Technische Universität München

Ingenieurfakultät Bau Geo Umwelt

Lehrstuhl für Statik

Shape optimization and sensitivity analysis of fluids, structures, and their interaction using Vertex Morphing parametrization

Reza Najian Asl

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

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender:

Prof. Dr.-Ing. habil. Fabian Duddeck

Prüfer der Dissertation:

1. Prof. Dr.-Ing. Kai-Uwe Bletzinger
2. Prof. Dr. Kurt Maute
3. Prof. Dr. Wulf Dettmer

Die Dissertation wurde am 31.05.2019 bei der Technischen Universität München eingereicht und durch die Ingenieurfakultät Bau Geo Umwelt am 16.08.2019 angenommen.

Schriftenreihe des Lehrstuhls für Statik TU München

Band 37

Reza Najian Asl

SHAPE OPTIMIZATION AND SENSITIVITY ANALYSIS OF FLUIDS, STRUCTURES, AND THEIR INTERACTION USING VERTEX MORPHING PARAMETRIZATION

München 2019

Abstract

Having reached a matured stage in shape design optimization for single disciplinary problems in both academic and industrial environments, the research community has paid great attention to multi-disciplinary optimization (MDO) for complex applications. Motivated by aerospace applications, this thesis targets specific class of MDO problems, namely node-based shape optimization in fluid-structure interaction.

Herein, Vertex Morphing as a consistent surface control technique is used for node-based shape optimization. The suitability of this technique has been assessed and demonstrated for a wide range of engineering applications. Moreover, in this contribution, a consistent formulation is presented for the implementation of numerous point-wise no-penetration constraints over arbitrary surfaces which act as bounding surfaces, also known as packaging constraints. To verify the consistency of the derivations two applications of CFD shape optimization in the automotive industry are presented.

Another main focus of this work is the consistent derivation of adjoint-based gradients of force-based objective functionals, which are defined on the deformed fluid-structure interface, with respect to the shape design variables of the undeformed interface geometry. For this purpose, all important aspects of the high-fidelity coupling between fluid and structure are covered, from solution strategy for direct and adjoint problems to criteria for mapping primal and adjoint fields between non-matching interface meshes. The novelty in this work is the consistent derivation of sets of adjoint coupling conditions which are immediately applicable to existing adjoint fluid and structure solvers. This is performed independent of the adjoint method, the discretization method, and the mathematical modelling of the flow. This enables us to take the utmost advantage of sophisticated single disciplinary adjoint solvers without the need to compute expensive cross-derivatives.

In order to demonstrate the validity and generality of the derivations, three adjoint FSI frameworks are considered: FEM-based using discrete adjoints, hybrid FEM-FVM-based using discrete-continuous hybrid adjoints, and hybrid FEM-FVM-based using discrete adjoints. The first one is used for benchmarking, studying adjoint solution strategies, and for comparing alternative approaches in coupled shape sensitivity analysis including a fully monolithic formulation. Whereas,

the third framework is used for shape sensitivity analysis of a representative aero-elastic wing based on the derived fully modular approach, which leverages primal and adjoint capabilities of available open-source codes. This analysis also includes a critical comparative assessment of consistent and conservative criteria for mapping primal and adjoint fields between non-matching interface meshes. Besides, special attention is paid to investigate the importance of considering the mesh motion problem in the adjoint FSI analysis.

Acknowledgments

Firstly, I would like to express my sincere gratitude to Prof. Dr.-Ing. Kai-Uwe Bletzinger for giving me the chance to do research at the Chair of Structural Analysis, and for the freedom he gave me in choosing my PhD topic and scheduling the work. I have been growing as an independent researcher in these years which will have an important influence to my future career and life.

My sincere thanks also go to my mentor Dr. Majid Hojjat who provided me an opportunity to join his field of research and work. Without his precious support and insightful comments, it would not be possible to conduct this research.

I thank gratefully Dr.-Ing. Roland Wüchner and Prof. Wulf Dettmer for the organization of my research abroad and the countless time and effort in discussions of research and publications. The support of the International Graduate School of Science and Engineering (IGSSE) of the Technische Universität München is gratefully acknowledged.

Besides, I would like to thank the rest of my thesis committee: Prof. Kurt Maute for being my co-examiner and for sharing his time and insights in my work. Also I want to thank Univ.-Prof. Dr.-Ing. habil. Fabian Duddeck for being the chairman of my committee.

My gratitude goes also to my fellows at the Chair of Structural Analysis for enjoyable moments as well as good advice and collaboration. I am especially grateful to Shahrokh Shayegan, Aditya Ghantasala, and Daniel Baumgärtner for their friendship and all the good times we shared together.

Last but not the least, I would like to thank my family for supporting me spiritually throughout writing this thesis and my life in Germany.

Reza Najian Asl
Technische Universität München
May 22, 2019

Contents

Nomenclature

Domains, Boundaries and Configurations

Ω	Computational domain
Ω^S	Structural domain
Ω^F	Fluid domain
Γ	Wet fluid-structure interface
Ψ	Boundary of Σ
X	Undeformed configuration
x	Deformed configuration
$\boldsymbol{X}$	Spatial/nodal coordinates at the undeformed configuration
$\boldsymbol{x}$	Spatial/nodal coordinates at the deformed configuration

Superscripts and Subscripts

$(\cdot)_i$	ith vector component
$(\cdot)_{i,j}$	Matrix component at ith row and jth column
$(\cdot)_\Gamma$	Quantity at the fluid-structure interface
$(\cdot)_{\Gamma,i}$	ith component of a vectorial quantity at the fluid-structure interface
$(\cdot)_\Omega$	Quantity inside the computational domain
$(\cdot)_X$	Quantity evaluated at the undeformed configuration
$(\cdot)_x$	Quantity evaluated at the deformed configuration

$(\cdot)^1$	X component
$(\cdot)^2$	Y component
$(\cdot)^3$	Z component
$(\cdot)^S$	Variable/quantity belongs to the structure
$(\cdot)^F$	Variable/quantity belongs to the fluid
$(\cdot)^M$	Variable/quantity belongs to the fluid mesh motion
$(\cdot)^{VM}$	Variable/quantity belongs to Vertex Morphing method
${}^k(\cdot)$	Value at kth iteration

Operators and Symbols

$\hat{(\cdot)}$	Relaxed quantity
$\|\|\cdot\|\|/\|\|\cdot\|\|_2$	L2 norm
$\cdot$	Vector inner product
$\times$	Vector cross product
$\otimes$	Vector outer product
$\delta(\cdot)$	First variation of a quantity
$\delta_{(\cdot\cdot)}(\cdot)$	First variation of $(\cdot)$ w.r.t $(\cdot\cdot)$
$\frac{\partial(\cdot)}{\partial(\cdot\cdot)}$	Partial derivative of $(\cdot)$ w.r.t $(\cdot\cdot)$
$\frac{d(\cdot)}{d(\cdot\cdot)}$	Total derivative of $(\cdot)$ w.r.t $(\cdot\cdot)$
$\nabla(\cdot)$	Spatial gradient operator
$\nabla_X(\cdot)$	Spatial gradient operator acting on the undeformed configuration
$\nabla_x(\cdot)$	Spatial gradient operator acting on the deformed configuration
$\nabla_{(\cdot\cdot)}(\cdot)$	Gradient of $(\cdot)$ with respect to $(\cdot\cdot)$
$\nabla\cdot(\cdot)$	Spatial divergence operator
$(\cdot)^{-1}$	Inverse of a tensor
$(\cdot)^T$	Transpose of a tensor
$\kappa(\cdot)$	Condition number of a tensor
$\Delta(\cdot)$	Increment, change of $(\cdot)$ in one iteration
$\boldsymbol{I}$	Identity matrix

Shape Optimization

J	Objective function to be minimized
Γ	Design surface boundary of the computational domain Ω
$\boldsymbol{X}_\Gamma$	Vector of nodal coordinates as shape design variables

$\boldsymbol{Q}$	State/primal vector
Γ_g	Subset of Γ subject to no-penetration constraint
Γ_h	Subset of Γ subject to slip constraint
$\boldsymbol{g}$	Vector of in-equality constraints
$\boldsymbol{g}_a$	Vector of active in-equality constraints
$\boldsymbol{g}_\Gamma$	Vector of in-equality no-penetration constraints over Γ_g
$\boldsymbol{h}$	Vector of equality constraints
$\boldsymbol{g}_\Gamma$	Vector of equality slip constraints over Γ_h
$\boldsymbol{r}$	Residual vector of the state/primal governing equations
$\boldsymbol{D}_j$	Damping direction at the jth node
$\boldsymbol{X}_{cpp}$	Closest point projection (CPP) of a given point
$\boldsymbol{p}$	Feasible search direction
$\boldsymbol{C}$	Jacobian of the linearized constraints w.r.t the nodal coordinates
$\boldsymbol{C}_m$	Jacobian of the linearized constraints w.r.t the design variables
L	Lagrange function
$\boldsymbol{\lambda}$	Vector of Lagrange multipliers
$\boldsymbol{T}$	Projection matrix
ζ	Penalty parameter

Vertex Morphing

$\boldsymbol{s}$	Control field/vector
F	Filter function
r	Filter radius
Σ	portion of Γ lying within a sphere of specified radius centered at Γ
$\boldsymbol{v}^{VM}$	VM design velocity
$\boldsymbol{A}$	Filtering mapping matrix
$\boldsymbol{A}^{FS}$	Filtering mapping matrix in FSI on non-matching meshes
$\boldsymbol{r}^{VM}$	Residual vector of VM parametrization

Fluid-Structure Interaction

$\boldsymbol{R}^S$	Strong form of structure governing equations
$\boldsymbol{R}^F$	Strong form of fluid governing equations
$\boldsymbol{R}^M$	Strong form of fluid mesh motion governing equations

$\boldsymbol{Q}$	FSI State variables
$\boldsymbol{w}$	Fluid State variables
$\boldsymbol{r}^S$	Full residual vector of structure
$\boldsymbol{r}^F$	Full residual vector of fluid
$\boldsymbol{r}^M$	Full residual vector of fluid mesh motion
$\boldsymbol{r}$	Full residual vector of FSI
$\boldsymbol{f}^{S,int}$	Vector of internal forces of structure
$\boldsymbol{f}^{S,ext}$	Vector of external forces of structure
$\mathcal{F}$	Fluid solver
$\mathcal{S}$	Structure solver
$\mathcal{M}$	Mesh motion solver
$\boldsymbol{\delta}$	Interface displacement residuals
$\boldsymbol{H}^{FS}$	Consistent transformation matrix from structure and fluid
$\boldsymbol{H}^{SF}$	Consistent transformation matrix from fluid to structure

Adjoint-based Shape Sensitivity Analysis

J	Force-based objective function
$\boldsymbol{d}$	Force projection vector
$\boldsymbol{d}^*$	Modified force projection vector
J^*	Force-based objective function with modified force projection vector
$\boldsymbol{\Psi}$	Vector of the adjoint variables associated with $\boldsymbol{r}$
$\boldsymbol{\Psi}^F$	Vector of the adjoint variables associated with $\boldsymbol{r}^F$
$\boldsymbol{\Psi}^S$	Vector of the adjoint variables associated with $\boldsymbol{r}^S$
$\boldsymbol{\Psi}^M$	Vector of the adjoint variables associated with $\boldsymbol{r}^M$
g^{FSI}	FSI adjoint-based shape gradient vector
g^F	Adjoint-based shape gradient vector of fluid problem
g^S	Adjoint-based shape gradient vector of structure problem
g^M	Adjoint-based shape gradient vector of fluid mesh motion problem
$\boldsymbol{f}^{S,a}$	Structural adjoint force (right-hand side of the adjoint structure problem)
$\boldsymbol{f}^{M,a}$	Domain-based fluid adjoint force for the adjoint mesh motion problem

Abbreviations

MDO	Multi-disciplinary optimization
FSI	Fluid-structure interaction
VM	Vertex Morphing
CAD	Computer-aided design
NURBS	Non-uniform rational B-spline
CPP	Closest point projection
CFD	Computational fluid dynamics
CSD	Computational structural dynamics
CS	Control script
FEM	Finite element method
FVM	Finite volume method
RANS	Reynolds-Averaged Navier–Stokes
AD	Automatic differentiation
AoA	Angle of attack
HVAC	Heating, ventilation, and air conditioning
GMRES	Generalized minimal residual
KKT	Karush–Kuhn–Tucker
SUPG	Streamline upwind Petrov–Galerkin
JST	Jameson-Schmidt-Turkel
PDE	Partial differential equation
SLP	Sequential linear programming

Chapter 1

Introduction

Numerical design optimization has received great attention in academia as well as in industry, where the aim is to maximize the ratio of the design performance to the design cost. The latter, in particular, involves complex reality and therefore issues like robustness, flexibility and automation of the computational framework have to be addressed. Recent developments in computational science, engineering and power have made it possible to remove simplifying assumptions which may result in infeasible design and even unphysical results. Typical examples can be found in multidisciplinary design problems, particularly fluid-structure interaction (FSI) problems where flexibility can have large impact on both the aerodynamic response [1, 6, 41, 78, 89] and the sensitivity/gradient of the response with respect to (w.r.t) design variables [7]. Among various research topics in multidisciplinary design optimization (MDO) [58], FSI shape sensitivity analysis and optimization, which are the main topics of discussion in this work, have been active areas of research over the past two decades. The origins of high-fidelity FSI optimization can be traced back to the works of Giunta et al. [32] and Maute et al. [59], both of which optimized aero-elastic wings with respect to few shape design variables.

1.1 Vertex Morphing for Node-based Shape Optimization

The shape parametrization techniques can be divided into two major categories: CAD-based and CAD-free, each of which has its own pros and cons. For example, the main advantage of the CAD-based approach over the others is that it is intrinsically linked to the CAD model of design surface at the cost of numerical differentiation of the CAD model w.r.t surface mesh. On the other hand, node-based parametrization as a CAD-free approach provides the richest design space which can result in shape modes that can not be captured by a coarse CAD model. For a detailed comparison and review of shape parametrization techniques, the reader is kindly referred to [75].

Vertex Morphing (VM), a consistent shape control technique that ensures smooth shape transitions, was introduced recently by Hojjat et al. [14, 39] as a node-based shape optimization tool to be used in industrial design processes. Simply speaking, Vertex Morphing introduces a design control space in which the mathematical optimization problem is defined in parallel to the geometry space, where shape updates are applied. Then, it defines a mapping, which enforces smoothness and the mesh quality criterion, to perform transforms between the geometry space and the control space. So far, the capabilities of this technique have been evaluated within design cycles at the BMW group, Volkswagen AG [63], and ONERA [11]. As the next step, packaging constraints (or, more generally, geometrical shape constraints) are introduced into Vertex Morphing methodology without loss of generality and robustness as an indispensable part of the industrial design process.

Unlike shape parametrization techniques, optimization under manufacturing constraints, specially packaging constraints, has not been well discussed and well reported in literature. A potential reason for this could be that shape parametrization itself is challenging and is still a subject of on-going research. Another reason could be the emergence of arbitrary geometrical constraints from industry, which is still lacking a robust and fully automated framework even for unconstrained shape optimization [76]. Different types of manufacturing and geometrical constraints can be found in literature. For example, Schmitt et al. [79, 80] enforced different types of symmetry, curvature, and demolding constraints in a node–based shape optimization technique. Thickness control has been investigated by many researchers because of its importance for manufacturability [4,

37]. Furthermore, various types of level-set based shape (and topology) optimization including feature control and geometric constraints can be found in [3, 33, 46, 61, 92].

This work introduces the first successful inclusion of packaging constraints in the framework of Vertex Morphing. Herein, packaging constraints will be termed as bounding surfaces. This is of great practical importance for maintaining precise control of packaging in industrial design processes. A comprehensive survey of packing (layout) problems and algorithms can be found in [88] and the references therein. Regarding the imposition of this type of manufacturing constraint, the constraint is quantified at the nodes of the design surface mesh [point-wise constraint [3, 28, 46]]. This is advantageous not only because of the strong imposition of the constraint [53] but also because of the shape parametrization technique used in this work. An intuitive approach to enforce the aforementioned point-wise geometrical constraints would be to formulate them in the geometry space. Then, incorporate them into the shape optimization problem in the design control space using the gradient projection technique [51]. In this way, incorporation of constraints is consistent with Vertex Morphing technique. To avoid numerical problems arising in the gradient projection step, the linear least squares approximation to the Lagrange multipliers is used. The properties of the aforementioned idea are:

1. The no-penetration constraint is defined point-wise to satisfy the constraint at every point of the optimization surface.

2. The feasibility of the shape is guaranteed at each optimization step using the gradient projection technique.

3. All properties of the Vertex Morphing method discussed in [14, 38, 39] are preserved, which means that smooth geometries are always generated regardless of how non-smooth the bounding surface is.

Finally, the results confirm that the proposed formulation provides a consistent frame for node-based shape optimization under geometric shape constraints.

1.2 Sensitivity Analysis Methods for Multidisciplinary Shape Optimisation

Fluid-structure interaction (FSI) as a complex multiphysics problem has received special attention from the research community not only because of its applications but also because of its complicated and challenging mathematics. Although a numerical solution of the primal FSI problem has been intensely studied for a variety of aspects, the adjoint FSI problem and especially black-box adjoint FSI coupling have not been well established in the literature. In computational FSI and also this work, the following topics have received considerable attention:

- *Coupling Strategy*: In general, the methods are classified into two main categories, the monolithic methods [30, 73] and the partitioned methods [19, 21, 27, 82]. The former method solves the whole primal FSI problem in a single coupled system, while the later solves the fluid and the structure separately by imposing the coupling conditions at the interface. For comparisons of the partitioned and monolithic FSI algorithms, the reader is referred to the following references [19, 35].

- *Non-Conforming Interfaces*: Due to the different physics, fluid and structure may require different mesh resolutions resulting in the non-matching interfaces. Therefore, in order to apply the interface boundary conditions, mapping is required between the non-matching interface meshes. Handling non-matching fluid and structural interface discretizations is considered a critical requirement for the development of a robust multi-disciplinary framework, especially when the partitioned coupling strategy is followed. The papers by [20] and [91] are recent works pertaining to this research topic in FSI.

Although there is a rich literature addressing the FSI problem (primal problem), the adjoint FSI problem and especially adjoint coupling conditions have not been covered in detail. Early attempts in adjoint-based shape sensitivity analysis for FSI were made by [60], [55] and [70]. This topic of research has been followed by [42, 56–58, 95]. Among recent trends and developments in this area, the following works are notable and addressed here. [77] established an open-source framework which is based on fixed-point iterations for the adjoint variables of the coupled system using an AD

tool. The main benefit of such an approach is that there is no need to construct the analytic Jacobian for the coupled physical problem. On the other hand, the computational cost might hinder the applicability to realistic problems. [47] presented a coupling framework for aeroelastic analysis and optimization using discrete adjoint-based gradients. They systematically derived the discrete adjoint corresponding to the steady aeroelastic analysis in a consistent way. Applicability of this approach might be limited in a partitioned adjoint FSI environment due to the lack of availability of the required cross-coupling/cross-dependent terms in every software package. A literature review of the studies by various authors shows that coupled-adjoint sensitivity analysis for high-fidelity aero-structural design is divided into two main formulations: a three-field formulation followed by [10, 47, 77, 94] and a two-field formulation followed by [26, 36, 84]. Three-field formulation accounts for aerodynamic, structural, and mesh deformation residuals in adjoint-based sensitivity analysis while two-field formulation does not include mesh motion in the sensitivity analysis. As a matter of fact, the mesh motion problem is only an auxiliary problem to maintaining good mesh quality in the fluid mesh deformation step and the FSI solution must be completely independent of the mesh and the mesh motion of choice. To the author's knowledge, there is no paper on a detailed comparison between these two formulations. Filling this gap serves as one of the main motivations of this work.

Although all mentioned references are concerned with the calculation of coupled adjoint-based sensitivities for aero-elastic system, the following issues have not been addressed in detail and are thus discussed within the present contribution:

- *Necessity of the fully coupled adjoint-based sensitivities.* Our studies show that, depending on the type of objective function, the amount of interface flexibility and the flow behavior around the interface, uncoupled adjoint-based sensitivities may deliver sufficiently accurate gradients for shape optimization.
- *Derivation of general and unique adjoint coupling conditions.* Though adjoint FSI analysis has been exploited by researchers, proper and unique coupling conditions for adjoint FSI problems have not been thoroughly discussed and documented in the literature. This work reviews most representative scenarios for primal and adjoint problems and derives respective adjoint boundary conditions.

- *Clear distinction between deformed and undeformed configurations of the interface.* This is of high importance for aerodynamic shape optimization of the fluid-structure interface, because shape design variables are defined on the undeformed interface while the objective is evaluated on the deformed interface. In most cases, the objective function is evaluated on the deformed interface but the optimal shape of the undeformed interface is of interest.

- *Evaluation of mapping criterion*, more specifically consistent and conservative, for exchanging adjoint structural displacements and adjoint forces across non-matching fluid and structure interfaces.

1.3 Outline

This thesis is structured as follows: In Chapter 2, the mathematical model of the constrained shape optimization problem in the geometry space and the control space of Vertex Morphing is presented. This includes a comprehensive discussion of active constraint detection. Numerical benchmarking, including discussions on numerical properties and different aspects of the method, is presented in Section 2.5.

In Chapter 3, the stationary fluid-structure interaction problem is presented. This includes a review of well-known strategies for the solution of the FSI problem. Next, criteria for mapping data fields between non-matching grids is briefly discussed in the context of FSI. Based on the introduced background, Chapter 4 focuses on the adjoint-based shape sensitivity analysis for stationary FSI problems, including two-field-based and three-field-based partitioning of the adjoint problem.

Chapters 5 and 6 present frameworks that are used to study important aspects of the high-fidelity coupling between fluid and structure, from solution strategies for adjoint problems to criteria for mapping primal and adjoint fields between non-matching interface meshes. Last but not least, Chapter 7 demonstrates and explores the applicability of the ideas through challenging large scale applications. Finally, Chapter 8 gives main conclusions of this work.

Chapter 2

Node-based Shape Optimization with Vertex Morphing

This chapter contains comprehensive discussions on the derivation and theoretical background of Vertex Morphing parametrization and the treatment of geometrical constraints, particularly slip condition over the boundary of the design surface, as well as, no-penetration condition on the design surface. Simply speaking, the goal is to find shapes/shape updates which do not penetrate non-penetrative surfaces/obstacles (possibly overlaps) while slide along the boundary edges of the design surface. In the following, the described shape optimization problem is formulated in the geometry space first, then it is posed in the framework of Vertex Morphing. This is followed by numerical analyses of VM in terms of the conditioning of the mapping operator and the feasibility of shapes generated by VM-based shape optimization.

Material for the present chapter is partially derived from the publication of the author and his coworkers([8]).

2.1 Constrained Shape Optimization

The goal is to optimize a discrete surface and satisfy inequality and equality constraints concurrently. Following the node-based approach [54], the shape optimization problem of interest may be defined as

$$\begin{aligned}
\min_{\boldsymbol{X}_\Gamma} \quad & J(\boldsymbol{X}_\Gamma, \boldsymbol{Q}(\boldsymbol{X}_\Gamma)) \\
\text{subject to} \quad & g_j(\boldsymbol{X}_\Gamma, \boldsymbol{Q}(\boldsymbol{X}_\Gamma)) \leq 0, \quad j = 1, \ldots, m_g. \\
& g_{\Gamma,j}(\boldsymbol{X}_{\Gamma_g,j}) \in \boldsymbol{g}, \quad j = 1, \ldots, m_{\Gamma_g}. \\
& h_j(\boldsymbol{X}_\Gamma, \boldsymbol{Q}(\boldsymbol{X}_\Gamma)) = 0, \quad j = 1, \ldots, m_h. \\
& h_{\Gamma,j}(\boldsymbol{X}_{\Gamma_h,j}) \in \boldsymbol{h}, \quad j = 1, \ldots, m_{\Gamma_h}. \\
& \boldsymbol{r}_j(\boldsymbol{X}_\Gamma, \boldsymbol{Q}(\boldsymbol{X}_\Gamma)) = \boldsymbol{0}, \quad j = 1, \ldots, m_\Omega + m_\Gamma.
\end{aligned} \tag{2.1}$$

where the terminology is taken from Figure 2.1, $\Gamma \subset \partial\Omega$ is the design boundary of the computational domain Ω; unless noted otherwise, $\boldsymbol{X}_\Gamma = \left[\boldsymbol{X}_{\Gamma,1}, \boldsymbol{X}_{\Gamma,2}, ..., \boldsymbol{X}_{\Gamma,m_\Gamma}\right]$, $\boldsymbol{X}_{\Gamma,i} \in \mathbb{R}^3$ denotes the vector of nodal coordinates of size $3m_\Gamma$ as the design variable vector; J, $\boldsymbol{g}$ and $\boldsymbol{h}$ respectively denote the objective function to be minimized, the vector of in-equality constraints and the vector of equality constraints. Here, Γ_g and Γ_h are the subsets of the boundary Γ which are subject to in-equality no-penetration constraints $\boldsymbol{g}_\Gamma(\boldsymbol{X}_{\Gamma_g})$ and equality slip constraints $\boldsymbol{h}_\Gamma(\boldsymbol{X}_{\Gamma_h})$, respectively. Also, $\boldsymbol{r}$ denotes the residual of the state/primal governing equations which may be non-linear, and $\boldsymbol{Q}$ denotes the state/primal vector. The geometric shape constraints are formulated point-wise, which means for each mesh point $\boldsymbol{X}_{\Gamma,j}$, scalar-valued constraints g_j and h_j are evaluated and satisfied separately.

As a matter of fact, the intended geometrical constraints can be formulated as follows:

$$\begin{aligned}
& h_{\Gamma_h,j}(\boldsymbol{X}_{\Gamma_h,j}) = \boldsymbol{D}_j \cdot \Delta\boldsymbol{X}_{\Gamma_h,j} = 0 \\
& g_{\Gamma_g,j}(\boldsymbol{X}_{\Gamma_g,j}) = -\boldsymbol{n}_c \cdot [\boldsymbol{X}_{\Gamma_g,j} - \boldsymbol{X}_{cpp}(\boldsymbol{X}_{\Gamma_g,j})] \leq 0
\end{aligned} \tag{2.2}$$

where the so-called gap function [49, 68] is used to formulate the no-penetration constraint $g_{\Gamma_g,j}$, $\boldsymbol{D}_j$ and $\Delta\boldsymbol{X}_{\Gamma_h,j}$ are, respectively, the damping direction (e.g. $\boldsymbol{D}_j = \boldsymbol{n}_{\Gamma_h,j}$) and the shape update at the jth node. $\boldsymbol{X}_{cpp}$ is determined based on the so-called closest point projection (CPP) of $\boldsymbol{X}_{\Gamma_g,j}$

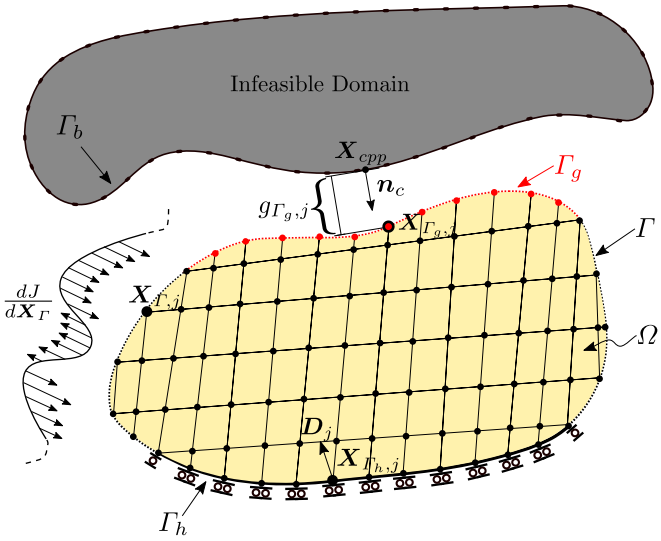


Figure 2.1: A schematic of geometrically constrained shape optimization problem including assessment of the no-penetration condition in two-dimensional space.

onto the bounding surface and $\boldsymbol{n}_c$ is the corresponding contact normal vector. Here, $\boldsymbol{n}_c$ is chosen to be the unit normal vector on the bounding surface at $\boldsymbol{X}_{cpp}$. For an overview of the CPP, including potential pitfalls, see, for example, [48, 49]. However, an approximate solution for finding $\boldsymbol{X}_{cpp}$ is the application of the kd-tree nearest neighbor search to the discrete bounding surface for $\boldsymbol{X}_{\Gamma_g,j}$.

Since the no-penetration condition is required to be fulfilled completely in each optimization iteration, the fixed-radius nearest neighbor search is used here. Therefore, the task is to find the closest point to $\boldsymbol{X}_{\Gamma_g,j}$ in the discrete bounding surface within a defined radius from $\boldsymbol{X}_{\Gamma_g,j}$. The radius can be fixed during the optimization process or dynamically adjusted according to the maximum shape update. As a result, a buffer layer next to the bounding surface/curve is introduced to prevent the potential penetration in each design iteration.

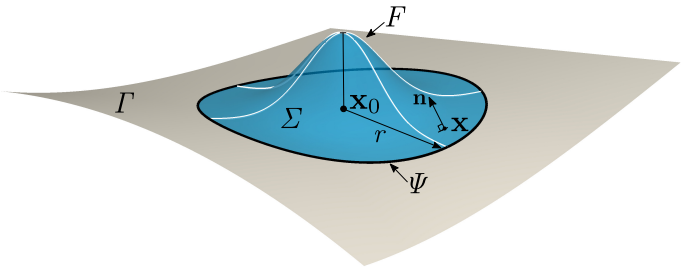


Figure 2.2: Notional schematic of the design surface (Γ), the filter function (F), the integration area (Σ) and the boundary of Σ (Ψ).

2.2 Vertex Morphing Technique

This section presents the implementation and computational details of our shape parametrization technique for numerical optimization. The core idea is to transfer consistently shape optimization problem from the spatial design space $\boldsymbol{X}_\Gamma$ (the geometry space) to a new design space $\boldsymbol{s}_\Gamma$ using a desirable map. This means that the mathematical optimization problem is solved in the new design space.

In a continuous manner, the three-dimensional geometry at point $\boldsymbol{X}_0 = (X_0^1, X_0^2, X_0^3)$ of the optimization surface Γ is generated from the surface control field $\boldsymbol{s} = (s^1, s^2, s^3)$ via a smoothing filter operation:

$$\boldsymbol{X}_0 = \int_{\Sigma(\boldsymbol{s},r)} F(\boldsymbol{X},\boldsymbol{X}_0)\, \boldsymbol{s} \otimes \boldsymbol{n} \cdot d\boldsymbol{\Sigma} = \int_{\Sigma(\boldsymbol{s},r)} F(\boldsymbol{X},\boldsymbol{X}_0)\, \boldsymbol{s}\, d\Sigma \tag{2.3a}$$

$$F(\boldsymbol{X},\boldsymbol{X}_0) = \frac{1}{\sqrt{2\pi}\, r}\, e^{-\frac{\left\|\boldsymbol{X}-\boldsymbol{X}_0\right\|^2}{2\, r^2}} \tag{2.3b}$$

where F could be any reasonable filter (kernel) function, here motivated by the probability density function; Σ is the portion of Γ which lies within a sphere of radius r and center $\boldsymbol{X}_0$; r is the filter radius (assumed to be constant); $\left\|\boldsymbol{X}-\boldsymbol{X}_0\right\|$ is the Euclidean distance to the center of the filter $\boldsymbol{X}_0$; $\boldsymbol{n}$ is the unit normal vector to the surface; $d\boldsymbol{\Sigma}=\boldsymbol{n}\,d\Sigma$ is the unit normal component of the surface element. See Fig. 2.2 for a notional schematic of the used notation. Note that, the subscript Γ is dropped to simplify the notation and $\boldsymbol{X}$ is the continuous 3D vector with one component for each spatial coordinate, i.e., $\boldsymbol{X}=(X^1,X^2,X^3)$. By selecting appropriate F and r, one can control the properties of this mapping and, subsequently, the produced shapes. Therefore, F and r can be regarded as design handles.

As seen in Eq. 2.3, Vertex Morphing constructs a nonlinear-mapping that establishes correspondence between the control space where the optimization problem is formulated and the geometry space where the shape update is applied. In order to enforce the consistency in the mapping [90], the following condition is applied:

$$\int_{\Sigma} F(\boldsymbol{X},\boldsymbol{X}_0)\; d\Sigma = 1.0 \tag{2.4}$$

Consistency is the basic criterion for mapping algorithms which specifies that a constant variation in the control field $\delta\boldsymbol{s}=(1,1,1)$ results in a uniform update in the geometry $\delta\boldsymbol{X}_0=(1,1,1)$ in this context.

Two key components in any gradient-based shape optimization method are the geometry update ($\Delta\boldsymbol{X}_0$) and the objective function gradient with respect to (w.r.t.) the design variables ($\frac{dJ}{d\boldsymbol{s}}$). Since derivatives of the objective function are always computed w.r.t to the spatial coordinates $\frac{dJ}{d\boldsymbol{X_0}}$, it is required to compute the shape derivative, i.e., the derivative of the surface coordinates w.r.t. control field $\frac{d\boldsymbol{X}_0}{d\boldsymbol{s}}$. From Eq. 2.3 one can determine the continuous derivative of the shape $\boldsymbol{X}_0$ with respect to the design parameter $\boldsymbol{s}$. For that purpose, the Leibniz theorem has to be applied to differentiate the surface integral in Eq. 2.3, for the ith component of point $\boldsymbol{X}_0$ the shape

derivative reads as follows:

$$\frac{dX_0^i}{ds^i} = \int_{\Sigma} \left(\boldsymbol{F}^i + \underline{\nabla \cdot (\boldsymbol{F}^i \otimes \boldsymbol{v}^{VM,i}) - (\nabla \cdot \boldsymbol{v}^{VM,i}) \boldsymbol{F}^i} \right) \cdot d\boldsymbol{\Sigma} - \oint_{\Psi} \left[\boldsymbol{v}^{VM,i} \times \boldsymbol{F}^i \right] \cdot d\boldsymbol{\Psi}, \quad i \in [1,3] \tag{2.5a}$$

$$\boldsymbol{F}^i = F s^i \boldsymbol{n} \tag{2.5b}$$

$$\boldsymbol{v}^{VM,i} = \frac{d\boldsymbol{X}}{ds^i} \tag{2.5c}$$

where Ψ is the edge or boundary of the domain Σ (see Fig. 2.2). The line integral in the above equations can be safely neglected due to the fact that the filter function F is by definition zero along the boundary Ψ. It should be noted that the left and right sides of Eq. 2.5a are dependent of each other, and, in fact, a partial differential equation (PDE) with an appropriate set of boundary conditions should be solved to obtain the shape derivatives for Vertex Morphing shape parametrization technique. The underlined terms in Eq. 2.5a have been neglected so far in the literature [8, 14, 15, 39], meaning that only the partial variation of the shape with respect to the control has been taken into account, i.e.,

$$\frac{dX_0^i}{ds^i} \approx \frac{\partial X_0^i}{\partial s^i} = \int_{\Sigma} F(\boldsymbol{X}, \boldsymbol{X}_0)\, d\Sigma \tag{2.6}$$

Note again that in the foregoing expressions for brevity the subscript Γ is dropped. One can choose an arbitrary number of design variables to discretize the design space. Unless noted otherwise, the control field is discretized with the same discretization as the geometry to have the largest design space possible for the optimization. Discretizing the governing equation of Vertex Morphing in Eq. 2.3 leads to the following nonlinear system of equations:

$$\boldsymbol{r}^{VM}(\boldsymbol{X}_\Gamma(\boldsymbol{s}), \boldsymbol{s}) = \boldsymbol{X}_\Gamma - \boldsymbol{A}(\boldsymbol{X}_\Gamma) \,.\, \boldsymbol{s}_\Gamma = \boldsymbol{0} \tag{2.7}$$

where $\boldsymbol{r}^{VM}$ is the residual vector of Vertex Morphing parametrization, $\boldsymbol{X}_\Gamma = [\boldsymbol{X}_{\Gamma,1}, \boldsymbol{X}_{\Gamma,2}, ..., \boldsymbol{X}_{\Gamma,m}]$, $\boldsymbol{X}_{\Gamma,i} = (X^1_{\Gamma,i}, X^2_{\Gamma,i}, X^3_{\Gamma,i}) \in \mathbb{R}^3$ denotes the vector of nodal coordinates of size $3m$, $\boldsymbol{s}_\Gamma = [\boldsymbol{s}_{\Gamma,1}, ..., \boldsymbol{s}_{\Gamma,m}]$, $\boldsymbol{s}_{\Gamma,i} = (s^1_{\Gamma,i}, s^2_{\Gamma,i}, s^3_{\Gamma,i})$

$\in \mathbb{R}^3$ is the vector of control variables, $\boldsymbol{A}$ is the filtering matrix which is constructed based on the geometry $\boldsymbol{X}_\Gamma$ and projects the control field onto the geometry. The property of consistency in Eq. 2.4 is reflected as the unit row sum in the operator $\boldsymbol{A}$.

In the context of gradient-based shape optimization, depending on the parametrization technique, one has to derive formulas/rules for the geometry update as well as for the computation of gradients. The governing equations of VM (Eqs. 2.3, 2.7) are nonlinear with respect to the geometry, therefore a suitable solution strategy is needed to compute the updated geometry from the updated control variables. Given the control field and the objective gradients at the kth optimization iteration, the new design may be calculated as:

$$\begin{aligned} {}^{k+1}\boldsymbol{s}_\Gamma &= {}^{k}\boldsymbol{s}_\Gamma + {}^{k}\Delta\boldsymbol{s}_\Gamma \\ &= {}^{k}\boldsymbol{s}_\Gamma + \alpha\, {}^{k}\nabla_{\boldsymbol{s}_\Gamma} J \end{aligned} \tag{2.8}$$

Here α is the optimization step length. Having the new control field ${}^{k+1}\boldsymbol{s}_\Gamma$, Newton methods can be used to compute the new discrete geometry ${}^{k+1}\boldsymbol{X}_\Gamma$ iteratively until convergence as:

$$\begin{aligned} {}^{k+1,n+1}\boldsymbol{X}_\Gamma &= {}^{k+1,n}\boldsymbol{X}_\Gamma + {}^{k+1,n}\Delta\boldsymbol{X}_\Gamma \\ {}^{k+1,n}\Delta\boldsymbol{X}_\Gamma &= -\left[\frac{\partial\, {}^{k+1,n}\boldsymbol{r}^{VM}}{\partial\, {}^{k+1,n}\boldsymbol{X}_\Gamma}\right]^{-1} \cdot {}^{k+1,n}\boldsymbol{r}^{VM} \end{aligned} \tag{2.9}$$

herein, k and n are the current optimization and Newton iteration indices, $\frac{\partial\, {}^{k+1,n}\boldsymbol{r}^{VM}}{\partial\, {}^{k+1,n}\boldsymbol{X}_\Gamma}$ is the Jacobian or tangent stiffness matrix of the Vertex Morphing. Taking the total derivative of Eq. 2.7 with respect to the design geometry, one gets

$$\frac{\partial\, {}^{k+1,n}\boldsymbol{r}^{VM}}{\partial\, {}^{k+1,n}\boldsymbol{X}_\Gamma} = \boldsymbol{I} - \frac{\partial\, {}^{k+1,n}\boldsymbol{A}}{\partial\, {}^{k+1,n}\boldsymbol{X}_\Gamma} \cdot {}^{k+1}\boldsymbol{s}_\Gamma \tag{2.10}$$

where $\boldsymbol{I}$ is an identity matrix of size $3m_\Gamma \times 3m_\Gamma$. A simplification, in order to avoid computationally expensive process of determining the shape update, is to use the incomplete linearization of the VM residuals as follows:

$${}^{k+1,n}\Delta\boldsymbol{X}_\Gamma \approx -\boldsymbol{I} \cdot {}^{k+1,n}\boldsymbol{r}^{VM} \tag{2.11}$$

By assuming that the filtering matrix $\boldsymbol{A}$ is invariant with respect to the new geometry, equations 2.9 can be further simplified as:

$$^{k}\Delta \boldsymbol{X}_{\Gamma} \approx {}^{k}\boldsymbol{A}({}^{k}\boldsymbol{X}_{\Gamma}) \cdot {}^{k}\Delta \boldsymbol{s}_{\Gamma} \tag{2.12}$$

The above has appeared so far as the shape update rule in [8, 14, 15, 39]. Careful comparison of Eq. 2.12 with Eq. 2.9 reveals that the simplified update rule does not require neither the inversion of the VM Jacobian nor the control field itself, rather it requires only the control field update.

