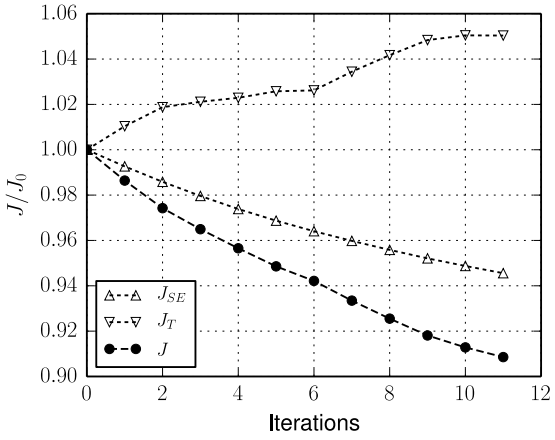


Figure 3.24: The relative objective evolutions.

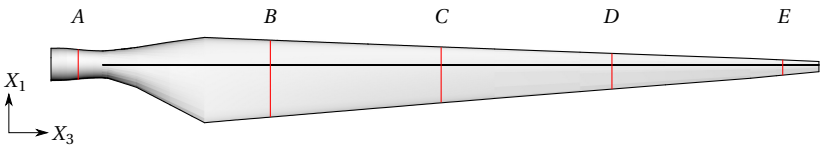
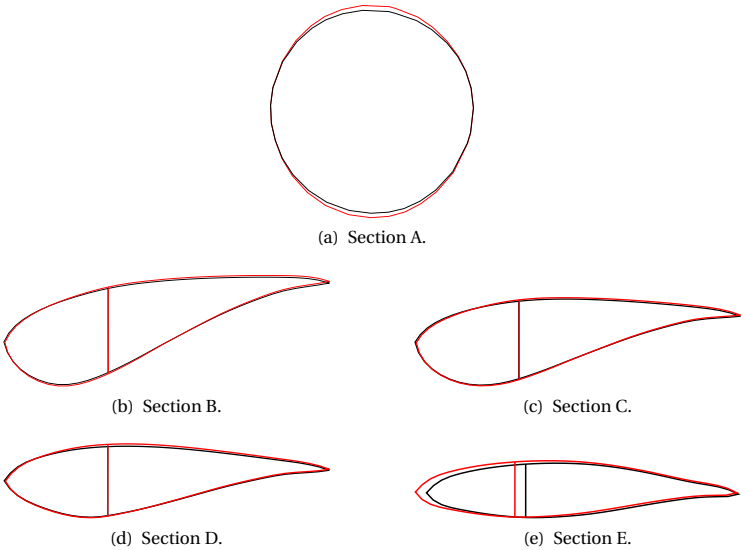


Figure 3.25: The section (red) positions for the comparison of the initial and the optimized shapes at optimization iteration 11.



(a) Section A.

(b) Section B.

(c) Section C.

(d) Section D.

(e) Section E.

Figure 3.26: The comparison of the optimized geometry sections (red) against the initial (black) configuration.

Chapter 4

Conclusions and Outlook

4.1 Conclusions

In this thesis, high fidelity simulation methods for structural and fluid mechanics are discussed, and consecutively, a CAD-integrated MDO workflow for FSI problems is realized. Regarding the high fidelity structural analysis methods, two updated dynamic analysis methods are introduced. Foremost, the geometrical nonlinearities and the stress stiffening effects on the dynamic behaviour of the large scale structures due to the constant loading are investigated. It is shown on a varying complexity of demonstration cases that these effects can be included in the dynamic phase even if a linear transient analysis method is adopted. The applicability of the introduced dynamic analysis methods to large scale cases is verified by means of measurement data that is collected from a hybrid (steel-concrete) wind turbine tower. Furthermore, robust coupling methods for steady state FSI analyses are discussed, which is a prerequisite of MDO problems that incorporate FSI as MDA. An implicit coupling method based on a mixture of analytic derivatives and least-squares approximations is proposed. The performance of the method is compared against the conventional methods such as Gauß-Seidel and also other popular least-squares based interface quasi Newton methods such as IBQNLS, IQNILS and MVQN. Given that the structural solver is available to the user, it can be concluded that the introduced method improves the convergence properties when compared to the considered methods within this work. The improved performance stems from the analytical derivative inclusion in the tangent stiffness of the FSI problem, when the problem is seen from the structural point of view. The applicability of the method for the optimization iterations is also investigated on a practical example. In addition, the sensitivity filtering properties of the Mortar method are revealed. The use of the Mortar method is extended to the IGA field and a sensitivity filtering method is established, which relies on the underlying CAD model of the numerical models. It is shown that the introduced sensitivity filtering method can be extended to encompass the geometric equality constraints of the optimization problem. The sensitivity filtering properties are demonstrated for various discretization techniques. Last but not least, a CAD-integrated MDO workflow is established. The realized workflow is applied on the MDO of the CX-100 wind turbine blade. It can be once more emphasized that the application of the Mortar operator in combination with IGA brings benefits in MDO workflows by acting as a master surface for transferring

the design variables to the NURBS or B-Spline control points of the CAD model. In particular, when a partitioned scheme is employed for the solution of the FSI problem, the nonconforming nature of the shared design surface discretizations and the discrete sensitivity fields can be overcome by the introduced method. In addition, it is also shown on an industrial scale showcase that by making use of the properties of the CAD model, one can perform free-form shape optimization on part of the design surface, while the remaining can be forced to follow a predefined pattern as a design decision.

