National Technical University of Athens
School of Mechanical Engineering
Fluids Section
Laboratory of Thermal Turbomachines
Parallel CFD & Optimization Unit

The Continuous Adjoint Method in Aerodynamic and Conjugate Heat Transfer Shape Optimization, for Turbulent Flows

PhD Thesis

Konstantinos T. Gkaragkounis

Supervisor: Kyriakos C. Giannakoglou,
Professor NTUA

Athens, 2020

National Technical University of Athens
School of Mechanical Engineering
Fluids Section
Laboratory of Thermal Turbomachines
Parallel CFD & Optimization Unit

The Continuous Adjoint Method in Aerodynamic and Conjugate Heat Transfer Shape Optimization, for Turbulent Flows

PhD Thesis

Konstantinos T. Gkaragkounis

Examination Committee:

1. K.C. Giannakoglou (Supervisor)*
   Professor, NTUA, School of Mechanical Engineering
2. S. Voutsinas*
   Professor, NTUA, School of Mechanical Engineering
3. K. Mathioudakis*
   Professor, NTUA, School of Mechanical Engineering
4. I. Anagnostopoulos
   Professor, NTUA, School of Mechanical Engineering
5. L. Kaiktsis
   Professor, NTUA, School of Naval Architecture and Marine Engineering
6. K. Belibassakis
   Professor, NTUA, School of Naval Architecture and Marine Engineering
7. V. Riziotis
   Assistant Professor, NTUA, School of Mechanical Engineering

* Member of the Advisory Committee.

Athens, 2020

Abstract

The goal of this PhD thesis is to further develop continuous adjoint-based shape optimization methods of high accuracy and low computational cost, able to deal with one- (fluid mechanics) or many-discipline problems (two disciplines: Conjugate Heat Transfer, CHT). In addition, adjoint-based methods for tracing the Pareto front in multi-objective optimization are proposed and assessed. Finally, gradient-based methods for the analysis and optimization under uncertainties in CHT problems, based on the intrusive variant of the Polynomial Chaos Expansion method are developed, programmed and applied. To facilitate the above methods, continuous adjoint methods, developed in previous PhD theses for incompressible fluid flows [174, 125, 70] and successfully applied in industrial aero- and hydrodynamic shape optimizations, are enriched and extended. This PhD thesis is structured along the following four axes:

The first axis is about the proper treatment of grid sensitivities i.e. grid nodes variations w.r.t. the design variables, when developing the continuous adjoint method. Internal grid displacement was initially considered in the computation of sensitivity derivatives (SDs) in which grid sensitivities appear within Field Integrals; this method is abbreviated as the *FI* adjoint. The computation of the grid sensitivities in the interior of the domain requires as many grid displacements as the number of the design variables. To avoid this costly computation, in [70], the first continuous adjoint method involving the formulation and solution of adjoint grid displacement equations was presented. The so-computed SDs consisted solely of Surface Integrals and, thus, this method is abbreviated as the *E-SI* (Enhanced-*SI*) adjoint. "Enhanced" is used to distinguish this method from frequently used adjoint formulations also leading to SDs with surface integrals, in which the corresponding terms are omitted (Severed SI adjoint). In [70], to derive the *E-SI* adjoint, grid displacement was assumed to be governed by Laplace PDEs, since this was convenient for developing the adjoint method to the grid displacement model (GDM). In this PhD thesis, this assumption is assessed, by taking into account that, in many cases, GDMs other than Laplace PDEs are used. Comparisons between SDs computed for commonly used GDMs (volumetric B-Splines, Delaunay Graph, Inverse Distance Weighting) and the Laplace PDEs conclude that the choice of the GDM has negligible effect on the SDs. This is a relieving finding since there is no need to reformulate the *E-SI* adjoint, if different GDMs are in use.

The second axis of this thesis concerns the extension of the *E-SI* and *FI* continuous adjoint methods, developed in previously completed PhD theses in the PCOpt/NTUA for pure flow problems by considering eddy viscosity variations, to CHT problems. In the latter, except for the fluid flow, the energy equation in the fluid and the heat conduction in the solid, are simulated by also accounting for in-

teractions along their interface. To derive the adjoint method in an exact way, the turbulence model PDEs (in this thesis the Spalart–Allmaras is used) are differentiated. Through numerical investigations, it is recorfirmed also in CHT problems that the "frozen turbulence" assumption, i.e. any approach that neglects the eddy viscosity variation, during the formulation of the adjoint equations, mitigates the accuracy of SDs. In addition, the proper treatment of grid sensitivities is extended to CHT problems as well. Laplace PDEs are assumed to govern the internal grid displacement of the fluid and solid grids. Adjoint grid displacement equations are derived for the solid domain and those for the fluid, acquired for pure fluid flows, are extended by terms emanating from the differentiation of the fluid energy equation. A comparison between the *E-SI* and *FI* methods reconfirm the computational gain from the use of the *E-SI* adjoint, this time in CHT problems. Also, it is shown that neglecting the internal grid displacement, by using the Severed *SI* method, damages the SDs accuracy in CHT problems. Studies similar to those performed for pure fluid flows demonstrate that, the choice of the GDM has negligible effect on the SDs, in CHT problems too.

In the third axis of this thesis, adjoint methods for pure fluid flows and CHT problems are used to assist gradient-based methods tracing the front of non-dominated solutions, a.k.a. the Pareto front, in multi-objective optimization problems. In this PhD thesis, a prediction-correction algorithm (in some variants) tracing the Pareto front is developed. The prediction-correction variants are initialized by a point on the front, obtained by carrying out a single objective optimization for one of the objectives only. In the prediction and correction steps, different systems of equations are derived by treating the Karush-Kuhn-Tucker optimality conditions in two different ways. The costly computation of the exact Hessian matrix of the objective functions, which appear in the equations solved to update the design variables, is avoided. Instead, two alternative approaches are used: (a) the computation of Hessian-vector products driving a Krylov subspace solver and (b) the approximation of the Hessian via the BFGS method. To compute Hessian-vector products at the lowest cost possible, new systems of equations are derived by applying the Direct Differentiation method on the primal and adjoint PDEs. Between the two approaches, the one approximating the Hessian matrices in the prediction step is shown to be the less expensive. It is also demonstrated that omitting the prediction step increases the cost of tracing the Pareto front. Further comparisons of the less expensive prediction-correction variant and weighted-sum approaches for computing Pareto fronts demonstrate the superiority of the former in terms of computational cost.

The last axis of this thesis regards shape optimizations with the presence of environmental and manufacturing uncertainties for CHT problems, based on the intrusive Polynomial Chaos Expansion (iPCE). Primal fields in the corresponding equations and boundary conditions for a single operation point (deterministic equations) are expanded by using weighted sums of orthogonal polynomials for

chaos order equal to 2. Adjoint equations are then derived by differentiating the iPCE primal PDEs. The new systems of equations are solved to compute statistical moments and their derivatives at low cost. To verify the accuracy of the computed moments, comparisons are made between the iPCE method and more computationally expensive ones, such as the non-intrusive PCE (niPCE) and Monte Carlo. Problems, in which uncertainties are related to the flow boundary conditions along the inlet of the fluid domain and the insulation thickness between fluids and solids, are investigated. These uncertainties are considered by performing expansions of the inlet and Fluid-Solid Interface conditions, leading to new conditions for the PCE coefficients.

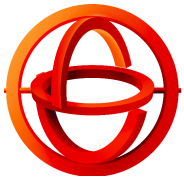
Case studies regarding both pure fluid flow and CHT problems, are presented. Developed methods are first validated/tested in 2D problems with pure fluid flows around isolated airfoils or inside ducts and CHT problems with a duct attached to a solid body. Regarding pure fluid flow problems, multi-objective optimizations for isolated airfoils, with contradictory goals being the exerted lift and drag forces, are performed. Regarding CHT problems, an internally cooled turbine cascade vane is optimized targeting min. mean solid temperature. To do so, its contour shape and the positions of the internal cooling holes are modified. The shape of the 2D cooling channel is also optimized for min. total pressure losses subjected to constraints which prevent the solid body from overheating. In addition, the cooling efficiency of a 3D internal cooling system is optimized. The cooling channel is redesigned for two different objective functions: the heat flux absorbed by the coolant and the part of the solid body volume with the highest temperature values. Moreover, shape optimizations are performed for a car-engine cylinder head. The internal cooling channel immersed in the engine block is redesigned for two different targets: min. max. solid temperature values and min. total pressure losses in the coolant. Finally, a multi-objective optimization is performed in a case in which cooling fins are attached to a heated solid body. In specific, a Pareto front is computed, with the same objective functions as in the optimization of the car engine cylinder-head.

The necessary software has been programmed in the open-source CFD toolbox OpenFOAM©, which provides a cell-centered, collocated, finite-volume infrastructure for discretizing PDEs and solving linearized systems of equations.

Keywords: Conjugate Heat Transfer, Aerodynamic Shape Optimization, Continuous Adjoint Method, Adjoint Turbulence Model, Grid Sensitivities, Grid Displacement Method, Pareto Fronts, Uncertainty Quantification, Optimization under Uncertainties, Intrusive Polynomial Chaos Expansion

Acknowledgments

The research work was supported by the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT), under the HFRI PhD Fellowship grant (GA.no. 1796).



H.F.R.I.
Hellenic Foundation for
Research & Innovation

GENERAL SECRETARIAT FOR
RESEARCH AND TECHNOLOGY

Thanks

I would like to express my sincere gratitude to those people who helped me complete this PhD thesis. First of all, I would like to thank the supervisor of my thesis, Professor Kyriakos Giannakoglou, who introduced me to the PCOpt/NTUA and gave me the chance to study all those interesting topics presented in this thesis. His guidance and suggestions helped me to solve a number of scientific problems. I would also like to thank him for the time invested into detailed proofreading of various scientific0 articles and this text.

In addition, I would like to express my gratitude towards Spyridon Voutsinas, Professor NTUA, and Konstantinos Mathioudakis, Professor NTUA, for their constructive remarks on the thesis presentation and their participation in the advisory committee of this PhD thesis.

Moreover, I would like to thank current and former members of the PCOpt/NTUA. Foremost, I would like to thank Dr. Evangelos Papoutsis-Kiachagias, for his insightful suggestions regarding various scientific topics and his limitless patience during scientific discussions on adjoint methods and coding practices. Moreover, I would like to express my gratitude towards a former member of the PCOpt/NTUA, Dr. Ioannis Kavvadias, who provided guidance at the very beginning of my thesis. I would also like to thank Konstantinos Samouchos for our fruitful discussions on various mathematical aspects regarding the formulation of adjoint methods. I am also grateful about the technical support regarding the infrastructure used during my thesis, for which Dr. Varvara Asouti and Dr. Dimitris Kapsoulis were responsible. Finally, I would like to acknowledge Dr. Xenofon Trompoukis, Dr. Konstantinos Tsiakas, James Koch, Flavio Gagliardi, Thanasis Liatsikouras, John Vrionis, Morteza Morfaredi, Themis Skamagkis, Andreas Margetis and Marina Kontou.

During this thesis, nothing would be possible without the continuous support of my parents and my sister, who did not stop encouraging me. Last but not least, I would like to thank my cousins, my friends and a few other close to me persons, who encouraged me during this thesis.

An expert is a person who has made all the mistakes that can be made in a very narrow field.

Niels Bohr

Abbreviations

CFD	Computational Fluid Dynamics
CHT	Conjugate Heat Transfer
CPs	Control Points
DD	Direct Differentiation
DG	Delaunay Graphs
EAs	Evolutionary Algorithms
EFS	Equivalent Flow Solutions
E-SI	Enhanced Surface Integral (adjoint formulation)
FDs	Finite Differences
FI	Field Integral (adjoint formulation)
FSI	Fluid-Solid Interface
GBMs	Gradient-Based Methods
GDM	Grid Displacement Model
IDW	Inverse Distance Weighting
iPCE	intrusive Polynomial Chaos Expansion
KKT	Karush-Kuhn-Tucker
l.h.s.	Left-Hand Side
MOO	Multi-Objective Optimization
niPCE	non-intrusive Polynomial Chaos Expansion
OuU	Optimization under Uncertainties
PCE	Polynomial Chaos Expansion
PCOpt/NTUA	Parallel CFD & Optimization Unit of the NTUA
PDEs	Partial Differential Equations
PDF	Probability Density Function
PP	Pareto Point
QoI	Quantities of Interest
RANS	Reynolds-Averaged Navier-Stokes
r.h.s.	Right-Hand Side
SDs	Sensitivity Derivatives
SI	Surface Integral (adjoint formulation)
SOO	Single-Objective Optimization
SQP	Sequential Quadratic Programming
UQ	Uncertainty Quantification
VBS	Volumetric B-Splines
w.r.t.	with respect to

Case Indices

Contents

Chapter 1

Introduction

The rapid increase in computational power and the development of efficient computational methods allowed the widespread usage of computational fluid dynamics (CFD) in many engineering applications, concerning the flow analysis around aircrafts and automotive vehicles, simulations of flows through thermal and hydraulic turbomachines and so on. The availability of reliable CFD codes has been followed by the development of a variety of methods aiming at the design/optimization of aero/hydrodynamic components. This is the scientific area, called *CFD-based optimization*, this thesis contributes to.

In shape optimization, the geometry to be redesigned is controlled by a number of variables $b_n,\ n = 1, ..., N$, referred to as the *design or optimization variables*. These, for instance, can be the coefficients of the Bézier–Bernstein polynomials parameterizing the shape of an airfoil. The mapping from the design variables to the geometry is called shape parameterization. The aerodynamic performance of the geometry to be optimized is quantified by the *objective function* J, which is usually expressed in terms of a volume or surface integral. Examples of objective functions are force components, such as the drag or lift (likely in the form of non-dimensional coefficients) exerted on isolated airfoils, the total pressure losses between the inlets and outlets of a duct, the efficiency of turbomachinery blade rows etc. The optimization problem is often constrained by equality or inequality constraints pertinent to the aerodynamic or hydrodynamic performance or geometrical properties of the shape to be optimized. The values objective function and constraints take on depend on the design variables and the flow variables; the latter are obtained by numerically solving the equations governing the fluid flow, the so-called *"primal" equations*, such as the Reynolds Averaged Navier-Stokes (RANS) PDEs. The cases studied in this thesis are either pure fluid flow problems, in which only the fluid flow equations are solved or Conjugate Heat Transfer (CHT) problems, in which the heat transfer between fluids and solids is considered while solving the governing equations on both of them. To do so, along with the equations governing the pure fluid flows, the energy equation in the fluid

and the heat conduction one in the solid are also solved. In all cases studied in this thesis, the fluid flow is governed by the RANS equations, for viscous, incompressible flows. The developed optimization methods aim at computing the values of the design variables that minimize the objective function and satisfy the constraints, if any. Throughout this thesis, maximization problems are reformulated as minimization ones.

CFD-based optimization methods are either stochastic [24, 100, 133, 13] or deterministic [47, 14, 15, 113, 91, 102, 157]. Among the stochastic optimization methods, Evolutionary Algorithms (EAs) are in widespread use. EAs apply natural evolution operations, like crossover, mutation and elitism, so that a set of candidate solutions forming a generation evolves to the next one while searching the optimal solution. Due to the randomized search, almost the entire design space is explored and entrapment to local minima can be avoided. Thus, the global extrema of the investigated objective functions can be reached, provided that enough candidate solutions are evaluated (i.e. the flow equations are solved for them and the values of the objective function are evaluated). An additional advantage of EAs is their flexibility; since no access to the source code of the flow solver is required, EAs may accommodate *black box* evaluation software. Also, EAs can compute fronts of non-dominated solutions in multi-objective optimization (MOO) problems, the so-called *Pareto fronts*. On the other hand, a high computational cost is required to reach the optimal solution(s), since the number of the CFD evaluations scale with the number of the design variables. Different kinds of methods have been developed to decrease the computational cost of EA-based optimizations. These include the use of Parallel EAs with the concurrent, centralized or decentralized, evaluation and evolution of candidate solutions [52, 4], the use of asynchronous EAs which are suitable for heterogeneous multiprocessor platforms [5, 11], the use of metamodel-assisted EAs where a surrogate evaluation model is used as often as possible to avoid the execution of time-consuming evaluation software [30, 43] and the use of hierarchical EAs [52, 87]. Contributions to these techniques, in addition to their hybridization with deterministic optimization algorithms, have been developed in PhD theses accomplished at the Parallel CFD & Optimization Unit of the National Technical University of Athens (PCOpt/NTUA), [48, 69, 10, 68, 41, 80, 83], and will not be discussed further.

Deterministic or gradient-based methods (GBMs) start with a given geometry and improve it by using the derivatives of the objective function J in hand with respect to (w.r.t.) the design variables (the gradient components are also referred to as the *sensitivity derivatives, SDs*). Since the random spawning of candidate solutions is avoided and the computed SDs yield the direction along which the design variables should be updated to minimize an objective function, GBMs require less optimization cycles than EAs, although entrapment in local minima is possible. The efficiency of GBMs is strongly related to the method used to compute the required SDs. Finite Differences (FDs) stand for the simplest method to compute

SDs. Infinitesimally small perturbations ϵ are imposed to each design variable b_n. By perturbing each design variable, both the boundary and the interior grid nodes are displaced. Then, the primal equations are numerically solved to compute the corresponding values of J. For second-order accurate SDs, central differences are used

$$\frac{\delta J(\mathbf{b})}{\delta b_n} = \frac{J(b_1, \ldots, b_n + \epsilon, \ldots, b_N) - J(b_1, \ldots, b_n - \epsilon, \ldots, b_N)}{2\epsilon} \tag{1.1}$$

So, for each design variable, the primal equations are solved twice, resulting in a total cost of $2N$ solutions of the primal PDEs, i.e. $2N$ Equivalent Flow Solutions (EFS). Thus, using FDs in large scale/industrial optimization problems is practically infeasible. Another weakness of the FDs method is that SDs are sensitive to the chosen value of ϵ and, often, FDs are computed for several values of ϵ, by thus performing an ϵ-independence study. This process increases the computational cost even more. Despite the aforementioned weaknesses, FDs are often used to verify the accuracy of adjoint-computed SDs, due to their simplicity.

A second way to compute SDs is through the complex variable method [109, 112]. Derivatives of J w.r.t. b_n are computed as

$$\frac{\delta J(\mathbf{b})}{\delta b_n} = \frac{Im\left[J(b_1, \ldots, b_n + i\epsilon, \ldots, b_N)\right]}{\epsilon} \tag{1.2}$$

where $i = \sqrt{-1}$ and Im is the imaginary part of J. The cost also scales linearly with N. In addition, to compute the imaginary part of J, the source code has to be modified to handle complex variables, instead of real ones. For those two reasons, the complex variable method is not a common approach for computing SDs. An advantage of this method is that the dependence of the SDs accuracy on the choice of ϵ is practically negligible.

The direct differentiation (DD) is an additional way of computing SDs [134]. J is differentiated w.r.t. b_n, leading to equations which include the variations of the flow variables, e.g. velocity, pressure etc, a.k.a. the primal flow variables, w.r.t. b_n. To compute these variations, the primal equations are also differentiated w.r.t. b_n [134], giving rise to N systems of equations similar to the primal ones. Thus, a new solver has to be developed. By solving the newly derived equations, the derivatives of the flow variables w.r.t. b_n are computed, at a cost of N EFS, and, then, the SDs can be computed. Due to the aforementioned shortcomings, the use of DD is preferred in algorithms computing high-order SDs, [120, 123] and in Truncated-Newton methods [125].

Compared to the aforementioned methods, the adjoint method is the most cost-efficient one since SDs are computed at a cost which is independent of N. In the adjoint method, the costly computation of the primal variables derivatives appearing in DD is avoided. To do so, additional degrees of freedom, in the form of Lagrange multipliers, a.k.a. the "adjoint" variables, are introduced in the

optimization problem. Instead of computing the derivative of J, an augmented objective function or Lagrangian L, is defined, being equal to the sum of J and the integrated residuals of the primal equations, multiplied by the adjoint variables. Since the primal equations are satisfied, computing the derivatives of either J or L w.r.t. b_n is the same. The additional degrees of freedom help overcoming the expensive computation of the primal variables derivatives, by zeroing the expressions that multiply them. This leads to the adjoint equations and boundary conditions. The terms that remain in the derivatives of L w.r.t. b_n depend on both the primal and adjoint variables and form the adjoint SDs. To compute them, the primal and adjoint equations have to be solved first. Thus, the relevant cost per optimization cycle is two EFS only.

Adjoint methods are classified into continuous and discrete ones. In the *continuous adjoint* method [57, 62, 58, 8], J is augmented by the residuals of the primal equations in their continuous form. A mathematical development of L leads to the adjoint equations which are linear PDEs (together with their boundary conditions); the PDEs need to be discretized and numerically solved, for computing the fields of the adjoint variables. Regarding the *discrete adjoint* method [33, 6, 46], J is augmented by the discretized residuals of the primal equations. After differentiating L, a system of adjoint equations derived directly in discrete form, is solved to compute the adjoint variables.

Apart from assisting GBMs, adjoint methods are also used to compute sensitivity maps. These stand for an illustration of the normal component of the derivative of J w.r.t. the coordinates of the wall boundary nodes. Sensitivity maps are used to highlight the areas where improvement in the performance according to the selected objective function has the greatest potential and are a valuable tool for the designer, even if automatic optimization is not applied. Even when an optimization process takes place, sensitivity maps are often computed beforehand to locate which areas of the geometry make sense to be parameterized. In addition, sensitivity maps can be used to update the geometry during a shape optimization process, without using any other parameterization. In such a case, filters smoothing out the noizy parts of the sensitivity maps are applied before updating the geometry.

When an optimization process is carried out, the computation of the adjoint SDs is followed by the update of the design variables, leading to the deformation of the shape. When updating the design variables, two cases are possible: (a) Only the boundary of the computational domain is controlled by b_n and the internal grid nodes coordinates x_i are adapted accordingly, in a separate phase. (b) Both the boundary and internal grid nodes are simultaneously controlled by b_n. In both cases, there is an underlying model governing the displacement of the internal grid nodes, referred to as the Grid Displacement Model (GDM).

Regarding the continuous adjoint method, this thesis is dealing with three different ways to formulate the adjoint problem and compute SDs. Two aspects

differentiate those three formulations (a) the appearance or not of field integrals, except for surface ones, in the SDs expressions and (b) the consideration (or not) of the internal grid displacement. The former is related to the cost of the adjoint method, for reasons explained below. The latter is crucial for the accuracy of the computed SDs, as demonstrated for either continuous [70, 72] or discrete adjoint methods [163]. The three continuous adjoint formulations are listed below:

1. According to [57, 90], SDs expressions including Field Integrals arise; for this reason, this method is abbreviated as the **FI adjoint**. This method considers internal grid displacement, due to the presence, inside these field integrals, of the derivatives of the grid coordinates w.r.t. b_n, i.e. $\delta x_i/\delta b_n$, which are the so-called "grid sensitivities". To compute them, one may use central FDs, which require $2N$ grid displacements. An alternative is to differentiate the GDM equations w.r.t. b_n. As a result, expressions similar to the GDM equations arise, which provide $\delta x_i/\delta b_n$, at the cost of N grid displacements. By using either of the two alternatives, the cost of computing $\delta x_i/\delta b_n$ might become comparable to 1 EFS, depending on the grid size and the used GDM. It is important that the *FI* adjoint computes accurate SDs [57, 72, 90].

2. In [70, 72], a different approach is followed to account for the internal grid displacement, yielding SDs which consist of surface integrals, computed at lower cost. Instead of computing the costly $\delta x_i/\delta b_n$ as in the *FI* adjoint, the GDM equations are considered as additional primal PDEs included into L with additional Lagrange multipliers. Then, the adjoint GDM equations are derived and solved, to compute the adjoint grid displacement field. In other words, the main idea behind the adjoint method used to avoid the costly computation of the flow variables derivatives w.r.t. b_n, is now implemented to avoid that of $\delta x_i/\delta b_n$ in the interior of the domain. Deriving adjoint equations for a selected GDM has become a common practice in discrete adjoint [56, 99, 103, 104, 105, 111]. Nevertheless, in continuous adjoint, the adjoint to a GDM has been developed more recently [70, 72], for a Laplace-based GDM [63]. The so-computed SDs are as accurate the *FI*-computed ones [70, 72], but the relevant cost is much lower. This method stands for an Enhanced version of methods presented earlier in the literature which compute less accurate SDs. This is why this method is referred to as the **E-SI adjoint**.

3. Following [8, 60], SDs consisting of surface integrals arise but, in contrast to the *E-SI* adjoint, the internal grid displacement is neglected, by eliminating the corresponding terms in the SDs, for reasons explained later in this chapter. This reduces the accuracy of the so-computed SDs, but the relevant cost is much lower than that of the *FI* adjoint, since the computation of $\delta x_i/\delta b_n$ in the interior of the domain is avoided. The presence of only Surface Integrals in the SDs expressions and the omission of terms considering the

internal grid displacement result in calling this method as the **"Severed" SI adjoint**.

This thesis is structured along the four following axes:

- Develop the continuous adjoint method for CHT problems, by considering eddy viscosity variations and properly treating the grid sensitivities.
- Investigate the effect of the choice of the GDM on the SDs values and extend the adjoint GDM technique to various GDMs used in shape optimization.
- Develop efficient GBMs for Pareto tracing, assisted by the continuous adjoint method.
- Perform cost-efficient Uncertainty Quantification and Optimization under Uncertainties for CHT problems, based on iPCE for evaluating statistical moments and their derivatives.

A literature survey of the fields, which the aforementioned methods belong to, follows. For the sake of completeness, contributions in adjoint methods developed in previous PhD theses completed at PCOpt/NTUA are summarized. Finally, the outline of the thesis is given in section 1.6.

1.1 Literature Survey on Adjoint Methods for Single-Discipline Problems

Optimization in problems governed by PDEs was firstly carried out by means of the control theory by Lions in [88]. In aerodynamics, Pironneau [136] was the first to develop adjoint methods for potential flows [136]. In [57], Jameson presented, for the first time in the literature, the adjoint method for compressible inviscid flows, without though demonstrating its use. Applications based on the adjoint method were later presented in [59, 61, 139, 140, 142, 143]. In [140], the adjoint to the potential flow equations was used to perform optimizations in 2D airfoils. Both drag reduction and inverse design problems were tackled. In [143], the adjoint to inviscid flows was used to assist similar optimization problems as in [140] as well as problems with constraints, which were imposed as penalty functions in the Lagrangian. In [141], the adjoint method for 3D compressible inviscid flows was used in the inverse design of wing and wing-body configurations. In [139, 142], complete aircraft optimizations for inviscid flows and compressible fluids using the adjoint method were performed; distributed memory computer systems were used to solve the equations. In [59, 61], the adjoint method was extended to problems governed by the compressible Navier–Stokes equations. In [61], shape optimizations of the ONERA M6 wing and, in [59], multipoint constrained optimizations of aircrafts can be found. The aforementioned adjoint methods were

developed for structured grids. All previous works were based on SD expressions including both surface and field integrals, resembling the *FI* adjoint presented in this thesis.

In [8], Anderson presented a continuous adjoint formulation which leads to SDs consisting only of surface integrals, or the *SI* adjoint. The grid sensitivities, i.e. the grid nodes derivatives w.r.t. the design variables, are neglected in the interior of the domain. Inviscid compressible and laminar incompressible flows were studied on unstructured grids. The SDs were compared with those computed by discrete adjoint and FDs. On coarser grids, small discrepancies between the SDs of continuous adjoint method and FDs appeared. By refining the grid, differences tended to vanish, except for the regions close to the trailing edge, where they became larger. This indicated that the effect of neglecting the grid sensitivities in the interior of the domain does not vanish as the mesh is refined. Avoiding the computation of the grid sensitivities over the whole domain made this adjoint method preferable to the *FI* adjoint, due to the gains in the computational cost. In [60], a new continuous adjoint formulation for compressible inviscid flows, leading to SDs with only surface integrals, was presented by Jameson, for structured grids. This method was referred to as "reduced-gradient" adjoint since no grid sensitivities were needed inside the computational domain; in this thesis, it is referred to as "Severed" *SI* adjoint. This method was generalized in [118], by Papadimitriou and Giannakoglou, for any grid type. Recently, in [72], Kavvadias et al. were the first to develop the continuous adjoint method to the GDM; that is the aforementioned *E-SI* adjoint. This method still leads to SDs consisting of surface integrals and, hence, is almost as cheap as the (Severed) *SI* adjoint. Also, considering the internal grid displacement makes the *E-SI* method as accurate as the *FI* adjoint. The differentiation of a Laplace GDM yields adjoint grid displacement PDEs, which provide the adjoint grid displacement field. In this way, the costly computation of the grid sensitivities in the interior of the computational domain is avoided. In this thesis, the effect of assuming that a Laplace GDM governs the internal grid displacement is investigated in Chapter 5.

In [21] and [33], the first second-order discrete adjoint methods were presented for compressible inviscid flows and applied to the inverse design of multi-element 2D airfoils and wings. In [34], a parallelized software solving the adjoint to the Euler equations was used in shape optimizations of a complete aircraft and a wing-body configuration. The first discrete adjoint method for viscous flows was presented in [35]. In [44, 46], an iterative scheme based on the Runge-Kutta algorithm was proposed for solving the adjoint PDEs, which guarantees duality between the primal and adjoint systems in discrete form. This property ensures that the same value of the objective function is computed at each iteration, using either the primal or adjoint variables. Also the same asymptotic convergence rate between the primal and adjoint systems is acquired. This property is of high importance, since the adjoint PDEs are guaranteed to converge as long as

the primal ones do so. On the other hand, when the (steady) primal equations converge only up to a point and, then, their residuals start to oscillate, such as in cases with vortex-shedding, convergence of the adjoint equations is not guaranteed. Various methods have been proposed to circumvent this problem, such as those presented in [44, 46, 27, 99], based on the usage of Krylov subspace methods. In [103, 56, 99, 111], the adjoint method to the used GDMs was used, to avoid the computation of the grid sensitivities and reduce the computational cost even further.

Comparisons between the continuous and adjoint methods are made in [8, 45, 106, 107, 134]. During the first years of the development of adjoint methods for CFD-based optimization there was a debate on the superiority of the continuous or discrete adjoints. However, nowadays, the maturity of both of them is such that they can compute accurate SDs. Still, some main differences between the two methods remain. First, discrete adjoint methods compute SDs which are consistent with the discretized primal equations, leading to machine accurate SDs. In continuous adjoint methods, discretization schemes, applied to the adjoint equations and SDs, which are on their continuous form, lead to different stencils than those existing in their discrete adjoint counterparts. Secondly, differences arise in the treatment of the boundary conditions in continuous and discrete adjoint, leading to noticeable differences in the computed fields. A relevant demonstration can be found in [106], in which differences in the computed adjoint fields in the cells inside the boundary layer are presented, whereas at cells away from the boundary, fields are similar. Finally, the CPU cost and memory requirements of continuous adjoint are significantly lower than those of the discrete adjoint.

In turbulent flows, the differentiation of the turbulence model equations is necessary to guarantee that SDs are accurately computed. In the literature of discrete adjoint methods, the differentiation of the turbulence model PDE(s) is common. In [7, 19, 75, 76, 97, 108, 110, 135] , the discrete adjoint to commonly used turbulence models, such as the Spalart-Allmaras, k-ϵ, k-ω and k–ω SST ones can be found. Neglecting the variation of the eddy viscosity and making the "frozen turbulence" assumption was demonstrated to often deteriorate the accuracy of the SDs [29]. In the same publication, it was shown that even when differentiating a different turbulence model than that included in the primal problem, the SDs accuracy is mitigated. One decade ago, the differentiation of the turbulence models was not a common practice in the continuous adjoint method. Zymaris et al., [175] (work performed at PCOpt/NTUA) was the first to develop the continuous adjoint for a turbulence model. In particular, the adjoint equation to the low-Reynolds number (low-Re) Spalart–Allmaras turbulence model for incompressible flows was derived. In [176], the adjoint to the high-Reynolds number (high-Re) k-ϵ model along with the notion of adjoint wall functions were introduced. In [20], Bueno-Orovio et al., extended the adjoint to the Spalart–Allmaras model to compressible flows. Also, to avoid computing the derivatives of the distance field

w.r.t. design variables, an adjoint PDE pertaining to the Eikonal equation was derived for computing the adjoint distance field. In [127], the adjoint to the high-Re Spalart-Allmaras turbulence model was derived. The notion of adjoint wall functions, first presented in [176], was extended to the Spalart–Allmaras model and the adjoint to the Eikonal equation was extended to incompressible flows for both low- and high-Re turbulence models. In [131], the adjoint to the low-Re number Launder-Sharma k-ϵ model was developed. In [70, 71], the adjoint to the $k-\omega$ SST turbulence models was presented, for both the low- and the high-Re model variants. Also, in [171], the adjoint to the $k-\omega$ SST model for compressible flows can be found. In this thesis, the Spalart–Allmaras turbulence model and the corresponding adjoint method are used.

1.2 Literature Survey on Adjoint Methods for CHT Problems

The simulation of heat transfer between fluids and solids, known as CHT analysis, appears in a variety of applications, such as turbine cooling systems that make use of internal and film cooling [55, 82, 84, 73], cooling of electronics [3], heat sinks [78, 156] etc.

Beyond just solving the CHT problem (analysis problem), there is a great interest in the shape or topology optimization of relevant devices. In shape optimization, evolutionary algorithms [16, 39] and GBMs [12, 94] have been utilized. For the latter, an adjoint method can be used to compute the gradient of J, since its cost does not scale with the number of design variables.

The continuous adjoint method for topology optimization in CHT problems, used to design fluid passages inside solid bodies for a given objective function and possible constraints, was presented in recent publications [64, 93, 165]. In porosity-based topology optimization, due to the changing Fluid Solid Interface (FSI), the heat conductivity of fluids and solids is usually interpolated in the design domain in terms of the porosity field. In [165], optimizations under geometrical and thermal constraints were performed in a 2D square domain with various inlets and outlets and immersed heat sources. In [93], a U-bend and a straight duct with a backward-facing step, considered as simplified components of a turbine cooling system, were optimized. However, in both [93] and [165], the adjoint method was based on the "frozen turbulence" assumption, since the adjoint to the turbulence model equations was not formulated. In [64], the adjoint method was developed for the solid and simplified fluid flow PDEs, in order to perform topology optimization in 2D cross-sections of heat exchangers and increase their conductance. In [155], the continuous adjoint method was derived by using a 2D reduced model based on the Navier-Stokes equations, in order to optimize 2D finned heat sinks in contact with a flat heat source at their bottom.

Adjoint methods have also been developed for shape optimization in CHT problems, [2, 95]. In fact, only recently the continuous adjoint method for the equa-

tions governing both the fluid and solid domains was introduced, in [95]. Before [95], adjoints to the PDEs governing either the fluid [1, 51] or the solid [36] were developed. By imposing appropriate temperature conditions along the wall boundaries [1, 2, 36] or by using surrogate objective functions [51], no adjoint PDEs for the solid domain were derived and only the fluid flow ones were solved and vice-versa. In [1, 2], adjoint-based optimizations under constraints were performed for internal and film-cooled turbine blades. Assuming a constant temperature distribution along the wall, the solid heat conduction PDEs were either not solved at all [1] or not differentiated [2], even though the complete CHT primal problem was solved in [2]. In [51], the adjoint method to the Navier-Stokes equations was used to drive the multi-objective optimization of a 3D duct, with turbulent flow, in contact with a heat-conducting solid body. By amalgamating the total pressure losses within the duct and a surrogate heat transfer metric based on the wall-shear stress affecting the heat flux through the FSI into J, a pure fluid flow problem and its adjoint were solved. In [36], the adjoint to the solid heat conduction equation was developed for and applied to turbine blades, for constrained inverse design problems. For the turbine wall temperature, a Robin-type condition was imposed along the gas-blade and blade-cooling holes interfaces, by equating the heat fluxes through the solid and the fluid, using an empirical heat transfer coefficient and assuming constant gas and coolant temperatures. In contrast to [1, 2, 36, 51], in [95] the continuous adjoint method to the equations of both domains was derived, without though the adjoint to the turbulence model PDEs. Adjoint gradients were used to perform constrained optimizations, in order to minimize the entropy generation in non-cooled and internally cooled turbine blades.

In the context of this PhD thesis, in [49], the continuous adjoint method for CHT shape optimization was developed by avoiding, for the first time, the "frozen turbulence" assumption and by considering the internal grid displacement, by means of both the *FI* and *E-SI* adjoint methods. The significance of both practices is discussed in Chapter 2, for pure fluid flows, and results are extended and reconfirmed in Chapter 3 which pertains to CHT problems.

1.3 Literature Survey on Pareto Tracing using GBMs

The maturity of adjoint methods which compute the necessary gradients at low cost allowed the development of a few gradient-based Pareto front tracing algorithms. In the most common approach, a number of different single-objective optimization (SOO) problems, handling the sum of differently weighted objective functions, are solved. In [173], the latter was demonstrated to be computationally less expensive than evolutionary algorithms. However, the lack of an obvious mapping between weights and the Pareto front members may lead to a poor distribution of the latter [22, 77, 148, 152]. Also, if the Pareto front is not convex,

this approach fails to compute all Pareto Points (PPs) [22, 77, 152, 173].

The method proposed in [77] can be seen as a possible remedy to the poor exploration of the Pareto front caused by the improper selection of the weight value-sets. First, a few SOO problems are solved by choosing different weight values. Then, by evaluating euclidean distances between the computed PPs, less explored front areas are enriched by solving a series of constrained optimization problems. The method is supposed to compute non-convex Pareto fronts, being though sensitive to the initialization. In [152], a steepest descent method, tackling non-convex and discontinuous Pareto fronts, was presented. To compute the front in two-objective problems, its two edges are initially computed. Then, starting from the optimal solution w.r.t. the first objective function, a two-step algorithm is used. At the first step, a new point, which does not necessarily lie on the front, is computed by perturbing the design variables using the gradient of the second objective. At the second step, an auxiliary problem, presented in [37], is solved to bring the point back to the Pareto front.

In [23], the Normal-Boundary Intersection (NBI) method was presented. Minima of each objective function are first computed and, then, independent constrained optimization subproblems are solved, in which one objective function is minimized, whereas the rest are constrained to reach a target value. The selected target values correspond to equidistributed PPs. Equality constraints of [23], can be replaced by inequality ones [40, 150, 151] in each subproblem solved to trace a new PP.

In the Pareto tracing algorithm proposed in [148], the very first PP results from the solution of a SOO problem, in which one objective function is minimized. To compute each new PP, a constrained minimization problem is solved by means of a prediction-correction scheme. Equality constraints are imposed to all but the objective function to be minimized, by defining target values for them. The advantage is the control over the distance (in the objective space) between two PPs. In both steps, the equations updating the design variables are formulated based on the Karush-Kuhn-Tucker (KKT) conditions. In the prediction step, a linear system providing the derivatives of the design variables and Lagrange multipliers w.r.t. the target objectives is formulated and solved. The current PP is, then, updated using these derivatives. Upon completion of this step, due to non-linearities, an iterative correction step is required to bring this point onto the front. This is based on the Sequential Quadratic Programming (SQP) method [113]. Hessian matrices appear on the l.h.s. of the systems solved to update the design variables in both steps; their exact computation comes at a prohibitive cost, since it scales with the number of design variables. In [148], in order to avoid computing the Hessian matrix, the prediction step is omitted; the effect of the prediction step on the efficiency of the method, if the Hessian matrix is approximated, is an unanswered question further investigated in this thesis. Instead, in [148], a constant update of the design variables and Lagrange multipliers is used at the prediction

step, followed by a BFGS-assisted SQP algorithm for the correction step.

In [160, 161], a prediction-correction algorithm was also developed to compute Pareto fronts in 3D shape optimizations of turbomachinery rows. To compute a new PP, an infeasible target point in the objective space is defined and the euclidean distance from it is minimized. To do so, the prediction step of [148] is reformulated and an unconstrained minimization problem is solved in the correction step.

In this PhD thesis, in Chapter 6, the Pareto front is computed through successive applications of a prediction-correction algorithm, conceptually based on the formulation proposed in [148] and further investigated by the authors' group in [67]. Three variants of the prediction-correction scheme are assessed, looking for the least expensive one. In all variants, the exact computation of the Hessian matrix is avoided in both steps, and this makes the cost independent of the number of design variables.

1.4 Literature Survey on Uncertainty Quantification and Optimization under Uncertainties

In either pure fluid flow or CHT problems, environmental and/or manufacturing uncertainties, expressed by the "uncertain variables", propagate in the device performance. This causes uncertainties in its operation, regarding for instance the exerted forces in external aerodynamics or the cooling effectiveness in CHT problems etc, also referred to as the Quantities of Interest (QoI). In most of the CFD applications, uncertainties in both simulations and optimizations are neglected and investigations are carried out at the same application without uncertainties, based on the so-called "deterministic" methods. In the last decades though, various methods quantifying the uncertainty in the performance of devices have emerged, with the aim of computing statistical moments, such as the mean value and standard deviation of the QoI. Processes and methods computing the statistical moments of QoI are considered as Uncertainty Quantification (UQ) techniques.

According to [54], UQ methods can be classified into intrusive and non-intrusive ones. Non-intrusive methods utilize the simulation software as a "black-box" and run it a number of times to evaluate the statistical moments. Non-intrusive methods can be classified into Sampling-based, Numerical Integration, Stochastic Metamodel and Local Estimation methods [54]. The main representative of Sampling methods is the Monte Carlo method [81]. In the latter, a number of samples, corresponding to different values of the uncertain variables, are generated by assuming a probability distribution. Each such sample is used as input to the deterministic simulation software. Upon completion of the evaluations, the solution corresponding to each sample is post-processed in order to compute the

value of the QoI and then, the statistical moments. The Monte Carlo method is straightforward in its use and the moments convergence rate does not depend on the number of the uncertain variables. However, to accurately compute the moments, a huge number of the single-point software evaluations are needed, which might reach even the order of $O\left(10^6\right)$ for aerodynamic applications, [172]. Numerical Integration methods apply numerical integration algorithms, such as the Gauss Quadrature method [86], to compute statistical moments of the QoI. To do so, products of weights and values of objective functions at selected operating points are summed up. The main drawback is that the cost scales exponentially with the number of the uncertain variables [54].

Stochastic Metamodel approaches produce representations of the QoI in the form of weighted sums of basis functions in order to estimate the moments. Main representatives are the Polynomial Chaos Expansion [31, 26] (PCE; in its non-intrusive form, it is abbreviated as niPCE) and Stochastic Collocation [26, 31, 137]. In the niPCE, the coefficients are unknown and given the distribution of the uncertain variables, specific classes of orthogonal polynomials are used as basis functions. The unknown coefficients are computed through Galerkin projections and Gauss Quadrature Integration [31, 32, 115], the Linear Regression method [31, 32, 53] etc. In Stochastic Collocation, the same Gaussian nodes as those in the niPCE method based on Gauss Quadrature Integration can be used, but Lagrange polynomials are used as basis functions. In Local Estimation methods, such as the method of moments [138, 66], the QoI is expanded around the mean values the uncertain variables, by means of Taylor-series. The number of evaluations is small for first and second-order Taylor series, but the accuracy of the estimated moments is reduced in highly non-linear systems. In the second-order accurate method, to compute the mean-value and standard deviation of a QoI, its first and second derivatives w.r.t. the uncertain variables, computed at their mean value, are needed.

Regarding intrusive UQ methods, the equations governing the problem without uncertainties are modified and new software must be programmed. The most common representative is the intrusive Polynomial Chaos Expansion (iPCE) [17, 114, 124], in which the flow variables of the equations without uncertainties are analyzed in sums of orthogonal polynomials and unknown coefficient fields. For each unknown coefficient, a new equation is derived by using Galerkin projections. For the same order of accuracy, iPCE is less expensive than niPCE. A drawback of iPCE is that the equations must be mathematically derived and programmed anew, when the number of uncertain variables or the desired order of accuracy change.

Beyond computing statistical moments, interest lays in redesigning a device, so that its performance becomes less sensitive to uncertainties. Such a problem is often referred to as Optimization under Uncertainties (OuU). Before the appearance of UQ methods, OuU was performed in a naive way, by multiplying

QoI, evaluated at various operating points, with empirical or predefined weights, such as safety factors in structural problems. The risk of such approaches was the appearance of either conservative solutions or less reliable ones, depending on the choice of the weights [169]. To perform OuU by means of UQ methods and GBMs, the adjoint method is beneficial due to the low cost of computing SDs of the mean value and standard deviation w.r.t. the design variables. So far, in the literature, adjoint methods were developed for pure fluid flows based on the PCE method [147, 153] and the method of moments [38, 126]. In this PhD thesis, iPCE primal and adjoint PDEs for CHT problems undergoing uncertainties are derived, see Chapter 7. In the studied problems, there are both environmental and manufacturing uncertainties. With the developed methods, shape optimizations are carried out, too.

1.5 Adjoint Methods Developed at PCOpt/NTUA

PCOpt/NTUA has been developing and using adjoint methods for almost two decades. During this period of time, eight PhD theses have primarily or partially dealt with adjoint methods. Their contributions are listed below:

In Papadimitriou's thesis, [117], mostly continuous and also discrete adjoint methods were developed for inviscid and viscous compressible flows solved with the in-house time-marching code, to solve optimization problems in thermal turbomachinery cascades and external aerodynamics. Regarding the continuous adjoint method, emphasis was laid on formulating a method, in a common way for both structured and unstructured meshes, so that SDs are computed based only on surface integrals. The discrete adjoint method was used for the optimal adaptation of unstructured meshes, targeting the computation of functions, expressed in terms of integrals, with increased accuracy on the coarsest mesh possible. Finally, a new radial basis function network, assisted by adjoint SDs, was developed, with improved predictive capabilities for use along with EAs as a surrogate evaluation model. Applications included the design/optimization of 2D and 3D thermal turbomachinery cascades and isolated airfoils.

In Asouti's thesis, [10], both continuous and discrete adjoint methods for low-Mach compressible flows have been developed within the CPU-based version of the in-house software "PUMA". The continuous adjoint method was formulated in two ways: (a) the preconditioning of the derived adjoint PDEs and (b) the development of the adjoint to the preconditioned flow equations. Also, the discrete adjoint was derived for the preconditioned flow equations. Applications included the design/optimization of 2D compressor cascades and isolated airfoils.

In Zymaris' thesis [174], the first continuous adjoint formulation for turbulent flows was developed. In particular, the adjoint method was derived for (a) the low-Reynolds (low-Re) number variant of the Spalart–Allmaras turbulence model and (b) the high-Reynolds (high-Re) number $k-\epsilon$ model, both for incompressible

flows. An additional topic investigated was flow control optimization for turbulent flows, in which the goal was to redesign active flow control devices, by optimizing the positions and types (suction or blowing) of jets. The method and software, programmed in OpenFOAM©, were applied to the design/optimization of 2D ducts, isolated airfoils and turbomachinery cascades. Also, sensitivity maps were computed for simplified and real 3D car geometries.

In Zervogiannis' thesis [170], a "hand differentiated" discrete adjoint for turbulent flows was developed within the CPU-based version of the in-house software "PUMA", for hybrid meshes, in order to compute first- and second-order derivatives. The adjoint was also used in a posteriori error analysis, for the optimal adaptation of hybrid grids, targeting a certain level of error in the computation of objectives functions, using the coarser possible mesh. Optimizations of isolated airfoils and turbomachinery cascades were presented.

Kontoleontos, [80], expanded the adjoint method developed in [174], to deal with optimization problems governed by incompressible, turbulent flows in the presence of heat transfer. Also, the continuous adjoint method for topology optimization problems was developed, based on the low-Re Spalart–Allmaras model. Finally, the combination of the developed adjoint with an asynchronous, metamodel-assisted EA was studied, to accelerate the solution of MOO problems concerning 2D turbomachinery cascades, heat exchangers and ducts.

In Papoutsis-Kiachagias' thesis, [125], shape, flow control and topology optimization techniques were investigated. Code development was made in OpenFOAM©. The continuous adjoint was developed for the low-Re Launder-Sharma k-ϵ, [131], and the high-Re Spalart–Allmaras, [127], models, For the Spalart–Allmaras model, the adjoint wall functions technique was extended to cell-centered finite-volume schemes by applying a no-slip velocity condition at the wall boundaries. A further enhancement for both low- and high-Re variants of the Spalart–Allmaras model was the development of the adjoint Hamilton-Jacobi (Eikonal) equation, to compute the adjoint distance field and avoid, thus, using FDs for the variation of the distance field. In addition, a new method for OuU, based on the method of moments, was developed. Novelties of the proposed method were the combined use of DD and discrete adjoint for the computation of third-order derivatives, [126]. The DD was also used to compute, at low cost, matrix-vector products needed by the Truncated-Newton method which was used for topology optimization. Applications included, among others, optimization of real car geometries as well as hydraulic turbomachines, [128].

In the PhD thesis of Kavvadias [70], various developments regarding the adjoint method have been made. First, the adjoint to the $k-\omega$ SST, for both its low-Re and high-Re variants, was developed. In addition, the *E-SI* adjoint was formulated, in which the development of the adjoint grid displacement PDEs allowed the computation of accurate SDs, expressed by surface integrals, at low cost. Also, the Recursive Projection Method [74, 149] was programmed, to increase the

stability of the continuous adjoint solvers. Moreover, the adjoint to the unsteady Navier-Stokes was developed too. Finally, adjoint methods for rotor-stator interaction flows were developed. Applications included the computation of sensitivity maps and optimizations for cars, hydraulic turbomachines etc. The software accommodating the aforementioned methods was developed within OpenFOAM©.

Finally, in Tsiakas' thesis [158], the adjoint to compressible flows was developed in the GPU-based version of the in-house software "PUMA", by following the *FI* and *SI* adjoint formulations. Among others, this thesis includes the application of the developed adjoint methods along with enhanced parameterization tools also developed in the context of the same PhD thesis, to 3D turbomachines and transonic wings.

The present thesis builds upon material developed during the previous ones and, especially, [125] and [70]. In particular, both the adjoint to the turbulence model and the adjoint to the GDM (i.e. the *E-SI* adjoint), developed for pure fluid flows, are extended to CHT problems. In addition, the assumption in the *E-SI* adjoint that the grid displacement is governed by Laplace PDEs is assessed. The developed adjoint methods for pure fluid flows and CHT problems are used to assist GBMs for Pareto front tracing in multi-objective optimization problems. Finally, to carry out UQ and OuU for CHT problems at as low cost as possible, the iPCE method is developed for the primal and adjoint CHT equations.

1.6 Thesis Outline

This PhD thesis consists of seven chapters, including the Introduction, which are summarized below.

In Chapter 2, the derivation of the *FI* and *E-SI* adjoint methods is presented for steady-state flows of incompressible fluids. For turbulent flows, closure is effected by the Spalart–Allmaras model. Verifications of SDs are presented and also shape optimizations are illustrated.

In Chapter 3, the derivation of the *FI*, *SI* and *E-SI* adjoint methods for CHT problems is presented. As in Chapter 2, the Spalart–Allmaras model is used. Through academic cases, the accuracy of the adjoint SDs is verified against FDs and results regarding the "frozen-turbulence" assumption and the proper treatment of grid sensitivities, presented in Chapter 2, are extended from pure fluid flows to CHT problems. Two 2D applications are examined, regarding the shape optimization of an internally cooled turbine cascade and a cooling channel.

In Chapter 4, two 3D CHT applications regarding internal cooling systems are investigated. In the first application, two independent Single-Objective Optimizations (SOOs) are performed for a U-Bend cooling duct immersed inside a heated solid body. In the first SOO, the goal is to minimize the max. temperature over the solid body whereas, in the second, to maximize the heat flux passing from the solid body to the fluid. The optimized geometries are compared. In the second ap-

plication, two different optimizations are performed to study how the minimization of the max. solid temperature affects the volume-weighted total pressure losses of the flow in the cooling channel of a car-engine cylinder head, and vice versa.

In Chapter 5, the effect of the selected GDM on the sensitivities is investigated. First, FDs are used to compute SDs for incompressible and CHT problems, by using four different GDMs, namely a volumetric B-Splines morpher, Laplace PDEs, Delaunay Graph and Inverse Distance Weighting. Also, the *E-SI* and the *FI* adjoints are used; the latter considers the aforementioned GDMs through properly computed grid sensitivities. A term-by-term analysis of the *FI*-computed SDs is carried out and shape optimizations are presented to support the findings of the survey.

In Chapter 6, variants of a prediction-correction algorithm for computing Pareto fronts with GBMs are presented. First, the mathematical background of the prediction and correction steps and a complete shape optimization framework incorporating the variants are given. The variants are compared for flows around isolated airfoils. For this kind of problems, the Direct Differentiation of the primal and adjoint PDEs, needed to compute matrix-vector products driving a linear-Restarted GMRES solver, is also presented. It is consistently shown that using Quasi-Newton methods in both the prediction and correction steps is the fastest way to trace the Pareto front. This is used to compute the Pareto front in a 3D CHT shape optimization problem, in which the contradictory goals are: min. volume-weighted total pressure losses of the air passing through cooling fins attached to a solid body and min. max. solid temperature values.

In Chapter 7, a detailed derivation of the primal and adjoint PDEs regarding the intrusive version of the Polynomial Chaos Expansion method for CHT problems is presented. Uncertain variables are independent from each other and are governed by the normal probability density function (PDF). Problems with manufacturing and environmental uncertainties are studied. To verify the accuracy of the computed mean values and standard deviations of the QoI, the non-intrusive version of PCE as well as the Monte Carlo method are programmed and used. To perform shape optimizations, the iPCE adjoint equations are derived and the derivatives of the mean value and standard deviation of the QoI are verified against FDs. Then, shape optimizations are performed, either for the mean value or the standard deviation of the mean temperature of a solid body attached to S-shaped duct, by considering uncertainties in the inlet velocity magnitude and temperature. Studies are also performed for problems with manufacturing uncertainties in the thickness of the thermal insulation between the solid and the fluid domains.

Software was programmed/extended in the open-source CFD toolbox OpenFOAM©, which provides a cell-centered, collocated, finite-volume infrastructure for discretizing PDEs on hybrid grids and solving algebraic systems of equations.

Chapter 2

The Continuous Adjoint Method for Incompressible Turbulent Flows

In this chapter, the derivation of the *FI*, (Severed) *SI* and *E-SI* adjoint methods for turbulent flows of incompressible fluids is presented. Closure is effected by the Spalart–Allmaras turbulence model, which is differentiated when deriving the continuous adjoint method. This is done to avoid making the "frozen turbulence" assumption, in which the variation of the eddy viscosity w.r.t. b_n is neglected, often damaging the accuracy of sensitivity derivatives (SDs), [125, 70, 174].