According to the chain rule of differentiation, the objective derivative with respect to the shape controls reads:

$$\frac{dJ}{d^{k}\boldsymbol{s}_{\Gamma}} = \frac{dJ}{d^{k}\boldsymbol{X}_{\Gamma}} \cdot \frac{d^{k}\boldsymbol{X}_{\Gamma}}{d^{k}\boldsymbol{s}_{\Gamma}} \tag{2.13}$$

where $\dfrac{dJ}{d^{k}\boldsymbol{X}_{\Gamma}}$ and $\dfrac{d^{k}\boldsymbol{X}_{\Gamma}}{d^{k}\boldsymbol{s}_{\Gamma}}$ are the objective derivatives with respect to mesh coordinates and shape derivatives of Vertex Morphing, respectively. The former is computed by adjoint methods in this case, and the latter can be computed by differentiating the residual vector in Eq. 2.7 with respect to the control vector:

$$\frac{d^{k}\boldsymbol{r}^{VM}}{d^{k}\boldsymbol{s}_{\Gamma}} = \frac{d^{k}\boldsymbol{X}_{\Gamma}}{d^{k}\boldsymbol{s}_{\Gamma}} - \frac{\partial^{k}\boldsymbol{A}}{\partial^{k}\boldsymbol{X}_{\Gamma}} \cdot {}^{k}\boldsymbol{s}_{\Gamma} \cdot \frac{d^{k}\boldsymbol{X}_{\Gamma}}{d^{k}\boldsymbol{s}_{\Gamma}} - {}^{k}\boldsymbol{A} = \boldsymbol{0} \tag{2.14}$$

and then solving the above for the shape gradients leads to:

$$\frac{d^{k}\boldsymbol{X}_{\Gamma}}{d^{k}\boldsymbol{s}_{\Gamma}} = \left[\boldsymbol{I} - \frac{\partial^{k}\boldsymbol{A}}{\partial^{k}\boldsymbol{X}_{\Gamma}} \cdot {}^{k}\boldsymbol{s}_{\Gamma}\right]^{-1} \cdot {}^{k}\boldsymbol{A} \tag{2.15}$$

Under the assumption that the mapping matrix does not change much during shape optimization, the shape gradient reduces to the following known expression [8, 14, 15, 39]:

$$\frac{d^{k}\boldsymbol{X}_{\Gamma}}{d^{k}\boldsymbol{s}_{\Gamma}} \approx {}^{k}\boldsymbol{A} \tag{2.16}$$

Comparing this expression with Eq. 2.15, it is observed that, as expected, the approximated shape derivatives avoid not only matrix inverse but also the need to compute the control field $\boldsymbol{s}_{\Gamma}$. An important observation is

that the combination of the approximated shape update rule Eq. 2.12 and the approximated shape derivatives Eq. 2.16 has the desired property of conserving the objective value improvement as follows:

$$\frac{dJ}{d^k\boldsymbol{X}_\Gamma} \cdot {}^k\Delta\boldsymbol{X}_\Gamma = \frac{dJ}{d^k\boldsymbol{X}_\Gamma} \cdot {}^k\boldsymbol{A} \cdot {}^k\Delta\boldsymbol{s}_\Gamma = \frac{dJ}{d^k\boldsymbol{s}_\Gamma} \cdot {}^k\Delta\boldsymbol{s}_\Gamma \tag{2.17}$$

where the left hand side is equal to the change in the objective due to the geometry update and the right hand side is equal to the change in the objective due to the control update. This equivalence can be seen as a proof of the consistency between the problem formulation in the geometry space and the problem formulation in the control space.

In the remainder of this thesis, wherever Vertex Morphing shape parametrization is used, the filtering matrix $\boldsymbol{A}$ itself will be used for the mapping of the variation in the control parameters ${}^k\Delta\boldsymbol{s}_\Gamma$ to find the geometry update ${}^k\Delta\boldsymbol{X}_\Gamma$, and for the mapping of the nodal sensitivities from the geometry space to the design control space using a transpose operation.

Finally, it is worth to mention that Vertex Morphing can preserve the main features of the design surface. More precisely, it allows shape changes which are not affecting the “design character”, aesthetic and geometrical features. As mentioned previously, F and r are the design handles which control the feature preservation. A good example is the preservation of feature lines (sharp edges), which could be achieved by a proper choice of radius r [38]. The reason for this is that all the features smaller than the radius r are only subject to a bulk and rigid motion, without considerable shape deformation. Another example would be the thickness control which results in the no self-penetration property of the design surface during optimization. In this case, the wall thickness is seen as a design feature and the surface points which are close to each other move together, no matter to which side of the wall they belong. For details, the interested reader is referred to [38].

2.3 Analogy of VM to CAD-based Parametrization

It is interesting to compare carefully Vertex Morphing as a consistent node-based shape parametrization technique against CAD-based methods as

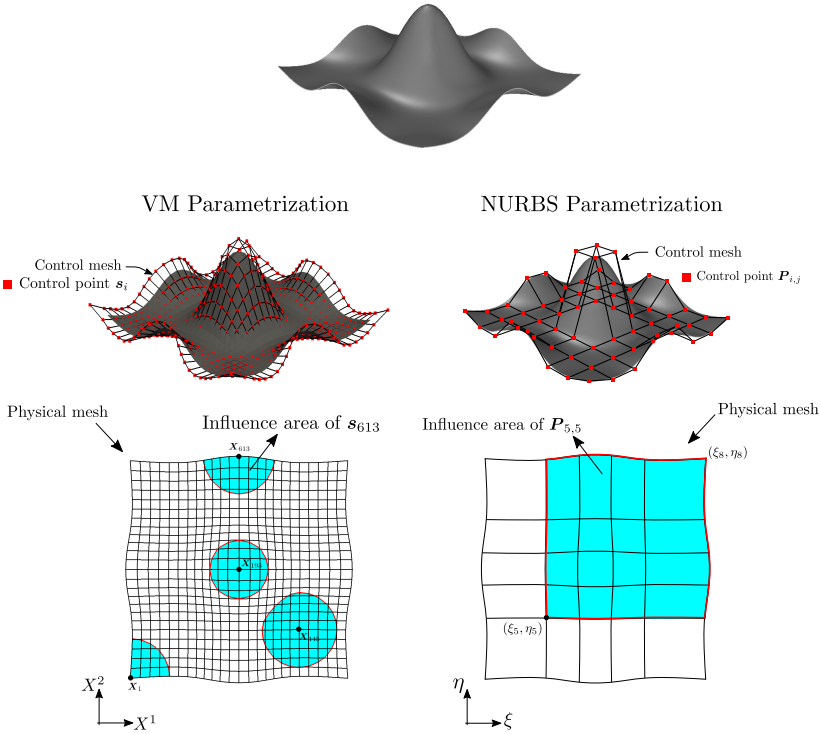


Figure 2.3: VM (a node-based parametrization) vs. NURBS (a CAD-based parametrization). The VM filter function is of order 2 and the NURBS basis function is a third order polynomial.

actual industrial standard for shape optimization. Principally, CAD-based parametrization for a surface geometry consists of four main parts [17, 67] (see also Fig. 2.3):

- *Control point set*: It is a set of Cartesian coordinates which determine the shape. In general, control point $\boldsymbol{P}_{i,j}$ does not interpolate the actual geometry but each point on the geometry is computed by a weighted sum of a set of control points.

- *Control mesh*: It is basically the connectivity between the control points. In other words, control mesh interpolates the control points. Due to the non-interpolatory property of the control points, the control mesh also does not lie on the actual geometry.
- *Physical mesh*: It is a structured discretization of the actual geometry. Knot vectors $\boldsymbol{\Xi}$ and $\boldsymbol{H}$ define the resolution of the physical mesh in the curvilinear directions ξ, η. By definition, control and physical meshes require different resolutions.
- *Basis function*: Each control point is assigned a polynomial function (basis function) which specifies the physical mesh intervals in which the corresponding control point is active. A basis function is usually denoted by $N_{i,p}$ in which i corresponds to the ith control point, and p corresponds with the degree of the basis function.

In NURBS-based parametrization, the three-dimensional geometry at point $\boldsymbol{X}_0 = (X_0^1, X_0^2, X_0^3)$ of the optimization surface Γ is computed as follows:

$$\boldsymbol{X}_0(\xi,\eta) = \frac{\sum_{i=1}^{k}\sum_{j=1}^{l} N_{i,p}(\xi) N_{j,q}(\eta) w_{i,j} \boldsymbol{P}_{i,j}}{\sum_{m=1}^{k}\sum_{n=1}^{l} N_{m,p}(\xi) N_{n,q}(\eta) w_{m,n}}, \quad 0 \leqslant \xi, \eta \leqslant 1 \tag{2.18}$$

where $\boldsymbol{P}_{i,j}$ and $w_{i,j}$ are control points and the corresponding weights, $N_{i,p}(\xi)$ and $N_{j,q}(\eta)$ are p-th and q-th degree B-spline basis functions defined as:

$$p = 0,$$

$$N_{i,0}(\xi) = \begin{cases} 1, & \xi_i \leqslant \xi < \xi_{i+1} \\ 0, & \text{otherwise} \end{cases}$$

$$p \geqslant 1,$$

$$N_{i,p}(\xi) = \begin{cases} \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi), & \xi_i \leqslant \xi < \xi_{i+p+1} \\ 0, & \text{otherwise} \end{cases}$$

$$\boldsymbol{\Xi} = \{\underbrace{0,\ldots,0}_{p+1},\ldots,\xi_i,\ldots,\underbrace{1,\ldots,1}_{p+1}\}_{k+p+1}$$

(2.19)

respectively,

$$
\begin{aligned}
&q = 0,\\
&\quad N_{j,0}(\eta) = \begin{cases} 1, & \eta_j \leqslant \eta < \eta_{j+1} \\ 0, & \text{otherwise} \end{cases}\\
&q \geqslant 1,\\
&\quad N_{j,q}(\xi) = \begin{cases} \frac{\eta-\eta_j}{\eta_{j+q}-\eta_j} N_{j,q-1}(\eta) + \frac{\eta_{j+q+1}-\eta}{\eta_{j+q+1}-\eta_{j+1}} N_{j+1,q-1}(\eta), & \eta_j \leqslant \eta < \eta_{j+q+1} \\ 0, & \text{otherwise} \end{cases}\\
&\quad \boldsymbol{H} = \{\underbrace{0,\ldots,0}_{q+1},\ldots,\eta_j,\ldots,\underbrace{1,\ldots,1}_{q+1}\}_{l+q+1}
\end{aligned}
\tag{2.20}
$$

As illustrated in Fig. 2.3, each control point $\boldsymbol{P}_{i,j}$ has the influence area determined by the assigned basis functions, i.e. $N_{i,p}$ and $N_{j,q}$. The basis functions are equal to zero everywhere except for the interval $[\xi_i, \xi_{i+p+1}) \times [\eta_j, \eta_{j+q+1})$, defining the area of influence of $\boldsymbol{P}_{i,j}$. For ease of comparison against VM, the surface equation 2.18 can be cast into the compact form:

$$\boldsymbol{X}_0(\xi,\eta) = \sum_{i=1}^{k}\sum_{j=1}^{l} R_{i,j}(\xi,\eta)\boldsymbol{P}_{i,j} \tag{2.21a}$$

$$R_{i,j}(\xi,\eta) = \frac{N_{i,p}(\xi)N_{j,q}(\eta)w_{i,j}}{\sum_{m=1}^{k}\sum_{n=1}^{l} N_{m,p}(\xi)N_{n,q}(\eta)w_{m,n}} \tag{2.21b}$$

where $R_{i,j}(\xi,\eta)$ are so-called rational basis functions.
Now reconsider the geometry generation rule of VM in Eq. 2.3, where the control space is discretized by m control points, and the Gaussian filter function F is replaced with a polynomial function of order p:

$$\boldsymbol{X}_0 = \sum_{i=1}^{m} R_i(\boldsymbol{X}_0, \boldsymbol{X}_i)\boldsymbol{s}_i \tag{2.22a}$$

$$R_i(\boldsymbol{X}_0, \boldsymbol{X}_i) = \frac{F_{i,p+1}(\boldsymbol{X}_0, \boldsymbol{X}_i)}{\sum_{j=1}^{m} F_{j,p+1}(\boldsymbol{X}_0, \boldsymbol{X}_j)} \tag{2.22b}$$

$$F_{i,p+1}(\boldsymbol{X}_0,\boldsymbol{X}_i)=\begin{cases}\int F_{i,p}\,d\Gamma=\int\left(1-\frac{\|\boldsymbol{X}_i-\boldsymbol{X}_0\|}{r}\right)^p d\Gamma\approx\alpha\left(1-\frac{\|\boldsymbol{X}_i-\boldsymbol{X}_0\|}{r}\right)^{p+1}, & \|\boldsymbol{X}_i-\boldsymbol{X}_0\|\leqslant r\\ 0, & \text{otherwise}\end{cases} \tag{2.22c}$$

As is evident from Fig. 2.3, like the NURBS-based parametrization, each control point $\boldsymbol{s}_i$ has the influence area determined by the assigned function, i.e. $F_{i,p}$. The filter function is equal to zero everywhere except for the portion of Γ which lies within a sphere of radius r and center $\boldsymbol{X}_i$, defining the area of influence of $\boldsymbol{s}_i$. In Eq. 2.22c, note carefully that, due to the surface integral in the VM's geometry definition (Eqs. 2.3), the polynomial-based filter function of order p, i.e. $F_{i,p}$, results in a higher order polynomial-based shape function, i.e. $F_{i,p+1}$.

Number of control points is one of important differences between the presented shape parametrization techniques. While VM-based parametrization provides the richest design space for optimization, NURBS-based one usually describes the surface with a limited number of control points, unless refinement is made. However consistency between design and analysis in CAD-based optimization is considered to be a great advantage.

2.4 Shape Optimization in the Control Space

By knowing the mapping from the control space to the geometry space, one can formulate the shape optimization problem with geometric shape constraints in the control space. Mathematically the problem reads as follows:

$$\begin{aligned}
\min_{\boldsymbol{s}_\Gamma}\quad & J(\boldsymbol{X}_\Gamma(\boldsymbol{s}_\Gamma),\boldsymbol{Q}(\boldsymbol{X}_\Gamma))\\
\text{subject to}\quad & g_j(\boldsymbol{X}_\Gamma(\boldsymbol{s}_\Gamma),\boldsymbol{Q}(\boldsymbol{X}_\Gamma))\leq 0,\quad j=1,\ldots,m_g.\\
& g_{\Gamma,j}(\boldsymbol{X}_{\Gamma_g,j})\in\boldsymbol{g},\quad j=1,\ldots,m_{\Gamma_g}.\\
& h_j(\boldsymbol{X}_\Gamma(\boldsymbol{s}_\Gamma),\boldsymbol{Q}(\boldsymbol{X}_\Gamma))=0,\quad j=1,\ldots,m_h.\\
& h_{\Gamma,j}(\boldsymbol{X}_{\Gamma_h,j})\in\boldsymbol{h},\quad j=1,\ldots,m_{\Gamma_h}.\\
& \boldsymbol{r}_j(\boldsymbol{X}_\Gamma(\boldsymbol{s}_\Gamma),\boldsymbol{Q}(\boldsymbol{X}_\Gamma))=\boldsymbol{0},\quad j=1,\ldots,m_\Omega+m_\Gamma.
\end{aligned} \tag{2.23}$$

Following gradient-based optimization procedure, the gradients of the objective or constraint functions w.r.t. the design variables $\boldsymbol{s}_\Gamma$ can be

obtained from Eqs. 2.13,2.16. As the optimization problem is solved in several consecutive iterations, the new design in each iteration can be calculated using:

$$^{k+1}\boldsymbol{s}_\Gamma = {}^{k}\boldsymbol{s}_\Gamma + \alpha\, {}^{k}\boldsymbol{p} \tag{2.24}$$

where k is the current optimization iteration number; ${}^{k}\boldsymbol{s}_\Gamma$ is the current design vector; ${}^{k+1}\boldsymbol{s}_\Gamma$ is the new design vector which is unknown; α is the step length and can be determined using the so-called line search; ${}^{k}\boldsymbol{p}$ is the feasible search direction here, the steepest-descent direction respecting the constraints. One has to note that the constraints in Eq. 2.23 can be divided broadly into two types viz. geometric and state constraints.

In large-scale optimal shape design of complex geometries, it is necessary to fulfil the geometric and state constraint at each optimization step while satisfying both surface smoothness and the mesh regularity criteria. The state constraints have been classically and strongly enforced to the shape optimization problems using the adjoint method, which is discussed in detail in Chapter 4. To fulfil the geometric constraints, one can employ the successful Rosen's gradient projection method [34, Chapter 5], [71, 72], which projects the descent direction $\nabla_{\boldsymbol{s}} J$ onto the subspace tangent to the active inequality and equality constraints. To do this, it is required to linearize the active inequality and equality constraints about the current design ${}^{k}\boldsymbol{s}_\Gamma$ (remember that the optimization problem is defined in the control space):

$$\begin{bmatrix} {}^{k}\boldsymbol{g}_a \\ \boldsymbol{h} \end{bmatrix}_{(m_{g_a}+m_h)\times 1} \approx \begin{bmatrix} \frac{d^{k}\boldsymbol{g}_a}{d^{k}\boldsymbol{s}_\Gamma} \\ \frac{d\boldsymbol{h}}{d^{k}\boldsymbol{s}_\Gamma} \end{bmatrix} \cdot {}^{k}\boldsymbol{s}_\Gamma - {}^{k}\boldsymbol{b} = {}^{k}\boldsymbol{C}_m \cdot {}^{k}\boldsymbol{s}_\Gamma - {}^{k}\boldsymbol{b} \approx \boldsymbol{0} \tag{2.25}$$

where at the kth iteration ${}^{k}\boldsymbol{g}_a$ is the vector of the active inequality constraints, m_{g_a} is the number of active points on the design surface, ${}^{k}\boldsymbol{b}$ is a constant vector, ${}^{k}\boldsymbol{C}_m$ is the Jacobian of the linearized constraints with respect to design variables. The major task is the calculation of ${}^{k}\boldsymbol{C}_m$. This can be done by linearizing Eq. 2.2 about the current geometry ${}^{k}\boldsymbol{X}_\Gamma$ and then mapping the Jacobian matrix to the design space using matrix ${}^{k}\boldsymbol{A}$:

$$ {}^{k}\boldsymbol{C}_{m} = \begin{bmatrix} \frac{d^{k}\boldsymbol{g}_{a}}{d^{k}\boldsymbol{X}_{\Gamma}} \\ \\ \frac{d\boldsymbol{h}}{d^{k}\boldsymbol{X}_{\Gamma}} \end{bmatrix} \cdot \frac{d({}^{k}\boldsymbol{X}_{\Gamma})}{d({}^{k}\boldsymbol{s}_{\Gamma})} = {}^{k}\boldsymbol{C} \,.\, {}^{k}\boldsymbol{A} \tag{2.26} $$

where ${}^{k}\boldsymbol{C}$ is the Jacobian of the linearized constraints with respect to the nodal coordinates. In an iterative solution, a feasible design is of interest; this means that both designs ${}^{k}\boldsymbol{s}$ and ${}^{k+1}\boldsymbol{s}$ satisfy Eq. 2.25:

$$ {}^{k}\boldsymbol{C}_{m} \cdot {}^{k}\boldsymbol{p} = 0 \tag{2.27} $$

where ${}^{k}\boldsymbol{p}$ is the unknown feasible search direction. Then, finding a steepest descent direction satisfying Eq. 2.27 can be a solution to the following optimization problem:

$$ \begin{aligned} \min_{{}^{k}\boldsymbol{p}} \quad & ({}^{k}\boldsymbol{p})^{T}.\nabla_{{}^{k}\boldsymbol{s}} J \\ \text{subject to} \quad & {}^{k}\boldsymbol{C}_{m}.{}^{k}\boldsymbol{p} = 0 \\ & ({}^{k}\boldsymbol{p})^{T} \,.\, {}^{k}\boldsymbol{p} = 1 \end{aligned} \tag{2.28} $$

where $\nabla_{{}^{k}\boldsymbol{s}} J$ is computed from Eqs. 2.13,2.16. Note that $\nabla_{{}^{k}\boldsymbol{X}_{\Gamma}} J$ is computed by the adjoint method therefore it fulfils the state constraints, i.e., ${}^{k}\boldsymbol{r}({}^{k}\boldsymbol{X}_{\Gamma}, {}^{k}\boldsymbol{Q}({}^{k}\boldsymbol{X}_{\Gamma})) = \boldsymbol{0}$. Simply speaking, ${}^{k}\boldsymbol{p}$ has to be computed such that the most negative directional derivative is achieved while satisfying the constraints. Note that the linear sub-optimization problem in 2.28 is basically the particularization of the Sequential Linear Programming (SLP). Among all constrained optimization methods, the so-called Lagrange function is used herein to transfer the constrained optimization 2.28 into a unconstrained problem by augmenting the objective function with the constraints as follows:

$$ L = ({}^{k}\boldsymbol{p})^{T}.\nabla_{{}^{k}\boldsymbol{s}} J - ({}^{k}\boldsymbol{p})^{T} \cdot ({}^{k}\boldsymbol{C}_{m})^{T} \cdot \boldsymbol{\lambda} - (({}^{k}\boldsymbol{p})^{T} \,.\, {}^{k}\boldsymbol{p} - 1)\mu \tag{2.29} $$

where $\boldsymbol{\lambda}$ and μ are the Lagrange multipliers associated with the given constraints. The solution of the above mentioned constrained optimization problem has to satisfy the necessary Karush-Kuhn-Tucker (KKT) conditions:

$$\begin{aligned}
\frac{dL}{d^k\boldsymbol{p}} &= \nabla_{^k\boldsymbol{s}} J - ({}^k\boldsymbol{C}_m)^T \boldsymbol{\lambda} - 2\mu\, {}^k\boldsymbol{p} = 0 \\
\frac{dL}{d\boldsymbol{\lambda}} &= ({}^k\boldsymbol{p})^T \cdot ({}^k\boldsymbol{C}_m)^T = 0 \\
\frac{dL}{d\mu} &= ({}^k\boldsymbol{p})^T \cdot {}^k\boldsymbol{p} - 1 = 0
\end{aligned} \tag{2.30}$$

By solving this system of equations, we obtain:

$${}^k\boldsymbol{\lambda} = ({}^k\boldsymbol{C}_m \cdot ({}^k\boldsymbol{C}_m)^T)^{-1} \cdot {}^k\boldsymbol{C}_m \cdot \nabla_{^k\boldsymbol{s}} J \tag{2.31}$$

and

$$\begin{aligned}
{}^k\boldsymbol{p} &= \frac{1}{2\mu}\left[\nabla_{^k\boldsymbol{s}} J - ({}^k\boldsymbol{C}_m)^T \cdot ({}^k\boldsymbol{C}_m \cdot ({}^k\boldsymbol{C}_m)^T)^{-1} \cdot {}^k\boldsymbol{C}_m \cdot \nabla_{^k\boldsymbol{s}} J\right] \\
&= \frac{1}{2\mu}\left[\nabla_{^k\boldsymbol{s}} J - ({}^k\boldsymbol{C}_m)^T \cdot {}^k\boldsymbol{\lambda}\right] \\
&= \frac{1}{2\mu}\, \boldsymbol{T}\, \nabla_{^k\boldsymbol{s}} J
\end{aligned} \tag{2.32}$$

where $\boldsymbol{T}$ is the projection matrix. Generally, the factor $\frac{1}{2\mu}$ does not play any role in the search direction; therefore, the feasible search direction for a minimization problem is

$${}^k\boldsymbol{p} = -\boldsymbol{T}\, \nabla_{^k\boldsymbol{s}} J \tag{2.33}$$

Note that Eq. 2.31 requires the solution of a linear system of equations of size $(m_h + m_{g_a}) \times (m_h + m_{g_a})$, where m_h and m_{g_a} are the numbers of equality constraints and active inequality constraints, respectively.

For reasons of clarity and, in particular, brevity, the feasible search direction is computed for the case of a single active constraint, as follows:

$$\begin{aligned}
{}^k\lambda &= \frac{\nabla_{^k\boldsymbol{s}} g_a \cdot \nabla_{^k\boldsymbol{s}} J}{\nabla_{^k\boldsymbol{s}} g_a \cdot \nabla_{^k\boldsymbol{s}} g_a} \\
{}^k\boldsymbol{p} &= \nabla_{^k\boldsymbol{s}} J - {}^k\lambda \nabla_{^k\boldsymbol{s}} g_a = \nabla_{^k\boldsymbol{s}} J - \left(\frac{\nabla_{^k\boldsymbol{s}} g_a \cdot \nabla_{^k\boldsymbol{s}} J}{\left\|\nabla_{^k\boldsymbol{s}} g_a\right\|}\right) \cdot \frac{\nabla_{^k\boldsymbol{s}} g_a}{\left\|\nabla_{^k\boldsymbol{s}} g_a\right\|}
\end{aligned} \tag{2.34}$$

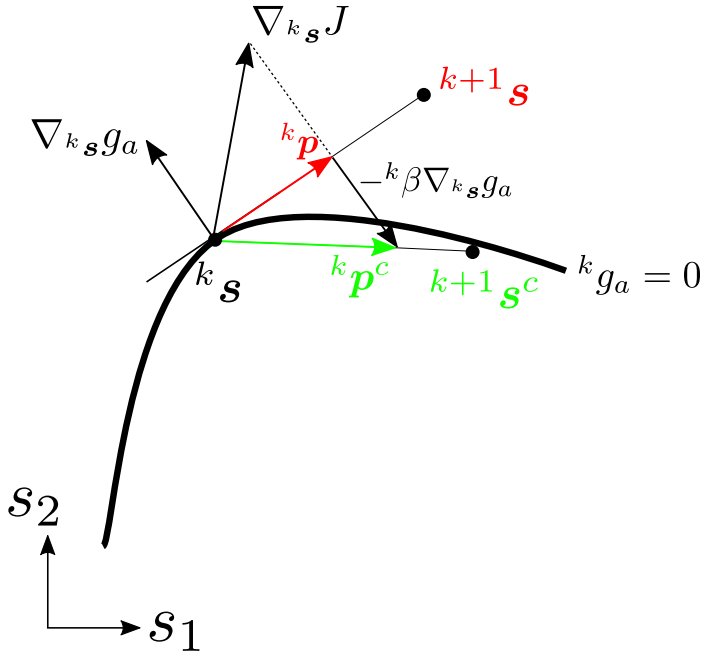


Figure 2.4: Gradient projection and correction.

Figure 2.4 represents the projection step schematically. As can be observed from the figure, if the constraint is highly non-linear with respect to design variables, the update in the tangent subspace, i.e. ${}^{k+1}\boldsymbol{s} - {}^{k}\boldsymbol{s} = \alpha\,{}^{k}\boldsymbol{p}$, no longer follows exactly the constraint boundary. Therefore, a correction move is required to bring the design update back to the feasible domain. The projected descent direction may be corrected as:

$$\begin{aligned} {}^{k}\boldsymbol{p}^{c} &= {}^{k}\boldsymbol{p} - {}^{k}\beta\nabla_{k\boldsymbol{s}} g_a \\ {}^{k}\beta &= {}^{k-1}\beta + \zeta \cdot {}^{k}g_a \end{aligned} \tag{2.35}$$

where ${}^{k}\boldsymbol{p}^{c}$ is the corrected feasible direction, ζ is the penalty parameter and must be chosen large enough but remains of finite value.

Depending on the radius of influence in Eq. 2.3 and the number of design variables, ${}^{k}\boldsymbol{A}$ and, subsequently, ${}^{k}\boldsymbol{C}_m$ can be considerably large and

sparse, which is the case for industrial shape optimization problems. Thus, it makes sense to exploit sparsity by using iterative linear solvers for solving Eq. 2.31. Since iterative solvers are more sensitive to the conditioning of the problem, depending on the properties of the matrix ${}^{k}\boldsymbol{C}_{m}$, special attention has to be given to the choice of a robust and efficient iterative solver. The next section covers this topic in detail. As a conclusion of this section, Algorithm 1 outlines the steps and required computations for constrained shape optimization by means of Vertex Morphing neglecting the non-linearity of the mapping matrix.

2.5 Numerical Analysis of VM Mapping

This section particularly discusses the condition of the system of linear equations in Eq. 2.31 which is directly linked to the condition of $\boldsymbol{C}_m$ and, subsequently, $\boldsymbol{A}$. For simplicity, the superscript k that indicates the optimization iteration number will be dropped from the notation in the upcoming paragraphs. For the analysis purpose, Fig. 2.5 provides an introductory example to verify our formulation and implementation of geometrical constraints. The objective is to minimize the difference between the discrete shape $\boldsymbol{X}$ and the discrete target curve $\boldsymbol{X}^{target}$ inside the feasible domain:

$$J = \sum_{i=1}^{50} \left\| \boldsymbol{X}_i - \boldsymbol{X}_i^{target} \right\| \tag{2.36}$$

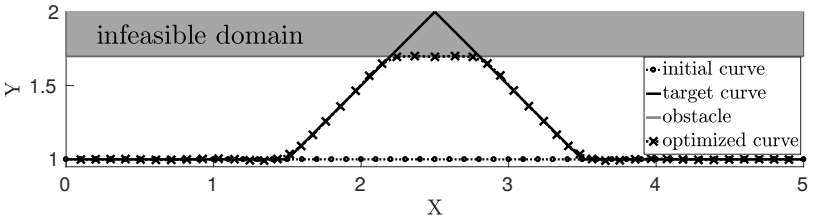


Figure 2.5: Two-dimensional example of a node-based, geometrically constrained shape optimization.

In the figure, the infeasible domain is colored in gray. As expected and can be seen from the figure, the discrete design curve gets very close to the

Algorithm 1 Constrained shape optimization

1: **procedure**
//initialization
2: ${}^0\boldsymbol{X}_\Gamma \leftarrow$ discretize design surface Γ
3: discretize bounding surface Γ_b
4: create kd-trees on Γ and Γ_b
//optimization loop
5: **for** $k = 1, 2, \ldots$ **do**
// solve state problem
6: ${}^k\boldsymbol{r}({}^k\boldsymbol{X}_\Gamma, \boldsymbol{Q}({}^k\boldsymbol{X}_\Gamma)) = \boldsymbol{0}$
// solve adjoint problem
7: $\dfrac{dJ}{d({}^k\boldsymbol{X}_\Gamma)} \leftarrow$ calculate shape gradients of objective
// find active inequality constraints
8: ${}^k g_j \geq 0,\ j = 1, \ldots, m_{g_a}$.
// calculate shape gradients of active constraints
9: ${}^k\boldsymbol{C} \leftarrow \begin{bmatrix} \frac{d^k \boldsymbol{g}_a}{d^k \boldsymbol{X}_\Gamma} \\ \\ \frac{d\boldsymbol{h}}{d^k \boldsymbol{X}_\Gamma} \end{bmatrix}_{(m_h + m_{g_a}) \times 3m_\Gamma}$
// calculate mapping matrix
10: ${}^k\boldsymbol{A} \leftarrow \dfrac{d({}^k\boldsymbol{X}_{\Gamma,i})}{d({}^k\boldsymbol{s}_\Gamma)} = \int_\Sigma F(\boldsymbol{X}, \boldsymbol{X}_i)\, d\Sigma,\ i = 1, \ldots, m.$
// map gradients of objective to control space
11: $\dfrac{dJ}{d({}^k\boldsymbol{s}_\Gamma)} \leftarrow \dfrac{dJ}{d({}^k\boldsymbol{X}_\Gamma)} \cdot {}^k\boldsymbol{A}$
// map gradients of active constraints to control space
12: ${}^k\boldsymbol{C}_m \leftarrow {}^k\boldsymbol{C} \cdot {}^k\boldsymbol{A}$
// solve for Lagrange multipliers
13: minimize $\left\| ({}^k\boldsymbol{C}_m)^T \boldsymbol{\lambda} - \nabla_{{}^k\boldsymbol{s}} J = 0 \right\|_2$
// calculate projected descent direction
14: ${}^k\boldsymbol{p} \leftarrow -\boldsymbol{T}\, \nabla_{{}^k\boldsymbol{s}} J$
// correct projected descent direction
15: ${}^k\boldsymbol{p}^c \leftarrow {}^k\boldsymbol{p} - ({}^k\boldsymbol{C}_m)^T \cdot {}^k\boldsymbol{\beta}, {}^k\boldsymbol{\beta} = {}^{k-1}\boldsymbol{\beta} - \zeta \cdot [{}^k\boldsymbol{g}_a, {}^k\boldsymbol{h}]^T$
//calculate design update
16: ${}^k\Delta\boldsymbol{s} \leftarrow \alpha\, {}^k\boldsymbol{p}^c$
//calculate geometry update
17: ${}^k\Delta\boldsymbol{X}_\Gamma \leftarrow {}^k\boldsymbol{A} \cdot {}^k\Delta\boldsymbol{s}_\Gamma$
18: **check convergence**
19: **end for**
20: **end procedure**

boundary of the feasible domain but does not penetrate the obstacle (possibly overlaps). For this example and the subsequent shape optimization problems, gradient-based shape optimization using the steepest-descent technique with a constant step size is applied.

First we start with analyzing the operator matrix $\boldsymbol{A}$, and then evaluate the condition of the linear system in Eq. 2.31. Remember that F and r are the design handles and they determine the numerical properties of $\boldsymbol{A}$. For example, the degree of sparsity of the matrix is inverse proportional to the radius r. We now reconsider the two-dimensional example shown in Fig. 2.5 and study the 2-norm condition number κ of $\boldsymbol{A}$ for different filter radii and mesh densities.

Fig. 2.6 shows semi-logarithmic diagrams of the dependency of $\kappa(A)$ on the filter radius r and the grid resolution for the first and the last optimization iteration. We can clearly see that the conditioning of $\boldsymbol{A}$ deteriorates as either the filter radius or the mesh density is increased. This is expected since the operator enforces geometrical dependency and smoothness locally; i.e., the denser the mesh in the r neighborhood of a node, the more linear dependencies exist between the rows of matrix $\boldsymbol{A}$. Another important observation is that the final mapping matrix has a better condition number than the initial one. This can be explained by referring to Eq. 2.3 and recognizing the fact that the filtering operator causes linear dependencies between the rows of matrix $\boldsymbol{A}$. In the next step, we investigate the ill-conditioned character of the $\boldsymbol{C}_m$ matrix, which is inherited from matrix $\boldsymbol{A}$. Attention should be drawn to the fact that this character directly affects the conditioning of the following square system of equations (Eq. 2.31):

$$(\boldsymbol{C}_m.\boldsymbol{C}_m^T)\boldsymbol{\lambda} = \boldsymbol{C}_m \nabla_{\boldsymbol{s}} J \tag{2.37}$$

which is known as the system of normal equations associated with the following least squares problem [74]:

$$\text{minimize } \left\| \boldsymbol{C}_m^T \boldsymbol{\lambda} - \nabla_{\boldsymbol{s}} J = 0 \right\|_2 \tag{2.38}$$

Note that Eq. 2.38 is an over-determined system, i.e., $\boldsymbol{C}_m^T$ is a rectangular matrix of size $3m \times n$, where n is the number of active constraints. Also note that if $\boldsymbol{C}_m$ is poorly conditioned, then solving Eq. 2.37 can be a

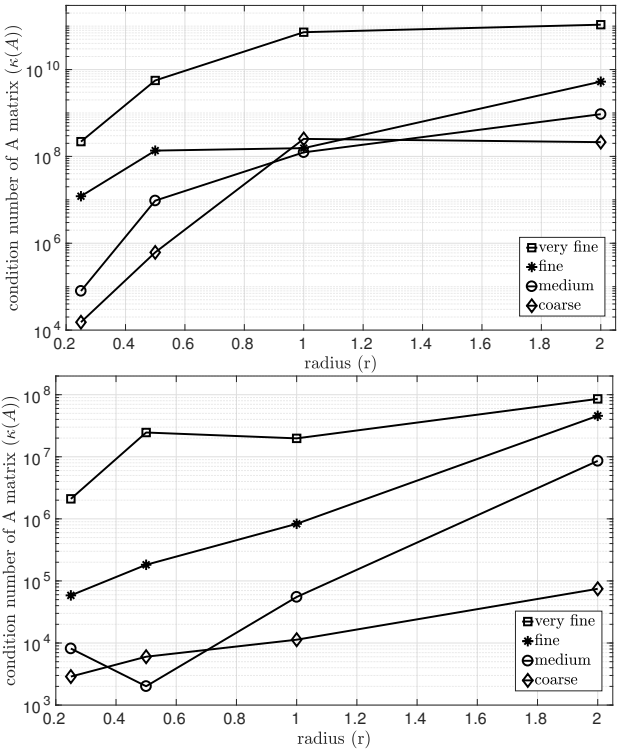


Figure 2.6: Condition number of $\boldsymbol{A}$ vs. filter radius for different mesh resolutions. Top: the first iteration, down: the last iteration.

computationally inefficient approach [74]. In fact, it can be shown that the following relation holds [74]:

$$\kappa(\boldsymbol{C}_m.\boldsymbol{C}_m^T) = \kappa^2(\boldsymbol{C}_m) \tag{2.39}$$

Assume, for example, that $\kappa(\boldsymbol{C}_m) = 1e^8$. Then Eq. 2.39 reads $\kappa(\boldsymbol{C}_m.\boldsymbol{C}_m^T) = 1e^{16}$, leading to numerical problems in the solution of Eq. 2.37.

Fig. 2.7 shows the conditioning of the $\boldsymbol{C}_m$ matrix at different radii and mesh densities. Its behavior is similar to that of matrix $\boldsymbol{A}$. From this, we conclude that both $\boldsymbol{C}_m$ and $\boldsymbol{A}$ are poorly conditioned matrices by nature. Furthermore, the number of the columns in both matrices is equal to $3m$,

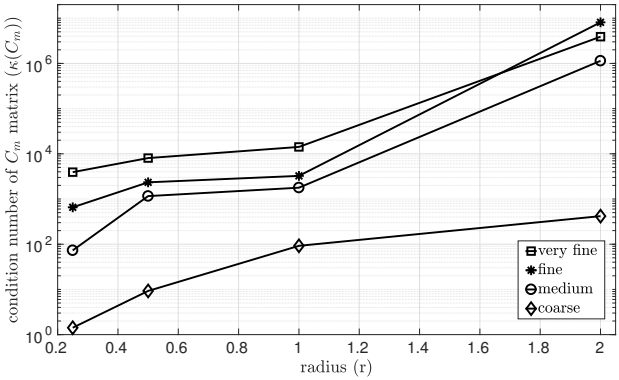


Figure 2.7: Condition number of $\boldsymbol{C}_m$ vs. filter radius for different mesh resolutions.

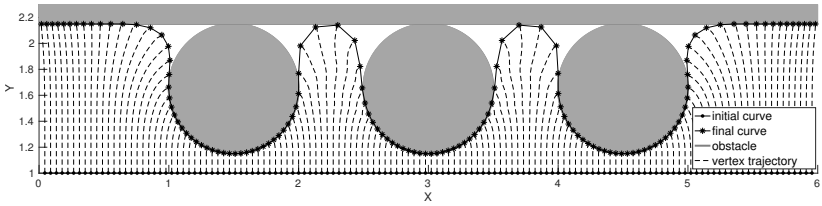


Figure 2.8: Shape update trajectories of a geometrically constrained optimization using Vertex Morphing.

which can be too big owing to the node-based shape parametrization. On the other hand, numerical investigations by Pyzara et al. [69] show that the value of a condition number scales with the size of the matrix. Therefore, the use of standard least squares iterative solvers is highly beneficial and recommended for geometrically constrained shape optimization using Vertex Morphing for industrial applications.

Finally, as a proof-of-concept design case, we study an illustrative example depicted in Fig. 2.8. A uniform gradient field $\nabla_{\boldsymbol{X}} J = (0,-1,0)$ is applied to a discretized horizontal line (design geometry space), resulting in a uniform and smooth translation of the vertices towards the infinity. However, three circles and a horizontal line are put as non-penetrative obstacles

in the way of the vertices in order to geometrically constrain this pseudo optimization problem. In Fig. 2.8, the final shape after many optimization iterations as well as the design trajectory of each vertex are drawn. Clearly, the trajectories respect and satisfy the non-penetration condition over the obstacles, while the design curve stays smooth and regular. An important observation is that not only is the presented treatment of geometrical constraints consistent with Vertex Morphing but Vertex Morphing also provides a consistent frame for node-based shape optimization.

Chapter 3

FSI: Governing Equations and Solution Methods

This chapter starts with mathematical description of stationary fluid-structure interaction problem including a continuous form of the governing equations and an appropriate set of steady coupling conditions at the fluid-structure interface. Since the flow past rotating wind turbine blade is one of the applications of interest, the system of fluid governing equations are also transferred into a non-inertial reference frame that rotates with the blade. This enables us to recover a steady solution for steady-state adjoint-based shape sensitivity analysis. Without loss in generality, the equations and the interface boundary conditions are then discretized and written in discrete residual form. It is important to emphasize that all subsequent derivations are independent of discretization method, e.g. finite-element and finite-volume methods. This is followed by a review of well-known strategies for the solution of the primal/direct FSI problem, including a brief explanation of data mapping between non-matching meshes and geometries in fluid-structure interaction.

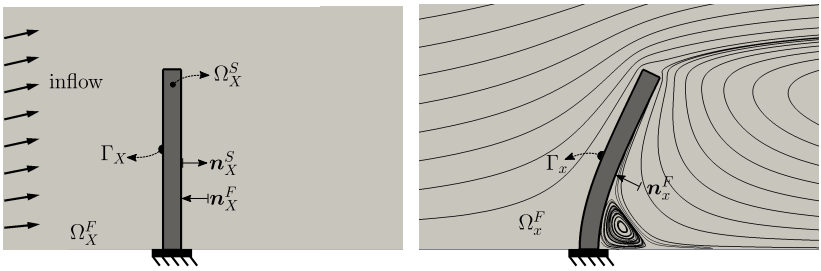


Figure 3.1: Notional schematic of the fluid domain Ω^F, the structure domain Ω^S and the interaction interface Γ, as well as the definition of the interface surface normals. Left: the unreformed state, Right: the equilibrium state.