4.2 Outlook

Regarding the robust dynamic analysis methods, one can additionally adopt modal transient analysis and accordingly the directional derivatives of the system matrices with respect to the chosen prominent eigenmodes for a more accurate determination of the transient response. A similar approach is adopted in [83]. This type of approach can also be utilized for the construction of reduced-order models (ROM) by reducing the system to the modal DOFs paving the way for the digital twin concept.

In addition, in the context of FSI, one can also employ adjoint methodologies for the computation of the exact tangent contributions that are related to the loading conditions. For instance, [84] employs a similar approach and suggests adopting an optimization approach for the solution of the FSI problem by defining an appropriate objective function that reduces the interface residual. Moreover, [85], [86] point out that the adjoint methods can be employed for the computation of the Hessian. Given that the function to be differentiated is of a potential type, the Hessian gives essentially the stiffness matrix related to the defined potential. In [85], [86], the excessive necessary computational effort is also mentioned. Still, this method can be employed and used as a reference case for performance comparisons for newly developed coupling algorithms when dealing with practical FSI cases. In the frame of this work, the input and output modes for the computation of the approximate Jacobians are computed on matching discretizations or by mapping the input and output modes onto the structural discretization. The effect of the mapping operation can also be investigated and a general mapping rule for the matrices should be defined. This is addressed subsequently in this section.

Moreover, the properties of the least-squares methods can be employed in a further extent in MDO. There exist several MDO architectures which incorporate reduced-order models [4]. It is already shown within this work that use of iteration history can improve the accuracy of the approximated Jacobians and reduce the computational effort. Thus, as the optimization iterations advance, the accuracy of the approximations improve. As already stated in [46], one can employ further methods such as QR-decomposition or singular value decomposition for the procedures necessary for the approximation of system Jacobians. These methods should be investigated

for their computational efficiency, accuracy and potential use not only in FSI but also in the MDO architectures.

The incorporation of the coupled sensitivity analysis in this frame is of high importance since the variation of the state variables are co-dependent. Thus, the employed MDO methodology can be extended to integrate the coupled sensitivity analysis, too.

Furthermore, the sensitivity filtering properties of the Mortar method should be investigated for optimal discretization of the master surface or the CAD description in this particular case. In addition, the Mortar method can be employed for reconstructing the optimized shape's discrete numerical models. The optimal discretization necessary for describing the discrete models should also be pointed out. For instance, one can adopt the hierarchical methods for this purpose as in [87], [88]. Moreover, in the context of a posteriori CAD reconstruction techniques, the collocation methods can also be beneficial due to the simplicity in the implementation aspects. In the frame of this work, the NURBS and the B-Spline shape functions are considered for the description of the CAD geometries. In practice one might prefer keeping the primitive shapes such as circle, arc, etc. since these could be the part of the design choice. Then the transformation operators should be constructed in such a way that the design handles still remain as the defining variables of these primitive shapes.

Remarks On the Computation of the Fluid Load Tangents On Dissimilar Discretizations

In practice, when partitioned solution procedures are employed, the structural and the fluid discretizations of the FSI interface typically do not match due to the efficiency reasons, different requirements on the mesh quality etc. This results in a need for a transformation operation between the dissimilar discretizations. The transformations can either be applied on the coupling variables and the fluid load tangent can be computed using the mapped quantities or the computed tangent stiffness matrices can be mapped using the transformation rules for the coupling variables. In case the transformation should be applied on the fluid load tangent that is computed on the fluid discretization of the interface, the mentioned operation can be performed using various methods as discussed in [39], [76] and many others. For instance the Mortar Mapping method can be

employed for the construction of the transformation operators, see also Sec. 3.2.2. Given a constant transformation operator $\mathbf{T}$ for the motion transfer between the dissimilar discretizations of the same interaction surface, one can write the following:

$$\hat{\mathbf{d}}_{\Gamma_\mathrm{f}} = \mathbf{T}\,\hat{\mathbf{d}}_{\Gamma_\mathrm{s}} \,, \tag{4.1a}$$

$$\Delta\hat{\mathbf{d}}_{\Gamma_\mathrm{f}} = \mathbf{T}\,\Delta\hat{\mathbf{d}}_{\Gamma_\mathrm{s}} \,, \tag{4.1b}$$

$$\delta\hat{\mathbf{d}}_{\Gamma_\mathrm{f}} = \mathbf{T}\,\delta\hat{\mathbf{d}}_{\Gamma_\mathrm{s}} \,. \tag{4.1c}$$