2.1 Primal Equations

The governing equations of the (turbulent) flow problems studied in this thesis are the steady-state RANS equations for incompressible fluids, written as [85]

$$R^p = -\frac{\partial v_i}{\partial x_i} = 0 \tag{2.1a}$$

$$R^v_i = v_j \frac{\partial v_i}{\partial x_j} + \frac{\partial p}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} = 0 \tag{2.1b}$$

$$R^{\widetilde{\nu}} = v_j \frac{\partial \widetilde{\nu}}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu}}{\partial x_j}\right] - \frac{c_{b2}}{\sigma}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right)^2 - \widetilde{\nu}\mathcal{P}(\widetilde{\nu}) + \widetilde{\nu}\mathcal{D}(\widetilde{\nu}) = 0 \tag{2.1c}$$

$$R^{\Delta} = \frac{\partial (c_j \Delta)}{\partial x_j} - \Delta \frac{\partial^2 \Delta}{\partial x_j^2} - 1 = 0, \quad c_j = \frac{\partial \Delta}{\partial x_j} \tag{2.1d}$$

which correspond to the continuity, momentum, turbulence model (Spalart-Allmaras) and Hamilton-Jacobi (Eikonal) equations. These equations are solved using the SIMPLE algorithm [132]. In the above equations, p, v_i, τ_{ij} are the static pressure divided by the constant fluid density, velocity and stress components ($\tau_{ij} =$

$(\nu+\nu_t)\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)$ where ν, ν_t stand for the bulk and turbulent viscosities), respectively. The eddy viscosity coefficient ν_t is expressed for the Spalart–Allmaras model in terms of $\widetilde{\nu}$ as follows

$$\nu_t = \widetilde{\nu} f_{v_1} \tag{2.2}$$

The production and dissipation terms are given by

$$\mathcal{P}(\widetilde{\nu}) = c_{b1}\widetilde{Y}\ , \qquad \mathcal{D}(\widetilde{\nu}) = c_{w1} f_w(\widetilde{Y})\frac{\widetilde{\nu}}{\Delta^2} \tag{2.3}$$

where $\widetilde{Y}$ is computed through

$$\widetilde{Y} = Y f_{v_3} + \frac{\widetilde{\nu}}{\Delta^2\kappa^2} f_{v_2}\ , \qquad Y = \left| e_{ijk}\frac{\partial v_k}{\partial x_j}\right| \tag{2.4}$$

with Y standing for the vorticity magnitude and Δ being the min. distance of cell-centres (since a cell-centered finite-volume scheme is used) from the wall boundaries, [159] computed by solving eq. 2.1d. The turbulence model functions read

$$\begin{aligned}
f_{v_1} &= \frac{\chi^3}{\chi^3 + c_{v_1}^3}\ , \quad f_{v_2} = \frac{1}{\left(1+\frac{\chi}{c_{v_2}}\right)^3} \\
f_{v_3} &= \frac{(1+\chi f_{v1})}{c_{v2}}\left[3\left(1+\frac{\chi}{c_{v2}}\right)+\left(\frac{\chi}{c_{v2}}\right)^2\right]\left(1+\frac{\chi}{c_{v2}}\right)^{-3} \\
\chi &= \frac{\widetilde{\nu}}{\nu}\ , \quad f_w = g\left(\frac{1+c_{w_3}^6}{g^6+c_{w_3}^6}\right)^{1/6} \\
g &= r + c_{w_2}(r^6 - r)\ , \quad r = \frac{\widetilde{\nu}}{\widetilde{Y}\kappa^2\Delta^2}
\end{aligned}$$

and the constants are $c_{b1}=0.1355$, $c_{b2}=0.622$, $\kappa=0.41$, $\sigma=2/3$, $c_{w1}=\frac{c_{b1}}{\kappa^2}+\frac{(1+c_{b2})}{\sigma}$, $c_{w2}=0.3$, $c_{w3}=2$, $c_{v1}=7.1$ and $c_{v2}=5$. The Levi-Civita symbol, e_{ijk}, used in the vorticity magnitude, is

$$e_{ijk} = \begin{cases} +1 & (i,j,k) \in (1,2,3),\ (2,3,1),\ (3,1,2) \\ -1 & (i,j,k) \in (1,3,2),\ (3,2,1),\ (2,1,3) \\ 0 & i=j,\ j=k,\ k=i \end{cases} \tag{2.5}$$

The boundary conditions used to "close" the primal problem are (a) Dirichlet ones for v_i and $\widetilde{\nu}$ along with a zero Neumann condition for p at the inlet S_I and wall S_W (which is decomposed into $S_{W_p} \cup S_{W_{np}}$ pertaining to the controlled/parameterized and unparameterized boundaries)[1] domain boundaries and (b) a, usually zero, Dirichlet condition for p along with zero Neumann conditions for v_i and $\widetilde{\nu}$ at the outlet S_O.

2.2 The *FI* Continuous Adjoint

The *FI* continuous adjoint method is derived for problems governed by the RANS equations, see eqs. 2.1a-2.1d, following the technique presented in [57, 90]. The mathematical development presented in the next sections is based on the total/-material derivative of an arbitrary quantity Φ, $\delta\Phi/\delta b_n$, which accounts for the total change in Φ due to variations in b_n, as

$$\frac{\delta\Phi}{\delta b_n} = \frac{\partial\Phi}{\partial b_n} + \frac{\partial\Phi}{\partial x_k}\frac{\delta x_k}{\delta b_n} \tag{2.6}$$

In eq. 2.6, $\frac{\partial\Phi}{\partial b_n}$ pertains to the Φ variations caused by changes in the design and flow variables, without considering space deformations from the continuous point of view (or the displacement of the internal grid nodes from the discrete point of view); space deformation is considered by the second term on the r.h.s. of eq. 2.6.

2.2.1 Objective Function Definition and Differentiation

In gradient-based optimization, the gradient of the objective function should be computed, in order to update the design variables. A general objective function J, consisting of both surface and field integrals, is expressed as

$$J = J_S + J_\Omega \tag{2.7}$$

where

$$J_S = \int_S j_{S_i} n_i dS \tag{2.8a}$$

$$J_\Omega = \int_\Omega j_\Omega d\Omega \tag{2.8b}$$

[1] When using high–Re number turbulence models with wall functions, the "law of the wall" is used along the walls.

In the above equations, j_S and j_Ω are the integrands defined along the boundary S and within the volume Ω of the computational domain, respectively, and $\mathbf{n}$ stands for the outward normal unit vector along S.

2.2.2 Differentiation of J_S

The derivative of J_S is written as

$$\frac{\delta J_S}{\delta b_n} = \int_S \frac{\delta j_{S_i}}{\delta b_n} n_i dS + \int_S j_{S_i} \frac{\delta n_i}{\delta b_n} dS + \int_S j_{S_i} n_i \frac{\delta (dS)}{\delta b_n} \tag{2.9}$$

Using the chain rule, $\delta j_{S_i}/\delta b_n$ becomes

$$\frac{\delta j_{S_i}}{\delta b_n} = \frac{\partial j_{S_i}}{\partial v_k}\frac{\delta v_k}{\delta b_n} + \frac{\partial j_{S_i}}{\partial p}\frac{\delta p}{\delta b_n} + \frac{\partial j_{S_i}}{\partial \tau_{kj}}\frac{\delta \tau_{kj}}{\delta b_n} + j^g_{S_{i,k}}\frac{\delta x_k}{\delta b_n} \tag{2.10}$$

where $j^g_{S_{i,k}}$ considers any dependency of j_{S_i} on the grid nodes x_i, i.e. $j^g_{S_{i,k}} = \frac{\partial j_{S_i}}{\partial x_k}$. After substituting eq. 2.10 to eq. 2.9, we get

$$\begin{aligned}\frac{\delta J_S}{\delta b_n} = &\int_S \frac{\partial j_{S_i}}{\partial v_k} n_i \frac{\delta v_k}{\delta b_n} dS + \int_S \frac{\partial j_{S_i}}{\partial p} n_i \frac{\delta p}{\delta b_n} dS + \int_S \frac{\partial j_{S_i}}{\partial \tau_{kj}} n_i \frac{\delta \tau_{kj}}{\delta b_n} dS \\ &+ \int_S j^g_{S_{i,k}} \frac{\delta x_k}{\delta b_n} n_i dS + \int_S j_{S_i} \frac{\delta n_i}{\delta b_n} dS + \int_S j_{S_i} n_i \frac{\delta (dS)}{\delta b_n}\end{aligned} \tag{2.11}$$

Derivatives of geometrical quantities appearing in eq. 2.11 are further developed by using the following identities from the differential geometry [50],

$$\frac{\delta n_i}{\delta b_n} = -\frac{\partial_t}{\partial x_i}\left(\frac{\delta x_k}{\delta b_n} n_k\right) \tag{2.12a}$$

$$\frac{\delta (dS)}{\delta b_n} = -\kappa \frac{\delta x_k}{\delta b_n} n_k dS \tag{2.12b}$$

where $\partial_t()/\partial x_i$ and κ stand for the tangential derivative and mean curvature of the surface, respectively.

2.2.3 Differentiation of J_Ω

The derivatives of J_Ω are

$$\frac{\delta J_\Omega}{\delta b_n} = \int_\Omega \frac{\delta j_\Omega}{\delta b_n} d\Omega + \int_\Omega j_\Omega \frac{\delta (d\Omega)}{\delta b_n} \tag{2.13}$$

to be further expanded by considering the dependency of J on the primal variables. In general, j_Ω may contain:

(a) non-differential expressions of v_i, p, $\widetilde{\nu}$ which, if differentiated w.r.t. b_n, yield $\frac{\partial j_\Omega}{\partial \Phi}\frac{\delta \Phi}{\delta b_n}$, with Φ denoting any of the aforementioned three flow variables,

(b) spatial derivatives of v_i, p, $\widetilde{\nu}$, the differentiation of which yields $\frac{\delta}{\delta b_n}\left(\frac{\partial \Phi}{\partial x_i}\right)$, further developed through the following formula

$$\frac{\delta}{\delta b_n}\left(\frac{\partial \Phi}{\partial x_i}\right) = \frac{\partial}{\partial x_i}\left(\frac{\delta \Phi}{\delta b_n}\right) - \frac{\partial \Phi}{\partial x_k}\frac{\partial}{\partial x_i}\left(\frac{\delta x_k}{\delta b_n}\right) \tag{2.14}$$

applied to any quantity Φ (its proof can be found in [116]). After applying this in terms included inside field integrals, the Gauss divergence theorem and integration by parts are used and, both surface and field integrals including $\delta\Phi/\delta b_n$, arise,

(c) both (a) and (b).

Therefore, eq. 2.13 becomes

$$\begin{aligned}\frac{\delta J_\Omega}{\delta b_n} = &\int_\Omega j^v_{\Omega,i}\frac{\delta v_i}{\delta b_n}d\Omega + \int_\Omega j^p_\Omega \frac{\delta p}{\delta b_n}d\Omega + \int_\Omega j^{\widetilde{\nu}}_\Omega \frac{\delta \widetilde{\nu}}{\delta b_n}d\Omega + \int_S j^v_{S,i}\frac{\delta v_i}{\delta b_n}dS \\ &+ \int_S j^p_S \frac{\delta p}{\delta b_n}dS + \int_S j^{\widetilde{\nu}}_S \frac{\delta \widetilde{\nu}}{\delta b_n}dS\end{aligned} \tag{2.15}$$

where j^Φ_Ω includes the partial derivative $\partial j_\Omega/\partial\Phi$ plus any term that might result from the use of the Gauss divergence theorem for integrals of the form $\int_\Omega \frac{\delta}{\delta b_n}\left(\frac{\partial \Phi}{\partial x_j}\right) d\Omega$. Also, j^Φ_S accounts for the terms multiplying $\delta\Phi/\delta b_n$ inside surface integrals; the latter are non-zero only if j_Ω includes spatial derivatives[2] of v_i, p or $\widetilde{\nu}$.

[2] The objective functions used in this thesis don't yield, when differentiated w.r.t. b_n, surface integrals containing the derivatives of τ_{ij} w.r.t. b_n; hence, the latter are not included in eq. 2.15, in order to keep the notation as simple as possible.

Regarding the last integral of eq. 2.13, it can be proven, see [60] for structured and [119] for unstructured grids, that

$$\frac{\delta(d\Omega)}{\delta b_n} = \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \tag{2.16}$$

Instead of the above equation, $\delta(d\Omega)/\delta b_n$ could be computed by either Finite Differences (FDs) or by differentiating the cell volume algebraic expressions, which are functions of the grid nodes, w.r.t. b_n. By substituting eqs. 2.15 and 2.16 into eq. 2.13, this becomes

$$\begin{aligned}\frac{\delta J_\Omega}{\delta b_n} = &\int_\Omega j^v_{\Omega,i}\frac{\delta v_i}{\delta b_n}d\Omega + \int_\Omega j^p_\Omega \frac{\delta p}{\delta b_n}d\Omega + \int_\Omega j^{\widetilde{\nu}}_\Omega \frac{\delta \widetilde{\nu}}{\delta b_n}d\Omega + \int_S j^v_{S,i}\frac{\delta v_i}{\delta b_n}dS \\ &+ \int_S j^p_S \frac{\delta p}{\delta b_n}dS + \int_S j^{\widetilde{\nu}}_S \frac{\delta \widetilde{\nu}}{\delta b_n}dS + \int_\Omega j_\Omega \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\end{aligned} \tag{2.17}$$

2.2.4 Final Expression of the J Derivatives

Herein, a distinction is made between the controlled/parameterized wall boundaries S_{W_p}, the shape of which is controlled by the design variables b_n, and the uncontrolled/unparameterized walls, which remain fixed when b_n are updated. Thus, only along S_{W_p}, $\delta x_k/\delta b_n$ are not zero. With this in mind, and by substituting eqs. 2.11 and 2.17 into the derivative of eq. 2.7, the final expression for the derivatives of any function J w.r.t. to b_n becomes

$$\begin{aligned}\frac{\delta J}{\delta b_n} = &\int_S \left(\frac{\partial j_{S_k}}{\partial v_i}n_k + j^v_{S,i}\right)\frac{\delta v_i}{\delta b_n}dS + \int_S \left(\frac{\partial j_{S_i}}{\partial p}n_i + j^p_S\right)\frac{\delta p}{\delta b_n}dS + \int_S \frac{\partial j_{S_i}}{\partial \tau_{kj}}n_i\frac{\delta \tau_{kj}}{\delta b_n}dS \\ &+ \int_S j^{\widetilde{\nu}}_S \frac{\delta \widetilde{\nu}}{\delta b_n}dS + \int_{S_{W_p}} j^g_{S_{i,k}}\frac{\delta x_k}{\delta b_n}n_i dS + \int_{S_{W_p}} j_{S_i}\frac{\delta n_i}{\delta b_n}dS + \int_{S_{W_p}} j_{S_i}n_i\frac{\delta(dS)}{\delta b_n} \\ &+ \int_\Omega j^v_{\Omega,i}\frac{\delta v_i}{\delta b_n}d\Omega + \int_\Omega j^p_\Omega \frac{\delta p}{\delta b_n}d\Omega + \int_\Omega j^{\widetilde{\nu}}_\Omega \frac{\delta \widetilde{\nu}}{\delta b_n}d\Omega + \int_\Omega j_\Omega \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\end{aligned} \tag{2.18}$$

Eq. 2.18 includes the material derivatives $\delta()/\delta b_n$ of the flow variables w.r.t. the design variables. To compute them, the Direct Differentiation (DD) of the primal equations would be required, leading to the solution of N systems of equations, similar to the primal ones. Hence, DD has a computational cost of approximately N Equivalent Flow Solutions (EFS), i.e. as if the flow PDEs were solved N times. This cost is drastically reduced by formulating and numerically solving the adjoint equations.

2.2.5 Augmented Objective Function L

The first step in developing the adjoint method is the definition of the augmented objective function (Lagrangian) L

$$L = J + \int_\Omega q R^p d\Omega + \int_\Omega u_i R_i^v d\Omega + \int_\Omega \widetilde{\nu}_a R^{\widetilde{\nu}} d\Omega + \int_\Omega \Delta^a R^\Delta d\Omega \tag{2.19}$$

where q, u_i, $\widetilde{\nu}_a$, Δ^a are the adjoint to the pressure, velocity, turbulence model variable and distance fields. Differentiating L w.r.t. b_n yields

$$\begin{aligned}\frac{\delta L}{\delta b_n} = \frac{\delta J}{\delta b_n} + \int_\Omega q \frac{\delta R^p}{\delta b_n} d\Omega + \int_\Omega u_i \frac{\delta R_i^v}{\delta b_n} d\Omega + \int_\Omega \widetilde{\nu}_a \frac{\delta R^{\widetilde{\nu}}}{\delta b_n} d\Omega + \int_\Omega \Delta^a \frac{\delta R^\Delta}{\delta b_n} d\Omega \\ + \int_\Omega \left(q R^p + u_i R_i^v + \widetilde{\nu}_a R^{\widetilde{\nu}} + \Delta^a R^\Delta \right) \frac{\partial}{\partial x_k} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega\end{aligned} \tag{2.20}$$

where the last term on the r.h.s. of eq. 2.20 is equal to zero, due to the presence of the residuals of the primal equations. Of course, since the residuals of the primal PDEs are equal to zero, the objective function gradient can be given by $\delta L/\delta b_n$ instead of $\delta J/\delta b_n$. As seen in eqs. 2.19, additional degrees of freedom, in the form of the adjoint variables, are introduced in L, allowing to get rid of the terms multiplying the derivatives of the primal variables in $\delta L/\delta b_n$. Otherwise, the gradients of the primal variables w.r.t. b_n must be computed, through the DD of the primal PDEs [134], at cost equal to N EFS. The elimination of the expressions multiplying the derivatives of the primal variables results to as many adjoint PDEs as the primal ones per objective function, along with the corresponding boundary conditions and the SD expression. Solving the adjoint PDEs has almost the same cost as for the primal PDEs. Thus, by using the adjoint method during each optimization cycle, the cost of computing the derivatives of each objective function becomes equal to 2 EFS instead of $N+1$.

The derivation of the adjoint PDEs, boundary conditions and the SD expressions require, first of all, the mathematical development of each field integral on the r.h.s. of eq. 2.20. The development for the first term is presented in sections 2.2.2–2.2.4, leading to eq. 2.18. The detailed development for the rest of the terms follows in the next sections.

2.2.6 Differentiation of the R^p Integral

The first integral on the r.h.s. of eq. 2.20 is developed through the Gauss divergence theorem and eq. 2.14, yielding

$$\begin{aligned}\int_\Omega q\frac{\delta R^p}{\delta b_n}d\Omega &= -\int_\Omega q\frac{\delta}{\delta b_n}\left(\frac{\partial v_i}{\partial x_i}\right)d\Omega = -\int_\Omega q\frac{\partial}{\partial x_i}\left(\frac{\delta v_i}{\delta b_n}\right)d\Omega + \int_\Omega q\frac{\partial v_i}{\partial x_k}\frac{\partial}{\partial x_i}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\ &= -\int_S qn_i\frac{\delta v_i}{\delta b_n}dS + \int_\Omega \frac{\partial q}{\partial x_i}\frac{\delta v_i}{\delta b_n}d\Omega + \int_\Omega q\frac{\partial v_i}{\partial x_k}\frac{\partial}{\partial x_i}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \end{aligned} \tag{2.21}$$

2.2.7 Differentiation of the R_i^v Integral

The second integral on the r.h.s. of eq. 2.20 is developed as

$$\int_\Omega u_i\frac{\delta R_i^v}{\delta b_n}d\Omega = \underbrace{\int_\Omega u_i\frac{\delta}{\delta b_n}\left(v_j\frac{\partial v_i}{\partial x_j}\right)d\Omega}_{A^v} + \underbrace{\int_\Omega u_i\frac{\delta}{\delta b_n}\left(\frac{\partial p}{\partial x_i}\right)d\Omega}_{B^v} \underbrace{-\int_\Omega u_i\frac{\delta}{\delta b_n}\left(\frac{\partial \tau_{ij}}{\partial x_j}\right)d\Omega}_{C^v} \tag{2.22}$$

Each of the three terms is developed as

$$\begin{aligned} A^v &= \int_\Omega u_i\frac{\partial v_i}{\partial x_j}\frac{\delta v_j}{\delta b_n}d\Omega + \int_\Omega u_iv_j\frac{\partial}{\partial x_j}\left(\frac{\delta v_i}{\delta b_n}\right)d\Omega - \int_\Omega u_iv_j\frac{\partial v_i}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\ &= \int_\Omega u_j\frac{\partial v_j}{\partial x_i}\frac{\delta v_i}{\delta b_n}d\Omega + \int_S u_iv_jn_j\frac{\delta v_i}{\delta b_n}dS - \int_\Omega \frac{\partial(u_iv_j)}{\partial x_j}\frac{\delta v_i}{\delta b_n}d\Omega \\ &\quad - \int_\Omega u_iv_j\frac{\partial v_i}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \end{aligned} \tag{2.23}$$

$$\begin{aligned} B^v &= \int_\Omega u_i\frac{\partial}{\partial x_i}\left(\frac{\delta p}{\delta b_n}\right)d\Omega - \int_\Omega u_i\frac{\partial p}{\partial x_k}\frac{\partial}{\partial x_i}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\ &= \int_S u_in_i\frac{\delta p}{\delta b_n}dS - \int_\Omega \frac{\partial u_i}{\partial x_i}\frac{\delta p}{\delta b_n}d\Omega - \int_\Omega u_i\frac{\partial p}{\partial x_k}\frac{\partial}{\partial x_i}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \end{aligned} \tag{2.24}$$

$$\begin{aligned} C^v &= -\int_\Omega u_i\frac{\partial}{\partial x_j}\left(\frac{\delta \tau_{ij}}{\delta b_n}\right)d\Omega + \int_\Omega u_i\frac{\partial \tau_{ij}}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\ &= -\int_S u_in_j\frac{\delta \tau_{ij}}{\delta b_n}dS + \int_\Omega \frac{\partial u_i}{\partial x_j}\frac{\delta \tau_{ij}}{\delta b_n}d\Omega + \int_\Omega u_i\frac{\partial \tau_{ij}}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\ &= -\int_S u_in_j\frac{\delta \tau_{ij}}{\delta b_n}dS + \int_\Omega u_i\frac{\partial \tau_{ij}}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega + \int_\Omega \frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)\frac{\delta \nu_t}{\delta b_n}d\Omega \\ &\quad + \int_\Omega (\nu+\nu_t)\frac{\partial u_i}{\partial x_j}\left[\frac{\partial}{\partial x_j}\left(\frac{\delta v_i}{\delta b_n}\right) - \frac{\partial v_i}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) + \frac{\partial}{\partial x_i}\left(\frac{\delta v_j}{\delta b_n}\right) - \frac{\partial v_j}{\partial x_k}\frac{\partial}{\partial x_i}\left(\frac{\delta x_k}{\delta b_n}\right)\right]d\Omega \end{aligned}$$

$$
\begin{aligned}
=&-\int_S u_i n_j \frac{\delta\tau_{ij}}{\delta b_n} dS + \int_\Omega u_i \frac{\partial \tau_{ij}}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega + \int_\Omega \frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)\frac{\delta \nu_t}{\delta b_n} d\Omega \\
&+\int_S \nu_{eff}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) n_j \frac{\delta v_i}{\delta b_n} dS - \int_\Omega \frac{\partial}{\partial x_j}\left[\nu_{eff}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right]\frac{\delta v_i}{\delta b_n} d\Omega \\
&-\int_\Omega \nu_{eff}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\frac{\partial v_i}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega
\end{aligned} \tag{2.25}
$$

Substituting eqs. 2.23–2.25 into eq. 2.22 leads to

$$
\begin{aligned}
\int_\Omega u_i \frac{\delta R_i^v}{\delta b_n} d\Omega =& \int_\Omega u_j \frac{\partial v_j}{\partial x_i}\frac{\delta v_i}{\delta b_n} d\Omega + \int_S u_i v_j n_j \frac{\delta v_i}{\delta b_n} dS - \int_\Omega \frac{\partial (u_i v_j)}{\partial x_j}\frac{\delta v_i}{\delta b_n} d\Omega \\
&-\int_\Omega u_i v_j \frac{\partial v_i}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega + \int_S u_i n_i \frac{\delta p}{\delta b_n} dS - \int_\Omega \frac{\partial u_i}{\partial x_i}\frac{\delta p}{\delta b_n} d\Omega \\
&-\int_\Omega u_j \frac{\partial p}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega - \int_S u_i n_j \frac{\delta \tau_{ij}}{\delta b_n} dS + \int_\Omega u_i \frac{\partial \tau_{ij}}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\
&+\int_\Omega \frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)\frac{\delta \nu_t}{\delta b_n} d\Omega + \int_S \nu_{eff}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) n_j \frac{\delta v_i}{\delta b_n} dS \\
&-\int_\Omega \nu_{eff}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\frac{\partial v_i}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\
&-\int_\Omega \frac{\partial}{\partial x_j}\left[\nu_{eff}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right]\frac{\delta v_i}{\delta b_n} d\Omega
\end{aligned} \tag{2.26}
$$

2.2.8 Differentiation of the $R^{\tilde{\nu}}$ Integral

The third field integral on the r.h.s. of eq. 2.19, namely that including the residual of the Spalart–Allmaras model, is differentiated w.r.t. b_n, yielding

$$
\begin{aligned}
\frac{\delta}{\delta b_n}\int_\Omega \tilde{\nu}_a R^{\tilde{\nu}} d\Omega =& \underbrace{\frac{\delta}{\delta b_n}\int_\Omega \tilde{\nu}_a v_j \frac{\partial \tilde{\nu}}{\partial x_j} d\Omega}_{A^{\tilde{\nu}}} - \underbrace{\frac{\delta}{\delta b_n}\int_\Omega \tilde{\nu}_a \frac{\partial}{\partial x_j}\left[\left(\frac{\nu+\tilde{\nu}}{\sigma}\right)\frac{\partial \tilde{\nu}}{\partial x_j}\right] d\Omega}_{B^{\tilde{\nu}}} \\
&- \underbrace{\frac{\delta}{\delta b_n}\int_\Omega \tilde{\nu}_a \frac{c_{b2}}{\sigma}\left(\frac{\partial \tilde{\nu}}{\partial x_j}\right)^2 d\Omega}_{C^{\tilde{\nu}}} + \underbrace{\frac{\delta}{\delta b_n}\int_\Omega \tilde{\nu}_a \tilde{\nu}\left[-\mathcal{P}(\tilde{\nu})+\mathcal{D}(\tilde{\nu})\right] d\Omega}_{D^{\tilde{\nu}}}
\end{aligned} \tag{2.27}
$$

Each of the terms on the r.h.s. of eq. 2.27 is further developed as

$$
A^{\tilde{\nu}} = \int_\Omega \tilde{\nu}_a \frac{\partial \tilde{\nu}}{\partial x_i}\frac{\delta v_i}{\delta b_n} d\Omega + \int_\Omega \tilde{\nu}_a v_j \frac{\partial}{\partial x_j}\left(\frac{\delta \tilde{\nu}}{\delta b_n}\right) d\Omega - \int_\Omega \tilde{\nu}_a v_j \frac{\partial \tilde{\nu}}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega
$$

$$
\begin{aligned}
&= \int_\Omega \widetilde{\nu}_a \frac{\partial \widetilde{\nu}}{\partial x_i} \frac{\delta v_i}{\delta b_n} d\Omega + \int_S \widetilde{\nu}_a v_j n_j \frac{\delta \widetilde{\nu}}{\delta b_n} dS - \int_\Omega \frac{\partial \left(\widetilde{\nu}_a v_j\right)}{\partial x_j} \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
&- \int_\Omega \widetilde{\nu}_a v_j \frac{\partial \widetilde{\nu}}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{2.28}
$$

$$
\begin{aligned}
B^{\tilde{\nu}} &= -\int_\Omega \widetilde{\nu}_a \frac{\partial}{\partial x_j} \left\{ \frac{\delta}{\delta b_n} \left[\left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_j} \right] \right\} d\Omega + \int_\Omega \widetilde{\nu}_a \frac{\partial}{\partial x_k} \left[\left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_j} \right] \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&= -\int_S \widetilde{\nu}_a n_j \frac{\delta}{\delta b_n} \left[\left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_j} \right] dS + \int_\Omega \frac{\partial \widetilde{\nu}_a}{\partial x_j} \frac{\delta}{\delta b_n} \left[\left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_j} \right] d\Omega \\
&+ \int_\Omega \widetilde{\nu}_a \frac{\partial}{\partial x_k} \left[\left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_j} \right] \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&= -\int_S \frac{\widetilde{\nu}_a}{\sigma} n_j \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{\delta \widetilde{\nu}}{\delta b_n} dS - \int_S \widetilde{\nu}_a \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) n_j \frac{\delta}{\delta b_n} \left(\frac{\partial \widetilde{\nu}}{\partial x_j} \right) dS + \int_\Omega \frac{\partial \widetilde{\nu}_a}{\partial x_j} \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{1}{\sigma} \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
&+ \int_\Omega \frac{\partial \widetilde{\nu}_a}{\partial x_j} \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\delta}{\delta b_n} \left(\frac{\partial \widetilde{\nu}}{\partial x_j} \right) d\Omega + \int_\Omega \widetilde{\nu}_a \frac{\partial}{\partial x_k} \left[\left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_j} \right] \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&= -\int_S \frac{\widetilde{\nu}_a}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} n_j \frac{\delta \widetilde{\nu}}{\delta b_n} dS - \int_S \widetilde{\nu}_a \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) n_j \frac{\delta}{\delta b_n} \left(\frac{\partial \widetilde{\nu}}{\partial x_j} \right) dS + \int_\Omega \frac{\partial \widetilde{\nu}_a}{\partial x_j} \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{1}{\sigma} \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
&+ \int_\Omega \frac{\partial \widetilde{\nu}_a}{\partial x_j} \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial}{\partial x_j} \left(\frac{\delta \widetilde{\nu}}{\delta b_n} \right) d\Omega - \int_\Omega \frac{\partial \widetilde{\nu}_a}{\partial x_j} \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&+ \int_\Omega \widetilde{\nu}_a \frac{\partial}{\partial x_k} \left[\left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_j} \right] \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&= -\int_S \frac{\widetilde{\nu}_a}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} n_j \frac{\delta \widetilde{\nu}}{\delta b_n} dS - \int_S \widetilde{\nu}_a \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) n_j \frac{\delta}{\delta b_n} \left(\frac{\partial \widetilde{\nu}}{\partial x_j} \right) dS + \int_\Omega \frac{\partial \widetilde{\nu}_a}{\partial x_j} \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{1}{\sigma} \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
&+ \int_S \frac{\partial \widetilde{\nu}_a}{\partial n} \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\delta \widetilde{\nu}}{\delta b_n} dS - \int_\Omega \frac{\partial}{\partial x_j} \left[\frac{\partial \widetilde{\nu}_a}{\partial x_j} \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \right] \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
&+ \int_\Omega \widetilde{\nu}_a \frac{\partial}{\partial x_k} \left[\left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_j} \right] \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&- \int_\Omega \frac{\partial \widetilde{\nu}_a}{\partial x_j} \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{2.29}
$$

$$
\begin{aligned}
C^{\tilde{\nu}} &= -\int_\Omega 2\widetilde{\nu}_a \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{\delta}{\delta b_n} \left(\frac{\partial \widetilde{\nu}}{\partial x_j} \right) d\Omega \\
&= -\int_\Omega 2\widetilde{\nu}_a \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{\partial}{\partial x_j} \left(\frac{\delta \widetilde{\nu}}{\delta b_n} \right) d\Omega + \int_\Omega 2\widetilde{\nu}_a \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{\partial \widetilde{\nu}}{\partial x_x} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&= -\int_S 2\widetilde{\nu}_a \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} n_j \frac{\delta \widetilde{\nu}}{\delta b_n} dS + \int_\Omega \frac{\partial}{\partial x_j} \left(2\widetilde{\nu}_a \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} \right) \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega
\end{aligned}
$$

$$+\int_{\Omega} 2\widetilde{\nu_a}\frac{c_{b2}}{\sigma}\frac{\partial\widetilde{\nu}}{\partial x_j}\frac{\partial\widetilde{\nu}}{\partial x_x}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \tag{2.30}$$

$$D^{\widetilde{\nu}}=\int_{\Omega}\widetilde{\nu_a}\left[-\mathcal{P}(\widetilde{\nu})+\mathcal{D}(\widetilde{\nu})\right]\frac{\delta\widetilde{\nu}}{\delta b_n}d\Omega+\int_{\Omega}\widetilde{\nu_a}\widetilde{\nu}\left(-\frac{\delta\mathcal{P}}{\delta b_n}+\frac{\delta\mathcal{D}}{\delta b_n}\right)d\Omega \tag{2.31}$$

The differentiation of the production and dissipation terms, in the last integral of eq. 2.31, yields

$$-\frac{\delta\mathcal{P}}{\delta b_n}+\frac{\delta\mathcal{D}}{\delta b_n}=\mathcal{C}_{\widetilde{\nu}}\frac{\partial\widetilde{\nu}}{\partial b_n}+\mathcal{C}_{\Delta}\frac{\partial\Delta}{\partial b_n}+\mathcal{C}_Y\frac{1}{Y}e_{mjk}\frac{\partial v_k}{\partial x_j}e_{mli}\left[\frac{\partial}{\partial x_l}\left(\frac{\delta v_i}{\delta b_n}\right)-\frac{\partial v_i}{\partial x_\lambda}\frac{\partial}{\partial x_l}\left(\frac{\delta x_\lambda}{\delta b_n}\right)\right] \tag{2.32}$$

where

$$\mathcal{C}_Y=-\left(c_{b_1}+c_{w_1}\mathcal{C}\frac{r\widetilde{\nu}}{\widetilde{Y}\Delta^2}\right)f_{v_3} \tag{2.33}$$

$$\mathcal{C}_{\Delta}=-\frac{2}{\Delta^3}\left[c_{w_1}r\,\mathcal{C}\frac{\widetilde{\nu}}{\Delta^2}\left(\Delta^2+\frac{\widetilde{\nu}f_{v_2}}{\kappa^2\widetilde{Y}}\right)+c_{w_1}f_w\widetilde{\nu}-c_{b_1}\frac{f_{v_2}}{\kappa^2}\widetilde{\nu}\right] \tag{2.34}$$

$$\mathcal{C}_{\widetilde{\nu}}=-\left(c_{b_1}+c_{w_1}\mathcal{C}\frac{r\widetilde{\nu}}{\widetilde{Y}\Delta^2}\right)\left(\frac{\partial f_{v_3}}{\partial\widetilde{\nu}}Y+\frac{f_{v_2}}{\kappa^2\Delta^2}+\frac{\partial f_{v_2}}{\partial\widetilde{\nu}}\frac{\widetilde{\nu}}{\kappa^2\Delta^2}\right)+c_{w_1}\mathcal{C}\frac{r}{\Delta^2}+c_{w_1}\frac{f_w}{\Delta^2} \tag{2.35}$$

$$C=\left[1+c_{w_2}(6r^5-1)\right]\frac{c_{w_3}^6}{g^6+c_{w_3}^6}\left(\frac{1+c_{w_3}^6}{g^6+c_{w_3}^6}\right)^{1/6} \tag{2.36}$$

$$\frac{\partial f_{v_1}}{\partial\widetilde{\nu}}=\frac{1}{\widetilde{\nu}}\frac{3\chi^3c_{v_1}^3}{\left(\chi^3+c_{v_1}^3\right)^2} \tag{2.37}$$

$$\frac{\partial f_{v_2}}{\partial\widetilde{\nu}}=-\frac{3}{\nu c_{v_2}}\left(1+\frac{\chi}{c_{v_2}}\right)^{-4} \tag{2.38}$$

$$\begin{aligned}\frac{\partial f_{v_3}}{\partial\widetilde{\nu}}=&\frac{1}{c_{v_2}}\left(\frac{f_{v_1}}{\nu}+\chi\frac{\partial f_{v_1}}{\partial\widetilde{\nu}}\right)\left[3\left(1+\frac{\chi}{c_{v_2}}\right)+\left(\frac{\chi}{c_{v_2}}\right)^2\right]\left(1+\frac{\chi}{c_{v_2}}\right)^{-3}\\&+\frac{1+\chi f_{v_1}}{\nu c_{v_2}^2}\left(3+2\frac{\chi}{c_{v_2}}\right)\left(1+\frac{\chi}{c_{v_2}}\right)^{-3}\\&-3\frac{1+\chi f_{v_1}}{\nu c_{v_2}^2}\left[3\left(1+\frac{\chi}{c_{v_2}}\right)+\left(\frac{\chi}{c_{v_2}}\right)^2\right]\left(1+\frac{\chi}{c_{v_2}}\right)^{-4}\end{aligned} \tag{2.39}$$

By substituting eq. 2.32 into eq. 2.31, the following expression

$$
\begin{aligned}
D^{\widetilde{\nu}} = & \int_{\Omega} \widetilde{\nu}_a \left[-\mathcal{P}(\widetilde{\nu})+\mathcal{D}(\widetilde{\nu})\right] \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega + \int_{S} \widetilde{\nu}_a \widetilde{\nu} \frac{C_Y}{Y} e_{jql} \frac{\partial v_l}{\partial x_q} e_{jki} n_k \frac{\delta v_i}{\delta b_n} dS \\
& - \int_{\Omega} e_{jki} \frac{\partial}{\partial x_k} \left(\widetilde{\nu}_a \widetilde{\nu} \frac{C_Y}{Y} e_{jql} \frac{\partial v_l}{\partial x_q} \right) \frac{\delta v_i}{\delta b_n} d\Omega + \int_{\Omega} \widetilde{\nu}_a \widetilde{\nu} C_{\widetilde{\nu}} \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
& + \int_{\Omega} \widetilde{\nu}_a \widetilde{\nu} C_{\Delta} \frac{\delta \Delta}{\delta b_n} d\Omega - \int_{\Omega} \widetilde{\nu}_a \widetilde{\nu} \frac{C_Y}{Y} e_{iql} \frac{\partial v_l}{\partial x_q} e_{ij\lambda} \frac{\partial v_\lambda}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{2.40}
$$

arises. Finally, by substituting eqs. 2.28, 2.29, 2.30 and 2.40 to eq. 2.27, then

$$
\begin{aligned}
\frac{\delta}{\delta b_n} \int_{\Omega} \widetilde{\nu}_a R^{\widetilde{\nu}} d\Omega = & \int_{\Omega} \widetilde{\nu}_a \frac{\partial \widetilde{\nu}}{\partial x_i} \frac{\delta v_i}{\delta b_n} d\Omega + \int_{S} \widetilde{\nu}_a v_j n_j \frac{\delta \widetilde{\nu}}{\delta b_n} dS - \int_{\Omega} \frac{\partial \left(\widetilde{\nu}_a v_j\right)}{\partial x_j} \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
& - \int_{\Omega} \widetilde{\nu}_a v_j \frac{\partial \widetilde{\nu}}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega - \int_{S} \frac{\widetilde{\nu}_a}{\sigma} \frac{\partial \widetilde{\nu}}{\partial n} \frac{\delta \widetilde{\nu}}{\delta b_n} dS \\
& - \int_{S} \widetilde{\nu}_a n_j \left(\frac{\nu+\widetilde{\nu}}{\sigma} \right) \frac{\delta}{\delta b_n} \left(\frac{\partial \widetilde{\nu}}{\partial x_j} \right) dS + \int_{\Omega} \frac{\partial \widetilde{\nu}_a}{\partial x_j} \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{1}{\sigma} \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
& + \int_{S} \frac{\partial \widetilde{\nu}_a}{\partial n} \left(\frac{\nu+\widetilde{\nu}}{\sigma} \right) \frac{\delta \widetilde{\nu}}{\delta b_n} dS - \int_{\Omega} \frac{\partial}{\partial x_j} \left[\frac{\partial \widetilde{\nu}_a}{\partial x_j} \left(\frac{\nu+\widetilde{\nu}}{\sigma} \right) \right] \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
& - \int_{\Omega} \frac{\partial \widetilde{\nu}_a}{\partial x_j} \left(\frac{\nu+\widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
& + \int_{\Omega} \widetilde{\nu}_a \frac{\partial}{\partial x_k} \left[\left(\frac{\nu+\widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}}{\partial x_j} \right] \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
& - \int_{S} 2\widetilde{\nu}_a \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} n_j \frac{\delta \widetilde{\nu}}{\delta b_n} dS + \int_{\Omega} \frac{\partial}{\partial x_j} \left(2\widetilde{\nu}_a \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} \right) \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
& + \int_{\Omega} 2\widetilde{\nu}_a \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{\partial \widetilde{\nu}}{\partial x_x} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega + \int_{\Omega} \widetilde{\nu}_a \left[-\mathcal{P}(\widetilde{\nu})+\mathcal{D}(\widetilde{\nu})\right] \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
& + \int_{S} \widetilde{\nu}_a \widetilde{\nu} \frac{C_Y}{Y} e_{jql} \frac{\partial v_l}{\partial x_q} e_{jki} n_k \frac{\delta v_i}{\delta b_n} dS - \int_{\Omega} e_{jki} \frac{\partial}{\partial x_k} \left(\widetilde{\nu}_a \widetilde{\nu} \frac{C_Y}{Y} e_{jql} \frac{\partial v_l}{\partial x_q} \right) \frac{\delta v_i}{\delta b_n} d\Omega \\
& - \int_{\Omega} \widetilde{\nu}_a \widetilde{\nu} \frac{C_Y}{Y} e_{iql} \frac{\partial v_l}{\partial x_q} e_{ij\lambda} \frac{\partial v_\lambda}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega + \int_{\Omega} \widetilde{\nu}_a \widetilde{\nu} C_{\widetilde{\nu}} \frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega \\
& + \int_{\Omega} \widetilde{\nu}_a \widetilde{\nu} C_{\Delta} \frac{\delta \Delta}{\delta b_n} d\Omega
\end{aligned} \tag{2.41}
$$

2.2.9 Differentiation of the R^{Δ} Integral

The field integral of the residual of the Eikonal equation is developed as

$$\int_{\Omega}\Delta^{a}\frac{\delta R^{\Delta}}{\delta b_{n}}d\Omega=\underbrace{\int_{\Omega}\Delta^{a}\frac{\delta}{\delta b_{n}}\left[\frac{\partial\left(c_{j}\Delta\right)}{\partial x_{j}}\right]d\Omega}_{A^{\Delta}}-\int_{\Omega}\Delta^{a}\frac{\partial^{2}\Delta}{\partial x_{j}^{2}}\frac{\delta\Delta}{\delta b_{n}}d\Omega$$
$$\underbrace{-\int_{\Omega}\Delta^{a}\Delta\frac{\delta}{\delta b_{n}}\left(\frac{\partial^{2}\Delta}{\partial x_{j}^{2}}\right)d\Omega}_{B^{\Delta}} \tag{2.42}$$

The first and last integrals on the r.h.s. of eq. 2.42 are rewritten as

$$\begin{aligned}
A^{\Delta}=&\int_{\Omega}\Delta^{a}\frac{\partial}{\partial x_{j}}\left[\frac{\delta\left(c_{j}\Delta\right)}{\delta b_{n}}\right]d\Omega-\int_{\Omega}\Delta^{a}\frac{\partial\left(c_{j}\Delta\right)}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega\\
=&\int_{S}\Delta^{a}\frac{\delta\left(c_{j}\Delta\right)}{\delta b_{n}}n_{j}dS-\int_{\Omega}\frac{\partial\Delta^{a}}{\partial x_{j}}\frac{\delta\left(c_{j}\Delta\right)}{\delta b_{n}}d\Omega-\int_{\Omega}\Delta^{a}\frac{\partial\left(c_{j}\Delta\right)}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega\\
=&\int_{S}\Delta^{a}\frac{\delta\left(c_{j}\Delta\right)}{\delta b_{n}}n_{j}dS-\int_{\Omega}\Delta^{a}\frac{\partial\left(c_{j}\Delta\right)}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega-\int_{\Omega}c_{j}\frac{\partial\Delta^{a}}{\partial x_{j}}\frac{\delta\Delta}{\delta b_{n}}d\Omega\\
&-\int_{\Omega}\Delta\frac{\partial\Delta^{a}}{\partial x_{j}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta\Delta}{\delta b_{n}}\right)d\Omega+\int_{\Omega}\Delta\frac{\partial\Delta^{a}}{\partial x_{j}}\frac{\partial\Delta}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega\\
=&\int_{S}\Delta^{a}\frac{\delta\left(c_{j}\Delta\right)}{\delta b_{n}}n_{j}dS-\int_{\Omega}\Delta^{a}\frac{\partial\left(c_{j}\Delta\right)}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega-\int_{\Omega}c_{j}\frac{\partial\Delta^{a}}{\partial x_{j}}\frac{\delta\Delta}{\delta b_{n}}d\Omega\\
&-\int_{S}\Delta\frac{\partial\Delta^{a}}{\partial n}\frac{\delta\Delta}{\delta b_{n}}dS+\int_{\Omega}\frac{\partial}{\partial x_{j}}\left(\Delta\frac{\partial\Delta^{a}}{\partial x_{j}}\right)\frac{\delta\Delta}{\delta b_{n}}d\Omega+\int_{\Omega}\Delta\frac{\partial\Delta^{a}}{\partial x_{j}}\frac{\partial\Delta}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega
\end{aligned} \tag{2.43}$$

$$\begin{aligned}
B^{\Delta}=&-\int_{\Omega}\Delta^{a}\Delta\frac{\partial}{\partial x_{j}}\left(\frac{\delta c_{j}}{\delta b_{n}}\right)d\Omega+\int_{\Omega}\Delta^{a}\Delta\frac{\partial c_{j}}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega\\
=&-\int_{S}\Delta^{a}\Delta n_{j}\frac{\delta c_{j}}{\delta b_{n}}dS+\int_{\Omega}\frac{\partial\left(\Delta^{a}\Delta\right)}{\partial x_{j}}\frac{\delta c_{j}}{\delta b_{n}}d\Omega+\int_{\Omega}\Delta^{a}\Delta\frac{\partial c_{j}}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega\\
=&-\int_{S}\Delta^{a}\Delta n_{j}\frac{\delta c_{j}}{\delta b_{n}}dS+\int_{\Omega}\Delta^{a}\Delta\frac{\partial c_{j}}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega\\
&+\int_{\Omega}\frac{\partial\left(\Delta^{a}\Delta\right)}{\partial x_{j}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta\Delta}{\delta b_{n}}\right)d\Omega-\int_{\Omega}\frac{\partial\left(\Delta^{a}\Delta\right)}{\partial x_{j}}\frac{\partial\Delta}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega\\
=&-\int_{S}\Delta^{a}\Delta n_{j}\frac{\delta c_{j}}{\delta b_{n}}dS+\int_{\Omega}\Delta^{a}\Delta\frac{\partial c_{j}}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega+\int_{S}\frac{\partial\left(\Delta^{a}\Delta\right)}{\partial x_{j}}n_{j}\frac{\delta\Delta}{\delta b_{n}}dS\\
&-\int_{\Omega}\frac{\partial^{2}\left(\Delta^{a}\Delta\right)}{\partial x_{j}^{2}}\frac{\delta\Delta}{\delta b_{n}}d\Omega-\int_{\Omega}\frac{\partial\left(\Delta^{a}\Delta\right)}{\partial x_{j}}\frac{\partial\Delta}{\partial x_{k}}\frac{\partial}{\partial x_{j}}\left(\frac{\delta x_{k}}{\delta b_{n}}\right)d\Omega
\end{aligned} \tag{2.44}$$

By substituting eqs. 2.43 and 2.44 into eq. 2.42, this becomes

$$\begin{aligned}\int_\Omega \Delta^a \frac{\delta R^\Delta}{\delta b_n} d\Omega = &\int_S \Delta^a \frac{\delta\left(c_j\Delta\right)}{\delta b_n} n_j dS - \int_\Omega \Delta^a \frac{\partial\left(c_j\Delta\right)}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\ &- \int_\Omega c_j \frac{\partial \Delta^a}{\partial x_j}\frac{\delta\Delta}{\delta b_n} d\Omega - \int_S \Delta \frac{\partial \Delta^a}{\partial n}\frac{\delta\Delta}{\delta b_n} dS \\ &+ \int_\Omega \frac{\partial}{\partial x_j}\left(\Delta\frac{\partial\Delta^a}{\partial x_j}\right)\frac{\delta\Delta}{\delta b_n} d\Omega + \int_\Omega \Delta\frac{\partial\Delta^a}{\partial x_j}\frac{\partial\Delta}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\ &- \int_\Omega \Delta^a \frac{\partial^2\Delta}{\partial x_j^2}\frac{\delta\Delta}{\delta b_n} d\Omega - \int_S \Delta^a\Delta n_j \frac{\delta c_j}{\delta b_n} dS + \int_\Omega \Delta^a\Delta\frac{\partial c_j}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\ &+ \int_S \frac{\partial\left(\Delta^a\Delta\right)}{\partial x_j} n_j \frac{\delta\Delta}{\delta b_n} dS - \int_\Omega \frac{\partial^2\left(\Delta^a\Delta\right)}{\partial x_j^2}\frac{\delta\Delta}{\delta b_n} d\Omega \\ &- \int_\Omega \frac{\partial\left(\Delta^a\Delta\right)}{\partial x_j}\frac{\partial\Delta}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \end{aligned} \tag{2.45}$$

2.2.10 Final Expression of $\delta L/\delta b_n$

By gathering and re-arranging terms included in eqs. 2.18, 2.21, 2.26, 2.41 and 2.45, the derivatives of L w.r.t. b_n become

$$\begin{aligned}\frac{\delta L}{\delta b_n} = &\int_S \left(u_i v_j n_j + \tau_{ij}^a n_j - q n_i + \widetilde{\nu}_a\widetilde{\nu}\frac{C_Y}{Y} e_{jql}\frac{\partial v_l}{\partial x_q} e_{jki} n_k + \frac{\partial j_{S_k}}{\partial v_i} n_k + j_{S,i}^v\right)\frac{\delta v_i}{\delta b_n} dS \\ &+ \int_S \left(u_i n_i + \frac{\partial j_{S_i}}{\partial p} n_i + j_S^p\right)\frac{\delta p}{\delta b_n} dS + \int_S \left(-u_i n_j + \frac{\partial j_{S_k}}{\partial \tau_{ij}} n_k\right)\frac{\delta \tau_{ij}}{\delta b_n} dS \\ &+ \int_S \left[\widetilde{\nu}_a v_j n_j - \frac{\widetilde{\nu}_a}{\sigma}\frac{\partial\widetilde{\nu}}{\partial n} + \frac{\partial\widetilde{\nu}_a}{\partial n}\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right) - 2\widetilde{\nu}_a\frac{c_{b2}}{\sigma}\frac{\partial\widetilde{\nu}}{\partial n} + j_S^{\widetilde{\nu}}\right]\frac{\delta\widetilde{\nu}}{\delta b_n} dS \\ &- \int_S \widetilde{\nu}_a n_j \left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\delta}{\delta b_n}\left(\frac{\partial\widetilde{\nu}}{\partial x_j}\right) dS + \int_S \left(\Delta^a\frac{\partial\Delta}{\partial n} + \Delta^a c_j n_j\right)\frac{\delta\Delta}{\delta b_n} dS \\ &+ \int_\Omega \left[u_j\frac{\partial v_j}{\partial x_i} - \frac{\partial\left(u_i v_j\right)}{\partial x_j} - \frac{\partial\tau_{ij}^a}{\partial x_j} + \frac{\partial q}{\partial x_i} + \widetilde{\nu}_a\frac{\partial\widetilde{\nu}}{\partial x_i} - \frac{\partial}{\partial x_l}\left(\widetilde{\nu}_a\widetilde{\nu}\frac{C_Y}{Y} e_{mjk}\frac{\partial v_k}{\partial x_j} e_{mli}\right)\right. \\ &\left. + j_{\Omega,i}^v\right]\frac{\delta v_i}{\delta b_n} d\Omega + \int_\Omega \left\{-\frac{\partial(v_j\widetilde{\nu}_a)}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\partial\widetilde{\nu}_a}{\partial x_j}\right] + \frac{1}{\sigma}\frac{\partial\widetilde{\nu}_a}{\partial x_j}\frac{\partial\widetilde{\nu}}{\partial x_j}\right. \\ &\left. + 2\frac{c_{b2}}{\sigma}\frac{\partial}{\partial x_j}\left(\widetilde{\nu}_a\frac{\partial\widetilde{\nu}}{\partial x_j}\right) + \widetilde{\nu}_a\widetilde{\nu}C_{\widetilde{\nu}} - \left(\mathcal{P}-\mathcal{D}\right)\widetilde{\nu}_a + \frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\frac{\delta\nu_t}{\delta\widetilde{\nu}} + j_\Omega^{\widetilde{\nu}}\right\}\frac{\delta\widetilde{\nu}}{\delta b_n} d\Omega \\ &+ \int_\Omega \left(-\frac{\partial u_i}{\partial x_i} + j_\Omega^p\right)\frac{\delta p}{\delta b_n} d\Omega + \int_\Omega \left[-2\frac{\partial}{\partial x_j}\left(\Delta^a\frac{\partial\Delta}{\partial x_j}\right) + \widetilde{\nu}\widetilde{\nu}_a C_\Delta\right]\frac{\delta\Delta}{\delta b_n} d\Omega \\ &+ \int_\Omega \left(-u_i v_j\frac{\partial v_i}{\partial x_k} - u_j\frac{\partial p}{\partial x_k} - \tau_{ij}^a\frac{\partial v_i}{\partial x_k} + u_i\frac{\partial\tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right)\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \end{aligned}$$

$$+\int_{\Omega} j_{\Omega}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega+\int_{S_{W_p}} j^{g}_{S_{i,k}}\frac{\delta x_k}{\delta b_n}n_i dS+\int_{S_{W_p}} j_{S,i}\frac{\delta\left(n_i dS\right)}{\delta b_n} \tag{2.46}$$

where $\tau^a_{ij}=(\nu+\nu_t)\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)$ is the adjoint stress tensor and

$$\begin{aligned}\Theta_{jk}=&-\widetilde{\nu}_a v_j\frac{\partial\widetilde{\nu}}{\partial x_k}+\widetilde{\nu}_a\frac{\partial}{\partial x_k}\left[\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\partial\widetilde{\nu}}{\partial x_j}\right]-\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\partial\widetilde{\nu}_a}{\partial x_j}\frac{\partial\widetilde{\nu}}{\partial x_k}\\&-\widetilde{\nu}_a\widetilde{\nu}\frac{C_Y}{Y}e_{iql}\frac{\partial v_l}{\partial x_q}e_{ij\lambda}\frac{\partial v_\lambda}{\partial x_k}+2\widetilde{\nu}_a\frac{c_{b2}}{\sigma}\frac{\partial\widetilde{\nu}}{\partial x_j}\frac{\partial\widetilde{\nu}}{\partial x_k}-2\Delta^a\frac{\partial\Delta}{\partial x_j}\frac{\partial\Delta}{\partial x_k}\end{aligned} \tag{2.47}$$

2.2.11 Field Adjoint Equations

By eliminating the expressions multiplying v_i, p, $\widetilde{\nu}$ and Δ w.r.t. b_n inside field integrals, the following adjoint PDEs arise

$$R^q=-\frac{\partial u_i}{\partial x_i}+j^p_{\Omega}=0 \tag{2.48a}$$

$$\begin{aligned}R^u_i=&u_j\frac{\partial v_j}{\partial x_i}-\frac{\partial(v_j u_i)}{\partial x_j}-\frac{\partial\tau^a_{ij}}{\partial x_j}+\frac{\partial q}{\partial x_i}+\widetilde{\nu}_a\frac{\partial\widetilde{\nu}}{\partial x_i}-\frac{\partial}{\partial x_l}\left(\widetilde{\nu}_a\widetilde{\nu}\frac{C_Y}{Y}e_{mjk}\frac{\partial v_k}{\partial x_j}e_{mli}\right)\\&+j^v_{\Omega,i}=0,\ i=1,2(,3)\end{aligned} \tag{2.48b}$$

$$\begin{aligned}R^{\widetilde{\nu_a}}=&-\frac{\partial(v_j\widetilde{\nu}_a)}{\partial x_j}-\frac{\partial}{\partial x_j}\left[\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\partial\widetilde{\nu}_a}{\partial x_j}\right]+\frac{1}{\sigma}\frac{\partial\widetilde{\nu}_a}{\partial x_j}\frac{\partial\widetilde{\nu}}{\partial x_j}+2\frac{c_{b2}}{\sigma}\frac{\partial}{\partial x_j}\left(\widetilde{\nu}_a\frac{\partial\widetilde{\nu}}{\partial x_j}\right)\\&+\widetilde{\nu}_a\widetilde{\nu}C_{\widetilde{\nu}}-(\mathcal{P}-\mathcal{D})\,\widetilde{\nu}_a+\frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)\frac{\delta\nu_t}{\delta\widetilde{\nu}}+j^{\widetilde{\nu}}_{\Omega}=0\end{aligned} \tag{2.48c}$$

$$R^{\Delta^a}=-2\frac{\partial}{\partial x_j}\left(\Delta^a\frac{\partial\Delta}{\partial x_j}\right)+\widetilde{\nu}\widetilde{\nu}_a C_{\Delta}=0 \tag{2.48d}$$

where R^q and R^u_i are the adjoint continuity and momentum PDEs, $R^{\widetilde{\nu_a}}$ stands for the adjoint Spalart–Allmaras PDE and R^{Δ^a} is the adjoint Eikonal equation. In this PhD thesis, eqs. 2.48a–2.48c are solved by the SIMPLE algorithm [132], i.e. the same algorithm used to solve the primal PDEs, eqs. 2.1a–2.1d. Eq. 2.48d can be solved once the rest of the adjoint equations have converged, since Δ^a is not included in any other adjoint equation.

2.2.12 Adjoint Boundary Conditions

The detailed derivation of the adjoint boundary conditions can be found in Appendix A. Along the inlet boundary S_I, the uncontrolled $S_{W_{np}}$ and controlled S_{W_p}

walls

$$u_j n_j = u_{\langle n \rangle} = -\frac{\partial j_{S_{I,i}}}{\partial p} n_i - j^p_{S_I} \tag{2.49a}$$

$$u^I_{\langle t \rangle} = \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^I_i n_j + \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^I_j n_i \tag{2.49b}$$

$$u^{II}_{\langle t \rangle} = \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^{II}_i n_j + \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^{II}_j n_i \tag{2.49c}$$

where t^I_i, t^{II}_i are the components of the tangent to the surface unit vectors, see Appendix A. Also, $\partial q/\partial n = 0$ and $\widetilde{\nu}_a = 0$. Regarding Δ^a, along S_I, $\Delta^a = 0$ and along $S_{W_{np}}$ and S_{W_p} a zero Neumann condition is imposed.