3.1 Continuous Fluid-Structure Interaction Problem

As illustrated in Figure 3.1, the system under consideration consists of three main parts: fluid domain Ω^F, structural domain Ω^S and wet fluid–structure interface Γ. Note that the superscripts F,S denote that the variable/quantity belongs to the fluid and structure respectively as the convention throughout this text. As has been usually done and will be pursued here, Eulerian and total Lagrangian approaches are used to describe fluid and structure motions, respectively. Note that the same descriptions have conventionally been used in single disciplinary solver implementations. A total Lagrangian approach formulates structural governing equations $\boldsymbol{R}^S$ with respect to the undeformed configuration X while a Eulerian approach formulates fluid governing equations $\boldsymbol{R}^F$ with respect to the deformed configuration x. In order to couple the governing equations, we require kinematic continuity as well as the equilibrium of interface traction fields at the fluid-structure interface. Here, the motion of the fluid domain is described by pseudo-structural governing equations $\boldsymbol{R}^M$. Assuming steady-state conditions, the continuous form of the problem can be written as:

$$\boldsymbol{R}^F\left(\boldsymbol{w},\boldsymbol{x}^F\right)=\boldsymbol{0} \qquad \text{in } \Omega_x^F \tag{3.1a}$$

$$\boldsymbol{R}^S\left(\boldsymbol{u}^S,\boldsymbol{X}^S\right)=\boldsymbol{0} \qquad \text{in } \Omega_X^S \tag{3.1b}$$

$$\boldsymbol{R}^M\left(\boldsymbol{u}^F,\boldsymbol{X}^F\right)=\boldsymbol{0} \qquad \text{in } \Omega_X^F \tag{3.1c}$$

subject to

$$\boldsymbol{v}_\Gamma = \mathbf{0} \qquad \text{on } \Gamma_x \tag{3.1d}$$

$$\boldsymbol{u}^S_\Gamma - \boldsymbol{u}^F_\Gamma = \mathbf{0} \qquad \text{on } \Gamma \tag{3.1e}$$

$$\boldsymbol{X}^F_\Omega + \boldsymbol{u}^F_\Omega = \boldsymbol{x}^F_\Omega \qquad \text{in } \Omega^F_X \tag{3.1f}$$

$$\boldsymbol{X}^F_\Gamma + \boldsymbol{u}^F_\Gamma = \boldsymbol{x}^F_\Gamma \qquad \text{on } \Gamma_X \tag{3.1g}$$

$$\boldsymbol{\sigma}^F \cdot \boldsymbol{n}^F + \boldsymbol{\sigma}^S \cdot \boldsymbol{n}^S = \mathbf{0} \qquad \text{on } \Gamma_x \tag{3.1h}$$

where $\boldsymbol{x}$ and $\boldsymbol{X}$ denote the Cartesian coordinates of deformed and undeformed configurations, respectively. The quantity $\boldsymbol{w}$ denotes the state variables of the fluid, typically velocities $\boldsymbol{v}$ with the pressure p or the density and the internal energy. The displacement fields $\boldsymbol{u}^S$ and $\boldsymbol{u}^F$ represent the displacements in Ω^S_X and in Ω^F_X, respectively. The vector $\boldsymbol{n}$ is the unit normal vector at the interface Γ. $\boldsymbol{\sigma}$ is the Cauchy stress tensor (i.e., stress measured in the deformed configuration). Throughout this work, we use the following conventions:

- The subscript Ω designates the value of the fluid or structural parameter inside the computational domain.
- The subscript Γ indicates the quantity at the fluid-structure interface, e.g., $\boldsymbol{u}^F_\Gamma$ denotes the displacement of the fluid interface.
- The subscripts x and X have been appended to the domains and the interface to carefully distinguish between deformed and undeformed configurations.

Note that above conditions are applied essentially independently of the laws of continuum mechanics used to model the behavior of the fluid and the structure.

3.2 Governing Equations for Fluid

$\boldsymbol{R}^F$ represents the continuum equations that govern fluid flow. Principally, any fluid flow problem could be formulated in any reference frame and solved in that frame subsequently, as long as the boundary conditions are kept consistent. As has been usually done and will be pursued here, Eulerian frame of reference x is chosen to describe the fluid motion. An

Eulerian reference frame can be inertial or moving with the body of interest, for example, moving meshes in FSI and rotating reference frames in rotor applications. Although in this work, only the two limiting cases of an inviscid compressible fluid and a viscous incompressible fluid are considered, the motion of fluid can be described by the full Navier–Stokes compressible equations, from which all the types of governing flow equations can be derived in any reference frame (e.g., rotating reference frame). Defining a conservative variable $\boldsymbol{w} = (\rho^F, \rho^F \boldsymbol{v}, \rho^F E)$, their steady-state formulation for a viscous, compressible, Newtonian flow can be written in the following form:

$$\begin{aligned} R^{F,1} &= \nabla_x \cdot (\rho^F (\boldsymbol{v} - \boldsymbol{v}_\Omega)) = 0 \\ [R^{F,2}, R^{F,3}, R^{F,4}]^T &= \nabla_x . (\rho^F \boldsymbol{v} \otimes (\boldsymbol{v} - \boldsymbol{v}_\Omega)) + \nabla_x p - \nabla_x \cdot (\mu^F \boldsymbol{\tau}) - \boldsymbol{q} = \boldsymbol{0} \\ R^{F,5} &= \nabla_x \cdot (\rho^F (\boldsymbol{v} - \boldsymbol{v}_\Omega) E + p\boldsymbol{v} - \boldsymbol{\tau} \cdot \boldsymbol{v} - (\mu^F / Pr) C_p \nabla_x T) = 0 \end{aligned} \tag{3.2}$$

where ρ^F is the fluid density, $\boldsymbol{v}$ and $\boldsymbol{v}_\Omega$ (Unless explicitly stated otherwise, $\boldsymbol{v}_\Omega = \boldsymbol{0}$) respectively represent the flow velocity and the fluid domain velocity in all dimensions, p is the physical pressure, $\boldsymbol{q}$ is a generic source term vector, E is the total energy of the flow per unit mass, μ^F is the fluid viscosity, Pr is the Prandtl number, C_p is the specific heat, T is the temperature, and $\boldsymbol{\tau}$ is the viscous stress tensor and is defined as

$$\boldsymbol{\tau} = \nabla_x \boldsymbol{v} + \nabla_x \boldsymbol{v}^T - \frac{2}{3} \boldsymbol{I} \left(\nabla_x \cdot \boldsymbol{v} \right) \tag{3.3}$$

Note that the operator ∇_x in the above equation and in Eq. 3.2 denotes the derivatives with respect to the deformed configuration x. After having solved the governing flow equations for a given set of boundary conditions, the fluid Cauchy stress tensor reads

$$\boldsymbol{\sigma}^F = p\boldsymbol{I} - \mu^F \boldsymbol{\tau} \tag{3.4}$$

where $\boldsymbol{I}$ is the identity matrix. We have to remark that in the case of inviscid flow, only the pressure field contributes to the stress tensor, furthermore, the no-slip condition in Eq. 3.1d gets modified to the Euler slip condition (i.e. $\boldsymbol{v} \cdot \boldsymbol{n}^F = 0$). The flow passing over an isolated body which is rotating at a constant angular velocity is intrinsically unsteady. However, the flow can

be approximated as a steady-state solution by transforming the system of governing equations into a non-inertial reference frame that rotates with the body of interest. This is advantageous because it simplifies the CFD and shape sensitivity analyses and consequently makes the optimal shape achievable. With this transformation, the following domain velocity and source term are applied to the system in 3.2:

$$\boldsymbol{v}_{\Omega} = \boldsymbol{\omega} \times \boldsymbol{r}, \qquad \boldsymbol{q} = -\rho^{F}(\boldsymbol{\omega} \times \boldsymbol{v}) \tag{3.5}$$

where $\boldsymbol{\omega} = \{\omega^1, \omega^2, \omega^3\}$ is the steady angular velocity of the rotating body and $\boldsymbol{r}$ is the position vector pointing from the rotation center (x_o^1, x_o^2, x_o^3) to a point (x^1, x^2, x^3) in the flow domain.

3.3 Governing Equations for Structure

Following a total Lagrangian approach, the static and continuous conservation of momentum written in terms of the second Piola-Kirchhoff stress $\boldsymbol{S}$ and the Lagrangian coordinates $\boldsymbol{X}$ of structural domain is

$$[R^{S,1}, R^{S,2}, R^{S,3}]^T = \nabla_X \cdot (\boldsymbol{F} \cdot \boldsymbol{S}) + \rho^S \boldsymbol{b} = \boldsymbol{0} \tag{3.6}$$

Here, $\boldsymbol{F} = \nabla_X \boldsymbol{u}^S$ represents the deformation gradient; ρ^S is the density of the structural domain; $\boldsymbol{b}$ is the volumetric body force. Note that ∇_X indicates the spatial gradient operator acting on the undeformed configuration X. The second Piola-Kirchhoff stress tensor $\boldsymbol{S}$ is related to the Green-Lagrangian strains via

$$\boldsymbol{S} = \boldsymbol{C} : \boldsymbol{E} \quad \text{with} \quad \boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{F}^T \cdot \boldsymbol{F} - \boldsymbol{I}\right) \tag{3.7}$$

where $\boldsymbol{C}$ denotes the material tensor. Furthermore, since different stress measures are used in Eq. 3.1h and Eq. 3.6, the following identity from continuum mechanics [12] is applied:

$$\boldsymbol{n}_X^S \cdot \boldsymbol{S} \, d\Gamma_X = \boldsymbol{n}_x^S \cdot \boldsymbol{\sigma}^S \, d\Gamma_x = -\boldsymbol{n}_x^F \cdot \boldsymbol{\sigma}^F \, d\Gamma_x \tag{3.8}$$

Last but not least, the equations governing the displacement of the fluid domain can then be written as

$$[R^{M,1}, R^{M,2}, R^{M,3}]^T = \nabla_X \cdot \boldsymbol{\sigma}^M = \boldsymbol{0} \tag{3.9}$$

Considering the case of pseudo-linear elasticity, $\boldsymbol{\sigma}^M$ is defined as

$$\boldsymbol{\sigma}^M = \lambda \mathrm{tr}\left(\boldsymbol{\epsilon}\left(\boldsymbol{u}^F\right)\right)\boldsymbol{I} + 2\mu^M \boldsymbol{\epsilon}\left(\boldsymbol{u}^F\right) \tag{3.10}$$

where tr() is the trace operator, λ and μ^M are the Lame constants, and $\boldsymbol{\epsilon}$ is the strain tensor:

$$\boldsymbol{\epsilon}\left(\boldsymbol{u}^F\right) = \frac{1}{2}\left(\nabla_X \boldsymbol{u}^F + (\nabla_X \boldsymbol{u}^F)^T\right) \tag{3.11}$$

3.4 Discrete Fluid-Structure Interaction Problem

In order to solve the explained coupled problem numerically, spatial discretization of the governing equations and all unknown fields is required. Having discretized the fluid and structure domains with m^F and m^S nodes, respectively, the residual form of the stationary FSI problem in Eq. 3.1 reads independently of the spatial discretization scheme as follows:

$$\boldsymbol{r}^F\left(\boldsymbol{w}, \boldsymbol{x}^F\right) = \begin{bmatrix} \left[\boldsymbol{r}^F_\Omega\right]_{m^F_\Omega \times 1} \\ \\ \left[\boldsymbol{r}^F_\Gamma\right]_{m^F_\Gamma \times 1} \end{bmatrix} = [\boldsymbol{0}]_{m^F \times 1} \tag{3.12a}$$

$$\begin{aligned} \boldsymbol{r}^S\left(\boldsymbol{u}^S, \boldsymbol{X}^S, \boldsymbol{w}, \boldsymbol{x}^F\right) &= \begin{bmatrix} \left[\boldsymbol{r}^S_\Omega\right]_{m^S_\Omega \times 1} \\ \\ \left[\boldsymbol{r}^S_\Gamma\right]_{m^S_\Gamma \times 1} \end{bmatrix} = [\boldsymbol{0}]_{m^S \times 1} \\ &= \boldsymbol{f}^{S,ext}\left(\boldsymbol{w}, \boldsymbol{x}^F\right) - \boldsymbol{f}^{S,int}\left(\boldsymbol{u}^S, \boldsymbol{X}^S\right) \end{aligned} \tag{3.12b}$$

$$\begin{aligned} \boldsymbol{r}^M\left(\boldsymbol{u}^F, \boldsymbol{X}^F, \boldsymbol{u}^S\right) &= \begin{bmatrix} \left[\boldsymbol{r}^M_\Omega\right]_{m^F_\Omega \times 1} \\ \\ \left[\boldsymbol{r}^M_\Gamma\right]_{m^F_\Gamma \times 1} \end{bmatrix} = [\boldsymbol{0}]_{m^F \times 1} \\ &= \boldsymbol{f}^{M,ext} - \boldsymbol{f}^{M,int}\left(\boldsymbol{u}^F, \boldsymbol{X}^F\right) \end{aligned} \tag{3.12c}$$

subject to

$$\boldsymbol{v}_\Gamma = \boldsymbol{0} \qquad \text{on } \Gamma_x \tag{3.12d}$$

$$\boldsymbol{r}_\Gamma^M = \boldsymbol{u}_\Gamma^S - \boldsymbol{u}_\Gamma^F = \boldsymbol{0} \qquad \text{on } \Gamma \tag{3.12e}$$

$$\boldsymbol{X}_\Omega^F + \boldsymbol{u}_\Omega^F = \boldsymbol{x}_\Omega^F \qquad \text{in } \Omega_X^F \tag{3.12f}$$

$$\boldsymbol{X}_\Gamma^F + \boldsymbol{u}_\Gamma^F = \boldsymbol{x}_\Gamma^F \qquad \text{on } \Gamma_X \tag{3.12g}$$

$$\boldsymbol{r}_\Gamma^S = \boldsymbol{f}_\Gamma^{S,ext} - \boldsymbol{f}_\Gamma^F = \boldsymbol{0} \qquad \text{on } \Gamma \tag{3.12h}$$

where $\boldsymbol{r}^F$, $\boldsymbol{r}^S$, and $\boldsymbol{r}^M$ are the full residual vectors (including the internal and the boundary values) of the fluid, the structure, and the mesh motion, respectively. $\boldsymbol{f}^{S,int}$ and $\boldsymbol{f}^{S,ext}$ are the vector of internal forces and the vector of external forces in the structure, respectively, while $\boldsymbol{f}^{M,int}$ and $\boldsymbol{f}^{M,ext}$ are the same terms of the mesh motion. While $\boldsymbol{f}^{S,ext}$ is composed of the interface boundary forces due to the fluid flow $\boldsymbol{f}_\Gamma^{S,ext}\left(\boldsymbol{w},\boldsymbol{x}^F\right) = \boldsymbol{f}_\Gamma^F$ and the remaining $\boldsymbol{f}_\Omega^{S,ext}$ (other specified structural loads such as gravity), $\boldsymbol{f}^{M,ext}$ is generally zero. Note also that in Eq. 3.12, $\boldsymbol{x}^F$ represents the nodal coordinates of the fluid mesh in the deformed configuration, while $\boldsymbol{X}^S$ and $\boldsymbol{X}^F$ are the nodal coordinates of the structural and fluid meshes in the undeformed configuration, respectively. We emphasize that all of the variable vectors in Eq. 3.12 are the discrete quantities composed of the interface Γ and the remaining Ω contributions.

3.4.1 Primal solution strategies

Mathematically speaking, fluid-structure interaction problems have a high degree of nonlinearity which is associated with the nonlinearities in the governing equations of fluid and structure. For the solution of the obtained nonlinear system of equations in Eq. 3.12, there are two main methods, namely monolithic and partitioned. While the monolithic approach solves for all primary unknowns in a single system of equations, the partitioned approach breaks down the coupled problem into multiple partitioned subsystems and a set of interface constraints. Although the monolithic approach usually results in a robust and accurate numerical method, a single software package needs to be able to model all the necessary physics. Therefore, it is not possible to reuse any existing solvers. Instead, the partitioned approach allows existing simulation programs of the each discipline to be reused which brings big benefits: on the one hand, it reduces the time

it takes to develop the software; on the other hand, the scope of FSI problems that can be solved is broadened. However, the partitioned approach may suffer from numerical stability and accuracy issues. For a more thorough review of the primal FSI coupling techniques, refer to [40]. To bring out the full generality and completeness, both methods will be discussed in the sequence. Last but not least, the issue of non-matching discrete interfaces is addressed in the context of partitioned FSI.

Monolithic FSI

Simply speaking, we are looking for a vector of primary unknowns $\boldsymbol{Q} = \left[\boldsymbol{w}, \boldsymbol{u}^S, \boldsymbol{u}^F\right]$ which satisfies the following system

$$\boldsymbol{r} = \begin{bmatrix} \boldsymbol{r}^F\left(\boldsymbol{w}, \boldsymbol{x}^F\left(\boldsymbol{u}^F\right)\right) \\ \boldsymbol{r}^S\left(\boldsymbol{u}^S, \boldsymbol{w}\right) \\ \boldsymbol{r}^M\left(\boldsymbol{u}^F, \boldsymbol{u}^S\right) \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \tag{3.13}$$

where $\boldsymbol{r}$ is the complete residual vector which consists of the residual vectors of fluid, structure and mesh motion, respectively. In the next step, the implicit residual vector $\boldsymbol{r}$ is linearized with respect to the unknown solution vector $\boldsymbol{Q}$ and solved for using Newton's method as

$${}^{n+1}\boldsymbol{Q} = {}^{n}\boldsymbol{Q} + \left[\frac{-\partial\, {}^{n}\boldsymbol{r}}{\partial\, {}^{n}\boldsymbol{Q}}\right]^{-1} \cdot {}^{n}\boldsymbol{r} \tag{3.14}$$

herein, ${}^{n}\boldsymbol{Q} = \left[{}^{n}\boldsymbol{w}, {}^{n}\boldsymbol{u}^S, {}^{n}\boldsymbol{u}^F\right]$ is the value of primary unknowns at the end of the nth iteration; $\frac{\partial\, {}^{n}\boldsymbol{r}}{\partial\, {}^{n}\boldsymbol{Q}}$ is the Jacobian or tangent stiffness matrix of the FSI problem. Keeping in mind the dependences indicated in Eq. 3.12, in every Newton-Raphson iteration, one clearly needs to solve for the following linear system of equations

$$-\begin{bmatrix} \frac{\partial\, {}^{n}\boldsymbol{r}^F}{\partial\, {}^{n}\boldsymbol{w}} & \boldsymbol{0} & \frac{\partial\, {}^{n}\boldsymbol{r}^F}{\partial\, {}^{n}\boldsymbol{u}^F} \\ \frac{\partial\, {}^{n}\boldsymbol{r}^S}{\partial\, {}^{n}\boldsymbol{w}} & \frac{\partial\, {}^{n}\boldsymbol{r}^S}{\partial\, {}^{n}\boldsymbol{u}^S} & \frac{\partial\, {}^{n}\boldsymbol{r}^S}{\partial\, {}^{n}\boldsymbol{u}^F} \\ \boldsymbol{0} & \frac{\partial\, {}^{n}\boldsymbol{r}^M}{\partial\, {}^{n}\boldsymbol{u}^S} & \frac{\partial\, {}^{n}\boldsymbol{r}^M}{\partial\, {}^{n}\boldsymbol{u}^F} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{w} \\ \Delta\boldsymbol{u}^S \\ \Delta\boldsymbol{u}^F \end{bmatrix} = \begin{bmatrix} {}^{n}\boldsymbol{r}^F \\ {}^{n}\boldsymbol{r}^S \\ {}^{n}\boldsymbol{r}^M \end{bmatrix} \tag{3.15}$$

where $\frac{\partial^n \boldsymbol{r}^F}{\partial^n \boldsymbol{u}^F}$ is the derivative of the fluid residuals w.r.t the nodal displacements of the fluid mesh, $\frac{\partial^n \boldsymbol{r}^S}{\partial^n \boldsymbol{w}}$ and $\frac{\partial^n \boldsymbol{r}^S}{\partial^n \boldsymbol{u}^F}$ are the derivatives of the structure residuals w.r.t fluid state vector and fluid nodal displacements, respectively, and $\frac{\partial^n \boldsymbol{r}^M}{\partial^n \boldsymbol{u}^S}$ is the derivative of the mesh motion residuals w.r.t structure state vector. The computation of these cross partial derivatives is expensive from the computational and implementation point of view, therefore it puts a burden on the coupling of the fluid and the structure governed by complicated partial differential equations (PDE). On the other hand, having the full and exact Jacobian matrix of the FSI problem is particularly advantageous in terms of the consistent derivation and implementation of the adjoint problem.

Partitioned FSI

The FSI problem stated continuously in Eq. 3.1 and discretely in Eq. 3.12 constitutes a coupled set of non-linear and linear subproblems that can be solved separately and iteratively until the interface conditions, the equilibrium of tractions and kinematic continuity, are satisfied. This results in the so-called Gauss-Seidel fixed-point iterations for a strongly coupled partitioned fluid-structure interaction. Among the partitioned coupling techniques for FSI [9], we use the so-called Dirichlet–Neumann partitioned procedure which is by far the most widely used strategy, both for simplicity and because of wide range of applicability to single disciplinary solvers. This technique treats the fluid domain as the Dirichlet partition, i.e. it takes the prescribed interface displacements as the Dirichlet boundary condition for the mesh motion problem, and the structure domain as the Neumann partition loaded with interface fluid forces.

With the Dirichlet–Neumann partitioned procedure, we break down the stationary FSI problem into the fluid, the structure and the mesh motion subproblems which are treated by black box solvers as

$$\boldsymbol{f}_\Gamma^F = \mathcal{F}(\boldsymbol{x}^F) \tag{3.16a}$$

$$\boldsymbol{u}_\Gamma^S = \mathcal{S}(\boldsymbol{f}_\Gamma^F) \tag{3.16b}$$

$$\boldsymbol{x}^F = \mathcal{M}(\boldsymbol{u}_\Gamma^S) \tag{3.16c}$$

In the equation, $\mathcal{F}$ represents the fluid solver that takes the new position of the fluid mesh $\boldsymbol{x}^F$ as input and outputs the interface load $\boldsymbol{f}_\Gamma^F$ (nodal forces or tractions), $\mathcal{S}$ represents the structure solver that takes $\boldsymbol{f}_\Gamma^F$

as input and outputs $\boldsymbol{u}_\Gamma^S$, and the mesh motion solver $\mathcal{M}$ which outputs the deformed fluid mesh $\boldsymbol{x}^F$ according to $\boldsymbol{u}_\Gamma^S$. A fixed point iteration can be formulated either with one of $\mathcal{S}$ and $\mathcal{F}$ or both. For example, a non-linear fixed point iteration can be formulated as

$$\boldsymbol{u}_\Gamma^S = \mathcal{S} \circ \mathcal{F} \circ \mathcal{M}(\boldsymbol{u}_\Gamma^S) \tag{3.17}$$

meaning that the mesh motion problem is solved first with the given $\boldsymbol{u}_\Gamma^S$, then the fluid solver takes the new mesh and computes the interface forces, finally the structure solver computes the interface displacements using the output from the fluid $\boldsymbol{f}_\Gamma^F$. The output $\boldsymbol{u}_\Gamma^S$ is fed back in an iterative way. The interface residuals of one iteration is defined as

$$\boldsymbol{\delta} = \mathcal{S} \circ \mathcal{F} \circ \mathcal{M}(\boldsymbol{u}_\Gamma^S) - \boldsymbol{u}_\Gamma^S \tag{3.18}$$

When $\|\boldsymbol{\delta}\|_2$ is below a given tolerance ε, the fixed point iteration is converged and the steady state of equilibrium is achieved.

Non-matching discretizations

The partitioned treatment of the FSI problem naturally enables us to handle FSI problems with non-matching discrete interfaces, which is particularly advantageous for large-scale high-fidelity FSI problems with different mesh resolution requirements. When applying the coupling conditions to non-matching meshes, mapping is needed before transferring the information. Although development and assessment of mapping algorithms are not in the scope of this work, we investigated the difference in accuracy between the so-called conservative and consistent mapping criteria [16, 25]. Whichever mapping algorithm is chosen to enforce coupling conditions (Eqs. 3.12e and 3.12h) over non-matching interface meshes, its outcome reads

$$\begin{aligned} \boldsymbol{u}_\Gamma^F &= \boldsymbol{H}^{FS} \cdot \boldsymbol{u}_\Gamma^S \\ \boldsymbol{t}_\Gamma^S &= \boldsymbol{H}^{SF} \cdot \boldsymbol{t}_\Gamma^F \end{aligned} \tag{3.19}$$

$\boldsymbol{H}^{FS}$ transfers consistently the displacement of the structure interface points $\boldsymbol{X}_\Gamma^S$ to that of the fluid interface points $\boldsymbol{X}_\Gamma^F$, while $\boldsymbol{H}^{SF}$ assigns the fluid interface tractions $\boldsymbol{t}_\Gamma^F$ to the structure interface tractions $\boldsymbol{t}_\Gamma^S$ consistently. In order to ensure that a constant displacement and constant

pressure fields are exactly interpolated over the interface, the coefficients of the mapping matrix are renormalized to produce unit row sums. The resulting matrix is the so-called consistent mapping matrix. The conservative mapping matrix can be derived from the interface energy conservation equation and an existing displacement mapping matrix: given $\boldsymbol{H}^{FS}$, it is derived as

$$\boldsymbol{f}_{\Gamma}^{S,ext} = (\boldsymbol{H}^{FS})^{T} \cdot \boldsymbol{f}_{\Gamma}^{F} \tag{3.20}$$

where $\boldsymbol{f}_{\Gamma}^{S,ext}$ and $\boldsymbol{f}_{\Gamma}^{F}$ are the interface forces from the structure and the fluid, respectively. In this way, the discrete form of the energy conservation at the interface is satisfied as the following relation holds:

$$(\boldsymbol{f}_{\Gamma}^{S,ext})^{T} \cdot \boldsymbol{u}_{\Gamma}^{S} = (\boldsymbol{f}_{\Gamma}^{F})^{T} \cdot \boldsymbol{u}_{\Gamma}^{F} \tag{3.21}$$

While the displacement is usually mapped using consistent mapping, the force/traction can be mapped either consistently or conservatively. Although a conservative load transfer guarantees the conservation of the resultant force and energy, it can lead to spurious oscillations in the pressure received by the structure. On the other hand, the consistent mapping results in differences in the resultant force and energy. A thorough mathematical discussion of this topic can be found, for instance, in [16] and [91]. A critical and comparative assessment of mapping criteria for certain mapping algorithms is a part of this work and will be discussed in Sec. 6.3. In this thesis, the following nomenclature is adopted for general data field mapping between non-matching meshes in fluid-structure interaction: *consistent-conservative mapping* which refers to mapping structure data field (e.g., displacement field) consistently while mapping the fluid data field (e.g., force field) conservatively, *consistent-consistent mapping* which refers to mapping both structure and fluid data fields consistently.

Last but not least, Algorithm 2 details the Gauss-Seidel algorithm for a stationary FSI problem with arbitrary non-matching interface meshes using consistent-conservative mapping. In this algorithm, n denotes the current iteration, and $\hat{\boldsymbol{u}}_{\Gamma}^{S}$ is the relaxed interface displacements. Due to the simplicity of implementation and efficiency, the Aitken relaxation is chosen as the default relaxation scheme in this work.

Algorithm 2 Partitioned (or iterative) work flow for stationary FSI using consistent-conservative mapping

//initialize the mapping matrix between $\boldsymbol{X}_\Gamma^F$ and $\boldsymbol{X}_\Gamma^S$
$\boldsymbol{H}^{FS}$
$n = 1$
//initialize interface displacements
${}^n\hat{\boldsymbol{u}}_\Gamma^S = \boldsymbol{0}$
//FSI strong coupling loop
while $\|{}^n\boldsymbol{\delta}\|_2 > \varepsilon$ **do**
// map consistently the relaxed interface displacements
    ${}^n\hat{\boldsymbol{u}}_\Gamma^F = \boldsymbol{H}^{FS} \cdot {}^n\hat{\boldsymbol{u}}_\Gamma^S$
// solve mesh motion problem
    ${}^n\boldsymbol{x}^F = \mathcal{M}({}^n\hat{\boldsymbol{u}}_\Gamma^F)$
// solve fluid problem
    ${}^n\boldsymbol{f}_\Gamma^F = \mathcal{F}({}^n\boldsymbol{x}^F)$
// map conservatively the interface forces
    ${}^n\boldsymbol{f}_\Gamma^{S,ext} = (\boldsymbol{H}^{FS})^T \cdot {}^n\boldsymbol{f}_\Gamma^F$
// solve structure problem
    ${}^n\boldsymbol{u}_\Gamma^S = \mathcal{S}({}^n\boldsymbol{f}_\Gamma^{S,ext})$
// compute interface displacement residuals
    ${}^n\boldsymbol{\delta} = {}^n\boldsymbol{u}_\Gamma^S - {}^n\hat{\boldsymbol{u}}_\Gamma^S$
    compute ${}^{n+1}\hat{\boldsymbol{u}}_\Gamma^S$ **based on** $\{{}^1\boldsymbol{\delta}, {}^2\boldsymbol{\delta}, \cdots, {}^n\boldsymbol{\delta}\}$ **and** $\{{}^1\boldsymbol{u}_\Gamma^S, {}^2\boldsymbol{u}_\Gamma^S, \cdots, {}^n\boldsymbol{u}_\Gamma^S\}$ **(relaxation, etc.)**
    $n = n + 1$
end while

Chapter 4

Adjoint-based Shape Sensitivity Analysis

Having chosen node-based shape parametrization, which provides the richest design space possible for optimization of a discrete geometry, the next step is to efficiently compute gradients of cost function with respect to shape design variables, the so-called shape sensitivity analysis. Adjoint-based shape sensitivity analysis, is still the most efficient and the state of the art technique for the gradient computation and it is used in this work. Adjoint-based shape sensitivity analysis is done either on the discrete level, called discrete adjoint analysis, or on the continuous level, called continuous adjoint analysis. A literature review on the studies from various authors shows that the structures community most commonly uses the discrete adjoint method while the fluid community is mostly driven by the continuous adjoint method.

This chapter presents firstly the derivation of adjoint-based shape gradients for the fluid-structure interaction problem (Chapter3) discretely. Then cost function along with contributions from fluid and structure residuals are differentiated continuously which results in a set of continuous PDEs and boundary conditions and, most importantly, closed form expres-

sions for the fluid shape sensitivities.

4.1 Problem Formulation

Having obtained the equilibrium state of a static aeroelastic system, we formulate the aeroelastic shape sensitivity analysis problem as follows:

We seek to compute the gradients of force-based objective functions defined along the deformed fluid-structure interface Γ_x with respect to shape design variables which specify the undeformed geometry of the interface.

Special attention has to be given to the following points: 1) the objective is a force-based function which depends on the interface forces, 2) shape design variables define the shape of the undeformed interface and the nodal coordinates are considered as the design variables, 3) the balance equations of the aeroelastic system are considered as equality constraints. Keeping in mind that the structure is described by the total Lagrangian approach while the fluid is described by the Eulerian formulation, the shape optimization problem corresponding to the shape sensitivity analysis problem of interest is defined as

$$\begin{aligned} &\min_{\boldsymbol{X}_\Gamma} && J\left(\boldsymbol{w}, \boldsymbol{u}^S, \boldsymbol{x}^F\right) \\ &\text{subject to} \\ &&& \boldsymbol{r} = \begin{bmatrix} \boldsymbol{r}^F\left(\boldsymbol{w}, \boldsymbol{x}^F\right) \\ \boldsymbol{r}^S\left(\boldsymbol{u}^S, \boldsymbol{w}, \boldsymbol{x}^F, \boldsymbol{X}_\Gamma\right) \\ \boldsymbol{r}^M\left(\boldsymbol{u}^F, \boldsymbol{u}^S, \boldsymbol{X}_\Gamma\right) \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \end{aligned} \tag{4.1}$$

where J denotes a force-based objective (drag, lift, strain energy etc.) to be minimized; $\boldsymbol{X}_\Gamma = [\boldsymbol{X}_{\Gamma,1}, \boldsymbol{X}_{\Gamma,2}, ..., \boldsymbol{X}_{\Gamma,m_\Gamma}]$, $\boldsymbol{X}_{\Gamma,i} \in \mathbb{R}^3$ denotes the vector of the interface nodal coordinates of size $3m_\Gamma$ as the design variable vector; $\boldsymbol{r}$ is the residual vector of the aeroelastic system. It should be noted that $\boldsymbol{r}$ contains not only the discretized state equations of each subproblem (Eqs. 3.12a-3.12c) but also the interface coupling conditions (Eqs. 3.12e-3.12h) as indicated by dependencies in Eq. 4.1. Also note that in the above formulation, we assumed that the interface meshes are matching, i.e. $\boldsymbol{X}_\Gamma^F = \boldsymbol{X}_\Gamma^S = \boldsymbol{X}_\Gamma$ and $m_\Gamma^F = m_\Gamma^S = m_\Gamma$.

4.2 Force-based Functionals

In the context of CFD and FSI optimization, force-based objective functionals (lift, drag, strain energy) are the most relevant and popular aerodynamic surface integrals to be optimized. The fluid force acting on the deformed fluid-structure interface in some arbitrary directions $\boldsymbol{d}$ (force projection vector field) may be evaluated continuously as

$$J\left(\boldsymbol{w},\boldsymbol{u}^S,\boldsymbol{x}^F\right)=\int_{\Gamma_x}\boldsymbol{d}\cdot\left(p\boldsymbol{I}-\mu^F\boldsymbol{\tau}\right)\cdot\boldsymbol{n}^F\;d\Gamma \tag{4.2a}$$

and discretely as

$$J=\sum_{i=1}^{m}\boldsymbol{d}_i^T\cdot\boldsymbol{f}_{\Gamma,i}^{S,ext}=\sum_{i=1}^{m}\boldsymbol{d}_i^T\cdot\boldsymbol{f}_{\Gamma,i}^{F} \tag{4.2b}$$

where m is the total number of points on the interface; $\boldsymbol{d}_i$ and $\boldsymbol{f}_{\Gamma,i}^{S,ext}/\boldsymbol{f}_{\Gamma,i}^{F}$ are respectively the force projection vector at node i and the aerodynamic load from fluid to structure at node i. $\boldsymbol{d}$ could be a constant or spatially varying vector field, which is chosen to relate the force on the surface to a desired quantity of interest. For example, in the drag minimization $\boldsymbol{d}_i$ is chosen to be a unit vector opposite to the direction of the incoming flow. Furthermore the strain energy of an aeroelastic system can be linearly approximated by plugging the interface displacement field as the force projection vector field in Eq. 4.2, i.e. $\boldsymbol{d}_i=\boldsymbol{u}_{\Gamma,i}^S$. The nodal force projection vectors relevant to the applications in Chapter 7 are compiled in Table 4.1 where α is the free-stream angle of attack, β is the side-slip angle, $\boldsymbol{r}_i$ is the position vector pointing from the rotation center (x_o^1,x_o^2,x_o^3) to a point (x_i^1,x_i^2,x_i^3) on the interface, and $\boldsymbol{u}_{\Gamma,i}^S$ is the nodal displacement of the objective surface.

4.3 Adjoint-based sensitivity analysis

The goal is to compute the sensitivity derivatives of an objective function to a number of shape design variables, $\frac{dJ}{d\boldsymbol{X}_\Gamma}$, while satisfying a greater number of equality constraints which describe the state of equilibrium of the system. The adjoint method has been shown to be the most efficient way

Table 4.1: Definitions of $\boldsymbol{d}_i$ for different objective functions

Objective	$\boldsymbol{d}_i$
Drag	$(\cos\alpha\cos\beta, \sin\alpha\cos\beta, \sin\beta)$
Lift	$(-\sin\alpha, \cos\alpha, 0)$
Total Power	$\boldsymbol{\omega} \times \boldsymbol{r}_i$
Strain Energy	$\boldsymbol{u}^S_{\Gamma,i}$

with a cost practically independent of the number of design variables. In deriving the adjoint equations, there is principally the choice between a fully discrete approach [29], a fully continuous approach [36] or a combination of the two called the hybrid adjoint approach[5, 86]. Upon examining the literature, shape sensitivity analysis for structures is confined to finite element-based discrete adjoint solvers whereas aerodynamic shape sensitivity analysis is dominated by the finite volume-based continuous and discrete adjoint solvers. In order to make a full exploitation of the actual developments and implementations in single disciplinary adjoint solvers, the following derivation of the adjoint FSI problem is performed in a general way and independently of the used discretization and adjoint methods for each discipline.

In this section, the adjoint system of the FSI problem of interest is consistently derived in such a way that it inherits important characteristics of the primal problem, particularly coupling strategy. Similar to the primal FSI problem, the adjoint FSI problem is a coupled problem consisting of four main parts namely adjoint fluid, adjoint structure, adjoint mesh motion and adjoint coupling conditions. Hence, like any coupled system, it can be solved with either the monolithic strategy or the partitioned strategy. If the full and exact Jacobian matrix of the FSI problem is available from the final converged iteration of primal analysis (Eq. 3.15), the adjoint FSI system is then easily constructed by taking the transpose of the Jacobian. On the other hand, if the full and exact Jacobian is not readily available or the primal problem is solved in a partitioned manner, the adjoint solution may be achieved by a partitioned procedure capable of splitting the linear adjoint FSI problem. As discussed in the introduction, there are two approaches for partitioning the adjoint FSI problem: 1) Three-field formulation which

accounts for aerodynamic, structural, and mesh deformation residuals in adjoint-based sensitivity analysis, 2) two-field formulation which does not include the mesh motion problem for the sensitivity analysis. For the purposes of the present work, we will study the above three mentioned coupling schemes. Before proceeding to the derivation of the adjoint FSI problem, special attention needs to be given to the following remarks:

Remark 1. The adjoint problem of a coupled FSI system is also a coupled system and it consists of sub-adjoint problems which are coupled at the interface of fluid and structure via adjoint coupling conditions.

Remark 2. Mesh motion is only meant to preserve the mesh quality in the discretized fluid domain, otherwise mesh motion and its adjoint do not exist in the continuous (not discretized) primal and adjoint problems, respectively.

Remark 3. As one of the findings in the literature and in this work, it is worth mentioning that the fully discrete adjoint approach appears to be the most accurate choice among the given alternatives. However, it also requires memory and CPU-intense evaluations of the partial derivatives and more importantly, their actual availability in a given code environment. Since the required cross derivatives are not ready to use within already available open-source multiphysics frameworks, an efficient approach that is accurate enough has to be found. The requirement is that the intended approach integrates easily into a partitioned co-simulation software environment with adjoint capabilities in the single disciplinary solvers.

4.3.1 Monolithic adjoint FSI

If the tangent matrix of the aero-elastic system $\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{Q}}$ is available and accessible, then one can easily use it for the adjoint problem. Knowing $\boldsymbol{r}$ is the complete residual vector of the coupled system, the Lagrange function corresponding to the shape optimization problem in Eq. 4.1 reads

$$L(\boldsymbol{Q},\boldsymbol{x}^F,\boldsymbol{X}_\Gamma) = J(\boldsymbol{Q},\boldsymbol{x}^F) + \boldsymbol{\Psi}^T \cdot \boldsymbol{r}(\boldsymbol{Q},\boldsymbol{x}^F,\boldsymbol{X}_\Gamma) \tag{4.3}$$

where superscript T denotes the transpose operator; $\boldsymbol{\Psi} = \left[\boldsymbol{\Psi}^F, \boldsymbol{\Psi}^S, \boldsymbol{\Psi}^M\right]$ is the vector of the adjoint variables (Lagrange multipliers) associated with the complete residual vector ($\boldsymbol{r} = \left[\boldsymbol{r}^F, \boldsymbol{r}^S, \boldsymbol{r}^M\right]$). Remember J is a force-based functional depending only on the primary unknowns $\boldsymbol{Q}$ and the

fluid mesh points $\boldsymbol{x}^F$ (see Section 4.2). The total variation of L with respect to the undeformed shape of the wet surface $\boldsymbol{X}_\Gamma$ hence reads:

$$\frac{dL}{d\boldsymbol{X}_\Gamma} = \frac{dJ}{d\boldsymbol{X}_\Gamma} + \boldsymbol{\Psi}^T \cdot \frac{d\boldsymbol{r}}{d\boldsymbol{X}_\Gamma} \tag{4.4}$$

Applying the chain rule of differentiation, the total derivative is as follows:

$$\begin{aligned}\frac{dL}{d\boldsymbol{X}_\Gamma} = &\left(\frac{\partial J}{\partial \boldsymbol{Q}} + \boldsymbol{\Psi}^T \cdot \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{Q}}\right) \cdot \frac{d\boldsymbol{Q}}{d\boldsymbol{X}_\Gamma} + \\ &\left(\frac{\partial J}{\partial \boldsymbol{x}^F} + \boldsymbol{\Psi}^T \cdot \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{x}^F}\right) \cdot \frac{d\boldsymbol{x}^F}{d\boldsymbol{X}_\Gamma}|_{\boldsymbol{Q}=const.} + \\ &\left(\boldsymbol{\Psi}^T \cdot \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{X}_\Gamma}\right) \cdot \frac{d\boldsymbol{X}_\Gamma}{d\boldsymbol{X}_\Gamma}\end{aligned} \tag{4.5}$$

where the first term on the right-hand side accounts for the sensitivities of the state variables $\boldsymbol{Q}$ with respect to the coordinates of the grid points on the interface in the undeformed state $\boldsymbol{X}_\Gamma$. The remaining terms on the right-hand side stand for the shape sensitivities of the objective function and the residual vector. To avoid having to calculate the computationally expensive term $\frac{d\boldsymbol{Q}}{d\boldsymbol{X}_\Gamma}$, the term multiplying this quantity is set to zero to give rise to the adjoint equation

$$\left[\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{Q}}\right]^T \cdot \boldsymbol{\Psi} = \left[\frac{-\partial J}{\partial \boldsymbol{Q}}\right]^T \tag{4.6}$$

where the right-hand side is the partial derivative of the functional with respect to variations in state/primary variables. The main advantage of monolithic over partitioned is that the adjoint system is easily derived by taking the transpose of full tangent stiffness matrix.