Moreover, stemming from the virtual work expression in Eqn. (2.6):

$$\delta\hat{\mathbf{d}}_{\Gamma_\mathrm{f}}^{T}\,\hat{\mathbf{F}}_{\Gamma_\mathrm{f}}^{\mathrm{f}} = \delta\hat{\mathbf{d}}_{\Gamma_\mathrm{s}}^{T}\,\mathbf{T}^{T}\,\hat{\mathbf{F}}_{\Gamma_\mathrm{f}}^{\mathrm{f}} \,, \tag{4.2}$$

and the derivative of the residual contribution of the above equation gives;

$$\mathbf{T}^{T}\frac{\mathrm{d}\hat{\mathbf{F}}_{\Gamma_\mathrm{s}}^{\mathrm{f}}}{\mathrm{d}\hat{\mathbf{d}}_{\Gamma_\mathrm{f}}^{T}}\Delta\hat{\mathbf{d}}_{\Gamma_\mathrm{f}} = \mathbf{T}^{T}\underbrace{\frac{\mathrm{d}\hat{\mathbf{F}}_{\Gamma_\mathrm{s}}^{\mathrm{f}}}{\mathrm{d}\hat{\mathbf{d}}_{\Gamma_\mathrm{f}}^{T}}}_{\mathbf{K}_{\Gamma_\mathrm{f}}^{\mathrm{f}}}\mathbf{T}\,\Delta\hat{\mathbf{d}}_{\Gamma_\mathrm{s}} \,, \tag{4.3a}$$

$$\mathbf{K}_{\Gamma_\mathrm{s}}^{\mathrm{f}} = \mathbf{T}^{T}\,\mathbf{K}_{\Gamma_\mathrm{f}}^{\mathrm{f}}\,\mathbf{T} \,. \tag{4.3b}$$

Above equation defines a mapping operation for the computed interface Jacobian between dissimilar discretizations. This can be in particular beneficial when the mapping operator is defined not only for displacement type of fields but which also incorporate the rotations and cordal errors into account. In addition, the multiplication of the stiffness contribution with the transpose of the mapping operator from the left complies with the conservative mapping for force type of fields as described in [76].

Bibliography

[1] *Federal Ministry for Economic Affairs and Energy (BMWi)*. https://www.bmwi.de/Navigation/EN/Home/home.html. Accessed: 25.07.2019.

[2] J. Sobieszczanski-Sobieski. "Multidisciplinary Design Optimization: An Emerging New Engineering Discipline." In: *Advances in Structural Optimization*. Ed. by J. Herskovits. Vol. 25. Solid Mechanics and Its Applications. Dordrecht: Springer Netherlands, 1995, pp. 483–496. ISBN: 978-94-010-4203-1. DOI: 10.1007/978-94-011-0453-1_14.

[3] R. Wüchner. "Mechanik und Numerik der Formfindung und Fluid-Struktur-Interaktion von Membrantragwerken." Dissertation. Munich, Germany: Technical University of Munich, 2007.

[4] J. R. R. A. Martins and A. B. Lambe. "Multidisciplinary Design Optimization: A Survey of Architectures." In: *AIAA JOURNAL* 51.9 (2013), pp. 2049–2075. DOI: 10.2514/1.J051895.

[5] K.-U. Bletzinger. "Shape Optimization." In: *Encyclopedia of computational mechanics*. Ed. by E. Stein, R. d. Borst, and T. J. R. Hughes. Vol. 3. Chichester, West Sussex, England and Hoboken, NJ: Wiley, 2018, pp. 1–42. ISBN: 9781119003793. DOI: 10.1002/9781119176817.ecm2109.

[6] R. G. Hendrickson. *A Survey of Sensitivity Analysis Methodology*. Washington DC, USA: U.S. Department of Commerce, 1984.

[7] N. Camprubi Estebo. "Design and Analysis in Shape Optimization of Shells." Dissertation. Munich, Germany: Technical University of Munich, 2005.

[8] K.-U. Bletzinger. "A consistent frame for sensitivity filtering and the vertex assigned morphing of optimal shape." In: *Structural and Multidisciplinary Optimization* (2014). DOI: 10.1007/s00158-013-1031-5.

[9] O. Sigmund and K. Maute. "Sensitivity filtering from a continuum mechanics perspective." In: *Structural and Multidisciplinary Optimization* 46.4 (2012), pp. 471–475. DOI: 10.1007/s00158-012-0814-4.

[10] J. A. Samareh. "Survey of Shape Parameterization Techniques for High-Fidelity Multidisciplinary Shape Optimization." In: *AIAA JOURNAL* 39.5 (2001), pp. 877–884. DOI: 10.2514/2.1391.

[11] B. Louhichi, G. N. Abenhaim, and A. S. Tahan. "CAD/CAE integration: updating the CAD model after a FEM analysis." In: *The International Journal of Advanced Manufacturing Technology* 76.1-4 (2015), pp. 391–400. DOI: 10.1007/s00170-014-6248-y.