Along the outlet boundary S_O,

$$q = u_{\langle n \rangle} v_{\langle n \rangle} + 2(\nu + \nu_t)\frac{\partial u_{\langle n \rangle}}{\partial n} + \widetilde{\nu}_a \widetilde{\nu} \frac{C_Y}{Y} e_{jql} \frac{\partial v_l}{\partial x_q} e_{jki} n_k n_i + \frac{\partial j_{S_{O,k}}}{\partial v_i} n_k n_i + j^v_{S_{O,i}} n_i = 0 \tag{2.50}$$

Regarding the normal component of the adjoint velocity, the condition $\partial u_{\langle n \rangle}/\partial n = 0$ is imposed and the tangential adjoint velocity components are given by

$$v_{\langle n \rangle} u^l_{\langle t \rangle} + (\nu + \nu_t)\left(\frac{\partial u^l_{\langle t \rangle}}{\partial n} + \frac{\partial u_{\langle n \rangle}}{\partial t^l}\right) + \widetilde{\nu}_a \widetilde{\nu} \frac{C_Y}{Y} e_{jql} \frac{\partial v_l}{\partial x_q} e_{jki} n_k t^l_i + \frac{\partial j_{S_{O,k}}}{\partial v_i} n_k t^l_i + j^v_{S_{O,i}} t^l_i = 0 \tag{2.51}$$

Regarding $\widetilde{\nu}_a$,

$$\widetilde{\nu}_a v_j n_j + \frac{\partial \widetilde{\nu}_a}{\partial n}\left(\frac{\nu + \widetilde{\nu}}{\sigma}\right) + j^{\widetilde{\nu}}_S = 0 \tag{2.52}$$

and $\Delta^a = 0$.

Over the symmetry planes S_S, if any,

$$u_{\langle n \rangle} = 0 \tag{2.53}$$

and

$$\frac{\partial u^l_{\langle t \rangle}}{\partial n} = 0 \tag{2.54}$$

As seen in eqs. 2.53 and 2.54, the symmetry plane conditions for the adjoint velocity are similar to those of the primal problem. Regarding q and $\widetilde{\nu}_a$, zero Neumann boundary conditions are imposed. Also, $\Delta^a = 0$.

2.2.13 *FI* Adjoint Sensitivities

After satisfying the adjoint mean-flow PDEs and boundary conditions, and considering eqs. 2.46 and A.39, the final expression of the SDs arises, reading

$$\begin{aligned}\left.\frac{\delta J}{\delta b_n}\right|_{FI} = &\int_\Omega \left(-u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right) \frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\ &+ \int_\Omega j_\Omega \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega + \int_{S_{W_p}} j^g_{S_{i,k}} \frac{\delta x_k}{\delta b_n} n_i dS + \int_{S_{W_p}} j_{S,i} \frac{\delta\left(n_i dS\right)}{\delta b_n} \\ &- \int_{S_{W_p}} \left[\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k t_l^I t_m^I \tau_{ij} \frac{\delta(t_i^I t_j^I)}{\delta b_n}\right] dS - \int_{S_{W_p}} \left[\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k t_l^{II} t_m^{II} \tau_{ij} \frac{\delta(t_i^{II} t_j^{II})}{\delta b_n}\right] dS \\ &- \int_{S_{W_p}} \left[\left(-u_{\langle n\rangle} + \frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k n_l n_m\right) \tau_{ij} \frac{\delta(n_i n_j)}{\delta b_n}\right] dS \\ &- \int_{S_{W_p}} \left[\left(\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k (t_l^{II} t_m^I + t_l^I t_m^{II})\right) \tau_{ij} \frac{\delta(t_i^{II} t_j^I)}{\delta b_n}\right] dS \end{aligned} \tag{2.55}$$

Due to the presence of Field Integrals on the r.h.s. of the above equation, this adjoint method is abbreviated as the **FI adjoint**. The first term on the r.h.s. of eq. 2.55 requires the computation of the grid sensitivities at the internal grid nodes, which is conducted after differentiating w.r.t. b_n the equations of the Grid Displacement Model (GDM). This results in expressions which are similar to the GDM equations and have to be evaluated N times. Thus, the computation of grid sensitivities increases the cost of the *FI* adjoint.

2.3 The (Severed) *SI* and *E-SI* Adjoints

To overcome the need of computing $\delta x_k/\delta b_n$ in the interior of Ω and to reduce, even further, the cost of computing the adjoint sensitivities, alternative continuous adjoint formulations have also been developed [60, 8, 70, 72]. In [60, 8], the proposed method avoids considering internal grid displacement, by neglecting terms which were proven in [70, 72] to account for the internal grid displacement. In [70, 72], a different approach is followed. To consider the internal grid displacement m_i but avoid at the same time the expensive computation of grid sensitivities, the adjoint to the GDM equations is derived. The same idea has been applied in the *FI* adjoint to avoid the expensive computation of the variations of the flow variables, when computing the SDs. In [70, 72], a Laplace-based GDM is assumed and the Lagrangian, given by eq. 2.19, is extended by the field integrals

of the Laplace PDEs residuals, reading

$$R_i^m = \frac{\partial^2 m_i}{\partial x_j^2} = 0 \tag{2.56}$$

multiplied by the adjoint grid displacement field m_i^a; the new Lagrangian reads

$$L = J + \int_\Omega q R^p d\Omega + \int_\Omega u_i R_i^v d\Omega + \int_\Omega \widetilde{\nu}_a R^{\widetilde{\nu}} d\Omega + \int_\Omega \Delta^a R^\Delta d\Omega + \int_\Omega m_i^a R_i^m d\Omega \tag{2.57}$$

Developing the adjoint method for a Laplace-based GDM is convenient for two reasons. First, adjoint PDEs can be derived for GDMs expressed as PDEs, in contrast to GDMs, which are described by algebraic equations, such as those based on a volumetric B-Splines morpher [129], the Delaunay Graph [89] and the Inverse Distance Weighting model [92], see section 5.1. Also, as seen below, the simple form of the Laplace PDEs allows for a straightforward derivation of the adjoint GDM equations, in contrast to GDMs governed by more complicated PDEs, such as the Linear Elasticity Model, [28]. This assumption of a Laplace-based GDM is assessed in Chapter 5.

The derivation of the *E-SI* and (Severed) *SI* adjoint methods can be carried out following [70, 72] and [125], respectively, where the Leibniz theorem for integral variations is applied to develop the derivatives of eqs. 2.19 and 2.57. To avoid the lengthy mathematical development from scratch, an alternative derivation of the SDs expressions is followed. Based on the use of the material derivative for differentiating eq. 2.57 w.r.t. b_n, the first 5 terms on its r.h.s. are differentiated as presented in section 2.2, leading to eq. 2.46. The last term is differentiated by considering that

$$x_i^{new}(b_n) = x_i(b_n) = x_i^{old} + m_i\left(b_n, x_i^{old}\right) \tag{2.58}$$

where superscripts old and new denote the current mesh, where the primal PDEs are solved, and the displaced one. Based on eq. 2.58,

$$\frac{\delta x_i}{\delta b_n} = \frac{\delta m_i}{\delta b_n} \tag{2.59}$$

and by differentiating the integral including eq. 2.56

$$\begin{aligned}\frac{\delta}{\delta b_n}\int_\Omega m_i^a R_i^m d\Omega &= \int_\Omega m_i^a \frac{\delta}{\delta b_n}\left(\frac{\partial^2 m_i}{\partial x_j^{old^2}}\right) d\Omega = \int_\Omega m_i^a \frac{\partial^2}{\partial x_j^{old^2}}\left(\frac{\delta m_i}{\delta b_n}\right) d\Omega \\ &= \int_S m_i^a n_j \frac{\delta}{\delta b_n}\left(\frac{\partial m_i}{\partial x_j^{old}}\right) dS - \int_\Omega \frac{\partial m_i}{\partial x_j^{old}} \frac{\partial}{\partial x_j^{old}}\left(\frac{\delta m_i}{\delta b_n}\right) d\Omega\end{aligned}$$

$$
\begin{aligned}
&= \int_S m_i^a n_j \frac{\delta}{\delta b_n}\left(\frac{\partial m_i}{\partial x_j^{old}}\right) dS - \int_S \frac{\partial m_i^a}{\partial n}\frac{\delta m_i}{\delta b_n} dS + \int_\Omega \frac{\partial^2 m_i^a}{\partial x_j^{old^2}}\frac{\delta m_i}{\delta b_n} dS \\
&= \int_S m_i^a n_j \frac{\delta}{\delta b_n}\left(\frac{\partial m_i}{\partial x_j^{old}}\right) dS - \int_S \frac{\partial m_i^a}{\partial n}\frac{\delta x_i}{\delta b_n} dS + \int_\Omega \frac{\partial^2 m_i^a}{\partial x_j^{old^2}}\frac{\delta x_i}{\delta b_n} dS
\end{aligned} \tag{2.60}
$$

To simplify notation, spatial derivatives on the current grid are denoted as $\partial/\partial x_i$ instead of $\partial/\partial x_i^{old}$ and regarding grid sensitivities, $\delta x_i/\delta b_n$ is used instead of $\delta x_i^{new}/\delta b_n$. By substituting eqs. 2.46 and 2.60 into the derivative of eq. 2.57 and applying the Gauss divergence theorem to the fifth field integral of eq. 2.46, the following expression arises

$$
\begin{aligned}
\frac{\delta L}{\delta b_n} = &\int_S \left(u_i v_j n_j + \tau_{ij}^a n_j - q n_i + \widetilde{\nu}_a \widetilde{\nu}\frac{C_Y}{Y} e_{jql}\frac{\partial v_l}{\partial x_q} e_{jki} n_k + \frac{\partial j_{S_k}}{\partial v_i} n_k + j_{S,i}^v\right)\frac{\delta v_i}{\delta b_n} dS \\
&+ \int_S \left(u_i n_i + \frac{\partial j_{S_i}}{\partial p} n_i + j_S^p\right)\frac{\delta p}{\delta b_n} dS + \int_S \left(-u_i n_j + \frac{\partial j_{S_k}}{\partial \tau_{ij}} n_k\right)\frac{\delta \tau_{ij}}{\delta b_n} dS \\
&+ \int_S \left[\widetilde{\nu}_a v_j n_j - \frac{\widetilde{\nu}_a}{\sigma}\frac{\partial \widetilde{\nu}}{\partial n} + \frac{\partial \widetilde{\nu}_a}{\partial n}\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right) - 2\widetilde{\nu}_a \frac{c_{b2}}{\sigma}\frac{\partial \widetilde{\nu}}{\partial n} + j_S^{\widetilde{\nu}}\right]\frac{\delta \widetilde{\nu}}{\delta b_n} dS \\
&- \int_S \widetilde{\nu}_a n_j \left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\delta}{\delta b_n}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right) dS + \int_S \left(\Delta^a \frac{\partial \Delta}{\partial n} + \Delta^a c_j n_j\right)\frac{\delta \Delta}{\delta b_n} dS \\
&+ \int_S m_i^a n_j \frac{\delta}{\delta b_n}\left(\frac{\partial m_i}{\partial x_j}\right) dS - \int_S \frac{\partial m_i^a}{\partial n}\frac{\delta x_i}{\delta b_n} dS \\
&+ \int_\Omega R_i^u \frac{\delta v_i}{\delta b_n} d\Omega + \int_\Omega R^q \frac{\delta p}{\delta b_n} d\Omega + \int_\Omega R^{\widetilde{\nu_a}}\frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega + \int_\Omega R^{\Delta^a}\frac{\delta \Delta}{\delta b_n} d\Omega \\
&+ \int_S \left(-u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right) n_j \frac{\delta x_k}{\delta b_n} dS \\
&+ \int_\Omega \left[\frac{\partial^2 m_k^a}{\partial {x_j}^2} - \frac{\partial}{\partial x_j}\left(-u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right)\right. \\
&\left. - \frac{\partial j_\Omega}{\partial x_k}\right]\frac{\delta x_k}{\delta b_n} d\Omega + \int_{S_{W_p}} j_\Omega n_k \frac{\delta x_k}{\delta b_n} dS + \int_{S_{W_p}} j_{S_{i,k}}^g \frac{\delta x_k}{\delta b_n} n_i dS + \int_{S_{W_p}} j_{S,i} \frac{\delta\left(n_i dS\right)}{\delta b_n}
\end{aligned} \tag{2.61}
$$

By setting the multipliers of $\frac{\delta v_i}{\delta b_n}$, $\frac{\delta p}{\delta b_n}$, $\frac{\delta \widetilde{\nu}}{\delta b_n}$ and $\frac{\delta \Delta}{\delta b_n}$ to zero, the same adjoint PDEs, boundary conditions and contributions to adjoint SDs arise. To avoid the expensive computation of $\delta x_k/\delta b_n$ in the interior of the domain, the expressions multiplying them, included in the fifth field integral of eq. 2.61, are set equal to zero, giving rise to the adjoint grid displacement PDEs, reading

$$
\frac{\partial^2 m_k^a}{\partial {x_j}^2} = \frac{\partial}{\partial x_j}\left(-u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right)
$$

$$+\frac{\partial j_\Omega}{\partial x_k}=0,\ \ k=1,2(,3) \tag{2.62}$$

Since $\partial m_i/\partial x_j$ is unknown along all boundaries, the seventh integral in eq. 2.61 is eliminated by setting $m_i^a=0$. Eq. 2.62 is solved after the adjoint mean-flow PDEs, eqs. 2.48a–2.48d, and before computing the adjoint SDs, which are given by the following expression

$$\begin{aligned}\left.\frac{\delta J}{\delta b_n}\right|_{E-SI}=&\int_{S_{W_p}}\left(-u_iv_j\frac{\partial v_i}{\partial x_k}-u_j\frac{\partial p}{\partial x_k}-\tau_{ij}^a\frac{\partial v_i}{\partial x_k}+u_i\frac{\partial \tau_{ij}}{\partial x_k}+q\frac{\partial v_j}{\partial x_k}+\Theta_{jk}\right)n_j\frac{\delta x_k}{\delta b_n}dS\\&+\int_{S_{W_p}}j_\Omega n_k\frac{\delta x_k}{\delta b_n}dS+\int_{S_{W_p}}j^g_{S_i,k}\frac{\delta x_k}{\delta b_n}n_idS+\int_{S_{W_p}}j_{S,i}\frac{\delta\left(n_idS\right)}{\delta b_n}-\int_{S_{W_p}}\frac{\partial m_i^a}{\partial n}\frac{\delta x_i}{\delta b_n}dS\\&-\int_{S_{W_p}}\left[\frac{\partial j_{S_{W_p},k}}{\partial\tau_{lm}}n_kt_l^It_m^I\tau_{ij}\frac{\delta(t_i^It_j^I)}{\delta b_n}\right]dS-\int_{S_{W_p}}\left[\frac{\partial j_{S_{W_p},k}}{\partial\tau_{lm}}n_kt_l^{II}t_m^{II}\tau_{ij}\frac{\delta(t_i^{II}t_j^{II})}{\delta b_n}\right]dS\\&-\int_{S_{W_p}}\left[\left(-u_{\langle n\rangle}+\frac{\partial j_{S_{W_p},k}}{\partial\tau_{lm}}n_kn_ln_m\right)\tau_{ij}\frac{\delta(n_in_j)}{\delta b_n}\right]dS\\&-\int_{S_{W_p}}\left[\left(\frac{\partial j_{S_{W_p},k}}{\partial\tau_{lm}}n_k(t_l^{II}t_m^I+t_l^It_m^{II})\right)\tau_{ij}\frac{\delta(t_i^{II}t_j^I)}{\delta b_n}\right]dS\end{aligned} \tag{2.63}$$

As demonstrated in [70, 72] and section 2.4, SDs computed by either eqs. 2.55 and 2.63 are practically equivalently accurate. The fifth integral on the r.h.s. of eqs. 2.63 requires the solution of the adjoint grid displacement PDEs, eq. 2.62. In [70, 72], it was demonstrated that

$$-\int_{S_{W_p}}\frac{\partial m_i^a}{\partial n}\frac{\delta x_i}{\delta b_n}dS=\int_{S_{W_p}}\left(qR^p+u_iR_i^v+\widetilde{\nu}_aR^{\widetilde{\nu}}+\Delta^aR^\Delta\right)n_k\frac{\delta x_k}{\delta b_n}dS \tag{2.64}$$

Thus, SDs could alternatively be computed as

$$\begin{aligned}\left.\frac{\delta J}{\delta b_n}\right|_{SI}=&\int_{S_{W_p}}\left(-u_iv_j\frac{\partial v_i}{\partial x_k}-u_j\frac{\partial p}{\partial x_k}-\tau_{ij}^a\frac{\partial v_i}{\partial x_k}+u_i\frac{\partial \tau_{ij}}{\partial x_k}+q\frac{\partial v_j}{\partial x_k}+\Theta_{jk}\right)n_j\frac{\delta x_k}{\delta b_n}dS\\&+\int_{S_{W_p}}j_\Omega n_k\frac{\delta x_k}{\delta b_n}dS+\int_{S_{W_p}}j^g_{S_i,k}\frac{\delta x_k}{\delta b_n}n_idS+\int_{S_{W_p}}j_{S,i}\frac{\delta\left(n_idS\right)}{\delta b_n}\\&+\int_{S_{W_p}}\left(qR^p+u_iR_i^v+\widetilde{\nu}_aR^{\widetilde{\nu}}+\Delta^aR^\Delta\right)n_k\frac{\delta x_k}{\delta b_n}dS\\&-\int_{S_{W_p}}\left[\frac{\partial j_{S_{W_p},k}}{\partial\tau_{lm}}n_kt_l^It_m^I\tau_{ij}\frac{\delta(t_i^It_j^I)}{\delta b_n}\right]dS-\int_{S_{W_p}}\left[\frac{\partial j_{S_{W_p},k}}{\partial\tau_{lm}}n_kt_l^{II}t_m^{II}\tau_{ij}\frac{\delta(t_i^{II}t_j^{II})}{\delta b_n}\right]dS\end{aligned}$$

$$
\begin{aligned}
&-\int_{S_{W_p}}\left[\left(-u_{\langle n\rangle}+\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}}n_k n_l n_m\right)\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n}\right]dS \\
&-\int_{S_{W_p}}\left[\left(\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}}n_k(t_l^{II}t_m^I+t_l^I t_m^{II})\right)\tau_{ij}\frac{\delta(t_i^{II}t_j^I)}{\delta b_n}\right]dS
\end{aligned}
\tag{2.65}
$$

In this case, eqs. 2.62 are not solved, since the so-computed SDs do not depend on m_i^a. More details about this derivation can be found in [125]. In both eqs. 2.63 and 2.65, SDs consist only of surface integrals in contrast to the *FI* adjoint. A common approach in the literature [60, 8] is the elimination of the $fifth$ on the r.h.s. of eq. 2.65, by assuming that since the residuals of the primal PDEs are zero in the interior of the computational domain, they asymptotically tend to zero on the boundaries. Even when this term is computed though, numerical error might be introduced due to the discretization of higher-order spatial derivatives along the boundary, impairing the accuracy of the computed SDs, [70, 72]. It is there shown that the elimination of this term may lead to less accurate SDs, compared to SDs computed by eqs. 2.55 and 2.63. Thus, the adjoint method which leads to SDs computed by eq. 2.65 without the fifth integral, is expressed exclusively by Surface Integrals and is, thus, called **(Severed) *SI* adjoint**. On the other hand, the adjoint method leading to eq. 2.63 is called **Enhanced-*SI* adjoint**. Term "Enhanced" is used since SDs are of higher accuracy than those computed by the (Severed) *SI* adjoint [72, 70].

In the next section, applications are used to investigate the differences between the various adjoint formulations.

2.4 Indicative Comparisons of the Adjoint Formulations

In this section, the differences in the SDs values computed by the aforementioned adjoint methods and the effect on the results of shape optimizations, are studied using some new test cases, over and above to [72, 70, 125]. Case 1, fig. 2.1, aims at the comparison between the *FI*, (Severed) *SI* and *E-SI* adjoint methods for the turbulent flow around the NACA4412 isolated airfoil, for which $Re = 10^6$, the farfield flow angle is $\alpha_\infty = 0^o$ and the non-dimensional distance of the first cell-center off the wall is $y^+ < 0.06$. Adjoint sensitivities are computed for the lift, given by

$$
J^{forces} = \int_{S_{W_p}} \left(p\delta_i^j - \tau_{ij}\right) r_i n_j dS \tag{2.66}
$$

with $\mathbf{r}$ being the vector upon which the total force vector is projected. For the lift, this is perpendicular to the freestream flow. Design variables are the x and y coordinates of 18 NURBS curve control points (CPs), fig. 2.1a, parameterizing the pressure and suction sides of the airfoil. SDs based on the *FI*, *SI* and *E-SI* adjoint

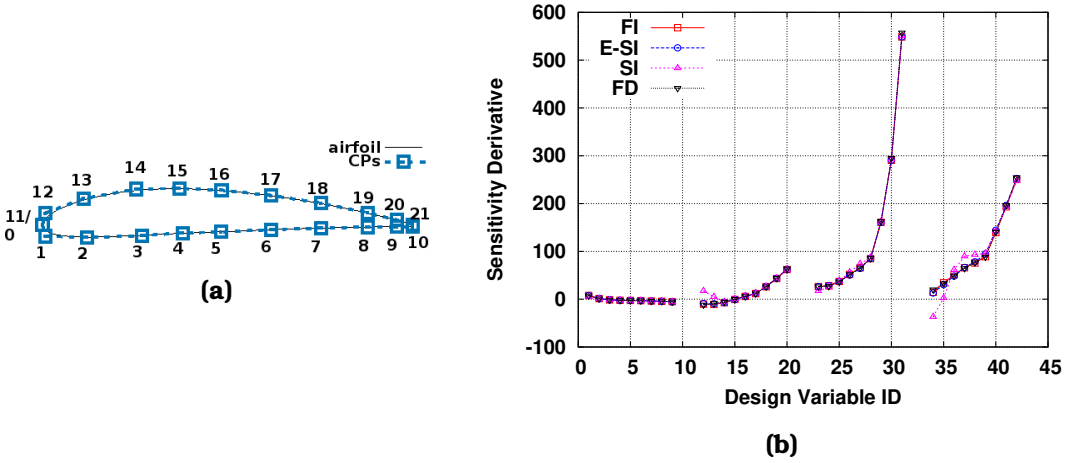


Figure 2.1: Case 1: (a) Geometry and parameterization. 22 NURBS CPs parameterize the pressure and suction sides of the NACA4412 airfoil. (b) Lift SDs computed by the *FI*, *SI* and *E-SI* adjoint methods and FDs. The first and last CPs for each side of the airfoil are excluded. The first half SDs correspond to the x coordinates and the second half to the y coordinates of the CPs.

methods are compared with FDs in fig. 2.1b. The *E-SI* and *FI* SDs reproduce the FDs, whereas the (Severed) *SI* ones have noticable differences and for some CPs, they have even the opposite sign. More details about the differentiation of this objective function can be found in Appendix B. The presented results are in agreement with applications presented in [70, 72] for other flow problems and suggest that the (Severed) *SI* adjoint should be avoided. This application also shows that neglecting the internal grid displacement mitigates the accuracy of the computed SDs; this finding is in agreement with [163], based on discrete adjoint.

In fig. 2.2, the fields of the primal and adjoint velocity magnitudes are plotted. Regarding the adjoint flow, since $u_i = -r_i$ along the airfoil wall, see eqs. A.24 and Appendix B, the adjoint velocity vector at each point has the opposite direction of the lift, see fig. 2.2c. Also, the adjoint flow follows a circular counter-clockwise path around the airfoil, fig. 2.2b, entering the domain through the airfoil pressure side and exiting through the suction side. Finally, a part of the adjoint flow seems to recirculate between the inlet and the leading edge of the airfoil. Since $u_i\big|_{S_I} = 0$, the adjoint flow reaches the inlet and reflects back towards the airfoil, as if rebounding on a solid wall.

Case 2, fig. 2.3, concerns the comparison of the *FI* and *E-SI* adjoints for the turbulent flow ($Re = 10^5$, $y^+ < 1$) through an S-shaped duct. The objective function

is the volume-weighted total pressure losses difference between S_I and S_O,

$$J^{pt} = -\int_{S_{I,O}} \left(p + \frac{1}{2} v_k^2 \right) v_j n_j dS \tag{2.67}$$

and design variables are the y coordinates of 9 volumetric B-Splines CPs parameterizing the central part of the duct, fig. 2.3a. More details on the volumetric B-Splines morpher used in this thesis can be found in Chapter 5, where the choice of the grid displacement models on the SDs accuracy is discussed.

In fig. 2.3b, *FI* and *E-SI*-computed SDs are verified against FDs. Both adjoint methods are reproducing the outcome of FDs. The *FI* and *E-SI* adjoints are also used to carry out shape optimizations, with a Quasi-Newton updating the design variables. The convergence of this method is known to be more sensitive than Steepest Descent or the Conjugate Gradient method to the accuracy of SDs [29]. This is because BFGS approximates the Hessian matrix, i.e. second derivatives of J w.r.t. b_n, by using differences between SDs of the current and previous optimization cycles to update the previous Hessian matrix [113]. Thus, slight differences in SDs values computed by different methods cause deviations between the corresponding Hessian matrices. Here, the damped version of the BFGS method is used [113]. The convergence histories of the two optimizations, plotted in fig. 2.3c, are practically the same. The optimized shape, with $\sim 8.5\%$ lower J^{pt} value, is obtained after 8 optimization cycles and the difference in the value of the objective function between the optimized geometries is 0.005%. These results confirm that the differences between SDs computed by the *FI* and *E-SI* adjoints are negligible. In fig. 2.4, the fields of velocity magnitude are plotted for the baseline and optimized geometries. To minimize J^{pt}, the cross-section area at the middle of the duct becomes larger and the S-shape is smoothed out. As a result, the velocity magnitude at the middle of the duct is reduced, leading to lower friction losses and, thus, J^{pt}.

In Case 3, fig. 2.5, it is demonstrated that considering the internal grid displacement is not enough for computing accurate SDs; the latter is achieved if the adjoint to the turbulent model equation and wall functions are implemented. The exact differentiation of the wall functions technique for the Spalart–Allmaras model, as described in [125], is used. In this case, the lift SDs are computed for the turbulent flow ($Re = 6\ 10^6$, $y^+ < 10$) around the NACA4415 airfoil, with the farfield flow angle being $\alpha_\infty = 3^o$. Design variables are the y components of the 20 volumetric B-Splines CPs parameterizing the airfoil contour, see fig. 2.5a. FDs are used to compute reference SDs. In fig. 2.5b, the adjoint method based on the adjoint to the wall function technique perfectly matches the FDs. On the other hand, significant discrepancies from the FD-computed SDs occur, in case turbulence is not differentiated.

2.5 Summary

In this chapter, the detailed derivation of the *FI* and *E-SI* continuous adjoint methods has been presented, for steady-state turbulent flows of incompressible fluids. The system of primal PDEs was closed by the Spalart–Allmaras turbulence model and the Eikonal PDE provided the required distance field; in the *E-SI* adjoint, the Laplace PDE governing the internal grid displacement are considered as additional PDEs. In contrast to the (Severed) *SI* adjoint, which made the assumption of fixed internal grid nodes, both *FI* and *E-SI* adjoints considered the internal grid displacement in different ways and computed accurate SDs. The *FI* adjoint led to SDs including field integrals of grid sensitivities which were computed by solving PDEs, arising after differentiating the used Laplace GDM w.r.t. the design variables, at a cost of N equivalent grid displacements. The *E-SI* adjoint led to SDs expressed only in terms of surface integrals. The costly computation of the grid sensitivities was avoided by developing and solving the adjoint GDM equations. In this way, the adjoint displacement field was provided and used to compute the SDs.

In two applications, the differences between the alternative adjoint formulations were studied. For the flow around an isolated airfoil, the *FI* and *E-SI* adjoints were able to reproduce FDs, in contrast to the (Severed) *SI* adjoint, for the lift being the objective function. In an internal aerodynamics problem, SDs of the volume-weighted total pressure losses, computed by the *FI* and *E-SI* adjoints and verified against FDs, assisted shape optimizations. It was demonstrated that differences between the SDs of the *FI* and *E-SI* adjoints are negligible, since the outcome of optimizations driven by Quasi-Newton methods, which are known to be quite sensitive to the SDs accuracy, was practically the same. Finally, in an external aerodynamics problem, it was demonstrated that even if the internal grid displacement is considered, accurate SDs are not computed if the turbulence model PDEs were not differentiated.

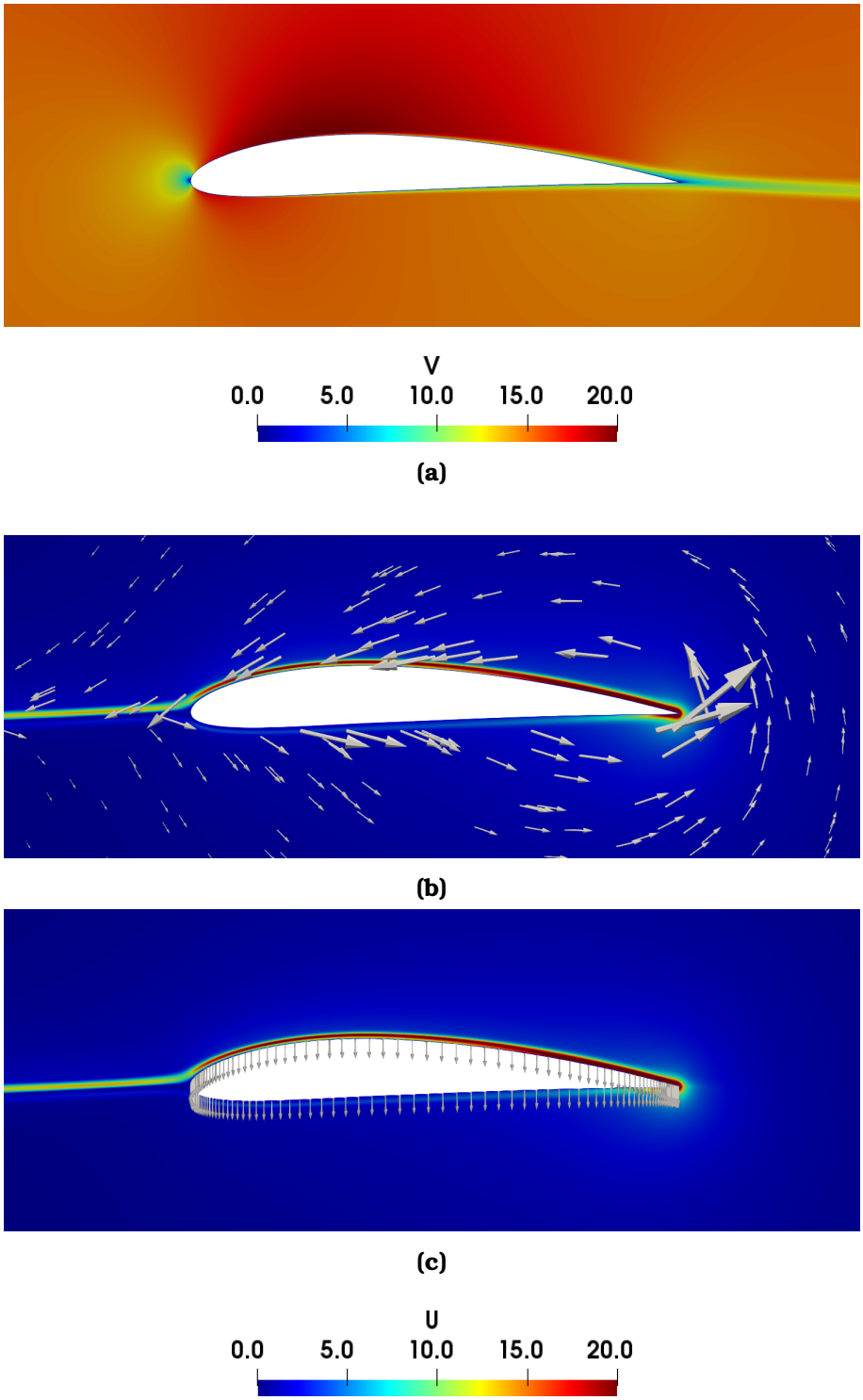
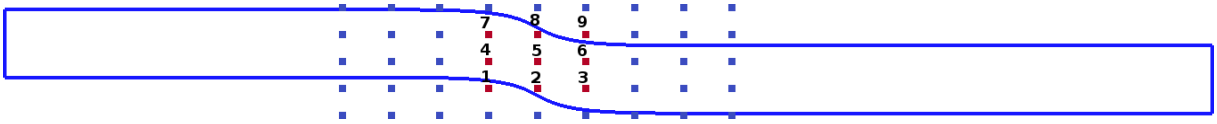
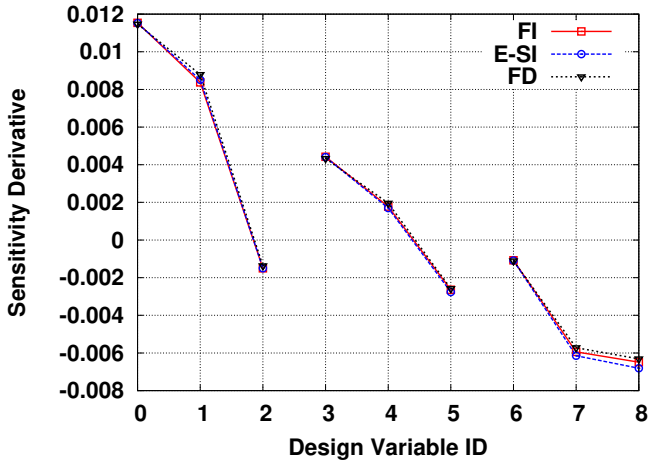


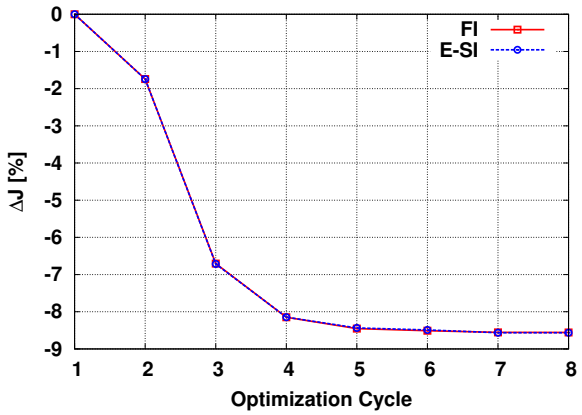
Figure 2.2: Case 1: Magnitude of (a) primal and (b,c) adjoint velocity fields. In (b) adjoint velocity vectors in the interior of the field and in (c) only along the airfoil wall are plotted.



(a) Geometry and parameterization



(b) Total pressure losses SDs



(c) Convergence of the objective function

Figure 2.3: Case 2: (a) Red CPs are considered in SDs computation. (b) *FI* and *E-SI* sensitivities verified against FDs. (c) Convergence of shape optimizations driven by the *FI* and *E-SI* adjoint methods.

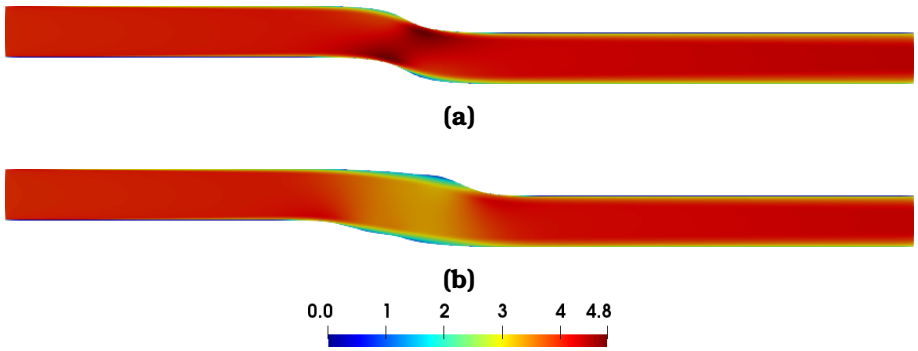


Figure 2.4: Case 2: Fields of velocity magnitude over (a) the initial and (b) optimized geometries.

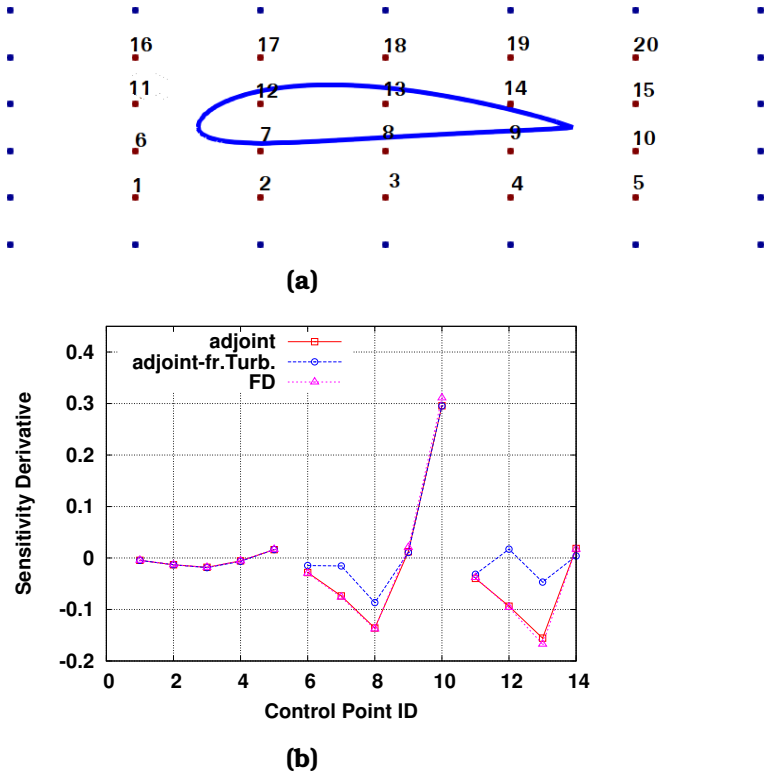


Figure 2.5: Case 3: (a) Geometry and parameterization, based on volumetric B-Splines CPs. Red CPs are considered in SDs computation. (b) SDs are computed for the y coordinates of the red colored CPs of (a).

Chapter 3

Continuous Adjoint Method for Conjugate Heat Transfer Problems

In this chapter, the mathematical development of the *FI*, *E-SI* and (Severed) *SI* adjoint methods for Conjugate Heat Transfer (CHT) problems with incompressible flows is presented. Emphasis is laid on the differentiation of the fluid energy and solid heat conduction equations. The differentiation of the rest of the primal PDEs has already been presented in detail in Chapter 2. In section 3.4, the accuracy of the SDs computed by the three adjoint methods is validated against FDs. Also, the effect of the "frozen turbulence" assumption on the accuracy of the adjoint SDs in CHT problems is investigated. In sections 3.5 and 3.6, optimizations of a 2D internally cooled turbine blade and an S-shaped cooling channel are presented.

3.1 The CHT Primal Problem

Throughout this chapter, quantities pertaining to the fluid and the solid domains are marked with superscripts F and S, respectively. For the sake of simplicity, the derivation of the adjoint method is presented for a single fluid Ω^F and a single solid Ω^S domain; extension to problems with more that one fluid or solid domains is straightforward.

Over Ω^F, the energy equation is solved along with the steady-state RANS equations for incompressible turbulent flows, closed by the one-equation Spalart-Allmaras turbulence model and the Hamilton-Jakobi (Eikonal) PDEs. The fluid flow PDEs read

$$R^p = -\frac{\partial v_j}{\partial x_j} = 0 \tag{3.1}$$

$$R^v_i = v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} + \frac{\partial p}{\partial x_i} = 0, \quad i = 1, 2(, 3) \tag{3.2}$$

$$R^{T^F} = \rho^F v_j c_p \frac{\partial T^F}{\partial x_j} + \rho^F \frac{v_j}{2} \frac{\partial v_k^2}{\partial x_j} - \frac{\partial}{\partial x_j}\left(k^F \frac{\partial T^F}{\partial x_j}\right) = 0 \tag{3.3}$$

$$R^{\tilde{\nu}} = v_j \frac{\partial \tilde{\nu}}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\frac{\nu+\tilde{\nu}}{\sigma}\right)\frac{\partial \tilde{\nu}}{\partial x_j}\right] - \frac{C_{b2}}{\sigma}\left(\frac{\partial \tilde{\nu}}{\partial x_j}\right)^2 - \tilde{\nu}\mathcal{P}(\tilde{\nu}) + \tilde{\nu}\mathcal{D}(\tilde{\nu}) = 0 \tag{3.4}$$

$$R^{\Delta} = \frac{\partial (c_j \Delta)}{\partial x_j} - \Delta \frac{\partial^2 \Delta}{\partial x_j^2} - 1 = 0, \; c_j = \frac{\partial \Delta}{\partial x_j} \tag{3.5}$$

where

$$k^F = \rho^F c_p \alpha_{eff} = \rho^F c_p \left(\frac{\nu}{Pr} + \frac{\nu_t}{Pr_t}\right) \tag{3.6}$$

is the fluid thermal conductivity, ρ^F the constant fluid density, c_p the specific heat transfer coefficient under constant pressure, Pr and Pr_t are the laminar and turbulent Prandtl numbers. The laminar, turbulent and effective thermal diffusivities are also defined as α, α_t and α_{eff}, with $\alpha_{eff} = \alpha + \alpha_t$, $\alpha = \nu/Pr$, $\alpha_t = \nu_t/Pr_t$. The rest of the terms in eqs. 3.1-3.5 can be found in Chapter 2.

On the other hand, heat conduction over Ω^S is governed by

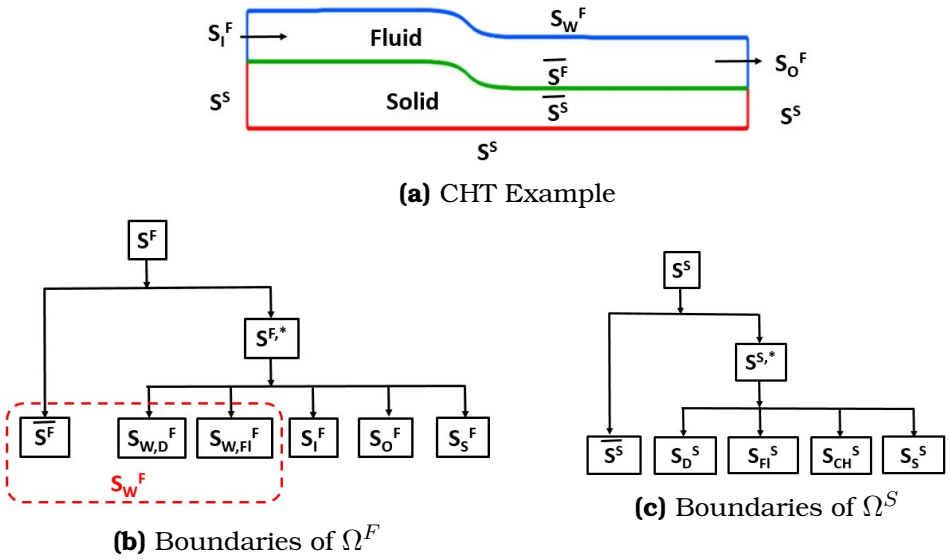
$$R^{T^S} = -\frac{\partial}{\partial x_j}\left(k^S \frac{\partial T^S}{\partial x_j}\right) = 0 \tag{3.7}$$

where k^S is the solid head conductivity. In CHT problems tackled in this thesis, the fluid and solid primal PDEs are discretized and solved in different grids, which are in contact with each other through the Fluid-Solid Interface (FSI), $\overline{S}$. Since eqs. 3.1, 3.2, 3.4 and 3.5 do not depend on T^F (and T^S), eqs. 3.3 and 3.7 can be solved after making the rest of the primal PDEs converge.

In fig. 3.1, an indicative CHT problem is presented, along with the different types of boundaries enclosing Ω^F and Ω^S. For v_i, p, $\tilde{\nu}$ and Δ, the boundary conditions can be found in Chapter 2. Regarding T^F, along S_I^F and S_O^F, Dirichlet and zero Neumann conditions are imposed, respectively. Different kinds of boundary conditions can be imposed along S_W^F, which is decomposed as $S^F = S_{W,D}^F \cup S_{W,Fl}^F \cup \overline{S}^F$. $S_{W,D}^F$ and $S_{W,Fl}^F$ are walls with fixed temperature T^F and heat flux distributions and $\overline{S}^F$ corresponds to the FSI, as seen from Ω^F, see fig. 3.1b.

The Ω^S boundaries are decomposed as $S^S = S_D^S \cup S_{Fl}^S \cup \overline{S}^S \cup S_{CH}^S$, where S_D^S and S_{Fl}^S are boundaries with fixed T^S and fixed heat-flux distributions, respectively, see see fig. 3.1c. In both Ω^F and Ω^S, adiabatic boundaries are represented by S_{Fl}^F and S_{Fl}^S, with a zero Neumann condition being imposed on temperature along them. Along S_{CH}^S, regarding 2D cooling holes, a heat flux preservation condition is imposed, reading

$$-k^S \frac{\partial T^S}{\partial n} = h\left(T^S - T_{cool}\right) \tag{3.8}$$



(a) CHT Example

(b) Boundaries of Ω^F

(c) Boundaries of Ω^S

Figure 3.1: (a) An indicative CHT problem: A solid body (red) in contact with a fluid (blue) passing through an S-shaped duct. The flow enters through S_I^F in Ω^F and exits from S_O^F. The green line stands for the FSI. (b,c) Classification of boundary types for Ω^F and Ω^S respectively. FSI boundaries are denoted by $\overline{S}$ while non-FSI ones by S^*. Regarding S_W^F and S^S, in this chapter, subscripts p, np, are used to denote whether the boundaries are controlled, or not, by the parameterization.

where T_{cool} the temperature value of a fluid passing by S_{CH}^S.

Along both sides of the FSI boundaries ($\overline{S}^F$, $\overline{S}^S$), heat-flux conservation and temperature equality lead to the FSI conditions

$$T^S\big|_{\overline{S}_S} = T^F\big|_{\overline{S}_F} \tag{3.9}$$

$$k^S \frac{\partial T^S}{\partial n}\bigg|_{\overline{S}^S} = -k^F \frac{\partial T^F}{\partial n}\bigg|_{\overline{S}^F} \tag{3.10}$$

For the outward unit normal vector at any point of the interface, $\mathbf{n}\big|_{\overline{\mathbf{S}}^\mathbf{F}} = -\mathbf{n}\big|_{\overline{\mathbf{S}}^\mathbf{S}}$. By discretizing eq. 3.10 as

$$k^F \frac{T^F\big|_{\overline{S}^F} - T^{F_I}}{FF_I} = -k^S \frac{T^S\big|_{\overline{S}^S} - T^{S_I}}{SS_I} \tag{3.11}$$

where T^{F_I} and T^{S_I} are the T^F and T^S values at the first barycenter off the wall,

see fig. 3.2, and due to eq. 3.9, eq. 3.11 yields

$$T^F\big|_{\overline{S}^F} = \frac{\frac{k^F}{FF_I}T^{F_I} + \frac{k^S}{SS_I}T^{S_I}}{\left(\frac{k^F}{FF_I} + \frac{k^S}{SS_I}\right)} \tag{3.12a}$$

$$T^S\big|_{\overline{S}^S} = \frac{\frac{k^F}{FF_I}T^{F_I} + \frac{k^S}{SS_I}T^{S_I}}{\left(\frac{k^S}{SS_I} + \frac{k^F}{FF_I}\right)} \tag{3.12b}$$

This means that a Dirichlet boundary condition is used for both T^F and T^S along the FSI boundary. In each iteration of the primal CHT solver, since the fluid energy PDE (eq. 3.3) is solved prior to the solid heat conduction equation (eq. 3.7), T^{F_I} and T^{S_I} values might differ during the numerical solution of eqs. 3.3 and 3.7. Thus, only when the primal PDEs are converged for both Ω^F and Ω^S, eqs. 3.9 and 3.10 are valid.

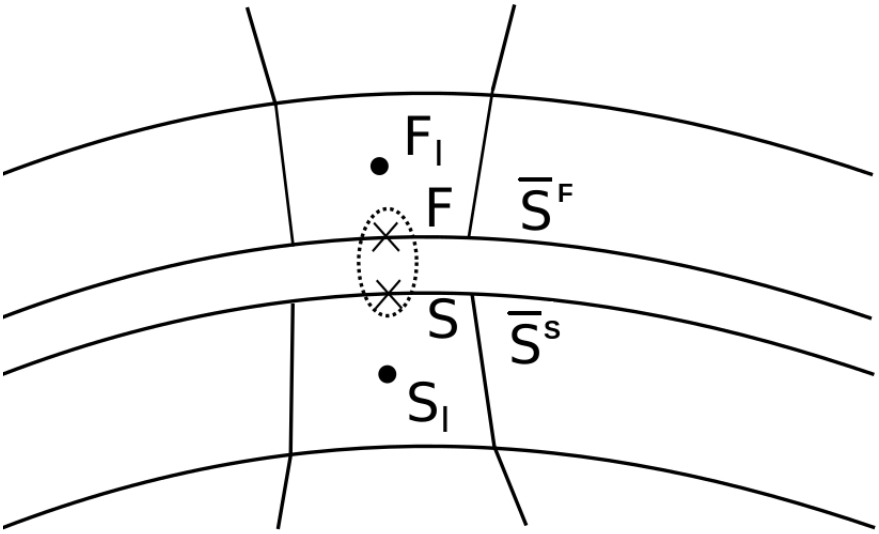


Figure 3.2: The Fluid-Solid Interface. Faces F and S coincide. F_I and S_I are the barycentres of the first cells off the fluid and solid boundaries, respectively.

3.2 The *FI* Adjoint for CHT Problems

To derive the *FI* adjoint for CHT problems, L is redefined as

$$L = J + \int_{\Omega^F} qR^p d\Omega + \int_{\Omega^F} u_i R_i^v d\Omega + \int_{\Omega^F} \widetilde{\nu}_a R^\nu d\Omega + \int_{\Omega^F} \Delta^a R^\Delta d\Omega + \sum_{D=F,S} \int_{\Omega^D} T_a^D R^{T^D} d\Omega \tag{3.13}$$

with T_a^D standing for the adjoint temperature for the fluid ($D=F$) and solid ($D=S$) domains. The derivative of L w.r.t. b_n reads

$$\frac{\delta L}{\delta b_n} = \frac{\delta J}{\delta b_n} + \frac{\delta}{\delta b_n}\int_{\Omega^F} qR^p d\Omega + \frac{\delta}{\delta b_n}\int_{\Omega^F} u_i R_i^v d\Omega + \frac{\delta}{\delta b_n}\int_{\Omega^F} \widetilde{\nu}_a R^\nu d\Omega + \frac{\delta}{\delta b_n}\int_{\Omega^F} \Delta^a R^\Delta d\Omega + \frac{\delta}{\delta b_n}\int_{\Omega^F} T_a^F R^{T^F} d\Omega + \frac{\delta}{\delta b_n}\int_{\Omega^S} T_a^S R^{T^S} d\Omega \tag{3.14}$$

The mathematical development of the second, third, fourth and fifth terms on the r.h.s. of eq. 3.14 can be found in Chapter 2. The mathematical development of the first term has to be extended, since in CHT problems objective functions are defined not only for the fluid but also for the solid and, the dependence of J on both T^F and T^S has to be considered, see section 3.2.1. The mathematical development of the last two terms on the r.h.s. of eq. 3.14 is due, in order to derive the adjoint method for CHT problems, see sections 3.2.2 and 3.2.3.

In the development that follows, subscripts p, np are used for the fluid or solid wall boundaries which are either controlled or not by the parameterization.

3.2.1 Objective Functions and their Derivatives

In CHT problems, objective functions J can be defined for either the fluid or the solid or both. In these problems, the definition of J for the fluid, see sections 2.2.2 and 2.2.3, has to be extended, to account for the dependence of $j_{S_i}^F$ and j_Ω^F (eqs. 2.8a and 2.8b) on T^F. Also, objective functions defined along the boundary are extended to take the following form

$$J_S^F = \int_{S^F} j_{S,i}^F n_i \frac{\delta(dS)}{\delta b_n} + \int_{S^F} j_S^F \frac{\delta(dS)}{\delta b_n} \tag{3.15}$$

The last term on the r.h.s. eq. 3.15 is necessary to consider the dependence of J_S^F on the heat flux $Q^F = k^F \frac{\partial T^F}{\partial n}$ and is developed as

$$\frac{\delta J_S^F}{\delta b_n} = \int_{S^F} \frac{\partial j_S^F}{\partial Q^F} \frac{\delta Q^F}{\delta b_n} dS + \int_{S^F} j_S^F \frac{\delta(dS)}{\delta b_n} \tag{3.16}$$

Hence, by considering eq. 3.16 and following a similar procedure as in sections 2.2.3– 2.2.4,

$$\frac{\delta J^F}{\delta b_n} = \int_{S^F} \left(\frac{\partial j_{S_k}^F}{\partial v_i} n_k + j_{S,i}^{F,v} \right) \frac{\delta v_i}{\delta b_n} dS + \int_{S^F} \left(\frac{\partial j_{S_i}^F}{\partial p} n_i + j_S^{F,p} \right) \frac{\delta p}{\delta b_n} dS$$

$$
\begin{aligned}
&+\int_{S^F}\frac{\partial j^F_{S_i}}{\partial \tau_{kj}}n_i\frac{\delta\tau_{kj}}{\delta b_n}dS+\int_{S^F}\left(\frac{\partial j^F_{S_i}}{\partial T^F}n_i+j^{F,T}_S\right)\frac{\delta T^F}{\delta b_n}dS\\
&+\int_{S^F}\frac{\partial j^F_S}{\partial Q^F}\frac{\delta Q^F}{\delta b_n}dS+\int_{S^F_{W_p}}j^{F,g}_{S_{i,k}}\frac{\delta x_k}{\delta b_n}n_i dS+\int_{S^F_{W_p}}j^F_{S_i}\frac{\delta n_i}{\delta b_n}dS\\
&+\int_{S^F_{W_p}}j^F_{S_i}n_i\frac{\delta(dS)}{\delta b_n}+\int_{S^F_{W_p}}j^F_S\frac{\delta(dS)}{\delta b_n}+\int_{\Omega^F}j^{F,v}_{\Omega,i}\frac{\delta v_i}{\delta b_n}d\Omega+\int_{\Omega^F}j^{F,p}_\Omega\frac{\delta p}{\delta b_n}d\Omega\\
&+\int_{\Omega^F}j^{F,\widetilde{\nu}}_\Omega\frac{\delta\widetilde{\nu}}{\delta b_n}d\Omega+\int_{\Omega^F}j^{F,T}_\Omega\frac{\delta T^F}{\delta b_n}d\Omega+\int_{\Omega^F}j^F_\Omega\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega
\end{aligned}
\tag{3.17}
$$

Similarly for J^S, its gradient w.r.t. b_n reads

$$
\begin{aligned}
\frac{\delta J^S}{\delta b_n}=&\int_{S^S}\left(\frac{\partial j^S_{S_i}}{\partial T^S}n_i+j^{S,T}_S\right)\frac{\delta T^S}{\delta b_n}dS+\int_{S^S}\frac{\partial j^S_S}{\partial Q^S}\frac{\delta Q^S}{\delta b_n}dS\\
&+\int_{S^S_{W_p}}j^{S,g}_{S_{i,k}}\frac{\delta x_k}{\delta b_n}n_i dS+\int_{S^S_{W_p}}j^S_{S_i}\frac{\delta n_i}{\delta b_n}dS+\int_{S^S_{W_p}}j^S_{S_i}n_i\frac{\delta(dS)}{\delta b_n}+\int_{S^S_{W_p}}j^S_S\frac{\delta(dS)}{\delta b_n}\\
&+\int_{\Omega^S}j^{S,T}_\Omega\frac{\delta T^S}{\delta b_n}d\Omega+\int_{\Omega^S}j^S_\Omega\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega+\int_{\Omega^S}j^{S,g}_\Omega\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega
\end{aligned}
\tag{3.18}
$$

In eq. 3.18, the last term accounts for dependencies of J^S on geometrical quantities. Such terms in J^F do not appear in any case in this thesis and are, thus, omitted. To clarify the notation used, the superscript in $j^S_{S_i}$ denotes that J is defined for the solid and the subscript that the objective is defined along (part of) the Ω^S boundary.