Expanding the total derivative of the coupling conditions Eqs. 3.12f-3.12g with respect to the design variables while keeping the fluid displacement constant, we can write,

$$\frac{d\boldsymbol{x}^F}{d\boldsymbol{X}_\Gamma}|_{\boldsymbol{u}^F=const.} = \begin{bmatrix} \frac{d\boldsymbol{x}^F_\Omega}{d\boldsymbol{X}_\Gamma} \\ \\ \frac{d\boldsymbol{x}^F_\Gamma}{d\boldsymbol{X}_\Gamma} \end{bmatrix} = \begin{bmatrix} \frac{d\boldsymbol{X}^F_\Omega}{d\boldsymbol{X}_\Gamma} + \frac{d\boldsymbol{u}^F_\Omega}{d\boldsymbol{X}_\Gamma} \\ \\ \frac{d\boldsymbol{X}^F_\Gamma}{d\boldsymbol{X}_\Gamma} + \frac{d\boldsymbol{u}^F_\Gamma}{d\boldsymbol{X}_\Gamma} \end{bmatrix} = \begin{bmatrix} [\boldsymbol{0}]_{m^F_\Omega \times m_\Gamma} \\ \\ \boldsymbol{I}_{m_\Gamma \times m_\Gamma} \end{bmatrix} \tag{4.7}$$

where the fact that $\boldsymbol{X}_{\Omega}^{F}$ does not depend on the design variables has been used. Considering the above relationship, the non-zero terms in Eq. 4.5 read:

$$\frac{dL}{d\boldsymbol{X}_{\Gamma}} = \frac{\partial J}{\partial \boldsymbol{x}_{\Gamma}} + \boldsymbol{\Psi}^{T} \cdot \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{x}_{\Gamma}} + \boldsymbol{\Psi}^{T} \cdot \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{X}_{\Gamma}} \tag{4.8}$$

A further simplification is possible by expanding the shape derivatives of the complete residual vector as

$$\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{x}_{\Gamma}} = \begin{bmatrix} \frac{\partial \boldsymbol{r}^{F}(\boldsymbol{w},\boldsymbol{x}^{F})}{\partial \boldsymbol{x}_{\Gamma}} \\ \frac{\partial \left(\boldsymbol{f}^{S,ext}(\boldsymbol{w},\boldsymbol{x}^{F}) - \boldsymbol{f}^{S,int}(\boldsymbol{u}^{S},\boldsymbol{X}^{S})\right)}{\partial \boldsymbol{x}_{\Gamma}} \\ \frac{\partial \left(\boldsymbol{f}^{M,ext} - \boldsymbol{f}^{M,int}(\boldsymbol{u}^{F},\boldsymbol{X}^{F})\right)}{\partial \boldsymbol{x}_{\Gamma}} \end{bmatrix} = \begin{bmatrix} \frac{\partial \boldsymbol{r}^{F}}{\partial \boldsymbol{x}_{\Gamma}} \\ [\boldsymbol{0}]_{m_{\Omega}^{S} \times m_{\Gamma}} \\ \frac{\partial \boldsymbol{f}_{\Gamma}^{S,ext}}{\partial \boldsymbol{x}_{\Gamma}} \\ [\boldsymbol{0}]_{m^{F} \times m_{\Gamma}} \end{bmatrix} \tag{4.9a}$$

$$\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{X}_{\Gamma}} = \begin{bmatrix} \frac{\partial \boldsymbol{r}^{F}(\boldsymbol{w},\boldsymbol{x}^{F})}{\partial \boldsymbol{X}_{\Gamma}} \\ \frac{\partial \left(\boldsymbol{f}^{S,ext}(\boldsymbol{w},\boldsymbol{x}^{F}) - \boldsymbol{f}^{S,int}(\boldsymbol{u}^{S},\boldsymbol{X}^{S})\right)}{\partial \boldsymbol{X}_{\Gamma}} \\ \frac{\partial \left(\boldsymbol{f}^{M,ext} - \boldsymbol{f}^{M,int}(\boldsymbol{u}^{F},\boldsymbol{X}^{F})\right)}{\partial \boldsymbol{X}_{\Gamma}} \end{bmatrix} = \begin{bmatrix} [\boldsymbol{0}]_{m^{F} \times m_{\Gamma}} \\ -\frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_{\Gamma}} \\ -\frac{\partial \boldsymbol{f}^{M,int}}{\partial \boldsymbol{X}_{\Gamma}} \end{bmatrix} \tag{4.9b}$$

where we made use of the fact that only the wet boundary part of $\boldsymbol{f}^{S,ext}$ is a function of the deformed interface nodal coordinates $\boldsymbol{x}_{\Gamma}$, i.e.,

$$\begin{aligned} (\boldsymbol{\Psi}_{\Omega}^{S})^{T} \cdot \frac{\partial \boldsymbol{f}_{\Omega}^{S,ext}}{\partial \boldsymbol{x}_{\Gamma}} &= [\boldsymbol{0}]_{m_{\Omega}^{S} \times m_{\Gamma}} \\ (\boldsymbol{\Psi}^{S})^{T} \cdot \frac{\partial \boldsymbol{f}^{S,ext}}{\partial \boldsymbol{x}_{\Gamma}} &= (\boldsymbol{\Psi}_{\Gamma}^{S})^{T} \cdot \frac{\partial \boldsymbol{f}_{\Gamma}^{S,ext}}{\partial \boldsymbol{x}_{\Gamma}} \end{aligned} \tag{4.10}$$

Finally, plugging 4.9 and 4.2 into 4.8, the coupled sensitivity equation is obtained as follows:

$$\frac{dL}{d\boldsymbol{X}_\Gamma} = \underbrace{\left(\overbrace{(\boldsymbol{\Psi}_\Gamma^S + \boldsymbol{d})^T \cdot \frac{\partial \boldsymbol{f}_\Gamma^{S,ext}}{\partial \boldsymbol{x}_\Gamma}}^{\frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma}} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma} \right)}_{\boldsymbol{g}_\Gamma^{*F}} + \left(\underbrace{-(\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{f}^{M,int}}{\partial \boldsymbol{X}_\Gamma}}_{\boldsymbol{g}_\Gamma^M} \underbrace{-(\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma}}_{\boldsymbol{g}_\Gamma^S} \right), \tag{4.11}$$

or in short

$$\boldsymbol{g}_\Gamma^{FSI} = \boldsymbol{g}_\Gamma^{*F} + \boldsymbol{g}_\Gamma^M + \boldsymbol{g}_\Gamma^S$$

where $\boldsymbol{g}_\Gamma^{*F}$ is the adjoint-based shape gradients of the fluid at the interface while $\boldsymbol{g}_\Gamma^M, \boldsymbol{g}_\Gamma^S$ are the same term of the mesh motion and the structure at the interface respectively, and J^* is the fluid force acting on the interface in the direction of the modified force projection vector field,

$$\boldsymbol{d}^* = \boldsymbol{d} + \boldsymbol{\Psi}_\Gamma^S \tag{4.12}$$

We may now note the following points for the adjoint-based shape sensitivity analysis of an FSI problem formulated using the Eulerian-Lagrangian approach (i.e. Eqs. 3.1 and 3.12):

1. The shape gradients of the fluid problem, including the modified force-based objective function and the fluid residuals, are computed in the **deformed** configuration. This is consistent with the primal problem formulation.

2. Since the structure and fluid mesh motion problems are defined and solved in the undeformed configuration, the gradients of their residuals are computed with respect to the **undeformed** shape of the interface.

Table 4.2: Summary of partitioned adjoint-based sensitivity analysis for FSI using three-field formulation

	Discipline		
	Fluid	Structure	Mesh motion
Adjoint equation	$\boldsymbol{r}^{F,a} = \left[\frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{w}}\right]^T \cdot \boldsymbol{\Psi}^F + \left(\frac{\partial J^*}{\partial \boldsymbol{w}}\right)^T = \boldsymbol{0}$	$\boldsymbol{r}^{S,a} = \left[\frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{u}^S}\right]^T \cdot \boldsymbol{\Psi}^S - \boldsymbol{f}^{S,a} = \boldsymbol{0}$	$\boldsymbol{r}^{M,a} = \left[\frac{\partial \boldsymbol{f}^{M,int}}{\partial \boldsymbol{u}^F}\right]^T \cdot \boldsymbol{\Psi}^M - \boldsymbol{f}^{M,a} = \boldsymbol{0}$
Coupling condition	$\boldsymbol{d}^* = \boldsymbol{d} + \boldsymbol{\Psi}^S_\Gamma$	$\boldsymbol{f}^{S,a}_\Gamma = \frac{\partial \boldsymbol{d}}{\partial \boldsymbol{u}^S_\Gamma} \cdot \boldsymbol{f}^{S,ext}_\Gamma + \boldsymbol{\Psi}^M_\Gamma$	$\boldsymbol{f}^{M,a} = \boldsymbol{g}^{*F} = \left(\frac{\partial J^*}{\partial \boldsymbol{x}^F} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}^F}\right)^T$
Sensitivity contribution	$\boldsymbol{g}^{*F}_\Gamma = \frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma}$	$\boldsymbol{g}^S_\Gamma = -(\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma}$	$\boldsymbol{g}^M_\Gamma = -(\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{f}^{M,int}}{\partial \boldsymbol{X}_\Gamma}$
Total sensitivity equation	$\frac{dL}{d\boldsymbol{X}_\Gamma} = \frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma} - (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma} - (\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{f}^{M,int}}{\partial \boldsymbol{X}_\Gamma} = \boldsymbol{g}^{*F}_\Gamma + \boldsymbol{g}^S_\Gamma + \boldsymbol{g}^M_\Gamma = \boldsymbol{g}^{FSI}$		

4.3.2 Partitioned adjoint FSI by three-field formulation

If the primal FSI problem is solved using a partitioned block Gauss-Seidel method, then the coupled adjoint-based sensitivities can be calculated in a partitioned way. In a manner consistent with the primal FSI problem, we define a Lagrange function that augments the objective function to incorporate the state constraints (Eq. 4.1):

$$\begin{aligned} &L(\boldsymbol{w},\boldsymbol{u}^S,\boldsymbol{u}^F,\boldsymbol{x}^F,\boldsymbol{X}_\Gamma)= \\ &J\left(\boldsymbol{w},\boldsymbol{u}^S,\boldsymbol{x}^F\right)+(\boldsymbol{\Psi}^F)^T\cdot\boldsymbol{r}^F(\boldsymbol{w},\boldsymbol{x}^F)+ \\ &(\boldsymbol{\Psi}^S)^T\cdot\left[\boldsymbol{f}^{S,ext}\left(\boldsymbol{w},\boldsymbol{x}^F\right)-\boldsymbol{f}^{S,int}\left(\boldsymbol{u}^S,\boldsymbol{X}_\Gamma\right)\right]+ \\ &(\boldsymbol{\Psi}^M)^T\cdot\boldsymbol{r}^M(\boldsymbol{u}^F,\boldsymbol{u}^S,\boldsymbol{X}_\Gamma) \end{aligned} \tag{4.13}$$

Note that in the above equation, we replaced $\boldsymbol{r}^S\left(\boldsymbol{u}^S,\boldsymbol{w},\boldsymbol{x}^F,\boldsymbol{X}_\Gamma\right)$ with its expanded form $\boldsymbol{f}^{S,ext}\left(\boldsymbol{w},\boldsymbol{x}^F\right)-\boldsymbol{f}^{S,int}\left(\boldsymbol{u}^S,\boldsymbol{X}_\Gamma\right)$. Please refer to Eq. 3.12 for the mentioned dependencies in the residual vectors. The gradient of the above Lagrangian with respect to design variables reads

$$\begin{aligned} &\frac{dL}{d\boldsymbol{X}_\Gamma}= \\ &\left[\frac{\partial J}{\partial\boldsymbol{w}}\cdot\frac{d\boldsymbol{w}}{d\boldsymbol{X}_\Gamma}+\frac{\partial J}{\partial\boldsymbol{u}^S}\cdot\frac{d\boldsymbol{u}^S}{d\boldsymbol{X}_\Gamma}+\frac{\partial J}{\partial\boldsymbol{x}^F}\cdot\frac{d\boldsymbol{x}^F}{d\boldsymbol{X}_\Gamma}\right]+ \\ &(\boldsymbol{\Psi}^F)^T\cdot\left[\frac{\partial\boldsymbol{r}^F}{\partial\boldsymbol{w}}\cdot\frac{d\boldsymbol{w}}{d\boldsymbol{X}_\Gamma}+\frac{\partial\boldsymbol{r}^F}{\partial\boldsymbol{x}^F}\cdot\frac{d\boldsymbol{x}^F}{d\boldsymbol{X}_\Gamma}\right]+ \\ &(\boldsymbol{\Psi}^S)^T\cdot\left[\frac{\partial\boldsymbol{f}^{S,ext}}{\partial\boldsymbol{w}}\cdot\frac{d\boldsymbol{w}}{d\boldsymbol{X}_\Gamma}+\frac{\partial\boldsymbol{f}^{S,ext}}{\partial\boldsymbol{x}^F}\cdot\frac{d\boldsymbol{x}^F}{d\boldsymbol{X}_\Gamma}-\right. \\ &\qquad\left.\frac{\partial\boldsymbol{f}^{S,int}}{\partial\boldsymbol{u}^S}\cdot\frac{d\boldsymbol{u}^S}{d\boldsymbol{X}_\Gamma}-\frac{\partial\boldsymbol{f}^{S,int}}{\partial\boldsymbol{X}_\Gamma}\right]+ \\ &(\boldsymbol{\Psi}^M)^T\cdot\left[\frac{\partial\boldsymbol{r}^M}{\partial\boldsymbol{u}^F}\cdot\frac{d\boldsymbol{u}^F}{d\boldsymbol{X}_\Gamma}+\frac{\partial\boldsymbol{r}^M}{\partial\boldsymbol{u}^S}\cdot\frac{d\boldsymbol{u}^S}{d\boldsymbol{X}_\Gamma}+\frac{\partial\boldsymbol{r}^M}{\partial\boldsymbol{X}_\Gamma}\right] \end{aligned} \tag{4.14}$$

Before proceeding with the derivation of adjoint equations and the sensitivity equation, we expand $\frac{d\boldsymbol{x}^F}{d\boldsymbol{X}_\Gamma}$ by taking the total derivative of the geometrical compatibility conditions in Eqs. 3.12f-3.12g

$$\frac{d\boldsymbol{x}^F}{d\boldsymbol{X}_\Gamma}=\frac{d\boldsymbol{X}^F}{d\boldsymbol{X}_\Gamma}+\frac{d\boldsymbol{u}^F}{d\boldsymbol{X}_\Gamma} \tag{4.15}$$

After plugging the above equation in Eq. 4.14 and reordering the terms, we obtain

$$
\begin{aligned}
&\frac{dL}{d\boldsymbol{X}_\Gamma} = \\
&\left[\frac{\partial J}{\partial \boldsymbol{w}} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{w}} + (\boldsymbol{\Psi}_\Gamma^S)^T \cdot \frac{\partial \boldsymbol{f}_\Gamma^{S,ext}}{\partial \boldsymbol{w}}\right] \cdot \frac{d\boldsymbol{w}}{d\boldsymbol{X}_\Gamma} + \\
&\left[\frac{\partial J}{\partial \boldsymbol{u}^S} - (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{u}^S} + (\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{r}^M}{\partial \boldsymbol{u}^S}\right] \cdot \frac{d\boldsymbol{u}^S}{d\boldsymbol{X}_\Gamma} + \\
&\left[(\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{r}^M}{\partial \boldsymbol{u}^F} + \frac{\partial J}{\partial \boldsymbol{x}^F} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}^F} + \right. \\
&\left.(\boldsymbol{\Psi}_\Gamma^S)^T \cdot \frac{\partial \boldsymbol{f}_\Gamma^{S,ext}}{\partial \boldsymbol{x}^F}\right] \cdot \frac{d\boldsymbol{u}^F}{d\boldsymbol{X}_\Gamma} + \\
&\left[\left(\frac{\partial J}{\partial \boldsymbol{x}^F} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}^F} + (\boldsymbol{\Psi}_\Gamma^S)^T \cdot \frac{\partial \boldsymbol{f}_\Gamma^{S,ext}}{\partial \boldsymbol{x}^F}\right) \cdot \frac{d\boldsymbol{X}^F}{d\boldsymbol{X}_\Gamma}|_{\boldsymbol{Q}=const.} + \right. \\
&\left.(\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{r}^M}{\partial \boldsymbol{X}_\Gamma} - (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma}\right]
\end{aligned}
\tag{4.16}
$$

It must be mentioned that in the above derivations, we made use of the fact that only the wet boundary part of $\boldsymbol{f}^{S,ext}$ is a function of the flow variables $\boldsymbol{w}$ and the nodal coordinates of the deformed fluid domain $\boldsymbol{x}^F$, i.e.,

$$
\begin{aligned}
(\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}}{\partial \boldsymbol{w}} &= (\boldsymbol{\Psi}_\Gamma^S)^T \cdot \frac{\partial \boldsymbol{f}_\Gamma^{S,ext}}{\partial \boldsymbol{w}} \\
(\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}}{\partial \boldsymbol{x}^F} &= (\boldsymbol{\Psi}_\Gamma^S)^T \cdot \frac{\partial \boldsymbol{f}_\Gamma^{S,ext}}{\partial \boldsymbol{x}^F}
\end{aligned}
\tag{4.17}
$$

While the terms multiplying $\frac{d\boldsymbol{w}}{d\boldsymbol{X}_\Gamma}$, $\frac{d\boldsymbol{u}^S}{d\boldsymbol{X}_\Gamma}$ and $\frac{d\boldsymbol{u}^F}{d\boldsymbol{X}_\Gamma}$, respectively, are eliminated respectively by satisfying the adjoint fluid problem, the adjoint structure problem and the adjoint mesh motion problem, the expression in the last brackets gives rise to the coupled shape gradients. This is in accordance with the three-field formulation for the adjoint analysis of FSI, which accounts for the adjoint mesh motion problem in contrast to the two-field formulation. In the following subsections, three-field adjoint systems as

well as coupled-adjoint sensitivity equations are presented and discussed in detail.

- **Coupled adjoint fluid problem**: The Lagrange multipliers $\boldsymbol{\Psi}^F$ and $\boldsymbol{\Psi}^S_\Gamma$ are chosen such that the variation w.r.t. the fluid state variables vanishes identically:

$$\left[\frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{w}}\right]^T \cdot \boldsymbol{\Psi}^F = -\left(\frac{\partial J}{\partial \boldsymbol{w}} + (\boldsymbol{\Psi}^S_\Gamma)^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}_\Gamma}{\partial \boldsymbol{w}}\right)^T \tag{4.18}$$

  Observe that this equation is basically the well-known adjoint system of equations for fluids, i.e.

$$\left[\frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{w}}\right]^T \cdot \boldsymbol{\Psi}^F = -\left(\frac{\partial J}{\partial \boldsymbol{w}}\right)^T, \tag{4.19}$$

  which is augmented by the fluid state variation of the interface force vector projected onto the interface adjoint displacements. After plugging the force-based functional Eq. 4.2 into Eq. 4.18, the coupled adjoint fluid problem reads

$$\begin{aligned}\left[\frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{w}}\right]^T \cdot \boldsymbol{\Psi}^F &= -\left((\boldsymbol{d} + \boldsymbol{\Psi}^S_\Gamma)^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}_\Gamma}{\partial \boldsymbol{w}}\right)^T \\ &= -\left((\boldsymbol{d}^*)^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}_\Gamma}{\partial \boldsymbol{w}}\right)^T \\ &= -\left(\frac{\partial J^*}{\partial \boldsymbol{w}}\right)\end{aligned} \tag{4.20}$$

  where $\boldsymbol{d}^* = \boldsymbol{d} + \boldsymbol{\Psi}^S_\Gamma$ is the modified force projection vector field. As a result, we find that in the coupled adjoint sensitivity analysis there is no need to compute the extra partial derivative and add it as a source term to the adjoint fluid solution, instead the interface adjoint displacements only need to be added to the force projection vector field.

- **Coupled adjoint structure problem**: To avoid the computationally expensive operation of computing variation of structure displacements w.r.t the shape design variables, i.e. $\frac{d\boldsymbol{u}^S}{d\boldsymbol{X}_\Gamma}$, the following system

needs to be solved:

$$\left[\frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{u}^S}\right]^T \cdot \boldsymbol{\Psi}^S = \left(\frac{\partial J}{\partial \boldsymbol{u}^S} + (\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{r}^M}{\partial \boldsymbol{u}^S}\right)^T = \boldsymbol{f}^{S,a} \tag{4.21}$$

where $\frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{u}^S}$ is the symmetric tangential stiffness matrix of structure; $\boldsymbol{f}^{S,a}$ (the right-hand side of the adjoint structure problem) is interpreted as the structural adjoint force. Note that since the tangent matrix is a symmetric self-adjoint matrix, the same primal structural solver can be reused with a given adjoint force. For the force-based functional defined in Eq. 4.2, the structural adjoint force (the right-hand side of Eq. 4.21) is calculated as follows:

$$\begin{aligned}
\boldsymbol{f}^{S,a}_{\Omega^S} &= [\boldsymbol{0}]_{m^S_\Omega \times 1} \\
\boldsymbol{f}^{S,a}_{\Gamma} &= \left(\left(\boldsymbol{f}^{S,ext}_{\Gamma}\right)^T \cdot \frac{\partial \boldsymbol{d}}{\partial \boldsymbol{u}^S_\Gamma} + \boldsymbol{d}^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}_{\Gamma}}{\partial \boldsymbol{u}^S_\Gamma} + (\boldsymbol{\Psi}^M_\Gamma)^T \cdot \frac{\partial \boldsymbol{r}^M_\Gamma}{\partial \boldsymbol{u}^S_\Gamma}\right)^T
\end{aligned} \tag{4.22}$$

where $(\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{r}^M}{\partial \boldsymbol{u}^S}$ in Eq. 4.21 is replaced with $(\boldsymbol{\Psi}^M_\Gamma)^T \cdot \frac{\partial \boldsymbol{r}^M_\Gamma}{\partial \boldsymbol{u}^S_\Gamma}$ in Eq. 4.22 due to the fact that only the interface residuals of the mesh motion problem depend on the interface structural displacements (see Eq. 3.12e). Note that the term $\frac{\partial \boldsymbol{d}}{\partial \boldsymbol{u}^S_\Gamma}$ is zero for a pure aerodynamic force objective and it is the identity matrix for the strain energy objective. Furthermore, as the external force applied to the structure does not depend explicitly on the structural displacements, the term multiplying $\frac{\partial \boldsymbol{f}^{S,ext}_{\Gamma}}{\partial \boldsymbol{u}^S_\Gamma}$ vanishes. Finally, the last term in Eq. 4.22 can be simplified by taking the partial derivative of Eq. 3.12e with respect to the interface structural displacements as follows:

$$\frac{\partial \boldsymbol{r}^M_\Gamma}{\partial \boldsymbol{u}^S_\Gamma} = \boldsymbol{I}_{m_\Gamma \times m_\Gamma} \tag{4.23}$$

When we take the above-mentioned findings into account, we obtain for the structural adjoint force at the kth node on the interface

$$\boldsymbol{f}^{S,a}_{\Gamma,k} = \frac{\partial \boldsymbol{d}_k}{\partial \boldsymbol{u}^S_{\Gamma,k}} \cdot \boldsymbol{f}^{S,ext}_{\Gamma,k} + \boldsymbol{\Psi}^M_{\Gamma,k} \tag{4.24}$$

where $\boldsymbol{d}_k$, $\boldsymbol{u}^S_{\Gamma,k}$ and $\boldsymbol{\Psi}^M_{\Gamma,k}$ are the force projection vector, structural displacement and fluid adjoint displacement at the node k, respectively. Considering the general expression for the uncoupled adjoint structure problem:

$$\left[\frac{\partial \boldsymbol{r}^S}{\partial \boldsymbol{u}^S}\right]^T \cdot \boldsymbol{\Psi}^S = \left[-\frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{u}^S}\right]^T \cdot \boldsymbol{\Psi}^S = -\left(\frac{\partial J}{\partial \boldsymbol{u}^S}\right)^T \tag{4.25}$$

and comparing it to Eq. 4.21, we can conclude that the coupled adjoint structure displacements can be computed using a single disciplinary adjoint structural solver with contributions from the fluid adjoint displacements over the FSI interface.

- **Coupled adjoint mesh motion problem**: As the last step in the three-field-based adjoint analysis, we solve for the fluid adjoint displacements using

$$\begin{aligned}&\left[\frac{\partial \boldsymbol{r}^M}{\partial \boldsymbol{u}^F}\right]^T \cdot \boldsymbol{\Psi}^M = \\ &-\left(\boldsymbol{d}^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}_\Gamma}{\partial \boldsymbol{x}^F} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}^F} + (\boldsymbol{\Psi}^S_\Gamma)^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}_\Gamma}{\partial \boldsymbol{x}^F}\right)^T\end{aligned} \tag{4.26}$$

Since the pseudo-linear elasticity equations are used to describe the motion of the fluid domain, the residual vector $\boldsymbol{r}^M$ is composed primarily of the vectors of external and internal forces as written in Eq. 3.12c. As a result, the coupled adjoint mesh motion problem simplifies to:

$$\left[\frac{\partial \boldsymbol{f}^{M,int}}{\partial \boldsymbol{u}^F}\right]^T \cdot \boldsymbol{\Psi}^M = \left(\frac{\partial J^*}{\partial \boldsymbol{x}^F} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}^F}\right)^T = \boldsymbol{f}^{M,a} \tag{4.27}$$

where $\frac{\partial \boldsymbol{f}^{M,ext}}{\partial \boldsymbol{u}^F} = \boldsymbol{0}$ is applied due to full Dirichlet boundary conditions for the mesh motion (i.e. $\boldsymbol{f}^{M,ext} = \boldsymbol{0}$), $\frac{\partial \boldsymbol{f}^{M,int}}{\partial \boldsymbol{u}^S}$ is the symmetric tangential stiffness matrix of mesh motion. Note that in arriving at the above equation, we have merged the first and the last terms on

the right-hand side of Eq. 4.26. An important observation is that the right-hand side is a body-force-like term and here we name it the domain-based fluid adjoint force $\boldsymbol{f}^{M,a}$ for the adjoint mesh motion problem.

Given that the adjoint-based sensitivity of a general and uncoupled fluid problem w.r.t to the fluid domain reads:

$$\boldsymbol{g}^{F} = \left(\frac{\partial J}{\partial \boldsymbol{x}^{F}} + (\boldsymbol{\Psi}^{F})^{T} \cdot \frac{\partial \boldsymbol{r}^{F}}{\partial \boldsymbol{x}^{F}} \right)^{T}, \tag{4.28}$$

the reader can easily verify that the domain-based fluid adjoint force $\boldsymbol{f}^{M,a}$ is basically the adjoint-based and domain-based sensitivity of the coupled force-based functional J^{*}, i.e.

$$\boldsymbol{g}^{*F} = \boldsymbol{f}^{M,a} = \left(\frac{\partial J^{*}}{\partial \boldsymbol{x}^{F}} + (\boldsymbol{\Psi}^{F})^{T} \cdot \frac{\partial \boldsymbol{r}^{F}}{\partial \boldsymbol{x}^{F}} \right)^{T} \tag{4.29}$$

Note carefully that $\boldsymbol{g}^{*F}$ is a vector of dimension $3m^{F}$.

- **Coupled sensitivity equation**: Having systematically derived strongly coupled adjoint systems, we compute the coupled shape sensitivities from the remaining non-zero terms in Eq. 4.16 as follows:

$$\begin{aligned} &\frac{dL}{d\boldsymbol{X}_{\Gamma}} = \\ &\left((\boldsymbol{\Psi}_{\Gamma}^{S} + \boldsymbol{d})^{T} \cdot \frac{\partial \boldsymbol{f}_{\Gamma}^{S,ext}}{\partial \boldsymbol{x}^{F}} + (\boldsymbol{\Psi}^{F})^{T} \cdot \frac{\partial \boldsymbol{r}^{F}}{\partial \boldsymbol{x}^{F}} \right) \cdot \frac{d\boldsymbol{X}^{F}}{d\boldsymbol{X}_{\Gamma}}|_{\boldsymbol{Q}=const.} + \\ &(\boldsymbol{\Psi}^{M})^{T} \cdot \frac{\partial \boldsymbol{r}^{M}}{\partial \boldsymbol{X}_{\Gamma}} - (\boldsymbol{\Psi}^{S})^{T} \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_{\Gamma}} \end{aligned} \tag{4.30}$$

which may be further simplified by the fact that $\frac{\partial \boldsymbol{f}^{M,ext}}{\partial \boldsymbol{X}_\Gamma} = \boldsymbol{0}$ and introducing Eq. 4.7 as:

$$\begin{aligned}\frac{dL}{d\boldsymbol{X}_\Gamma} = & \underbrace{\frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma}}_{\boldsymbol{g}_\Gamma^{*F}} \\ & \underbrace{-(\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{f}^{M,int}}{\partial \boldsymbol{X}_\Gamma}}_{\boldsymbol{g}_\Gamma^{M}} \underbrace{-(\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma}}_{\boldsymbol{g}_\Gamma^{S}} \\ = & \boldsymbol{g}_\Gamma^{*F} + \boldsymbol{g}_\Gamma^{M} + \boldsymbol{g}_\Gamma^{S}\end{aligned} \tag{4.31}$$

where J^* is the force-based functional with the modified force projection vector field (i.e. $\boldsymbol{d}^* = \boldsymbol{d} + \boldsymbol{\Psi}_\Gamma^S$). In the above, one distinguishes between the derivatives w.r.t the deformed interface shape and the gradients w.r.t undeformed interface shape. This is exactly consistent with the formulation of the primal problem.

The following remarks highlight some of the important observations about partitioned adjoint-based sensitivity analysis for FSI using three-field formulation:

Remark 4. Comparing Eq. 4.31 with Eq. 4.11 shows that the closed-form expression for the sensitivity equation is independent of the solution strategy for the adjoint FSI problem.

Remark 5. By careful inspection of Eqs. 4.27-4.31, it turns out that the adjoint-based and domain-based sensitivities of the coupled fluid problem,

$$\boldsymbol{g}^{*F} = \begin{bmatrix} \boldsymbol{g}_\Omega^{*F} \\ \\ \boldsymbol{g}_\Gamma^{*F} \end{bmatrix} = \boldsymbol{f}^{M,a}, \tag{4.32}$$

contribute not only to the coupled adjoint mesh motion problem but also to the coupled shape sensitivities.

Table 4.2 summarizes the findings and outcome of partitioned adjoint-based sensitivity analysis for FSI using three-field formulation. Note that

Table 4.3: Summary of partitioned adjoint-based sensitivity analysis for FSI using two-field formulation

	Discipline	
	Fluid	Structure
Adjoint equation	$\boldsymbol{r}^{F,a} = \left[\frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{w}}\right]^T \cdot \boldsymbol{\Psi}^F + \left(\frac{\partial J^*}{\partial \boldsymbol{w}}\right)^T = \boldsymbol{0}$	$\boldsymbol{r}^{S,a} = \left[\frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{u}^S}\right]^T \cdot \boldsymbol{\Psi}^S - \boldsymbol{f}^{S,a} = \boldsymbol{0}$
Coupling condition	$\boldsymbol{d}^* = \boldsymbol{d} + \boldsymbol{\Psi}^S_\Gamma$	$\boldsymbol{f}^{S,a}_\Gamma = \left(\left(\boldsymbol{f}^{S,ext}_\Gamma\right)^T \cdot \frac{\partial \boldsymbol{d}}{\partial \boldsymbol{u}^S_\Gamma} + \left(\boldsymbol{f}^{F,a}_\Gamma\right)^T\right)^T$, $\boldsymbol{f}^{F,a}_\Gamma = \boldsymbol{g}^{*F}_\Gamma = \left(\frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma}\right)^T$
Sensitivity contribution	$\boldsymbol{g}^{*F}_\Gamma = \frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma}$	$\boldsymbol{g}^S_\Gamma = -(\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma}$
Total sensitivity equation	$\frac{dL}{d\boldsymbol{X}_\Gamma} = \frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma} - (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma} = \boldsymbol{g}^{*F}_\Gamma + \boldsymbol{g}^S_\Gamma = \boldsymbol{g}^{FSI}$	

the coupled adjoint equations are represented in a residual form, e.g., $\boldsymbol{r}^{F,a}$ is the discrete residuals of the coupled adjoint fluid problem. Based on

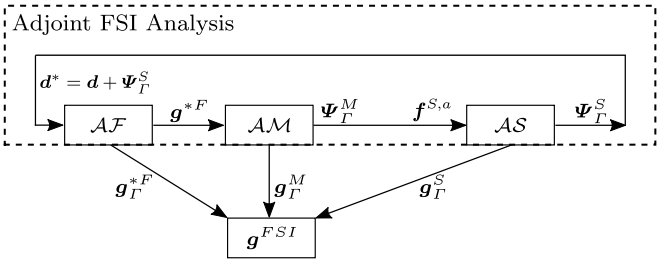


Figure 4.1: Schematic representation of FSI shape sensitivity analysis using black box adjoint solvers and the three-field formulation.

the above discussions and findings, we conclude that there exist adjoint quantities to be exchanged between the adjoint solvers. Analogous to the primal problem (see Section 3.4.1), the partitioned adjoint-based FSI sensitivity analysis in Table 4.2 can be implemented by single disciplinary adjoint solvers in a black box manner. Figure 4.1 illustrates the flows of information in the three-field-based adjoint shape sensitivity analysis, where

- $\mathcal{AF}$ is the adjoint fluid solver that takes the modified force projection vector field as input and outputs the domain-based adjoint sensitivity $\boldsymbol{g}^{*F}$ that contains the internal and the boundary values.
- $\mathcal{AM}$ is the adjoint mesh motion solver whose input is the domain-based fluid adjoint force/sensitivity $\boldsymbol{f}^{M,a} = \boldsymbol{g}^{*F}$ and outputs are the interface fluid adjoint displacement $\boldsymbol{\Psi}_\Gamma^M$ and the adjoint-based gradients of the mesh motion problem w.r.t the interface shape (i.e. $\boldsymbol{g}_\Gamma^M$).
- $\mathcal{AS}$ is the adjoint structure solver which receives the structural adjoint force $\boldsymbol{f}^{S,a}$ and outputs the interface structural adjoint displacement $\boldsymbol{\Psi}_\Gamma^S$ and the adjoint-based gradients of the structure problem w.r.t the interface shape (i.e. $\boldsymbol{g}_\Gamma^S$).

Given non-matching meshes at the interface, Algorithm 3 outlines the steps and required computations in a Gauss-Seidel solution process, where n

denotes the current iteration and $\hat{\boldsymbol{\Psi}}_{\Gamma}^{S}$ is the relaxed interface structural adjoint displacements. In this algorithm, consistent–conservative mapping procedure is employed to transfer the interface structural adjoint displacements consistently and the interface fluid adjoint displacements conservatively.

Algorithm 3 Work flow of a block Gauss-Seidel coupling algorithm for the adjoint FSI problem summarized in Table 4.2

//initialize the mapping matrix between $\boldsymbol{X}_{\Gamma}^{F}$ and $\boldsymbol{X}_{\Gamma}^{S}$
$\boldsymbol{H}^{FS}$
$n = 1$
//initialize interface adjoint structure displacements
${}^{n}\hat{\boldsymbol{\Psi}}_{\Gamma}^{S} = \mathbf{0}$
//adjoint FSI strong coupling loop
while $\left\| {}^{n}\boldsymbol{r}^{\Gamma} \right\|_2 > \varepsilon$ **do**
// map consistently interface adjoint structure displacements and modify force projection vector field
    ${}^{n}\boldsymbol{d}^{*} = \boldsymbol{d} + \boldsymbol{H}^{FS} \cdot {}^{n}\hat{\boldsymbol{\Psi}}_{\Gamma}^{S}$
// solve adjoint fluid problem
    ${}^{n}\boldsymbol{g}^{*F} = \mathcal{AF}({}^{n}\boldsymbol{d}^{*})$
// solve adjoint mesh motion problem
    ${}^{n}\boldsymbol{\Psi}_{\Gamma}^{M} = \mathcal{AM}({}^{n}\boldsymbol{g}^{*F})$
// map conservatively fluid adjoint displacements on the interface and calculate interface structural adjoint force
    ${}^{n}\boldsymbol{f}_{\Gamma}^{S,a} = \frac{\partial \boldsymbol{d}}{\partial \boldsymbol{u}_{\Gamma}^{S}} \cdot \boldsymbol{f}_{\Gamma}^{S,ext} + (\boldsymbol{H}^{FS})^{T} \cdot {}^{n}\boldsymbol{\Psi}_{\Gamma}^{M}$
// solve adjoint structure problem
    ${}^{n}\boldsymbol{\Psi}_{\Gamma}^{S} = \mathcal{AS}({}^{n}\boldsymbol{f}^{S,a})$
// compute adjoint structure displacement residuals on the interface
    ${}^{n}\boldsymbol{\delta} = {}^{n}\boldsymbol{\Psi}_{\Gamma}^{S} - {}^{n}\hat{\boldsymbol{\Psi}}_{\Gamma}^{S}$
    compute ${}^{n+1}\hat{\boldsymbol{\Psi}}_{\Gamma}^{S}$ **based on** $\{{}^{1}\boldsymbol{\delta}, {}^{2}\boldsymbol{\delta}, \cdots, {}^{n}\boldsymbol{\delta}\}$ **and** $\{{}^{1}\boldsymbol{\Psi}_{\Gamma}^{S}, {}^{2}\boldsymbol{\Psi}_{\Gamma}^{S}, \cdots, {}^{n}\boldsymbol{\Psi}_{\Gamma}^{S}\}$ **(relaxation, etc.)**
    $n = n + 1$
end while

4.3.3 Partitioned adjoint FSI by two-field formulation

In many black box adjoint solvers, shape gradients are calculated and accessible only for the surface (boundary) points, and therefore, the domain-based fluid adjoint force $\boldsymbol{f}^{M,a}$ is not available to solve the coupled adjoint mesh motion problem (Eq. 4.27). On the other hand, the mesh motion problem is only an auxiliary problem to maintaining good mesh quality in the fluid mesh deformation step and it essentially does not exist in the continuous form of FSI problem. With the condition that the aero-elastic solution is mesh independent, it is a valid assumption to completely ignore the mesh motion problem in the coupled adjoint sensitivity analysis, the so-called two-field formulation of the adjoint FSI problem. In the case of matching meshes at the interface, the Lagrange function of two-field formulation is defined as follows:

$$\begin{aligned} &L(\boldsymbol{w},\boldsymbol{u}^S,\boldsymbol{x}_\Gamma,\boldsymbol{X}_\Gamma) = \\ &J\left(\boldsymbol{w},\boldsymbol{u}^S,\boldsymbol{x}_\Gamma\right) + (\boldsymbol{\Psi}^F)^T \cdot \boldsymbol{r}^F(\boldsymbol{w},\boldsymbol{x}_\Gamma) + \\ &(\boldsymbol{\Psi}^S)^T \cdot \left[\boldsymbol{f}^{S,ext}\left(\boldsymbol{w},\boldsymbol{x}_\Gamma\right) - \boldsymbol{f}^{S,int}\left(\boldsymbol{u}^S,\boldsymbol{X}_\Gamma\right)\right] \end{aligned} \tag{4.33}$$

Based on two-field formulation, we assume that the variation of the internal fluid displacements with respect to the shape design variables is zero, while no assumption is made about the variation of the fluid displacements on the interface, i.e.

$$\frac{d\boldsymbol{u}_\Omega^F}{d\boldsymbol{X}_\Gamma} \approx [\boldsymbol{0}]_{m_\Omega^F \times m_\Gamma} \tag{4.34a}$$

$$\frac{d\boldsymbol{u}_\Gamma^F}{d\boldsymbol{X}_\Gamma} = \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{X}_\Gamma} \neq [\boldsymbol{0}]_{m_\Gamma \times m_\Gamma} \tag{4.34b}$$

and the total derivative of the geometrical compatibility conditions in Eqs. 3.12f-3.12g is subsequently approximated via

$$\frac{d\boldsymbol{x}_\Omega^F}{d\boldsymbol{X}_\Gamma} = \frac{d\boldsymbol{u}_\Omega^F}{d\boldsymbol{X}_\Gamma} + \frac{d\boldsymbol{X}_\Omega^F}{d\boldsymbol{X}_\Gamma} = \frac{d\boldsymbol{u}_\Omega^F}{d\boldsymbol{X}_\Gamma} \approx [\boldsymbol{0}]_{m_\Omega^F \times m_\Gamma} \tag{4.35a}$$

$$\frac{d\boldsymbol{x}_\Gamma^F}{d\boldsymbol{X}_\Gamma} = \frac{d\boldsymbol{x}_\Gamma}{d\boldsymbol{X}_\Gamma} = \frac{d\boldsymbol{u}_\Gamma^F}{d\boldsymbol{X}_\Gamma} + \frac{d\boldsymbol{X}_\Gamma^F}{d\boldsymbol{X}_\Gamma} = \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{X}_\Gamma} + \boldsymbol{I}_{m_\Gamma \times m_\Gamma} \tag{4.35b}$$

where matching interface meshes is assumed. The gradient of the mesh-independent Lagrangian in Eq. 4.33 with respect to the design variables

reads

$$
\begin{aligned}
&\frac{dL}{d\boldsymbol{X}_\Gamma} = \\
&\left[\frac{\partial J}{\partial \boldsymbol{w}} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{w}} + (\boldsymbol{\Psi}_\Gamma^S)^T \cdot \frac{\partial \boldsymbol{f}_\Gamma^{S,ext}}{\partial \boldsymbol{w}}\right] \cdot \frac{d\boldsymbol{w}}{d\boldsymbol{X}_\Gamma} + \\
&\left[\frac{\partial J}{\partial \boldsymbol{u}^S} + (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}}{\partial \boldsymbol{x}_\Gamma} \cdot \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{u}^S} - (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{u}^S} + \right. \\
&\left.\left(\frac{\partial J}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma}\right) \cdot \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{u}^S}\right] \cdot \frac{d\boldsymbol{u}^S}{d\boldsymbol{X}_\Gamma} + \\
&\left[\frac{\partial J}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}_\Gamma^S)^T \cdot \frac{\partial \boldsymbol{f}_\Gamma^{S,ext}}{\partial \boldsymbol{x}_\Gamma} - (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma}\right]
\end{aligned}
\tag{4.36}
$$

wherein the following equivalent form of Eq. 4.35b is used to group the terms that contribute to the variation of the structural displacements:

$$
\begin{aligned}
&\frac{d\boldsymbol{x}_\Gamma}{d\boldsymbol{X}_\Gamma} = \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{X}_\Gamma} + [\mathbf{1}]_{m_\Gamma \times m_\Gamma} = \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{u}^S} \cdot \frac{d\boldsymbol{u}^S}{d\boldsymbol{X}_\Gamma} + \boldsymbol{I}_{m_\Gamma \times m_\Gamma}, \\
&\frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{u}^S} = \begin{bmatrix} \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{u}_\Omega^S} & \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{u}_\Gamma^S} \end{bmatrix} = \begin{bmatrix} [\mathbf{0}]_{m_\Gamma \times m_\Omega^S} & \boldsymbol{I}_{m_\Gamma \times m_\Gamma} \end{bmatrix}
\end{aligned}
\tag{4.37}
$$