[12] J. M. Park, B. C. Lee, S. W. Chae, and K. Y. Kwon. "Surface reconstruction from FE mesh model." In: *Journal of Computational Design and Engineering* (2018). DOI: 10.1016/j.jcde.2018.05.004.

[13] A. Brune, T. Weber Martins, and R. Anderl. "Morphing boxes for the integration of shape optimization in the product design process." In: *Computer-Aided Design and Applications* 15.2 (2018), pp. 219–226. DOI: 10.1080/16864360.2017.1375672.

[14] T. W. Sederberg and S. R. Parry. "Free-form deformation of solid geometric models." In: *ACM SIGGRAPH Computer Graphics* 20.4 (1986), pp. 151–160. DOI: 10.1145/15886.15903.

[15] G. Yu, J.-D. Müller, D. Jones, and F. Christakopoulos. "CAD-based shape optimisation using adjoint sensitivities." In: *Computers & Fluids* 46.1 (2011), pp. 512–516. DOI: 10.1016/j.compfluid.2011.01.043.

[16] A. Emiroğlu, R. Wüchner, and K.-U. Bletzinger. “Treating Non-conforming Sensitivity Fields by Mortar Mapping and Vertex Morphing for Multi-disciplinary Shape Optimization.” In: *AeroStruct: Enable and Learn How to Integrate Flexibility in Design*. Ed. by R. Heinrich. Vol. 138. Notes on Numerical Fluid Mechanics and Multidisciplinary Design. Cham: Springer International Publishing, 2018, pp. 135–154. ISBN: 9783319720197. DOI: 10.1007/978-3-319-72020-3_9.

[17] A. Emiroğlu, A. Apostolatos, R. Wüchner, and K.-U. Bletzinger. “Regenerating CAD Models with OpenCASCADE and pythonOCC from Numerical Models with Application to Shape Optimization.” In: *7th GACM Colloquium on Computational Mechanics*. 2017.

[18] *Open CASCADE*. https://www.opencascade.com/. Accessed: 09.02.2019.

[19] T. Paviot. *Open CASCADE Community Edition*. https://github.com/tpaviot/oce. Accessed: 09.02.2019. 2016.

[20] T. Paviot. *pythonOCC, 3D CAD/CAE/PLM development framework for the Python programming language, PythonOCC- 3D CAD Python*. http://www.pythonocc.org/. Accessed: 09.02.2019. 2016.

[21] O. C. Zienkiewicz, R. L. Taylor, J. Zhu, P. Nithiarasu, and D. D. Fox. *The finite element method*. [6. ed. ff.] Amsterdam: Elsevier Butterworth-Heinemann, 2005. ISBN: 9781856176309.

[22] K.-U. Bletzinger. *Carat++*. http://www.st.bgu.tum.de/software/forschung/carat/. Accessed: 14.11.2018.

[23] G. Schuhmacher, F. Daoud, Ö. Petersson, and M. Wagner. “Multidisciplinary Airframe Design Optimisation.” In: 28th International Congress of the Aeronautical Sciences. 2012.

[24] R. Heinrich, ed. *AeroStruct: Enable and Learn How to Integrate Flexibility in Design: Contributions to the Closing Symposium of the German Research Initiative AeroStruct, October 13-14, 2015, Braunschweig, Germany*. Vol. 138. Notes on Numerical Fluid Mechanics and Multidisciplinary Design. Cham: Springer International Publishing, 2018. ISBN: 9783319720197. DOI: 10.1007/978-3-319-72020-3.

[25] R. W. Ogden. *Non-Linear Elastic Deformations*. Dover Civil and Mechanical Engineering. Newburyport: Dover Publications, 2013. ISBN: 9780486696485.

[26] F. B. Hildebrand. *Methods of Applied Mathematics*. Dover Books on Mathematics. Newburyport: Dover Publications, 2012. ISBN: 9780486670027.

[27] E33 Committee. *Test Method for Measuring Vibration-Damping Properties of Materials*. West Conshohocken, PA. DOI: 10.1520/E0756-05R17.

[28] J. W. Rayleigh. *The theory of sound*. 2nd ed. New York: Dover, 1945.

[29] I. Chowdhury and S. P. Dasgupta. "Computation of Rayleigh damping coefficients for large systems." In: *The Electronic Journal of Geotechnical Engineering* 8.0 (2003), pp. 1–11.

[30] N. M. Newmark. "A method of computation for structural dynamics." In: *Journal of the engineering mechanics division* 85.3 (1959), pp. 67–94.

[31] J. C. Anderson and F. Naeim. *Basic Structural Dynamics*. Hoboken, NJ, USA: John Wiley & Sons, Inc, 2012. ISBN: 9781118279137. DOI: 10.1002/9781118279137.

[32] P. D. Greening. "Dynamic finite element modelling and updating of loaded structures." Dissertation. 1999.

[33] J. H. Ferziger and M. Perić. *Computational Methods for Fluid Dynamics*. third, rev. edition. Berlin, Heidelberg and s.l.: Springer Berlin Heidelberg, 2002. ISBN: 9783540420743. DOI: 10.1007/978-3-642-56026-2.