3.2.2 Differentiation of the R^{T^F} Integral

The integral on the r.h.s. of eq. 2.20 is developed as

$$
\begin{aligned}
\int_\Omega T^F_a\frac{\delta R^{T^F}}{\delta b_n}d\Omega=&\underbrace{\int_{\Omega^F}T^F_a\frac{\delta}{\delta b_n}\left(\rho^F c_p v_i\frac{\partial T^F}{\partial x_i}\right)d\Omega}_{A^{T^F}}+\underbrace{\int_{\Omega^F}T^F_a\frac{\delta}{\delta b_n}\left(\rho^F v_i v_j\frac{\partial v_j}{\partial x_i}\right)d\Omega}_{B^{T^F}}\\
&\underbrace{-\int_{\Omega^F}T^F_a\frac{\delta}{\delta b_n}\left[\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T^F}{\partial x_j}\right)\right]d\Omega}_{C^{T^F}}
\end{aligned}
\tag{3.19}
$$

Each of the three terms is developed as

$$
\begin{aligned}
A^{T^F} &= \int_{\Omega^F} \rho^F c_p T_a^F \frac{\partial T^F}{\partial x_i} \frac{\delta v_i}{\delta b_n} d\Omega + \int_{\Omega^F} \rho^F c_p T_a^F v_i \frac{\delta}{\delta b_n} \left(\frac{\partial T^F}{\partial x_i} \right) d\Omega \\
&= \int_{\Omega^F} \rho^F c_p T_a^F \frac{\partial T^F}{\partial x_i} \frac{\delta v_i}{\delta b_n} d\Omega + \int_{\Omega^F} \rho^F c_p T_a^F v_i \frac{\partial}{\partial x_i} \left(\frac{\delta T^F}{\delta b_n} \right) d\Omega \\
&- \int_{\Omega^F} \rho^F c_p T_a^F v_j \frac{\partial T^F}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&= \int_{\Omega^F} \rho^F c_p T_a^F \frac{\partial T^F}{\partial x_i} \frac{\delta v_i}{\delta b_n} d\Omega + \int_{S_F} \rho^F c_p T_a^F v_i n_i \frac{\delta T^F}{\delta b_n} dS \\
&- \int_{\Omega^F} \frac{\partial}{\partial x_i} \left(\rho^F c_p T_a^F v_i \right) \frac{\delta T^F}{\delta b_n} d\Omega - \int_{\Omega^F} \rho^F c_p T_a^F v_j \frac{\partial T^F}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{3.20a}
$$

$$
\begin{aligned}
B^{T^F} &= \int_{\Omega^F} \rho^F T_a^F v_j \frac{\partial v_j}{\partial x_i} \frac{\delta v_i}{\delta b_n} d\Omega + \int_{\Omega^F} \rho^F T_a^F v_i \frac{\partial v_j}{\partial x_i} \frac{\delta v_j}{\delta b_n} d\Omega + \int_{\Omega^F} \rho^F T_a^F v_i v_j \frac{\delta}{\delta b_n} \left(\frac{\partial v_j}{\partial x_i} \right) d\Omega \\
&= \int_{\Omega^F} \rho^F T_a^F v_j \frac{\partial v_j}{\partial x_i} \frac{\delta v_i}{\delta b_n} d\Omega + \int_{\Omega^F} \rho^F T_a^F v_i \frac{\partial v_j}{\partial x_i} \frac{\delta v_j}{\delta b_n} d\Omega + \int_{\Omega^F} \rho^F T_a^F v_i v_j \frac{\partial}{\partial x_i} \left(\frac{\delta v_j}{\delta b_n} \right) d\Omega \\
&- \int_{\Omega^F} \rho^F T_a^F v_i v_j \frac{\partial v_j}{\partial x_k} \frac{\partial}{\partial x_i} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&= \int_{\Omega^F} \rho^F T_a^F v_j \frac{\partial v_j}{\partial x_i} \frac{\delta v_i}{\delta b_n} d\Omega + \int_{\Omega^F} \rho^F T_a^F v_j \frac{\partial v_i}{\partial x_j} \frac{\delta v_i}{\delta b_n} d\Omega + \int_{S^F} \rho^F T_a^F v_j n_j v_i \frac{\delta v_i}{\delta b_n} dS \\
&- \int_{\Omega^F} \frac{\partial}{\partial x_j} \left(\rho^F T_a^F v_j v_i \right) \frac{\delta v_i}{\delta b_n} d\Omega - \int_{\Omega^F} \rho^F T_a^F v_j v_i \frac{\partial v_i}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{3.20b}
$$

$$
\begin{aligned}
C^{T^F} &= - \int_{\Omega^F} T_a^F \frac{\partial}{\partial x_j} \left[\frac{\delta}{\delta b_n} \left(k^F \frac{\partial T^F}{\partial x_j} \right) \right] d\Omega + \int_{\Omega^F} T_a^F \frac{\partial}{\partial x_k} \left(k^F \frac{\partial T^F}{\partial x_j} \right) \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&= - \int_{S^F} T_a^F n_j \frac{\delta}{\delta b_n} \left(k^F \frac{\partial T^F}{\partial x_j} \right) dS + \int_{\Omega^F} \rho^F c_p \frac{\partial T^F}{\partial x_j} \frac{\partial T_a^F}{\partial x_j} \frac{\delta \alpha_{eff}}{\delta b_n} d\Omega \\
&+ \int_{\Omega^F} k^F \frac{\partial T_a^F}{\partial x_j} \frac{\delta}{\delta b_n} \left(\frac{\partial T^F}{\partial x_j} \right) d\Omega + \int_{\Omega^F} T_a^F \frac{\partial}{\partial x_k} \left(k^F \frac{\partial T^F}{\partial x_j} \right) \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&= - \int_{S^F} T_a^F n_j \frac{\delta}{\delta b_n} \left(k^F \frac{\partial T^F}{\partial x_j} \right) dS + \int_{\Omega^F} \rho^F c_p \frac{\partial T^F}{\partial x_j} \frac{\partial T_a^F}{\partial x_j} \frac{\delta \alpha_{eff}}{\delta b_n} d\Omega \\
&+ \int_{\Omega^F} k^F \frac{\partial T_a^F}{\partial x_j} \frac{\partial}{\partial x_j} \left(\frac{\delta T^F}{\delta b_n} \right) d\Omega - \int_{\Omega^F} k^F \frac{\partial T_a^F}{\partial x_j} \frac{\partial T^F}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&+ \int_{\Omega^F} T_a^F \frac{\partial}{\partial x_k} \left(k^F \frac{\partial T^F}{\partial x_j} \right) \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{3.20c}
$$

Since $\alpha_{eff}=\nu_t/Pr_t$, then $\frac{\delta\alpha_{eff}}{\delta b_n}=\frac{1}{Pr_t}\frac{\delta\nu_t}{\delta\widetilde{\nu}}\frac{\delta\widetilde{\nu}}{\delta b_n}$ and eq. 3.20c becomes

$$\begin{aligned}C^{T^F}=&-\int_{S^F}T_a^F n_j\frac{\delta}{\delta b_n}\left(k^F\frac{\partial T^F}{\partial x_j}\right)dS+\int_{\Omega^F}\rho^F c_p\frac{\partial T^F}{\partial x_j}\frac{\partial T_a^F}{\partial x_j}\frac{1}{Pr_t}\frac{\delta\nu_t}{\delta\widetilde{\nu}}\frac{\delta\widetilde{\nu}}{\delta b_n}d\Omega\\&+\int_{S^F}k^F\frac{\partial T_a^F}{\partial n}\frac{\delta T^F}{\delta b_n}dS-\int_{\Omega^F}\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_a^F}{\partial x_j}\right)\frac{\delta T^F}{\delta b_n}d\Omega\\&+\int_{\Omega^F}\left[-k^F\frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_k}+T_a^F\frac{\partial}{\partial x_k}\left(k^F\frac{\partial T^F}{\partial x_j}\right)\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\end{aligned}\tag{3.21a}$$

Substituting eqs. 3.20a, 3.20b and 3.21a into eq. 3.19 leads to the following expression

$$\begin{aligned}\int_\Omega T_a^F\frac{\delta R^{T^F}}{\delta b_n}d\Omega=&\int_{\Omega^F}\rho^F c_p T_a^F\frac{\partial T^F}{\partial x_i}\frac{\delta v_i}{\delta b_n}d\Omega+\int_{S_F}\rho^F c_p T_a^F v_i n_i\frac{\delta T^F}{\delta b_n}dS\\&-\int_{\Omega^F}\frac{\partial}{\partial x_i}\left(\rho^F c_p T_a^F v_i\right)\frac{\delta T^F}{\delta b_n}d\Omega-\int_{\Omega^F}\rho^F c_p T_a^F v_j\frac{\partial T^F}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\&+\int_{\Omega^F}\rho^F T_a^F v_j\frac{\partial v_j}{\partial x_i}\frac{\delta v_i}{\delta b_n}d\Omega+\int_{\Omega^F}\rho^F T_a^F v_j\frac{\partial v_i}{\partial x_j}\frac{\delta v_i}{\delta b_n}d\Omega\\&+\int_{S^F}\rho^F T_a^F v_j n_j v_i\frac{\delta v_i}{\delta b_n}dS-\int_{\Omega^F}\frac{\partial}{\partial x_j}\left(\rho^F T_a^F v_j v_i\right)\frac{\delta v_i}{\delta b_n}d\Omega\\&-\int_{\Omega^F}\rho^F T_a^F v_j v_i\frac{\partial v_i}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega-\int_{S^F}T_a^F n_j\frac{\delta}{\delta b_n}\left(k^F\frac{\partial T^F}{\partial x_j}\right)dS\\&+\int_{\Omega^F}\rho^F c_p\frac{\partial T^F}{\partial x_j}\frac{\partial T_a^F}{\partial x_j}\frac{1}{Pr_t}\frac{\delta\nu_t}{\delta\widetilde{\nu}}\frac{\delta\widetilde{\nu}}{\delta b_n}d\Omega+\int_{S^F}k^F\frac{\partial T_a^F}{\partial n}\frac{\delta T^F}{\delta b_n}dS\\&+\int_{\Omega^F}\left[T_a^F\frac{\partial}{\partial x_k}\left(k^F\frac{\partial T^F}{\partial x_j}\right)-k^F\frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_k}\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\&-\int_{\Omega^F}\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_a^F}{\partial x_j}\right)\frac{\delta T^F}{\delta b_n}d\Omega\end{aligned}\tag{3.22}$$

3.2.3 Differentiation of the R^{T^S} Integral

The last term on the r.h.s. of eq. 3.14 is developed as

$$\begin{aligned}\frac{\delta}{\delta b_n}\int_{\Omega^S}T_a^S R^{T^S}d\Omega=&\frac{\delta}{\delta b_n}\int_{\Omega^S}T_a^S\left[-\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T^S}{\partial x_j}\right)\right]d\Omega\\=&-\int_{\Omega^S}T_a^S\frac{\partial}{\partial x_j}\left[\frac{\delta}{\delta b_n}\left(k^S\frac{\partial T^S}{\partial x_j}\right)\right]d\Omega\end{aligned}$$

$$
\begin{aligned}
&+\int_{\Omega^S} T_a^S \frac{\partial}{\partial x_k}\left(k^S \frac{\partial T^S}{\partial x_j}\right)\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\
&=-\int_{S^S} T_a^S n_j \frac{\delta}{\delta b_n}\left(k^S \frac{\partial T^S}{\partial x_j}\right)dS+\int_{\Omega^S} \frac{\partial T_a^S}{\partial x_j}\frac{\delta}{\delta b_n}\left(k^S \frac{\partial T^S}{\partial x_j}\right)d\Omega \\
&+\int_{\Omega^S} T_a^S \frac{\partial}{\partial x_k}\left(k^S \frac{\partial T^S}{\partial x_j}\right)\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\
&=-\int_{S^S} T_a^S n_j \frac{\delta}{\delta b_n}\left(k^S \frac{\partial T^S}{\partial x_j}\right)dS+\int_{\Omega^S} k^S \frac{\partial T_a^S}{\partial x_j}\frac{\partial}{\partial x_j}\left(\frac{\delta T^S}{\delta b_n}\right)d\Omega \\
&+\int_{\Omega^S} T_a^S \frac{\partial}{\partial x_k}\left(k^S \frac{\partial T^S}{\partial x_j}\right)\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\
&-\int_{\Omega^S} k^S \frac{\partial T^S}{\partial x_k}\frac{\partial T_a^S}{\partial x_j}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\
&=-\int_{S^S} T_a^S n_j \frac{\delta}{\delta b_n}\left(k^S \frac{\partial T^S}{\partial x_j}\right)dS+\int_{S^S} k^S \frac{\partial T_a^S}{\partial n}\frac{\delta T^S}{\delta b_n}dS \\
&+\int_{\Omega^S}\left[T_a^S \frac{\partial}{\partial x_k}\left(k^S \frac{\partial T^S}{\partial x_j}\right)-k^S \frac{\partial T^S}{\partial x_k}\frac{\partial T_a^S}{\partial x_j}\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\
&-\int_{\Omega^S} \frac{\partial}{\partial x_j}\left(k^S \frac{\partial T_a^S}{\partial x_j}\right)\frac{\delta T^S}{\delta b_n}d\Omega
\end{aligned}
\tag{3.23}
$$

3.2.4 Final Expression of $\delta L/\delta b_n$

By gathering together the terms developed in sections 3.2.1–3.2.3, as well as the derivatives of second, third and fourth terms on the r.h.s. of eq. 3.14, see Chapter 2, and after rearranging terms, the following expression arises

$$
\begin{aligned}
\frac{\delta L}{\delta b_n} = \frac{\delta L}{\delta b_n}\bigg|_{inco} &+\int_{\Omega^F}\left[\rho^F c_p T_a^F \frac{\partial T^F}{\partial x_i}+\rho^F T_a^F v_k \frac{\partial v_k}{\partial x_i}+\rho^F v_j T_a^F \frac{\partial v_i}{\partial x_j}-\frac{\partial}{\partial x_j}\left(\rho^F v_i v_j T_a^F\right)\right]\frac{\delta v_i}{\delta b_n}d\Omega \\
&+\int_{S^F}\rho^F T_a^F v_i v_j n_j \frac{\delta v_i}{\delta b_n}dS+\int_{\Omega^F}\left[-\frac{\partial}{\partial x_j}\left(\rho^F c_p v_j T_a^F\right)-\frac{\partial}{\partial x_j}\left(k^F \frac{\partial T_a^F}{\partial x_j}\right)+j_\Omega^{F,T}\right]\frac{\delta T^F}{\delta b_n}d\Omega \\
&\underbrace{+\int_{S^F}\left(\rho^F c_p T_a^F v_j n_j+k^F \frac{\partial T_a^F}{\partial n}+\frac{\partial j_{S_i}^F}{\partial T^F}n_i+j_S^{F,T}\right)\frac{\delta T^F}{\delta b_n}dS}_{I_1} \\
&\underbrace{+\int_{S^F}\left(-T_a^F n_j\right)\frac{\delta}{\delta b_n}\left(k^F \frac{\partial T^F}{\partial x_j}\right)dS}_{I_2}\underbrace{+\int_{S^S}\left(-T_a^S n_j\right)\frac{\delta}{\delta b_n}\left(k^S \frac{\partial T^S}{\partial x_j}\right)dS}_{I_3}
\end{aligned}
$$

$$
\begin{aligned}
&+\underbrace{\int_{S^S}\left(k^S\frac{\partial T_a^S}{\partial n}+\frac{\partial j_{S,i}^S}{\partial T^S}n_i+j_S^{S,T}\right)\frac{\delta T^S}{\delta b_n}dS}_{I_4}+\underbrace{\int_{S^F}\frac{\partial j_S^F}{\partial Q^F}\frac{\delta Q^F}{\delta b_n}dS}_{I_5}\\
&+\underbrace{\int_{S^S}\frac{\partial j_S^S}{\partial Q^S}\frac{\delta Q^S}{\delta b_n}dS}_{I_6}+\int_{\Omega^F}\rho^F c_p\frac{\partial T^F}{\partial x_j}\frac{\partial T_a^F}{\partial x_j}\frac{1}{Pr_t}\frac{\delta\nu_t}{\delta\widetilde{\nu}}\frac{\delta\widetilde{\nu}}{\delta b_n}d\Omega\\
&+\int_{\Omega^F}\Bigg[-\rho^F c_p T_a^F v_j\frac{\partial T^F}{\partial x_k}-\rho^F T_a^F v_i v_j T_a^F\frac{\partial v_i}{\partial x_k}-k^F\frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_k}\\
&+T_a^F\frac{\partial}{\partial x_k}\left(k^F\frac{\partial T^F}{\partial x_j}\right)\Bigg]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega-\int_{\Omega^S}\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_a^S}{\partial x_j}\right)\frac{\delta T^S}{\delta b_n}d\Omega\\
&+\int_{\Omega^S}\left[-k^S\frac{\partial T_a^S}{\partial x_j}\frac{\partial T^S}{\partial x_k}+T_a^S\frac{\partial}{\partial x_k}\left(k^S\frac{\partial T^S}{\partial x_j}\right)\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\
&+\int_{S_{W_p}^S}j_{S_{i,k}}^{S,g}\frac{\delta x_k}{\delta b_n}n_i dS+\int_{S_{W_p}^S}j_{S_i}^S\frac{\delta(n_i dS)}{\delta b_n}+\int_{\Omega^S}\left(j_\Omega^S+j_\Omega^{S,g}\right)\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\
&+\int_{S_{W_p}^F}j_S^F\frac{\delta(dS)}{\delta b_n}+\int_{S_{W_p}^S}j_S^S\frac{\delta(dS)}{\delta b_n}
\end{aligned}
\tag{3.24}
$$

where $\frac{\delta L}{\delta b_n}\big|_{inco}$ is given by eq. 2.46.

3.2.5 Field Adjoint Equations

In CHT problems, the Field Adjoint Equations arise by setting the expressions multiplying the variations of the primal variables, inside field integrals, to zero. By doing so, the adjoint momentum and turbulence model PDEs derived for pure fluid flows, see Chapter 2, receive additional contributions from the differentiated energy equation over Ω^F. The resulting adjoint CHT PDEs are

$$R^q=-\frac{\partial u_j}{\partial x_j}+j_\Omega^{F,p}=0 \tag{3.25a}$$

$$
\begin{aligned}
R_i^u=&\,u_j\frac{\partial v_j}{\partial x_i}-\frac{\partial\left(v_j u_i\right)}{\partial x_j}-\frac{\partial\tau_{ij}^a}{\partial x_j}+\frac{\partial q}{\partial x_i}+\widetilde{\nu}_a\frac{\partial\widetilde{\nu}}{\partial x_i}-\frac{\partial}{\partial x_l}\left(\widetilde{\nu}_a\widetilde{\nu}\frac{\mathcal{C}_Y}{Y}e_{mjk}\frac{\partial v_k}{\partial x_j}e_{mli}\right)\\
&+\rho_F c_p T_a^F\frac{\partial T^F}{\partial x_i}+\rho_F T_a^F v_k\frac{\partial v_k}{\partial x_i}-\rho_F v_i v_k\frac{\partial T_a^F}{\partial x_k}+j_{\Omega,i}^{F,v}=0,\quad i=1,2(,3)
\end{aligned}
\tag{3.25b}
$$

$$R_F^{T_a}=-\frac{\partial(v_j T_a^F)}{\partial x_j}-\frac{\partial}{\partial x_j}\left(\alpha_{eff}\frac{\partial T_a^F}{\partial x_j}\right)+j_\Omega^{F,T}=0 \tag{3.25c}$$

$$R^{\widetilde{\nu_a}}=-v_j\frac{\partial\widetilde{\nu}_a}{\partial x_j}-\frac{\partial}{\partial x_j}\left[\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\partial\widetilde{\nu}_a}{\partial x_j}\right]+\frac{1}{\sigma}\frac{\partial\widetilde{\nu}_a}{\partial x_j}\frac{\partial\widetilde{\nu}}{\partial x_j}+2\frac{C_{b2}}{\sigma}\frac{\partial}{\partial x_j}\left(\widetilde{\nu}_a\frac{\partial\widetilde{\nu}}{\partial x_j}\right)$$

$$+\widetilde{\nu}_a\widetilde{\nu}C_{\widetilde{\nu}}\left(\widetilde{\nu},v_i\right)-\left(\mathcal{P}-\mathcal{D}\right)\widetilde{\nu}_a+\frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)\frac{\delta\nu_t}{\delta\widetilde{\nu}}$$

$$+\rho_F\frac{c_p}{Pr_t}\frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_j}\frac{\delta\nu_t}{\delta\widetilde{\nu}}+j_\Omega^{F,\widetilde{\nu}}=0 \tag{3.25d}$$

$$R^{\Delta^a}=-2\frac{\partial}{\partial x_j}\left(\Delta^a\frac{\partial\Delta}{\partial x_j}\right)+\widetilde{\nu}\widetilde{\nu}_aC_\Delta=0 \tag{3.25e}$$

$$R^{T_a^S}=-\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_a^S}{\partial x_j}\right)+j_\Omega^{S,T}=0 \tag{3.25f}$$

Eqs. 3.25a–3.4 are solved in Ω^F and eq. 3.25f in Ω^S. In the above equations, eq. 3.25c is the adjoint energy PDE in the fluid domain and eq. 3.25f the adjoint heat conduction equations of the solid. Since both of these equations do not involve any adjoint variable other than T_a^F and T_a^S, they can be solved prior to solving the rest of the adjoint PDEs. Finally, the seventh, eighth and ninth terms on the r.h.s. of eq. 3.25b and the eighth term on the r.h.s. of eq. 3.25d are contributions from the differentiated fluid energy PDE. The rest of the terms on the r.h.s. of eqs. 3.25b and 3.25d form the adjoint momentum eq. 2.48b and turbulence model eq. 2.48c PDEs in pure fluid flow problems.

3.2.6 Adjoint Boundary Conditions

In this section, the adjoint boundary conditions for CHT problems are derived in detail. Distinction is made between parameterized walls (denoted by subscript p), which stand for walls controlled by the design variables and non-parameterized ones (denoted by np).

3.2.6.1 Extension of Pure-Flow Boundary Conditions to CHT Problems

Differences in the boundary conditions regarding u_i, q and $\widetilde{\nu}_a$ between CHT and pure fluid flows are discussed. As in the adjoint momentum and turbulence model PDEs, some of the boundary conditions of the aforementioned variables receive contributions from the differentiated fluid energy equation. Along S_I^F and S_W^F, since Dirichlet conditions are imposed on v_i and $\widetilde{\nu}$, $\delta v_i/\delta b_n=0$ and $\delta\widetilde{\nu}/\delta b_n=0$, eliminating the second and sixth integrals on the r.h.s. of eq. 3.24. As a result, along S_I^F and S_W^F, the boundary conditions for pure fluid flow problems, see section 2.2.12, remain the same in CHT problems.

On the other hand, differences occur in the boundary conditions along S_O^F, regarding u_i and q. Since $\partial v_i/\partial n=0$, $\delta v_i/\delta b_n$ is not zero and to eliminate the second integral on the r.h.s. of eq. 3.24, its integrand has to be considered in the

boundary conditions for u_i and q, which now become

$$q = u_{\langle n\rangle} v_{\langle n\rangle} + 2\left(\nu+\nu_t\right)\frac{\partial u_{\langle n\rangle}}{\partial n} + \widetilde{\nu}_a\widetilde{\nu}\frac{C_Y}{Y}e_{jql}\frac{\partial v_l}{\partial x_q}e_{jki}n_k n_i + \frac{\partial j^F_{S_{O,k}}}{\partial v_i}n_k n_i$$
$$+ j^{F,v}_{S_{O,i}} n_i + \rho^F T^F_a v^2_{\langle n\rangle} = 0 \tag{3.26}$$

Also, $\partial u_n/\partial n = 0$ and the tangential adjoint velocity components are given by

$$v_{\langle n\rangle} u^l_{\langle t\rangle} + \left(\nu+\nu_t\right)\left(\frac{\partial u^l_{\langle t\rangle}}{\partial n} + \frac{\partial u_{\langle n\rangle}}{\partial t^l}\right) + \widetilde{\nu}_a\widetilde{\nu}\frac{C_Y}{Y}e_{jql}\frac{\partial v_l}{\partial x_q}e_{jki}n_k t^l_i + \frac{\partial j^F_{S_{O,k}}}{\partial v_i}n_k t^l_i$$
$$+ j^{F,v}_{S_{O,i}} t^l_i + \rho^F T^F_a v_{\langle n\rangle} v^l_{\langle t\rangle} = 0 \tag{3.27}$$

Finally, since $\partial T^F / \partial n = 0$ along S^F_O, the sixth integral vanishes and the boundary condition for $\widetilde{\nu}_a$ is the same as in pure fluid flow problems, see eq. 2.52.

3.2.6.2 Derivation of Boundary Conditions for T^F_a and T^S_a along the FSI

Since eqs. 3.9 and 3.10, then

$$\left.\frac{\delta T^S}{\delta b_n}\right|_{\overline{S}^F} = \left.\frac{\delta T^F}{\delta b_n}\right|_{\overline{S}^S} \tag{3.28a}$$

$$\left.\frac{\delta}{\delta b_n}\left(k^S\frac{\partial T^S}{\partial n}\right)\right|_{\overline{S}^S} = -\left.\frac{\delta}{\delta b_n}\left(k^F\frac{\partial T^F}{\partial n}\right)\right|_{\overline{S}^F} \tag{3.28b}$$

To derive the adjoint FSI conditions for T^F_a and T^S_a, integrals I_1, I_3, I_4 and I_6, written along $\overline{S}^F$ and $\overline{S}^S$ are used. Also, one has to consider that $\mathbf{n}$ is the outward unit normal vector and, thus, $n_i\big|_{\overline{S}^F} = -n_i\big|_{\overline{S}^S}$ By using eqs. 3.28a and 3.28b and since $v_i = 0$ along the FSI, the following expressions are derived

$$\begin{aligned}
I_1+I_2+I_3+I_4+I_5+I_6 &= \int_{\overline{S}^F}\left(k^F\frac{\partial T^F_a}{\partial n} + \frac{\partial j^F_{S_i}}{\partial T^F}n_i + j^{F,T}_S\right)\frac{\delta T^F}{\delta b_n}dS + \int_{\overline{S}^F}\left(\frac{\partial j^F_S}{\partial Q^F} - T^F_a\right)\frac{\delta Q^F}{\delta b_n}dS \\
&+ \int_{\overline{S}^F} T^F_a k^F\frac{\partial T^F}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS + \int_{\overline{S}^S}\left(\frac{\partial j^S_S}{\partial Q^S} - T^S_a\right)\frac{\delta Q^S}{\delta b_n}dS \\
&+ \int_{\overline{S}^S} T^S_a k^S\frac{\partial T^S}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS + \int_{\overline{S}^S}\left(k^S\frac{\partial T^S_a}{\partial n} + \frac{\partial j^S_{S,i}}{\partial T^S}n_i + j^{S,T}_S\right)\frac{\delta T^S}{\delta b_n}dS \\
&= \int_{\overline{S}^F}\left(k^F\frac{\partial T^F_a}{\partial n} + \frac{\partial j^F_{S_i}}{\partial T^F}n_i + j^{F,T}_S + k^S\frac{\partial T^S_a}{\partial n^S} + \frac{\partial j^S_{S,i}}{\partial T^S}n^S_i + j^{S,T}_S\right)\frac{\delta T^F}{\delta b_n}dS
\end{aligned}$$

$$+\int_{\overline{S}^F}\left(\frac{\partial j_F^S}{\partial Q^F}-T_a^F-\frac{\partial j_S^S}{\partial Q^S}+T_a^S\right)\frac{\delta Q^F}{\delta b_n}dS+\int_{\overline{S}^F}T_a^F k^F\frac{\partial T^F}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS$$
$$+\int_{\overline{S}^S}T_a^S k^S\frac{\partial T^S}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS \tag{3.29}$$

By eliminating the first two integrals on the r.h.s. of the above equation, the following boundary conditions arise

$$\frac{\partial j_F^S}{\partial Q^F}-T_a^F\big|_{\overline{S}^F}=\frac{\partial j_S^S}{\partial Q^S}-T_a^S\big|_{\overline{S}^S} \tag{3.30a}$$

$$\left(k^F\frac{\partial T_a^F}{\partial n}+\frac{\partial j_{S_i}^F}{\partial T^F}n_i+j_S^{F,T}\right)\Bigg|_{\overline{S}^F}=-\left(k^S\frac{\partial T_a^S}{\partial n}+\frac{\partial j_{S,i}^S}{\partial T^S}n_i+j_S^{S,T}\right)\Bigg|_{\overline{S}^S} \tag{3.30b}$$

and the last two integrals contribute to the SDs.

3.2.6.3 Derivation of Boundary Conditions for T_a^F along $S^{F,*}$

The detailed derivation of the adjoint boundary conditions for the non-FSI boundaries, $S^{F,*}$, of Ω^F, see fig. 3.1b, follows. To do so, I_1, I_2 and I_5 integrals on the r.h.s. of eq. 3.24 are used.

Along S_I^F, T^F takes a fixed value; thus $\frac{\delta T^F}{\delta b_n}=0$ and, consequently, I_1 vanishes. To eliminate I_2+I_5, $\frac{\partial j_S^F}{\partial Q^F}=T_a^F$. Along S_O^F, $\partial T^F/\partial n=0$ and thus, $\frac{\delta}{\delta b_n}\left(\frac{\partial T^F}{\partial n}\right)=0$. Since S_O^F is not parameterized, $\frac{\delta n_i}{\delta b_n}=0$. As a result, $I_2=I_5=0$. To eliminate I_1, the following boundary condition is used for T_a^F along S_O^F

$$\rho^F c_p T_a^F v_j n_j+k^F\frac{\partial T_a^F}{\partial n}+\frac{\partial j_{S_i}^F}{\partial T^F}n_i+j_S^{F,T}=0 \tag{3.31}$$

To derive the boundary conditions along $S^{F,*}$, I_2+I_5 is written as

$$I_2+I_5=\int_{S_W^{F,*}}\left(\frac{\partial j_S^F}{\partial Q^F}-T_a^F\right)\frac{\delta}{\delta b_n}\left(k^F\frac{\partial T^F}{\partial n}\right)dS+\int_{S_{W_p}^{F,*}}T_a^F k^F\frac{\partial T^F}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS=0 \tag{3.32}$$

Along $S_{W,D}^F$, since $T^F=ct$, $\frac{\delta T^F}{\delta b_n}=0$ and I_1 is zeroed. To avoid computing the first integral on the r.h.s. of eq. 3.32, written for $S_{W,D}^F$, $\frac{\partial j_S^F}{\partial Q^F}=T_a^F$ and the second integral is a contribution to the SDs.

Along, $S_{W,Fl}^F$, defining an objective function depending on Q^F makes no sense, thus $\partial j^F/\partial Q^F$ is neglected. Also, $\frac{\delta}{\delta b_n}\left(k^F\frac{\partial T^F}{\partial n}\right)=0$ and the first integral on the

r.h.s. of eq. 3.32, written for $S^F_{W,Fl}$, is zeroed; the second is a contribution to the SDs. To eliminate I_1, since $v_i=0$ along $S^F_{W,Fl}$,

$$k^F\frac{\partial T^F_a}{\partial n}+\frac{\partial j^F_{S_i}}{\partial T^F}n_i+j^{F,T}_S=0 \tag{3.33}$$

The same condition is imposed on adiabatic walls, being a special case of $S^F_{W,Fl}$.

Along S^F_S, J^F is not defined, $\frac{\delta n_i}{\delta b_n}=0$ and $\frac{\partial T^F}{\partial n}=0$ $\frac{\delta}{\delta b_n}\left(\frac{\partial T^F}{\partial n}\right)=0$. Thus, $I_2=I_5!=0$. To eliminate I_1, since $v_{\langle n\rangle}=0$, then

$$\left.\frac{\partial T^F_a}{\partial n}\right|_{S^F_S}=0 \tag{3.34}$$

is used as the boundary condition for T^F_a along S^F_S.

3.2.6.4 Derivation of Boundary Conditions for T^S_a along $S^{S,*}$

The adjoint boundary conditions for the non-FSI boundaries $S^{S,*}$ of Ω^S, see fig. 3.1c, are derived based on the I_3, I_4 and I_6 integrals on the r.h.s. of eq. 3.24. The sum of I_3 and I_6 is written as

$$I_3+I_6=\int_{S^S_W}\left(\frac{\partial j^S_S}{\partial Q^S}-T^S_a\right)\frac{\delta}{\delta b_n}\left(k^F\frac{\partial T^S}{\partial n}\right)dS+\int_{S^S_{W_p}}T^S_a k^S\frac{\partial T^S}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS=0 \tag{3.35}$$

Along $S^S_{W,D}$, since T^S takes on a fixed value, $\frac{\delta T^S}{\delta b_n}=0$. As a result, I_4 along $S^S_{W,D}$ is zeroed. In eq. 3.35, written for $S^S_{W,D}$, the first integral on the r.h.s. is zeroed by setting $\frac{\partial j^S_S}{\partial Q^S}=T^S_a$. Also, the second integral is a contribution to the SDs.

For the same reason as for $S^F_{W,Fl}$, also $\partial j^S/\partial Q^S$ is neglected. The first integral of the r.h.s. of eq. 3.35, written along $S^S_{W,Fl}$, is zeroed since $\frac{\delta}{\delta b_n}\left(k^F\frac{\partial T^S}{\partial n}\right)=0$; the second is a contribution to the SDs. To eliminate I_4,

$$k^S\frac{\partial T^S_a}{\partial n}+\frac{\partial j^S_{S,i}}{\partial T^S}n_i+j^{S,T}_S=0 \tag{3.36}$$

Regarding S^S_{CH}, by differentiating eq. 3.8 w.r.t. b_n, the following equation arises

$$-k^S\frac{\delta}{\delta b_n}\left(\frac{\partial T^S}{\partial n}\right)=h\frac{\delta T^S}{\delta b_n} \tag{3.37}$$

and the sum of the I_3, I_4 and I_6 integrals can be written as

$$I_3+I_4+I_6=\int_{S^S_{CH}}\left(k^S\frac{\partial T^S_a}{\partial n}+\frac{\partial j^S_{S,i}}{\partial T^S}n_i+j^{S,T}_S+hT^S_a-\frac{\partial j^S_S}{\partial Q^S}\right)\frac{\delta T^S}{\delta b_n}dS$$
$$+\int_{S^S_{CH,p}}T^S_a k^S\frac{\partial T^S}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS \tag{3.38}$$

The elimination of the first integral on the r.h.s. of eq. 3.38 yields

$$k^S\frac{\partial T^S_a}{\partial n}+\frac{\partial j^S_{S,i}}{\partial T^S}n_i+j^{S,T}_S+h\left(T^S_a-\frac{\partial j^S_S}{\partial Q^S}\right)=0 \tag{3.39}$$

and the last integral is a contribution to the SDs.

Along S^S_S, the objective function is not defined and since $\frac{\delta n_i}{\delta b_n}=0$ and $\frac{\partial T^S}{\partial n}=0$. $\frac{\delta}{\delta b_n}\left(\frac{\partial T^S}{\partial n}\right)=0$. Thus, $I_3=I_6=0$. To eliminate I_4, the following condition

$$\left.\frac{\partial T^S_a}{\partial n}\right|_{S^S_S}=0 \tag{3.40}$$

must be imposed.

3.2.6.5 Contributions to the SDs

During the derivation of the adjoint boundary conditions, contributions to the SDs arise; these are gathered together as

$$C_{BCs}=\int_{\bar{S}^F_p\cup S^F_{W,D,p}\cup S^F_{W,Fl,p}}T^F_a k^F\frac{\partial T^F}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS+\int_{\bar{S}^S_p\cup S^S_{W,D,p}\cup S^S_{W,Fl,p}\cup S^S_{CH,p}}T^S_a k^S\frac{\partial T^S}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS \tag{3.41}$$

3.2.7 *FI* Adjoint Sensitivities

After satisfying the adjoint mean-flow PDEs and boundary conditions and considering eqs. 3.24, 2.46 and 3.41, the SDs take the form

$$\left.\frac{\delta J}{\delta b_n}\right|_{FI}=\int_{\Omega^F}\Big[-u_iv_j\frac{\partial v_i}{\partial x_k}-u_j\frac{\partial p}{\partial x_k}-\tau^a_{ij}\frac{\partial v_i}{\partial x_k}+u_i\frac{\partial \tau_{ij}}{\partial x_k}+q\frac{\partial v_j}{\partial x_k}+\Theta_{jk}-\rho^F c_p T^F_a v_j\frac{\partial T^F}{\partial x_k}$$

$$
\begin{aligned}
&-\rho^F T_a^F v_i v_j T_a^F \frac{\partial v_i}{\partial x_k} - k^F \frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_k} + T_a^F \frac{\partial}{\partial x_k}\left(k^F \frac{\partial T^F}{\partial x_j}\right)\Bigg]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\
&+\int_{\Omega^S}\left[-k^S \frac{\partial T_a^S}{\partial x_j}\frac{\partial T^S}{\partial x_k} + T_a^S \frac{\partial}{\partial x_k}\left(k^S \frac{\partial T^S}{\partial x_j}\right)\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\
&-\int_{S^F_{W_p}}\left[\frac{\partial j^F_{S_{W_{p,k}}}}{\partial \tau_{lm}} n_k t_l^I t_m^I \tau_{ij} \frac{\delta(t_i^I t_j^I)}{\delta b_n}\right] dS - \int_{S^F_{W_p}}\left[\frac{\partial j^F_{S_{W_{p,k}}}}{\partial \tau_{lm}} n_k t_l^{II} t_m^{II} \tau_{ij} \frac{\delta(t_i^{II} t_j^{II})}{\delta b_n}\right] dS \\
&-\int_{S^F_{W_p}}\left[\left(-u_{\langle n\rangle} + \frac{\partial j^F_{S_{W_{p,k}}}}{\partial \tau_{lm}} n_k n_l n_m\right)\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n}\right] dS \\
&-\int_{S^F_{W_p}}\left[\left(\frac{\partial j^F_{S_{W_{p,k}}}}{\partial \tau_{lm}} n_k (t_l^{II} t_m^I + t_l^I t_m^{II})\right)\tau_{ij}\frac{\delta(t_i^{II} t_j^I)}{\delta b_n}\right] dS + \int_{\Omega^S} j_\Omega^{S,g}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\
&+\sum_{D=F,S}\left[\int_{S^D_{W_p}} j^{D,g}_{S_{i,k}}\frac{\delta x_k}{\delta b_n} n_i dS + \int_{S^D_{W_p}} j^D_{S_i}\frac{\delta(n_i dS)}{\delta b_n} + \int_{S^D_{W_p}} j^D_S \frac{\delta(dS)}{\delta b_n} + \int_{\Omega^D} j^D_\Omega \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega\right] \\
&+\int_{\overline{S}^F_p \cup S^F_{W,D,p} \cup S^F_{W,Fl,p}} k^F T_a^F \frac{\partial T^F}{\partial x_j}\frac{\delta n_j}{\delta b_n} dS + \int_{\overline{S}^S_p \cup S^S_{W,D,p} \cup S^S_{W,Fl_p} \cup S^S_{CHp}} k^S T_a^S \frac{\partial T^S}{\partial x_j}\frac{\delta n_j}{\delta b_n} dS
\end{aligned} \tag{3.42}
$$

As expected also for CHT problems, in the *FI* adjoint, field integrals in both Ω^F and Ω^S on the r.h.s. of eq. 3.42 arise, including the grid sensitivities. As explained in section 2.2.13, grid sensitivities are computed through the differentiation w.r.t. b_n of the used GDM equations or FDs, at a cost that scales with N.

3.3 The *E-SI* and Severed *SI* Adjoint Methods

To develop the *E-SI* adjoint for CHT problems [49], the Lagrangian expression given by eq. 3.13 is extended by field integrals including the Laplace GDM residuals, for both Ω^F and Ω^S. In this way, L is redefined as

$$
\begin{aligned}
L = J &+ \int_{\Omega^F} q R^p d\Omega + \int_{\Omega^F} u_i R_i^v d\Omega + \int_{\Omega^F} \widetilde{\nu}_a R^\nu d\Omega + \int_{\Omega^F} \Delta^a R^\Delta d\Omega + \sum_{D=F,S}\Bigg[\int_{\Omega_D} T_a^D R_D^T d\Omega \\
&+ \int_{\Omega^D} m_i^{D,a} R_i^{D,m} d\Omega\Bigg]
\end{aligned} \tag{3.43}
$$

Following an approach similar to Chapter 2, the five first terms on the r.h.s. of eq. 3.43 are developed as in section 2.3, leading to eq. 3.24. Also, the last term is

developed as in section 2.3, for both Ω^F and Ω^S. As a result,

$$
\begin{aligned}
\frac{\delta L}{\delta b_n} = &\int_{S^F}\left(u_i v_j n_j + \tau^a_{ij} - q n_i + \widetilde{\nu_a}\widetilde{\nu}\frac{C_Y}{Y} e_{jql}\frac{\partial v_l}{\partial x_q} e_{jki} n_k + \rho^F T^F_a v_i v_j n_j + \frac{\partial j_{S_k}}{\partial v_i} n_k + j^v_{S,i}\right)\frac{\delta v_i}{\delta b_n} dS \\
&+ \int_{S^F}\left(u_i n_i + \frac{\partial j_{S_i}}{\partial p} n_i + j^p_S\right)\frac{\delta p}{\delta b_n} dS + \int_S \left(-u_i n_j + \frac{\partial j_{S_k}}{\partial \tau_{ij}} n_k\right)\frac{\delta \tau_{ij}}{\delta b_n} dS \\
&+ \int_{S^F}\left[\widetilde{\nu_a} v_j n_j - \frac{\widetilde{\nu_a}}{\sigma}\frac{\partial \widetilde{\nu}}{\partial n} + \frac{\partial \widetilde{\nu_a}}{\partial n}\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right) - 2\widetilde{\nu_a}\frac{c_{b2}}{\sigma}\frac{\partial \widetilde{\nu}}{\partial n} + j^{\widetilde{\nu}}_S\right]\frac{\delta \widetilde{\nu}}{\delta b_n} dS \\
&- \int_{S^F}\widetilde{\nu_a} n_j\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right)\frac{\delta}{\delta b_n}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right) dS + \int_{S^F}\left(\Delta^a \frac{\partial \Delta}{\partial n} + \Delta^a c_j n_j\right)\frac{\delta \Delta}{\delta b_n} dS \\
&+ \sum_{D=F,S}\left[\int_{S^D} m^{D,a}_i n_j \frac{\delta}{\delta b_n}\left(\frac{\partial m_i}{\partial x_j}\right) dS - \int_{S^D}\frac{\partial m^{D,a}_i}{\partial n}\frac{\delta x_i}{\delta b_n} dS + \int_{S^D_{W_p}} j^D_{S,i}\frac{\delta\left(n_i dS\right)}{\delta b_n}\right] \\
&+ \int_{\Omega^F} R^u_i \frac{\delta v_i}{\delta b_n} d\Omega + \int_{\Omega^F} R^q \frac{\delta p}{\delta b_n} d\Omega + \int_{\Omega^F} R^{\widetilde{\nu_a}}\frac{\delta \widetilde{\nu}}{\delta b_n} d\Omega + \int_{\Omega^F} R^{\Delta^a}\frac{\delta \Delta}{\delta b_n} d\Omega \\
&+ \sum_{D=F,S}\left[\int_{\Omega^D} R^{T^D_a}\frac{\delta T^D}{\delta b_n} d\Omega\right] + \int_{S^F_{W_P}}\left\{\left[- u_j\frac{\partial p}{\partial x_k} - \tau^a_{ij}\frac{\partial v_i}{\partial x_k} + u_i\frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right.\right. \\
&\left.\left. - k^F\frac{\partial T^F_a}{\partial x_j}\frac{\partial T^F}{\partial x_k} + T^F_a\frac{\partial}{\partial x_k}\left(k^F\frac{\partial T^F}{\partial x_j}\right) + j^{F,g}_{S_{j,k}}\right] n_j + j^F_\Omega n_k\right\}\frac{\delta x_k}{\delta b_n} dS \\
&+ \int_{\Omega^F}\left\{\frac{\partial^2 m^{F,a}_k}{\partial {x_j}^2} - \frac{\partial}{\partial x_j}\left[- u_i v_j\frac{\partial v_i}{\partial x_k} - u_j\frac{\partial p}{\partial x_k} - \tau^a_{ij}\frac{\partial v_i}{\partial x_k} + u_i\frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right.\right. \\
&\left.\left. - \rho^F c_p T^F_a v_j\frac{\partial T^F}{\partial x_k} - \rho^F T^F_a v_i v_j T^F_a\frac{\partial v_i}{\partial x_k} - k^F\frac{\partial T^F_a}{\partial x_j}\frac{\partial T^F}{\partial x_k} + T^F_a\frac{\partial}{\partial x_k}\left(k^F\frac{\partial T^F}{\partial x_j}\right)\right]\right. \\
&\left. - \frac{\partial j^F_\Omega}{\partial x_k}\right\}\frac{\delta x_k}{\delta b_n} d\Omega + \int_{S^F}\left(\rho^F c_p T^F_a v_j n_j + k^F\frac{\partial T^F_a}{\partial n} + \frac{\partial j^F_{S_i}}{\partial T^F} n_i + j^{F,T}_S\right)\frac{\delta T^F}{\delta b_n} dS \\
&- \int_{S^{F,*}} k^F T^F_a n_j\frac{\delta}{\delta b_n}\left(\frac{\partial T^F}{\partial x_j}\right) dS - \int_{S^{F,*}}\rho^F c_p T^F_a\frac{\partial T^F}{\partial n}\frac{1}{Pr_t}\frac{\delta \nu_t}{\delta \widetilde{\nu}}\frac{\delta \widetilde{\nu}}{\delta b_n} dS \\
&- \int_{\overline{S}^F} T^F_a n_j\frac{\delta}{\delta b_n}\left(k^F\frac{\partial T^F}{\partial x_j}\right) dS - \int_{S^S} T^S_a n_j\frac{\delta}{\delta b_n}\left(k^S\frac{\partial T^S}{\partial x_j}\right) dS \\
&+ \int_{S^S}\left(k^S\frac{\partial T^S_a}{\partial n} + \frac{\partial j^S_{S,i}}{\partial T^S} n_i + j^{S,T}_S\right)\frac{\delta T^S}{\delta b_n} dS - \int_{\Omega^S}\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T^S_a}{\partial x_j}\right)\frac{\delta T^S}{\delta b_n} d\Omega \\
&+ \int_{S^S_{W_p}}\left\{\left[-k^S\frac{\partial T^S_a}{\partial x_j}\frac{\partial T^S}{\partial x_k} + T^S_a\frac{\partial}{\partial x_k}\left(k^S\frac{\partial T^S}{\partial x_j}\right) + j^{S,g}_{S_{i,k}}\right] n_j + \left(j^S_\Omega + j^{S,g}_\Omega\right) n_k\right\}\frac{\delta x_k}{\delta b_n} dS \\
&- \int_{\Omega^S}\left\{\frac{\partial^2 m^{S,a}_k}{\partial {x_j}^2} - \frac{\partial}{\partial x_j}\left[-k^S\frac{\partial T^S_a}{\partial x_j}\frac{\partial T^S}{\partial x_k} + T^S_a\frac{\partial}{\partial x_k}\left(k^S\frac{\partial T^S}{\partial x_j}\right)\right]\right.
\end{aligned}
$$

$$-\frac{\partial\left(j_{\Omega}^{S}+j_{\Omega}^{S,g}\right)}{\partial x_k}\Bigg\}\frac{\delta x_k}{\delta b_n}d\Omega \tag{3.44}$$

By comparing eqs. 3.24 and eqs. 3.44, it can be observed that expressions multiplying the derivatives of p, v_i, $\widetilde{\nu}$, Δ, T^F and T^S w.r.t. b_n inside field and surface integrals are the same. Thus, the same adjoint PDEs and boundary conditions arise as in the *FI* adjoint. In addition, by zeroing the multipliers of $\delta x_k/\delta b_n$ inside field integrals over Ω^F and Ω^S, the adjoint grid displacement PDEs

$$\begin{aligned}\frac{\partial^2 m_k^{F,a}}{\partial {x_j}^2}=\frac{\partial}{\partial x_j}\Bigg[&-u_i v_j\frac{\partial v_i}{\partial x_k}-u_j\frac{\partial p}{\partial x_k}-\tau_{ij}^a\frac{\partial v_i}{\partial x_k}+u_i\frac{\partial \tau_{ij}}{\partial x_k}+q\frac{\partial v_j}{\partial x_k}+\Theta_{jk}-\rho^F c_p T_a^F v_j\frac{\partial T^F}{\partial x_k}\\&-\rho^F T_a^F v_i v_j T_a^F\frac{\partial v_i}{\partial x_k}-k^F\frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_k}+T_a^F\frac{\partial}{\partial x_k}\left(k^F\frac{\partial T^F}{\partial x_j}\right)\Bigg]-\frac{\partial j_{\Omega}^F}{\partial x_k}\end{aligned} \tag{3.45a}$$

$$\frac{\partial^2 m_k^{S,a}}{\partial {x_j}^2}=\frac{\partial}{\partial x_j}\left[-k^S\frac{\partial T_a^S}{\partial x_j}\frac{\partial T^S}{\partial x_k}+T_a^S\frac{\partial}{\partial x_k}\left(k^S\frac{\partial T^S}{\partial x_j}\right)\right]-\frac{\partial\left(j_{\Omega}^S+j_{\Omega}^{S,g}\right)}{\partial x_k} \tag{3.45b}$$

arise. To eliminate the seventh term on the r.h.s. of eq. 3.44, $m_i^{F,a}=m_i^{S,a}=0$ along the boundaries of Ω^F and Ω^S. The remaining terms in eq. 3.44 lead to the *E-SI* adjoint SDs

$$\begin{aligned}\left.\frac{\delta J}{\delta b_n}\right|_{E-SI}=&\int_{S_{W_P}^F}\Bigg\{\Bigg[-u_j\frac{\partial p}{\partial x_k}-\tau_{ij}^a\frac{\partial v_i}{\partial x_k}+u_i\frac{\partial \tau_{ij}}{\partial x_k}+q\frac{\partial v_j}{\partial x_k}+\Theta_{jk}-k^F\frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_k}\\&+T_a^F\frac{\partial}{\partial x_k}\left(k^F\frac{\partial T^F}{\partial x_j}\right)+j_{S_{j,k}}^{F,g}\Bigg]n_j+j_{\Omega}^F n_k\Bigg\}\frac{\delta x_k}{\delta b_n}dS\\&+\int_{S_{W_p}^S}\left\{\left[-k^S\frac{\partial T_a^S}{\partial x_j}\frac{\partial T^S}{\partial x_k}+T_a^S\frac{\partial}{\partial x_k}\left(k^S\frac{\partial T^S}{\partial x_j}\right)+j_{S_{i,k}}^{S,g}\right]n_j+\left(j_{\Omega}^S+j_{\Omega}^{S,g}\right)n_k\right\}\frac{\delta x_k}{\delta b_n}dS\\&-\int_{S_{W_p}^F}\left[\frac{\partial j_{S_{W_p,k}}^F}{\partial \tau_{lm}}n_k t_l^I t_m^I\tau_{ij}\frac{\delta(t_i^I t_j^I)}{\delta b_n}\right]dS-\int_{S_{W_p}^F}\left[\frac{\partial j_{S_{W_p,k}}^F}{\partial \tau_{lm}}n_k t_l^{II} t_m^{II}\tau_{ij}\frac{\delta(t_i^{II} t_j^{II})}{\delta b_n}\right]dS\\&-\int_{S_{W_p}^F}\left[\left(-u_{\langle n\rangle}+\frac{\partial j_{S_{W_p,k}}^F}{\partial \tau_{lm}}n_k n_l n_m\right)\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n}\right]dS\\&-\int_{S_{W_p}^F}\left[\left(\frac{\partial j_{S_{W_p,k}}^F}{\partial \tau_{lm}}n_k(t_l^{II}t_m^I+t_l^I t_m^{II})\right)\tau_{ij}\frac{\delta(t_i^{II}t_j^I)}{\delta b_n}\right]dS\\&+\int_{\overline{S}_p^F\cup S_{W,D,p}^F\cup S_{W,Fl,p}^F}k^F T_a^F\frac{\partial T^F}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS+\int_{\overline{S}_p^S\cup S_{W,D,p}^S\cup S_{W,Fl_p}^S\cup S_{CH_p}^S}k^S T_a^S\frac{\partial T^S}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS\end{aligned}$$

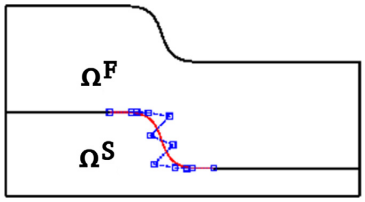


Figure 3.3: Geometry and FSI parameterization in Cases 4 and 5, involving the turbulent flow in a $2D$ S-shaped duct (Ω^F) attached to the solid body (Ω^S).

$$+\sum_{D=F,S}\left[\int_{S^D_{W_p}} j^D_{S,i}\frac{\delta\left(n_i dS\right)}{\delta b_n}+\int_{S^D_{W_p}} j^D_S\frac{\delta\left(dS\right)}{\delta b_n}-\int_{S^D_{W_p}}\frac{\partial m_i^{D,a}}{\partial n}\frac{\delta x_i}{\delta b_n}dS\right] \tag{3.46}$$

As in Chapter 2, by omitting the last term on the r.h.s. of eq. 3.46, written for both the fluid and solid, the (Severed) *SI* adjoint for CHT problems arises. In such a case, since $m_i^{D,a}$ is not needed, eqs. 3.45a and 3.45b are not solved.

3.4 Comparison of the Alternative Adjoint Formulations

In this section, two studies are carried out to demonstrate the advantages of developing the continuous adjoint method according to the way presented in section 3.3, namely by (a) formulating and solving the adjoint to the turbulence model equation(s) and (b) considering internal grid displacement according to the *E-SI* adjoint. In both studies, the same CHT problem is considered, involving the turbulent flow ($Re=10^4,\ y^+<0.018$ at the first cell barycentres off the wall) in a $2D$ S-shaped duct and the adjacent solid body (fig. 3.3). Along S_I^F, $T^F=291.2\ K$ and along the non-FSI solid walls $T^S=300\ K$. Also, $c_p=4181\ J/kg/K$, $Pr=0.1$ and $k^S=60\ J/K/m/sec$. The two studies are based on different objective functions. SDs are computed w.r.t. the x and y coordinates of 12 NURBS CPs (in blue as seen in fig. 3.3, connected by a dotted line), excluding the first and last of them, which parameterize the central part of the FSI boundary (in red in fig. 3.3).

In Case 4, SDs are computed with the *FI*, (Severed) *SI* and *E-SI* adjoints and FDs, for the J^{penT} function defined as

$$J^{penT}=\int_{\Omega^S} j_\Omega^{S,T}d\Omega \tag{3.47}$$

where $j_\Omega^{S,T}=\frac{j^{penT}}{V_S}$, and

$$j^{penT}=\begin{cases}\left[1-\frac{1}{1+e^{k_2(T^S-T_{crit})+k_1}}\right] & ,if\ T^S\le T_{crit}\\ \alpha\left(T^S-T_{crit}\right)+\beta & ,if\ T^S>T_{crit}\end{cases} \tag{3.48}$$

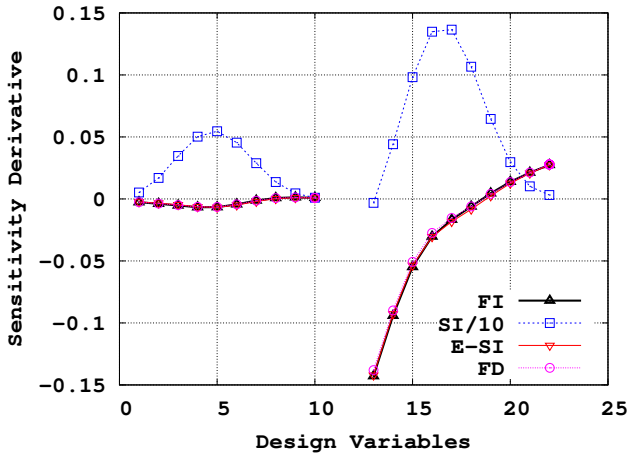


Figure 3.4: Case 4: SDs of J^{penT} computed using the *FI*, (Severed) *SI* and *E-SI* adjoints as well as FDs.

$$V^S = \int_{\Omega^S} d\Omega \tag{3.49}$$

and $k_1 = log\left(\frac{1}{1-f_{max}} - 1\right)$, $k_2 = \frac{log\left(\frac{1}{1-f_{mim}}-1\right)-k_1}{T_{safe}-T_{crit}}$, $\alpha = \frac{k_2 e^{k_1}}{(1+e^{k_1})^2}$, $\beta = \frac{e^{k_1}}{1+e^{k_1}}$ with f_{min}, f_{max} being two user-defined constants and T_{safe} and T_{crit} ($T_{safe} < T_{crit}$) two temperature thresholds. $T^S > T_{crit}$ corresponds to an absolutely undesirable local overheating whereas T_{safe} pertains to the temperature below which safe operation is guaranteed. The choice of these values depends on the thermal properties of the solid body material. Eq. 3.47 measures the part of Ω^S with temperature values exceeding a threshold defined by the designer. More details about this objective function can be found in Appendix B.3. In fig. 3.4, it can be seen that the SDs computed by the *E-SI* adjoint match those resulting from the *FI* adjoint and FDs, whereas the (Severed) *SI*-based SDs are wrongly signed for some of the CPs and are scaled down by 10 to fit into the same plot.