Comparing Eq. 4.36 with Eq. 4.16, there is no update in the coupled adjoint fluid problem, whereas the coupled adjoint structure problem is modified as follows:

$$
\begin{aligned}
&\left[\frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{u}^S}\right]^T \cdot \boldsymbol{\Psi}^S = \\
&\left(\frac{\partial J}{\partial \boldsymbol{u}^S} + (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,ext}}{\partial \boldsymbol{x}_\Gamma} \cdot \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{u}^S} + \right. \\
&\left.\left(\frac{\partial J}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma}\right) \cdot \frac{d\boldsymbol{u}_\Gamma^S}{d\boldsymbol{u}^S}\right)^T = \boldsymbol{f}^{S,a}
\end{aligned}
\tag{4.38}
$$

As before, the right-hand side of the adjoint structure problem is named the structural adjoint force and is represented by $\boldsymbol{f}^{S,a}$. Note that, although the second term on the right-hand side of Eq. 4.38 is a function of the

adjoint displacement vector $\boldsymbol{\Psi}^S$, it is intentionally assigned to the structural adjoint force in order not to modify or in order to reuse the tangent stiffness matrix of the structure. If we reconsider our force-based functional which is defined as the weighted sum of the nodal aerodynamic forces (weights: force projection vector field $\boldsymbol{d}$), then the structural adjont force vector has non-zero values only on the fluid–structure interface Γ i.e.,

$$\begin{aligned}
&\boldsymbol{f}_{\Omega}^{S,a} = \mathbf{0} \\
&\boldsymbol{f}_{\Gamma}^{S,a} = \\
&\left(\left(\boldsymbol{f}_{\Gamma}^{S,ext}\right)^T \cdot \frac{\partial \boldsymbol{d}}{\partial \boldsymbol{u}_{\Gamma}^S} + (\boldsymbol{\Psi}_{\Gamma}^S + \boldsymbol{d})^T \cdot \frac{\partial \boldsymbol{f}_{\Gamma}^{S,ext}}{\partial \boldsymbol{x}_{\Gamma}} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_{\Gamma}} \right)^T
\end{aligned} \tag{4.39}$$

The boundary adjoint force $\boldsymbol{f}_{\Gamma}^{S,a}$ can be further simplified by introducing the modified force-based objective function J^* as

$$\begin{aligned}
\boldsymbol{f}_{\Gamma}^{S,a} &= \left(\left(\boldsymbol{f}_{\Gamma}^{S,ext}\right)^T \cdot \frac{\partial \boldsymbol{d}}{\partial \boldsymbol{u}_{\Gamma}^S} + \left(\boldsymbol{f}_{\Gamma}^{F,a}\right)^T \right)^T \\
\boldsymbol{f}_{\Gamma}^{F,a} &= \left(\frac{\partial J^*}{\partial \boldsymbol{x}_{\Gamma}} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_{\Gamma}} \right)^T
\end{aligned} \tag{4.40}$$

and $\boldsymbol{f}_{\Gamma}^{F,a}$ is termed the boundary-based fluid adjoint force. Knowing that the adjoint-based sensitivity of a general and uncoupled fluid problem w.r.t to the interface boundary reads:

$$\boldsymbol{g}_{\Gamma}^{F} = \frac{\partial J}{\partial \boldsymbol{x}_{\Gamma}} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_{\Gamma}}, \tag{4.41}$$

we can easily verify that the boundary-based fluid adjoint force $\boldsymbol{f}_{\Gamma}^{F,a}$ is basically the adjoint-based and boundary-based sensitivity of the coupled force-based functional J^*, i.e.

$$\boldsymbol{g}_{\Gamma}^{*F} = \boldsymbol{f}_{\Gamma}^{F,a} = \frac{\partial J^*}{\partial \boldsymbol{x}_{\Gamma}} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_{\Gamma}}. \tag{4.42}$$

After having derived the adjoint fluid and structure problems, the coupled sensitivity equation based on two-field formulation can be calculated from

the remaining terms in Eq. 4.36:

$$\frac{dL}{d\boldsymbol{X}_\Gamma} = \underbrace{\frac{\partial \boldsymbol{J}^*}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma}}_{\boldsymbol{g}_\Gamma^{*F}} \underbrace{-(\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma}}_{\boldsymbol{g}_\Gamma^S} \tag{4.43}$$

Algorithm 4 Work flow of a block Gauss-Seidel coupling algorithm for the adjoint FSI problem summarized in Table 4.3

//initialize the mapping matrix between $\boldsymbol{X}_\Gamma^F$ and $\boldsymbol{X}_\Gamma^S$
$\boldsymbol{H}^{FS}$
$n = 1$
//initialize interface adjoint structure displacements
${}^n\hat{\boldsymbol{\Psi}}_\Gamma^S = \boldsymbol{0}$
//adjoint FSI strong coupling loop
while $\left\| {}^n\boldsymbol{r}^\Gamma \right\|_2 > \varepsilon$ **do**
// map consistently interface adjoin structure displacements and modify force projection vector field
  ${}^n\boldsymbol{d}^* = \boldsymbol{d} + \boldsymbol{H}^{FS} \cdot {}^n\hat{\boldsymbol{\Psi}}_\Gamma^S$
// solve adjoint fluid problem
  ${}^n\boldsymbol{g}_\Gamma^F = \mathcal{AF}({}^n\boldsymbol{d}^*)$
// map conservatively boundary-based adjoint fluid forces/sensitivities and calculate interface structural adjoint forces
  ${}^n\boldsymbol{f}_\Gamma^{S,a} = \frac{\partial \boldsymbol{d}}{\partial \boldsymbol{u}_\Gamma^S} \cdot \boldsymbol{f}_\Gamma^{S,ext} + (\boldsymbol{H}^{FS})^T \cdot {}^n\boldsymbol{g}_\Gamma^F$
// solve adjoint structure problem
  ${}^n\boldsymbol{\Psi}_\Gamma^S = \mathcal{AS}({}^n\boldsymbol{f}^{S,a})$
// compute adjoint structure displacement residuals on the interface
  ${}^n\boldsymbol{\delta} = {}^n\boldsymbol{\Psi}_\Gamma^S - {}^n\hat{\boldsymbol{\Psi}}_\Gamma^S$
  compute ${}^{n+1}\hat{\boldsymbol{\Psi}}_\Gamma^S$ **based on** $\{{}^1\boldsymbol{\delta}, {}^2\boldsymbol{\delta}, \cdots, {}^n\boldsymbol{\delta}\}$ **and** $\{{}^1\boldsymbol{\Psi}_\Gamma^S, {}^2\boldsymbol{\Psi}_\Gamma^S, \cdots, {}^n\boldsymbol{\Psi}_\Gamma^S\}$ **(relaxation, etc.)**
  $n = n + 1$
end while

The careful reader have identified that the adjoint-based and boundary-based sensitivities of the coupled force-based functional contribute to both the structural adjoint force and the coupled sensitivity equation. The

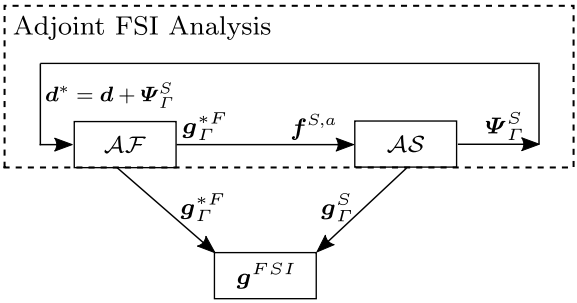


Figure 4.2: Schematic representation of FSI shape sensitivity analysis using black box adjoint solvers and the two-field formulation.

results of the above sensitivity analysis are provided in Table 4.3. Furthermore, Figure 4.2 shows a schematic representation of the work flow of the two-field-based adjoint shape sensitivity analysis. Lastly, Algorithm 4 specifies the steps and required computations with the consistent-conservative treatment of non-matching grids at the interface.

4.4 Continuous Adjoint-based Sensitivity Analysis for Fluids Coupled with Structures

In this section, for the sake of completeness and generality, adjoint-based shape sensitivity analysis is performed continuously for the fluid in an Eulerian reference frame, whether it is fixed or moving (e.g. rotating reference frame). Furthermore, in order to ensure that the above derivations for the coupled adjoint fluid problem is independent of the adjoint approach, the force-based objective functional in Section 4.2 is augmented not only with the flow state equations/residuals through the Lagrange multipliers, but also with an extra force-based functional coming from the structure. The Lagrangian function corresponding to the fluid problem of interest reads

$$\begin{aligned} L^*(\boldsymbol{w},\boldsymbol{x}_\Gamma) &= J(\boldsymbol{x}_\Gamma,\boldsymbol{w}) + \int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \boldsymbol{R}^F(\boldsymbol{x},\boldsymbol{w})\, d\Omega + \int_{\Gamma_x} \boldsymbol{\Psi}^S \cdot \left(p\,\boldsymbol{I} - \mu^F \boldsymbol{\tau}\right) \cdot \boldsymbol{n}^F \; d\Gamma \\ &= J^*(\boldsymbol{x}_\Gamma,\boldsymbol{w}) + \int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \boldsymbol{R}^F(\boldsymbol{x},\boldsymbol{w})\, d\Omega \end{aligned} \tag{4.44}$$

Herein Ω_x^F is the fluid domain in the deformed configuration x; $\boldsymbol{R}^F$ is the governing continuum equations of the fluid; J^* is the total interface force in the direction of $\boldsymbol{d}^*$. Note that the classical continuous Lagrangian function for fluids can be recovered by simply setting $\boldsymbol{\Psi}_\Gamma^S = \boldsymbol{0}$. The total variation of L^* with respect to the deformed shape x hence reads:

$$\delta L^* = \delta_{\boldsymbol{w}} L^* + \delta_{\boldsymbol{x}_\Gamma} L^* \tag{4.45}$$

While $\delta_{\boldsymbol{x}_\Gamma} L^*$ is the shape sensitivity that is being sought, setting $\delta_{\boldsymbol{w}} L^*$ to zero yields a set of fluid adjoint PDEs and boundary conditions. It should be intuitively clear that $\delta_{\boldsymbol{w}} L^* = 0$ is the continuous adjoint formulation to eliminate the variation of the fluid state $\frac{d\boldsymbol{w}}{d\boldsymbol{X}_\Gamma}$ in Eqs. 4.16 and 4.36, i.e.

$$\begin{aligned} &\delta_{\boldsymbol{w}} \left(J^*(\boldsymbol{x},\boldsymbol{w}) + \int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \boldsymbol{R}^F \; d\Omega \right) = \boldsymbol{0} \\ &\approx \\ &\frac{\partial J^*}{\partial \boldsymbol{w}} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{w}} = \boldsymbol{0} \end{aligned} \tag{4.46}$$

The approximation in the above equation becomes an equality in the continuous limit of infinitesimally fine mesh resolution. On the other hand $\delta_{\boldsymbol{x}_\Gamma} L^*$ provides a closed form expression which is a continuous form of the adjoint-based and boundary-based sensitivity of the coupled force-based functional J^* in Eq. 4.42, i.e.

$$\delta_{\boldsymbol{x}_\Gamma}\left(J^*(\boldsymbol{x},\boldsymbol{w})+\int_{\Omega_x^F}(\boldsymbol{\Psi}^F)^T\cdot\boldsymbol{R}^F\ d\Omega\right)\approx \boldsymbol{g}_\Gamma^{*F}=\boldsymbol{f}_\Gamma^{F,a}=\frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma}+(\boldsymbol{\Psi}^F)^T\cdot\frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma} \tag{4.47}$$

An important observation is that not only are the objective function and the fluid governing equations a function of the deformed configuration x but the volume integral also is a function of the deformed configuration x. By the applying the Leibnitz rule, the above equation reads

$$\delta_{\boldsymbol{x}_\Gamma} J^*+\int_{\Omega_x^F}(\boldsymbol{\Psi}^F)^T\cdot\frac{\partial \boldsymbol{R}^F}{\partial \boldsymbol{x}_\Gamma}\ d\Omega+\underbrace{\int_{\Gamma_x}(\boldsymbol{\Psi}^F)^T\cdot\boldsymbol{R}^F\cdot\boldsymbol{n}^F\ d\Gamma}_{\text{LBterm}} \tag{4.48}$$

The boundary integral in the above equation is called LBterm [45]. Note that in the literature, it has been usually assumed that the flow equations are satisfied along the boundary. As a result, LBterm has been ignored, which is true for fine grids. Apart from this assumption, it is difficult to compute the residuals of the Navier–Stokes equations for a viscous fluid along the boundary, especially the second-order spatial derivatives of velocity must be computed along Γ_x, which is not straightforward to be computed accurately. A full discussion on the effect the LBterm on the gradient accuracy of can be found in [45].

4.4.1 Total Variation of Force-based Functional

Following the chain rule theorem, the total variation of the continuous J^* with respect to a shape perturbation at Γ_x is

$$\begin{aligned}\delta J^*=\delta_{\boldsymbol{x}_\Gamma} J^*+\delta_{\boldsymbol{w}} J^*=\ &\delta_{\boldsymbol{x}_\Gamma}\int_{\Gamma_x}\boldsymbol{d}^*\cdot\left(\boldsymbol{I}\,p-\mu^F\boldsymbol{\tau}(\boldsymbol{v})\right)\cdot\boldsymbol{n}^F\ d\Gamma\ +\\ &\int_{\Gamma_x}\boldsymbol{d}^*\cdot\left(\boldsymbol{I}\,\delta p-\mu^F\boldsymbol{\tau}(\boldsymbol{\delta v})\right)\cdot\boldsymbol{n}^F\ d\Gamma\end{aligned} \tag{4.49}$$

Note that the first boundary integral is exactly equivalent to the first term in Eq. 4.48. Here the variation of the effective viscosity has been neglected. This is known as the frozen viscosity assumption, which is commonly followed with the continuous adjoint approach.

Basically three approaches have been proposed in the literature on the calculation of $\delta_{\boldsymbol{x}_\Gamma} J^*$, namely: finite difference, analytic differentiation and differential geometry which is followed here. Following the derivation by T. Economon [22], the total variation of the functional can be simplified to

$$\delta J^* = \int_{\Gamma_x} \left[\boldsymbol{d}^* \cdot \boldsymbol{q} + \nabla \boldsymbol{d}^* : (\boldsymbol{I}\, p - \mu^F \boldsymbol{\tau}) - (\boldsymbol{I}\, p - \mu^F \boldsymbol{\tau}) \cdot \boldsymbol{n}^F \cdot (\boldsymbol{n}^F \cdot \nabla_x) \boldsymbol{d}^* \right] \boldsymbol{n}^F \, d\Gamma$$
$$+ \int_{\Gamma_x} \boldsymbol{d}^* \cdot \left(\boldsymbol{I}\, \delta p - \mu^F \boldsymbol{\tau}(\delta \boldsymbol{v}) \right) \cdot \boldsymbol{n}^F \, d\Gamma \tag{4.50}$$

One should note that in the absence of the source term (i.e. $\boldsymbol{q} = \boldsymbol{0}$) and the spatially constant direction vector, the first boundary integral in Eq. 4.50 vanishes.

4.4.2 Viscous Incompressible Adjoint Flow

This section derives the adjoint fluid system continuously for the viscous incompressible flow (i.e., the Mach number M below 0.3) on the basis of Eq. 4.46, as follows:

$$\delta_{\boldsymbol{v}} J^* + \delta_p J^* + \int_{\Omega_x^F} [\psi^p, \boldsymbol{\psi}^{\boldsymbol{v}}]^T \cdot \delta_p \boldsymbol{R}^F \, d\Omega \; + \int_{\Omega_x^F} [\psi^p, \boldsymbol{\psi}^{\boldsymbol{v}}]^T \cdot \delta_{\boldsymbol{v}} \boldsymbol{R}^F \, d\Omega = \boldsymbol{0} \tag{4.51}$$

where ψ^p and $\boldsymbol{\psi}^{\boldsymbol{v}}$ are the fluid Lagrange multipliers associated with the fluid pressure and velocity respectively. Here, variations with respect to the fluid density and the total energy are neglected respectively under the incompressibility and the low Mach number flow conditions. This equation is the basis for the derivation of the continuous adjoint Navier-Stokes equations. Following the derivation of Othmer [62], the frozen turbulence

assumption is introduced and then an integration by parts of all integrals over the fluid domain is performed as

$$\begin{aligned}
&\delta_{\boldsymbol{v}} J^* + \delta_p J^* + \int_{\partial\Omega_x^F} \left(\boldsymbol{\psi}^{\boldsymbol{v}} \cdot \boldsymbol{n}^F\right) \delta p \; d\Gamma + \int_{\Omega_x^F} \left(-\nabla_x \cdot \boldsymbol{\psi}^{\boldsymbol{v}}\right) \delta p \; d\Omega \\
&+ \int_{\partial\Omega_x^F} \left(\rho^F \boldsymbol{n}^F (\boldsymbol{\psi}^{\boldsymbol{v}} \cdot \boldsymbol{v}) + \rho^F \boldsymbol{\psi}^{\boldsymbol{v}} \left(\left(\boldsymbol{v} - \boldsymbol{v}_\Omega\right) \cdot \boldsymbol{n}^F\right)\right) \cdot \delta \boldsymbol{v} \; d\Gamma \\
&+ \int_{\partial\Omega_x^F} \left(\mu^F \boldsymbol{n}^F \cdot \boldsymbol{\tau}(\boldsymbol{\psi}^{\boldsymbol{v}}) - \psi^p \boldsymbol{n}^F\right) \cdot \delta \boldsymbol{v} \; d\Gamma \\
&- \int_{\partial\Omega_x^F} \mu^F \boldsymbol{n}^F \cdot \boldsymbol{\tau}(\delta \boldsymbol{v}) \cdot \boldsymbol{\psi}^{\boldsymbol{v}} \; d\Gamma \\
&+ \int_{\Omega_x^F} \Big(-\rho^F \nabla_x \boldsymbol{\psi}^{\boldsymbol{v}} \cdot \boldsymbol{v} - \rho^F \left(\left(\boldsymbol{v} - \boldsymbol{v}_\Omega\right) \cdot \nabla_x\right) \boldsymbol{\psi}^{\boldsymbol{v}} - \nabla_x \cdot (\mu^F \boldsymbol{\tau}(\boldsymbol{\psi}^{\boldsymbol{v}})) \\
&+ \nabla_x \psi^p + \boldsymbol{q}^a\Big) \cdot \delta \boldsymbol{v} \; d\Omega
\end{aligned} \tag{4.52}$$

Herein $\partial\Omega_x^F$ represents the complete boundary of the deformed fluid domain including the coupling interface Γ, $\boldsymbol{q}^a = -\boldsymbol{\omega} \times \boldsymbol{\psi}^{\boldsymbol{v}}$ is the adjoint source term. Note that the fluid problem and it's adjoint are formulated in the deformed configuration x; therefore spatial gradients ∇ and the normal vector $\boldsymbol{n}^F$ are calculated in the deformed configuration. Equation 4.52 has to be fulfilled for any $\delta \boldsymbol{v}$ and δp that satisfy the primal Navier-Stokes equations which generally can only be accomplished if the terms vanish individually. For the integrals over the fluid domain this requirement gives rise to the adjoint Navier-Stokes equations:

$$-\rho^F \nabla_x \boldsymbol{\psi}^{\boldsymbol{v}} \cdot \boldsymbol{v} - \rho^F \left(\left(\boldsymbol{v} - \boldsymbol{v}_\Omega\right) \cdot \nabla_x\right) \boldsymbol{\psi}^{\boldsymbol{v}} - \nabla_x \cdot (\mu^F \boldsymbol{\tau}(\boldsymbol{\psi}^{\boldsymbol{v}})) + \nabla_x \psi^p + \boldsymbol{q}^a = \boldsymbol{0} \tag{4.53a}$$

$$\nabla \cdot \boldsymbol{\psi}^{\boldsymbol{v}} = 0 \tag{4.53b}$$

The boundary terms in Eq. 4.52 together with the terms related to a variation of the objective function define the adjoint boundary conditions for the fluid. Table 4.4 lists the adjoint boundary conditions for the force-based objective functional introduced in Section 4.2. For the sake of completeness, here we drive the adjoint boundary conditions on the interface for

the force-based functional. Following the findings in Eqs. 4.50 and 4.52, the remaining boundary terms read on the interface as:

$$\begin{aligned}
&\int_{\Gamma_x} \boldsymbol{d}^* \cdot \left(\boldsymbol{I}\delta p - \mu^F \boldsymbol{\tau}(\boldsymbol{\delta v})\right) \cdot \boldsymbol{n}^F \; d\Gamma + \int_{\Gamma_x} \left(\boldsymbol{\psi^v} \cdot \boldsymbol{n}^F\right) \delta p \; d\Gamma \\
&+ \int_{\Gamma_x} \left(\rho^F \boldsymbol{n}^F (\boldsymbol{\psi^v} \cdot \boldsymbol{v}) + \rho^F \boldsymbol{\psi^v} \left(\left(\boldsymbol{v} - \boldsymbol{v}_\Omega\right) \cdot \boldsymbol{n}^F\right)\right) \cdot \boldsymbol{\delta v} \; d\Gamma \\
&+ \int_{\Gamma_x} \left(\mu^F \boldsymbol{n}^F \cdot \boldsymbol{\tau}(\boldsymbol{\psi^v}) - \psi^p \boldsymbol{n}^F\right) \cdot \boldsymbol{\delta v} \; d\Gamma \\
&- \int_{\Gamma_x} \mu^F \boldsymbol{n}^F \cdot \boldsymbol{\tau}(\boldsymbol{\delta v}) \cdot \boldsymbol{\psi^v} \; d\Gamma = 0
\end{aligned} \tag{4.54}$$

Table 4.4: Common adjoint boundary conditions for the adjoint incompressible Navier–Stokes equations

boundary type	$\boldsymbol{\psi^v}$	ψ^p
interface	$-\boldsymbol{d}^*$	$\boldsymbol{n}_x^F \cdot \nabla_x \psi^p = 0$
wall	$\boldsymbol{0}$	$\boldsymbol{n}_x^F \cdot \nabla_x \psi^p = 0$
inflow	$\boldsymbol{0}$	$\boldsymbol{n}_x^F \cdot \nabla_x \psi^p = 0$
slip	$\boldsymbol{n}_x^F \cdot \boldsymbol{\psi^v} = (\boldsymbol{n}_x^F . \nabla_x)(\boldsymbol{\psi^v} - \boldsymbol{n}_x^F \cdot \boldsymbol{\psi^v}) = 0$	$\boldsymbol{n}_x^F \cdot \nabla_x \psi^p = 0$
outflow	$-\rho^F \psi^p \boldsymbol{n}_x^F + \rho^F \left(\boldsymbol{n}_x^F \cdot \left(\boldsymbol{v} - \boldsymbol{v}_\Omega\right)\right) \boldsymbol{\psi^v} + \mu^F (\boldsymbol{n}_x^F \cdot \nabla_x) \boldsymbol{\psi^v} = \boldsymbol{0}$	

Note that the velocity at the interface is fixed to zero. Therefore $\boldsymbol{\delta v} = \boldsymbol{0}$ on Γ_x and the terms in the second and the third lines of Eq. 4.54 vanish. From the remaining terms we obtain a boundary condition for the adjoint fluid velocity on the coupling interface

$$\boldsymbol{\psi^v} = -\boldsymbol{d}^* \quad \text{on} \quad \Gamma_x \tag{4.55}$$

Note that these derivations do not impose a condition for the adjoint pressure ψ^p on the coupling interface. As Othmer suggests in a single-disciplinary context [62] one can use a zero gradient boundary condition for ψ^p, see table 4.4.

Having computed adjoint variables, one can compute the variation of the fluid residuals with respect to a variation in the interface shape (the volume integral in Eq. 4.48), i.e.:

$$\int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{R}^F}{\partial \boldsymbol{x}_\Gamma} \, d\Omega = \int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \delta_{\boldsymbol{x}_\Gamma} \boldsymbol{R}^F \, d\Omega \tag{4.56}$$

The above volume term can be recasted into a pure boundary integral formulation which may be computed with simple vector operations. The idea and derivation in the following closely follows the approach of Othmer [62] which is based on the work of Soto and Löhner [83]. Starting point for the derivation is the total variation of the fluid governing equations:

$$\delta \boldsymbol{R}^F = \delta_{\boldsymbol{x}_\Gamma} \boldsymbol{R}^F + \delta_{\boldsymbol{v}} \boldsymbol{R}^F + \delta_p \boldsymbol{R}^F = 0 \tag{4.57}$$

It can be rewritten as

$$\delta_{\boldsymbol{x}_\Gamma} \boldsymbol{R}^F = -\delta_{\boldsymbol{v}} \boldsymbol{R}^F - \delta_p \boldsymbol{R}^F \tag{4.58}$$

Introducing this reformulation in Eq. 4.56 yields

$$\int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \delta_{\boldsymbol{x}_\Gamma} \boldsymbol{R}^F \, d\Omega = -\int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \delta_{\boldsymbol{v}} \boldsymbol{R}^F \, d\Omega - \int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \delta_p \boldsymbol{R}^F \, d\Omega \tag{4.59}$$

By performing integration by parts, as done in Eq. 4.52, all volume terms cancel out due to the satisfied adjoint fluid equations. By that we obtain

$$\begin{aligned}
&\int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \delta_{\boldsymbol{x}_\Gamma} \boldsymbol{R}^F \, d\Omega = -\int_{\Gamma_x} \left(\boldsymbol{\psi}^{\boldsymbol{v}} \cdot \boldsymbol{n}^F\right) \delta p \, d\Gamma \\
&- \int_{\Gamma_x} \left(\rho^F \boldsymbol{n}^F (\boldsymbol{\psi}^{\boldsymbol{v}} \cdot \boldsymbol{v}) + \rho^F \boldsymbol{\psi}^{\boldsymbol{v}} \left(\left(\boldsymbol{v} - \boldsymbol{v}_\Omega \right) \cdot \boldsymbol{n}^F \right) \right) \cdot \delta \boldsymbol{v} \, d\Gamma \\
&- \int_{\Gamma_x} \left(\mu^F \boldsymbol{n}^F \cdot \boldsymbol{\tau}(\boldsymbol{\psi}^{\boldsymbol{v}}) - \psi^p \boldsymbol{n}^F \right) \cdot \delta \boldsymbol{v} \, d\Gamma \\
&+ \int_{\Gamma_x} \mu^F \boldsymbol{n}^F \cdot \boldsymbol{\tau}(\delta \boldsymbol{v}) \cdot \boldsymbol{\psi}^{\boldsymbol{v}} \, d\Gamma
\end{aligned} \tag{4.60}$$

In this form $\delta \boldsymbol{v}$ and δp are unknown, which means we have to find an approximation. By knowing that only changes of the shape in surface normal direction affect the fluid residual, a Taylor series expansion to approximate $\delta \boldsymbol{v}$ and δp can be used as follows:

$$\delta \boldsymbol{v} = \beta(\boldsymbol{n}_x^F \cdot \nabla_x)\boldsymbol{v} + \mathcal{O}(\beta^2) \tag{4.61}$$

and

$$\delta p = \beta(\boldsymbol{n}_x^F \cdot \nabla_x)p + \mathcal{O}(\beta^2) \tag{4.62}$$

where $0 < \beta << 1$ is a real constant. Then the last integral expression in Eq. 4.60 can be neglected since it contains second order derivative information which can lead to implementation issues. Following all the preceding comments and consequently applying no-slip conditions for the velocity and zero-gradient conditions for the pressure, i.e. $\boldsymbol{v} = \boldsymbol{0}$ and $(\boldsymbol{n}_x^F \cdot \nabla_x)p = 0$ on Γ_x, the volume integral in Eq. 4.56 reduces to the following boundary integral:

$$\begin{aligned} &\int_{\Omega_x^F} (\boldsymbol{\Psi}^F)^T \cdot \delta_{\boldsymbol{x}_\Gamma} \boldsymbol{R}^F \, d\Omega = \\ &\int_{\Gamma_x} \boldsymbol{n}^F \left[\left(\psi^p \boldsymbol{n}^F - \mu^F \boldsymbol{n}^F \cdot \boldsymbol{\tau}(\boldsymbol{\psi}^{\boldsymbol{v}}) \right) \cdot (\boldsymbol{n}^F \cdot \nabla)\boldsymbol{v} \right] d\Gamma \end{aligned} \tag{4.63}$$

Eventually, $\delta_{\boldsymbol{x}_\Gamma} L^*$ can be computed by combining Eq. 4.63 and the remaining boundary terms from Eq. 4.50 as

$$\begin{aligned} &\delta_{\boldsymbol{x}_\Gamma} L^* = \\ &\int_{\Gamma_x} \left[\boldsymbol{d}^* \cdot \boldsymbol{q} + \nabla \boldsymbol{d}^* : (\boldsymbol{I} p - \mu^F \boldsymbol{\tau}) - (\boldsymbol{I} p - \mu^F \boldsymbol{\tau}) \cdot \boldsymbol{n}^F \cdot (\boldsymbol{n}^F \cdot \nabla_x)\boldsymbol{d}^* \right] \boldsymbol{n}^F \, d\Gamma + \\ &\int_{\Gamma_x} \boldsymbol{n}^F \left[\left(\psi^p \boldsymbol{n}^F - \mu^F \boldsymbol{n}^F \cdot \boldsymbol{\tau}(\boldsymbol{\psi}^{\boldsymbol{v}}) \right) \cdot (\boldsymbol{n}^F \cdot \nabla)\boldsymbol{v} \right] d\Gamma \end{aligned} \tag{4.64}$$

Note that, in the case of zero source terms and a constant force projection vector, the first integral vanishes.

Chapter 5

FSI Shape Sensitivity Analysis using Open-source Software Packages

The applicability of the derived partitioned algorithms 3 and 4 for adjoint-based shape sensitivity analysis of FSI problems is realized using a fully open-source framework. The framework is composed of state of the art multi-disciplinary solvers, namely KRATOS as structural solver and SU2 as fluid solver, which are tightly coupled at the FSI interface by means of the open-source software and general co-simulation environment EMPIRE.

- **KRATOS** [18, 52]: It is a framework for developing multi-disciplinary finite-element-based solvers. Besides a wide range of applicability, it offers a Python-based environment for coupling with third party libraries. Here, KRATOS serves the following tasks:

  (i) direct/primal structural analysis for given external forces acting on the interface.

  (ii) direct/primal pseudo structural analysis for given displacement field on the interface.

(iii) adjoint structural analysis for given external adjoint forces acting on the interface.

(iv) adjoint pseudo structural analysis for given domain-based adjoint forces acting on the fluid domain.

(v) adjoint-based shape sensitivity analysis.

- **SU2** [23, 85]: It is a finite-volume-based framework which has been developed mainly for fluid problems and it has primal and adjoint capabilities for applications ranging from incompressible to supersonic flows, for both steady-state and transient problems. Recent enhancements to SU2 offer high-level Python scripts which provide a wide range of functionalities, namely,

  (i) access to the nodal fluid forces across any boundary of interest.

  (ii) apply displacement field at the boundary nodes and move the internal nodes accordingly using the default mesh motion solver of SU2.

  (iii) access to the adjoint-based sensitivities for the surface and volume nodes of the fluid mesh.

  In addition to the mentioned capabilities, a functionality is introduced at the Python level to allow to set or modify an arbitrary force projection vector field across any boundary of interest for adjoint CFD analysis. This is needed in order to add the adjoint structure displacement field at the interface to the default force projection vector field (see Tables 4.3 and 4.2).

- **EMPIRE (Enhanced Multi Physics Interface Research Engine)** [24, 90]: It is an open-source coupling environment which allows co-simulation with several simulation codes. It also offers field mapping technologies [90] which allow the user to communicate between distinct surface discritizations existing on solvers of different physics. The core of the framework is implemented in C++ and offers APIs to different programing languages including Python.

In this work, owing to ease of manipulation, understanding and availability of Python API from SU2 and KRATOS, the Python API of EMPIRE is extensively used.

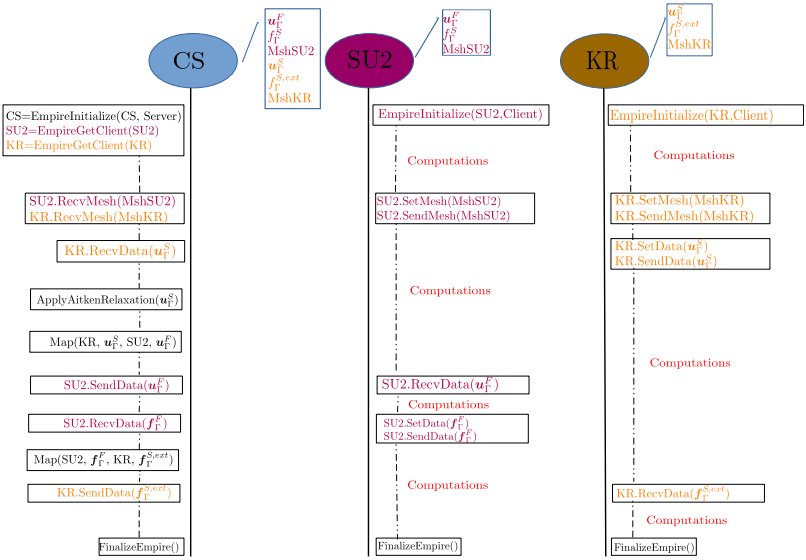


Figure 5.1: Data flow and control statements in CS, SU2 and KRATOS (KR)

EMPIRE employs a hybrid approach that makes it possible to use both client-server and peer-to-peer approaches to do the communication between the clients. This means that any of the clients can act as a server and coordinate the communication and data flow between the clients or the clients can do a peer-to-peer communication when required. This approach combined with the variable based communication makes EMPIRE a versatile framework to perform co-simulation. Here in the case of FSI described before, a Python control script (CS), which is also a client, is created to coordinate the data flow between the client solvers SU2 and KRATOS. The Python scripts used for running the CFD simulation with SU2 and CSD simulation with KRATOS are augmented to include the function calls from EMPIRE framework to do the communication with the control script. Though the same can be achieved by configuring either SU2 or KRATOS to coordinate the data flow, authors chose to create the control script to modularize different aspects of the co-simulation.

Figure 5.1 shows the outline of a simple control flow of the three clients involved in the FSI simulation: the control script (CS) which is the server here, the SU2 for CFD and the KRATOS for CSD. The variable based communication methodology of EMPIRE is evident here as it shows the communication API of EMPIRE framework is independent of the type of the data structure which is being transferred between the clients. Here a MPI based communication is chosen as both the clients SU2 and KRATOS make use of MPI for parallel execution of their respective solutions. The framework also allows users to define custom communication methods specific for the clients. The Python interface of EMPIRE allows direct access of the data structure which is stored in C++, thus facilitating modification of the data stored (data field and mesh information) without excessive overhead. This feature is used to transfer the data on the FSI interface, force field $\boldsymbol{f}_I^F$ from SU2 and displacement field $\boldsymbol{u}_I^S$ from KRATOS to EMPIRE. Finally, after transferring this data to server, it (data field and mesh information) is used to map data fields between different discretizations of fluid and structural interfaces and to communicate the appropriate data fields to the clients. The server which is the control script also employs an Aitken relaxation on the interface displacements $\boldsymbol{u}_I^s$ to accelerate the convergence of the fixed-point iterations between fluid and structure. Since the control script has access to all the data fields at Python level, it is to be noted that implementing, extending and experimenting relaxation schemes is relatively easy and not time consuming. This also facilitates usage of other tools at python level (for example scipy, numpy) together with EMPIRE framework.

The Listing 5.1 shows the illustrative parts of the control script (CS) employing a Gauss-Seidel communication strategy used for FSI simulation. Here the simplicity and readability of the definition of algorithm used are to be noted. The ease of modification of the algorithm by the users allows them to experiment with different possibilities.

```
1  from EMPIRE import *
2  import numpy as np
3  # Usage of independent modules together with EMPIRE
4  from quasi_newton_methods import aitken
5
6  initEMPIRE('CS','server')
7
8  KR_client.recvMesh('MshKR','KR_client')
9  SU2_client.recvMesh('MshSU2','SU2_client')
10
11 # Field needed for mapper definition
12 SU2_client.recvDataField('FD', 'SU2_client')
13 SU2_client.recvDataField('FP','SU2_client')
14
15 KR_client.recvDataField('SD','KR_client')
```

```
KR_client.recvDataField('SP','KR_client')

##Set up mappers
mapperSF = FEMapper.FEMapper('mapper1','Mortar',KR_client, 'MshKR', SU2_client, 'MshSU2' )

# Settings for FSI fixed point iterations
converged = 0
tolerance = 1E-6
iteration = 1
maxIter = 50
# Acceleration scheme
alg = aitken.Aitken( init_alpha = 0.5, init_alpha_max = 0.7)

# Direct access of the C++ data from python
# Setting SD to zero
for i in range(len(KR_client.dataField('SD').array)):
    KR_client.dataField('SD').array[i] = 0.0

while (not converged):
  #FSI Loop
  print ('|||  FSI Iteration :: ', iteration,' |||')

  # Mapping displacments from KR to SU2
  mapperSF.doConsistentMapping(KR_client, KR_client.dataField('SD'),SU2_client, SU2_client
      .dataField('FD'))

  # Sending mapped displacements to SU2
  SU2_client.sendDataField('FD', 'SU2_client')

  # Receiving pressure forces from SU2
  SU2_client.recvDataField('FP','SU2_client')

  # Mapping pressure forces from SU2 to KR
  mapperSF.doConservativeMapping(SU2_client, SU2_client.dataField('FP'),KR_client,
      KR_client.dataField('SP'))

  SDold = np.copy(KR_client.dataField('SD').array)
  # Receiving new displacements from KR
  KR_client.recvDataField('SD', 'KR_client')

  # Checking for convergence
  diff = KR_client.dataField('SD') - SDold
  # Saving the old difference
  if(iteration <= 1):
      diff_old = diff

  diffNorm = np.linalg.norm(diff)
  print ('\tNorm of the residue is :: ', diffNorm)
  if(diffNorm < tolerance or iteration>=maxIter):
      converged = true;

  #--- relaxation ---
  KR_client.dataField('SD') = SDold + alg.ComputeUpdate(diff, SDold)

  # Sending the convergence signal
  SU2_client.sendConvergenceSignal(converged);
  KR_client.sendConvergenceSignal(converged);
  iteration = iteration + 1
  diff_old = diff
  #--- END of FSI Loop
```

Listing 5.1: Code snippet of control script (CS) for FSI simulation

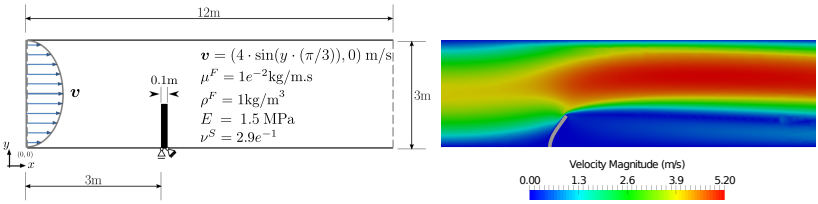
Chapter 6

Numerical Studies

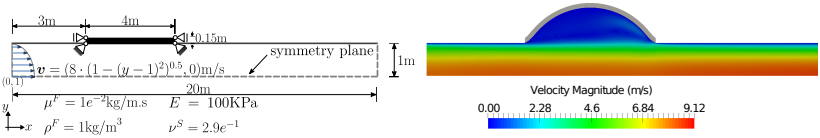
This chapter demonstrates that the derivations for coupled adjoint FSI analysis in Chapter 4 are general and independent of the discretization method, the mathematical modelling of the flow, and last but not least, the adjoint approach. For this purpose, three adjoint FSI frameworks are considered:

- **Fully FEM-based using discrete adjoints**
- **Hybrid FEM-FVM-based using discrete-continuous hybrid adjoints**
- **Hybrid FEM-FVM-based using discrete adjoints**

The first one is used for partitioned and monolithic analyses of FSI problems dealing with incompressible viscous fluids, while the third is employed to investigate the partitioned solution methods for primal and adjoint analyses of a flexible ONERA M6 wing immersed in a compressible inviscid flow. Here, FEM-based analyses, including primal and adjoint-based shape sensitivity analyses for fluids and structures, are performed using KRATOS. Whereas, SU2 and OpenFOAM are used for discrete and continuous based adjoint analyses, respectively. The primal and adjoint



(a) Case 1: Flexible beam in a channel.



(b) Case 2: Channel flow with a flexible wall section.