[34] H. Jasak, A. Jemcov, and Z. Tukovic. "OpenFOAM: A C++ Library for Complex Physics Simulations." In: *International Workshop on Coupled Methods in Numerical Dynamics, IUC*. 2007, pp. 1–20.

[35] T. Dhert, T. Ashuri, and Martins, Joaquim R. R. A. "Aerodynamic shape optimization of wind turbine blades using a Reynolds-averaged Navier-Stokes model and an adjoint method." In: *Wind Energy* 20.5 (2017), pp. 909–926. DOI: 10.1002/we.2070.

[36] *HELYX Open-source CFD for Enterprise | ENGYS*. https://engys.com/products/helyx. Accessed: 10.02.2019.

[37] P. R. Spalart and S. R. Allmaras. “A one-equation turbulence model for aerodynamic flows.” In: *30th Aerospace Sciences Meeting and Exhibit*. American Institute of Aeronautics and Astronautics, 1992. DOI: 10.2514/6.1992-439.

[38] *EMPIRE: Enhanced Multi Physics Interface Research Engine*. http://empire-multiphysics.com/. Accessed: 02.10.2018.

[39] T. Wang. “Development of Co-Simulation Environment and Mapping Algorithms.” Dissertation. Munich, Germany: Technical University of Munich, 2016.

[40] A. Apostolatos, F. Péan, A. Emiroglu, R. Wüchner, and K.-U. Bletzinger. “An isogeometric mortar surface coupling method for trimmmed multipatch CAD geometries with application to FSI.” In: *7th GACM Colloquium on Computational Mechanics*. 2017.

[41] A. B. Lambe and Martins, Joaquim R. R. A. “Extensions to the design structure matrix for the description of multidisciplinary design, analysis, and optimization processes.” In: *Structural and Multidisciplinary Optimization* 46.2 (2012), pp. 273–284. DOI: 10.1007/s00158-012-0763-y.

[42] M. Cervera, R. Codina, and M. Galindo. “On the computational efficiency and implementation of block-iterative algorithms for nonlinear coupled problems.” In: *Engineering Computations* 13.6 (1996), pp. 4–30. DOI: 10.1108/02644409610128382.

[43] U. Küttler and W. A. Wall. “Fixed-point fluid-structure interaction solvers with dynamic relaxation.” In: *Computational Mechanics* 43.1 (2008), pp. 61–72. DOI: 10.1007/s00466-008-0255-5.

[44] R. Haelterman. “Analytical study of the Least Squares Quasi-Newton method for interaction problems.” Dissertation. Ghent, Belgium: Ghent University, 2009.

[45] J. Vierendeels, L. Lanoye, J. Degroote, and P. Verdonck. “Implicit coupling of partitioned fluid-structure interaction problems with reduced order models.” In: *Computers & Structures* 85.11-14 (2007), pp. 970–976. DOI: 10.1016/j.compstruc.2006.11.006.

[46] J. Degroote, K.-J. Bathe, and J. Vierendeels. “Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction.” In: *Computers & Structures* 87.11-12 (2009), pp. 793–801. DOI: 10.1016/j.compstruc.2008.11.013.

[47] C. T. Geiss et al. “The Mistralwind Project-Towards a Remaining Useful Lifetime Analysis and Holistic Asset Management Approach for More Sustainability of Wind Turbine Structures.” In: *Structural health monitoring 2017*. Lancaster, Pennsylvania: DEStech Publications, Inc, 2017. ISBN: 978-1-60595-330-4. DOI: 10.12783/shm2017/14153.

[48] U. M. Ascher and L. R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM: Society for Industrial and Applied Mathematics, 1998. ISBN: 0898714125.

[49] W. Wunderlich and G. Kiener. *Statik der Stabtragwerke*. Wiesbaden: Vieweg+Teubner Verlag, 2004. ISBN: 9783519050612. DOI: 10.1007/978-3-322-80128-9.

[50] H. R. Schwarz. *Methode der finiten Elemente: Eine Einführung unter besonderer Berücksichtigung der Rechenpraxis*. 2., überarbeitete und erweiterte Auflage. Teubner Studienbücher Mathematik. Wiesbaden: Vieweg+Teubner Verlag, 1984. ISBN: 9783519123491. DOI: 10.1007/978-3-322-96758-9.

[51] A. Steinboeck, A. Kugi, and H. A. Mang. “Energy-consistent shear coefficients for beams with circular cross sections and radially inhomogeneous materials.” In: *International Journal of Solids and Structures* 50.11-12 (2013), pp. 1859–1868. DOI: 10.1016/j.ijsolstr.2013.01.030.

[52] *Max Bögl Wind AG*. https://www.max-boegl.de/. Accessed: 09.02.2019. 2019.

[53] J. Grünberg and J. Göhlmann. *Concrete structures for wind turbines*. Beton-Kalender Series. Berlin, Germany: Ernst & Sohn a Wiley brand, 2013. ISBN: 9783433030417. DOI: 10.1002/9783433603291.