In Case 5, the effect of omitting the differentiation of the turbulence model equation is investigated. Also, the assumption of omitting the contribution of the fluid energy equation to the adjoint turbulence model i.e. the eighth term in eq. 3.25d, is evaluated. To investigate the error introduced due to the "frozen turbulence" assumption, the *E-SI* adjoint is exclusively used. In addition, the SDs computed by the three adjoint methods are compared to FDs. The objective function is the mean temperature over the solid domain, defined as

$$J^{meanT} = \frac{\int_{\Omega^S} T^S d\Omega}{V^S} \tag{3.50}$$

The computed SDs are presented in fig. 3.5. Fig. 3.5 convincingly shows that it is

important to solve the adjoint turbulence model PDEs since such an omission may result in wrongly signed SD for at least the last 4 design variables. In addition, it is shown that solving the adjoint to the Spalart–Allmaras model PDE without considering the contributions stemming from the differentiated energy equation (i.e. omitting the eighth term from eq. 3.25d) leads to wrongly signed SDs as well. Also, in fig. 3.5, SDs computed with the *FI*, (Severed) *SI* and *E-SI* adjoints, by also considering the adjoint turbulence model equation, are shown. The *E-SI* gradient matches those computed by the *FI* adjoint and FDs whereas some (Severed) *SI* derivatives are wrongly signed and need to be divided by 100 to fit into the same plot. In terms of cost to compute the SDs, after numerically solving the primal and adjoint systems of equations, the time needed to compute the *E-SI*-based SDs is 7 % of that needed by the *FI* adjoint, see table 3.1. The difference in cost becomes higher by increasing the design variables number.

Adjoint formulation	**CPU Cost [sec]**	**% of FI**
FI	36.74	100
(Severed) *SI*	0.43	1.2
E-SI	2.67	7.2

Table 3.1: Case 5: Cost of computing the SDs of J^{meanT} with the *FI*, (Severed) *SI* and *E-SI* adjoints, for $N=20$. Computations are performed in parallel on four Intel(R) Xeon(R) CPU E5-2630 v2 @ 2.60GHz cores.

3.5 Case 6: Shape Optimization of an Internally Cooled 2D Turbine Cascade

In Case 6, a 2D CHT optimization of an internally cooled turbine blade is presented; the Ω^F and Ω^S grids consist of $\sim 30K$ and $\sim 19K$ cells, respectively. Regarding the fluid flow $Re = 53000$, the non-dimensional distance of the first barycentres off the wall is $y^+ < 0.15$. Ten distinct circular cooling holes exist inside the blade. Since the flow inside the coolant passages is perpendicular to the plane on which the CHT problem is solved, heat transfer along the coolant hole contours is modeled through eq. 3.8, with $h_{coolant} = 2000\ J/m^2/sec/K$ and $T_{cool} = 432\ K$ being the constant/uniform coolant temperature.

The function to be minimized is J^{meanT}. The design variables are (a) the x and y coordinates of 62 NURBS CPs parameterizing the pressure and suction sides of the blade airfoil and (b) the displacements of the 10 cooling holes in the x and y directions; the radii of the holes are kept fixed. In fig. 3.6, the initial and optimized designs are presented. By comparing them, one may notice that

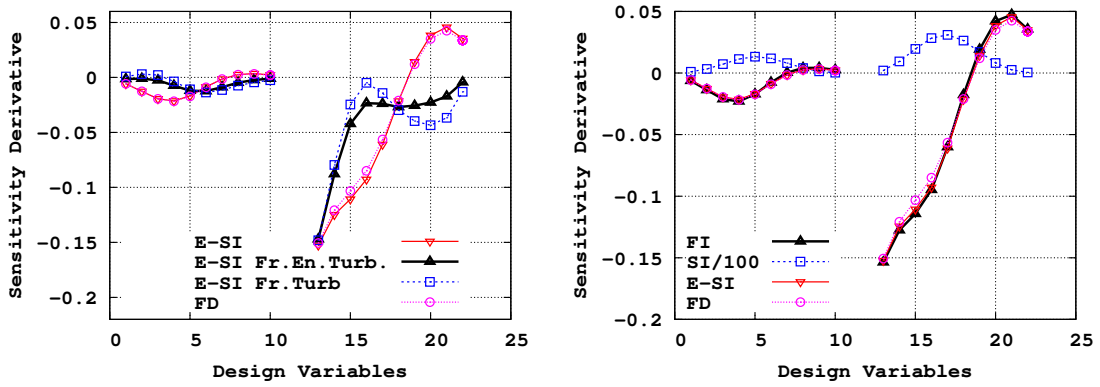


Figure 3.5: Case 5: SDs of J^{meanT}. Left: Effect of the "frozen turbulence" assumption on the SDs. Apart from computing SDs with the *E-SI* adjoint, the *E-SI* formulation was additionally used without differentiating the Spalart-Allmaras PDE (*E-SI* Fr.Turb.) as well as having differentiated it but omitting the contribution of the differentiated energy equation to the adjoint turbulence equation (*E-SI* Fr.En.Turb.). Right: Comparison of the *FI*, (Severed) *SI* and *E-SI* adjoints; in all of them, the adjoint to the Spalart-Allmaras PDE is included.

the optimization algorithm brings the pressure and suction sides closer to the cooling holes. The largest displacement occurs close to the leading and trailing edges where temperature values are higher. Also, the first five cooling holes (as seen from the leading to the trailing edge) are displaced closer to the leading edge, in order to lower the locally high T^S values. Close to the trailing edge, the blade thickness decreases but the optimization does not displace the holes there, even if these are free to move. In fig. 3.7, the objective function convergence is shown. After 8 optimization cycles, the mean temperature of the blade is reduced by $2.7\ K$; in this figure, the zero-th cycle corresponds to the initial geometry. In figs. 3.8 and 3.9, the distributions of T^F and T^S and the corresponding adjoint temperature distributions are presented over the initial and optimized geometries. Higher T^S values occur close to the leading and trailing edges, since those areas are mostly exposed to the increased fluid temperature. Also, the adjoint wake travels from the leading edge towards the inlet, in contrast to the primal one. By focusing on Ω^S, the decrease in the mean temperature results from the T^S drop in each and every cell of the blade, fig. 3.10; T^S mostly changes between the leading edge and the upper cooling hole which came closer, whereas a smaller drop in T^S occurs close to the trailing edge.

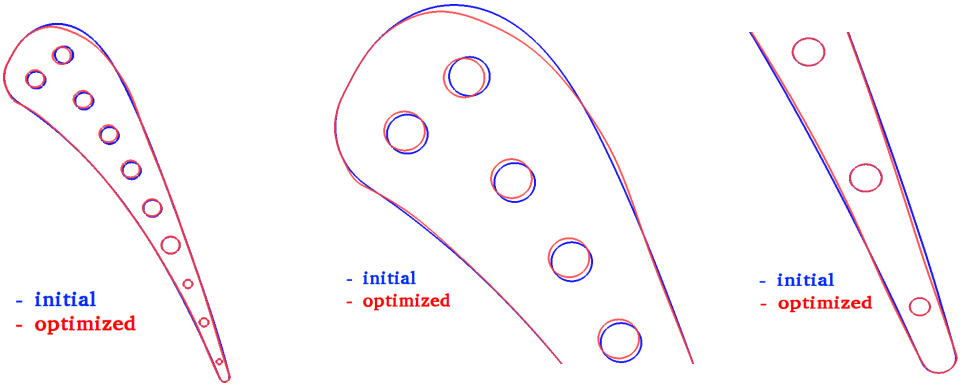


Figure 3.6: Case 6: Initial and optimized geometries, details close to the leading and trailing edges.

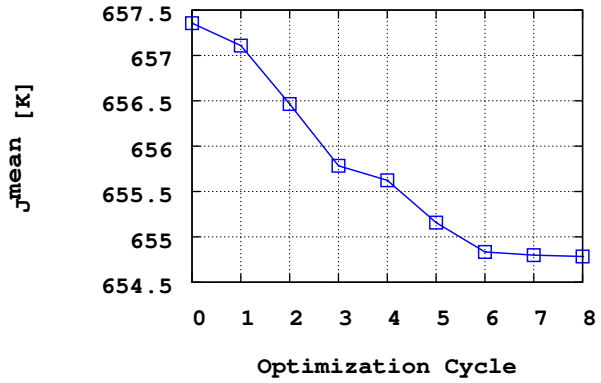


Figure 3.7: Case 6: Convergence of the objective function.

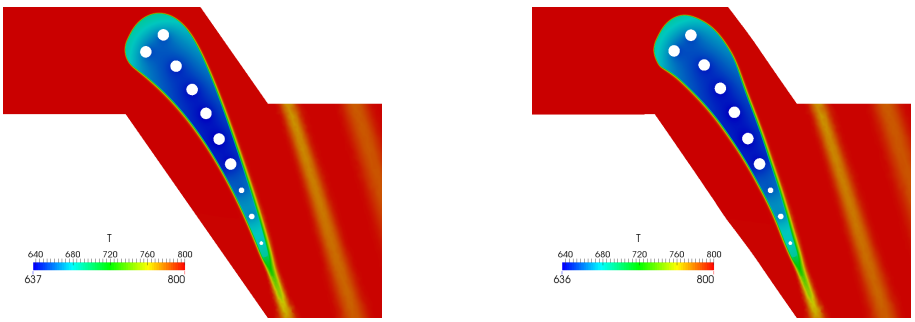


Figure 3.8: Case 6: Temperature distribution in both domains for the initial (left) and optimized (right) geometries.

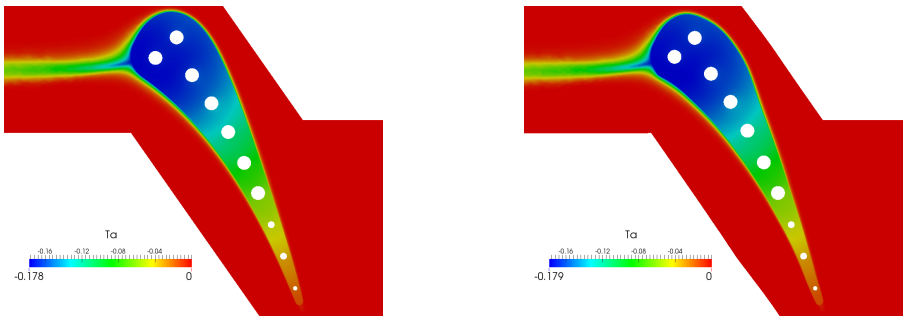


Figure 3.9: Case 6: Adjoint temperature distribution in both domains for the initial (left) and optimized (right) geometries.

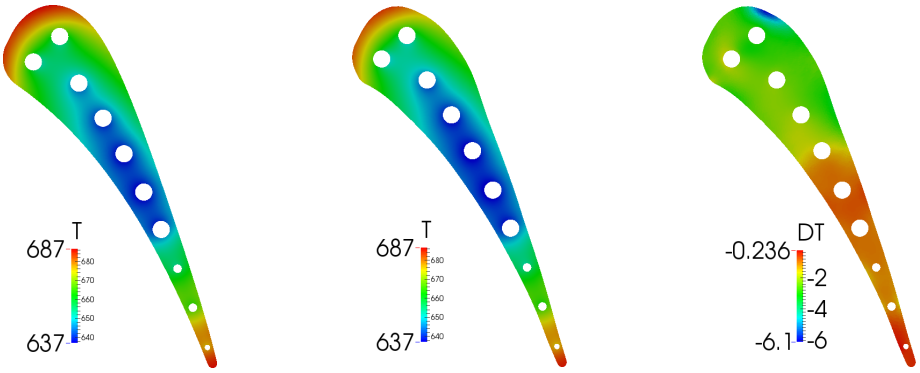


Figure 3.10: Case 6: T^S distributions over the initial (left) and optimized (mid) geometries along with their difference (right), plotted over the optimized geometry.

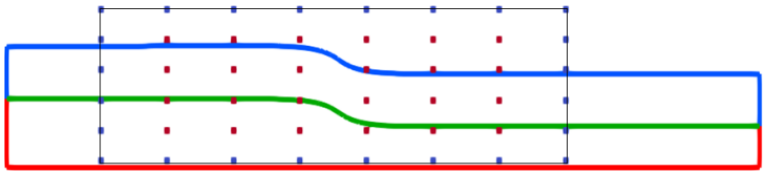


Figure 3.11: Case 7: Parameterization, based on volumetric B-Splines. The red CPs are free to move and the blue ones are kept fixed. More details on the volumetric B-Splines morpher can be found in Chapter 5, where the effect of the choice of the grid displacement models on the SDs values is discussed.

3.6 Case 7: Constraint Shape Optimization of an S-Shaped Cooling Channel

Case 7 regards a solid body which is cooled by a fluid passing above it, through an S-shaped cooling channel, fig. 3.11. To impose a max. T^S constraint during the shape optimization process, the CHT problem is needed to be solved. Comparisons are made with the case when the constraint is not imposed. The coolant flow is

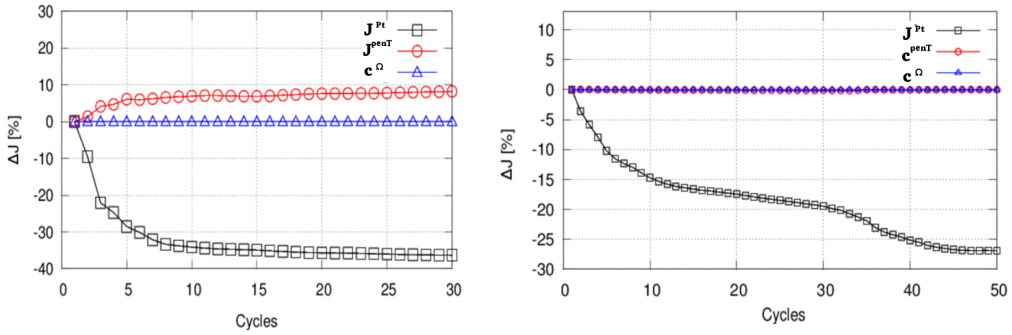


Figure 3.12: Case 7: Convergence for (left) Optimization 1 and (right) Optimization 2. In both cases, c^{Ω} denotes the equality constraint imposed on the Ω^S volume. Also, c^{penT} is the equality constraint imposed on J^{penT}.

turbulent ($y^+ < 1$, $Re = 10000$), entering with $T^F = 291.2\ K$, the rest of the non-FSI fluid boundaries are adiabatic and a constant temperature $T^S = 500\ K$ along the rest of the non-FSI boundaries is considered. Two different optimizations are carried out:

- Optimization 1: The target is to minimize J^{pt}, eq. B.3, under the equality constraint that the solid volume, $\int_{\Omega^S} dS$ remains the same. Also, J^{penT} is not considered during the optimization and its value is only computed and shown.

- Optimization 2: The target is to minimize J^{pt}, eq. B.3, under the equality constraints that both (a) the volume of the solid, $\int_{\Omega^S} d\Omega$ and (b) J^{penT} remain fixed. The second constraint is imposed to prevent the max. T^S values from increasing, when J^{pt} is minimized.

The parameterization can be seen in fig. 3.11. The results of the two optimizations are presented in fig. 3.12 and in Table 3.2. Both constrained problems

	Optimization 1	**Optimization 2**
$\boldsymbol{\Delta J^{pt}}$	-36.3%	-26.9%
$\boldsymbol{\Delta J^{penT}}$	$+8.2\%$	$+0.003\%$

Table 3.2: Case 7: Change of J^{pt} and J^{penT} for Optimizations 1 and 2.

are solved by using Sequential Quadratic Programming (SQP) [39]. Further discussion on the SQP method is made in Chapter 6. As seen in table 3.2. in both optimizations, the equality constraint imposed on the Ω^S volume is satisfied; also, in Optimization 2, the equality constraint imposed on J^{penT} is met. In addition, in both optimizations, J^{pt} is reduced. However, in Optimization 1, a larger reduction

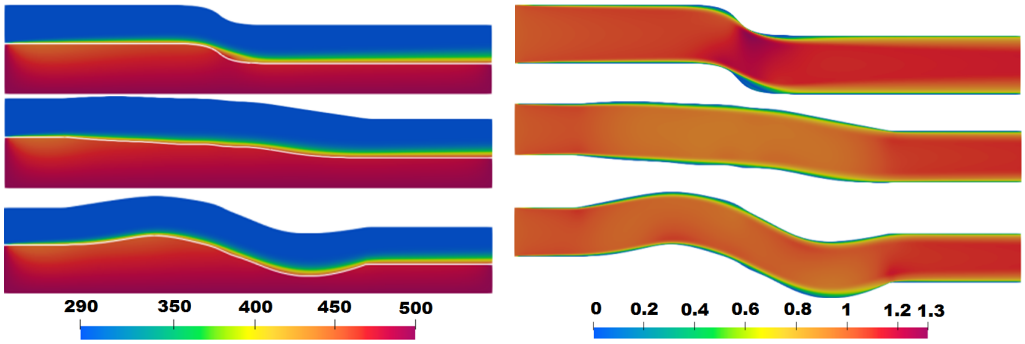


Figure 3.13: Case 7: Temperature (left) and velocity magnitude (right, x,y axes not in scale) fields over the baseline geometry (top) and the optimized geometries of Optimization 1 (middle) and Optimization 2 (bottom). In both cases, the same volume constraint is imposed.

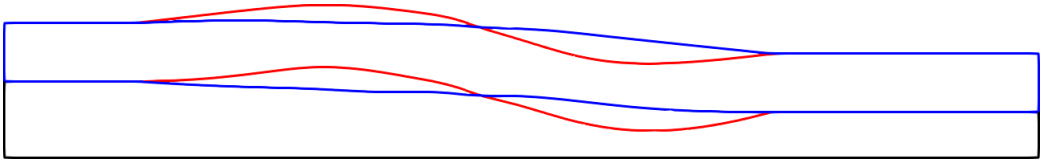


Figure 3.14: Case 7: Contours of the optimized geometries resulted from Optimization 1 (blue) and Optimization 2 (red).

in J^{pt} is achieved and a 8% increase in J^{penT} occurs. This is due to the fact that, in Optimization 1, focusing exclusively on J_{pt}, heat conduction between the solid and the coolant is slightly reduced compared to Optimization 2, due to a shorter FSI length, see figs. 3.13 and 3.14. Also, to minimize J^{pt}, velocity magnitudes are decreased, see fig. 3.13, deteriorating thus heat convection and inevitably leading to higher T^S and J^{penT}. In Optimization 2, velocity magnitude is again reduced so that J^{pt} is minimized. In this case though, to prevent J^{penT} from increasing, as happened in Optimization 1, the FSI contour became wavy. This increased the contact area between Ω^F and Ω^S and thus, the solid was cooled more effectively than the outcome of Optimization 1. Hence, the value J^{penT} remained the same as the initial one.

3.7 Conclusions

In this chapter, the continuous adjoint method for Conjugate Heat Transfer problems with turbulent flows was developed, according to the *FI*, (Severed) *SI* and *E-SI* adjoint methods. Emphasis was laid on the proper treatment of grid sensitivities and the exact differentiation of the turbulence model. Through numerical investigations, the advantages of the *E-SI* continuous adjoint, as initially proposed for pure fluid flow problems, see Chapter 2, were reconfirmed in CHT problems too.

In particular, it was shown that, by developing the adjoint to the grid displacement PDEs and avoiding the computation of grid sensitivities at the interior of the fluid and solid domains, accurate SDs were computed by the *E-SI* adjoint at a lower computational cost. Regarding the importance of differentiating the turbulence model equations, it was shown that the "frozen turbulence" assumption can have a deteriorating effect on the SDs accuracy and may even lead to wrongly signed SDs. In two optimization problems, thanks to the computation of accurate SDs, a noticeable reduction in the objective functions was observed within just a few optimization cycles, whereas constraints were also satisfied.

Chapter 4

3D Shape Optimizations in CHT Problems

In this chapter, two 3D shape optimization cases are studied by using the developed adjoint for CHT problems. The first one concerns a U-shaped cooling channel, fully placed inside a rectangular solid body (Case 8). In the second, optimizations are carried out for the cooling system of a car-engine cylinder-head (Case 9).

4.1 Case 8: U-Shaped Internal Cooling Channel

In Case 8, the cooling effectiveness of the internal cooling system of fig. 4.1a is optimized. The geometry to be optimized comprises a 3D U-shaped cooling duct, located inside a rectangular solid body. Along the non-FSI walls of Ω^S, a Dirichlet temperature condition ($T^S = 400\ K$) is imposed. The fluid flow is turbulent ($Re = 84000$ based on the duct hydraulic diameter, average $y^+ = 0.77$ for the first barycenters off the wall), with inlet velocity magnitude equal to $16.8 m/s$ being parallel to the axial direction and $T^F\big|_{S_I^F} = 288\ K$. Also, $c_p = 4181\ J/K/kg$, $Pr = 0.1$, $Pr_t = 1$ and $k^S = 60\ J/K/m/s$.

Two different optimizations, each focusing on a different objective function, are carried out:

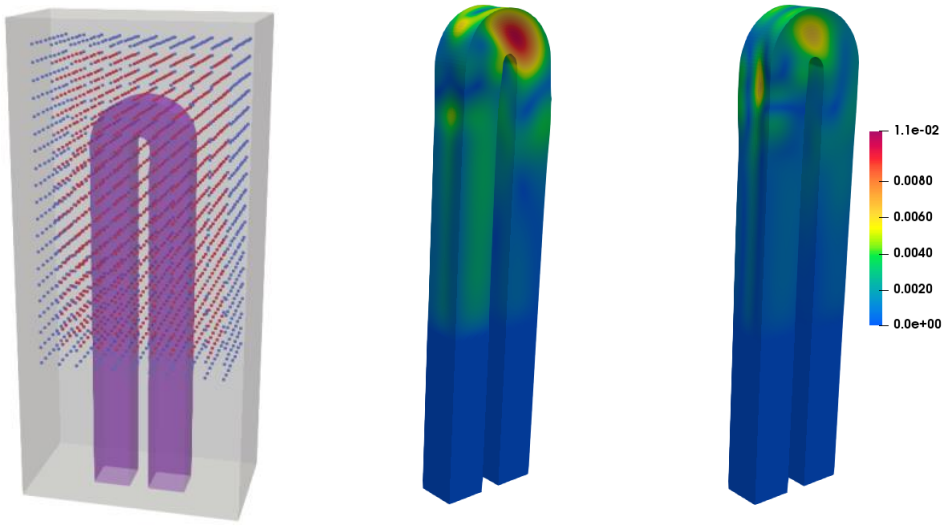
- Optimization 1: The goal is to minimize the volume of the solid with higher T^S values. This can be achieved by minimizing J^{penT}, see eq. B.5, for which $T_{safe} = 360\ K$ and $T_{crit} = 395\ K$.

- Optimization 2: The goal is to increase the heat flux from the heated solid body to the coolant (or, equivalently, to maximize the heat absorbed by the coolant), which is equivalent to minimizing J^{HF}, given by

$$J^{HF} = -\int_{\overline{S}^F} k^F \frac{\partial T^F}{\partial n} dS \tag{4.1}$$

corresponding to the integral of the heat flux over $\overline{S}^F$. The differentiation of J^{HF} w.r.t. b_n can be found in Appendix B.5.

Design variable are the x, y, z coordinates of the CPs in fig. 4.1a and all boundary CPs are kept fixed. Optimization 1 results in a reduction in J^{penT} by 2.7% and Optimization 2 in a reduction in J^{HF} by 7.5%. In figs. 4.1b and 4.1c, the cumulative displacement of the FSI boundary, for the two optimizations, is presented. Similarities between the two optimized solutions are observed: For both objective



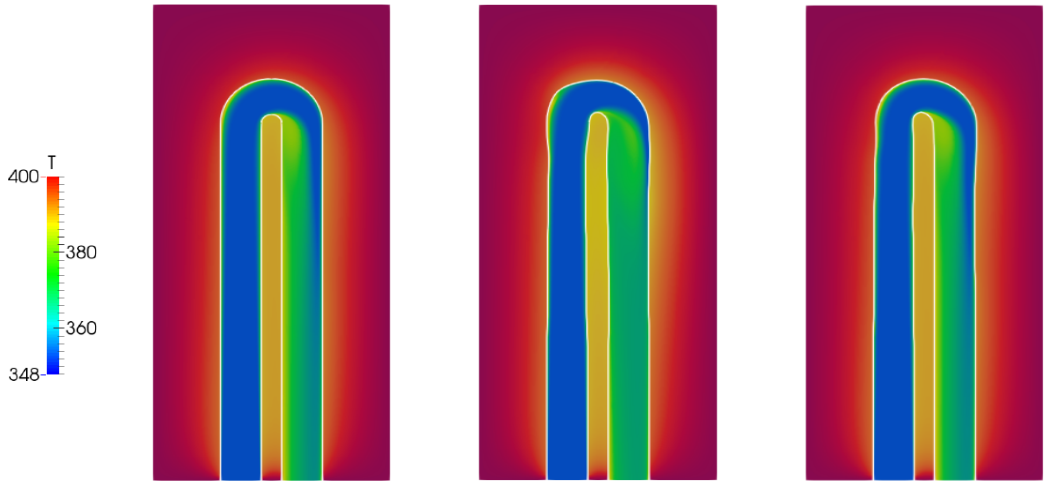
(a) Parameterization **(b)** Min. J^{penT} **(c)** Max. J^{HF}

Figure 4.1: Case 8: (a) A $11 \times 20 \times 9$ Volumetric B-Splines control box parameterized the geometry. CPs in blue are kept fixed and red ones are allowed to move. (b,c) FSI cumulative displacement for min. J^{penT} and min. J^{HF}, respectively.

functions, the FSI, as seen from Ω^F is displaced inwards, in order to reduce the cross-sectional area of the cooling channel, along its turning part. By looking at figs. 4.2 and 4.3, plotting the temperature and velocity magnitudes for the optimized geometries, the aforementioned geometric changes cause an increase in the velocity magnitude at the top of the U-shaped duct and at the part following the flow turning. As a result, heat convection is positively affected and thus smaller temperatures near the FSI occur, reducing thus J^{penT}; a larger temperature gradient and a bigger surface of the FSI leads also to the minimization of J^{HF}.

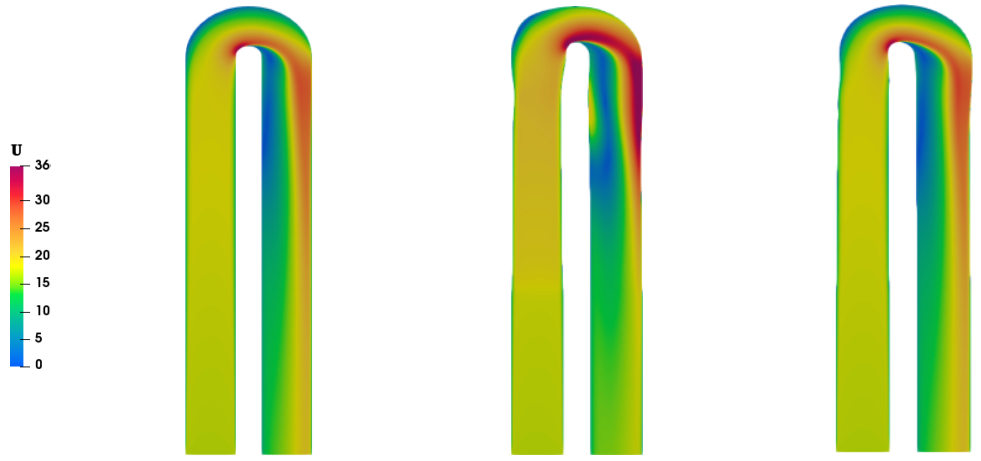
4.2 Case 9: Car-Engine Cooling System

Case 9 focuses on the behavior of the total pressure losses in the coolant flowing inside the cooling system of a car-engine cylinder-head when minimizing max. T^S values and vice-versa.



(a) Initial Geometry **(b)** Min. J^{penT} **(c)** Max. J^{HF}

Figure 4.2: Case 8: Temperature fields for Ω^F and Ω^S, plotted on a slice in the middle of the duct width. Initial (a) and optimized geometries regarding (b) min. J^{penT} and (c) min. J^{HF}.



(a) Initial Geometry **(b)** Min. J^{penT} **(c)** Max. J^{HF}

Figure 4.3: Case 8: Velocity magnitude over Ω^F for the initial (a) and optimized geometries regarding (b) min. J^{penT} and (c) min. J^{HF}.

The geometry can be seen in fig. 4.4. The fluid domain is enclosed by the cooling channel which is formed by the FSI boundary (in purple; it consists of two horizontal ducts connected by five curved ducts positioned between the cylinders), the inlet and outlet surfaces, with the coolant entering from the lower horizontal duct (marked as “inlet”) and exiting from the upper horizontal duct (“outlet”). The domain enclosed by the cylinders, the intake manifold, the exhaust pipes, the surface of the cooling channel and the outer rectangular surface, corresponds to

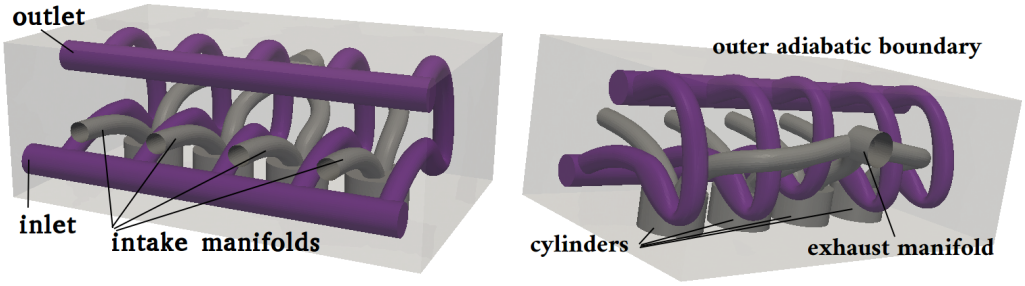


Figure 4.4: Case 9: Two perspective views of the same geometry.

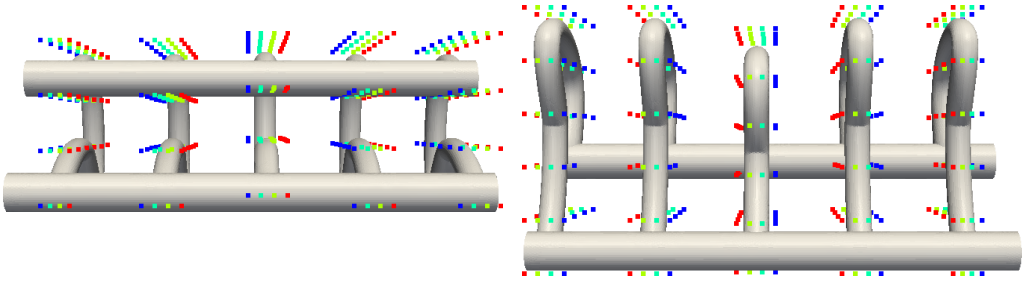


Figure 4.5: Case 9: Frontal (left) and top (right) views of the parameterization of each curved duct, based on Volumetric B-Splines. Each control box consists of $4 \times 4 \times 6$ CPs (along the x, y and z axis, respectively) with a degree of 3 for all parameterized directions. CPs are coloured based on their first parametric coordinate in their respective structured control grid.

Ω^S. The outer rectangular surface of the solid body is an adiabatic boundary. A constant heat flux is imposed along the air-intake manifolds ($Q = 1000\ W/m^2$), the four cylinders ($Q = 2000\ W/m^2$) and the exhaust duct ($Q = 2000\ W/m^2$). Inside the cylinder-head, high temperature values occur due to the high heat flux emanating from the four cylinders and the exhaust duct. The coolant flow is turbulent $Re = 34000$, $c_p = 3554\ J/kg/K$, $k^S = 168\ J/m/sec/K$, $T^F\big|_{S_I^F} = 382.15\ K$.

Shape parameterization is based on Volumetric B-Splines, using 5 independent control boxes, one for each curved duct. fig. 4.5. One boundary CP is kept fixed along the x and y axes and two along the z axis in order to preserve continuity between the parameterized and non-parameterized parts of the grids.

Two independent optimizations are carried out:

1. Optimization 1: The goal is to minimize the max. T^S values, expressed by J^{penT}, i.e. eq. B.5, in which $T_{safe} = 410\ K$ and $T_{crit} = 480\ K$. The volume-weighted total pressure losses J^{pt}, given by eq. B.3, between the inlet and the outlet of the cooling system is not considered during the optimization, but its value is computed.

2. Optimization 2: The goal is to reach min. J^{pt}, without considering J^{penT} as

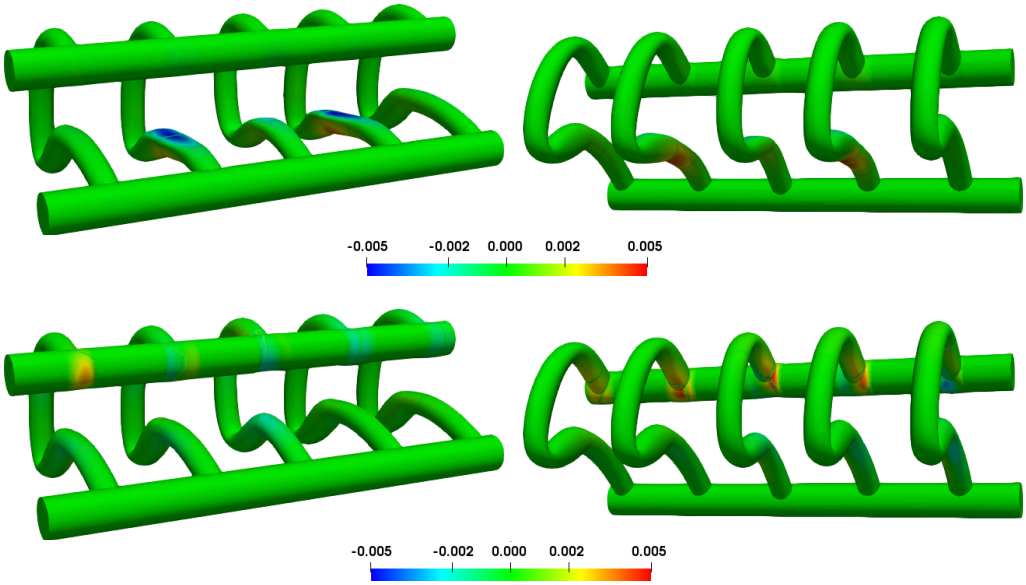


Figure 4.6: Case 9: Frontal (left) and rear (right) views of the cumulative displacement of the FSI, projected on $\mathbf{n}^{\mathbf{S}}$, for Optimizations 1 (top) and 2 (bottom). Blue and red colored areas indicate inwards and outwards displacement w.r.t. Ω^F, respectively.

a target; the value of J^{penT} is though computed and discussed.

The outcome of Optimization 1 is a reduction in J^{penT} by 15.9%. Although minimizing J^{pt} was not considered during the optimization, it was reduced by 1.5%. Regarding Optimization 2, J^{pt} was minimized by 5.6%. Also, J^{penT} was reduced by 0.8%, although it was not considered to be minimized. It can be observed that there is not a unique design corresponding to min. J^{penT} and min. J^{pt} at the same time. By looking at fig. 4.6, it can be observed that in order to minimize J^{penT}, the cooling ducts move downwards, closer to the cylinder-heads. On the other hand, J^{pt} is reduced by inflating the junctions between the horizontal duct leading to the outlet and the five vertical curved ducts. As seen in figs. 4.7, these geometrical changes have the following results in the velocity magnitude:

1. In Optimization 1, the velocity magnitude increases inside the lower horizontal duct, whereas in Optimization 2, this remains practically the same.

2. In Optimization 1, the velocity magnitude increases in the four vertical curved ducts, as seen in figs. 4.7 from right to left and in the fifth duct, becomes lower. In Optimization 2, the opposite changes in the velocity magnitude occur.

3. In Optimization 1, the velocity magnitude increases in the junctions of the upper horizontal duct and the first two vertical curved ducts, as seen in figs. 4.7 from right to left. In Optimization 2 though, the velocity magnitude

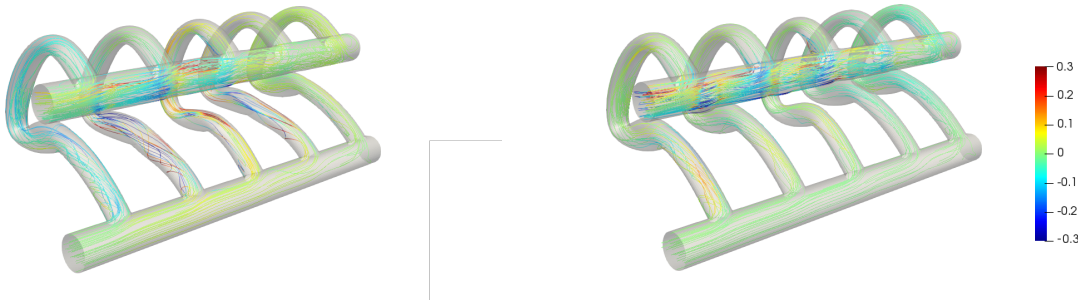


Figure 4.7: Case 9: Difference in velocity magnitude between the optimized and initial geometries for Optimizations 1 (left) and 2 (right).

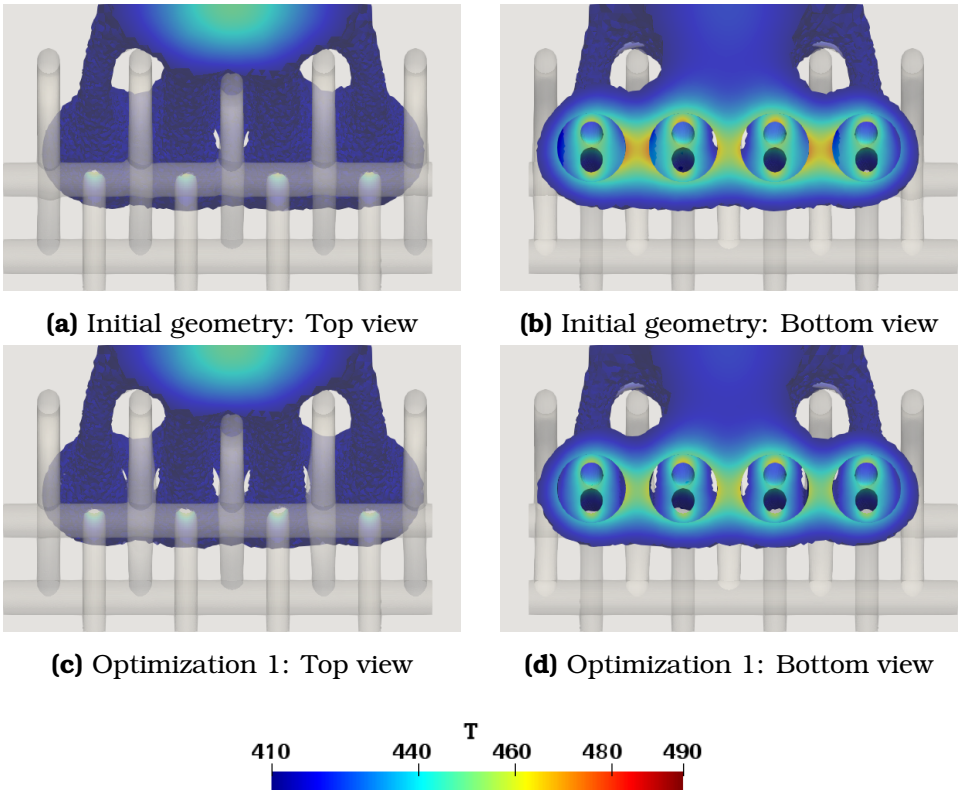
decreases at each junction between the upper horizontal duct and the five vertical ducts.

An increase in the velocity magnitude in Optimization 1 is desired to enhance the temperature convection in the coolant. However, in Optimization 2, in order to minimize the energy dissipated due to friction and, thus, J^{pt}, the reduction in the velocity magnitude is necessary.

Regarding Optimization 1, as seen in fig. 4.8, the Ω^S volume with T^S between T_{safe} and T_{crit} becomes smaller. Finally, close to the cylinder-heads, it can be observed that T^S drops.

4.3 Conclusions

In this chapter, the continuous adjoint method for CHT problems with turbulent flows, presented in Chapter 3, has been used to drive shape optimizations of 3D internal cooling systems, namely of a U-shaped cooling channel and an internally cooled car-engine cylinder-head. To optimize the cooling efficiency of the U-shaped cooling channel, two optimizations with different targets were performed. In the first optimization, the goal was the maximization of the heat flux absorbed by the coolant and in the second the minimization of max. T^S values. Both of the optimizations yielded similar mechanisms for optimized cooling effectiveness: By increasing the velocity magnitude of the coolant and the FSI area, the temperature convection and the temperature conduction were positively affected. Finally, by performing in a car-engine cylinder-head two different optimizations, one aiming to minimize J^{pt} and a second to minimize J^{penT}, it was observed that it was not possible to produce a new design that minimizes both of the targets at the same time. This was justified by the opposite change in the velocity magnitude for the two optimizations. To minimize J^{pt}, the velocity magnitude reduced so that friction losses become smaller. On the other hand, increasing the velocity magnitude was beneficial for minimizing J^{penT}, since in this way the temperature convection is enhanced.



(a) Initial geometry: Top view

(b) Initial geometry: Bottom view

(c) Optimization 1: Top view

(d) Optimization 1: Bottom view

Figure 4.8: Case 9: Volume of Ω^S with $T^S > T_{safe}$, for the initial (a and b) and optimized for min. J^{penT} (c and d) geometries.

Chapter 5

Effect of Grid Displacement Models on Sensitivity Derivatives in Shape Optimization

For both continuous [72, 49] and discrete [163] adjoint methods, it has been demonstrated that considering the displacement of the grid interior is important for accurately computing SDs. A few recent articles [105, 163] discuss the effect of the selected GDM on SDs, in discrete adjoint. Both are concerned with external aerodynamics and compressible fluid flows, with the force exerted on bodies being the objective function. In [163], GDMs such as a Finite Element analogy, Radial Basis Functions and a modified version of the spring analogy, are utilized to compute lift SDs for an isolated airfoil and all lead to practically identical results. In [105], a similar study is conducted using the Radial Basis Functions, linear elasticity and Delaunay Graph based GDMs and a minor effect on the SDs of a transonic wing is observed. In addition, [105] takes the study one step further and demonstrates that the effect of an inconsistent GDM differentiation (differentiating a GDM in the adjoint which is different than that used in the shape optimization loop) is almost negligible.

The goal here is to carry out a broader investigation of the effect of the selected GDM on the SDs values. The outcome of this study will show whether the *E-SI* adjoint, in which the convenient assumption of a Laplace GDM is made, needs to be reformulated when the GDM changes. To investigate the effect of the selected GDM on the SDs values, not only external aerodynamics as in [105, 163] but, also, internal aerodynamics, this time for incompressible fluid flows, are studied. In an attempt to generalize findings even further, CHT problems, with two domains controlled by the same GDM, are investigated too. A distinguishing difference w.r.t. [105, 163] which used discrete adjoint is that this study is exclusively based on continuous adjoint. For a thorough investigation, four successive steps are performed:

1. First, SDs computed by FDs for various GDMs are compared. This step refrains from involving adjoint SDs. Considering that FDs compute reference SDs, the objective of this investigation is to show if the latter are affected by the selected GDM or not. The outcome of this investigation is of interest for researchers developing both continuous and discrete adjoint methods.

2. SDs computed by the *E-SI* adjoint, in which a Laplace GDM is assumed and the adjoint to it is developed, are compared with the FDs based on the various GDMs of the first investigation.

3. A term-by-term analysis on the adjoint-based SDs is performed, to shed light into the findings of the first two steps. To facilitate this investigation, the *FI* continuous adjoint, including grid sensitivities at the interior of the computational domain is used. Grid sensitivities are directly influenced by the GDM, making the *FI* adjoint a good candidate for investigating the results of the first two steps.

4. Finally, the effect of GDM on the outcome of shape optimization loops is investigated. To do so, SDs are computed by considering GDMs which are either consistent or not with the actual GDM in use.

Beyond the Laplace GDM, studies include three GDMs frequently used in aerodynamic shape optimization, namely:

- a Delaunay Graph (DG) based model [89],
- the Inverse Distance Weighting (IDW) model [92] and
- a volumetric B-Splines morphing model (VBS) [129].

Below, first a brief description of each GDM is presented in section 5.1. Then, Cases 1,2 and 4 for which the adjoint SDs were verified against FDs in Chapters 2 and 3 are revisited, to carry out the four aforementioned steps.

5.1 Description of used GDMs

Laplace PDEs;

The grid displacement field m_i is assumed to be governed by a system of Laplace PDEs [63]

$$R_i^m = \frac{\partial^2 m_i}{\partial x_j^2} = 0, \;\; i = 1, 2(, 3) \tag{5.1}$$

with Dirichlet conditions corresponding to the known displacements along the boundaries.

IDW Model:

The IDW method [92, 164] explicitly interpolates the known boundary displacements m_i^B at the interior of the domain. The displacement m_i^I at an interior grid point is the sum of a displacement ϱ_i^I due to a rotation around the origin and a pure translation ξ_i^I, i.e. $m_i^I = \varrho_i^I + \xi_i^I$. Field ξ_i^I is computed as

$$\xi_i^I = \frac{\sum_B \omega_B \xi_i^B}{\sum_B \omega_B} \tag{5.2}$$

where $\omega_B = A\left[\left(\frac{L_{def}}{||x_i^I - x_i^B||}\right)^a + \left(\frac{\alpha L_{def}}{||x_i^I - x_i^B||}\right)^b\right]$ are the IDW weights, with A, a, α, b being user-defined constants, L_{def} is an estimated length of the deformation region [92] and $\xi_i^B = m_i^B - \varrho_i^B$. To compute ϱ_i^B, rotation quaternions $\mathcal{R}^B = \mathcal{R}_1^B \mathcal{R}_2^B$ are defined at the boundary points as the inner product of two quaternions, with $\mathcal{R}_1^B$ representing the rotation around the origin (in order to retain the angle between the normal unit vector at the boundary nodes and their position vector) and $\mathcal{R}_2^B$ representing the rotation around the axis of the unit normal vector at the boundary point [145, 98]. In 2D cases, $\mathcal{R}^B = \mathcal{R}_1^B$, since no rotation takes place around the unit normal vector at the boundary points. To compute ϱ_i^B, the displacement due to rotation is given by $P^B = \mathcal{R}^B X^B \left[\mathcal{R}^B\right]^T - X^B$ where $P^B = \left[0, \boldsymbol{\varrho^B}\right]$ and $X^B = \left[0, \mathbf{x^{B,old}}\right]$ [145], with $x_i^{B,old}$ corresponding to the boundary grid nodes coordinates before updating b_n. $\mathcal{R}^B$ are interpolated to the internal grid points as $\mathcal{R}^I = \frac{\sum_B \omega_B \mathcal{R}^B}{\sum_B \omega_B}$ and quaternions corresponding to the displacement due to rotation are computed as $P^I = \mathcal{R}^I X^I \left[\mathcal{R}^I\right]^T - X^I$ where $P^I = \left[0, \boldsymbol{\varrho^I}\right]$ and $X^I = \left[0, \mathbf{x^{I,old}}\right]$ [145], with $x_i^{I,old}$ being the internal grid nodes before displacing the grid. The interpolation of the boundary nodes rotation is optional but is preferred in cases structured-like layers are generated around geometries in viscous flows, in order to preserve grid orthogonality close to the boundary.

Delaunay Graph Model:

Regarding the Delaunay Graph GDM, first a set of representative boundary grid nodes are selected and for them, a Delaunay triangulation of the computational domain, is generated. Then, the graph element in which each internal grid node resides, is identified. By updating the boundary point positions, the graph elements are displaced and new coordinates of the internal grid nodes are computed as

$$x_i = \omega^k D_i^k \tag{5.3}$$

where D_i^k are the coordinates of the vertices of the graph element enclosing an internal node [89] and ω^k are the barycentric coordinates of this node w.r.t. the graph element nodes.

VBS Morpher:

The volumetric B-Splines GDM uses a lattice of CPs, C_i, enclosing (part of) the domain. Volumetric B-Splines can be used as either a surface parameterization tool or a GDM, depending on the set of controlled grid nodes [129]. The relationship between the grid nodes and C_i reads

$$x_i\left(u,v,w,C_i^{klm}\right)=U_{k,p_u}\left(u\right)V_{l,p_v}\left(v\right)W_{m,p_w}\left(w\right)C_i^{klm} \tag{5.4}$$

where u,v,w are the parametric coordinates of the grid nodes, U,V,W are the volumetric B-Splines functions [129] and p_u,p_v,p_w are their corresponding degrees. The basis functions are given by a recursive function, which is defined, for example for U, as

$$U_{k,pu}\left(u\right)=\frac{u-\xi_i}{\xi_{i+pu}-\xi_i}U_{k,pu-1}\left(u\right)+\frac{\xi_{i+pu+1}-u}{\xi_{i+pu+1}-\xi_{i+pu}}U_{k+1,pu-1}\left(u\right) \tag{5.5}$$

with

$$U_{k,0}\left(u\right)=\begin{cases}1 & if\ \xi_k\leq pu<\xi_{k+1}\\ 0 & elsewhere\end{cases} \tag{5.6}$$

In the above expression, $\xi_k\in[0,m]$, $m=n+pu+1$ are the components of a uniform knot vector $\boldsymbol{\xi}$

$$\boldsymbol{\xi}=[\underbrace{0,....,0}_{pu+1},\ \frac{1}{N_u},\ .....,\ \frac{N_u-1}{N_u},\ \underbrace{1....1}_{pu+1}] \tag{5.7}$$

where $N_u=n_u-pu+1$ and n_u is the number of CPs along the parametric coordinate u.

To compute u,v,w at each grid node, a 3×3 non-linear system of equations $x_i(u,v,w)-x_i^r=0,\ i=1,2,3$ is solved once for each grid point, with x_i^r being the nodal coordinates. With known parametric coordinates, the new coordinates of displaced grid nodes are computed using eq. 5.4 for the new C_i values.

5.2 Comparison of FD-based SDs for various GDMs

The effect of GDMs on the SDs computed by FDs is investigated for three cases already examined in previous chapters. In Case 1, the external aerodynamics

problem regarding the flow around the NACA4412 isolated airfoil, presented in section 2.4, is studied. Lift SDs are computed using the Laplace, DG and IDW GDMs. In Case 2, the internal flow problem within an S-shaped duct, presented in section 2.4, is investigated, by using all four GDMs mentioned above. The VBS are implemented in two different ways: (a) their CPs parameterize only the duct shape, without affecting the coordinates of the internal grid nodes, which are displaced separately using the other three GDMs and (b) the VBS morpher simultaneously displaces boundary and internal grid nodes.

Case 4 is concerned with the CHT problem presented in section 3.4, with J^{penT} being the objective function. Since NURBS CPs parameterize the geometry, all but the VBS-based GDMs are used here.

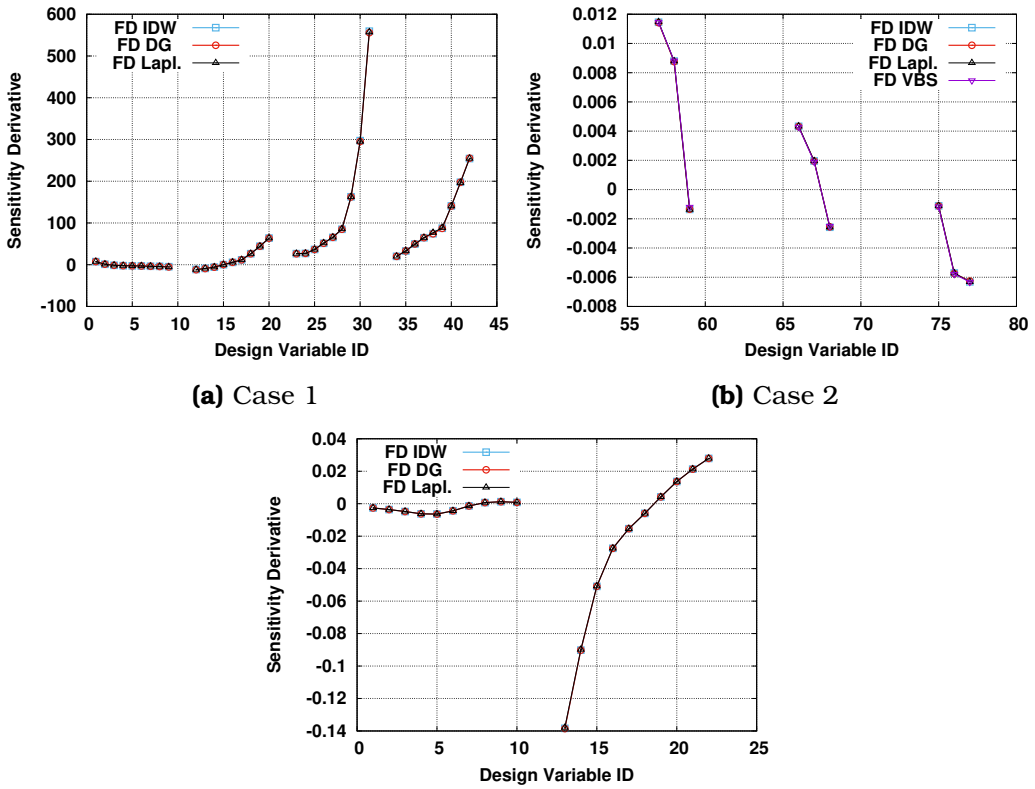
The FD-based SDs, computed for Cases 1,2 and 4 for the various GDMs are presented in fig. 5.1. Average deviation values are given in the captions of the corresponding figures. It is observed that the effect of the chosen GDM on the FD SDs is negligible, since the SDs corresponding to different GDMs practically coincide.

5.3 Comparison of *E-SI* Adjoint SDs and FDs

Based on results already presented in figs. 2.1b, 2.3b and 3.4 for Cases 1,2 and 4 respectively, the *E-SI* adjoint computes SDs which match those computed by FDs assuming a Laplace GDM. In particular, the average relative differences between the SDs are 5%, 5.5% and 8.6% respectively (without considering points with almost zero SDs). Since FDs performed for different GDMs lead to the same SDs, as shown in section 5.2, it is concluded that the *E-SI* adjoint, which assumes a Laplace GDM, can be safely used to compute accurate SDs at low computational cost, irrespective of the GDM in use.

5.4 Why SDs for Different GDMs are the Same?

A thorough investigation on the reasons of the matching of SDs computed with various GDMs is presented here. To do so, the *FI* adjoint is used. Although being computationally more expensive than the *E-SI* adjoint, the *FI* one does not assume a Laplace GDM. Instead it leads to SDs that include field integrals of grid sensitivities, which are directly affected by the used GDM, see eqs. 2.55 and 3.42. First, the differentiation of the GDMs expressions, which yields equations providing the grid sensitivities, is presented. Then, the SD expressions are scrutinized to shed light into the results of sections 5.2 and 5.3.



(a) Case 1

(b) Case 2

(c) Case 4

Figure 5.1: Comparison of FD-based SDs for Cases 1,2 and 4, computed for different GDMs. In Case 1, SDs computed for the IDW and DG differ from those computed for a Laplace GDM by 1.4% and 1.6%, on average, respectively (the 1fifth DV was excluded, since the Laplace-based SD value is close to zero). In Case 2, SDs computed for the IDW, DG and VBS differ from those computed for a Laplace GDM by 0.2%, 0.7% and 3%, on the average, respectively. In Case 4, SDs computed for the IDW and DG GDMs differ from those computed for a Laplace GDM by less than 0.5%.

5.4.1 Grid Sensitivities Computation for the used GDMs

In case of GDMs described by differential equations, as the Laplace one, N systems of PDEs resembling the GDM ones arise and are solved to compute $\delta x_i/\delta b_n$. When GDMs are described by algebraic expressions, such as the VBS, DG and IDW ones, their differentiation w.r.t. b_n yields N closed-form expressions, which are similar to the GDMs equations. Computing the grid sensitivities as described above has a cost equivalent to N grid displacements.