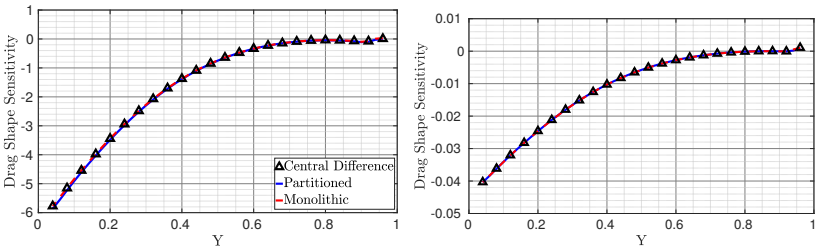
Figure 6.1: Test cases and their steady-state FSI solutions.

couplings between solvers are realized by EMPIRE as explained in the previous chapter.

6.1 FEM-based shape sensitivity analysis for FSI

As pointed out in the introduction, this work provides comprehensive discussions on the theoretical and crucial aspects of shape sensitivity analysis for FSI problems. For this purpose, two case studies in which an incompressible flow interacts with a highly flexible beam are considered. The computational model geometry, the flow boundary conditions, and the material and structural properties are clearly detailed for each case in Figure 6.1. The cases are computed in 2D, and consist of a cantilever beam immersed in a low-Reynolds number flow ($Re_{case1} \approx 570, Re_{case2} \approx 1200$). In both cases, the fluid is modelled by the incompressible Navier-Stokes equations and the beam is modelled by a hyperelastic continuum under plain-strain conditions.

A stabilized finite element method based on SUPG/PSPG [87] stabilization with first order triangular elements is used for the spatial discretization of the fluid, while the structure domain is discretized with standard quadrilateral elements. For the sake of error reduction, the fluid and structure



(a) $Re \approx 570$. (b) $Re \approx 5$.

Figure 6.2: Comparison of the FSI shape sensitivities computed using partitioned and monolithic strategies against finite differences at the front region of the interface in case 1.

domains are discretized with a conforming interface mesh. KRATOS finite element framework is used to perform all numerical experiments and verifications in this section.

The steady-state FSI solutions of both cases are depicted in Figure 6.1. As can be seen, both structures undergo large displacements due to laminar flow fields. For instance, in the test case 1, the horizontal displacement at the tip of the beam is about 50% of the cantilever length. It is also observed that the drag force on the structure drops by 38% through the transition from the undeformed state to the equilibrium state. Also, the strain energy of the structure varies from practically 0 N.m in the undeformed state to 1.1 Nm in the equilibrium state.

6.1.1 Numerical verification

Verification of the adjoint-based shape sensitivity analyses in Chapter 4 is numerically performed against the central difference method. Consider the test case 1, and the shape design variables are chosen to be the X coordinate of the grid points on the front region of the undeformed interface. As the target function for the shape sensitivity analysis, the total drag force acting on the cantilever beam is measured in the equilibrium steady state. Furthermore, in order to examine the impact of the flow regime on the quality of the shape sensitivities, the flow and material properties in case 1 are modified such that the same structural displacement field is achieved

for a very low Reynolds flow ($Re \approx 5$). Figure 6.2 shows the comparisons between the adjoint-based shape sensitivities and the central difference approximation to shape gradients. A very close agreement between different approaches to shape sensitivity analysis confirms the validity of the derivation of both the partitioned and the monolithic adjoint FSI problems in the previous section. From these results it is clear that the quality of the derived shape gradients is not affected by the flow regime.

6.1.2 Comparison of alternative approaches in shape sensitivity analysis

This section collects and compares scenarios that are commonly found in the literature for the shape sensitivity analysis of aeroelastic problems. For example, a pragmatic approach to performing a shape sensitivity analysis of an FSI problem is simply to use the adjoint-based shape sensitivities of the fluid problem in the equilibrium state between the interacting domains [11]. This means that the adjoint fluid problem is solved on the deformed fluid domain without coupling to the adjoint structure and then the shape sensitivities of the fluid are used for gradient-based shape optimization. Although this approach has been successful in the sense of the objective improvement, one has to compare the fully coupled sensitivities to the uncoupled ones because differences in shape sensitivity cause differences in optimal shapes. In the following, it is explored whether coupled adjoint-based shape sensitivities are necessary in both test cases. For this purpose, the strain energy of the structure as well as the relevant components of the total force exerted on the wet interface are considered for the coming comparisons. Remember that the fluid and structure meshes have matching interfaces and the interface nodes are chosen as design variables for the shape sensitivity analysis i.e.

$$\boldsymbol{X}_{\Gamma}^{F} = \boldsymbol{X}_{\Gamma}^{S} = \boldsymbol{X}_{\Gamma} \tag{6.1}$$

We consider the following four scenarios for comparison purposes:

- Scenario 1: The structure is assumed to be rigid. This results in zero adjoint fluid and structure displacement fields (i.e. $\boldsymbol{\Psi}^{S} = \boldsymbol{\Psi}^{M} = \boldsymbol{0}$). Meaning, only the fluid shape sensitivities, computed in the unde-

formed configuration, are taken into account, i.e.

$$\frac{dL}{d\boldsymbol{X}_\Gamma} = \frac{\partial J(\boldsymbol{w},\boldsymbol{X}_\Gamma)}{\partial \boldsymbol{X}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F(\boldsymbol{w},\boldsymbol{X}_\Gamma)}{\partial \boldsymbol{X}_\Gamma} \tag{6.2}$$

This scenario is considered in order to investigate the impact of the structural elasticity on the shape sensitivities.

- Scenario 2: The structure is flexible and it undergoes large deformations. The coupling is only performed for the primal FSI problem and the adjoint problems are computed separately without interface coupling. The corresponding shape sensitivity equation reads

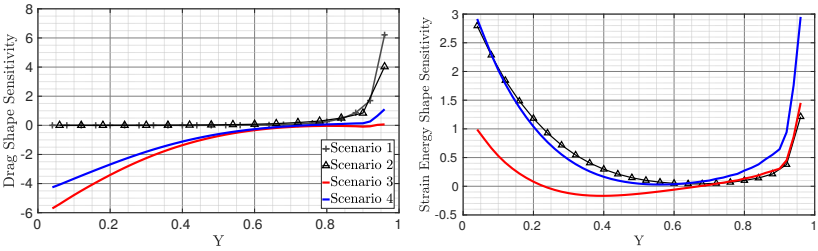
$$\frac{dL}{d\boldsymbol{X}_\Gamma} \simeq \frac{\partial J}{\partial \boldsymbol{x}_\Gamma} + (\tilde{\boldsymbol{\Psi}}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma} - (\tilde{\boldsymbol{\Psi}}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma} \tag{6.3}$$

where $\tilde{\boldsymbol{\Psi}}^F$ and $\tilde{\boldsymbol{\Psi}}^S$ are uncoupled adjoint variables associated with the fluid and the structure, respectively. Note that if a pure aerodynamic force like drag force is chosen as the objective, the term multiplying by the adjoint structure displacement vanishes from the sensitivity equation. The reason is that there is no source term from the objective function ($\frac{\partial J}{\partial \boldsymbol{u}^S} = \boldsymbol{0}$) or from the adjoint fluid variables. The reader may want to consult Tables 4.2-4.3.

- Scenario 3: The structure is highly flexible like in scenario 2 and the coupling is not only performed for the primal FSI problem but also for the adjoint FSI problem. For this scenario, the complete formulation of the adjoint FSI problem is used, including the mesh motion problem (see Table 4.2). The fully coupled adjoint-based sensitivities read

$$\frac{dL}{d\boldsymbol{X}_\Gamma} = \frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma} - (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma} - (\boldsymbol{\Psi}^M)^T \cdot \frac{\partial \boldsymbol{f}^{M,int}}{\partial \boldsymbol{X}_\Gamma} \tag{6.4}$$

Remember that this equation is the final form of the sensitivity equation and it is independent of the coupling strategy (i.e. partitioned or monolithic) used to compute the coupled adjoint variables.



(a) Total drag force.

(b) Strain energy.

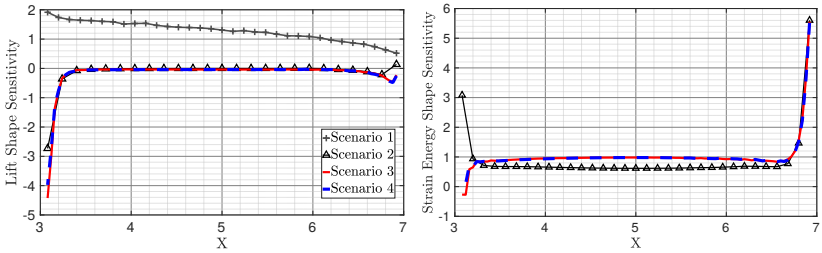
Figure 6.3: Gradient comparison study of scenarios for case 1.

- Scenario 4: The FSI problem is the same as that of scenarios 3 and 2 but shape sensitivites are calculated using the two-field formulation as the following (see Table 4.3):

$$\frac{dL}{d\boldsymbol{X}_\Gamma} = \frac{\partial J^*}{\partial \boldsymbol{x}_\Gamma} + (\boldsymbol{\Psi}^F)^T \cdot \frac{\partial \boldsymbol{r}^F}{\partial \boldsymbol{x}_\Gamma} - (\boldsymbol{\Psi}^S)^T \cdot \frac{\partial \boldsymbol{f}^{S,int}}{\partial \boldsymbol{X}_\Gamma} \tag{6.5}$$

The strain energy of the structure as well as the relevant components of the total force exerted on the wet interface are considered for the coming comparisons. Figure 6.3 contains all the scenarios for the shape sensitivity analysis along the front boundary of the wet interface in case 1. It is worth noting that the plotted sensitivities are gradients with respect to the X component of the interface nodal coordinates at the undeformed configuration.

Furthermore, since the structure is assumed to be rigid in scenario 1, the shape gradients of strain energy are not plotted. By visual inspection of the drag sensitivities it is seen that scenarios 1 and 2 provide gradient directions which are pointing to irrelevant directions compared to scenario 3 (the reference), especially close to the tip of the beam. On the other hand, scenario 4 which excludes the mesh motion problem from the adjoint sensitivity analysis provides a very close but not totally accurate search direction. From a practical point of view, however, scenario 4 is accurate enough for the gradient-based shape optimization. In the case of the strain energy objective function, it is observed that scenarios 2 and 4 behave similarly and close to the reference, and scenario 2 performs better than scenario 4 close to the tip of the beam.



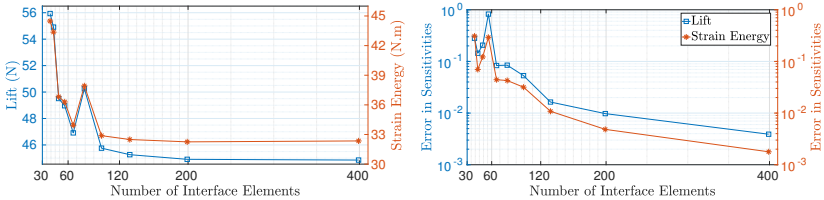
(a) Total lift force. (b) Strain energy.

Figure 6.4: Gradient comparison study of scenarios for case 2.

The results of test case 2 are reported in Figure 6.4. It is noted that pure CFD shape gradients of lift over the rigid interface (scenario 1) not only have a different magnitude but the sign is also different from the reference. By comparing the results of the lift objective with those of the strain energy, a definite and general observation about scenario 2 cannot be made. However, there is an excellent agreement between the shape sensitivities of scenario 4 and the reference (scenario 3) for both objective functions.

6.1.3 Three-field-based vs. two-field-based shape sensitivity analyses

As previously mentioned, the fluid mesh motion problem is only an auxiliary mean to maintain good mesh quality in the fluid mesh deformation step. In fact, the discrete solution must be completely independent of the mesh and the mesh motion of choice. Therefore, for a mesh-independent discrete solution of an FSI system, the shape gradients computed with both formulations should be identical. In order to illustrate the influence of mesh resolution on the inconsistencies between the shape gradients of the two-field formulation and those of the three-field formulation, mesh convergence study is performed for the test case 2. The relative L_2-error of shape sensitivities between two-field formulation and three-field formulation (reference) as well as the convergence histories of the objective functions are depicted in Fig. 6.5. As expected, the objective functions converge to constant values and the relative errors at convergence are less



(a) History of objective functions throughout mesh refinement.

(b) Relative error between two-field-based and three-field-based shape sensitivity analyses versus number of interface elements.

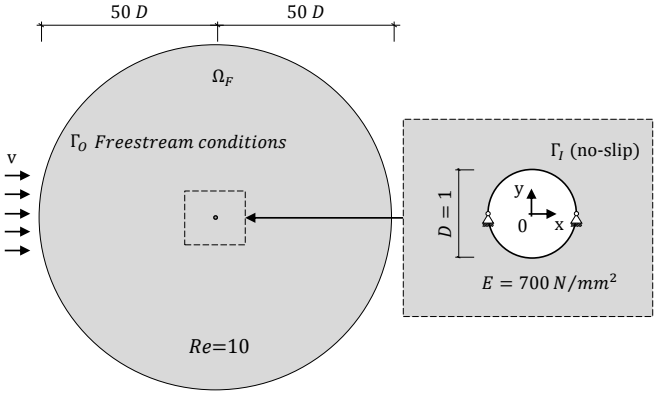
Figure 6.5: Mesh convergence studies for test case 2.

than 1%, as the mesh is refined. Therefore, provided that the FSI solution is mesh-independent, the mesh motion problem can be safely neglected in the coupled-adjoint shape sensitivity analysis.

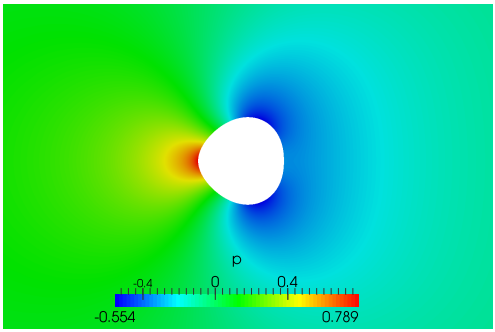
6.1.4 Conclusions

The following summarizes the conclusions from the above studies:

- The flexibility of the wet interface has a significant impact not only on the objective value but also on the objective shape gradients (in the sense of pattern, sign, and magnitude).
- It was observed that pure adjoint CFD sensitivities may have a different magnitude and in some cases opposite sign than the sensitivities obtained with a fully-coupled adjoint analysis. Hence, it becomes necessary to perform a coupled sensitivity analysis.
- It was observed that discrepancies between shape gradients computed with three-field and two-field formulations are associated with mesh resolution. In fact, the derivatives obtained through both formulations should match in the continuous limit of infinitesimally fine mesh resolution. Altogether, the two-field formulation of the adjoint FSI appears to be a valid trade off between the desired level of accuracy, and the availability of the domain-based adjoint sensitivities of the fluid.



(a) Computational model.



(b) Pressure field around deformed shell after FSI analysis (Close-up view).

Figure 6.6: Cylinder shell in laminar fluid flow - test case for continuous–discrete adjoint hybrid shape sensitivity analysis.

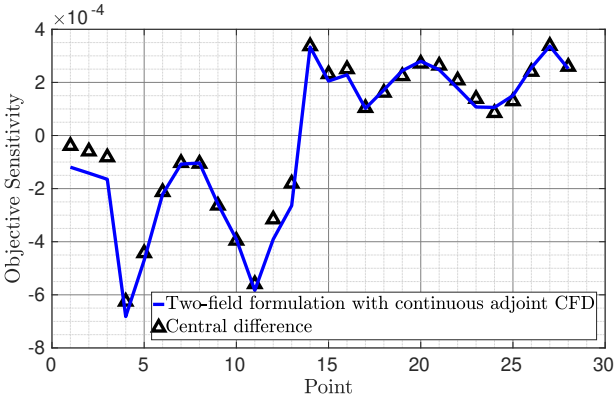
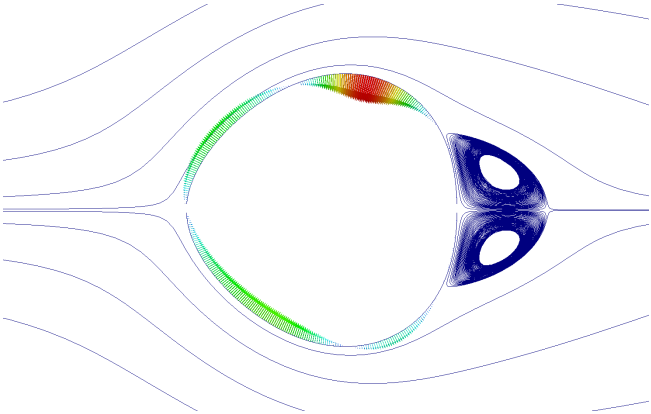


Figure 6.7: A direct comparison of the shape gradients over the cylinder for combined lift and drag objective. Interface primal and adjoint residuals were converged 10 orders of magnitude. Perturbations and sensitivities are projected onto X-component. The finite difference step was $e-5$.

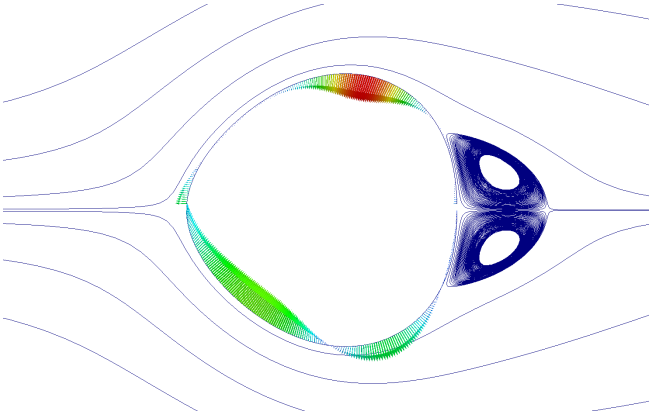
6.2 A continuous–discrete adjoint hybrid shape sensitivity analysis of a flexible body in a laminar flow

In this section, the two-field-based adjoint shape sensitivity analysis (section 4.3.3) is performed using the continuous adjoint formulation for fluid (section 4.4) and the discrete adjoint formulation for the structure. The intention is to ensure that the derived adjoint coupling conditions and the total sensitivity equation (Table 4.3) are independent of the adjoint formulation for the fluid. To this end, a model in which a non-linear cylinder shell is immersed in a laminar incompressible fluid flow ($Re = 10$) is used. Figure 6.6 specifies the model and presents the resulting pressure field after fluid-structure interaction analysis. Turbulence is intentionally excluded due to the included frozen turbulence assumption, which is to be investigated in follow-up research. To amplify the observed effects, shape sensitivity analysis is performed for a straightforward multi-objective optimization which combines lift maximization with drag minimization in a common objective using an equal weighting for both, i.e. $\boldsymbol{d}_i = (-1, 1, 0)$.

Considering matching interface meshes, Figure 6.7 compares two-field-based adjoint shape sensitivities against the reference for a number of



(a) Shape sensitivity vectors computed by uncoupled adjoint FSI analysis.



(b) Shape sensitivity vectors computed by fully coupled adjoint FSI analysis.

Figure 6.8: Comparison of uncoupled and coupled shape sensitivities.

points on the wet interface. Although discrepancies in magnitude is observed, the derivatives obtained with the two-field approach follow a very similar pattern. Importantly, Figure 6.8 illustrates the necessity of the coupled adjoint FSI analysis.

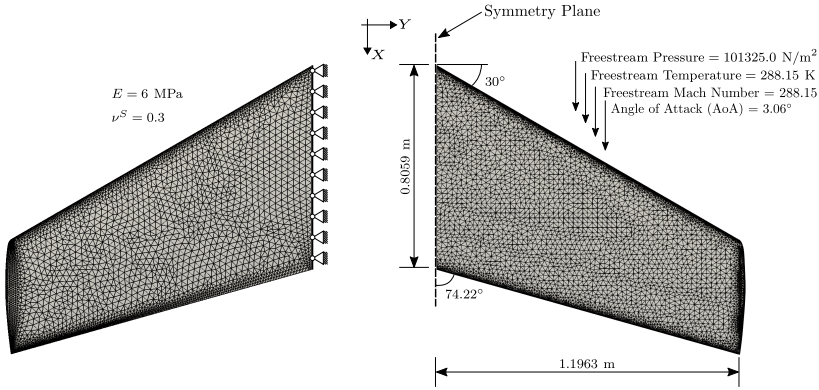


Figure 6.9: Description and surface discretization of ONERA M6 for FSI. Left: structural model, right: fluid model

6.3 Hybrid FEM-FVM-based shape sensitivity analysis of flexible ONERA M6

Using the framework of EMPIRE-KRATOS-SU2, Algorithms 3 and 4 are applied for the adjoint-based shape sensitivity analysis of the ONERA M6 wing immersed in a compressible inviscid fluid flow. In contrast to the usual analyses in the literature, we do not consider the wing to be rigid, but model it as a flexible solid structure clamped at the wing root, Figure 6.9. Doing so, an artificial fluid-structure interaction is introduced in the model so that the corresponding shape sensitivity analysis became an aeroelastic problem. The rather simple wing structure is chosen since the focus here is on the performance of the approaches derived in Chapter 4 for the shape sensitivity analysis of large scale FSI problems. For both the fluid analysis (CFD) and the structural analysis (CSD), steady cruise conditions are assumed. The details of the fluid and structural models are provided in the following paragraphs.

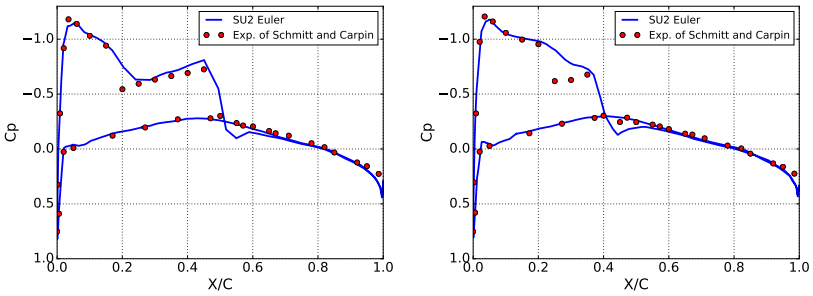
6.3.1 Fluid model

The steady-state transonic flow over the ONERA M6 wing at Mach 0.8395 and angle of attack of 3.06 degrees is computed using non-linear Euler

equations. A tetrahedral grid composed of 582,752 total elements and 108,396 nodes is used for the inviscid simulation. Figure 6.9 demonstrates a close-up view of the unstructured CFD surface mesh of the wing. The boundary conditions for the computational domain are the following: Euler slip condition on the wing surface, a symmetry plane to reflect the flow about the wing root to mimic the effect of the full wing planform, and a characteristic-based condition at a cubical far-field boundary. A second-order JST scheme is used to calculate the convective flux for the mean flow equations. Implicit, local time-stepping is used to converge the problem to the steady-state solution, and the linear system is solved using a flexible variant of GMRES (FGMRES).

6.3.2 CFD validation studies

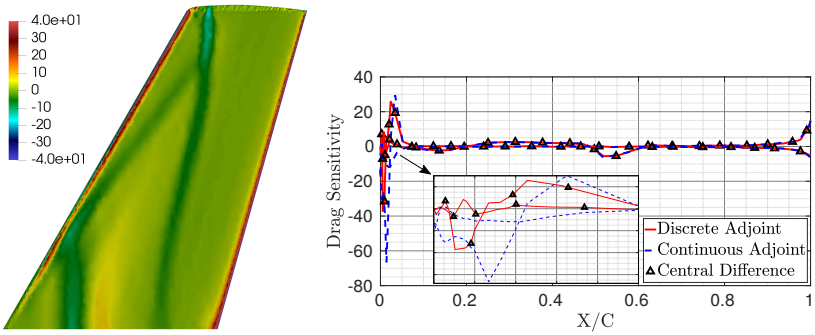
Although SU2 is comprehensively verified and validated in [66], for the sake of completeness, direct and adjoint Euler solvers from SU2 are verified and validated against the experimental data and the central difference approximation, respectively. Assuming a rigid wing structure, Figure 6.10 shows surface pressure coefficient distributions at two different span-wise stations of the wing. Overall, the computed results are in good agreement with the experimental data from [81], particularly along the lower surface and leading edge. In the next step, comparison is done for the shape gradients of the wing drag with respect to a set of the grid points lying at the wing span station Y/b = 0.65 given by the continuous adjoint, the AD-based discrete adjoint, and the central differencing. The gradient verification appears in Figure 6.11. This comparison is of great importance with respect to the validation of the above derived procedures for a partitioned adjoint FSI analysis. Based on Tables 4.3 and 4.2 which summarize coupling conditions for adjoints, boundary and domain-based adjoint sensitivities of the fluid for a force-based objective function form not only an important part of the structural adjoint force but also a part of the final sensitivity equation. Thus, any inaccuracy in the CFD gradient field intensifies errors in the FSI gradient field. As seen in the figure and as expected, the AD-based discrete adjoint and finite differencing gradients exhibit excellent agreement, while the continuous adjoint is in relatively close agreement with the reference. The reader is reffred to [2, 22, 23, 65] for details about the derivation and implementation of the adjoint solvers in SU2.



(a) Y/b = 0.65.

(b) Y/b = 0.8.

Figure 6.10: Comparison of Cp profiles from the experimental results of Schmitt and Carpin (blue circles) against SU2 computational results at different sections along the span of the wing, b.



(a) Discrete adjoint-based surface sensitivity contour for a drag objective function (upper surface).

(b) Comparison of the gradients at the section at 0.65 of the span.

Figure 6.11: Surface sensitivity and validation studies for a rigid ONERA M6 wing (upper surface). Perturbations and sensitivities are projected onto the surface normal.

6.3.3 Structural model

The wing structure is modelled as a solid using 4-node tetrahedral non-linear solid elements which allow the wing to undergo large deformations. For the purposes of the following studies, two finite-element meshes of the wing are used: First, an unstructured grid which consists of 16,446 nodes and 64,246 elements with a nonmatching interface discretization, Figure 6.9. Second, an unstructured grid which has a matching interface with the fluid mesh and it consists of 30,569 nodes and 123,245 elements. The later mesh serves as the validation of the coupled aero-structural sensitivities since it removes the mapping error at the interface, while the former is used for the assessment of the mapping algorithms and criteria for non-matching meshes in the FSI and adjoint FSI. It is assumed that the wing undergoes large deformations and it is made of hyperelastic material characterized by a Young's modulus E = 6MPa and Poisson ratio $\nu^S = 0.3$.

From the point view of the solution strategy, the Newton-Raphson iterative solver from KRATOS is used for the non-linear analysis (direct problem) and the linear static solver with a pseudo load vector (structural adjoint force $\boldsymbol{f}^{S,a}$) is used for the adjoint problem. Note that the structural equations are self-adjoint, since the nonlinear system tangent stiffness matrix (linearized about the system solution) is symmetric.

6.3.4 Steady-state aeroelastic analysis

Having set up the fluid and structural models in the baseline configuration, the steady-state aeroelastic solution was achieved by applying the primal/direct coupling conditions to the individual domains as boundary conditions, the so-called Dirichlet-Neumann partitioning (Algorithm 2).

In the following, investigation is carried out for the difference in accuracy between the mapping approaches, in particular *consistent-conservative* and *consistent-consistent*, as defined in Section 3.4.1. This is done for the nearest element interpolation and an enhanced version of the standard mortar method which have, among other things, similar formulation and popularity in practice. The standard mortar method [13] is based on minimizing the L2 norm of the deviation between two fields using the Galerkin method. The enhanced mortar method [91] solves the inconsistency problem of mortar methods at the curved edges of fluid-structure interfaces. In the nearest element interpolation, mapping is performed by projection

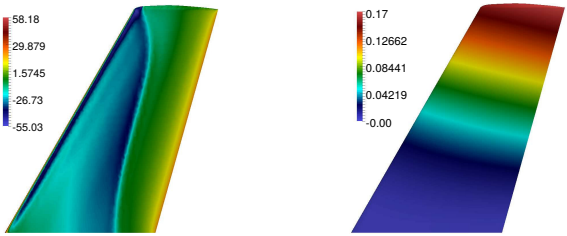
onto the nearest element and interpolation within it. For details, including derivation and critical comparison of several methods, the reader is referred to [16, 90].

First, the static aeroelastic analysis is performed with the matching discrete interfaces and its solution is taken as reference for the upcoming numerical comparisons. An important observation was that the drag and lift coefficients varied from 0.011739 and 0.286269 to 0.00502 and 0.1813, respectively, through the transition from the undeformed state to the equilibrium state. This highlights the importance of including the aero-elasticity and subsequent aeroelastic coupling for wing design and optimization.

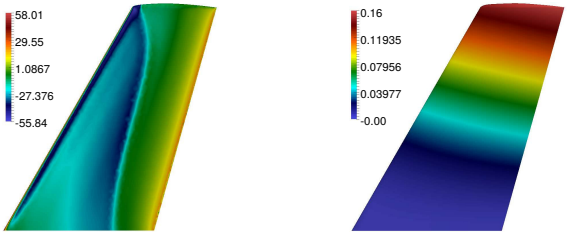
In the case of non-matching meshes (see Fig. 6.9), consistent mapping of displacement is carried out with both algorithms. The tractions/forces are mapped both consistently and conservatively using the nearest element interpolation but they are only mapped conservatively with the enhanced mortar methods. A literature review of the studies by various authors shows that the mortar methods are used in a *consistent-conservative* way. Furthermore, it has been reported that the conservative mapping of forces with the nearest element interpolation gives highly oscillatory results.

Figures 6.12 shows the interface traction field on the structure mesh and the interface displacement field on the fluid mesh for the conforming and non-conforming interfaces. Note that the plotted fields are evaluated at the static aero-elastic equilibrium of the wing. With nearest element interpolation, the consistent traction mapping gives reasonable results while the conservative traction mapping leads to unphysical oscillations in the pressure received by the structure. But force and energy are only approximately conserved in the former case. In comparison to the conforming meshes case, a satisfactory traction field on the structure mesh is obtained from the conservative mapping with the enhanced mortar method. As can be seen from a comparison of plots in Fig.6.12, both mapping algorithms in the static aeroelastic analysis give rise to almost the same displacement field on the fluid mesh, independent of the criterion used for the mapping of tractions. It is also observed a maximum deviation of 6% from the reference.

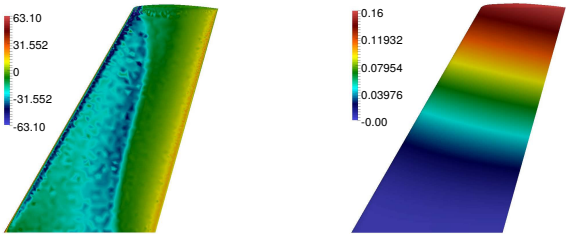
The obtained aeroelastic analysis results for the cases presented in Figure 6.12 are detailed in Table 6.1. From this table, it is clearly observed that the *consistent-consistent* mapping approach resulted in difference in the



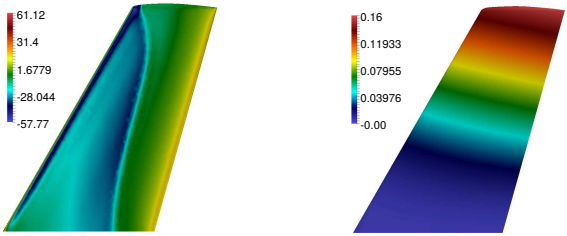
(a) Conforming interface.



(b) *consistent-consistent* mapping with nearest element interpolation.



(c) *consistent-conservative* mapping with nearest element interpolation.



(d) *consistent-conservative* mapping with enhanced mortar method.

Figure 6.12: Converged steady-state traction and displacement fields on the conforming and non-conforming interfaces. Left: interface traction field on the structure mesh, Right: interface Z displacement field on the fluid mesh.

Table 6.1: Static aero-elastic analysis of flexible ONERA M6

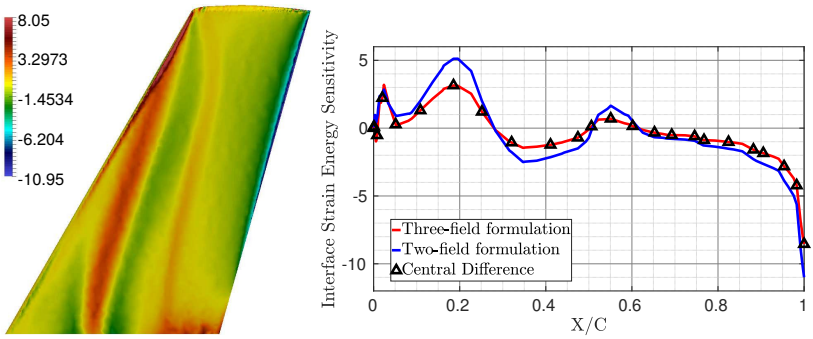
Case	Drag coefficient	Lift coefficient	Tip deflection, m	Interface energy on X_I^F, N.m	Interface energy on X_I^S, N.m	Number of Gauss-Seidel iterations, n
(a) Conforming interface	$5.052e^{-3}$	0.1813	0.1688	0.2880	0.2880	24
(b) *consistent-consistent* mapping with nearest element interpolation	$5.342e^{-3}$	0.1884	0.1593	0.2651	0.2251	18
(c) *consistent-conservative* mapping with nearest element interpolation	$5.355e^{-3}$	0.1887	0.1592	0.2886	0.2886	21
(d) *consistent-conservative* mapping with enhanced mortar method	$5.351e^{-3}$	0.1886	0.1592	0.2885	0.2885	22

interface energy. Nevertheless, it can be observed that all three approaches have the same aerodynamic and aeroelastic behaviors, like aerodynamic force coefficients.

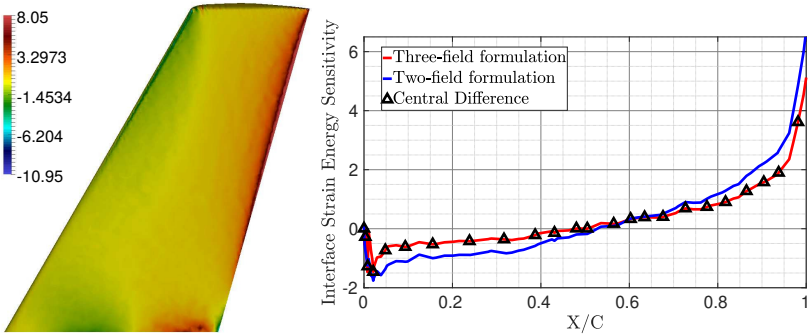
6.3.5 Shape sensitivity analysis

In the following paragraphs, attention is paid to different aspects of the derived partitioned strategies in Sections 4.3.2 and 4.3.3 for solving the adjoint problem of the flexible ONERA M6 wing in the framework from EMPIRE alongside KRATOS and SU2. As a first step, evaluation is done in terms of the accuracy of three-field and two field-formulations for the adjoint-based shape sensitivity analysis against finite difference shape gradient. Considering matching interface meshes, Figures 6.13 and 6.14, respectively, display the strain energy and drag sensitivity maps using three-field formulation for the adjoint FSI. They also compare both formulations against the reference for cross-section Y/b = 0.65. Although a discrepancy in magnitude is observed between the two formulations, the derivatives obtained with the two-field approach follow a very similar pattern.

Apart from the accuracy issue, the convergence behavior of the coupled adjoint analysis needs to be investigated and compared to the coupled primal/direct analysis. Figure 6.15 presents in a semi-logarithmic diagram the convergence histories of the interface displacement and the interface adjoint displacements for both formulations. As could be expected, the three-field-based adjoint FSI problem has faster and smoother conver-

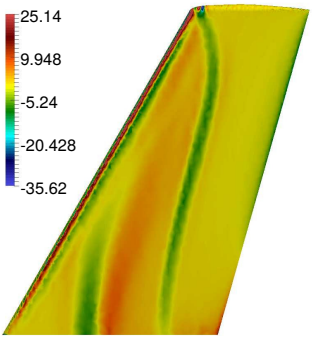


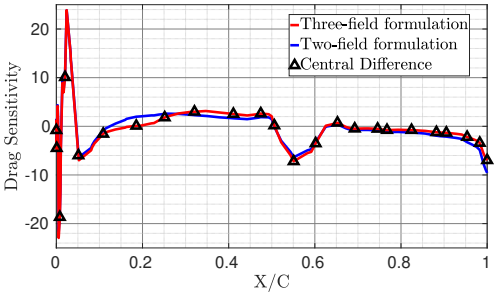
(a) Upper surface.



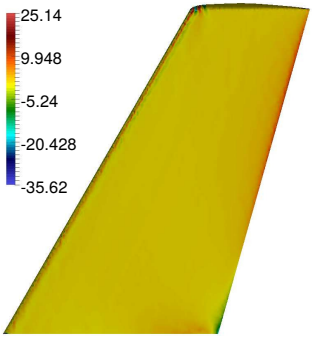
(b) Lower surface.

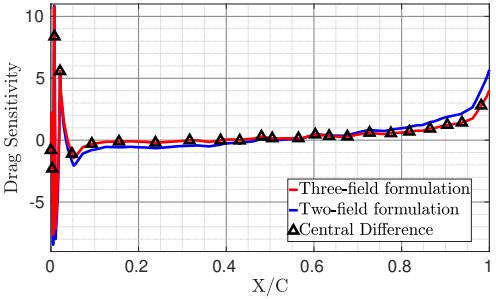
Figure 6.13: Coupled shape sensitivity analysis for the interface strain energy. Left: three-field-based surface sensitivity contours. Right: comparison of the gradients at the section at Y/b=0.65.





(a) Upper surface.





(b) Lower surface.

Figure 6.14: Coupled shape sensitivity analysis for the drag objective. Left: three-field-based surface sensitivity contours. Right: comparison of the gradients at the section at Y/b=0.65.

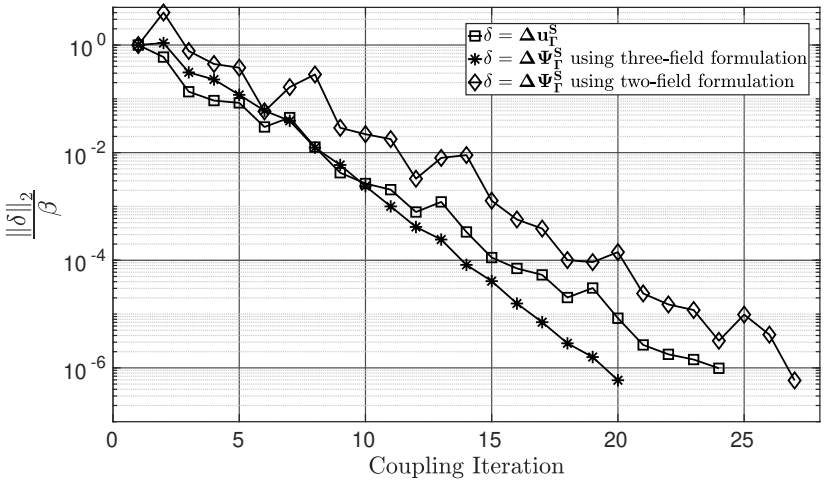
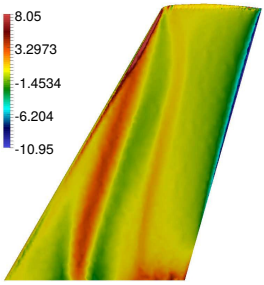


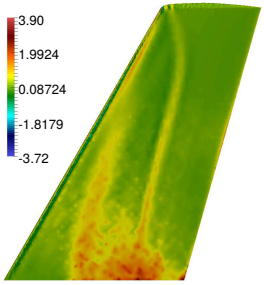
Figure 6.15: Interface residuals for the direct and adjoint FSI problems. β is the L2-norm of the interface residual vector $\boldsymbol{\delta}$ at the first iteration.

gence properties than the FSI and two-field-based adjoint FSI problems. This is due to the facts that FSI is a nonlinear problem whereas adjoint FSI is a linear problem, and also that, unlike the two-field formulation, the three-field formulation consistently derives the adjoint of FSI problem (i.e. without any assumption).

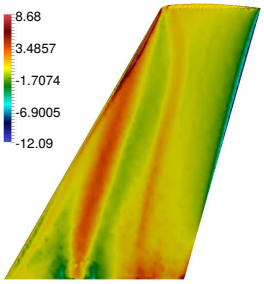
Last but not least, comparison is made between conservative and consistent approaches for the adjoint coupling of non-matching interface meshes for the two mapping techniques used in the primal analysis. For this purpose, studies are done for the same cases as the ones mentioned in Section 6.3.4. For each of these cases, the surface sensitivity map of the strain energy objective is provided in Figure 6.16. Conforming interface results and *consistent-conservative* results for both mapping techniques are in good agreement and a similar pattern can be observed visually. For the sake of clarity, Figure 6.17 also displays the surface sensitivity profiles along Y/b=0.65 for each case. We observe that the *consistent-conservative* mapping is the only choice for the accurate coupling of adjoint variables across non-matching fluid and structure interfaces. However, this was not the case for the coupling of primal variables. Furthermore, it is reasonable



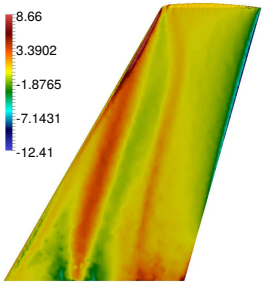
(a) Conforming interface.



(b) *consistent-consistent* mapping with nearest element interpolation.



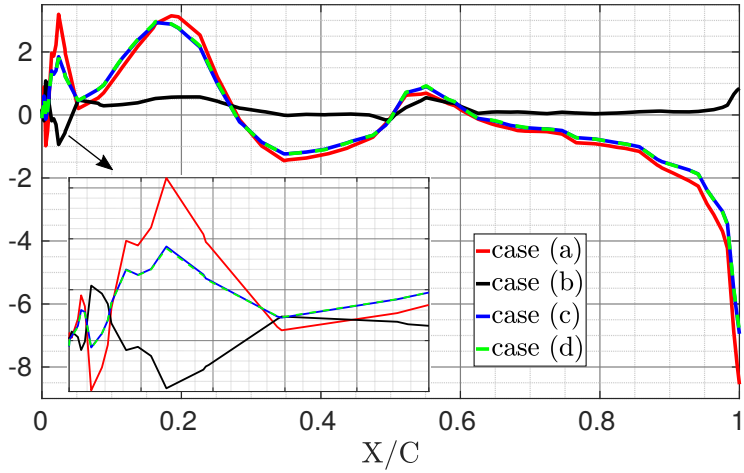
(c) *consistent-conservative* mapping with nearest element interpolation.