[54] C. Gehlen, K. Osterminski, K.-U. Bletzinger, R. Wüchner, and A. Emiroglu. “Monitoring and inspection of structures at large wind turbines - MISTRALWIND, Teilvorhaben 5 - Lebensdauerakte Windenergieanlagentürme : Schlussbericht : Förderzeitraum 2015-2018.” de. In: (2019). DOI: 10.2314/kxp:1667716301.

[55] M. Botz and C. U. Grosse. “Structural Health Monitoring von Windenergieanlagen.” In: 6. Jahrestagung mit 59. Forschungskolloquium des DAfStb. 2018.

[56] D. P. Bourne. “The Taylor-Couette Problem for Flow in a Deformable Cylinder.” Dissertation. Maryland, USA: University of Maryland, 2007.

[57] J. D. Anderson. *Computational Fluid Dynamics*. McGraw-Hill Education, 1995. ISBN: 0070016852.

[58] W. J. Gilbert. “Newton’s method for multiple roots.” In: *Chaos and Fractals*. Ed. by C. A. Pickover and C. A. Pickover. Burlington: Elsevier Science, 1998, pp. 327–329. ISBN: 9780444500021. DOI: 10.1016/B978-044450002-1/50055-2.

[59] A. C. Aitken. “XXV.-On Bernoulli’s Numerical Solution of Algebraic Equations.” In: *Proceedings of the Royal Society of Edinburgh* 46 (1927), pp. 289–305. DOI: 10.1017/S0370164600022070.

[60] S. B. Derek. *Design of 9-meter carbon-fiberglass prototype blades : CX-100 and TX-100 : final project report*. Tech. rep. Sept. 2007. DOI: 10.2172/921145.

[61] S. B. Derek. *CX-100 manufacturing final project report*. Tech. rep. Nov. 2007.

[62] B. R. Resor and J. A. Paquette. *A NuMAD model of the Sandia CX-100 blade*. Tech. rep. Oct. 2012. DOI: 10.2172/1055873.

[63] F. Alauzet and O. Pironneau. “Continuous and discrete adjoints to the Euler equations for fluids.” In: *International Journal for Numerical Methods in Fluids* 70.2 (2012), pp. 135–157. DOI: 10.1002/fld.2681. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/fld.2681.

[64] S. Nadarajah and A. Jameson. “A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization.” In: *38th Aerospace Sciences Meeting and Exhibit*. Reston, Virigina: American Institute of Aeronautics and Astronautics, 1102000, p. 5. DOI: 10.2514/6.2000-667.

[65] C. Othmer. “A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows.” In: *International Journal for Numerical Methods in Fluids* 58.8 (2008), pp. 861–877. DOI: 10.1002/fld.1770.

[66] G. K. Karpouzas, E. M. Papoutsis-Kiachagias, T. Schumacher, E. d. Villiers, K. C. Giannakoglou, and C. Othmer. “Adjoint Optimization for Vehicle External Aerodynamics.” In: *International Journal of Automotive Engineering* 7.1 (2016), pp. 1–7. DOI: 10.20485/jsaeijae.7.1_1.

[67] J. R. Martins, J. J. Alonso, and J. J. Reuther. “A Coupled-Adjoint Sensitivity Analysis Method for High-Fidelity Aero-Structural Design.” In: *Optimization and Engineering* 6.1 (2005), pp. 33–62. DOI: 10.1023/B:OPTE.0000048536.47956.62.

[68] K.-U. Bletzinger. *Lecture notes in Structural Optimization*. 2016.

[69] R. Dwight and J. Brezillon. “Effect of Various Approximations of the Discrete Adjoint on Gradient-Based Optimization.” In: *44th AIAA Aerospace Sciences Meeting and Exhibit*. [Reston, VA]: [American Institute of Aeronautics and Astronautics], 2006, p. 2005. ISBN: 978-1-62410-039-0. DOI: 10.2514/6.2006-690.

[70] I. S. Kavvadias, E. M. Papoutsis-Kiachagias, and K. C. Giannakoglou. “On the proper treatment of grid sensitivities in continuous adjoint methods for shape optimization.” In: *Journal of Computational Physics* 301 (2015), pp. 1–18. DOI: 10.1016/j.jcp.2015.08.012.

[71] W. Anderson and V. Venkatakrishnan. “Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation.” In: *Computers & Fluids* 28.4-5 (1999), pp. 443–480. DOI: 10.1016/S0045-7930(98)00041-3.

[72] C. Lozano, E. Andrés, M. Martín, and P. Bitrián. “Domain versus boundary computation of flow sensitivities with the continuous adjoint method for aerodynamic shape optimization problems.” In: *International Journal for Numerical Methods in Fluids* 70.10 (2012), pp. 1305–1323. DOI: 10.1002/fld.2743.