Laplace PDEs;

To compute grid sensitivities based on the Laplace GDM, eq. 5.1 is differenti-

ated w.r.t. b_n, yielding

$$\frac{\partial^2}{\partial x_j^2}\left(\frac{\delta x_i}{\delta b_n}\right)=0 \tag{5.8}$$

The above PDEs are solved by using the known values of $\delta x_i/\delta b_n$ from the parameterization as boundary conditions.

IDW Model:

To derive the closed-form expressions for the grid sensitivities, first the dependence of the internal and boundary grid nodes should be defined as $x_i^I\left(b_n, x_i^{I,old}(b_n^{old})\right)=m_i^I\left(x_i^{old}(b_n^{old}), m_i^B(b_n)\right)+x_i^{I,old}(b_n^{old})$ and $x_i^B(b_n)=m_i^B(b_n)+x_i^B(b_n^{old})$. By differentiating them w.r.t. b_n, $\frac{\delta x_i}{\delta b_n}\big|^I_{b_n=b_n^{old}}=\frac{\delta m_i}{\delta b_n}\big|^I_{b_n=b_n^{old}}$ and $\frac{\delta x_i}{\delta b_n}\big|^B_{b_n=b_n^{old}}=\frac{\delta m_i}{\delta b_n}\big|^B_{b_n=b_n^{old}}$. This means that $\frac{\delta m_i}{\delta b_n}\big|^B_{b_n=b_n^{old}}$ is also given by considering the analytical expression of the boundary parameterization. To compute $\frac{\delta m_i}{\delta b_n}\big|^I_{b_n=b_n^{old}}$ and, thus, $\frac{\delta x_i}{\delta b_n}\big|^I_{b_n=b_n^{old}}$.

$$\frac{\delta x_i}{\delta b_n}\bigg|^I_{b_n=b_n^{old}}=\frac{\delta \varrho_i}{\delta b_n}\bigg|^I_{b_n=b_n^{old}}+\frac{\delta \xi_i}{\delta b_n}\bigg|^I_{b_n=b_n^{old}} \tag{5.9}$$

where $\frac{\delta \xi_i}{\delta b_n}\Big|^I_{b_n=b_n^{old}}=\left(\frac{\sum_B \omega_B \frac{\delta \xi_i}{\delta b_n}\big|^B_{b_n=b_n^{old}}}{\sum_B \omega_B}\right)$ and $\frac{\delta \varrho_i}{\delta b_n}\big|^I_{b_n=b_n^{old}}=0$, since $\frac{\delta P}{\delta b_n}\big|^I_{b_n=b_n^{old}}=\frac{\delta \mathcal{R}}{\delta b_n}\big|^I_{b_n=b_n^{old}}X^I\left[\mathcal{R}^I\right]^T+\mathcal{R}^I X^I\left[\frac{\delta \mathcal{R}}{\delta b_n}\big|^I_{b_n=b_n^{old}}\right]^T=0$. For the same reason $\frac{\delta \varrho_i}{\delta b_n}\big|^B_{b_n=b_n^{old}}=0$ and thus $\frac{\delta \xi_i}{\delta b_n}\big|^B_{b_n=b_n^{old}}=\frac{\delta x_i}{\delta b_n}\big|^B_{b_n=b_n^{old}}$ As a result

$$\frac{\delta x_i}{\delta b_n}\bigg|^I_{b_n=b_n^{old}}=\frac{\sum_B \omega_B \frac{\delta x_i}{\delta b_n}\big|^B_{b_n=b_n^{old}}}{\sum_B \omega_B} \tag{5.10}$$

If the interpolation of the boundary rotation to the interior of the grid is not used, the total boundary displacement is interpolated to the interior of the grid through eq. 5.2, in which ξ_i^B and ξ_i^I are substituted by m_i^B and m_i^I respectively. In such a case, it can be proven in a straightforward way that grid sensitivities are also given by eq. 5.10.

Delaunay Graph Model:

Grid sensitivities are given by differentiating eq. 5.3 w.r.t. b_n, yielding

$$\frac{\delta x_i}{\delta b_n}=\omega^k\frac{\delta D_i^k}{\delta b_n}$$

where $\frac{\delta D_i^k}{\delta b_n}$ is either zero at the non-parameterized boundary points or known at the parameterized ones.

VBS Morpher:

To compute the grid sensitivities, eq. 5.4 is differentiated w.r.t. b_n, which coincide with the $x, y(, z)$ coordinates of the CPs, C_i. The grid sensitivities for the boundary and internal grid nodes are computed as

$$\begin{aligned} \frac{\delta x_i}{\delta b_n} &= \frac{\delta x_i}{\delta C_j^{abc}}\left(u, v, w, C_i^{klm}\right) = U_{k,pu}\left(u\right) V_{l,pv}\left(v\right) W_{m,pw}\left(w\right) \frac{\delta C_i^{klm}}{\delta C_j^{abc}} \\ \Rightarrow \frac{\delta x_i}{\delta b_n} &= \frac{\delta x_i}{\delta C_j^{abc}}\left(u, v, w, C_i^{klm}\right) = U_{a,pu}\left(u\right) V_{b,pv}\left(v\right) W_{c,pw}\left(w\right) \delta_i^j \end{aligned} \tag{5.11}$$

5.4.2 Comparison of *FI* Adjoint SDs and FDs

Before the term-by-term analysis of the *FI*-based SDs, a necessary verification of the *FI*-computed SDs is first made for all three examined cases, to demonstrate that the so-computed SDs are accurate and can safely be used in the term-by-term analysis that follows in section 5.4.3. In figs. 5.2, 5.3 and 5.4 for Cases 1,2 and 4, respectively, it is observed that the *FI*-computed SDs coincide with the corresponding FDs based on the same GDM.

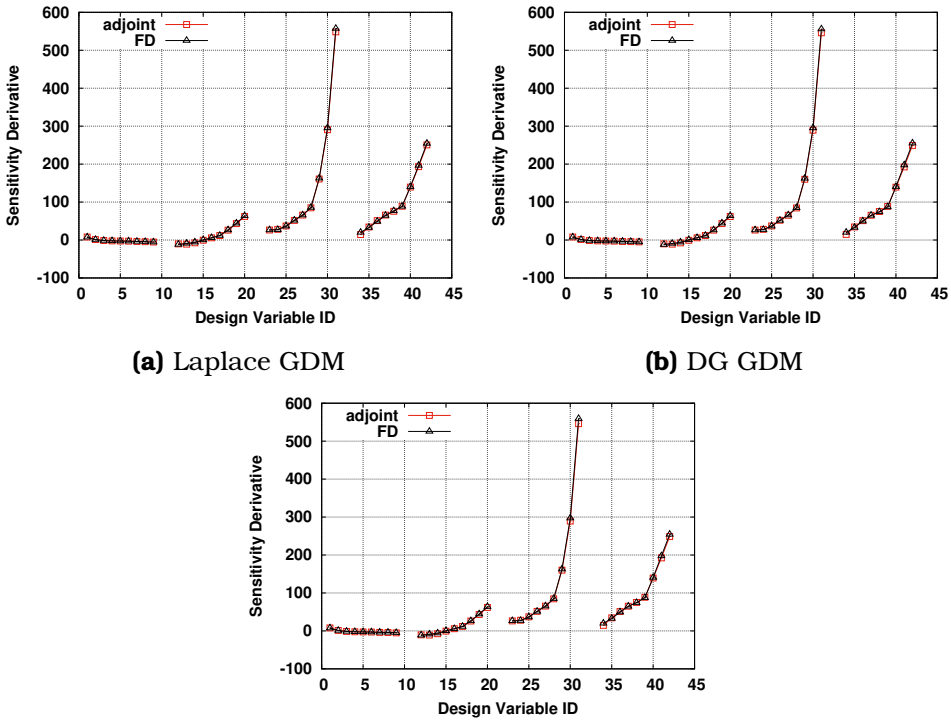
5.4.3 Scrutinizing Field Integrals of the Adjoint SDs

Considering eqs. 2.55 and 3.42, the terms that consider the effect of the chosen GDM are the field integrals including the grid sensitivities, i.e.

$$\begin{aligned} \mathcal{W}\left(\mathcal{E}\right) = &\int_{\Omega^F} \mathcal{F}_{jk}^F\left(\mathcal{E}\right) \frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega + \mathcal{E}\left[\int_{\Omega^S} \mathcal{F}_{jk}^S \frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega\right. \\ &\left. + \sum_{D=F,\mathcal{E}S} \int_{\Omega^D} j_\Omega^D \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega\right] \end{aligned} \tag{5.12}$$

where

$$\begin{aligned} \mathcal{F}_{jk}^F\left(\mathcal{E}\right) = &-u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q \frac{\partial v_j}{\partial x_k} + \Theta_{jk} + \mathcal{E}\left[-\rho^F c_p T_a^F v_j \frac{\partial T^F}{\partial x_k}\right. \\ &\left. - \rho^F T_a^F v_i v_j T_a^F \frac{\partial v_i}{\partial x_k} - k^F \frac{\partial T_a^F}{\partial x_j} \frac{\partial T^F}{\partial x_k} + T_a^F \frac{\partial}{\partial x_k}\left(k^F \frac{\partial T^F}{\partial x_j}\right)\right] \end{aligned} \tag{5.13a}$$



(a) Laplace GDM

(b) DG GDM

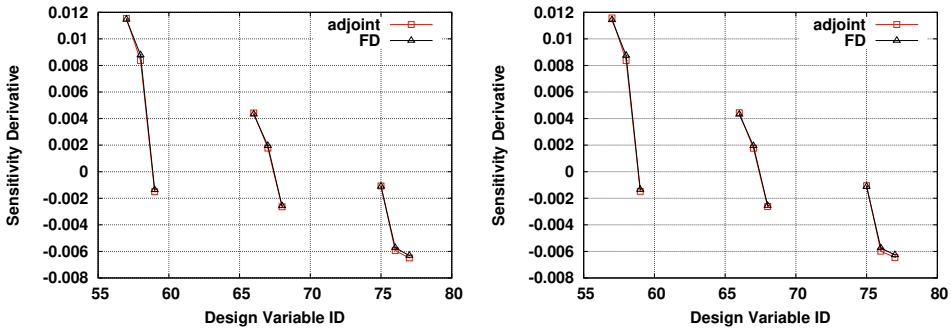
(c) IDW GDM

Figure 5.2: Comparison of FD-based and *FI* adjoint SDs, for Case 1; grid sensitivities used in the adjoint SDs are computed using the GDMs mentioned underneath the plots. Relative differences between the FD-based and adjoint SDs are on average below 5% (after excluding points with almost zero SDs).

$$\mathcal{F}^S_{jk} = -k^S \frac{\partial T^S_a}{\partial x_j} \frac{\partial T^S}{\partial x_k} + T^S_a \frac{\partial}{\partial x_k} \left(k^S \frac{\partial T^S}{\partial x_j} \right) \tag{5.13b}$$

In the above equations, $\mathcal{E}$ is a switch variable, introduced to distinguish between pure fluid flows ($\mathcal{E}=0$) and CHT problems ($\mathcal{E}=1$). The $\mathcal{W}$ term requires the computation of the grid sensitivities at the internal grid nodes, which is conducted through either analytical expressions (for the algebraic IDW, VBS and DG models) or by solving PDEs (for the Laplace GDM), see section 5.4.1. The cost of computing the grid sensitivities in such a way is proportional to N "equivalent" grid displacements.

In what follows, the $\mathcal{F}$ symbol corresponds to $\mathcal{F}^F$ and/or $\mathcal{F}^S$, see eqs. 5.13a and/or 5.13b, depending on the case. To understand the reasons for computing the same SDs for various GDMs, the $\mathcal{W}$ term, see eq. 5.12, including field integrals with the spatial gradients of the grid sensitivities is scrutinized. Note that the rest of the SD terms comprising the *FI*-computed SDs, see eqs. 2.55 and 3.42, consist only of surface integrals and are, hence, not affected by the GDM. In addition, the



(a) Laplace GDM **(b)** DG GDM

(c) IDW GDM **(d)** VBS GDM

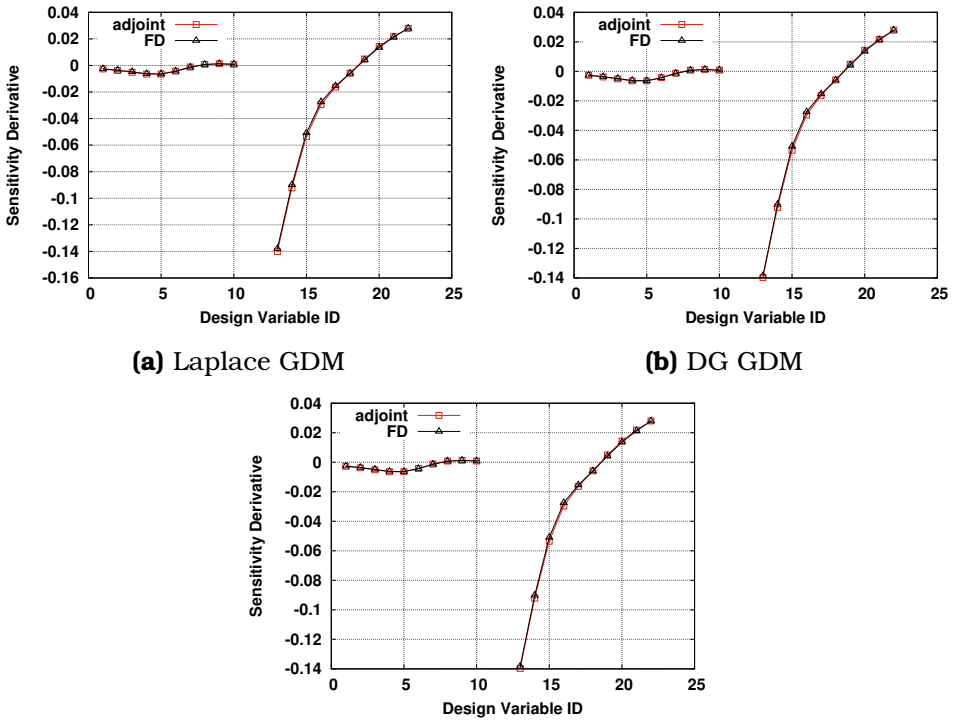
Figure 5.3: Comparison of FD-based and *FI* adjoint SDs, for Case 2; grid sensitivities used in the adjoint SDs are computed according to the GDMs mentioned underneath the plots. Relative differences between the FD-based and adjoint SDs are on average below 4.9%.

contribution of the $\mathcal{W}$ term to the SDs seems to be much more significant than that of the rest of the SDs terms, for almost all design variables, fig. 5.5.

5.4.4 Analysis of the $\mathcal{W}$ term

All field integrals within $\mathcal{W}$ include the product of terms depending on (a) the primal and adjoint fields which are independent of the grid sensitivities and (b) the spatial derivatives of the grid sensitivities. In this section, both of them are plotted, in an attempt to shed light into findings associated with fig. 5.1. Regarding figs. 5.6–5.16, studies focus on the y coordinates of the CPs, since their SDs have the largest (absolute) values.

In figs. 5.6 to 5.9, iso-areas of the magnitude of grid sensitivities for various GDMs over a part of the flow domain, along with their differences, are plotted for a selected CP in Cases 1,2 and 4. In all of them, close to the walls, $\delta x_i/\delta b_n$ seems to be very similar for all GDMs; in contrast, larger differences are observed far from the boundary. In figs. 5.10, 5.11 and 5.12, the magnitude of $\mathcal{F}$ is plotted



(a) Laplace GDM **(b)** DG GDM

(c) IDW GDM

Figure 5.4: Comparison of FD-based and *FI* adjoint SDs for Case 4; grid sensitivities used in the adjoint SDs are computed according to the GDMs mentioned underneath the plots. Relative differences between the FD-based and adjoint SDs are on average below 5.7%.

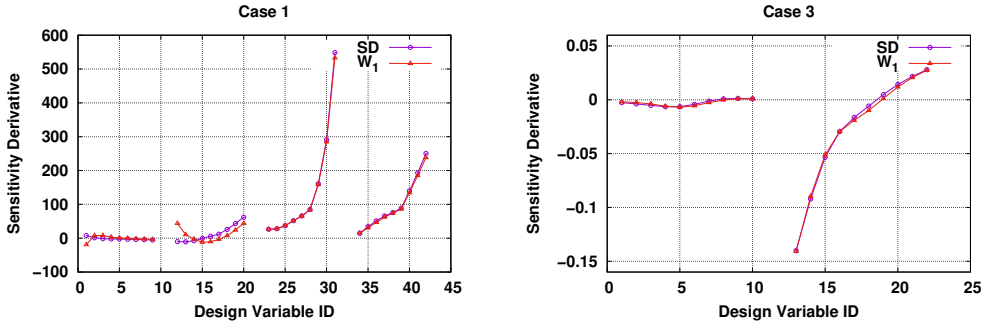


Figure 5.5: Comparison of the *FI* adjoint SD values and the $\mathcal{W}$ term, in Case 1 and Case 4. In Case 2 (omitted), SDs consist only of the $\mathcal{W}$ term (see Appendix B.2), so the two plotted curves coincide by definition. Since all GDMs lead to almost identical adjoint SD values, only those computed by the Laplace GDM are plotted.

for the three cases; in fig. 5.12, the multiplier of $\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)$ (in the last integral of $\mathcal{W}$ term, see eq. 5.12), corresponding to the objective function defined over Ω^S,

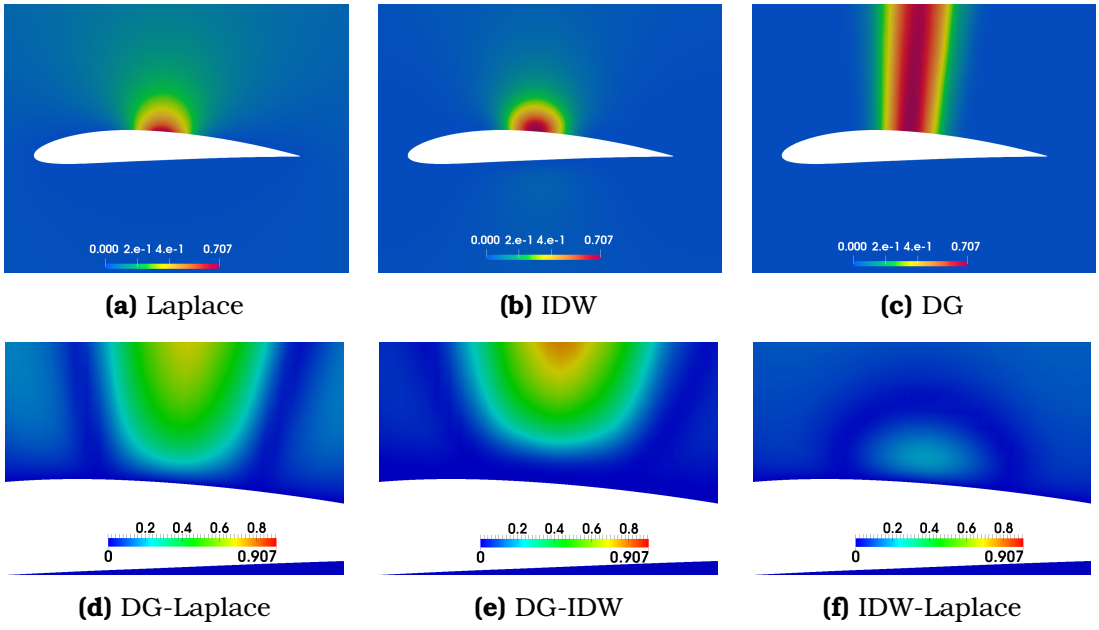
is also plotted for Case 4. Large $\mathcal{F}$ values appear close to the walls, where grid sensitivities are practically the same for all GDMs, and decay rapidly away from them. This means that large differences in grid sensitivities away from the walls are multiplied with an almost zero valued $\mathcal{F}$, leading to negligible contributions to the SDs from these areas. A similar behavior of the terms arising from the objective function used in Case 4, can be observed in fig. 5.12. This can also be seen by examining figs. 5.13, 5.14 and 5.16, which present the integrand of $\mathcal{W}$ at each cell centre, normalized by the $\mathcal{W}$ value, in $\%$. It can be observed that cells close to the parameterized walls contribute the most to $\mathcal{W}$. For instance, some cells next to the walls contribute up to almost 20% to its value. On the other hand, cells away from the walls contribute much less. It is thus clear that, for the cases examined, the highest contributions to the total SD values are close to the boundary. Moreover, in all cases, differences in the (normalized) integrands of $\mathcal{W}$ for the DG and IDW methods are minor. In figs. 5.13, 5.15 and 5.16, differences in the integrands of $\mathcal{W}$, computed for various GDMs and normalized by the total value of $\mathcal{W}$, in $\%$, are presented. Close to the walls, small areas of slight differences in $\mathcal{W}$ appear, which are of different sign and same magnitude and are practically cancelled out. Away from the walls, larger areas in which $\mathcal{W}$ is different appear; here, also, differences in integrands have different sign and same magnitude.

The findings justify the reasons for computing practically the same SDs with different GDMs, even when large differences in the computed grid sensitivities exist far from the wall boundaries. Conclusions drawn in this thesis in problems of different physics, are in agreement with [105, 163], in which the objective functions are forces exerted on isolated airfoils in compressible flows.

5.5 Effect of the Choice of the GDM on Shape Optimization

In the fourth and last step of the investigation, the aim is to carry out some optimization loops and demonstrate that differences in the adjoint SDs when assuming various inconsistent GDMs are so small that they do not, practically, affect the convergence of a shape optimization loop, when a Quasi-Newton method is used to update the design variables. The convergence of such methods is known to be more sensitive than Steepest Descent or the Conjugate Gradient method to the SD values, due to reasons explained in [29]. Here, the damped version of the BFGS method is used [113]. To do so, Case 2 is indicatively chosen among the three investigated cases to carry out shape optimization loops. Since, in this case, all four GDMs are used to compute the adjoint SDs, a wide range of comparisons between the performed optimizations can be made. Similar results are acquired in the other two cases and are omitted in the interest of space.

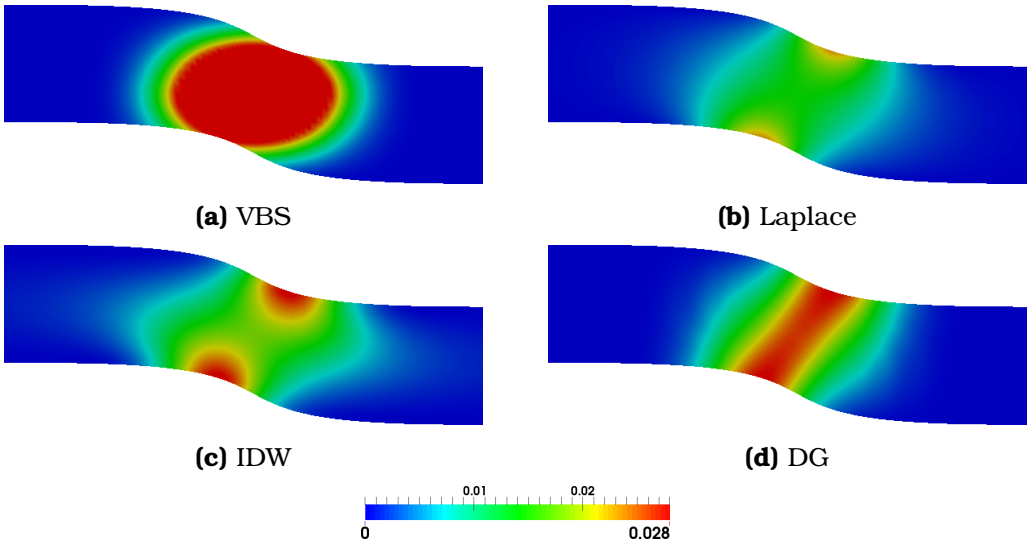
Three different GDMs, the VBS, DG and IDW ones, are used to deform the mesh. For each GDM, five optimizations are performed, in which adjoint SDs are computed by using either the *E-SI* or the *FI* methods; in the former, the adjoint



(a) Laplace **(b)** IDW **(c)** DG

(d) DG-Laplace **(e)** DG-IDW **(f)** IDW-Laplace

Figure 5.6: Case 1: First row: Magnitude of $\delta \mathbf{x}/\delta b_n$ computed using three GDMs. Second row: Difference in $\delta \mathbf{x}/\delta b_n$ magnitude between the GDMs used, normalized by the max. value of $\delta \mathbf{x}/\delta b_n$ computed by the parameterization along the whole boundary. Computations made for the sixteenth CP. Similar behavior (at least qualitatively) is exhibited for all CPs.

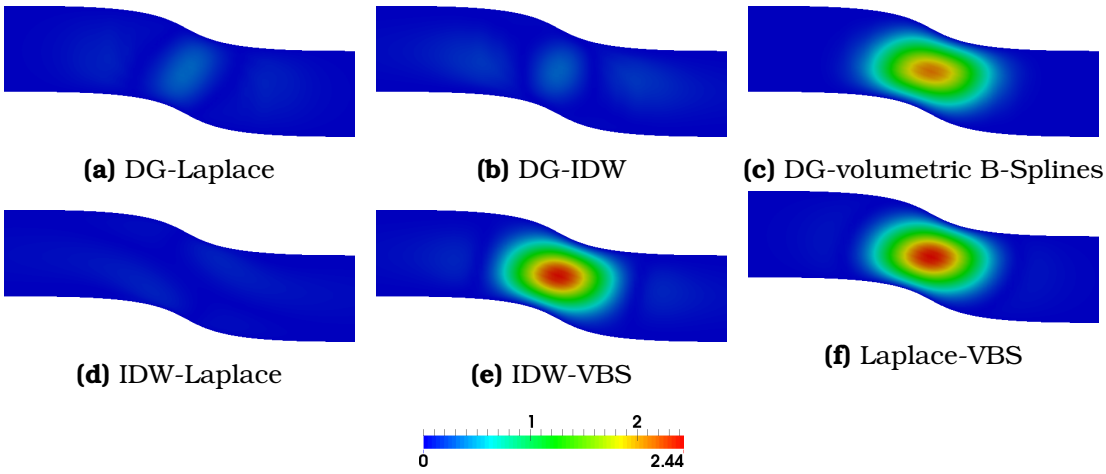


(a) VBS **(b)** Laplace

(c) IDW **(d)** DG

Figure 5.7: Case 2: Magnitude of $\delta \mathbf{x}/\delta b_n$ computed using four GDMs, for the fifth CP.

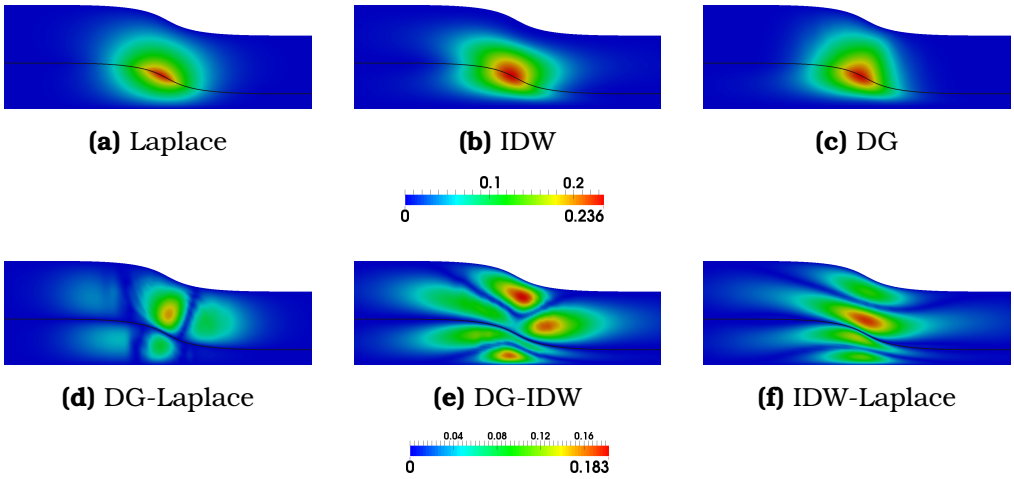
to Laplace GDM is used and in the latter, grid sensitivities are computed by assuming all four GDMs. In each optimization, the stopping criterion is set as



(a) DG-Laplace **(b)** DG-IDW **(c)** DG-volumetric B-Splines

(d) IDW-Laplace **(e)** IDW-VBS **(f)** Laplace-VBS

Figure 5.8: Case 2: Difference in $\delta \mathbf{x}/\delta b_n$ magnitude among the various GDMs, normalized by the max. value of $\delta \mathbf{x}/\delta b_n$ along the whole boundary. Comparisons made for the same CP as in fig. 5.7.



(a) Laplace **(b)** IDW **(c)** DG

(d) DG-Laplace **(e)** DG-IDW **(f)** IDW-Laplace

Figure 5.9: Case 4: First row: Magnitude of $\delta \mathbf{x}/\delta b_n$ computed using different GDMs, for the fifth CP. Second row: Difference in $\delta \mathbf{x}/\delta b_n$ magnitude among GDMs, normalized by the max. value of $\delta \mathbf{x}/\delta b_n$ along the whole boundary. The black line signifies the FSI boundary.

a lower than 0.01% difference in the objective function between two successive steps. Also, the update of the design variables is scaled by a coefficient η, the value of which is computed anew, in each optimization cycle, by a backtracking line-search algorithm satisfying the Armijo conditions [113]. The convergence histories of the optimizations are compared in fig. 5.17. For each one GDM used to deform the grid, convergence histories and optimized solutions are practically the same. When different GDMs are actually deforming the grid, in fig. 5.17d it is observed that negligible deviations in the convergence paths arise. In fig. 5.17e,

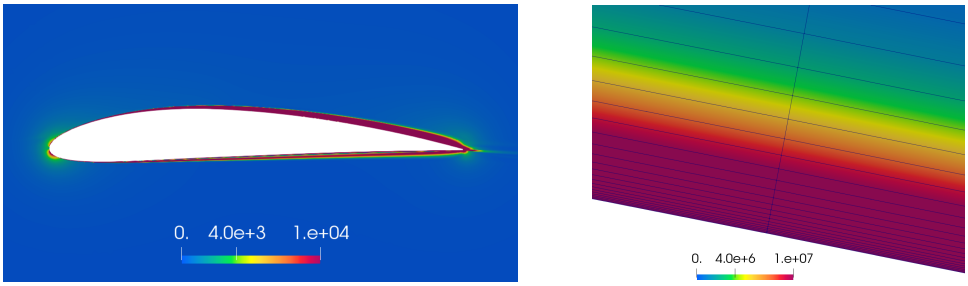


Figure 5.10: Case 1: Magnitude of $\mathcal{F}^F$ (left) and a close-up view (right) near the wall boundary (suction side; around 50% of chord).

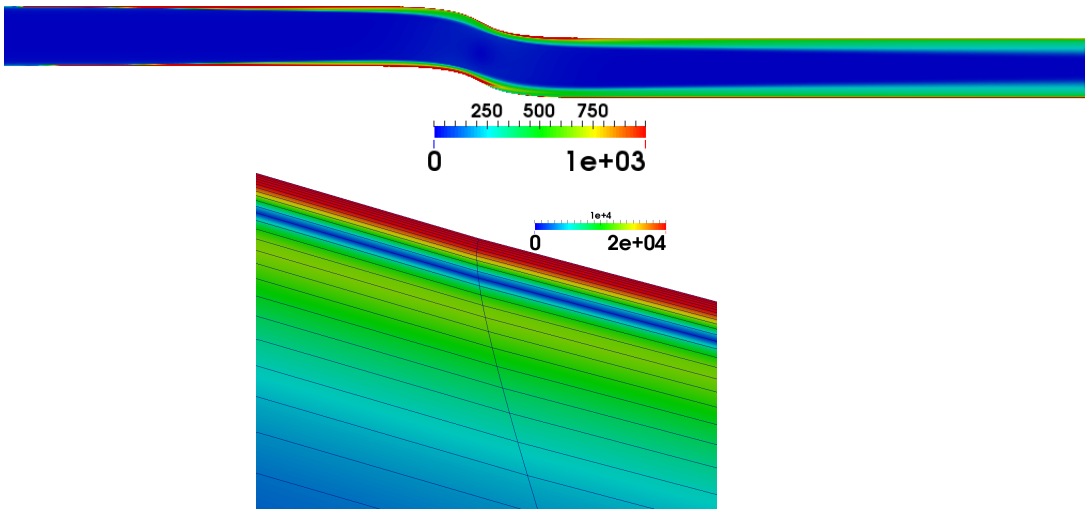


Figure 5.11: Case 2: Magnitude of $\mathcal{F}^F$ (top) and a close-up view of its distribution near the lower wall (bottom).

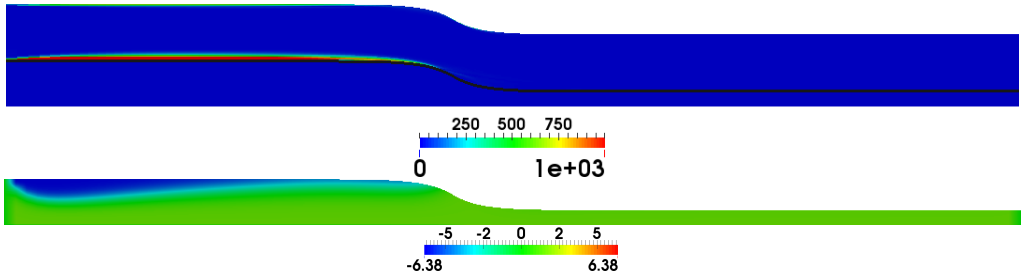
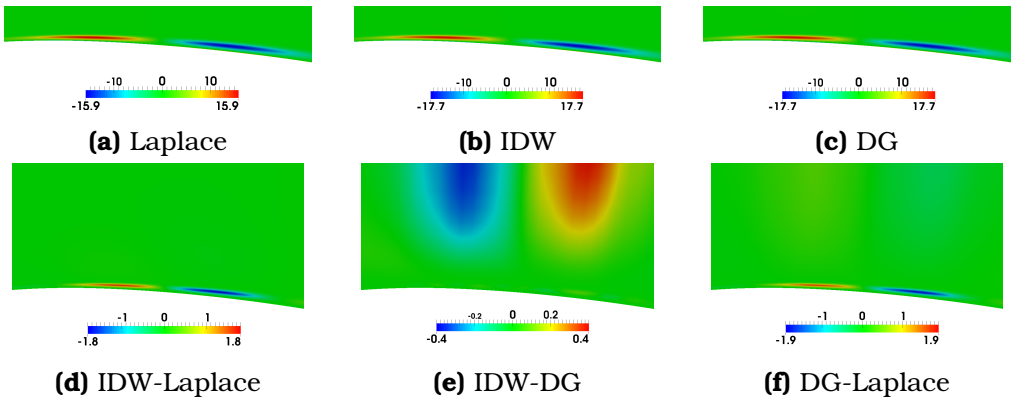


Figure 5.12: Case 4: Magnitude of $\mathcal{F}^F$, $\mathcal{F}^S$ (top) and distribution of $\frac{1}{V_S}\left(j^S - J^S\right)$ in Ω^S (bottom); the latter is included in the last integral on the r.h.s. of $\mathcal{W}$.

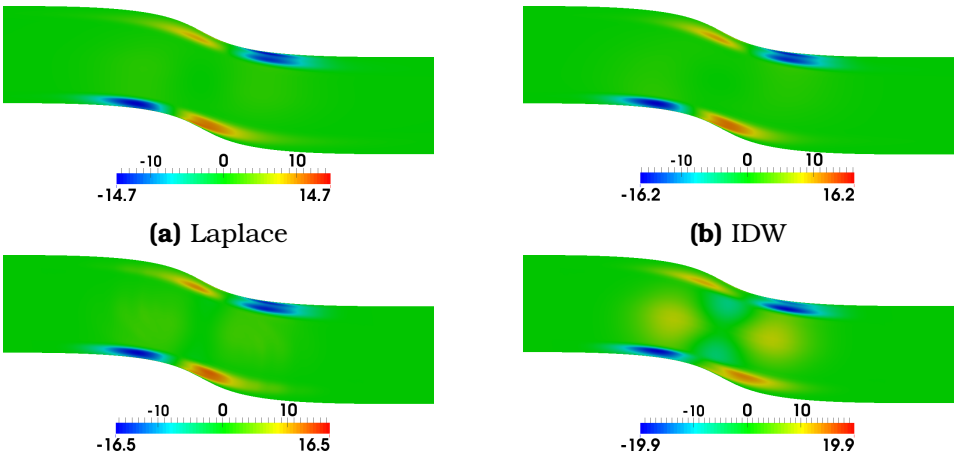
a comparison of some representative optimized geometries is made; optimizations with different SD-GDM pairs not included in the figure in the sake of clarity resulted to the same findings. It can be observed that, despite the use of different GDMs, the optimized geometries are almost identical. The optimized shape, with



(a) Laplace **(b)** IDW **(c)** DG

(d) IDW-Laplace **(e)** IDW-DG **(f)** DG-Laplace

Figure 5.13: Case 1: First row: Integrand of $\mathcal{W}$, computed for different GDMs, normalized by the $\mathcal{W}$ value, in $\%$, for the sixteenth CP. Second row: Difference, in $\%$, among the integrand of $\mathcal{W}$ computed for different GDMs, normalized by the $\mathcal{W}$ value.

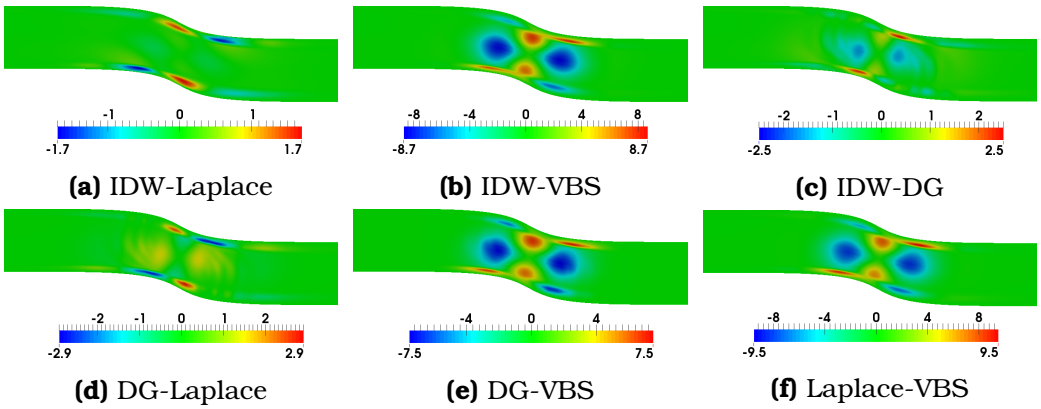


(a) Laplace **(b)** IDW

(c) DG **(d)** VBS

Figure 5.14: Case 2: Integrand of $\mathcal{W}$, computed using different GDMs, normalized by the $\mathcal{W}$ value, in $\%$, for the fifth CP.

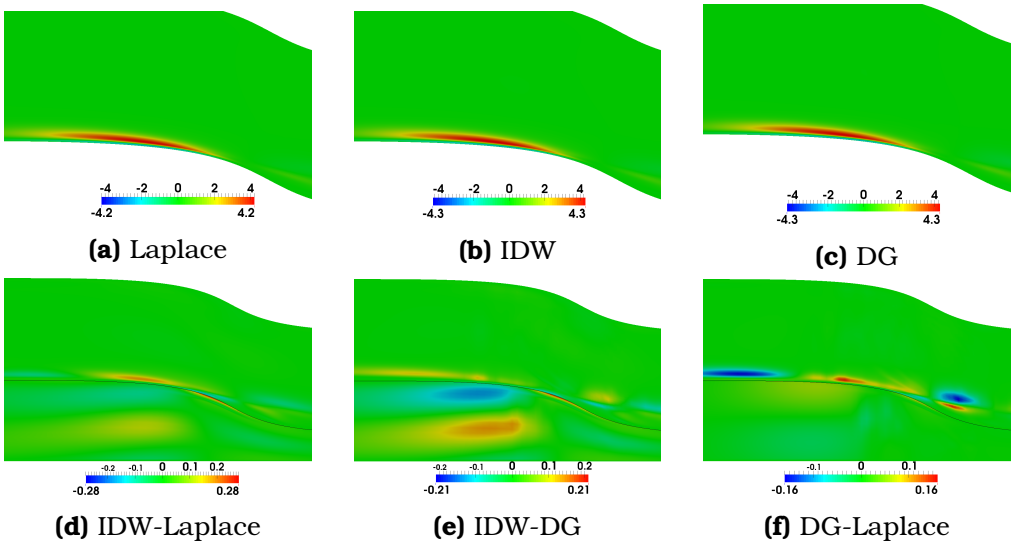
$\sim 8.5\%$ lower value of J^{pt} (expressed by eq. B.3), is obtained after 8 optimization cycles. A comparison between the initial and optimized geometries and the velocity magnitudes over them is made in section 2.4. It is concluded that, in the examined case, differences between the computed SDs are small enough, so that the same number of cycles is needed in each optimization to meet the above-mentioned convergence criterion.



(a) IDW-Laplace **(b)** IDW-VBS **(c)** IDW-DG

(d) DG-Laplace **(e)** DG-VBS **(f)** Laplace-VBS

Figure 5.15: Case 2: Difference, in $\%$, among the integrand of $\mathcal{W}$ computed for different GDMs, normalized by the $\mathcal{W}$ value.



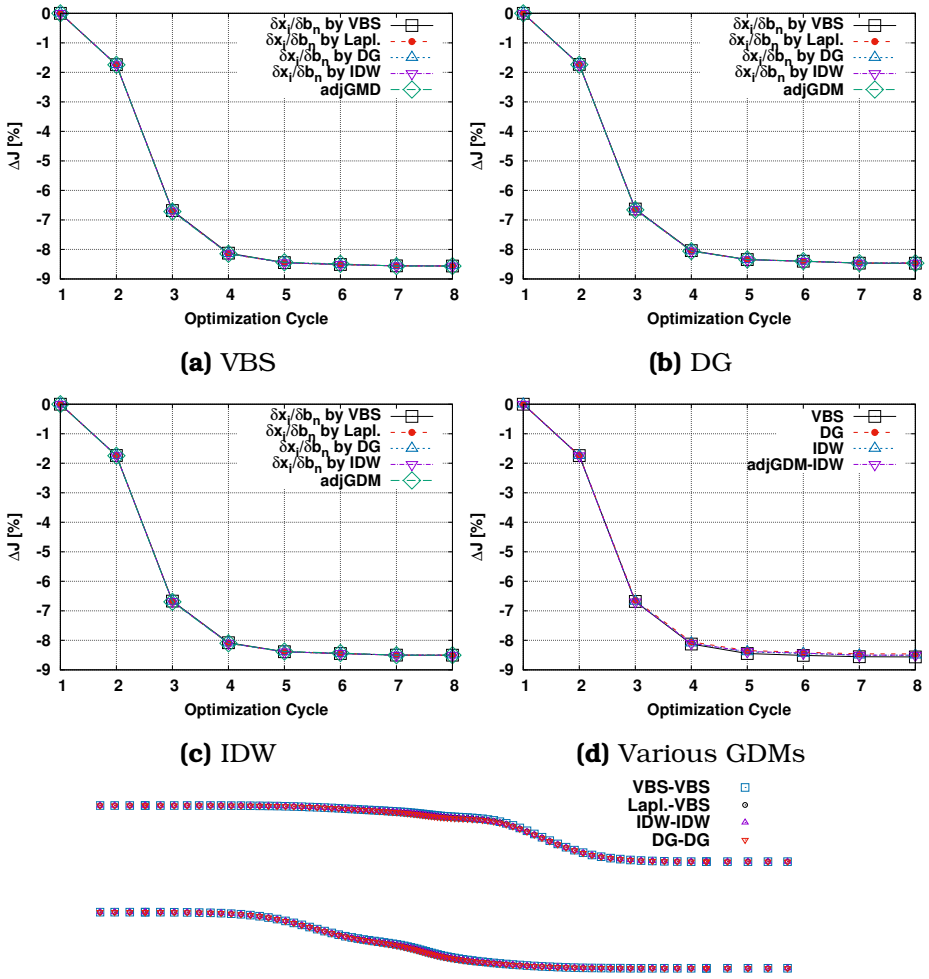
(a) Laplace **(b)** IDW **(c)** DG

(d) IDW-Laplace **(e)** IDW-DG **(f)** DG-Laplace

Figure 5.16: Case 4: First row: A close-up view of the integrand of $\mathcal{W}$, computed for different GDMs, normalized by the $\mathcal{W}$ value, in $\%$, for the fifth CP. Fields are plotted only in Ω^F, since the equivalent in Ω^S are much smaller. Second row: Difference, in $\%$, among the integrand of $\mathcal{W}$ computed for different GDMs, normalized by the $\mathcal{W}$ value, for both Ω^F and Ω^S.

5.6 Conclusions

This chapter focused on whether the choice of the grid displacement model in an optimization loop affects the SDs of the objective function and, through them, the outcome of the optimization. Investigations initially presented in [163, 105] were extended and enriched by focusing on (a) incompressible fluid flows, including internal and external aerodynamics and Conjugate Heat Transfer problems and



(a) VBS **(b)** DG

(c) IDW **(d)** Various GDMs

(e) Optimized geometries

Figure 5.17: Case 2: (a to c) Comparisons of J^{pt} convergence during optimizations using the GDMs mentioned underneath the plots to deform the mesh. In all cases, SDs are computed by both the *E-SI* (denoted as adjGDM) and *FI*; for the latter, grid sensitivities are computed by assuming all four GDMs in the adjoint method. (d): Comparison of the J^{pt} convergence during optimizations using different GDMs. In convergence paths denoted as VBS, DG and IDW, grid sensitivities are computed by means of the GDM used to deform the grid. Regarding the convergence path denoted as "adjGDM-IDW", *E-SI* adjoint SDs are computed and the IDW is the actual GDM in use. (e): Comparison of optimized geometries; the first component in the legends denotes which GDM is considered in the adjoint SDs, while the second which GDM is used to deform the mesh.

(b) the use of continuous adjoint. For this investigation, first FDs were used, in which the grid displacement was carried out by means of different GDMs, based on Laplace PDEs, a VBS morpher, the Delaunay Graph and the IDW model. This

study yielded very similar SDs for three different cases, regarding internal and external pure fluid flow problems and a CHT one. Since FDs stand for the reference method to compute SDs, results are equally significant for both continuous and discrete adjoint methods. An investigation on shape optimizations demonstrated that differences in the adjoint SDs were indeed small enough that even the BFGS method, relying on approximated Hessian matrices, is not affected throughout the optimization. A term-by-term analysis of the *FI*-based SDs revealed that, close to the parameterized/changing boundaries: (a) grid sensitivities are practically the same, for all GDMs and (b) the term (that depends on primal and adjoint fields) which multiplies the grid sensitivities inside field integrals takes on its highest values; that term decays rapidly away from the wall. This leads to almost identical SD values for all GDMs, despite large differences in $\delta x_i/\delta b_n$, for some of them, away from the walls. Based on these findings, it was concluded that the choice of the GDM does not affect the SD values, at least for the cases investigated in this chapter. As a result, the *E-SI* continuous adjoint which yields SDs free of field integrals of grid sensitivities by making the convenient assumption of a Laplace GDM, can safely be used to compute SDs at low cost, irrespective of the GDM used in the optimization.

Chapter 6

Adjoint-assisted Pareto Front Tracing in Aerodynamic and CHT Shape Optimization

In this chapter, the adjoint method is used to assist gradient-based methods tracing the Pareto front in multi-objective optimization (MOO) problems. The developed methods are conceptually based on a prediction-correction algorithm first presented in [148] and further studied by the author's group in [67]. This algorithm starts from an already computed Pareto Point (PP). Targets, corresponding to the values, on the next PP, of all but one of the contradictory goals of the MOO problem, are set. Then, by solving a constrained optimization problem, a new PP is computed. The prediction step is used to speed up the process of moving between two PPs. In the prediction step, based on the information for an already available PP, each new one is extrapolated. The extrapolated point in the objective space does not usually belong to the Pareto front. To bring the extrapolated point on the Pareto front, a correction step, aiming to satisfy the Karush-Kuhn-Tucker (KKT) first-order optimality conditions, is then used.

Three variants of the prediction-correction scheme are assessed, looking for the least expensive one. In all variants, the exact computation of the second derivatives of the Lagrangian w.r.t. b_n, i.e. the Hessian matrix, is avoided, since that would make the computation of the Pareto front unaffordable. In brief, the three evaluated schemes are:

1. Variant 1 in which, in the prediction step, the system of linear equations updating design variables and Lagrange multipliers is solved iteratively, using the linear-Restarted GMRES method [144]. The necessary Hessian-vector products result from the Direct Differentiation of the primal and adjoint equations. Such an approach is inspired from the truncated Newton method for single-objective aerodynamic shape optimization problems, proposed by the authors' group in [42, 122]. The correction step follows, by repetitively approximating the Hessian matrix through the damped BFGS method [113]; the latter requires the gradients of the objective and constraint functions

w.r.t. the design variables, computed by the continuous adjoint method.

2. Variant 2 which applies the BFGS method to approximate the Hessian in both the prediction and correction steps.

3. Variant 3 which omits the prediction step and solves only the system of equations corresponding to what is referred to as the "correction" step, to find the location of the new PP.

As seen above, the correction step is the same in all three variants. These are initially used to solve optimization problems pertaining to the drag minimization and lift maximization of two isolated airfoils. The less expensive variant among them is then used to compute the Pareto front in a 3D Conjugate Heat Transfer (CHT) problem. The minimization of the highest temperature in a solid device equipped with cooling fins to enhance heat transfer, and the total pressure losses of the air passing though the fins are the two objectives. Both laminar (isolated airfoil problems) and turbulent (in the CHT problem) flows are encountered.

In section 6.1, the systems of equations solved to update the design variables during the prediction and correction steps are discussed, followed by their combinations that form the proposed variants of the Pareto tracing algorithm, in section 6.2. In sections 6.3 and 6.4, the primal and adjoint methods for incompressible flows and the computation of matrix-vector products are presented. In section 6.5, the three variants are compared in external aerodynamics and, then, the one with the lowest computational cost is used to solve a 3D CHT problem. A comparison between the variant with the lowest computational cost and the weighted sum approach, is also made.

6.1 Mathematical Background

The presentation is limited to the minimization of two objective functions, J_1 and J_2, controlled by the design variables $b_n, n \in [1, N]$, subjected to an equality constraint ($c_g = 0$); a discussion on how to handle problems with more objective functions and/or constraints is deferred to the next section.

The process of computing each new PP is initialized by an already available one, denoted by $(J_1,\ J_2) = \left(J_1^{old},\ \hat{J}_2^{\,old}\right)$; the first PP is obtained by minimizing one objective (J_1, in all cases presented here) while satisfying the equality constraint $c_g = 0$. To move from one PP to the next, J_1 is minimized by simultaneously imposing an equality constraint $c_f = J_2 - \hat{J}_2^{\,new} = 0$, where $\hat{J}_2^{\,new}$ is a target value. Hence, the Lagrangian L becomes

$$L\left(b_n, \lambda_1, \lambda_2, \hat{J}_2\right) = J_1\left(b_n\right) - \lambda_1 c_f\left(b_n, \hat{J}_2\right) - \lambda_2 c_g\left(b_n\right) \tag{6.1}$$

where λ_1, λ_2 are Lagrange multipliers. To compute a new PP, the KKT first-order optimality conditions [113], expressed in an array form with $N+2$ equations,

$$\mathbf{K}\left(\mathbf{z}, \hat{J}_2\right) = \begin{bmatrix} \frac{\partial L}{\partial \mathbf{b}}\Big|_{\lambda_i, \hat{J}_2 = ct} \\ c_f \\ c_g \end{bmatrix} = \mathbf{0} \tag{6.2}$$

where $\mathbf{z} = [\mathbf{b}\ \boldsymbol{\lambda}]^T$, should be satisfied.

A standard way to solve the constrained optimization problem described by eq. 6.2 is through the SQP method [113]. By linearizing eqs. 6.2 w.r.t. $\mathbf{z}$, $K_i\left(\mathbf{z}^{new}\right) = K_i\left(\mathbf{z}^{old}\right) + \frac{\delta K_i}{\delta z_j}\Big|_{z_j^{old}} \Delta z_j = 0$, the following system of equations

$$\underbrace{\begin{bmatrix} \boldsymbol{\mathcal{H}} & -\frac{\delta c_f}{\delta \mathbf{b}}^T & -\frac{\delta c_g}{\delta \mathbf{b}}^T \\ \frac{\delta c_f}{\delta \mathbf{b}} & 0 & 0 \\ \frac{\delta c_g}{\delta \mathbf{b}} & 0 & 0 \end{bmatrix}_{old}}_{M} \begin{bmatrix} \boldsymbol{\Delta b} \\ \Delta\lambda_1 \\ \Delta\lambda_2 \end{bmatrix} = - \begin{bmatrix} \frac{\delta L}{\delta \mathbf{b}} \\ c_f \\ c_g \end{bmatrix}_{old} \tag{6.3}$$

must be solved, where $\boldsymbol{\mathcal{H}}$ is the Hessian matrix of L w.r.t. $\mathbf{b}$. Then, $\mathbf{b}$ and $\boldsymbol{\lambda}$ are updated as

$$\begin{aligned} \mathbf{b}^{new} &= \mathbf{b}^{old} + \alpha \boldsymbol{\Delta b}, \\ \boldsymbol{\lambda}^{new} &= \boldsymbol{\lambda}^{old} + \boldsymbol{\Delta \lambda} \end{aligned} \tag{6.4}$$

where α is either user-defined or computed by means of line-search (see next section). The successive application of eqs. 6.3 and 6.4 leads to a new PP.

A different way to move from one PP to the next emerges by considering that the KKT conditions are satisfied at both PPs. Since $\hat{J}_2$ can be seen as the parameter that controls the position on the Pareto front [148], along which $\frac{\delta K_i}{\delta \hat{J}_2}\Big|_{\hat{J}_2{}^{old}} = \frac{\partial K_i}{\partial \hat{J}_2}\Big|_{\hat{J}_2{}^{old}} + \frac{\partial K_i}{\partial z_j}\frac{\delta z_j}{\delta \hat{J}_2}\Big|_{\hat{J}_2{}^{old}} = 0$. This leads to the following system of equations [148]

$$\underbrace{\begin{bmatrix} \boldsymbol{\mathcal{H}} & -\frac{\delta c_f}{\delta \mathbf{b}}^T & -\frac{\delta c_g}{\delta \mathbf{b}}^T \\ \frac{\delta c_f}{\delta \mathbf{b}} & 0 & 0 \\ \frac{\delta c_g}{\delta \mathbf{b}} & 0 & 0 \end{bmatrix}_{old}}_{M = \frac{\partial \mathbf{K}}{\partial \mathbf{z}}\Big|_{\hat{J}_2{}^{old}}} \underbrace{\begin{bmatrix} \frac{\delta \mathbf{b}}{\delta \hat{J}_2} \\ \frac{\delta \lambda_1}{\delta \hat{J}_2} \\ \frac{\delta \lambda_2}{\delta \hat{J}_2} \end{bmatrix}}_{X = \frac{\delta \mathbf{z}}{\delta \hat{J}_2}\Big|_{\hat{J}_2{}^{old}}} = \underbrace{\begin{bmatrix} \mathbf{0} \\ 1 \\ 0 \end{bmatrix}}_{B = -\frac{\partial \mathbf{K}}{\partial \hat{J}_2}\Big|_{\hat{J}_2{}^{old}}} \tag{6.5}$$

After solving eq. 6.5, $\mathbf{b}$ and $\boldsymbol{\lambda}$ are updated as

$$\begin{aligned} \mathbf{b}^{new} &= \mathbf{b}^{old} + \alpha \frac{\delta \mathbf{b}}{\delta \hat{J}_2} \left(\hat{J}_2{}^{new} - \hat{J}_2{}^{old} \right), \\ \boldsymbol{\lambda}^{new} &= \boldsymbol{\lambda}^{old} + \frac{\delta \boldsymbol{\lambda}}{\delta \hat{J}_2} \left(\hat{J}_2{}^{new} - \hat{J}_2{}^{old} \right) \end{aligned} \tag{6.6}$$

Note that, in eqs. 6.5, the l.h.s. coefficient matrix is the same as in eqs. 6.3. A key feature of any iterative method that makes use of eqs. 6.5 and 6.6 is that the KKT conditions must be satisfied at the starting point, within each step, which is not a requirement for any solver based on eqs. 6.3 and 6.4.

So far, two different methods, one based on eqs. 6.3 and 6.4 and the other on eqs. 6.5 and 6.6 have been presented, to move between two PPs. In the next section, these will be combined into three algorithmic variants, able to trace the Pareto front.