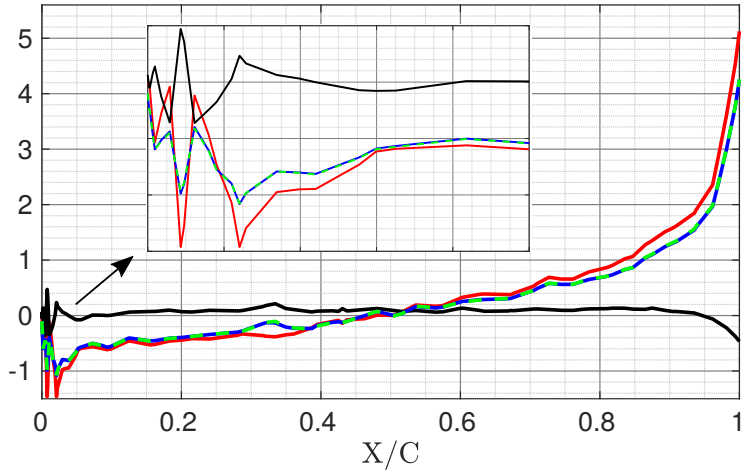
(d) *consistent-conservative* mapping with enhanced mortar method.

Figure 6.16: Strain energy sensitivity contours on the fluid mesh at the upper surface of the flexible ONERA M6 for conforming and non-conforming meshes.

to explain the discrepancies between the shape sensitivities of the conforming interface case and those of the *consistent-conservative* case by mapping errors introduced in both direct and adjoint problems. Recall that we observed a 6% deviation of the tip displacement in cases (c),(d). Based on the foregoing discussion, we conclude that the conservative mapping for the interface force and its adjoint along with consistent mapping for the interface structural displacement and its adjoint lead to a sufficiently accurate shape sensitivity.



(a) Upper surface.



(b) Lower surface.

Figure 6.17: Profiles of strain energy shape sensitivity in Fig. 6.16 at Y/b=0.65.

Chapter 7

Optimal Shape Design Applications

This chapter presents a variety of applications of Vertex Morphing technique to single-/multi-disciplinary shape optimization problems, ranging from internal and external incompressible flow applications to a transonic wing. The optimization approach used here is illustrated in Algorithm 1, which is based on an iterative application of a steepest descent algorithm with a constant step size. Apart from the state equations which are treated as equality constraints and satisfied by adjoint problem, geometric constraints like packaging and frozen/fixed points are applied using gradient projection technique as detailed in Chapter 2.

7.1 CFD Shape Optimization

7.1.1 Unconstrained: Sandia CX-100 Blade

To demonstrate the applicability and reliability of the non-inertial viscous adjoint system derived in 4.4.2, we make use of the Sandia CX-100 blade [93]. The details of the blade geometry definition can be found in [50]. The

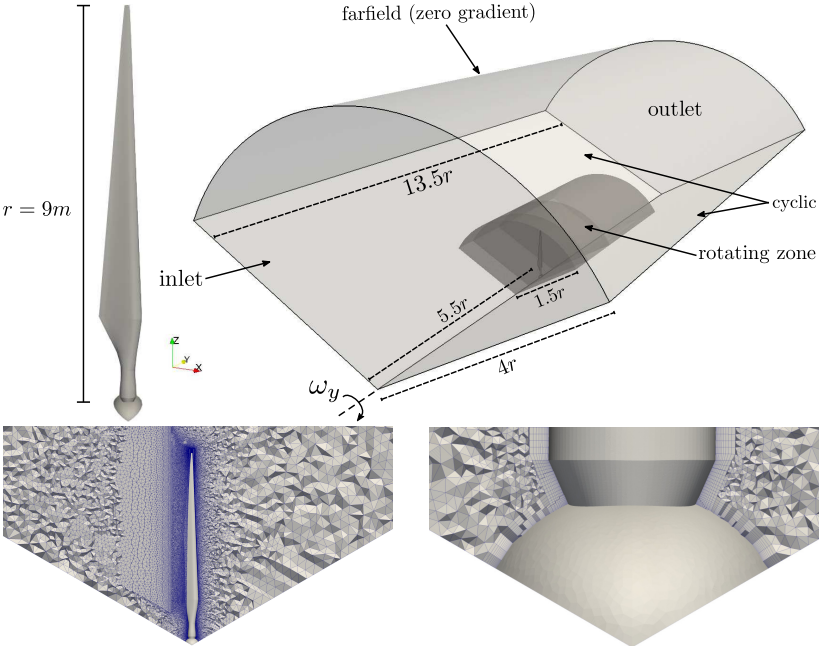
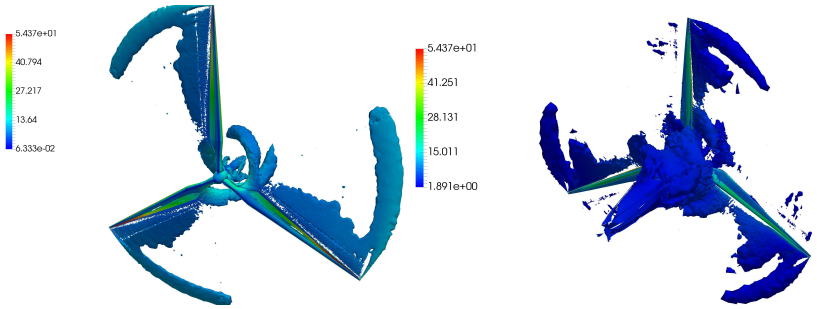


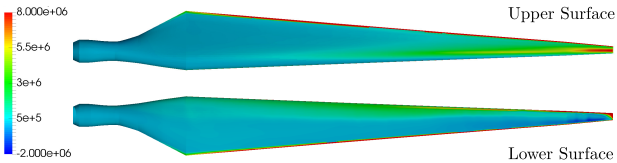
Figure 7.1: Rotationally-periodic computational model and mesh.

setup of the computational model and the problem mesh including a cross-sectional grid and a close up view near the hub are represented in Figure 7.1. For computational efficiency, only one-third of the rotor is modeled. The mesh generated by Pointwise has a resolved boundary layer with avg. Y plus 55 and consists of 7.5 million cells. The blade rotates about the y-axis with a constant rotating speed of 55 rpm and we assume the free stream wind flows normal to the rotation plane of the blade with a velocity of 10 m/s. The Spalart-Allmaras turbulence model is used for the steady-state RANS primal problem, while the turbulent viscosity is assumed frozen (frozen turbulence assumption) for the adjoint problem. Furthermore, the blade is assumed rigid, therefore there is no need for coupling physics and adjoints. The computed aerodynamic power of a single blade is 12.3 kW which compares well with the average value obtained from on-land experimental tests in [93].

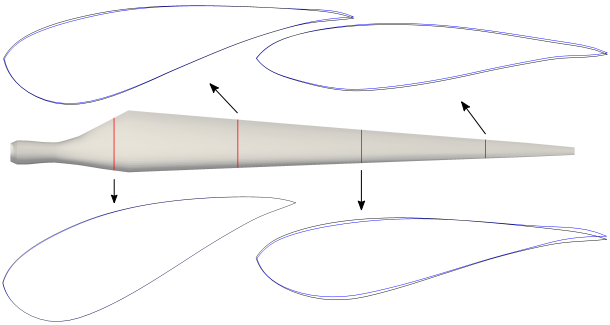


(a) Primal/state velocity. (b) Adjoint velocity.

Figure 7.2: Q-criterion isosurfaces for Q=5.



(a) Surface sensitivity contours of aerodynamic power.



(b) Cross-sections of the optimized blade.

Figure 7.3: Shape sensitivity analysis and optimization of CX-100 blade.

To improve the aerodynamic efficiency of the rotor, we have chosen the maximum power production of the Sandia CX-100 blade as objective function. The vortices in the flow and adjoint flow fields are visualized in Figure 7.2 by the iso-surfaces of the Q-criterion. As expected, primal and adjoint flows rotate and are convected in opposite directions. The whole blade surface was subject to optimization and after 20 iterations, around 4% improvement was achieved. Figure 7.3 displays the surface sensitivity field and certain cross-sections of the optimized blade.

Expectedly, it is observe that the most obvious shape variations happen around the trailing edge while it remains sharp. This is due to the facts that sharp edges have a large impact on the aerodynamic characteristics of the blade, and also that, Vertex Morphing parametrization technique preserves feature lines like sharp edges for a proper choice of filter radii.

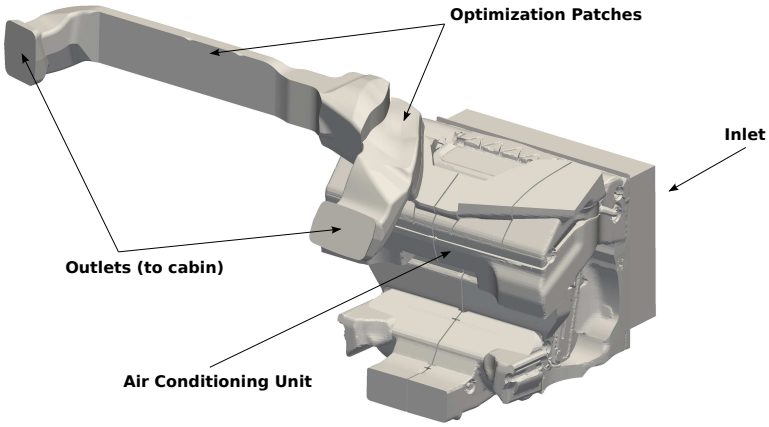


Figure 7.4: HVAC model.

7.1.2 Geometrically Constrained: BMW Air Ducts

By applying the presented framework to heating, ventilating, and air conditioning (HVAC) air ducts, three performance measures improved significantly in the presence of bounding surfaces. Fig. 7.4 represents the computational model including essential flow boundary conditions and

design surfaces. Fig. 7.5 provides the discretization of the design surfaces and the given bounding surface.

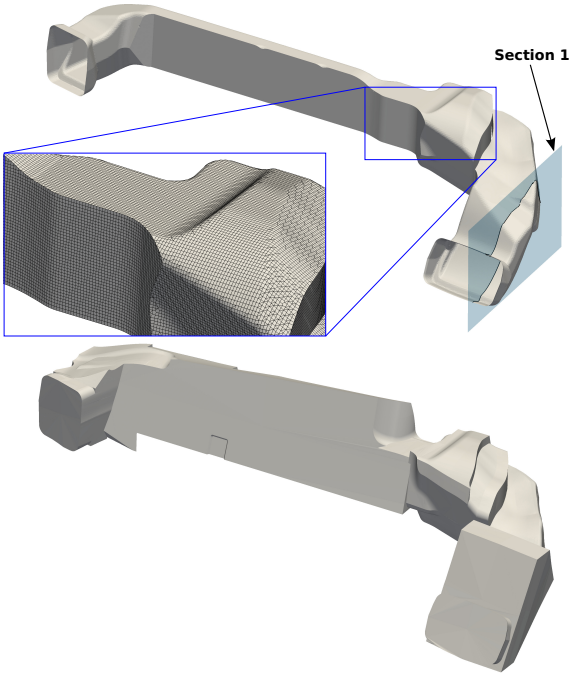


Figure 7.5: Geometry, discretization and selected cross section of the air ducts (top) and the geometry of the bounding surface (bottom).

The three objective functions of interest are air pressure loss, equal mass flow rate in the two ducts and flow uniformity in the outlets. As a common practice, the objectives are combined linearly into a single objective function using fixed weights w_i:

$$J = \int_{\Pi_{i,o1,o2}} (w_1\, p_t + w_2\, \rho^F\, (\boldsymbol{v}\,.\,\boldsymbol{n}^F))\; d\Pi + \int_{\Pi_{o1,o2}} w_3\, \|\boldsymbol{v} - \boldsymbol{v}^d\|^2\; d\Pi \qquad (7.1)$$

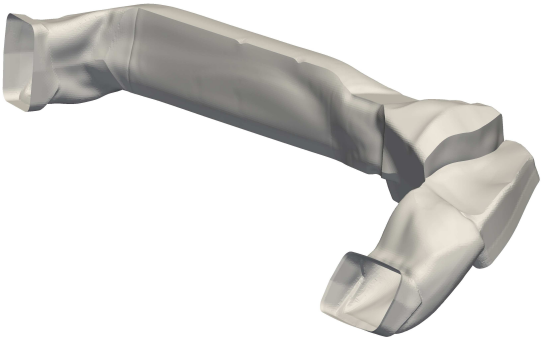


Figure 7.6: Constrained shape optimization of HVAC air ducts with a small filter radius (r).

Herein, Π_i and $\Pi_{o1,o2}$ are the inlet boundary of the HVAC and the outlet boundaries of the two ducts, respectively, p_t is the total fluid pressure, ρ is the fluid density, $\boldsymbol{v}$ is the fluid velocity, and $\boldsymbol{v}^d$ is the desired velocity in the outlet plane.

Table 7.1: Summary of the multi-objective optimization of the BMW air ducts

	Small filter radius (r)		Medium filter radius (r)	
Cost function	Unconstrained	Constrained	Unconstrained	Constrained
Total power loss	32 %	17.5 %	28 %	10%
Mass flow rate	56.5%	97%	56.5%	88%
Flow uniformity	43.5 %	32%	33.5%	25%

The outcome of the described optimization problem is summarized in Table 7.1, and the optimal design is shown in Fig. 7.6. Here, a relatively small filter radius is chosen to explore smaller features and high curvatures. In contrast to the pressure loss and the mass flow rate, the flow uniformity can be visualized by means of velocity vector and velocity contour at the outlets (see Fig. 7.4 for the definition of the outlets). Fig. 7.7 compares the velocity distribution at the outlets between the initial design and the

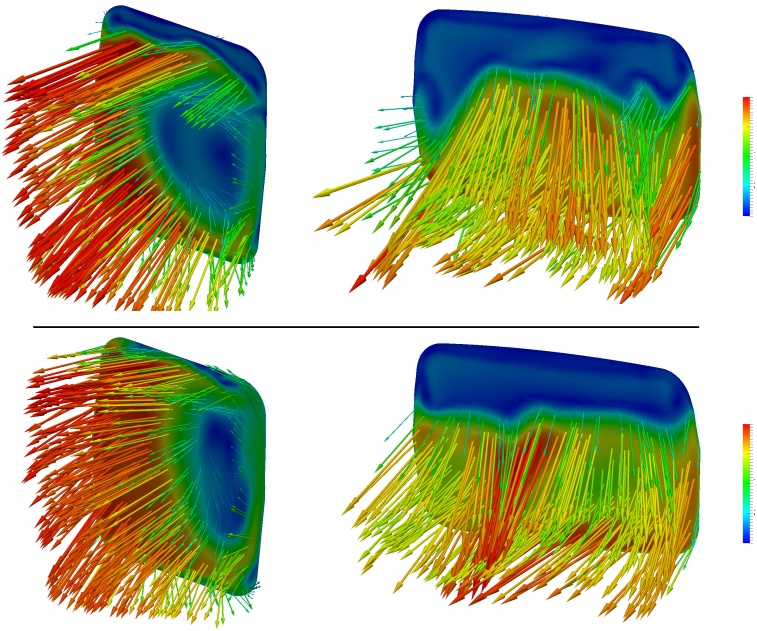


Figure 7.7: Velocity contours and vectors (colored by the magnitude of velocity) at the outlets. Initial distribution (top) and optimal distribution (bottom).

optimal design. It is observed that the very low velocity region (blue region) is minimized, specially at the left outlet.

Apart from the high-quality shape in terms of surface smoothness and manufacturability, Fig. 7.8 verifies the feasibility of the new design at a cross section of the right air duct (see Fig. 7.5). As marked in the figure, three regions can be identified for a critical and comparative assessment. Region 1: where the initial shape is overlapping with the bounding surface and stays feasible throughout the optimization. Region 2: where the initial shape is inside the feasible domain and then finally overlaps with the bounding surface. Region 3: where a very smooth transition occurs at the

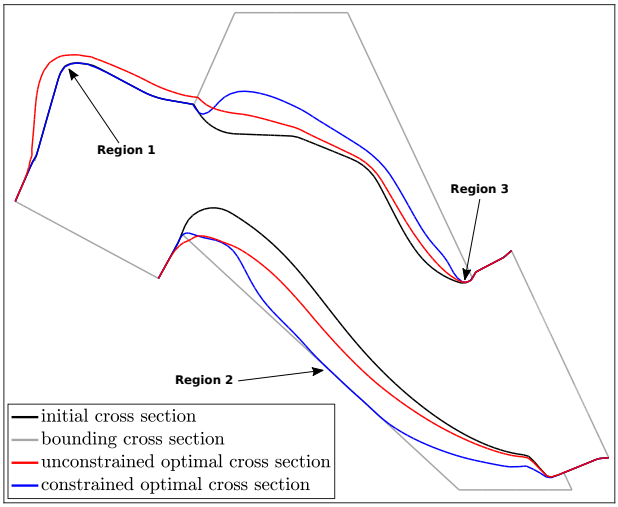


Figure 7.8: Comparison of geometrically constrained and unconstrained optimizations at Section 1.

border of the region overlapping with the bounding surface and the region which is free to move.

7.1.3 Geometrically Constrained: BMW Defroster

This section presents a successful application of the developed framework to the nozzle of a BMW defroster. The nozzle is responsible for clearing condensation and thawing frost from the windshield by blowing high velocity hot air. The main performance requirement for a defroster is fast and uniform defrosting from the bottom of the windshield to its top. Based on experimental investigations and physical interpretation of the above-mentioned requirements, wall shear stress over the windshield is chosen as the optimality criteria for indicating the efficiency of the defroster. In a similar study, Germanou et al. [31] performed an unconstrained CAD-based shape optimization of a passenger car defroster nozzle. In their study, the performance of a defroster nozzle is translated into the integral of the difference between the air velocity and the target (desirable) velocity over

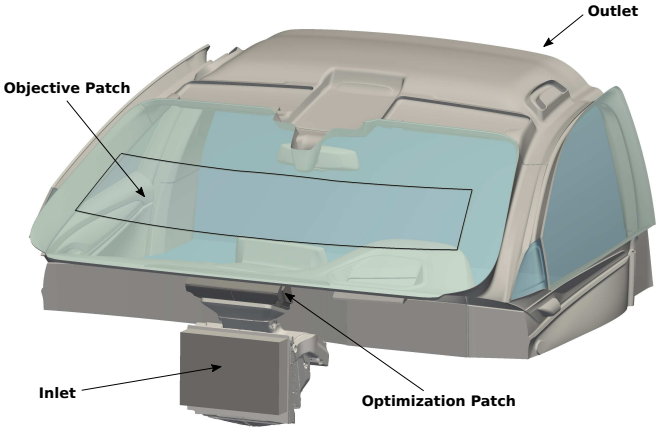


Figure 7.9: Half-model of car cabin.

a thin control volume appropriately defined close to the windshield and inside the car cabin.

Fig. 7.9 specifies the model including essential flow boundary conditions, the design surface, and the objective surface. No-slip conditions are set for the non-specified patches in the figure. Additionally, Fig. 7.10 shows a close-up view of the nozzle and the specified bounding surface.

Apart from the geometrical constraint, this example is a particularly challenging problem in the sense that the objective function has to reflect special requirements. Following the adjoint approach for shape sensitivity analysis, the following scalar-valued objective function is developed to condense the point-wise (face-wise in a finite volume discretization) shear stress maximization problem into a single objective function that has to be minimized:

$$J = \int_{\Pi} \frac{1}{\|(\tau_w + \alpha)\mathbf{t}\|} \, d\Pi \tag{7.2}$$

where Π is the objective patch (see Fig. 7.9); τ_w and α are the wall shear stress magnitude and the regularization term respectively; $\mathbf{t}$ is the surface unit tangent vector. α is introduced to avoid numerical problems resulting

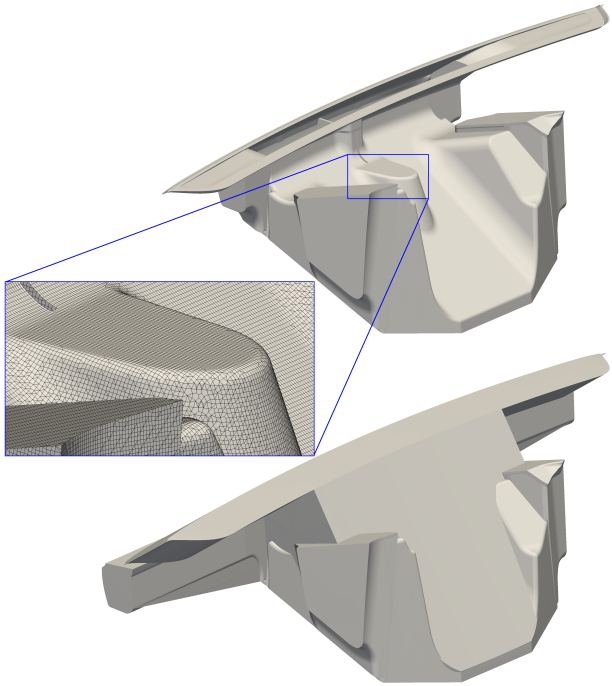


Figure 7.10: Geometry and discretization of the defroster nozzle (top) and bounding surface (bottom).

from a very small magnitude of the wall shear stress in the denominator of the objective function. It is also notable that by using finite volume discretization, the face with less wall shear stress gets more attention from the optimizer. Note that the shape gradients of the developed objective function are calculated using the continuous adjoint solver of Helyx® solvers and we skip the derivation of adjoint equations and boundary conditions. The reader interested in sensitivity analysis is referred to [43, 44, 62, 64].

Figs. 7.11 and 7.12 represent the change in the design surface and the objective surface, respectively. It is observed that major changes in the shape have resulted in significant improvements in both the pattern and the mean value of the wall shear stress over the windshield. To demon-

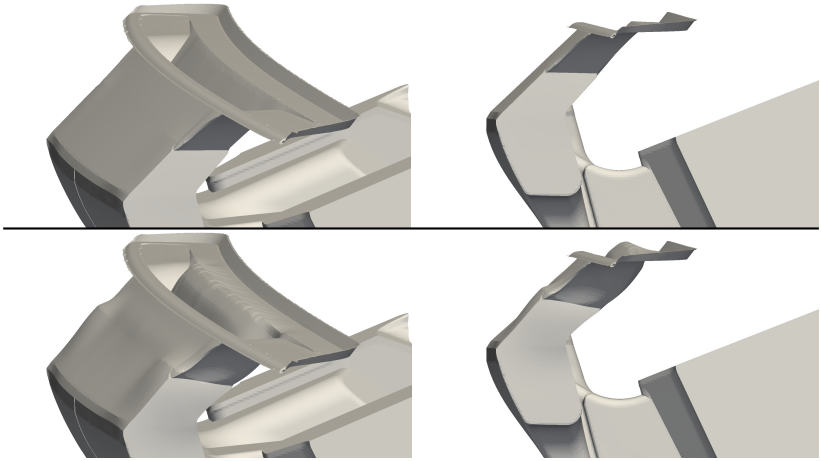
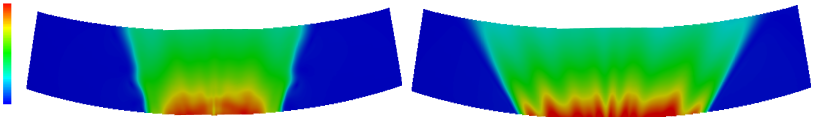


Figure 7.11: Different views of the nozzle. Initial shape (top) and the optimal shape (bottom).



(a) Initial distribution. (b) Optimal distribution.

Figure 7.12: Distribution of wall shear stress for constrained optimization.

strate the feasibility of the optimal design, two cross sections shown in Fig. 7.13 are selected for comparison purposes. As seen in Fig. 7.14, unlike the unconstrained optimized curve, which highly penetrates the bounding surface, the constrained one is point-wise feasible. Although the bounding surface has regions of high curvature, the adaptive approach introduced for the active constraint detection in Section 2.1 introduces point-wise feasible design throughout the optimization.

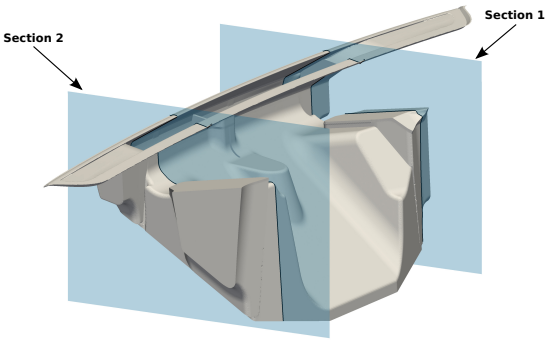
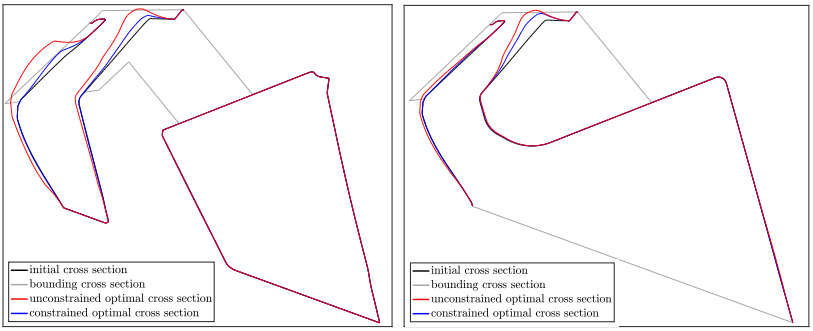


Figure 7.13: Selected cross sections of the defroster nozzle.



(a) Section 1. (b) Section 2.

Figure 7.14: Comparison of geometrically constrained and unconstrained optimization.

(a) BMW M8 GTE. (b) Wheel.

Figure 7.15

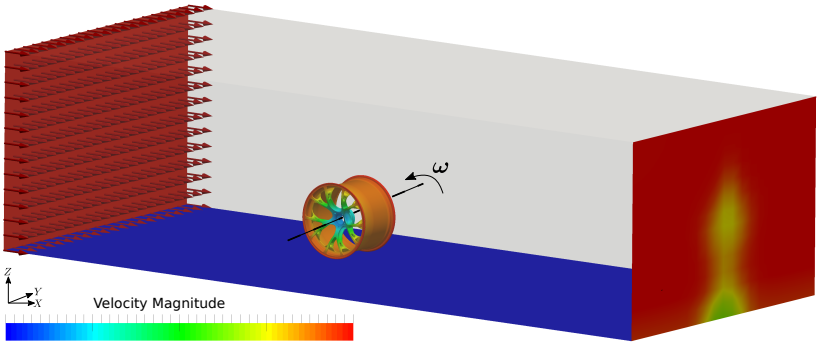
7.2 Multi-Disciplinary Shape Optimization: BMW Wheel

Having used Vertex Morphing technique to a variety of mono-physics shape optimization problems, as the next step we applied this technique to a multi-physics problem from BMW Motorsport: aerodynamic and structural shape design of the BMW M8 GTE wheel (Fig. 7.15).

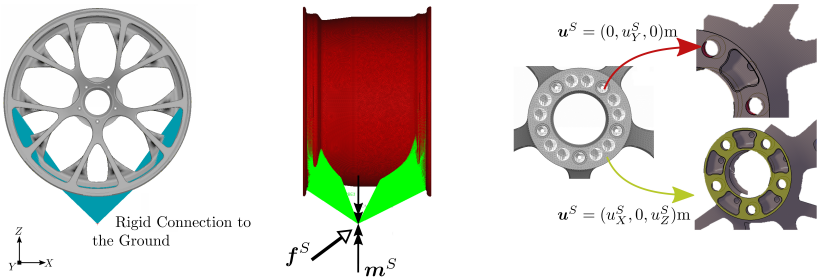
The problem under consideration is a constrained multi-objective shape optimization of the wheel which is characterized by aerodynamic and structural analyses. As can be seen in Figure 7.16(a), the spinning structure for which the aerodynamic performance has to be evaluated is put in a virtual wind tunnel, which has slip walls at the top and sides as well as wind input and output. The fluid is modeled with the incompressible Navier-Stokes equations discretized using finite volumes method. The numerical wind tunnel is composed of about 8 million cells.

The structural model of the wheel, including loading and support conditions, is shown in Figure 7.16(b). The model consists of about 1 million tetrahedral finite elements filling the entire inner volume of the wheel. Also rigid body elements are used to transfer/distribute the road load to the corresponding nodes on the rim (outer edge of a wheel, holding the tire).

The fluid and structural models, respectively, are built in OpenFOAM and Altair OptiStruct. It should be mentioned that no coupling (interaction) is considered between the models. However, the outer surface of the spokes, which is an interface between the fluid and the structure models, is chosen as the design surface. This means that the geometry of spokes is driven



(a) Fluid model.



(b) Structural model.

Figure 7.16: Computational models of the rim.

by both fluid and structural shape gradients. Figure 7.17 shows the CFD and CSD surface meshes on a spoke of the wheel. Obviously, the meshes do not match due to the different mesh requirements for the flow and structure. The structure interface mesh consists of 28000 nodes, while the fluid interface mesh consists of 145000 nodes. To deal with the non-matching meshes at the fluid-structure interface, we take full advantage of the important property of the Vertex Morphing technique of being able to discretize the geometry and control spaces with different mesh resolutions. For this purpose, the interface mesh with less resolution, here the structure mesh, is chosen as the control mesh and then the objective functions' gradients with w.r.t. the design variables ($\nabla_{\boldsymbol{s}} J^F, \nabla_{\boldsymbol{s}} J^S$) and the geometries' updates ($\Delta \boldsymbol{X}_\Gamma^F, \Delta \boldsymbol{X}_\Gamma^S$), are calculated respectively as

$$\frac{dJ^F}{d\boldsymbol{s}_\Gamma^S} = \frac{dJ^F}{d\boldsymbol{X}_\Gamma^F} \cdot \frac{d\boldsymbol{X}_\Gamma^F}{d\boldsymbol{s}_\Gamma^S} = \frac{dJ^F}{d\boldsymbol{X}_\Gamma^F} \cdot \left[\boldsymbol{A}^{FS}\right]_{3m_\Gamma^F \times 3m_\Gamma^S} \tag{7.3a}$$

$$\frac{dJ^S}{d\boldsymbol{s}_\Gamma^S} = \frac{dJ^S}{d\boldsymbol{X}_\Gamma^S} \cdot \frac{d\boldsymbol{X}_\Gamma^S}{d\boldsymbol{s}_\Gamma^S} = \frac{dJ^S}{d\boldsymbol{X}_\Gamma^S} \cdot \left[\boldsymbol{A}^{SS}\right]_{3m_\Gamma^S \times 3m_\Gamma^S} \tag{7.3b}$$

and

$$\Delta \boldsymbol{X}_\Gamma^F = \boldsymbol{A}^{FS} \,.\, \Delta \boldsymbol{s}_\Gamma^S \tag{7.4a}$$

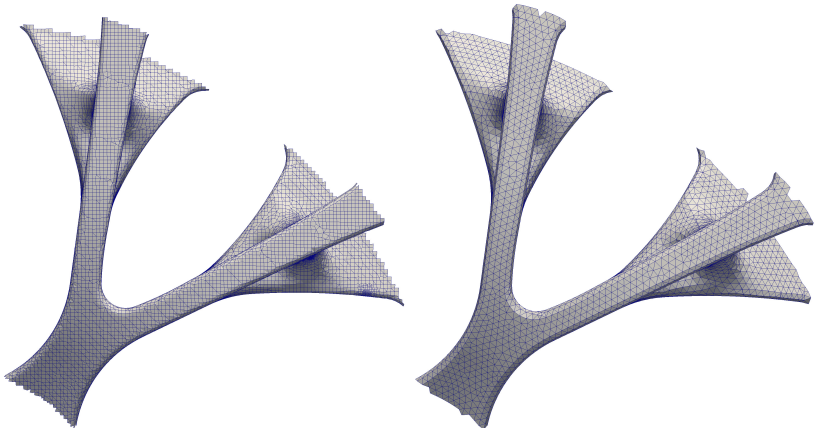
$$\Delta \boldsymbol{X}_\Gamma^S = \boldsymbol{A}^{SS} \,.\, \Delta \boldsymbol{s}_\Gamma^S \tag{7.4b}$$

where J^F and J^S represent objective functions whose spatial gradients are computed/available on the fluid surface mesh and the structure surface mesh, respectively. The matrices $\boldsymbol{A}^{FS}$ and $\boldsymbol{A}^{SS}$ are the operators which define the association between the not-matching interface meshes and the discretized design space. Based on the derivation of Vertex Morphing in Section 2.2, the entry in row i and column j of $\boldsymbol{A}^{FS}$ is computed as

$$\boldsymbol{A}_{ij}^{FS} = \frac{d\boldsymbol{X}_{\Gamma,i}^F}{d\boldsymbol{s}_{\Gamma,j}^S} = \int_{\Sigma^S} F(\boldsymbol{X}, \boldsymbol{X}_{\Gamma,i}^F)\, d\Sigma \tag{7.5}$$

where Σ^S is the portion of Γ^S which lies within a sphere of radius r and center $\boldsymbol{X}_{\Gamma,i}^F$.

The performance of the described multiphysics problem is quantified by a multi-objective target function. Minimization is performed for an

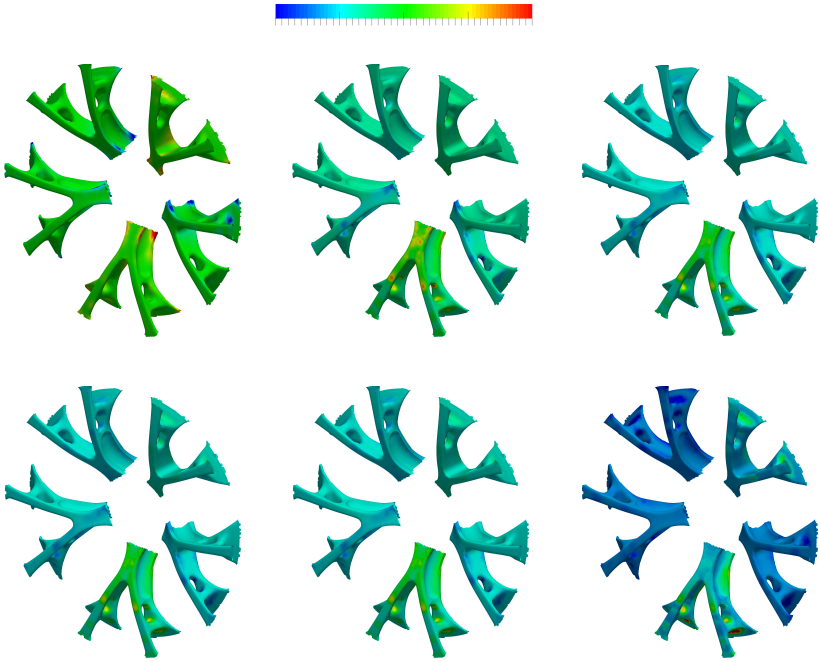


(a) CFD surface mesh. (b) CSD surface mesh.

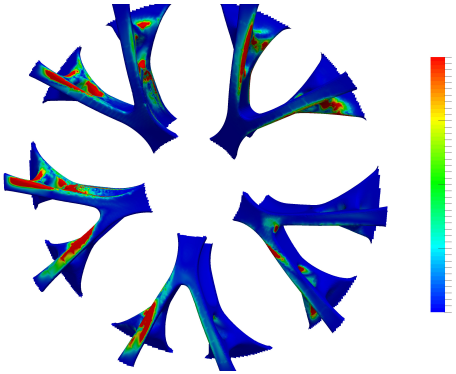
Figure 7.17: Surface discretizations of a spoke of the wheel.

equally weighted sum of the total drag force (J^F) acting on the wheel and the total strain energy (J^S) of the wheel for each load cases. Figure 7.18 plots the contours of the surface shape sensitivities. On the other hand, the design has to meet the following geometric constraints: (a) an equality constraint on the inner volume of the wheel and (b) a rotational symmetry condition on the design surface. While the former is a single scalar-valued constraint, the later results in numerous point-wise geometric constraints that can be handled in different ways. One has to notice that although the wheel geometry has the rotational symmetry property, the loads and the boundary conditions are not rotationally symmetric with respect to the rotation axis (Y-axis). Therefore it is not valid to model and analyze only one-fifth of the design surface (i.e., one spoke).

The rotational symmetry property of the wheel is explicitly focused in Figure 7.19. As can be seen, the wheel surface Γ is divided into five identical surfaces Γ_i, $i = 1, ..., 5$, each of which can be generated by rotating Γ_1 by $\theta_i = \frac{(i-1)\pi}{5}$ around the Y-axis. In general there are two main ways of enforcing the rotational symmetry constraint to the shape optimization problem, the choice depending largely upon the definition of shape design variables. The first is to choose one fifth of the wheel, e.g. Γ_1, as the design surface



(a) Total strain energy sensitivity for six load cases on Γ^S.



(b) Total drag sensitivity on Γ^F.

Figure 7.18: Shape sensitivity contours of the objectives on the spokes of the wheel.

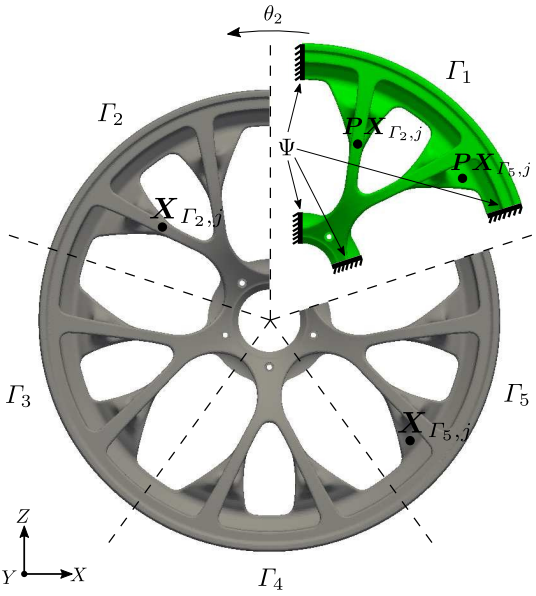


Figure 7.19: Partitioning of the rotationally symmetric wheel into 5 parts. The green part may serve as the design surface from which the whole wheel geometry can be generated.

and then control the wheel geometry during the shape optimization, and the second is to take the full wheel surface as the design space and then make the shape gradients rotationally symmetric. The former approach additionally requires enforcement of C^0 and C^1 continuity conditions along the boundary Ψ, while the latter could be computationally more demanding. Following the first approach and Vertex Morphing technique, the geometry at node j of the optimization surface Γ_i can be generated, using:

$$\boldsymbol{X}_{\Gamma_i,j} = \boldsymbol{T}_i \cdot \int_{\Sigma_1} F(\boldsymbol{X}, \boldsymbol{PX}_{\Gamma_i,j})\, \boldsymbol{s}_{\Gamma_1}\, d\Sigma, \boldsymbol{T}_i = \begin{bmatrix} \cos\theta_i & 0 & -\sin\theta_i \\ 0 & 1 & 0 \\ \sin\theta_i & 0 & \cos\theta_i \end{bmatrix} \tag{7.6a}$$

$$\boldsymbol{PX}_{\Gamma_i,j} = \begin{bmatrix} \cos\theta_i & 0 & \sin\theta_i \\ 0 & 1 & 0 \\ -\sin\theta_i & 0 & \cos\theta_i \end{bmatrix} \cdot \boldsymbol{X}_{\Gamma_i,j} \tag{7.6b}$$

where $\boldsymbol{PX}_{\Gamma_i,j}$ is the projection of $\boldsymbol{X}_{\Gamma_i,j}$ on the control surface Γ_1, Σ_1 is the portion of Γ_1 which lies within a sphere of radius r and center $\boldsymbol{PX}_{\Gamma_i,j}$. In case of non-matching interface meshes, the geometry at node j of the optimization surfaces Γ_i^S and Γ_i^F can be generated, respectively, as follows:

$$\boldsymbol{X}^S_{\Gamma_i,j} = \boldsymbol{T}_i \cdot \int_{\Sigma_1^S} F(\boldsymbol{X}, \boldsymbol{PX}^S_{\Gamma_i,j})\, \boldsymbol{s}^S_{\Gamma_1}\, d\Sigma \tag{7.7a}$$

$$\boldsymbol{X}^F_{\Gamma_i,j} = \boldsymbol{T}_i \cdot \int_{\Sigma_1^S} F(\boldsymbol{X}, \boldsymbol{PX}^F_{\Gamma_i,j})\, \boldsymbol{s}^S_{\Gamma_1}\, d\Sigma \tag{7.7b}$$

As mentioned before, another alternative to keep and enforce the rotational symmetry feature of the wheel is to symmetrize the shape gradients rotationally. This can be done by considering the following linear transformation:

$$\boldsymbol{X}_\Gamma = \begin{bmatrix} \boldsymbol{X}_{\Gamma_1} \\ \boldsymbol{X}_{\Gamma_2} \\ \boldsymbol{X}_{\Gamma_3} \\ \boldsymbol{X}_{\Gamma_4} \\ \boldsymbol{X}_{\Gamma_5} \end{bmatrix} = \begin{bmatrix} [\boldsymbol{AP}_{1,1}] \\ [\boldsymbol{AP}_{2,1}] \\ [\boldsymbol{AP}_{3,1}] \\ [\boldsymbol{AP}_{4,1}] \\ [\boldsymbol{AP}_{5,1}] \end{bmatrix} \cdot \boldsymbol{X}_{\Gamma_1}, \boldsymbol{AP}_{i,1} = \begin{bmatrix} \boldsymbol{T}_i & 0 & \dots & 0 \\ 0 & \boldsymbol{T}_i & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ddots & \boldsymbol{T}_i \end{bmatrix}_{3m_{\Gamma_i} \times 3m_{\Gamma_1}} \tag{7.8}$$

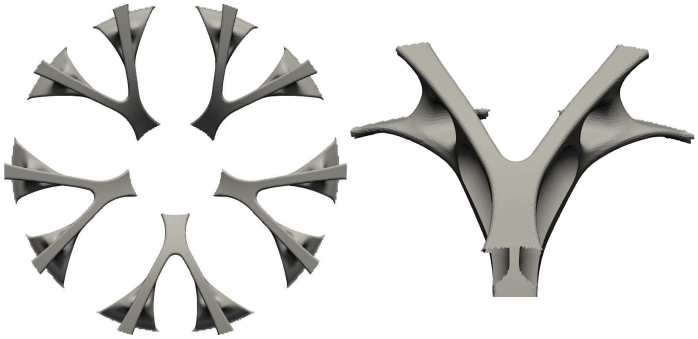
where $\boldsymbol{AP}_{i,1}, i = 1, ..., 5$ is the transformation matrix from surface Γ_i to Γ_1. Then the shape variations in the surface Γ are associated to the shape variations of the surface Γ_1 as follows:

$$\frac{d\boldsymbol{X}_\Gamma}{d\boldsymbol{X}_{\Gamma_1}} = \boldsymbol{AP} \tag{7.9}$$

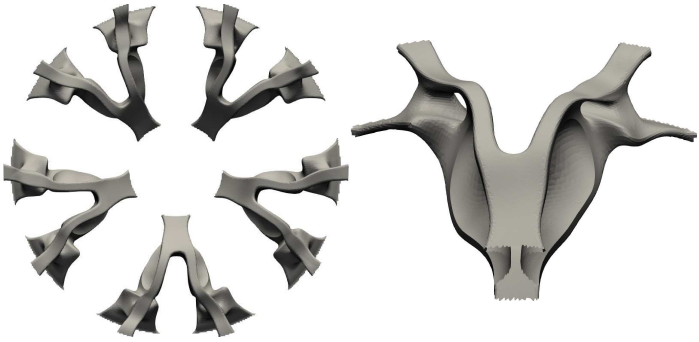
Finally, the rotationally symmetric form of objectives' gradients in Eq. 7.3 can be achieved by using the following equations:

$$\left.\frac{dJ^F}{d\boldsymbol{s}_\Gamma^S}\right|_{\text{rs}} = \frac{dJ^F}{d\boldsymbol{X}_\Gamma^F} \cdot \boldsymbol{AP}^F \cdot (\boldsymbol{AP}^F)^T \cdot \boldsymbol{A}^{FS}, \boldsymbol{AP}^F = \frac{d\boldsymbol{X}_\Gamma^F}{d\boldsymbol{X}_{\Gamma_1}^F} \tag{7.10a}$$

$$\left.\frac{dJ^S}{d\boldsymbol{s}_\Gamma^S}\right|_{\text{rs}} = \frac{dJ^S}{d\boldsymbol{X}_\Gamma^S} \cdot \boldsymbol{AP}^S \cdot (\boldsymbol{AP}^S)^T \cdot \boldsymbol{A}^{SS}, \boldsymbol{AP}^S = \frac{d\boldsymbol{X}_\Gamma^S}{d\boldsymbol{X}_{\Gamma_1}^S} \tag{7.10b}$$

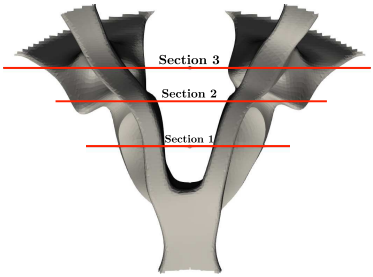


(a) Initial design.