[73] T. D. Economon, F. Palacios, and J. J. Alonso. “A Viscous Continuous Adjoint Approach for the Design of Rotating Engineering Applications.” In: *21st AIAA Computational Fluid Dynamics Conference*. Reston, Virginia: American Institute of Aeronautics and Astronautics, 6242013, p. 81. DOI: 10.2514/6.2013-2580.

[74] M. Hojjat, E. Stavropoulou, and K.-U. Bletzinger. “The Vertex Morphing method for node-based shape optimization.” In: *Computer Methods in Applied Mechanics and Engineering* 268 (2014), pp. 494–513. DOI: 10.1016/j.cma.2013.10.015.

[75] B. I. Wohlmuth. “Discretization Techniques Based on Domain Decomposition.” In: *Discretization methods and iterative solvers based on domain decomposition*. Ed. by B. I. Wohlmuth. Vol. 17. Lecture notes in computational science and engineering [LNCSE]. Berlin [etc.]: Springer, 2001, pp. 1–84. ISBN: 978-3-540-41083-6. DOI: 10.1007/978-3-642-56767-4_1.

[76] C. Farhat, M. Lesoinne, and P. Le Tallec. “Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity.” In: *Computer Methods in Applied Mechanics and Engineering* 157.1-2 (1998), pp. 95–114. DOI: 10.1016/S0045-7825(97)00216-8.

[77] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs. “Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement.” In: *Computer Methods in Applied Mechanics and Engineering* 194.39-41 (2005), pp. 4135–4195. DOI: 10.1016/j.cma.2004.10.008.

[78] A. Ehrl, A. Popp, V. Gravemeier, and W. A. Wall. “A dual mortar approach for mesh tying within a variational multiscale method for incompressible flow.” In: *International Journal for Numerical Methods in Fluids* 76.1 (2014), pp. 1–27. DOI: 10.1002/fld.3920.

[79] J. D. Sanders, T. A. Laursen, and M. A. Puso. “A Nitsche embedded mesh method.” In: *Computational Mechanics* 49.2 (2012), pp. 243–257. DOI: 10.1007/s00466-011-0641-2.

[80] A. Apostolatos, M. Breitenberger, R. Wüchner, and K.-U. Bletzinger. “Domain Decomposition Methods and Kirchhoff-Love Shell Multipatch Coupling in Isogeometric Analysis.” In: *Isogeometric analysis and applications 2014*. Ed. by B. Jüttler and B. Simeon. Vol. 107. Lecture Notes in Computational Science and Engineering. Cham, Heidelberg, and New York: Springer, 2015, pp. 73–101. ISBN: 978-3-319-23314-7. DOI: 10.1007/978-3-319-23315-4_4.

[81] A. Apostolatos. “Isogeometric Analysis of Thin-Walled Structures on Multipatch Surfaces in Fluid-Structure Interaction.” Dissertation. Munich, Germany: Technical University of Munich, 2019.

[82] T. Teschemacher, A. M. Bauer, T. Oberbichler, M. Breitenberger, R. Rossi, R. Wüchner, and K.-U. Bletzinger. “Realization of CAD-integrated shell simulation based on isogeometric B-Rep analysis.” In: *Advanced Modeling and Simulation in Engineering Sciences* 5.1 (2018), p. 276. DOI: 10.1186/s40323-018-0109-4.

[83] O. Weeger, U. Wever, and B. Simeon. “Nonlinear frequency response analysis of structural vibrations.” In: *Computational Mechanics* 54.6 (2014), pp. 1477–1495. DOI: 10.1007/s00466-014-1070-9.

[84] P. A. Kuberry. “An Optimization-Based Approach to Decoupling Fluid-Structure Interaction.” Dissertation. South Carolina, USA: Clemson University, 2015.

[85] D. I. Papadimitriou and K. C. Giannakoglou. “Aerodynamic Shape Optimization Using First and Second Order Adjoint and Direct Approaches.” In: *Archives of Computational Methods in Engineering* 15.4 (2008), pp. 447–488. DOI: 10.1007/s11831-008-9025-y.

[86] D. I. Papadimitriou and K. C. Giannakoglou. “Computation of the Hessian matrix in aerodynamic inverse design using continuous adjoint formulations.” In: *Computers & Fluids* 37.8 (2008), pp. 1029–1039. DOI: 10.1016/j.compfluid.2007.11.001.

[87] A.-V Vuong, C. Giannelli, B. Jüttler, and B. Simeon. “A hierarchical approach to adaptive local refinement in isogeometric analysis.” In: *Computer Methods in Applied Mechanics and Engineering* 200.49-52 (2011), pp. 3554–3567. DOI: 10.1016/j.cma.2011.09.004.

[88] D. Schillinger, L. Dedè, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, and T. J. R. Hughes. “An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces.” In: *Computer Methods in Applied Mechanics and Engineering* 249-252 (2012), pp. 116–150. DOI: 10.1016/j.cma.2012.03.017.