6.2 Pareto-Front Tracing Algorithms

To move from one PP to the next, it is possible to either use the SQP method (eqs. 6.3 and 6.4) or solve eqs. 6.5 and update $\mathbf{b}$ and $\boldsymbol{\lambda}$ through 6.6. In both cases, a single iteration leads to a point that does not generally belong to the Pareto front, due to non-linearities. Since the KKT conditions, eqs. 6.2, are not valid at that intermediate point, eqs. 6.5 cannot be solved there. Therefore, the SQP method is used to move from the intermediate point back to the Pareto front. The first step based on eqs. 6.5 and 6.6 performs the prediction, whereas successive solutions of eqs. 6.3 and 6.4 undertake the corrections.

Solving constrained SOO problems exclusively by means of the SQP method, i.e. without the prediction step, could certainly be a stand-alone procedure to compute the Pareto front. Thus, to investigate computational benefits when using the prediction step, three variants are programmed and compared. In all of them, the exact computation of the Hessian matrix is avoided, as its cost would be $N+1$ EFS, [121]; Hessian-vector products and approximations of the Hessian matrix are used instead. The three schemes devised to compute the Pareto front are (see also fig. 6.1):

- Variant 1: In the prediction step, eqs. 6.5 are solved by using the linear-Restarted GMRES method [144], to compute $\delta\mathbf{b}/\delta\hat{J}_2$ and $\delta\boldsymbol{\lambda}/\delta\hat{J}_2$ and update $\mathbf{b}$ and $\boldsymbol{\lambda}$ through eqs. 6.6. The necessary Hessian-vector products result from the Direct Differentiation of the primal and adjoint equations, at a cost of 2 EFS. The correction step is iteratively carried out by means of eqs. 6.3 and 6.4 and the required Hessian matrix is approximated through

the damped BFGS method [113]; the latter requires the gradients of the objective and constraint functions w.r.t. the design variables, computed using the continuous adjoint method, see section 6.3.

- Variant 2: The damped BFGS method provides approximations to the Hessian for use in both the prediction and correction steps.

- Variant 3: The prediction step is omitted and only the system of equations corresponding to what is referred to as the "correction" step in the previous two variants is utilized, i.e. eqs. 6.3, are solved and, then, eqs. 6.4 are used to update $\mathbf{b}$ and $\boldsymbol{\lambda}$. From a certain point of view, Variant 3 presents some similarities with the Normal-Boundary Intersection (NBI) method presented in [23]. Both approaches solve constrained optimization problems to compute the next PP, however the way of choosing the constraint values differs slightly. In NBI, the extreme PPs are computed first and the values of the objective(s) acting as constraints are, then, equi-distributed between them in the objective space. In contrast, Variant 3 computes only one extreme PP and updates the constraint values by a user-defined step size.

It can be observed that the correction step is the same in all variants. After computing a new PP, $\hat{J}_2$ is updated and the same procedure is repeated. The computation of the new PP terminates when the relative change in J_1 between two consecutive optimization cycles is less than a user-defined threshold ϵ_1 and, also, the absolute values of $c_f/\hat{J}_2$ and $c_g/\widehat{\Omega}$ ($\widehat{\Omega}$ is defined in section 6.3) become less than thresholds ϵ_2 and ϵ_3. A maximum number N_{corr} of allowed correction steps per PP computation, the desired number N_{PP} of PPs and $\Delta\hat{J}_2$ are also defined by the user.

In eqs. 6.4 and 6.6, the resulting Δb_n is optionally scaled by a step length α, computed by the Armijo backtracking line-search algorithm [113]; α should satisfy the Armijo condition

$$\phi^{new} < \phi^{old} + c_1 \alpha D\phi \Delta b_n \tag{6.7}$$

In eq. 6.7, $\phi = J_1 + \mu\left(|c_f| + |c_g|\right)$, μ is equal to $max\left(|\lambda_1|, |\lambda_2|\right) + 2\delta$ when $\mu < max\left(|\lambda_1|, |\lambda_2|\right) + \delta$, with δ, c_1 positive user-defined constants and $D\phi\Delta b_n = \frac{\delta J_1}{\delta b_n}\Delta b_n - \mu\left(|c_f| + |c_g|\right)$. For J_1, c_f and c_g known, starting from $a = 1$, a is regularly updated by fitting a quadratic polynomial of α to ϕ, the shape and mesh are updated, the primal equations are solved and J_1, c_f and c_g are re-evaluated until eq. 6.7 is satisfied.

In cases with more than two objectives, similar systems of equations are solved to trace the Pareto front, requiring though an algorithm providing the target values ($\hat{J}_2$), so that the front becomes well-populated and the overall computational cost remains as low as possible.

A SOO problem is solved, starting from the baseline geometry, in order to compute the first PP, (J_1^0, J_2^0), for which the constraint, c_g, is satisfied; set $\hat{J}_2^0 = J_2^0$.
for $k = 1, 2, \ldots, N_{PP}$ **do**
  Update $\hat{J}_2^k$.
  Solve the adjoint PDEs to compute $\delta J_1/\delta b_n,\ \delta c_f/\delta b_n$; compute $\delta c_g/\delta b_n$ analytically.
  if *Variant*$= 1$ *or 2* **then**
    Update $\mathbf{b}$ and $\boldsymbol{\lambda}$ through eqs. 6.5 and 6.6 (Prediction step) by optionally using line-search.
  else
    Update $\mathbf{b}$ and $\boldsymbol{\lambda}$ through eqs. 6.3 (SQP; optional use of line-search).
  end
  Update the shape/grid.
  Solve the primal equations; evaluate $\left(J_1^{k;0}, J_2^{k;0}\right)$, c_f and c_g.
  Set $j = 0$ and $J_1^{k;j-1} = J_1^{k-1}$
  while $(j < N_{corr})$ *and* $(\,|J_1^{k;j} - J_1^{k;j-1}|/|J_1^{k;j-1}| > \epsilon_1$ *or* $|c_f/\hat{J}_2| > \epsilon_2$ *or* $|c_g/\widehat{\Omega}| > \epsilon_3\,)$ **do**
    Solve the adjoint equations to compute $\delta J_1/\delta b_n,\ \delta c_f/\delta b_n$; compute $\delta c_g/\delta b_n$ analytically.
    Update $\mathbf{b}$ and $\boldsymbol{\lambda}$ through eqs. 6.3 (SQP; optional use of line-search).
    Update the shape/grid and set $j \leftarrow j+1$
    Solve the primal equations; evaluate $\left(J_1^{k;j}, J_2^{k;j}\right)$, c_f and c_g.
  end
  New PP $\left(J_1^k, J_2^k\right)$ computed.
end

Figure 6.1: Two-objective optimization algorithm, used to compute N_{PP} points on the front. Indices k and j denote the ID of the PP and the counter for the intermediate steps between two PPs, respectively.

6.3 Primal and Adjoint Equations in External Aerodynamics

To demonstrate the developed/programmed Pareto tracing method, without loss of generality, isolated airfoil cases aiming at min. drag (J_1) and max. lift (J_2) are used. Additionally, $c_g = \int_\Omega d\Omega - \widehat{\Omega} = 0$ must be satisfied throughout the optimization loops, with Ω being the volume of the computational domain and $\widehat{\Omega}$ a target value; in the examined cases, $\widehat{\Omega}$ is set equal to the volume of the initial domain. This constraint is imposed to maintain the airfoil area constant. The primal equations

for laminar flows are given by eq. 2.1a and 2.1b. The gradient of J^{forces} w.r.t. b_n is given by eq. B.1. and is used for both $\delta J_1/\delta b_n$ and $\delta J_2/\delta b_n$. Also, $\frac{\delta c_f}{\delta b_n} = \frac{\delta J_2}{\delta b_n}$ and the gradient of c_g, given by an analytical expression, reads $\frac{\delta c_g}{\delta b_n} = \int_\Omega \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega$.

By taking advantage of the linearity of the adjoint PDEs, only a single system of adjoint PDEs could be solved, for computing $\delta L/\delta b_n$. However, since $\delta J_1/\delta b_n$ and $\delta c_f/\delta b_n$ are needed separately by eqs. 6.3 and 6.5 separately, the cost of computing the adjoint sensitivities becomes equal to 2 EFS per optimization cycle. The final expression of $\delta L/\delta b_n$ is

$$\frac{\delta L}{\delta b_n} = \frac{\delta J_1}{\delta b_n} - \lambda_1 \frac{\delta J_2}{\delta b_n} - \lambda_2 \frac{\delta c_g}{\delta b_n} \tag{6.8}$$

Here, the volumetric B-Splines morpher [129] given by eq. 5.4 is used to parameterize the geometries and the surrounding grid; grid sensitivities are computed by eq. 5.11.

6.4 Matrix-Vector Product Computation

In the prediction step, eqs. 6.5 can be solved iteratively using the linear-Restarted GMRES method [144] which requires only matrix-vector products. This approach

$\mathbf{r^0} = M\mathbf{X}^0 - \mathbf{B}$, $\mathbf{s^1} = \frac{\mathbf{r^0}}{\|\mathbf{r^0}\|_2}$
for $j = 1, 2, \ldots, \mathcal{M}$ **do**
  $\mathbf{w^j} = M\mathbf{s^j}$
  for $i = 1, 2, \ldots, j$ **do**
    $h_{i,j} = \mathbf{w^j} \cdot \mathbf{s^i}$
    $\mathbf{s^{j+1}} = \mathbf{w^j} - \sum_{i=1}^{j} h_{i,j}\mathbf{s^i}$
  end
  $h_{j+1,j} = \|\mathbf{s^{j+1}}\|_2$
  $\mathbf{s^{j+1}} = \frac{\mathbf{s^{j+1}}}{h_{j+1,j}}$
end
Compute $\beta_1, \ldots, \beta_{\mathcal{M}}$ by solving the minimization problem
  $min\|\|\mathbf{r^0}\|_2\mathbf{e_1} - \overline{H}\boldsymbol{\beta}\|_2$
$\mathbf{X}^{\mathcal{M}} = \mathbf{X}^0 + \sum_{i=1}^{\mathcal{M}} \beta_i \mathbf{s^i}$

Figure 6.2: The Linear Restarted GMRES method, used to solve eqs. 6.5 $M\mathbf{X} = \mathbf{B}$. $\mathcal{M}$ stands for the number of GMRES bases $\mathbf{s^i}$, $\mathbf{e_1} = (1, 0, ..., 0)$ is of size $\mathcal{M} + 1$ and $\overline{H}_{ij} = h_{i,j}$.

overcomes the computation of the exact Hessian matrix included in matrix M. The multiplication of M with the vector $\mathbf{s}$ required by the GMRES method, see fig. 6.2, reads

$$M\mathbf{s} = \begin{bmatrix} \mathbf{P} \\ \sum_{m=1}^{N} \frac{\delta c_f}{\delta b_m} s_m \\ \sum_{m=1}^{N} \frac{\delta c_g}{\delta b_m} s_m \end{bmatrix} \tag{6.9}$$

where $\mathbf{P}$ is the product of the first N rows of matrix M with the first N components of vector $\mathbf{s}$, reading

$$P_n = \sum_{m=1}^{N} \frac{\delta^2 L}{\delta b_n \delta b_m} s_m - \frac{\delta c_f}{\delta b_n} s_{N+1} - \frac{\delta c_g}{\delta b_n} s_{N+2}, \; n = 1, ..., N \tag{6.10}$$

The first term on the r.h.s. of eq. 6.10 is the Hessian-vector product of the Lagrangian, given by

$$\frac{\delta^2 L}{\delta b_n \delta b_m} s_m = \frac{\delta^2 J_1}{\delta b_n \delta b_m} s_m - \lambda_1 \frac{\delta^2 J_2}{\delta b_n \delta b_m} s_m - \lambda_2 \frac{\delta^2 c_g}{\delta b_n \delta b_m} s_m, \; n = 1, ..., N \tag{6.11}$$

For any field Φ, let $\overline{\Phi}$ be defined as

$$\overline{\Phi} = \frac{\delta \Phi}{\delta b_m} s_m \tag{6.12}$$

In eq. B.1, $j_{S,i} = \left(p\delta_i^j - \tau_{ij}\right) r_i$ and thus,

$$\begin{aligned} \overline{j_{S,i}} &= \left(\overline{p}\delta_i^j - \overline{\tau_{ij}}\right) r_i \\ &= \left[\overline{p}\delta_i^j - \nu\left(\frac{\partial \overline{v_i}}{\partial x_j} + \frac{\partial \overline{v_j}}{\partial x_i}\right) + \nu\left(\frac{\partial v_i}{\partial x_k}\frac{\partial \overline{x_k}}{\partial x_j} + \frac{\partial v_j}{\partial x_k}\frac{\partial \overline{x_k}}{\partial x_i}\right)\right] r_i \end{aligned} \tag{6.13}$$

Then, based on eq. C.9, the projected second derivative of J^{forces} is

$$\begin{aligned} \frac{\delta^2 J^{forces}}{\delta b_n \delta b_m} s_m = \int_\Omega \Bigg\{ & -\overline{u_i} v_j \frac{\partial v_i}{\partial x_k} - u_i \overline{v_j} \frac{\partial v_i}{\partial x_k} - u_i v_j \frac{\partial \overline{v_i}}{\partial x_k} + u_i v_j \frac{\partial v_i}{\partial x_\lambda}\frac{\partial \overline{x_\lambda}}{\partial x_k} - \overline{u_j}\frac{\partial p}{\partial x_k} - u_j \frac{\partial \overline{p}}{\partial x_k} + u_j \frac{\partial p}{\partial x_\lambda}\frac{\partial \overline{x_\lambda}}{\partial x_k} \\ & - \nu\left(\frac{\partial \overline{u_i}}{\partial x_j} + \frac{\partial \overline{u_j}}{\partial x_i}\right)\frac{\partial v_i}{\partial x_k} + \nu\left(\frac{\partial u_i}{\partial x_\lambda}\frac{\partial \overline{x_\lambda}}{\partial x_j} + \frac{\partial u_j}{\partial x_\lambda}\frac{\partial \overline{x_\lambda}}{\partial x_i}\right)\frac{\partial v_i}{\partial x_k} - \tau_{ij}^a \frac{\partial \overline{v_i}}{\partial x_k} + \tau_{ij}^a \frac{\partial v_i}{\partial x_\lambda}\frac{\partial \overline{x_\lambda}}{\partial x_k} \\ & + \overline{u_i}\frac{\partial \tau_{ij}}{\partial x_k} + u_i \frac{\partial}{\partial x_k}\left[\nu\left(\frac{\partial \overline{v_i}}{\partial x_j} + \frac{\partial \overline{v_j}}{\partial x_i}\right)\right] - u_i \frac{\partial}{\partial x_k}\left[\nu\left(\frac{\partial v_i}{\partial x_\lambda}\frac{\partial \overline{x_\lambda}}{\partial x_j} + \frac{\partial v_j}{\partial x_\lambda}\frac{\partial \overline{x_\lambda}}{\partial x_i}\right)\right] \end{aligned}$$

$$
\begin{aligned}
&-u_i\frac{\partial\tau_{ij}}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}+\overline{q}\frac{\partial v_j}{\partial x_k}+q\frac{\partial\overline{v_j}}{\partial x_k}-q\frac{\partial v_j}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}\Bigg\}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\
&+\int_\Omega\left(-u_iv_j\frac{\partial v_i}{\partial x_k}-u_j\frac{\partial p}{\partial x_k}-\tau^a_{ij}\frac{\partial v_i}{\partial x_k}+u_i\frac{\partial\tau_{ij}}{\partial x_k}+q\frac{\partial v_j}{\partial x_k}\right)\left[\frac{\partial}{\partial x_j}\left(\frac{\delta^2x_k}{\delta b_n\delta b_m}\right)s_m\right.\\
&\left.-\frac{\partial}{\partial x_\lambda}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial\overline{x_\lambda}}{\partial x_j}+\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial\overline{x_\lambda}}{\partial x_\lambda}\right]d\Omega+\int_{S_W}\left(p\delta^j_i-\tau_{ij}\right)r_i\frac{\overline{\delta(n_jdS)}}{\delta b_n}\\
&+\int_{S_W}\left[\overline{p}\delta^j_i-\nu\left(\frac{\partial\overline{v_i}}{\partial x_j}+\frac{\partial\overline{v_j}}{\partial x_i}\right)+\nu\left(\frac{\partial v_i}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\frac{\partial v_j}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\right)\right]r_i\frac{\delta(n_jdS)}{\delta b_n}
\end{aligned}
\tag{6.14}
$$

To compute the product of $\delta^2c_g/\delta\mathbf{b}^2$ and $\mathbf{s}$, since $c_g=\int_\Omega d\Omega-\widehat{\Omega}$, then by using eq. C.8 with $j_\Omega=$, then

$$
\begin{aligned}
\frac{\delta^2c_g}{\delta b_n\delta b_m}s_m=&\int_\Omega\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial\overline{x_\lambda}}{\partial x_\lambda}d\Omega-\int_\Omega\frac{\partial}{\partial x_\lambda}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial\overline{x_\lambda}}{\partial x_k}d\Omega\\
&+\int_\Omega\frac{\partial}{\partial x_k}\left(\frac{\delta^2x_k}{\delta b_n\delta b_m}\right)s_md\Omega
\end{aligned}
\tag{6.15}
$$

where, since the grid displacement model, eq. 5.4, is linear w.r.t. b_m, by differentiating eq. 5.11 w.r.t. b_m and multiplying with s_m, leads to $\frac{\delta^2x_i}{\delta b_n b_m}=0$.

To evaluate eqs. 6.14, PDEs governing $\overline{v_i}$, $\overline{p}$, $\overline{u_i}$, and $\overline{q}$ are developed and solved; these are obtained by differentiating eqs. 2.1a, 2.1b, 2.48a and 2.48b w.r.t. b_n and projecting them to $\mathbf{s}$, see Appendix C

$$
\overline{R^p}=\frac{\partial\overline{v_j}}{\partial x_j}-\frac{\partial v_j}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}=0 \tag{6.16}
$$

$$
\begin{aligned}
\overline{R^v_i}=&\frac{\partial(\overline{v_i}v_j)}{\partial x_j}+\frac{\partial(v_i\overline{v_j})}{\partial x_j}-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial\overline{v_i}}{\partial x_j}+\frac{\partial\overline{v_j}}{\partial x_i}\right)\right]+\frac{\partial\overline{p}}{\partial x_i}-\frac{\partial(v_iv_j)}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}-\frac{\partial p}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\\
&+\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_i}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\frac{\partial v_j}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\right)\right]+\frac{\partial}{\partial x_k}\left[\nu\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)\right]\frac{\partial\overline{x_k}}{\partial x_j}=0
\end{aligned}
\tag{6.17}
$$

$$
\overline{R^q}=\frac{\partial\overline{u_j}}{\partial x_j}-\frac{\partial u_j}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}=0 \tag{6.18}
$$

$$
\begin{aligned}
\overline{R^u_i}=&\overline{u_j}\frac{\partial v_j}{\partial x_i}+u_j\frac{\partial\overline{v_j}}{\partial x_i}-\frac{\partial(\overline{u_i}v_j)}{\partial x_j}-\frac{\partial(u_i\overline{v_j})}{\partial x_j}-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial\overline{u_i}}{\partial x_j}+\frac{\partial\overline{u_j}}{\partial x_i}\right)\right]+\frac{\partial\overline{q}}{\partial x_i}\\
&-u_j\frac{\partial v_j}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}+\frac{\partial(v_ju_i)}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial u_i}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\frac{\partial u_j}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\right)\right]\\
&+\frac{\partial}{\partial x_k}\left[\nu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right]\frac{\partial\overline{x_k}}{\partial x_j}-\frac{\partial q}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}=0
\end{aligned}
\tag{6.19}
$$

The boundary conditions for $\overline{v_i},\overline{p}$ are discussed in section C.1. The derivation

Inlet	$\overline{u_i}=0,\ \frac{\partial\overline{q}}{\partial n}=0$
Outlet	$\frac{\partial\overline{u}_{\langle n\rangle}}{\partial n}=0,\ \overline{q}=\overline{u}_{\langle n\rangle}v_{\langle n\rangle}+u_n\overline{v}_{\langle n\rangle}+2\nu\frac{\partial\overline{u}_{\langle n\rangle}}{\partial n}$ $\overline{v}_{\langle n\rangle}u_{\langle t\rangle}+v_{\langle n\rangle}\overline{u}_{\langle t\rangle}+\nu\left(\frac{\partial\overline{u}_{\langle n\rangle}}{\partial t}+\frac{\partial\overline{u}_{\langle t\rangle}}{\partial n}\right)=0$
Walls	$\overline{u_i}=0,\ \frac{\partial\overline{q}}{\partial n}=\frac{\partial q}{\partial x_j}\frac{\partial\overline{x_j}}{\partial n}-\frac{\partial q}{\partial x_i}\overline{n_i}$

Table 6.1: Boundary conditions for the equations providing the Hessian-vector products for problems with the force as the objective function.

of $\overline{u_i}$ and $\overline{q}$ can be found in Appendix C.2 for any objective function; regarding J^{forces}, they are summarized in table 6.1. To acquire their specialized expressions for J^{forces}, one has to consider that along the inlet and the outlet, no contributions from the objective function arise, since J^{forces} are defined along the walls. Also, along the walls, $u_i=-r_i$ (already discussed in 2.4), $\frac{\partial js_{I,i}}{\partial p}n_i=r_in_i$, $-\frac{\partial js_{I,k}}{\partial\tau_{ij}}n_kt_in_j-\frac{\partial js_{I,k}}{\partial\tau_{ij}}n_kt_jn_i=r_it_i$, $\overline{\frac{\partial js_{I,i}}{\partial p}}n_i=0$, and $-\overline{\frac{\partial js_{I,k}}{\partial\tau_{ij}}}n_kt_in_j-\overline{\frac{\partial js_{I,k}}{\partial\tau_{ij}}}n_kt_jn_i=0$.

The additional cost for computing each matrix-vector product Ms is practically equal to that of solving eqs. 6.16 to 6.19. As already discussed, two different systems of adjoint PDEs are solved in each optimization cycle (separately for J_1 and J_2). Nevertheless, eqs. 6.18 and 6.19 are solved only once per matrix-vector computation (i.e. in each GMRES inner iteration), by taking advantage of the linearity of the adjoint equations, eqs. 2.48a, 2.48b. This allows the linear combination of the two sets of adjoint variables corresponding to J_1 and J_2 (or c_f) into a single set of adjoint variables, as

$$u_i\big|_{un}=u_i\big|_{J_1}-\lambda_1u_i\big|_{c_f},\ q\big|_{un}=q\big|_{J_1}-\lambda_1q\big|_{c_f} \tag{6.20}$$

Here, subscripts J_1, c_f and un ("unified") denote the adjoint variables computed by solving the adjoint equations for J_1, c_f and their linear combination, respectively. Once the unified adjoint variables are computed by eq. 6.20, they are used to solve eqs. 6.18 and 6.19.

6.5 Applications

6.5.1 Shape Optimization in External Aerodynamics

In this section, the three proposed variants (1 to 3) are compared in the shape optimization of the NACA0012 (Case 10) and NACA4412 (Case 11) airfoils. In both cases, the two objectives are min. drag and max. lift, under the (geometric)

constraint of airfoil area preservation. The chord-based Reynolds flow number is $Re = 1000$ in both cases and the infinite flow angles are $\alpha_\infty = 2^0$ and 1^0 for Cases 10 and 11, respectively. Also, structured grids with $\sim 38K$ and $\sim 60K$ cells, max. distances of the first cell barycenters off the wall, normalized with the chord length, equal to $2.5\ 10^{-4}$ and 10^{-6} and airfoil contours consisting of 399 and 599 points, have been generated for Cases 10 and 11, respectively. Once a PP has been computed, $\hat{J}_2$ is changed by 10% and 30% in Cases 10 and 11, respectively. The termination criteria, see fig. 6.1, for reaching the next PP are set to $\epsilon_1 = \epsilon_2 = \epsilon_3 = 0.1\%$ for Case 10 and 0.3% for Case 11.

In fig. 6.3, the initial geometry of each airfoil along with its parameterization, based on volumetric B-Splines, are presented. The airfoil of Case 10 is parameterized by a 5×4 control box while, in Case 11, a 5×8 box is used. The x and y coordinates of the red control points (CPs) in fig. 6.3 are used as design variables, resulting to 12 and 24 design variables in total for Cases 10 and 11, respectively.

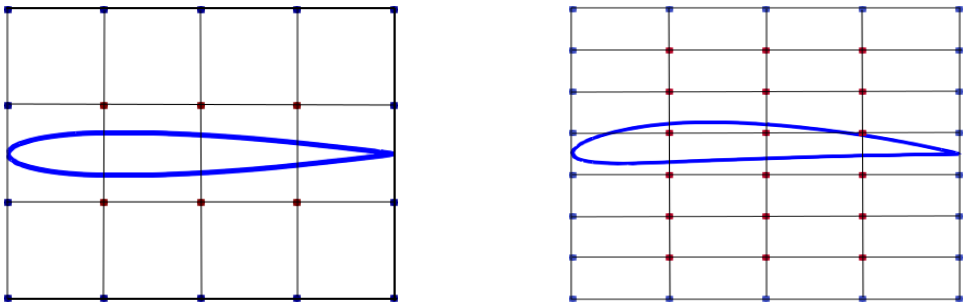


Figure 6.3: Parameterization based on volumetric B-Splines control boxes for Case 10 (left) and Case 11 (right).

To obtain the first PP, starting from the baseline geometries of both airfoils, constrained SOO problems are solved using the SQP method, for min. drag under the equality constraint on the airfoil area. In fig. 6.4, the convergence of the two SOO problems is presented. The baseline and optimized geometries along with the corresponding velocity fields are presented in figs. 6.6 and 6.7, respectively. In Case 10, the suction side becomes more flat and in Case 11, the mean camber line becomes less curved. Additionally, in Case 10, the drag integrand has decreased along the first 40% of the airfoil chord length and in Case 11, along the first 10% and between 35 to 100%, leading to a reduced drag force, see fig. 6.5.

Having computed the first PP, Variants 1 to 3 are used to trace the front of non-dominated solutions, see fig. 6.8. In Variant 1, two Restarted GMRES bases are used in both cases. The fronts of non-dominated solutions consist of 13 and 15 points for Cases 10 and 11, respectively, computed at the cost shown in table 6.2.

In table 6.3, the corresponding values of the hypervolume indicator [25] are presented; the PP corresponding to the min. J_2 computed by all variants was

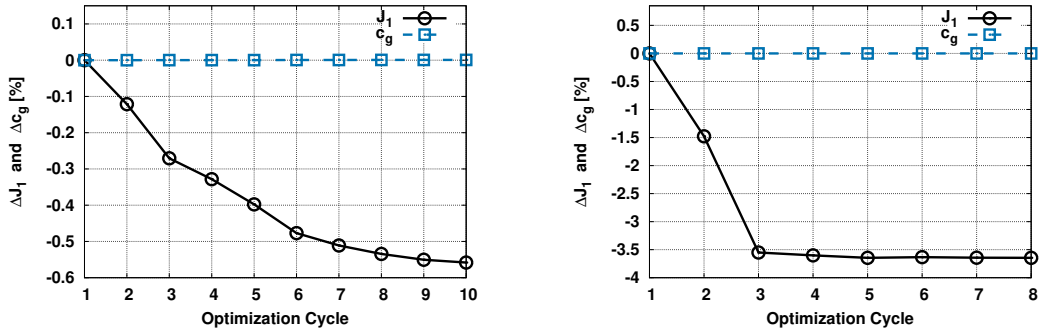
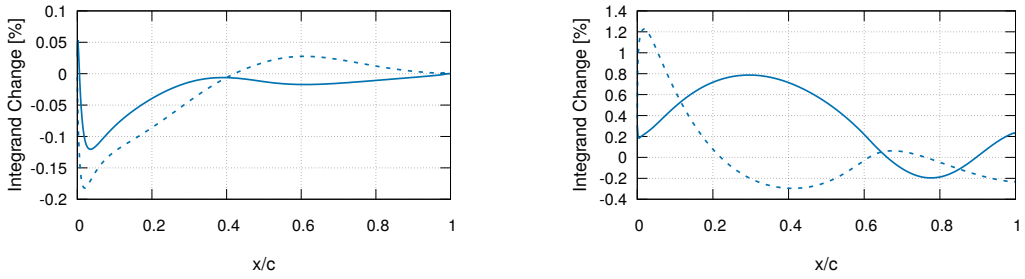
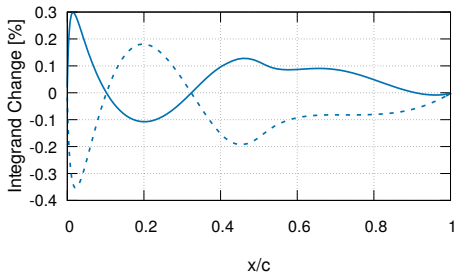


Figure 6.4: Convergence of the constrained SOO problems solved to find the first PP for Case 10 (left) and Case 11 (right) In both cases, the relative difference in the objective function values between the last two optimization cycles is below 0.01 %, which was the chosen stopping criterion. Also, the relative difference between the airfoil surface area and its initial value is of the order of 10^{-6} % for both cases. The euclidean norm of the KKT conditions, normalized with their initial value, are reduced to 0.11 and 0.04 for Case 10 and Case 11, respectively.

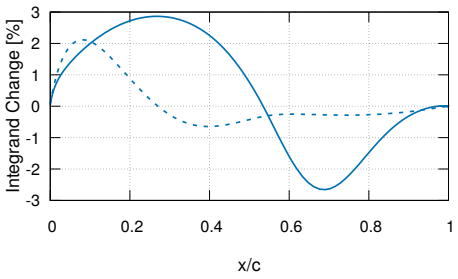


(a) Case 10: Min. drag

(b) Case 10: Max. lift



(c) Case 11: Min. drag



(d) Case 11: Max. lift

Figure 6.5: Difference in the drag (left) and lift (right) integrands, between the airfoils resulting from the min. drag and max.lift optimization and the baseline airfoils, computed along the chord length c. Changes in the force integrands are normalized with the resultant force value of the baseline geometry. Continuous lines pertain to the pressure and dashed lines to the suction sides of the airfoils.

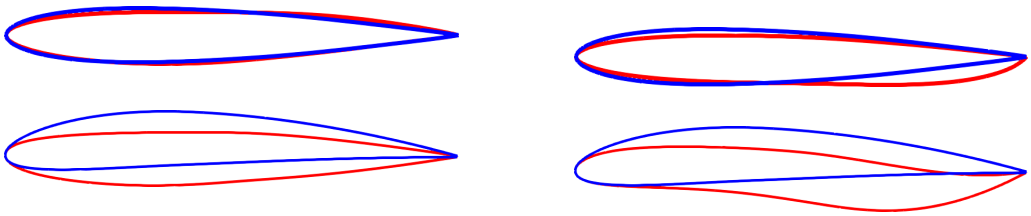


Figure 6.6: Min. drag (left) and max. lift (right) for the baseline (blue) and optimized (red) airfoils for Case 10 (top) and Case 11 (bottom).

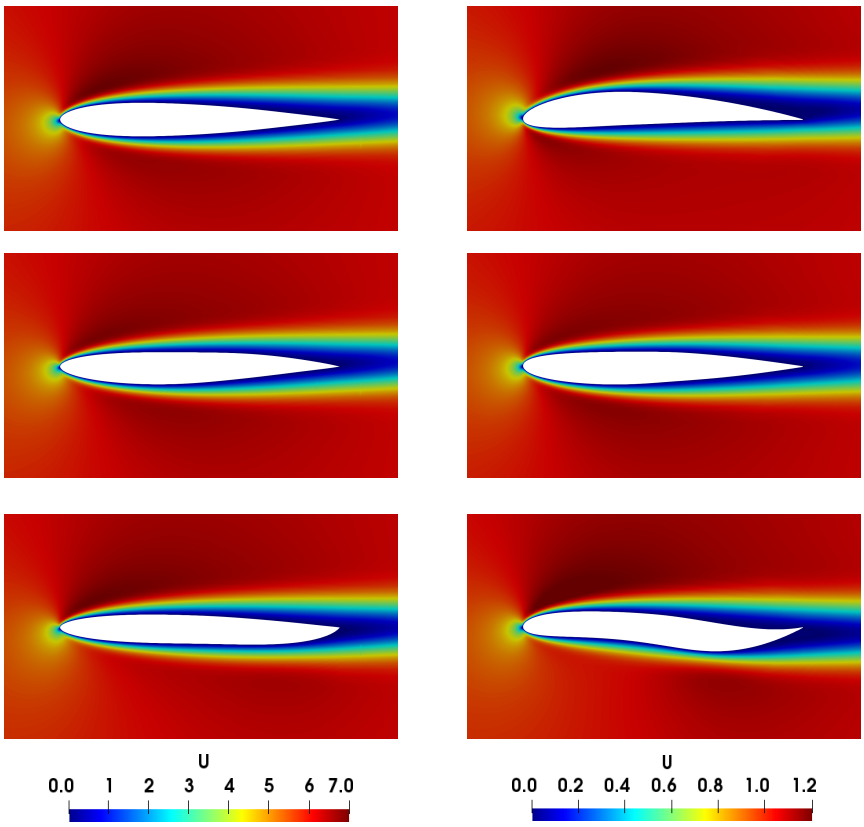


Figure 6.7: Velocity magnitude over the baseline (first row), the min. drag (second row) and the max. lift (third row) airfoils for the Case 10 (left) and Case 11 (right).

chosen as the nadir point. It can be observed that all fronts of non-dominated solutions are practically equivalent, since their hypervolume indices differ by no more than 1.2%. In terms of computational cost though, the difference between the three variants is more significant. Variant 1 is by far the most expensive one; the solution of additional PDEs (see eqs. 6.16 to 6.19) to compute matrix-vector products makes the cost for tracing the Pareto front more expensive. Variant 3 requires about 20% more EFS than Variant 2 due to, mainly, the higher number of optimization cycles required for the same number of PPs and, secondly, due to

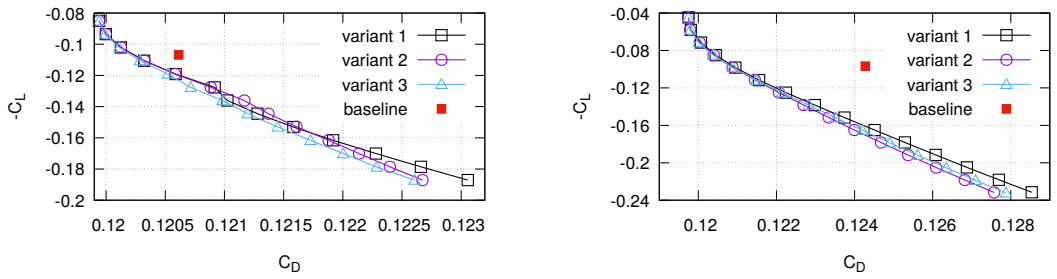


Figure 6.8: Pareto front approximations in Case 10 (left) and Case 11 (right), along with the baseline points in the objective space.

Case	Variant	Total EFS	Total Optimization Cycles	EFS due to line-search	EFS for paths of fig. 6.10
1	**1**	152	35	25	8
	2	81	26	5	6
	3	110	34	10	10
2	**1**	71	18	1	8
	2	50	17	1	6
	3	61	18	9	7

Table 6.2: Computational cost, measured in EFS, to move (a) from the first PP to the last (third to fifth columns) and (b) between the two PPs presented in fig. 6.10 (last column), for Cases 10 and 11. The total number of EFS spent for the solution of the primal equations during line-search iterations is presented in the second-to-last column. The cost for reaching the first PP is the same for all variants and is, thus, excluded from this table.

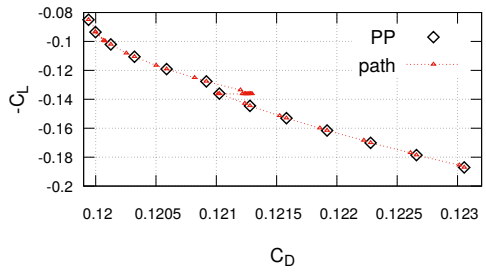
an increased line-search cost.

In fig. 6.9, the paths from each PP to the next one are illustrated while, in fig. 6.10, indicative close-up views are presented. In both cases, it can be observed that, in Variant 1, even though just a few GMRES bases are used, the prediction step yields a solution quite close to the next PP. In Case 10, the first step of Variant 3 fails to lead close to the next PP for the central part of the Pareto front; additionally, more line-search iterations are required, on average, compared to Variant 2. Therefore, it is concluded that the prediction step in Variant 2 is beneficial and should be retained, since it reduces the overall number of EFS compared to Variant 3.

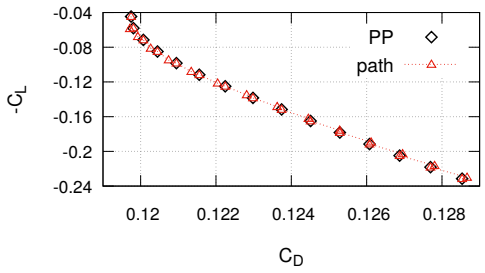
In both cases, to maximize lift, the pressure side close to the trailing edge moves downwards, fig. 6.6. Both the lift integrand, fig. 6.5, and the pressure difference between the airfoil sides, fig. 6.11, increase over the first part of the

Variant	Case 10	Case 11
1	0.89241	0.89157
2	0.89214	0.90113
3	0.90363	0.89881

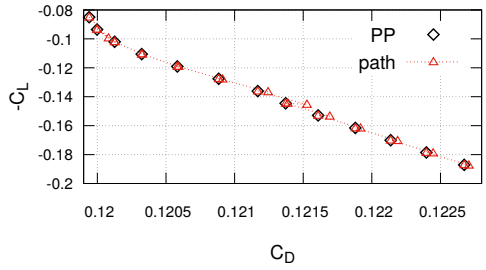
Table 6.3: Hypervolume indicator values for the computed fronts of non-dominated solutions.



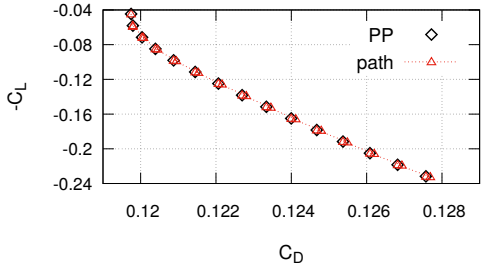
(a) Case 10: Variant 1



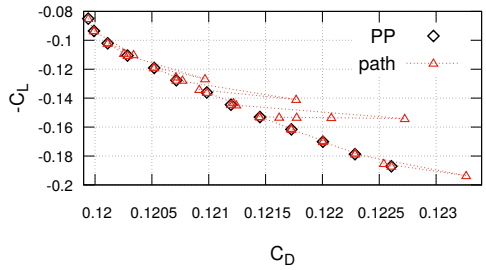
(b) Case 11: Variant 1



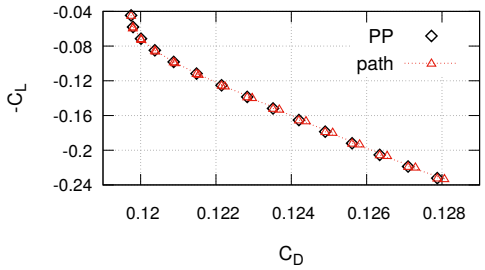
(c) Case 10: Variant 2



(d) Case 11: Variant 2

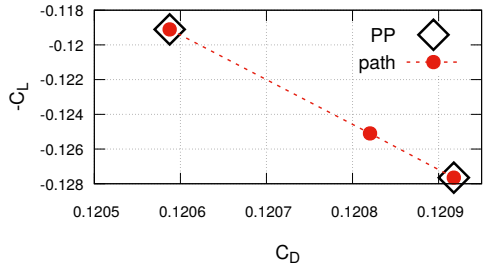


(e) Case 10: Variant 3

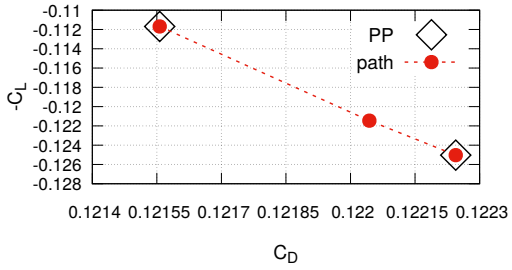


(f) Case 11: Variant 3

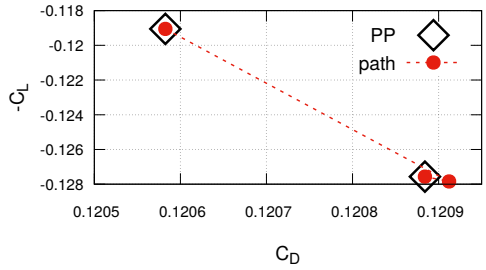
Figure 6.9: Paths connecting one PP to the next using Variants 1 to 3, for Cases 10 and 11.



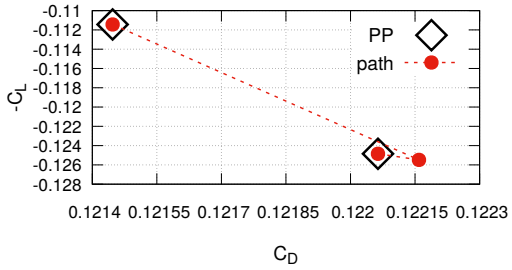
(a) Case 10: Variant 1



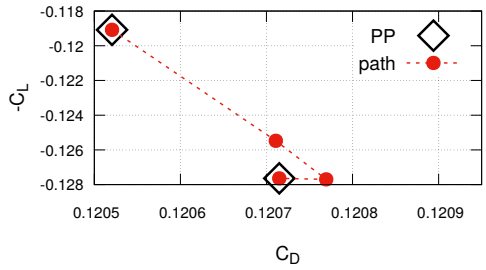
(b) Case 11: Variant 1



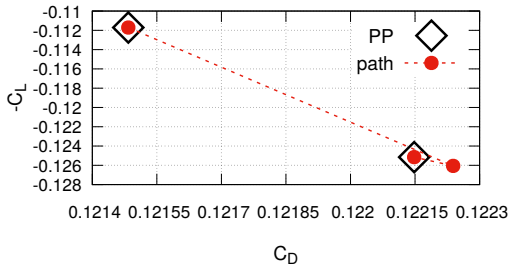
(c) Case 10: Variant 2



(d) Case 11: Variant 2



(e) Case 10: Variant 3



(f) Case 11: Variant 3

Figure 6.10: Indicative paths connecting one PP (top point) to the next (bottom point) In Case 10 (Case 11), the path between the fifth and sixth PPs (sixth and seventh PPs) are plotted.

chord length, leading to higher lift.

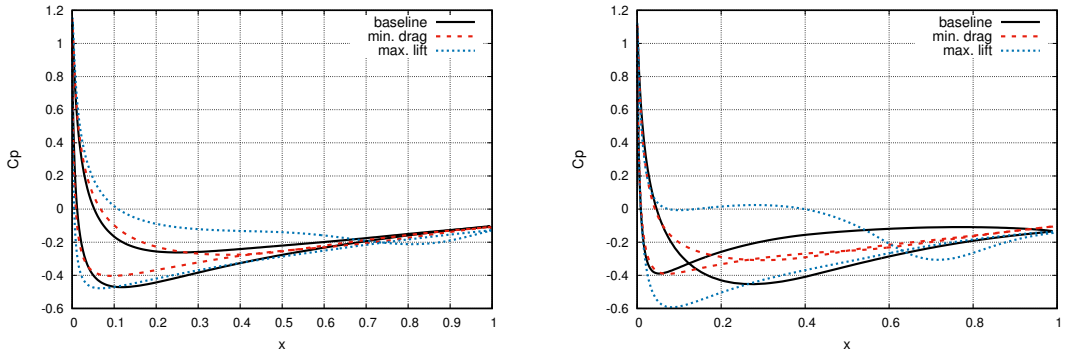


Figure 6.11: Pressure coefficient (Cp) distributions for Case 10 (left) and Case 11 (right) airfoils; distributions over the baseline (black), the min. drag (red) and the max. lift (blue color) airfoils are shown.

6.5.2 Comparison of Variant 2 with the Weighted-Sum Approach

In section 6.5.1, Variant 2 appeared to consistently be the cheapest one. Here, for one of the two cases presented (Case 11), Variant 2 is compared to the weighted sum minimization method [22], in which the goal is to minimize $J = w_1 J_1 + w_2 J_2$ constrained by $c_g = 0$. To do so, the SQP method is used. To compute a new PP, an optimization problem for a different combination of weights, here with $w_1 + w_2 = 1$, is solved. Regarding the initialization of each of them, two different approaches are used. The first, denoted by *WS-BSL*, starts from the initial/baseline geometry. In the second, denoted by *WS-cont*, the previously computed PP is used to initialize the search for the new PP corresponding to the new values of weights. The latter "moves" along the Pareto front, as Variants 1 to 3 do, after the first PP has been computed. In both weighted sum minimization approaches, convergence to a PP is achieved when the relative difference in both drag and lift between two optimization cycles is below ϵ_1, with the same convergence criterion for c_g as in Variants 1 to 3. Results are presented in fig. 6.12 and the relevant computational cost is presented in table 6.4. Compared to Variant 2, *WS-BSL* and *WS-cont* are $\times 2.1$ and $\times 3$ more expensive. Finally, moving along the Pareto front by perturbing the weights, as in *WS-cont*, is by $\sim 33\%$ cheaper than restarting the optimization from the baseline geometry for each point of the Pareto front.

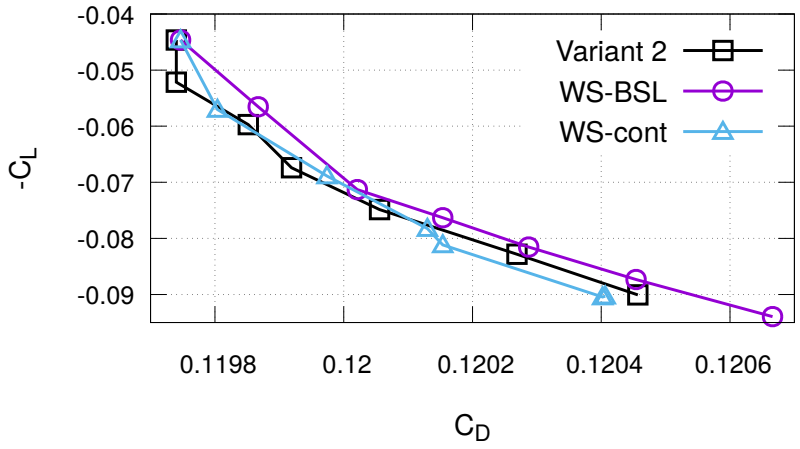


Figure 6.12: Case 11: Comparison of *WS-BSL* and *WS-cont* and Variant 2, over a part of the front. In the weighted sum minimization problems, $w_1 = 1,\ 0.99,\ 0.98,\ 0.977,\ 0.974,\ 0.971,\ 0.978$ and $w_2 = 1 - w_1$. Regarding Variant 2, each time a new PP has been found, $\hat{J}_2$ is changed by 17%.

Method	Total EFS
Variant 2	29
WS-BSL	92
WS-cont	62

Table 6.4: Case 11: Cost of computing the Pareto fronts of fig. 6.12, based on Variant 2, *WS-BSL* and *WS-cont*. The cost for the first PP, which is the same for all methods, is excluded.

6.5.3 Case 12: Shape Optimization in a 3D CHT Problem

The computation of a Pareto front pertaining to a CHT shape optimization problem is studied, using Variant 2 which has shown to consistently be the least expensive one, according to the studies performed in section 6.5.1. This case addresses the turbulent fluid flow of air passing between two successive cooling fins of a heated device. Objectives are the minimization of the volume-weighted total pressure losses between the inlet and outlet of the flow domain ($J_1 = J^{pt}$, eq. B.3) and the minimization of the high T^S areas, ($J_2 = J^{penT}$, eq. B.5, with $T_{safe} = 650\ K$ and $T_{crit} = 730\ K$). The primal and adjoint PDEs for both the fluid and solid domains, as well as the gradient expression can be found Chapter 3.

The geometry is presented in fig. 6.13. The cell numbers of the fluid and solid grids are $\sim 370K$ and $\sim 74K$, respectively. Also, $Re = 2.83 \times 10^5$, y^+ of the first

cell barycenters off the wall is less than 0.6, $Pr = 0.7$, $Pr_t = 1$, $c_p = 1006 J/kg/K$ and $k^S = 60 J/m/s/K$. At the bottom surface of the solid body, a constant heat flux of magnitude $Q = 4800\ W/m^2$ is imposed. The side walls of Ω^F and Ω^S are periodic. Along $\overline{S}^F$ and $\overline{S}^S$, conditions given by eqs. 3.9 and 3.10 for the primal problem and, eqs. 3.30a and 3.30b for the adjoint problem, are imposed. The rest of the non-FSI and non-periodic solid boundaries are assumed to be adiabatic.

To compute the first PP, a SOO for minimum J^{pt} is solved. This converges in 12 cycles and reduces J^{pt} by $\sim 18\%$. After the computation of the first PP, Variant 2 computes the remaining PPs, see fig. 6.14, at a cost of 120 EFS. In figs. 6.15 and 6.16, the velocity magnitude and the T^S values are presented for the baseline geometry and the ones corresponding to min. J^{pt} and min. J^{penT}. In order to minimize J^{pt}, the cross-section area of the fluid passage between two consecutive fins becomes wider and the velocity magnitude reduces. This, however, increases J^{penT} by almost 40%, with the max. T^S value increasing by $24.6\ K$. On the other hand, to minimize J^{penT}, the opposite changes are made, i.e. the fluid passage between the fins becomes narrower, the flow accelerates and thus, heat convection is enhanced. As a result, the max. T^S value reduces by $32.1\ K$ compared to the baseline geometry, at the expense of an increased J^{pt} by 27.8%.

6.6 Conclusions

Three variants of a prediction-correction algorithm for computing Pareto fronts in multi-objective optimization problems, solved by means of gradient-based methods supported by continuous adjoint, were investigated; the background of the prediction-correction algorithm has theoretically been founded in [148]. The three variants were compared in terms of computational cost, after ensuring that they computed practically the same approximations to the Pareto fronts. All of them required a starting point on the Pareto front, computed through a SOO run; for a fair comparison, the same starting point was used in all of them. The costly computation of the exact Hessian matrix to be used on the l.h.s. of the prediction and correction step equations was avoided through (a) the computation of Hessian-vector products, driving a linear-Restarted GMRES method and (b) the damped BFGS method, which approximates the Hessian by using the gradient of objective and constraint functions w.r.t. the design variables. The continuous adjoint method computed the latter. Matrix-vector products were computed after applying the Direct Differentiation method on the primal and adjoint equations. The three variants were used to compute fronts of non-dominated solutions on the drag-lift plane of two isolated airfoils, while retaining the airfoil area constant. All produced practically equivalent fronts of non-dominated solutions, with the variant that approximates the Hessian matrix in both steps through BFGS being the least expensive. The simplest variant, in which the prediction step is omit-

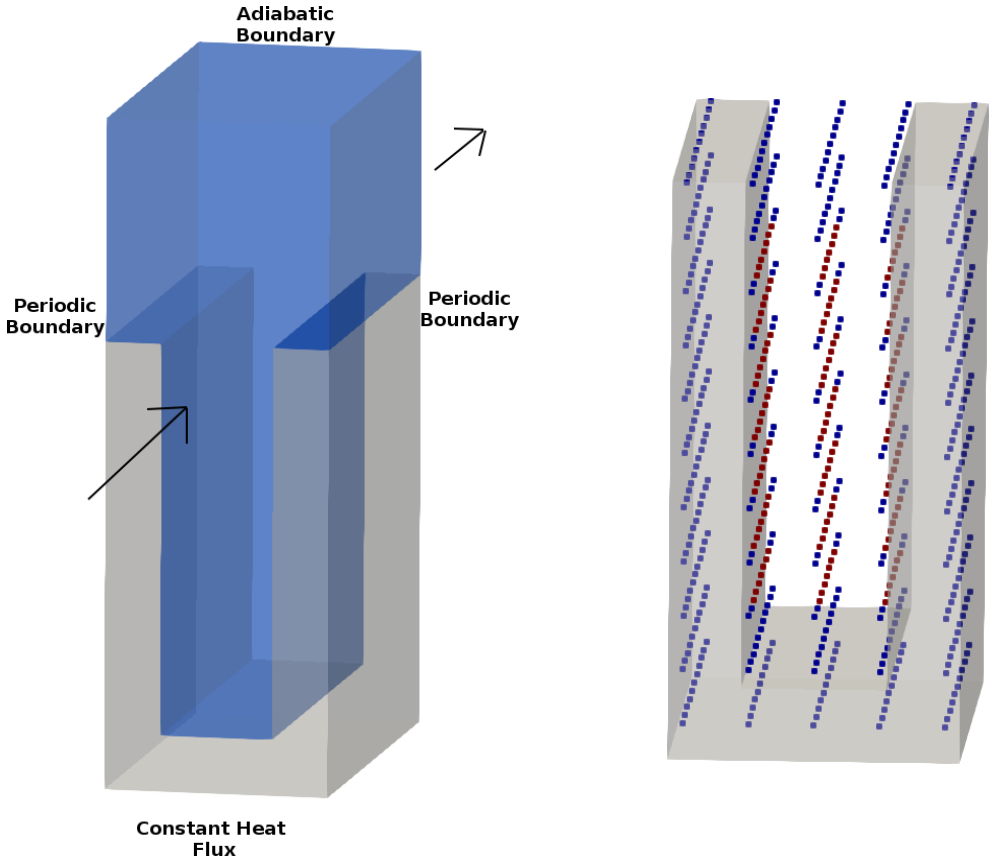


Figure 6.13: Case 12: Left: Fluid (blue) and solid (grey) domains. Arrows denote the fluid flow direction. Right: Parameterization based on volumetric B-Splines. The blue CPs are kept fixed while the red ones can be displaced during the optimization.

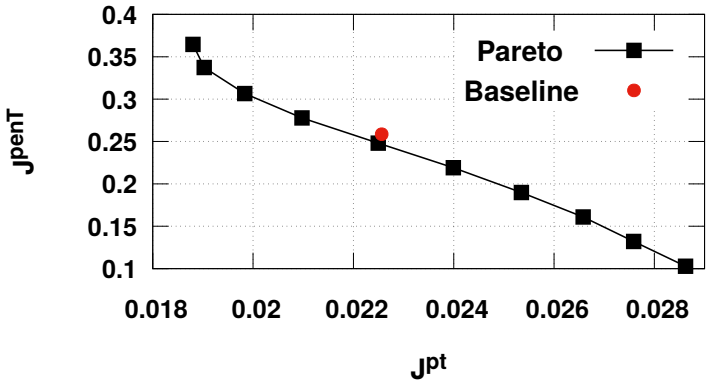
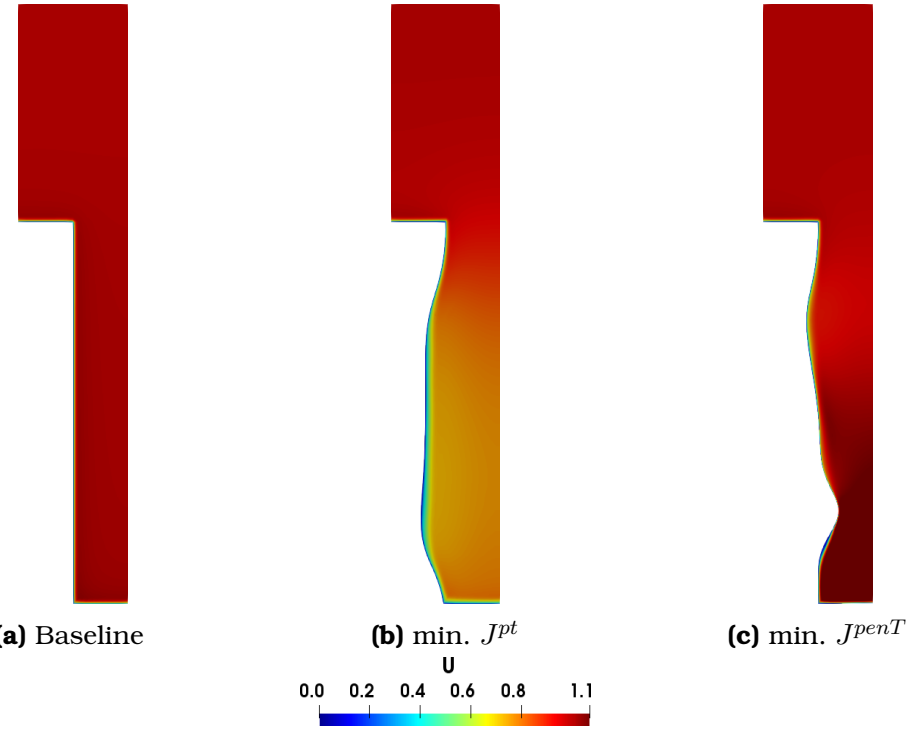


Figure 6.14: Case 12: Pareto front for min. J^{pt} and min. J^{penT}.