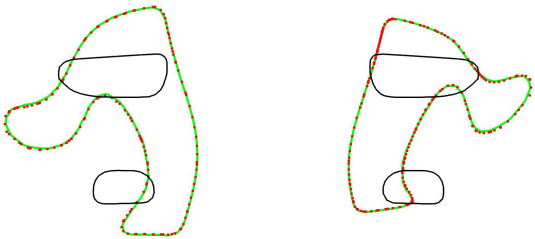
(b) Optimized design.

Figure 7.20: Shape optimization of the wheel's spokes.

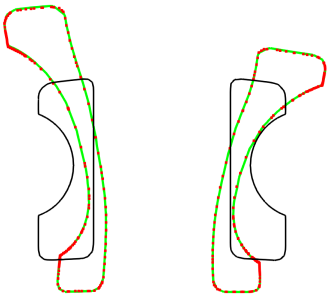


(a) Selected planes.

(b) Cross section 3.



(c) Cross section 2.



(d) Cross section 1.

Figure 7.21: Comparison of the cross section profiles at radially spaced planes. The red dotted and green solid lines, respectively, correspond to the optimized CFD and CSD surface meshes, while the black lines represent the initial cross sections.

The outcome of the described optimization problem with a medium-size filter radius is summarized in Table 7.2, and the optimal design is shown in Figure 7.20. Furthermore, more detailed comparison between the initial and optimized geometries is given in Figure 7.21. The cross section profiles of the fluid mesh (red dotted lines) is overlaid on those of the structure mesh (green solid lines) for the sake of illustrating a very good conformity of optimal fluid and structure meshes at the interface Γ. This observation confirms the applicability of Vertex Morphing technique to multi-physics shape optimization problems involving complex interface geometries with non-matching meshes.

Table 7.2: Summary of the multi-objective optimization of the BMW M8 GTE wheel.

Function	change %
Total drag	-35
Compliance 1	-17.76
Compliance 2	-16.38
Compliance 3	-9.54
Compliance 4	-19.47
Compliance 5	-19.47
Compliance 6	-19.40
Mass	+8.20

7.3 FSI Shape Optimization: Flexible ONERA M6 Wing

In this section, a multi-objective and multi-disciplinary shape optimization of the flexible ONERA M6 wing from section 6.3 is presented. The computational models and the setup of the coupled problem are the same as the ones introduced before. Being concerned with both the aerodynamic and structural performances of the wing, the shape is optimized for an equally weighted sum of the lift-to-drag ratio and the structural strain energy under equality constraint on the inner volume of the wing. Note that since both the objectives are force-based objective functional (see section 4.2), the equally weighted, linear combination of the objective functions can be effectively done by modifying the nodal force projection vector as follows:

$$\boldsymbol{d}_i = \frac{1}{2}\left(-\frac{\boldsymbol{d}^E}{\|\boldsymbol{d}^E\|} + \frac{\boldsymbol{u}^S_{\Gamma,i}}{\|\boldsymbol{u}^S_{\Gamma,i}\|}\right) \tag{7.11}$$

where $\boldsymbol{d}^E$ is a direction vector which maximizes the aerodynamic efficiency (lift-to-drag ratio) and may be defined as

$$\boldsymbol{d}^E = \frac{\boldsymbol{d}^L}{D} - \frac{L}{D^2}\boldsymbol{d}^D \tag{7.12}$$

where $\boldsymbol{d}^L$, L are respectively the direction vector and the value of the lift force, while $\boldsymbol{d}^D$, D are the same terms of the drag force. Figure 7.23 illustrates the sensitivity contribution of each discipline to the coupled adjoint-based shape sensitivities of the intended target function. Although it is observed that the mesh motion problem constitutes a very minor contribution to the overall coupled sensitivities, the role of the mesh motion problem should not be disregarded in the coupled adjoint analysis (see section 6.3).

As in the previous application, the fluid and structural domains are spatially discretized to different levels of refinement and the coarser wing surface mesh (structure surface mesh) is used to parametrized the surface using Vertex Morphing technique. Therefore, the filtering operations presented in Eqs. 7.3 and 7.4 are used in the optimization process.

The optimization has been run for several steps and it has resulted in a 32.4% increase in the lift-to-drag ratio and a 52% decrease in the total structural strain energy. The history of the optimization is presented in

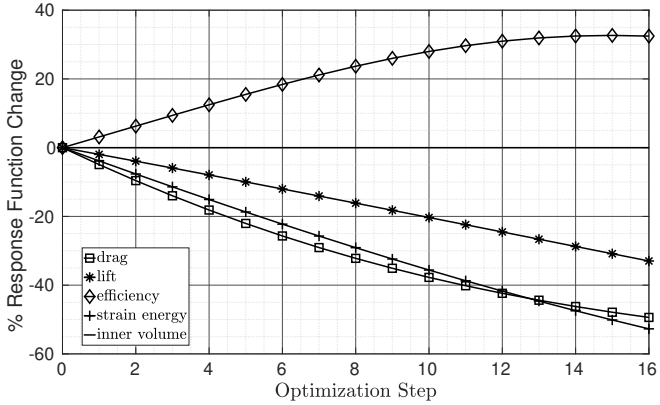
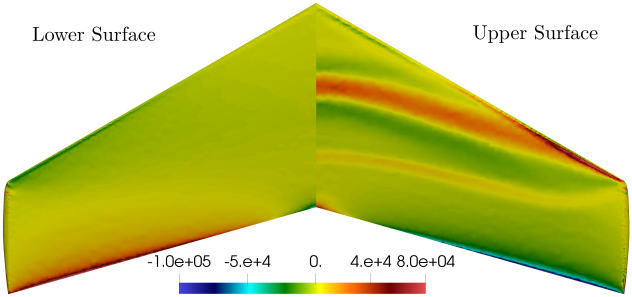
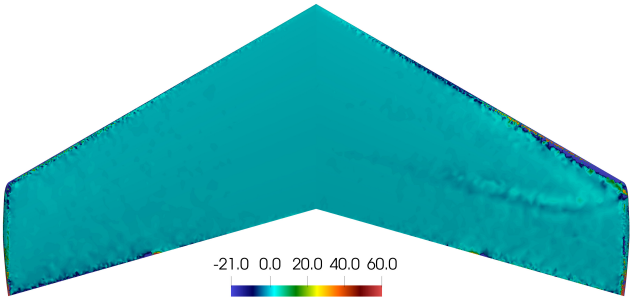


Figure 7.22: Optimization history for flexible ONERA M6 wing.

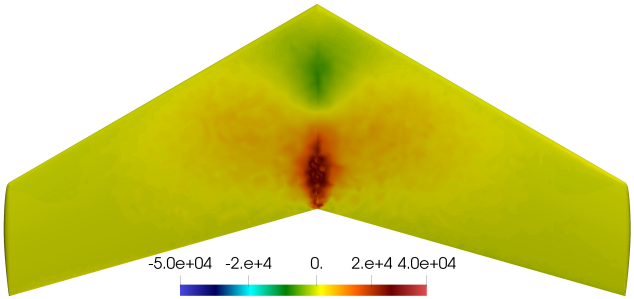
Figure 7.22. Furthermore, figures 7.24 compare the undeformed optimized wing sections with the baseline sections at 4 different spanwise stations. As seen in Figure 7.25, the strong shock wave that existed along the span has been reduced significantly, and the structure is stiffer, especially in the sense of the wing vertical Z-displacement.



(a) Normal shape gradients computed by the adjoint fluid solver.

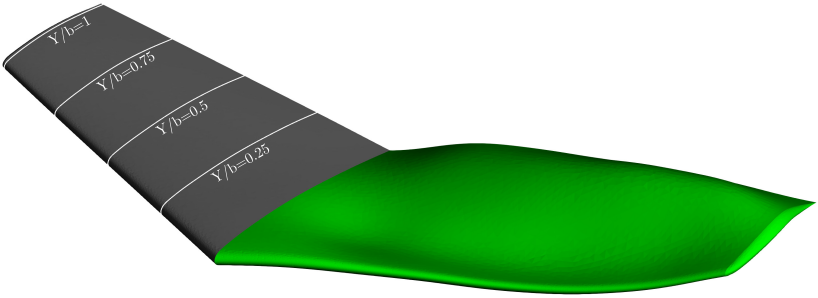


(b) Normal shape gradients computed by the adjoint mesh motion solver.



(c) Normal shape gradients computed by the adjoint structure solver.

Figure 7.23: Outcome of the fully coupled adjoint-based shape sensitivity analysis of the equally weighted sum of aerodynamic efficiency and structural strain energy using three-field-based partitioning. The shares of adjoint solvers in the final shape sensitivity are separately plotted.



(a) Left: baseline design, Right: final design scaled by 100 for better visualization.

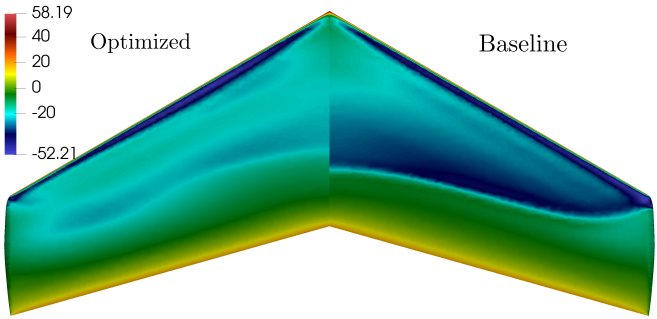
(b) Y/b=0.25.

(c) Y/b=0.5.

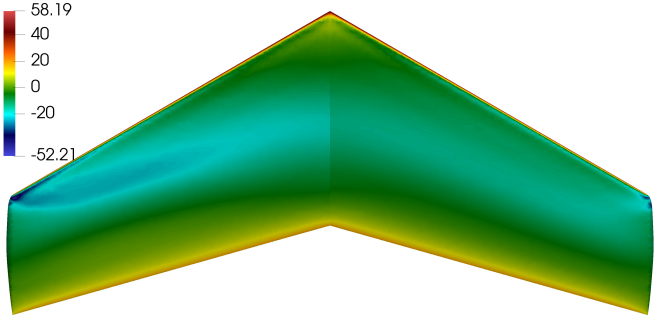
(d) Y/b=0.9.

(e) Y/b=1.

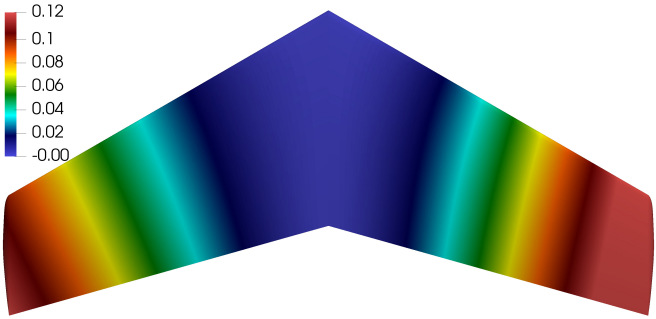
Figure 7.24: Original vs. Optimized ONERA M6. The initial and optimized airfoils are indicated by black and green lines, respectively.



(a) Surface traction field (kPa) of the upper surface.



(b) Surface traction field (kPa) of the lower surface.



(c) Z-displacement (m).

Figure 7.25: Traction and displacement contours on the flexible ONERA M6 wing. Left: final design, Right: baseline design.

Chapter 8

Conclusions

The idea of Vertex Morphing was generalized to include packaging constraints into shape optimization using Rosen's gradient projection method. Numerical properties of the mapping matrix of Vertex Morphing was studied in detail and recommendations for the gradient projection step were provided. It was observed that the presented shape parametrization technique makes it possible to generate high quality shapes in terms of surface smoothness and manufacturability. In addition, the gradient projection technique, in combination with adaptive detection of active and point-wise constraints, fulfills the feasibility of the shape throughout the design evolution. This is considered to be a great advantage for large-scale constrained optimization problems in which convergence is a challenge.

Adjoint-based shape sensitivity analysis for FSI problems was revisited from the mathematical and the numerical points of view, with a special emphasis on a partitioned solution of direct and adjoint FSI problems. Similar to the FSI problem, the adjoint FSI problem was partitioned using unique sets of Dirichlet and Neumann-type coupling conditions in such a way that it can be implemented using black-box adjoint solvers with minimal information about the source code. The structure domain becomes the Neumann partition, it takes the fluid force and the fluid adjoint force

along the interface as the Neumann boundary condition for the direct and adjoint structural problems, respectively. The fluid domain becomes the Dirichlet partition and it takes structural displacement and its adjoint as the Dirichlet boundary condition for the direct and adjoint fluid problems, respectively.

The impact of the adjoint of mesh motion problem, which is only a non-physical problem and does not exist in the continuum description of the FSI problem, on the quality of shape sensitivities was investigated in detail. Through numerical experiments, it turned out that excluding the mesh motion problem in the shape sensitivity analysis results in modifications of the sensitivity map in terms of values rather than the pattern. Thus, it will most probably not be critical for coupled optimization processes.

This thesis also investigated data mapping between non-matching meshes in direct and adjoint FSI problems. Tests with a representative aeroelastic wing showed that consistent mapping of adjoint structural displacements in combination with conservative mapping of adjoint forces yield sufficiently accurate coupled adjoint-based sensitivities.

Finally, promising results were achieved by applying the presented framework to industrial cases.

Bibliography

[1] Anders Ahlström. “Influence of wind turbine flexibility on loads and power production.” In: *Wind Energy* 9.3 (2006), pp. 237–249. ISSN: 10954244. DOI: 10.1002/we.167.

[2] T. Albring, M. Sagebaum, and N. R. Gauger. “A consistent and robust discrete adjoint solver for the SU2 framework – validation and application.” In: *Notes on Numerical Fluid Mechanics and Multidisciplinary Design* 132 (Apr. 15, 2016), pp. 77–86.

[3] G. Allaire, F. Jouve, and G. Michailidis. “Casting constraints in structural optimization via a level-set method.” In: *10th World Congress on Structural and Multidisciplinary Optimization* 1 (2013), pp. 1–10.

[4] Grégoire Allaire, François Jouve, and Georgios Michailidis. “Thickness control in structural optimization via a level set method.” In: *Structural and Multidisciplinary Optimization* (2014), pp. 1–41. ISSN: 16151488. DOI: 10.1007/s00158-016-1453-y.

[5] W. K. 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.

[6] Michael Andre, Máté Péntek, Kai Uwe Bletzinger, and Roland Wüchner. “Aeroelastic simulation of the wind-excited torsional vibration of a parabolic trough solar collector.” In: *Journal*

of Wind Engineering and Industrial Aerodynamics 165.January (2017), pp. 67–78. ISSN: 01676105. DOI: 10.1016/j.jweia.2017.03.005.

[7] Reza Najian Asl, Ihar Antonau, Aditya Ghantasala, Wulf Dettmer, Roland Wüchner, and Kai-uwe Bletzinger. "On the proper calculation of adjoint-based shape sensitivity derivatives for fluid-structure interaction with large displacements." In: *Structural and Multidisciplinary Optimization, submitted for publication* (2018).

[8] Reza Najian Asl, Shahrokh Shayegan, Armin Geiser, Majid Hojjat, and Kai-uwe Bletzinger. "A consistent formulation for imposing packaging constraints in shape optimization using Vertex Morphing parametrization." In: (2017).

[9] Santiago Badia, Fabio Nobile, and Christian Vergara. "Fluid-structure partitioned procedures based on Robin transmission conditions." In: *Journal of Computational Physics* 227.14 (2008), pp. 7027–7051. ISSN: 00219991. DOI: 10.1016/j.jcp.2008.04.006.

[10] Manuel Barcelos, Henri Bavestrello, and Kurt Maute. "A Schur-Newton-Krylov solver for steady-state aeroelastic analysis and design sensitivity analysis." In: *Computer Methods in Applied Mechanics and Engineering* 195 (2006), pp. 2050–2069. ISSN: 00457825. DOI: 10.1016/j.cma.2004.09.013.

[11] D. Baumgärtner, A. Viti, A. Dumont, G. Carrier, and K.-U. Bletzinger. "Comparison and combination of experience-based parametrization with Vertex Morphing in aerodynamic shape optimization of a forward-swept wing aircraft." In: *17th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. Ed. by American Institute of Aeronautics and Astronautics. Washington D.C., 2016. DOI: 10.2514/6.2016-3368.

[12] T. Belytschko, W.K. Liu, B. Moran, and K. Elkhodary. *Nonlinear Finite Elements for Continua and Structures*. Wiley, 2013. ISBN: 9781118700082.

[13] Bernardi, C., Maday, Y., and Patera, A.T. “A new non-conforming approach to domain decomposition: the mortar finite element method.” In: *In: Brezis H, Lions JL (eds) Nonlinear partial differential equa- tions and their applications, vol XI. College de France Seminar. Longman, Harlow* (1994), pp. 13–51.

[14] Kai-Uwe Bletzinger. “A consistent frame for sensitivity filtering and the vertex assigned morphing of optimal shape.” In: *Structural and Multidisciplinary Optimization* 49.6 (2014), pp. 873–895. ISSN: 1615-1488. DOI: 10.1007/s00158-013-1031-5.

[15] Kai-Uwe Bletzinger. “Shape Optimization.” In: *Encyclopedia of Computational Mechanics, 6 Volume Set, 2nd Edition*. Ed. by E Stein, R de Borst, and T J R Hughes. Vol. 2,3. Wiley, 2017. ISBN: 978-1-119-00379-3.

[16] A. de Boer, A. H. van Zuijlen, and H. Bijl. “Comparison of conservative and consistent approaches for the coupling of non-matching meshes.” In: *Computer Methods in Applied Mechanics and Engineering* 197.49-50 (2008), pp. 4284–4297. ISSN: 00457825. DOI: 10.1016/j.cma.2008.05.001.

[17] J Austin Cottrell, Thomas JR Hughes, and Yuri Bazilevs. *Isogeometric analysis: toward integration of CAD and FEA*. John Wiley & Sons, 2009.

[18] Pooyan Dadvand, Riccardo Rossi, and Eugenio Oñate. *An object-oriented environment for developing finite element codes for multi-disciplinary applications*. Vol. 17. 3. 2010, pp. 253–297. ISBN: 1183101090. DOI: 10.1007/s11831-010-9045-2.

[19] Joris Degroote, Robby Haelterman, Sebastiaan Annerel, Peter Bruggeman, and Jan Vierendeels. “Performance of partitioned procedures in fluid–structure interaction.” In: *Computers & structures* 88.7-8 (2010), pp. 446–457.

[20] Simone Deparis, Davide Forti, Paola Gervasio, and Alfio Quarteroni. “INTERNODES: an accurate interpolation-based method for coupling the Galerkin solutions of PDEs on subdomains featuring non-conforming interfaces.” In: *Computers and Fluids* 141 (2016), pp. 22–41. ISSN: 00457930. DOI: 10.1016/j.compfluid.2016.03.033.

[21] W. Dettmer and D. Perić. “A computational framework for fluid-structure interaction: Finite element formulation and applications.” In: *Computer Methods in Applied Mechanics and Engineering* 195.41-43 (2006), pp. 5754–5779. ISSN: 00457825. DOI: 10.1016/j.cma.2005.10.019.

[22] Thomas D Economon. “Optimal Shape Design Using an Unsteady Continuous Adjoint Approach.” PhD thesis. Stanford University, 2014.

[23] Thomas D Economon, Francisco Palacios, The Boeing Company, Long Beach, Sean R Copeland, Trent W Lukaczyk, and Juan J Alonso. “SU2: An Open-Source Suite for Multiphysics Simulation and Design.” In: 54.3 (2016). DOI: 10.2514/1.J053813.

[24] EMPIRE. *Enhanced Multi Physics Interface Research Engine*. http://empire-multiphysics.com. 2018.

[25] Charbel Farhat, Michael Lesoinne, and Patrick 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.

[26] Antonio Fazzolari, Nicolas R. Gauger, and Joël Brezillon. “Efficient aerodynamic shape optimization in MDO context.” In: *Journal of Computational and Applied Mathematics* 203.2 (June 2007), pp. 548–560. ISSN: 03770427. DOI: 10.1016/j.cam.2006.04.013.

[27] Carlos A. Felippa, K.C. Park, and Charbel Farhat. “Partitioned analysis of coupled mechanical systems.” In: *Computer Methods in Applied Mechanics and Engineering* 190.24 (2001). Advances in Computational Methods for Fluid-Structure Interaction, pp. 3247–3270. ISSN: 0045-7825. DOI: https://doi.org/10.1016/S0045-7825(00)00391-1.

[28] F. Fleissner, T. Haag, M. Hanss, and P. Eberhard. *Trends in Computational Contact Mechanics*. Vol. 58. 2011, pp. 121–134. ISBN: 978-3-642-22166-8. DOI: 10.1007/978-3-642-22167-5.

[29] K. W. K. Gaetan, G. J. Kennedy, and J. R. R. A. Martins. “Scalable parallel approach for high-fidelity steady-state aeroelastic analysis and adjoint derivative computations.” In: *AIAA Journal* 52 (2014), 935–951.

[30] Michael W Gee, Ulrich Küttler, and Wolfgang A Wall. “Truly monolithic algebraic multigrid for fluid–structure interaction.” In: *International Journal for Numerical Methods in Engineering* 85.8 (2011), pp. 987–1016.

[31] L A Germanou, E.M. Papoutsis-Kiachagias, A Delacroix, and K C Giannakoglou. “Defroster Nozzle Shape Optimization Using the Continuous Adjoint Method.” In: *VII European Congress on Computational Methods in Applied Sciences and Engineering* June (2016), p. 10.

[32] Anthony Giunta and Jaroslaw Sobieszczanski-Sobieski. “Progress toward using sensitivity derivatives in a high-fidelity aeroelastic analysis of a supersonic transport.” In: *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization* (1998). DOI: 10.2514/6.1998-4763.

[33] Xu Guo, Weisheng Zhang, and Wenliang Zhong. “Explicit feature control in structural topology optimization via level set method.” In: *Computer Methods in Applied Mechanics and Engineering* 272 (2014), pp. 354–378. ISSN: 0045-7825. DOI: 10.1016/j.cma.2014.01.010.

[34] Raphael T. Haftka and Zafer Gürdal, eds. *Elements of structural optimization*. Solid Mechanics And Its Applications. Springer Netherlands, 1992. ISBN: 978-94-011-2550-5.

[35] Matthias Heil, Andrew L Hazel, and Jonathan Boyle. “Solvers for large-displacement fluid–structure interaction problems: segregated versus monolithic approaches.” In: *Computational Mechanics* 43.1 (2008), pp. 91–101.

[36] J. P. Heners, L. Radtke, M. Hinze, and A. Düster. “Adjoint shape optimization for fluid–structure interaction of ducted flows.” In: *Computational Mechanics* (2017), pp. 1–18. ISSN: 01787675. DOI: 10.1007/s00466-017-1465-5.

[37] Van-nam Hoang and Gang-won Jang. “Topology optimization using moving morphable bars for versatile thickness control.” In: *Comput. Methods Appl. Mech. Engrg.* 317 (2017), pp. 153–173. ISSN: 0045-7825. DOI: 10.1016/j.cma.2016.12.004.

[38] Majid Hojjat. “Node-based parametrization for shape optimal design.” PhD thesis. Technische Universität München, 2014.

[39] Majid Hojjat, Electra Stavropoulou, and Kai Uwe Bletzinger. “The Vertex Morphing method for node-based shape optimization.” In: *Computer Methods in Applied Mechanics and Engineering* 268 (Jan. 2014), pp. 494–513. ISSN: 00457825. DOI: 10.1016/j.cma.2013.10.015.

[40] Gene Hou, Jin Wang, and Anita Layton. “Numerical Methods for Fluid-Structure Interaction — A Review.” In: *Communications in Computational Physics* 12.2 (2012), pp. 337–377. DOI: 10.4208/cicp.291210.290411s.

[41] Ming Chen Hsu, David Kamensky, Yuri Bazilevs, Michael S. Sacks, and Thomas J.R. Hughes. “Fluid–structure interaction analysis of bioprosthetic heart valves: significance of arterial wall deformation.” In: *Computational Mechanics* 54.4 (2014), pp. 1055–1071. ISSN: 01787675. DOI: 10.1007/s00466-014-1059-4. arXiv: NIHMS150003.

[42] Nicholas Jenkins and Kurt Maute. “An immersed boundary approach for shape and topology optimization of stationary fluid-structure interaction problems.” In: *Structural and Multidisciplinary Optimization* 54.5 (2016), pp. 1191–1208. ISSN: 16151488. DOI: 10.1007/s00158-016-1467-5.

[43] G K Karpouzas and E De Villiers. “Level-Set Based Topology Optimization Using the Continuous Adjoint Method.” In: June (2014), pp. 4–6.

[44] G.K. Karpouzas, E.M. Papoutsis-Kiachagias, T. Schumacher, E. de Villiers, K.C. Giannakoglou, and C. Othmer. “Adjoint Optimization for Vehicle External Aerodynamics.” In: *International Journal of Automotive Engineering* 7.January (2016), pp. 1–7. ISSN: 21850992. DOI: 10.20485/jsaeijae.7.1_1.

[45] 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. ISSN: 10902716. DOI: 10.1016/j.jcp.2015.08.012.

[46] Moritz Keuthen and Michael Ulbrich. "Moreau–Yosida regularization in shape optimization with geometric constraints." In: *Computational Optimization and Applications* 62.1 (2015), pp. 181–216. ISSN: 1573-2894. DOI: 10.1007/s10589-014-9661-0.

[47] Jan F Kiviaho, Kevin E Jacobson, Marilyn J Smith, and Graeme J Kennedy. "A Robust and Flexible Coupling Framework for Aeroelastic Analysis and Optimization." In: *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference* June (2017), pp. 1–23. DOI: 10.2514/6.2017-4144.

[48] Alexander Konyukhov and Karl Schweizerhof. "On the solvability of closest point projection procedures in contact analysis: Analysis and solution strategy for surfaces of arbitrary geometry." In: *Computer Methods in Applied Mechanics and Engineering* 197.33-40 (2008), pp. 3045–3056. ISSN: 00457825. DOI: 10.1016/j.cma.2008.02.009.

[49] Ján Kopačka, Dušan Gabriel, Jiří Plešek, and Miran Ulbin. "Assessment of methods for computing the closest point projection, penetration, and gap functions in contact searching problems." In: *International Journal for Numerical Methods in Engineering* 105.11 (2016). nme.4994, pp. 803–833. ISSN: 1097-0207. DOI: 10.1002/nme.4994.

[50] A. KOROBENKO, M.-C. HSU, I. AKKERMAN, J. TIPPMANN, and Y. BAZILEVS. "Structural Mechanics Modeling and Fsi Simulation of Wind Turbines." In: *Mathematical Models and Methods in Applied Sciences* 23.02 (2013), pp. 249–272. ISSN: 0218-2025. DOI: 10.1142/S0218202513400034.

[51] Daniel Kraft. "Geometric Constraints in Descent Methods for Shape Optimisation." In: *Mathematical Modelling and Numerical Analysis* (2015).

[52] Kratos Development Team. *The Kratos Multiphysics Open-source Project*. https://github.com/KratosMultiphysics. 2018.

[53] T.A. Laursen. *Computational Contact and Impact Mechanics*. Springer-Verlag Berlin Heidelberg, 2003. ISBN: 978-3-642-07685-5. DOI: 10.1007/978-3-662-04864-1.

[54] Chau Le, Tyler Bruns, and Daniel Tortorelli. "A gradient-based, parameter-free approach to shape optimization." In: *Computer Methods in Applied Mechanics and Engineering* 200.9-12 (Feb. 2011), pp. 985–996. ISSN: 00457825. DOI: 10.1016/j.cma.2010.10.004.

[55] E. Lund, H. MØller, and L.a. Jakobsen. "Shape design optimization of stationary fluid-structure interaction problems with large displacements and turbulence." In: *Structural and Multidisciplinary Optimization* 25 (2003), pp. 383–392. ISSN: 1615-147X. DOI: 10.1007/s00158-003-0288-5.

[56] Karthik Mani and Dimitri J. Mavriplis. "Adjoint-Based Sensitivity Formulation for Fully Coupled Unsteady Aeroelasticity Problems." In: *AIAA Journal* 47.8 (2009), pp. 1902–1915. ISSN: 0001-1452. DOI: 10.2514/1.40582.

[57] Meryem Marcelet, Jacques Peter, and Gérald Carrier. "Sensitivity analysis of a strongly coupled aero-structural system using the discrete direct and adjoint methods." In: *Revue européenne de mécanique numérique* 17.8 (2008), pp. 1077–1106. ISSN: 17797179. DOI: 10.3166/remn.17.1077-1106.

[58] Joaquim R. R. A. Martins and Andrew B. Lambe. "Multidisciplinary Design Optimization: A Survey of Architectures." In: *AIAA Journal* 51.9 (2013), pp. 2049–2075. ISSN: 0001-1452. DOI: 10.2514/1.J051895.

[59] K Maute, M Nikbay, and C Farhat. "Analytically based sensitivity analysis and optimization of nonlinear aeroelastic systems." In: *8th Symposium on Multidisciplinary Analysis and Optimization*. 2000, p. 4825.

[60] K. Maute, M. Nikbay, and C. Farhat. "Coupled Analytical Sensitivity Analysis and Optimization of Three-Dimensional Nonlinear Aeroelastic Systems." In: *AIAA Journal* 39.11 (Nov. 2001), pp. 2051–2061. ISSN: 0001-1452. DOI: 10.2514/2.1227.

[61] Georgios Michailidis. “Manufacturing Constraints and Multi-Phase Shape and Topology Optimization via a Level-Set Method.” PhD thesis. Ecole Polytechnique, 2014.

[62] 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* March (2008), pp. 861–877. DOI: 10.1002/fld.

[63] Carsten Othmer. “Adjoint methods for car aerodynamics.” In: *Journal of Mathematics in Industry* 4.1 (2014), pp. 1–23. ISSN: 2190-5983. DOI: 10.1186/2190-5983-4-6.

[64] Carsten Othmer, Eugene De Villiers, and Henry G Weller. “Implementation of a continuous adjoint for topology optimization of ducted flows.” In: *18th AIAA Computational Fluid Dynamics Conference*. Ed. by American Institute of Aeronautics and Astronautics. Miami, 2007.

[65] Francisco Palacios, Michael R Colonno, Aniket C Aranake, Alejandro Campos, Sean R Copeland, Thomas D Economon, Amrita K Lonkar, Trent W Lukaczyk, Thomas W R Taylor, and Juan J Alonso. “Stanford University Unstructured (SU 2): An open-source integrated computational environment for multi-physics simulation and design.” In: January (2013), pp. 1–60.

[66] Francisco Palacios et al. “Stanford University Unstructured (SU 2): Open-source Analysis and Design Technology for Turbulent Flows.” In: January (2014), pp. 1–33.

[67] Les Piegl and Wayne Tiller. *The NURBS book*. Springer Science & Business Media, 2012.

[68] Alexander Popp. “Mortar Methods for Computational Contact Mechanics and General Interface Problems.” PhD thesis. Technische Universität München, 2012.

[69] Anna Pyzara, Beata Bylina, and Jaroslaw Bylina. “The influence of a matrix condition number on iterative methods’ convergence.” In: *2011 Federated Conference on Computer Science and Information Systems (FedCSIS)* (2011), pp. 459–464.

[70] Joaquim R. R. A. Martins, Juan J. Alonso, and James J. Reuther. "High-Fidelity Aerostructural Design Optimization of a Supersonic Business Jet." In: *Journal of Aircraft* 41.3 (2004), pp. 523–530. ISSN: 0021-8669. DOI: 10.2514/1.11478.

[71] J. B. Rosen. "The Gradient Projection Method for Nonlinear Programming. Part I. Linear Constraints." In: *Journal of the Society for Industrial and Applied Mathematics* 8.1 (1960), pp. 181–217. ISSN: 03684245.

[72] J. B. Rosen. "The Gradient Projection Method for Nonlinear Programming. Part II. Nonlinear Constraints." In: *Journal of the Society for Industrial and Applied Mathematics* 9.4 (1961), pp. 514–532. ISSN: 03684245.

[73] P. B. Ryzhakov, R. Rossi, S. R. Idelsohn, and E. Oñate. "A monolithic Lagrangian approach for fluid–structure interaction problems." In: *Computational Mechanics* 46.6 (Nov. 2010), pp. 883–899. ISSN: 1432-0924. DOI: 10.1007/s00466-010-0522-0.

[74] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

[75] Jamshid A Samareh. "A survey of shape parameterization techniques." In: (1999).

[76] Jamshid a. Samareh. "Survey of shape parameterization techniques for high-fidelity multidisciplinary shape optimization." In: *AIAA Journal* 39.5 (2001), pp. 877–884. ISSN: 0001-1452. DOI: 10.2514/3.14814.

[77] R. Sanchez, T. Albring, R. Palacios, N. R. Gauger, T. D. Economon, and J. J. Alonso. "Coupled Adjoint-Based Sensitivities in Large-Displacement Fluid-Structure Interaction using Algorithmic Differentiation." In: *International Journal for Numerical Methods in Engineering* September 2017 (2017), pp. 1081–1107. ISSN: 00295981. DOI: 10.1002/nme.5700. arXiv: 1201.4903.

[78] M. Sayed, Th Lutz, E. Krämer, Sh Shayegan, A. Ghantasala, R. Wüchner, and K. U. Bletzinger. "High fidelity CFD-CSD aeroelastic analysis of slender bladed horizontal-axis wind turbine." In: *Journal of Physics: Conference Series* 753.4 (2016). ISSN: 17426596. DOI: 10.1088/1742-6596/753/4/042009.

[79] Oliver Schmitt, Jan Friederich, Stefan Riehl, and Paul Steinmann. "On the formulation and implementation of geometric and manufacturing constraints in node–based shape optimization." In: *Structural and Multidisciplinary Optimization* 53.4 (2016), pp. 881–892. ISSN: 1615-147X. DOI: 10.1007/s00158-015-1359-0.

[80] Oliver Schmitt and Paul Steinmann. "On curvature control in node-based shape optimization." In: *Pamm* 15.1 (2015), pp. 579–580. ISSN: 16177061. DOI: 10.1002/pamm.201510279.

[81] V. Schmitt and F. Charpin. *Pressure Distributions on the ONERA-M6-Wing at Transonic Mach Numbers*. Tech. rep. Report of the Fluid Dynamics Panel Working Group 04, AGARD AR 138, 1979.

[82] S. Sicklinger, V. Belsky, B. Engelmann, H. Elmqvist, H. Olsson, R. Wüchner, and K.-U. Bletzinger. "Interface Jacobian-based Co-Simulation." In: *International Journal for Numerical Methods in Engineering* 98.6 (2014), pp. 418–444. DOI: 10.1002/nme.4637. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.4637.

[83] O Soto and R Lohner. "On the computation of flow sensitivities from boundary integrals." In: *AIAA Paper* June (2004).

[84] Electra Stavropoulou. "Sensitivity analysis and regularization for shape optimization of coupled problems." PhD thesis. Technische Universität München, 2015.

[85] SU2. *Stanford University Unstructured*. https://su2code.github.io. 2018.

[86] T. W. R. Taylor, F. Palacios, K. Duraisamy, and J. J. Alonso. *Towards a hybrid adjoint approach for complex flow simulations*. Tech. rep. Center for Turbulence Research (CTR)-Stanford University, 2012, pp. 333–346.

[87] T.E. Tezduyar, S. Mittal, S.E. Ray, and R. Shih. "Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements." In: *Computer Methods in Applied Mechanics and Engineering* 95.2 (1992), pp. 221–242. ISSN: 0045-7825. DOI: https://doi.org/10.1016/0045-7825(92)90141-6.

[88] Santosh Tiwari, Hong Dong, Georges Fadel, Peter Fenyes, and Artemis Kloess. “A physically-based shape morphing algorithm for packing and layout applications.” In: *International Journal on Interactive Design and Manufacturing* 9.4 (2014), pp. 277–289. ISSN: 19552505. DOI: 10.1007/s12008-014-0237-0.

[89] M. Vanella, T. Fitzgerald, S. Preidikman, E. Balaras, and B. Balachandran. “Influence of flexibility on the aerodynamic performance of a hovering wing.” In: *Journal of Experimental Biology* 212.1 (2009), pp. 95–105. ISSN: 0022-0949. DOI: 10.1242/jeb.016428.

[90] Tianyang Wang. “Development of Co-Simulation Environment and Mapping Algorithms.” PhD thesis. Technische Universität München, 2016.

[91] Tianyang Wang, Roland Wüchner, Stefan Sicklinger, and Kai Uwe Bletzinger. “Assessment and improvement of mapping algorithms for non-matching meshes and geometries in computational FSI.” In: *Computational Mechanics* 57.5 (2016), pp. 793–816. ISSN: 01787675. DOI: 10.1007/s00466-016-1262-6.

[92] Qi Xia, Tielin Shi, Michael Yu Wang, and Shiyuan Liu. “A level set based method for the optimization of cast part.” In: *Structural and Multidisciplinary Optimization* 41.5 (2010), pp. 735–747. ISSN: 1615-1488. DOI: 10.1007/s00158-009-0444-7.

[93] Jose R Zayas and Wesley D Johnson. “3X-100 Blade Field Test.” In: *Wind Energy Technology Department, Sandia National Laboratories, Albuquerque, NM, Report No. SAND2007-5138* (2008).

[94] Zimi J. Zhang, Shahriar Khosravi, and David W. Zingg. “High-fidelity aerostructural optimization with integrated geometry parameterization and mesh movement.” In: *Structural and Multidisciplinary Optimization* 55.4 (2017), pp. 1217–1235. ISSN: 16151488. DOI: 10.1007/s00158-016-1562-7.

[95] Z.J. Zhang and D.W. Zingg. “Efficient monolithic solution algorithm for high-fidelity aerostructural analysis and optimization.” In: *AIAA Journal* 56.3 (2018), pp. 1–15. ISSN: 00011452. DOI: 10.2514/1.J056163.