Bisherige Titel der Schriftenreihe

Band	Titel
1	Frank Koschnick, *Geometrische Lockingeffekte bei Finiten Elementen und ein allgemeines Konzept zu ihrer Vermeidung*, 2004.
2	Natalia Camprubi, *Design and Analysis in Shape Optimization of Shells*, 2004.
3	Bernhard Thomee, *Physikalisch nichtlineare Berechnung von Stahlfaserbetonkonstruktionen*, 2005.
4	Fernaß Daoud, *Formoptimierung von Freiformschalen - Mathematische Algorithmen und Filtertechniken*, 2005.
5	Manfred Bischoff, *Models and Finite Elements for Thin-Walled Structures*, 2005.
6	Alexander Hörmann, *Ermittlung optimierter Stabwerkmodelle auf Basis des Kraftflusses als Anwendung plattformunabhängiger Prozesskopplung*, 2006.
7	Roland Wüchner, *Mechanik und Numerik der Formfindung und Fluid-Struktur-Interaktion von Membrantragwerken*, 2006.
8	Florian Jurecka, *Robust Design Optimization Based on Metamodeling Techniques*, 2007.
9	Johannes Linhard, *Numerisch-mechanische Betrachtung des Entwurfsprozesses von Membrantragwerken*, 2009.
10	Alexander Kupzok, *Modeling the Interaction of Wind and Membrane Structures by Numerical Simulation*, 2009.
11	Bin Yang, *Modified Particle Swarm Optimizers and their Application to Robust Design and Structural Optimization*, 2009.

Band Titel

12 Michael Fleischer, *Absicherung der virtuellen Prozesskette für Folgeoperationen in der Umformtechnik*, 2009.

13 Amphon Jrusjrungkiat, *Nonlinear Analysis of Pneumatic Membranes - From Subgrid to Interface*, 2009.

14 Alexander Michalski, *Simulation leichter Flächentragwerke in einer numerisch generierten atmosphärischen Grenzschicht*, 2010.

15 Matthias Firl, *Optimal Shape Design of Shell Structures*, 2010.

16 Thomas Gallinger, *Effiziente Algorithmen zur partitionierten Lösung stark gekoppelter Probleme der Fluid-Struktur-Wechselwirkung*, 2011.

17 Josef Kiendl, *Isogeometric Analysis and Shape Optimal Design of Shell Structures*, 2011.

18 Joseph Jordan, *Effiziente Simulation großer Mauerwerksstrukturen mit diskreten Rissmodellen*, 2011.

19 Albrecht von Boetticher, *Flexible Hangmurenbarrieren: Eine numerische Modellierung des Tragwerks, der Hangmure und der Fluid-Struktur-Interaktion*, 2012.

20 Robert Schmidt, *Trimming, Mapping, and Optimization in Isogeometric Analysis of Shell Structures*, 2013.

21 Michael Fischer, *Finite Element Based Simulation, Design and Control of Piezoelectric and Lightweight Smart Structures*, 2013.

22 Falko Hartmut Dieringer, *Numerical Methods for the Design and Analysis of Tensile Structures*, 2014.

23 Rupert Fisch, *Code Verification of Partitioned FSI Environments for Lightweight Structures*, 2014.

24 Stefan Sicklinger, *Stabilized Co-Simulation of Coupled Problems Including Fields and Signals*, 2014.

Band Titel

25 Madjid Hojjat, *Node-based parametrization for shape optimal design*, 2015.

26 Ute Israel, *Optimierung in der Fluid-Struktur-Interaktion - Sensitivitätsanalyse für die Formoptimierung auf Grundlage des partitionierten Verfahrens*, 2015.

27 Electra Stavropoulou, *Sensitivity analysis and regularization for shape optimization of coupled problems*, 2015.

28 Daniel Markus, *Numerical and Experimental Modeling for Shape Optimization of Offshore Structures*, 2015.

29 Pablo Suárez, *Design Process for the Shape Optimization of Pressurized Bulkheads as Components of Aircraft Structures*, 2015.

30 Armin Widhammer, *Variation of Reference Strategy - Generation of Optimized Cutting Patterns for Textile Fabrics*, 2015.

31 Helmut Masching, *Parameter Free Optimization of Shape Adaptive Shell Structures*, 2016.

32 Hao Zhang, *A General Approach for Solving Inverse Problems in Geophysical Systems by Applying Finite Element Method and Metamodel Techniques*, 2016.

33 Tianyang Wang, *Development of Co-Simulation Environment and Mapping Algorithms*, 2016.

34 Michael Breitenberger, *CAD-integrated Design and Analysis of Shell Structures*, 2016.

35 Önay Can, *Functional Adaptation with Hyperkinematics using Natural Element Method: Application for Articular Cartilage*, 2016.

36 Benedikt Philipp, *Methodological Treatment of Non-linear Structural Behavior in the Design, Analysis and Verification of Lightweight Structures*, 2017.

Band Titel

37 Michael Andre, *Aeroelastic Modeling and Simulation for the Assessment of Wind Effects on a Parabolic Trough Solar Collector*, 2018.

38 Andreas Apostolatos, *Isogeometric Analysis of Thin-Walled Structures on Multipatch Surfaces in Fluid-Structure Interaction*, 2019.