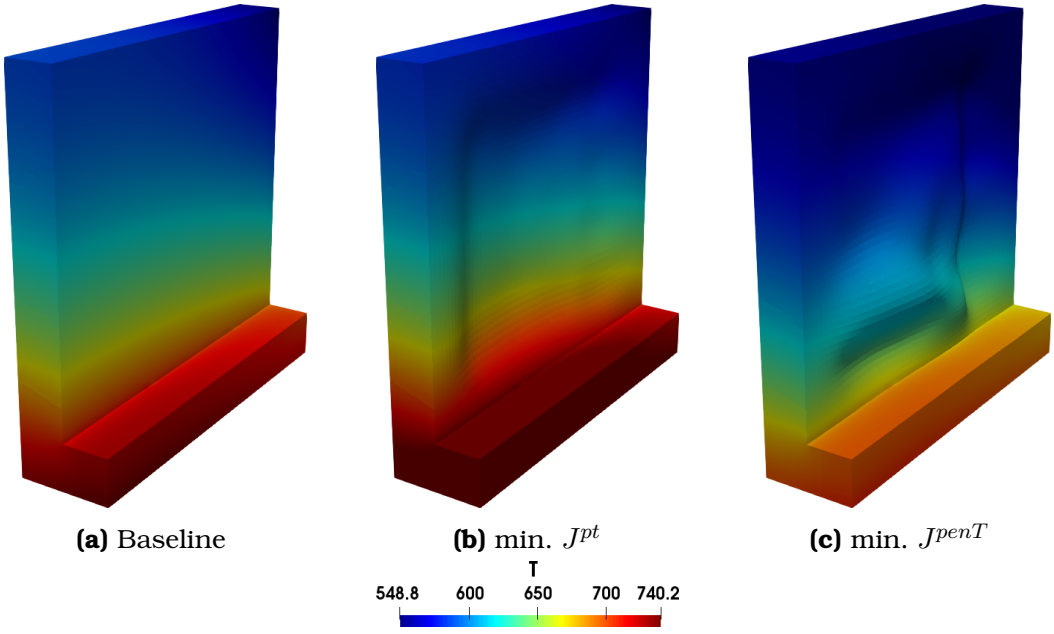
ted and only consecutive constrained optimization problems based on SQP are



(a) Baseline **(b)** min. J^{pt} **(c)** min. J^{penT}

Figure 6.15: Case 12: Velocity magnitude at a cross-section in the middle of the length of the fluid passage. Due to symmetry, only half of the fluid domain is presented.

solved, proved to be more expensive. Hence, it can be concluded that including the prediction step becomes beneficial in terms of cost reduction. Directly computing matrix-vector products in the prediction step has been shown to lead to the most expensive variant, since additional PDEs must be solved to compute the Hessian-vector products. The prediction-correction variant in which, BFGS approximated the Hessian in both steps was demonstrated to be less expensive than the weighted sum method, with the additional advantage of controlling the distance between the PPs in the objective space. Finally, using the most efficient variant, the Pareto front for a 3D Conjugate Heat Transfer application under turbulent flow conditions, with total pressure losses and max. solid temperature as the contradicting targets, was computed. The cost to compute 10 Pareto front members, being equal to 150 EFS, was affordable.



(a) Baseline **(b)** min. J^{pt} **(c)** min. J^{penT}

Figure 6.16: Case 12: T^S values. Due to symmetry, only half of the computational domain is presented.

Chapter 7

UQ and Optimization under Uncertainties in CHT Problems using intrusive PCE

This chapter contributes to the field of Uncertainty Quantification (UQ) and Optimization under Uncertainties (OuU), for CHT problems using continuous adjoint methods. First, the theory of the Polynomial Chaos Expansion (PCE) method, based on the expansion of any quantity into the weighted sum of orthogonal polynomials, is presented. As explained in section 1.4, PCE methods are discerned to intrusive (iPCE) and non-intrusive (niPCE) ones. In this section, emphasis is laid on the former, which has lower computational cost than the latter. The niPCE method is used to verify the iPCE one. First, the iPCE method is presented for CHT problems and laminar flows for computing the mean value and standard deviation of a Quantity of Interest (QoI), which could stand for the mean temperature of a solid body, the heat flux passing from the fluid to the solid etc. In the iPCE method, the flow variables inside the primal equations, corresponding to a single point of operation (deterministic[1] equations), are expanded and new systems of PDEs arise. Problems with uncertainties associated with the inlet conditions of the fluid flow and the thickness of the thermal insulation material between the fluid and solid are investigated and the corresponding iPCE boundary conditions are derived. To compute SDs of the mean value and standard deviation of the QoI, the iPCE primal PDEs are differentiated w.r.t. the design variables. This leads to new adjoint PDEs, boundary conditions and SD expressions. FDs are used to verify the iPCE-based adjoint SDs. Next, the niPCE method is presented. In the niPCE method, the PCE is applied to the QoI, rather than the flow variables. This yields unknown coefficients computed, in this thesis, through Galerkin Projections. The estimated, by both the iPCE and niPCE methods, statistical moments are also compared with Monte Carlo results. Finally, shape optimizations under uncertainties, based on the derived iPCE primal and adjoint equations, are car-

[1] Throughout this chapter, term "deterministic" refers to equations and/or quantities pertinent to the same case without uncertainties.

ried out. The optimizations aim at min. mean value or standard deviation of the QoI, in order to assess the developed iPCE method and, also, compare trends in the geometry deformations when minimizing each one of the two moments.

7.1 Theory of the PCE Method

The PCE method is a way to compute statistical moments, such as the mean value and/or standard deviation J', of the QoI J. To do so, any function f, which could stand for the flow variables (pressure, velocity components, etc) or even the QoI itself, is expressed in terms of a weighted sum of proper polynomials as

$$f = \sum_{m=0}^{M} f_m \psi_m(\boldsymbol{\xi}) \tag{7.1}$$

where $\boldsymbol{\xi} = (\xi_1, ..., \xi_K)$ is the vector of the K independent uncertain variables, defined in the multi-dimensional set $\boldsymbol{\mathcal{D}} = \mathcal{D}_1 \times \mathcal{D}_2 \times ... \times \mathcal{D}_K \subset \mathbb{R}^K$, where $\mathcal{D}_i \subset \mathbb{R}$ is the support range of the i-th uncertain variable. Uncertain variables could be related to environmental conditions, such as the infinite/inlet velocity magnitude or the fluid temperature etc, or even to manufacturing dimensions, such as the thickness of the thermal insulation between solids and fluids in the examined CHT problems. Each uncertain variable ξ_i is governed by its own stochastic distribution expressed by a specific Probability Density Function (PDF). Regarding eq. 7.1, $\psi_m(\boldsymbol{\xi}),\ m = 0, ..., M$ are multivariate orthogonal polynomials and f_m the unknown PCE coefficients. Two different approaches can be used to compute the mean value and standard deviation of a QoI. When eq. 7.1 is used to expand the deterministic primal and adjoint variables, then new PDEs arise. On the other hand, when applying eq. 7.1 to the QoI, the arising PCE coefficients f_m pertain to the QoI and are computed without deriving new PDEs; in this case, solving the deterministic primal PDEs at various value-sets of the uncertain variables is enough. The former approach corresponds to the iPCE and the latter to the niPCE. The number $M+1$ of PCE coefficients is given by

$$M+1 = \binom{K+\varpi}{K} = \frac{(K+\varpi)!}{K!\varpi!} \tag{7.2}$$

where ϖ is the user-defined chaos order and controls the accuracy of the iPCE and niPCE methods [167]. Each multivariate polynomial $\psi_m(\boldsymbol{\xi})$ used in eq. 7.1 is

Distribution	PDF	Polynomial Class	$w_i(\xi_i)$	$\mathcal{D}_i$
Uniform	$\frac{1}{2}$	Legendre $P_n(\xi_i)$	1	$[-1,1]$
Normal	$\frac{1}{\sqrt{2\pi}}e^{-\frac{\xi_i^2}{2}}$	Hermite $He_n(\xi_i)$	$e^{-\frac{\xi_i^2}{2}}$	$(-\infty,\infty)$
Beta	$\frac{(1-\xi_i)^\alpha(1+\xi_i)^\beta}{2^{(\alpha+\beta+1)}B(\alpha+1,\beta+1)}$	Jacobi $J_n^{\alpha,\beta}(\xi_i)$	$(1-\xi_i)^\alpha(1+\xi_i)^\beta$ $\alpha,\beta>-1$	$[-1,1]$
Exponential	$e^{-\xi_i}$	Laguerre $L_n(\xi_i)$	$e^{-\xi_i}$	$[0,\infty)$
Gamma	$\frac{\xi_i^\alpha e^{-\xi_i}}{\Gamma(\alpha+1)}$	Generalized Laguerre $L_n^\alpha(\xi_i)$	$x^\alpha e^{-\xi_i}$	$[0,\infty)$

Table 7.1: List of uncertain variable distributions with their PDFs, the corresponding orthogonal polynomials, weight functions and support ranges ($\mathcal{D}_i$) of ξ_i [167, 32, 31].

the product of K univariate orthogonal polynomials p as

$$\psi_m(\boldsymbol{\xi}) = \prod_{l=1}^{K} p_{i_l}(\xi_l) \tag{7.3}$$

where i_l denotes the degree of each univariate polynomial and $\sum_{l=1}^{K} i_l = \varpi$. The inner product of two univariate orthogonal polynomials is defined as

$$\langle p_i(\xi_k), p_j(\xi_k)\rangle_w = \int_{\mathcal{D}_k} p_i(\xi_k)p_j(\xi_k)w_k(\xi_k)d\xi_k = \langle p_i(\xi_k), p_i(\xi_k)\rangle_w \delta_i^j = ||p_i||_w^2 \delta_i^j = h_i^2\delta_i^j \tag{7.4}$$

where $w_k(\xi_k)$ is a weighting function pertaining to the stochastic distribution of ξ_k. Term "orthogonality" stems from the fact that the inner product of two polynomials of different degree is zero; otherwise their inner product is equal of h_i^2. If $h_i = 1$, the polynomials are referred to as "orthonormal", herein denoted by the symbol $\sim$ above them, i.e. $\widetilde{p}_i(\xi_k) = p_i(\xi_k)/h_i$. Different classes of univariate orthogonal polynomials exist, each of them being suitable for a specific stochastic distribution, see table 7.1 for frequently used distributions. In [167], a discussion is made on the usage of less suitable polynomials for some stochastic distributions. The advantages of choosing the suitable polynomial class instead of unsuitable ones are that (a) the distribution of an uncertain variable can be accurately reproduced by using less PCE coefficients and (b) by increasing M, the error in the estimations of the QoI mean value and standard deviation reduces faster (at an exponential rate). A common feature of all classes of orthogonal polynomials is that $p_0 = 1$. Let

the product of the weighting functions be defined as

$$W(\boldsymbol{\xi}) = \prod_{i=0}^{K} w_i(\xi_i) \tag{7.5}$$

By using eqs. 7.4 and 7.5, the inner product of two multivariate orthogonal polynomials is given by

$$\begin{aligned}\langle \psi_i(\boldsymbol{\xi}), \psi_j(\boldsymbol{\xi}) \rangle_w &= \int_{\boldsymbol{\mathcal{D}}} \psi_i(\boldsymbol{\xi}) \psi_j(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \int_{\boldsymbol{\mathcal{D}}} \prod_{l=1}^{K} p_{i_l}(\xi_l) \prod_{l=1}^{K} p_{j_l}(\xi_l) \prod_{l=1}^{K} w_l(\xi_l)\, d\boldsymbol{\xi} \\ &= \delta_i^j \prod_{l=1}^{K} \int_{\mathcal{D}_l} p_{i_l}^2(\xi_l) w_l(\xi_l) d\xi_l = \delta_i^j \int_{\boldsymbol{\mathcal{D}}} \psi_i^2(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \langle \psi_i(\boldsymbol{\xi}), \psi_i(\boldsymbol{\xi}) \rangle_w \delta_i^j \end{aligned} \tag{7.6}$$

To compute the w-norm of a $\psi_i(\boldsymbol{\xi})$ polynomial, since

$$\langle \psi_i(\boldsymbol{\xi}), \psi_i(\boldsymbol{\xi}) \rangle_w = \prod_{l=1}^{K} ||p_i||_w^2 = \prod_{l=1}^{K} h_i^2 \tag{7.7}$$

then

$$||\psi_k||_w = \prod_{l=1}^{K} ||p_{i_l}||_w = \prod_{l=1}^{K} h_i \tag{7.8}$$

7.2 The intrusive PCE Method

In the iPCE method, each primal and adjoint variable Φ is expanded by means of eq. 7.1, i.e.

$$\Phi = \sum_{m=0}^{M} \Phi_m \psi_m(\boldsymbol{\xi}) \tag{7.9}$$

leading to PDEs including unknown coefficient fields Φ_m multiplied by the orthogonal polynomials $\psi_m(\boldsymbol{\xi})$. Galerkin projections of these PDEs are then performed, leading to as many equations as the unknown coefficient fields. The newly derived equations are numerically solved by properly adapting the functionalities of the deterministic primal solver, [17, 124, 114]. In eq. 7.9, the multivariate orthogonal polynomials $\psi_m(\boldsymbol{\xi})$ are found after having defined the univariate ones. Let each uncertain variable follows the normal distribution with mean value μ and variance σ^2, i.e. $\mathcal{N}(\mu, \sigma^2)$, as in this thesis. Then, the probabilist's univariate Hermite

orthogonal polynomials [9] must be used. The first six of them are listed below

$$
\begin{aligned}
He_0(\xi_i) &= 1, \\
He_1(\xi_i) &= \xi_i, \\
He_2(\xi_i) &= \xi_i^2 - 1, \\
He_3(\xi_i) &= \xi_i^3 - 3\xi_i, \\
He_4(\xi_i) &= \xi_i^4 - 6\xi_i^2 + 3, \\
He_5(\xi_i) &= \xi_i^5 - 10\xi_i^3 + 15\xi_i, \\
&\dots
\end{aligned}
$$

A recursive formula may derive the polynomial of degree $k+1$, based on the polynomials of degrees k and $k-1$

$$
He_{k+1}(\xi_i) = \xi_i He_k - k He_{k-1} \tag{7.10}
$$

The w-norm of a Hermite polynomial of degree k is

$$
\|He_k\|_w^2 = \langle He_k(\xi_i), He_k(\xi_i) \rangle_w = \int_{-\infty}^{+\infty} He_k^2(\xi_i) w(\xi_i) dx = k! \tag{7.11}
$$

where w is given by

$$
w(\xi_i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{\xi_i^2}{2}} \tag{7.12}
$$

In this thesis, problems with two uncertain variables ($K=2$) are solved and the chaos order is chosen to be $\varpi = 2$. Thus, $M=5$, see eq. 7.2, and the multivariate orthogonal polynomials used in eq. 7.9 are

$$
\begin{aligned}
\psi_0(\boldsymbol{\xi}) &= He_0(\xi_1) He_0(\xi_2) = 1 \\
\psi_1(\boldsymbol{\xi}) &= He_1(\xi_1) He_0(\xi_2) = \xi_1 \\
\psi_2(\boldsymbol{\xi}) &= He_0(\xi_1) He_1(\xi_2) = \xi_2 \\
\psi_3(\boldsymbol{\xi}) &= He_1(\xi_1) He_1(\xi_2) = \xi_1 \xi_2 \\
\psi_4(\boldsymbol{\xi}) &= He_2(\xi_1) He_0(\xi_2) = \xi_1^2 - 1 \\
\psi_5(\boldsymbol{\xi}) &= He_0(\xi_1) He_2(\xi_2) = \xi_2^2 - 1
\end{aligned} \tag{7.13}
$$

Below, the development of the iPCE method is presented for CHT problems with laminar flows. By omitting the iPCE energy equation of the fluid and the iPCE heat conduction equations for the solid, the remaining PDEs can be used in pure fluid flow problems with uncertainties.

7.2.1 The iPCE Primal PDEs

The deterministic PDEs for CHT problems with laminar flows are repeated here for the sake of completeness

$$R^p = -\frac{\partial v_j}{\partial x_j} = 0 \tag{7.14}$$

$$R^v_i = v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] + \frac{\partial p}{\partial x_i} = 0, \quad i = 1, 2(, 3) \tag{7.15}$$

$$R^{T^F} = \rho^F c_p v_j \frac{\partial T^F}{\partial x_j} - \frac{\partial}{\partial x_j}\left(k^F \frac{\partial T^F}{\partial x_j}\right) = 0 \tag{7.16}$$

$$R^{T^S} = -\frac{\partial}{\partial x_j}\left(k^S \frac{\partial T^S}{\partial x_j}\right) = 0 \tag{7.17}$$

Since $M+1=6$, by expanding each of $v_i,\ p,\ T^F,\ T^S$ fields according to eq. 7.9, six unknown PCE coefficient fields for each variable emerge, resulting in 30 (in 2D, 36 in 3D) unknown fields in total. By expanding eqs. 7.14 to 7.17, 30 PDEs (in 2D, 36 in 3D) are derived. Their numerical solver is programmed in the OpenFOAM© software. For different values of K or ϖ, the number of PCE fields, in which each flow variable field is expanded, would change and different iPCE equations would arise. By expanding each of the primal variables using eq. 7.9, the following PDEs arise

$$R^p = -\sum_{m=0}^{M} \frac{\partial v_{j,m}}{\partial x_j} \psi_m(\boldsymbol{\xi}) = 0 \tag{7.18}$$

$$R^v_i = \sum_{m=0}^{M}\sum_{n=0}^{M} v_{j,m}\psi_m(\boldsymbol{\xi}) \frac{\partial v_{i,n}}{\partial x_j}\psi_n(\boldsymbol{\xi}) - \sum_{m=0}^{M} \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,m}}{\partial x_j} + \frac{\partial v_{j,m}}{\partial x_i}\right)\psi_m(\boldsymbol{\xi})\right] + \sum_{m=0}^{M} \frac{\partial p_m}{\partial x_i}\psi_m(\boldsymbol{\xi}) = 0, \quad i = 1, 2(, 3) \tag{7.19}$$

$$R^{T^F} = \sum_{m=0}^{M}\sum_{n=0}^{M} \rho^F c_p v_{j,m}\psi_m(\boldsymbol{\xi}) \frac{\partial T^F_n}{\partial x_j}\psi_n(\boldsymbol{\xi}) - \sum_{m=0}^{M} \frac{\partial}{\partial x_j}\left[k^F \frac{\partial T^F_m}{\partial x_j}\psi_m(\boldsymbol{\xi})\right] = 0 \tag{7.20}$$

$$R^{T^S} = -\sum_{m=0}^{M} \frac{\partial}{\partial x_j}\left[k^S \frac{\partial T^S_m}{\partial x_j}\psi_m(\boldsymbol{\xi})\right] = 0 \tag{7.21}$$

To compute the 30 unknown coefficient fields, 30 new PDEs are derived by performing Galerkin projections of eqs. 7.18 to 7.21 onto multivariate Hermite poly-

nomials $\psi_k(\boldsymbol{\xi}),\ k=0,...,M$, i.e.

$$
{}^{k}R^{p} = -\sum_{m=0}^{M}\int_{\boldsymbol{\mathcal{D}}}\frac{\partial v_{j,m}}{\partial x_j}\psi_m(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\,d\boldsymbol{\xi} = -\sum_{m=0}^{M}\frac{\partial v_{j,m}}{\partial x_j}\langle\psi_m(\boldsymbol{\xi}),\psi_k(\boldsymbol{\xi})\rangle_w = 0 \tag{7.22}
$$

$$
\begin{aligned}
{}^{k}R_i^{v} = &\sum_{m=0}^{M}\sum_{n=0}^{M}\int_{\boldsymbol{\mathcal{D}}} v_{j,m}\frac{\partial v_{i,n}}{\partial x_j}\psi_m(\boldsymbol{\xi})\psi_n(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\,d\boldsymbol{\xi}\\
&+\sum_{m=0}^{M}\int_{\boldsymbol{\mathcal{D}}}\frac{\partial p_m}{\partial x_i}\psi_m(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\,d\boldsymbol{\xi}\\
&-\sum_{m=0}^{M}\int_{\boldsymbol{\mathcal{D}}}\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,m}}{\partial x_j}+\frac{\partial v_{j,m}}{\partial x_i}\right)\psi_m(\boldsymbol{\xi})\right]\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\,d\boldsymbol{\xi}\\
&=\sum_{m=0}^{M}\sum_{n=0}^{M} v_{j,m}\frac{\partial v_{i,n}}{\partial x_j}\langle\psi_m(\boldsymbol{\xi}),\psi_n(\boldsymbol{\xi})\rangle_w\psi_k(\boldsymbol{\xi})+\sum_{m=0}^{M}\frac{\partial p_m}{\partial x_i}\langle\psi_m(\boldsymbol{\xi}),\psi_k(\boldsymbol{\xi})\rangle_w\\
&-\sum_{m=0}^{M}\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,m}}{\partial x_j}+\frac{\partial v_{j,m}}{\partial x_i}\right)\right]\langle\psi_m(\boldsymbol{\xi}),\psi_k(\boldsymbol{\xi})\rangle_w = 0, \qquad i=1,2(,3)
\end{aligned}\tag{7.23}
$$

$$
\begin{aligned}
{}^{k}R^{T^F} = &\sum_{m=0}^{M}\sum_{n=0}^{M}\int_{\boldsymbol{\mathcal{D}}}\rho^F c_p v_{j,m}\psi_m(\boldsymbol{\xi})\frac{\partial T_n^F}{\partial x_j}\psi_n(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\,d\boldsymbol{\xi}\\
&-\sum_{m=0}^{M}\int_{\boldsymbol{\mathcal{D}}}\frac{\partial}{\partial x_j}\left[k^F\frac{\partial T_m^F}{\partial x_j}\psi_m(\boldsymbol{\xi})\right]\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\,d\boldsymbol{\xi}\\
&=\sum_{m=0}^{M}\sum_{n=0}^{M}\rho^F c_p v_{j,m}\frac{\partial T_n^F}{\partial x_j}\langle\psi_m(\boldsymbol{\xi}),\psi_n(\boldsymbol{\xi}),\psi_k(\boldsymbol{\xi})\rangle_w\\
&-\sum_{m=0}^{M}\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_m^F}{\partial x_j}\right)\langle\psi_m(\boldsymbol{\xi}),\psi_k(\boldsymbol{\xi})\rangle_w = 0
\end{aligned}\tag{7.24}
$$

$$
\begin{aligned}
{}^{k}R^{T^S} = &-\sum_{m=0}^{M}\int_{\boldsymbol{\mathcal{D}}}\frac{\partial}{\partial x_j}\left[k^S\frac{\partial T_m^S}{\partial x_j}\psi_m(\boldsymbol{\xi})\right]\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\,d\boldsymbol{\xi}\\
&=-\sum_{m=0}^{M}\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_m^S}{\partial x_j}\right)\langle\psi_m(\boldsymbol{\xi}),\psi_k(\boldsymbol{\xi})\rangle_w = 0
\end{aligned}\tag{7.25}
$$

where k denotes the index of the orthogonal polynomial $\psi_k(\boldsymbol{\xi})$, defined in eq. 7.13, which the primal PDEs are projected on. By evaluating the double and triple inner products of the multivariate Hermite polynomials $\langle\psi_m(\boldsymbol{\xi}),\psi_k(\boldsymbol{\xi})\rangle_w$ and $\langle\psi_m(\boldsymbol{\xi}),\psi_n(\boldsymbol{\xi}),\psi_k(\boldsymbol{\xi})\rangle_w$, see Appendix D, 30 equations with 30 unknown PCE coefficient fields arise. The PDEs, grouped according to the k index value, read

- $\underline{k=0:}$

$$
\begin{aligned}
{}^{0}R^{p} &= -\frac{\partial v_{j,0}}{\partial x_j} = 0 && \text{(7.26a)}\\
{}^{0}R^{v}_{i} &= v_{j,0}\frac{\partial v_{i,0}}{\partial x_j} + v_{j,1}\frac{\partial v_{i,1}}{\partial x_j} + v_{j,2}\frac{\partial v_{i,2}}{\partial x_j} + v_{j,3}\frac{\partial v_{i,3}}{\partial x_j} + 2v_{j,4}\frac{\partial v_{i,4}}{\partial x_j} + 2v_{j,5}\frac{\partial v_{i,5}}{\partial x_j} \\
&\quad + \frac{\partial p_0}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,0}}{\partial x_j} + \frac{\partial v_{j,0}}{\partial x_i}\right)\right] = 0 && \text{(7.26b)}\\
{}^{0}R^{T^F} &= \rho^F c_p\left(v_{j,0}\frac{\partial T_0^F}{\partial x_j} + v_{j,1}\frac{\partial T_1^F}{\partial x_j} + v_{j,2}\frac{\partial T_2^F}{\partial x_j} + v_{j,3}\frac{\partial T_3^F}{\partial x_j} + 2v_{j,4}\frac{\partial T_4^F}{\partial x_j}\right. \\
&\quad \left. + 2v_{j,5}\frac{\partial T_5^F}{\partial x_j}\right) - \frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_0^F}{\partial x_j}\right) = 0 && \text{(7.26c)}\\
{}^{0}R^{T^S} &= -\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_0^S}{\partial x_j}\right) = 0 && \text{(7.26d)}
\end{aligned}
$$

- $\underline{k=1:}$

$$
\begin{aligned}
{}^{1}R^{p} &= -\frac{\partial v_{j,1}}{\partial x_j} = 0 && \text{(7.27a)}\\
{}^{1}R^{v}_{i} &= v_{j,1}\frac{\partial v_{i,0}}{\partial x_j} + (v_{j,0}+2v_{j,4})\frac{\partial v_{i,1}}{\partial x_j} + v_{j,3}\frac{\partial v_{i,2}}{\partial x_j} + v_{j,2}\frac{\partial v_{i,3}}{\partial x_j} + 2v_{j,1}\frac{\partial v_{i,4}}{\partial x_j} \\
&\quad + \frac{\partial p_1}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,1}}{\partial x_j} + \frac{\partial v_{j,1}}{\partial x_i}\right)\right] = 0 && \text{(7.27b)}\\
{}^{1}R^{T^F} &= \rho^F c_p\left[v_{j,1}\frac{\partial T_0^F}{\partial x_j} + (v_{j,0}+2v_{j,4})\frac{\partial T_1^F}{\partial x_j} + v_{j,3}\frac{\partial T_2^F}{\partial x_j} + v_{j,2}\frac{\partial T_3^F}{\partial x_j}\right. \\
&\quad \left. + 2v_{j,1}\frac{\partial T_4^F}{\partial x_j}\right] - \frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_1^F}{\partial x_j}\right) = 0 && \text{(7.27c)}\\
{}^{1}R^{T^S} &= -\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_1^S}{\partial x_j}\right) = 0 && \text{(7.27d)}
\end{aligned}
$$

- $\underline{k=2:}$

$$
\begin{aligned}
{}^{2}R^{p} &= -\frac{\partial v_{j,2}}{\partial x_j} = 0 && \text{(7.28a)}\\
{}^{2}R^{v}_{i} &= v_{j,2}\frac{\partial v_{i,0}}{\partial x_j} + v_{j,3}\frac{\partial v_{i,1}}{\partial x_j} + (2v_{j,5}+v_{j,0})\frac{\partial v_{i,2}}{\partial x_j} + v_{j,1}\frac{\partial v_{i,3}}{\partial x_j} + 2v_{j,2}\frac{\partial v_{i,5}}{\partial x_j} \\
&\quad + \frac{\partial p_2}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,2}}{\partial x_j} + \frac{\partial v_{j,2}}{\partial x_i}\right)\right] = 0 && \text{(7.28b)}\\
{}^{2}R^{T^F} &= \rho^F c_p\left[v_{j,2}\frac{\partial T_0^F}{\partial x_j} + v_{j,3}\frac{\partial T_1^F}{\partial x_j} + (2v_{j,5}+v_{j,0})\frac{\partial T_2^F}{\partial x_j} + v_{j,1}\frac{\partial T_3^F}{\partial x_j}\right. \\
&\quad \left. + 2v_{j,2}\frac{\partial T_5^F}{\partial x_j}\right] - \frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_2^F}{\partial x_j}\right) = 0 && \text{(7.28c)}\\
{}^{2}R^{T^S} &= -\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_2^S}{\partial x_j}\right) = 0 && \text{(7.28d)}
\end{aligned}
$$

- $\underline{k=3}$:

$$
\begin{aligned}
{}^{3}R^{p} &= -\frac{\partial v_{j,3}}{\partial x_j} = 0 && \text{(7.29a)}\\
{}^{3}R^{v}_{i} &= v_{j,3}\frac{\partial v_{i,0}}{\partial x_j} + v_{j,2}\frac{\partial v_{i,1}}{\partial x_j} + v_{j,1}\frac{\partial v_{i,2}}{\partial x_j} + (v_{j,0}+2v_{j,4}+2v_{j,5})\frac{\partial v_{i,3}}{\partial x_j} \\
&\quad + 2v_{j,3}\frac{\partial v_{i,4}}{\partial x_j} + 2v_{j,3}\frac{\partial v_{i,5}}{\partial x_j} + \frac{\partial p_3}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,3}}{\partial x_j} + \frac{\partial v_{j,3}}{\partial x_i}\right)\right] = 0 && \text{(7.29b)}\\
{}^{3}R^{T^F} &= \rho^F c_p\left[v_{j,3}\frac{\partial T_0^F}{\partial x_j} + v_{j,2}\frac{\partial T_1^F}{\partial x_j} + v_{j,1}\frac{\partial T_2^F}{\partial x_j} + (v_{j,0}+2v_{j,4}+2v_{j,5})\frac{\partial T_3^F}{\partial x_j}\right. \\
&\quad \left. + 2v_{j,3}\frac{\partial T_4^F}{\partial x_j} + 2v_{j,3}\frac{\partial T_5^F}{\partial x_j}\right] - \frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_3^F}{\partial x_j}\right) = 0 && \text{(7.29c)}\\
{}^{3}R^{T^S} &= -\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_3^S}{\partial x_j}\right) = 0 && \text{(7.29d)}
\end{aligned}
$$

- $\underline{k=4}$:

$$
\begin{aligned}
{}^{4}R^{p} &= -\frac{\partial v_{j,4}}{\partial x_j} = 0 && \text{(7.30a)}\\
{}^{4}R^{v}_{i} &= v_{j,4}\frac{\partial v_{i,0}}{\partial x_j} + v_{j,1}\frac{\partial v_{i,1}}{\partial x_j} + v_{j,3}\frac{\partial v_{i,3}}{\partial x_j} + (v_{j,0}+4v_{j,4})\frac{\partial v_{i,4}}{\partial x_j} + \frac{\partial p_4}{\partial x_j}
\end{aligned}
$$

$$-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,4}}{\partial x_j}+\frac{\partial v_{j,4}}{\partial x_i}\right)\right]=0 \tag{7.30b}$$

$${}^{4}R^{T^F}=\rho^F c_p\left[v_{j,4}\frac{\partial T_0^F}{\partial x_j}+v_{j,1}\frac{\partial T_1^F}{\partial x_j}+v_{j,3}\frac{\partial T_3^F}{\partial x_j}+(v_{j,0}+4v_{j,4})\frac{\partial T_4^F}{\partial x_j}\right]$$
$$-\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_4^F}{\partial x_j}\right)=0 \tag{7.30c}$$

$${}^{4}R^{T^S}=-\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_4^S}{\partial x_j}\right)=0 \tag{7.30d}$$

- $\underline{k=5}$:

$${}^{5}R^{p}=-\frac{\partial v_{j,5}}{\partial x_j}=0 \tag{7.31a}$$

$${}^{5}R_i^{v}=v_{j,5}\frac{\partial v_{i,0}}{\partial x_j}+v_{j,2}\frac{\partial v_{i,2}}{\partial x_j}+v_{j,3}\frac{\partial v_{i,3}}{\partial x_j}+(v_{j,0}+4v_{j,5})\frac{\partial v_{i,5}}{\partial x_j}+\frac{\partial p_5}{\partial x_j}$$
$$-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,5}}{\partial x_j}+\frac{\partial v_{j,5}}{\partial x_i}\right)\right]=0 \tag{7.31b}$$

$${}^{5}R^{T^F}=\rho^F c_p\left[v_{j,5}\frac{\partial T_0^F}{\partial x_j}+v_{j,2}\frac{\partial T_2^F}{\partial x_j}+v_{j,3}\frac{\partial T_3^F}{\partial x_j}+(v_{j,0}+4v_{j,5})\frac{\partial T_5^F}{\partial x_j}\right]$$
$$-\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_5^F}{\partial x_j}\right)=0 \tag{7.31c}$$

$${}^{5}R^{T^S}=-\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_5^S}{\partial x_j}\right)=0 \tag{7.31d}$$

Assuming that the cost of solving the deterministic PDEs is equal to that of solving the iPCE PDEs corresponding to each value of k, then the cost for evaluating μ_J and σ_J with the iPCE method is equivalent to $M+1=6$ EFS.

A segregated algorithm is used to solve eqs. 7.26 to 7.31. Within each iteration of the primal iPCE solver, each system of equations corresponding to a value of $k=0,...,5$ is solved by means of the SIMPLE algorithm, [132]. The solution of the iPCE heat transfer PDEs could start after making the iPCE continuity and momentum equations converge, instead of solving all of them simultaneously. All convection terms are discretized using second-order upwind schemes. Diffusion fluxes are discretized using central schemes and correction for non-orthogonality is additionally used. Also, spatial gradients are computed through the Gauss divergence scheme, using a linear interpolation from the cell-centers to the cell-faces.

7.2.2 iPCE Primal Boundary Conditions

Different procedures are followed to expand boundary conditions that include the moments of the uncertain variables and those that don't.

7.2.2.1 Conditions not Including Uncertain Variable Moments

Since most of the deterministic primal boundary conditions are either of Dirichlet or Neumann type, the iPCE conditions are derived as generally as possible for any quantity Φ; FSI conditions are studied separately.

Dirichlet conditions: Let c being a constant non-stochastic quantity, then

$$\Phi = c \Rightarrow \sum_{m=0}^{M} \Phi_m \psi_m(\boldsymbol{\xi}) = c \Rightarrow \sum_{m=0}^{M} \int_{\boldsymbol{\mathcal{D}}} \Phi_m \psi_m(\boldsymbol{\xi}) \psi_k(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \int_{\boldsymbol{\mathcal{D}}} c \psi_k(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$
$$\Rightarrow \Phi_m \langle \psi_m(\boldsymbol{\xi}), \psi_k(\boldsymbol{\xi}) \rangle_w = \Phi_m ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^k = c \int_{\boldsymbol{\mathcal{D}}} \psi_k(\boldsymbol{\xi}) \psi_0(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$
$$= c ||\psi_0(\boldsymbol{\xi})||_w \delta_k^0 \tag{7.32}$$

Thus, a Dirichlet condition for any deterministic primal variable Φ results to the same condition for Φ_0 and zero Dirichlet conditions for any $\Phi_i,\ i > 0$.

Neumann conditions: For the Neumann condition $\frac{\partial \Phi}{\partial n} = c$, where c is constant non-stochastic quantity, in case uncertainties do not affect the nodal coordinates

$$\frac{\partial \Phi}{\partial n} = c \Rightarrow \sum_{m=0}^{M} \frac{\partial \Phi_m}{\partial n} \psi_m(\boldsymbol{\xi}) = c \Rightarrow \sum_{m=0}^{M} \int_{\boldsymbol{\mathcal{D}}} \frac{\partial \Phi_m}{\partial n} \psi_m(\boldsymbol{\xi}) \psi_k(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \int_{\boldsymbol{\mathcal{D}}} c \psi_k(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$
$$\Rightarrow \frac{\partial \Phi_m}{\partial n} \langle \psi_m(\boldsymbol{\xi}), \psi_k(\boldsymbol{\xi}) \rangle_w = \frac{\partial \Phi_m}{\partial n} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^k = c \int_{\boldsymbol{\mathcal{D}}} \psi_k(\boldsymbol{\xi}) \psi_0(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = c \delta_k^0 \tag{7.33}$$

Thus, when a Neumann condition is to a deterministic field Φ, the same Neumann condition holds for Φ_0, whereas a zero Neumann condition is imposed to any $\Phi_i,\ i > 0$.

FSI conditions: By expanding $T^F,\ T^S$ into their coefficients, eqs. 3.9 to 3.10 are expanded as

$$\sum_{m=0}^{M} T_m^S \Big|_{\overline{S}^S} \psi_m(\boldsymbol{\xi}) = \sum_{m=0}^{M} T_m^F \Big|_{\overline{S}^F} \psi_m(\boldsymbol{\xi})$$

$$\Rightarrow \sum_{m=0}^{M} \int_{\mathcal{D}} T_m^S\big|_{\overline{S}^S} \psi_m(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \sum_{m=0}^{M} \int_{\mathcal{D}} T_m^F\big|_{\overline{S}^F} \psi_m(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$
$$\Rightarrow T_m^S\big|_{\overline{S}^S} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^k = T_m^F\big|_{\overline{S}^F} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^k$$
$$\Rightarrow T_m^S\big|_{\overline{S}^S} = T_m^F\big|_{\overline{S}^F} \tag{7.34}$$

and

$$\sum_{m=0}^{M} k^S \frac{\partial T_m^S}{\partial n}\bigg|_{\overline{S}^S} \psi_m(\boldsymbol{\xi}) = -\sum_{m=0}^{M} k^F \frac{\partial T_m^F}{\partial n}\bigg|_{\overline{S}^F} \psi_m(\boldsymbol{\xi})$$
$$\Rightarrow \sum_{m=0}^{M} \int_{\mathcal{D}} k^S \frac{\partial T_m^S}{\partial n}\bigg|_{\overline{S}^S} \psi_m(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = -\sum_{m=0}^{M} \int_{\mathcal{D}} k^F \frac{\partial T_m^F}{\partial n}\bigg|_{\overline{S}^F} \psi_m(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$
$$\Rightarrow k^S \frac{\partial T_m^S}{\partial n}\bigg|_{\overline{S}^S} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^k = -k^F \frac{\partial T_m^F}{\partial n}\bigg|_{\overline{S}^F} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^k$$
$$\Rightarrow k^S \frac{\partial T_m^S}{\partial n}\bigg|_{\overline{S}^S} = -\frac{\partial T_m^F}{\partial n}\bigg|_{\overline{S}^F} \tag{7.35}$$

7.2.2.2 Conditions Including Uncertain Variable Moments

Below two problems, the first with environmental and the second with manufacturing uncertainties, are studied. In both of them, the magnitude of the fluid velocity at the inlet is the first uncertain variable. The second uncertain variable is either T^F at the fluid domain inlet or the thickness of the thermal insulation L between Ω^F and Ω^S, respectively. The derivation of the relevant boundary conditions for the iPCE coefficients is based on formulas presented in Appendix D, eqs. 7.13 and the fact that the skewness of the uncertain variables, which are governed by a normal distribution, is zero.

Inlet Velocity Magnitude: Let the velocity magnitude $v = |v_i|$ follow the normal distribution $N(\mu_v, \sigma_v^2)$, with the inlet flow angle being unaffected. Then, each iPCE coefficient of v is acquired as

$$v = \sum_{m=0}^{M} v_m \psi_m(\boldsymbol{\xi}) = \mu_v + \sigma_v \xi_1$$
$$\Rightarrow \sum_{m=0}^{M} \int_{\mathcal{D}} v_m \psi_m(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \int_{\mathcal{D}} (\mu_v + \sigma_v \xi_1)\, \psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$
$$\Rightarrow v_m \langle \psi_m(\boldsymbol{\xi}), \psi_k(\boldsymbol{\xi}) \rangle_w \delta_m^k = v_k ||\psi_k(\boldsymbol{\xi})||_w^2 = \int_{\mathcal{D}} (\mu_v + \sigma_v \xi_1)\, \psi_k(\boldsymbol{\xi})W(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{7.36}$$

Thus

- $\underline{k=0:}$

$$v_0||\psi_0(\boldsymbol{\xi})||_w^2 = \int_{\boldsymbol{\mathcal{D}}} (\mu_v + \sigma_v \xi_1)\, \psi_0(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \int_{\boldsymbol{\mathcal{D}}} (\mu_v + \sigma_v \xi_1)\, W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$
$$\Rightarrow v_0 = \int_{\boldsymbol{\mathcal{D}}} \mu_v W(\boldsymbol{\xi})\, d\boldsymbol{\xi} + \int_{\boldsymbol{\mathcal{D}}} \sigma_v \xi_1 W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \mu_v \tag{7.37}$$

- $\underline{k=1:}$

$$v_1||\psi_1(\boldsymbol{\xi})||_w^2 = \int_{\boldsymbol{\mathcal{D}}} (\mu_v + \sigma_v \xi_1)\, \psi_1(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \int_{\boldsymbol{\mathcal{D}}} (\mu_v + \sigma_v \xi_1)\, \xi_1 W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$
$$\Rightarrow v_1 = \int_{\boldsymbol{\mathcal{D}}} \mu_v \xi_1 W(\boldsymbol{\xi})\, d\boldsymbol{\xi} + \int_{\boldsymbol{\mathcal{D}}} \sigma_v \xi_1^2 W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \sigma_v \tag{7.38}$$

- $\underline{k=2,...,5:}$

$$v_k||\psi_k(\boldsymbol{\xi})||_w^2 = \int_{\boldsymbol{\mathcal{D}}} (\mu_v + \sigma_v \xi_1)\, \psi_k(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = 0 \tag{7.39}$$

Inlet $\boldsymbol{T^F}$: Let T^F follow the normal distribution $N(\mu_T, \sigma_T^2)$. The derivation of the boundary conditions follows a similar procedure as for the velocity magnitude. By doing so, it is proven that

$$T_k^F = \begin{cases} \mu_T, & \text{if } k = 0 \\ \sigma_T, & \text{if } k = 2 \\ 0, & \text{if } k = 1, 3, 4, 5 \end{cases} \tag{7.40}$$

Insulation Thickness $\boldsymbol{L}$: The uncertainty in thermal insulation thickness L following the normal distribution $N(\mu_L, \sigma_L^2)$ is modeled by starting from the equations expressing the preservation of the heat flux along each point of the FSI, i.e.

$$\begin{aligned} k^F \frac{\partial T^F}{\partial n} &= -k^c \frac{T^F - T^S}{L} \\ k^S \frac{\partial T^S}{\partial n} &= -k^c \frac{T^S - T^F}{L} \end{aligned} \tag{7.41}$$

where k^c is the thermal conductivity of the insulation material and T^F, T^S the fluid and solid temperatures at the FSI. The development of the iPCE conditions is presented below for the fluid; the development for the solid is similar. Let, $\Delta T^{F,S} = T^F - T^S$, then

$$\begin{aligned} Q^F &= -k^c \frac{\Delta T^{F,S}}{L} \Rightarrow (\mu_L + \sigma_L \xi_2) \sum_{m=0}^{M} Q_m^F \psi_m(\boldsymbol{\xi}) = -\sum_{m=0}^{M} k^c \Delta T_m^{F,S} \psi_m(\boldsymbol{\xi}) \\ &\Rightarrow \sum_{m=0}^{M} \int_{\boldsymbol{\mathcal{D}}} (\mu_L + \sigma_L \xi_2)\, Q_m^F \psi_m(\boldsymbol{\xi}) \psi_k(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} \\ &= -\sum_{m=0}^{M} \int_{\boldsymbol{\mathcal{D}}} k^c \Delta T_m^{F,S} \psi_m(\boldsymbol{\xi}) \psi_k(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} \end{aligned} \tag{7.42}$$

Hence,

$$\begin{aligned} &\sum_{m=0}^{M} Q_m^F \left[\mu_L \langle \psi_m(\boldsymbol{\xi}), \psi_k(\boldsymbol{\xi}) \rangle_w + \sigma_L \langle \psi_2(\boldsymbol{\xi}), \psi_m(\boldsymbol{\xi}), \psi_k(\boldsymbol{\xi}) \rangle_w \right] \\ &= -k^c \Delta T_m^{F,S} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^k \end{aligned} \tag{7.43}$$

Thus,

- $\underline{k=0:}$

$$\begin{aligned} &\sum_{m=0}^{M} Q_m^F \left[\mu_L \langle \psi_m(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}) \rangle_w + \sigma_L \langle \psi_2(\boldsymbol{\xi}), \psi_m(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}) \rangle_w \right] \\ &= -k^c \Delta T_m^{F,S} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^0 \\ &\Rightarrow \mu_L Q_0^F + \sigma_L Q_2^F = -k^c \Delta T_0^F \end{aligned} \tag{7.44}$$

- $\underline{k=1:}$

$$\begin{aligned} &\sum_{m=0}^{M} Q_m^F \left[\mu_L \langle \psi_m(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}) \rangle_w + \sigma_L \langle \psi_2(\boldsymbol{\xi}), \psi_m(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}) \rangle_w \right] \\ &= -k^c \Delta T_m^{F,S} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^1 \\ &\Rightarrow \mu_L Q_1^F + \sigma_L Q_3^F = -k^c \Delta T_1^F \end{aligned} \tag{7.45}$$

- $\underline{k=2:}$

$$\begin{aligned}
&\sum_{m=0}^{M} Q_m^F \left[\mu_L \langle \psi_m(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi})\rangle_w + \sigma_L \langle \psi_2(\boldsymbol{\xi}), \psi_m(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi})\rangle_w\right] \\
&= -k^c \Delta T_m^{F,S} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^2 \\
&\Rightarrow \mu_L Q_2^F + \sigma_L \left(Q_0^F + 2Q_5^F\right) = -k^c \Delta T_2^F
\end{aligned} \tag{7.46}$$

- $\underline{k=3}$:

$$\begin{aligned}
&\sum_{m=0}^{M} Q_m^F \left[\mu_L \langle \psi_m(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi})\rangle_w + \sigma_L \langle \psi_2(\boldsymbol{\xi}), \psi_m(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi})\rangle_w\right] \\
&= -k^c \Delta T_m^{F,S} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^3 \\
&\Rightarrow \mu_L Q_3^F + \sigma_L Q_1^F = -k^c \Delta T_3^F
\end{aligned} \tag{7.47}$$

- $\underline{k=4}$:

$$\begin{aligned}
&\sum_{m=0}^{M} Q_m^F \left[\mu_L \langle \psi_m(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi})\rangle_w + \sigma_L \langle \psi_2(\boldsymbol{\xi}), \psi_m(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi})\rangle_w\right] \\
&= -k^c \Delta T_m^{F,S} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^4 \\
&\Rightarrow \mu_L Q_4^F = -k^c \Delta T_4^F
\end{aligned} \tag{7.48}$$

- $\underline{k=5}$:

$$\begin{aligned}
&\sum_{m=0}^{M} Q_m^F \left[\mu_L \langle \psi_m(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi})\rangle_w + \sigma_L \langle \psi_2(\boldsymbol{\xi}), \psi_m(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi})\rangle_w\right] \\
&= -k^c \Delta T_m^{F,S} ||\psi_m(\boldsymbol{\xi})||_w^2 \delta_m^5 \\
&\Rightarrow \mu_L Q_5^F + \sigma_L Q_2^F = -k^c \Delta T_5^F
\end{aligned} \tag{7.49}$$

Eqs. 7.76 to 7.80 are used to impose the boundary conditions for T_m^F. By substituting $Q^S = -Q^F$ and $\Delta T^{S,F} = T^S - T^F = -\Delta T^{F,S}$ into eqs. 7.76 to 7.80, the same boundary conditions for T_m^S arise. It is observed that the derived boundary conditions expressed by eqs. 7.76 to 7.80 couple PCE temperature coefficients of different indices, compared to the rest of the PCE boundary conditions.

7.2.2.3 Computation of the Mean Value and Standard Deviation of the QoI

Here, the expressions of the mean value and standard deviation of the QoI investigated in this thesis, which are the mean solid temperature, eq. B.10, and the

heat flux across the FSI, eq. B.12, are derived. The ν-th statistical moment of the QoI J is given by

$$\langle y^{\nu} \rangle = \int_{\boldsymbol{\mathcal{D}}} J^{\nu} W(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{7.50}$$

which, for $\nu = 1$ and $\nu = 2$ yields μ_J and $\mu_J^2 + \sigma_J^2$, respectively. The expressions of the mean value and standard deviation of J^{meanT} are, thus, acquired as

$$\begin{aligned}
\mu_{JmeanT} &= \int_{\boldsymbol{\mathcal{D}}} \left(\int_{\Omega^S} T^S d\Omega \right) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \int_{\boldsymbol{\mathcal{D}}} \int_{\Omega^S} \left[\sum_{m=0}^{M} T_m^S \psi_m(\boldsymbol{\xi}) \right] W(\boldsymbol{\xi})\, d\Omega d\boldsymbol{\xi} \\
&= \int_{\Omega^S} \left[\sum_{m=0}^{M} \int_{\boldsymbol{\mathcal{D}}} T_m^S \psi_m(\boldsymbol{\xi}) \psi_0(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} \right] d\Omega = \int_{\Omega^S} T_0^S d\Omega
\end{aligned} \tag{7.51}$$

and

$$\begin{aligned}
\sigma_{JmeanT}^2 &= \int_{\boldsymbol{\mathcal{D}}} \left(\int_{\Omega^S} T^S d\Omega \right)^2 W(\boldsymbol{\xi})\, d\boldsymbol{\xi} - \mu_{JmeanT}^2 \\
&= \int_{\boldsymbol{\mathcal{D}}} \int_{\Omega^S} \left(\sum_{m=0}^{M} T_m^S \psi_m(\boldsymbol{\xi}) \right)^2 W(\boldsymbol{\xi})\, d\Omega d\boldsymbol{\xi} - \mu_{JmeanT}^2 \\
&= \int_{\boldsymbol{\mathcal{D}}} \int_{\Omega^S} \left(\sum_{m_1=0}^{M} \sum_{m_2=0}^{M} T_{m_1}^S T_{m_2}^S \psi_{m_1}(\boldsymbol{\xi}) \psi_{m_2}(\boldsymbol{\xi}) \right) W(\boldsymbol{\xi})\, d\Omega d\boldsymbol{\xi} - \int_{\Omega^S} T_0^{S^2} d\Omega \\
&= \sum_{m=1}^{M} \int_{\Omega^S} T_m^{S^2} ||\psi_m(\boldsymbol{\xi})||_w^2 d\Omega
\end{aligned} \tag{7.52}$$

Hence,

$$\sigma_{JmeanT} = \sqrt{\sum_{m=1}^{M} \int_{\Omega^S} T_m^{S^2} ||\psi_m(\boldsymbol{\xi})||_w^2 d\Omega} \tag{7.53}$$

The moments of J^{HF} are derived through a similar development and read

$$\mu_{JHF} = \int_{\Omega^S} k^S \frac{\partial T_0^S}{\partial n} d\Omega \tag{7.54}$$

and

$$\sigma_{JHF} = \sqrt{\sum_{m=1}^{M} \int_{\Omega^S} \left(k^S \frac{\partial T_m^S}{\partial n} \right)^2 ||\psi_m(\boldsymbol{\xi})||_w^2 d\Omega} \tag{7.55}$$

7.2.3 Continuous Adjoint Method for the iPCE Primal Problem

The adjoint method is developed here for the iPCE primal PDEs, which were derived for CHT problems in section 7.2.1. The outcome of this development will allow the computation of SDs μ_J and σ_J, where J is a QoI. The mathematical development takes into account that the inlet velocity magnitude and temperature as well as the thermal insulation thickness are the uncertain variables and not the nodal coordinates x_i. In particular, the adjoint method is developed for the function J'

$$J' = \gamma_0 \mu_J + \gamma_1 \sigma_J = \sum_{D=F,S} \left(J_S'^D + J_\Omega'^D \right) \tag{7.56}$$

where γ_0, γ_1 are user-defined weights. Thus, by selecting the values of γ_0, γ_1, the SDs of either μ_J or σ_J (or any other linear combination), are computed. The gradient of J' w.r.t. b_n is developed by expanding eqs. 3.17, if defined for the fluid, or 3.18, if defined for the solid, i.e.

$$\begin{aligned}
\frac{\delta J'^F}{\delta b_n} = \sum_{l=0}^{5} \Bigg\{ & \int_{S^F} \left(\frac{\partial j'^F_{S_k}}{\partial v_{i,l}} n_k + j'^{F,v}_{S,i} \right) \frac{\delta v_{i,l}}{\delta b_n} dS + \int_{S^F} \left(\frac{\partial j'^F_{S_i}}{\partial p_l} n_i + j'^{F,p}_{S} \right) \frac{\delta p_l}{\delta b_n} dS \\
& + \int_{S^F} \frac{\partial j'^F_{S_i}}{\partial \tau_{kj,l}} n_i \frac{\delta \tau_{kj,l}}{\delta b_n} dS + \int_{S^F} \left(\frac{\partial j'^F_{S_i}}{\partial T_l^F} n_i + j'^{F,T}_{S,l} \right) \frac{\delta T_l^F}{\delta b_n} dS + \int_{S^F} \frac{\partial j'^F_S}{\partial Q_l^F} \frac{\delta Q_l^F}{\delta b_n} dS \\
& + \int_{\Omega^F} j'^{F,v}_{\Omega,i} \frac{\delta v_{i,l}}{\delta b_n} d\Omega + \int_{\Omega^F} j'^{F,p}_{\Omega} \frac{\delta p_l}{\delta b_n} d\Omega + \int_{\Omega^F} j'^{F,T}_{\Omega} \frac{\delta T_l^F}{\delta b_n} d\Omega \Bigg\} + \int_{S^F_{W_p}} j'^{F,g}_{S_{i,k}} \frac{\delta x_k}{\delta b_n} n_i dS \\
& + \int_{S^F_{W_p}} j'^F_{S_i} \frac{\delta n_i}{\delta b_n} dS + \int_{S^F_{W_p}} j'^F_{S_i} n_i \frac{\delta (dS)}{\delta b_n} + \int_{S^F_{W_p}} j'^F_S \frac{\delta (dS)}{\delta b_n} + \int_{\Omega^F} j'^F_\Omega \frac{\partial}{\partial x_k} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{7.57}$$

and

$$\begin{aligned}
\frac{\delta J'^S}{\delta b_n} = \sum_{l=0}^{5} \Bigg\{ & \int_{S^S} \left(\frac{\partial j'^S_{S_i}}{\partial T_l^S} n_i + j'^{S,T}_{S} \right) \frac{\delta T_l^S}{\delta b_n} dS + \int_{S^S} \frac{\partial j'^S_{S_i}}{\partial Q_l^S} \frac{\delta Q_l^S}{\delta b_n} dS + \int_{\Omega^S} j'^{S,T}_{\Omega} \frac{\delta T_l^S}{\delta b_n} d\Omega \Bigg\} \\
& + \int_{S^S_{W_p}} j'^S_S \frac{\delta (dS)}{\delta b_n} + \int_{S^S_{W_p}} j'^S_{S_i} n_i \frac{\delta (dS)}{\delta b_n} + \int_{S^S_{W_p}} j'^S_{S_i} \frac{\delta n_i}{\delta b_n} dS + \int_{S^S_{W_p}} j'^{S,g}_{S_i} \frac{\delta x_k}{\delta b_n} n_i dS \\
& + \int_{\Omega^S} \left(j'^S_\Omega + j'^{S,g}_\Omega \right) \frac{\partial}{\partial x_k} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{7.58}$$

7.2.3.1 iPCE Adjoint PDEs

The adjoint to the iPCE primal CHT PDEs starts by defining the Lagrangian L as

$$L = J' + \sum_{l=0}^{5} \left[\int_{\Omega^F} q_l \, {}^l R^p d\Omega + \int_{\Omega^F} u_{i,l} \, {}^l R^v_i d\Omega + \sum_{D=F,S} \left(\int_{\Omega^D} T^D_{a,l} \, {}^l R^{T^D} d\Omega \right) \right] \tag{7.59}$$

The derivative of the first term on the r.h.s. of the above equation w.r.t. b_n is given by eqs. 7.57 and 7.58. When differentiating the iPCE primal PDEs, eqs. 2.21, 2.23, 2.24, 2.25, 3.20a, 3.20b, 3.21a and 3.23 derived in sections 2,3 for CHT problems without uncertainties can be used here, enriched with indices corresponding to the iPCE coefficient fields, as

$$\begin{aligned}
-\int_{\Omega^F} q_\mu \frac{\partial v_{i,\nu}}{\partial x_i} d\Omega = &-\int_{S^F} q_\mu n_i \frac{\delta v_{i,\nu}}{\delta b_n} dS + \int_{\Omega^F} \frac{\partial q_\mu}{\partial x_i} \frac{\delta v_{i,\nu}}{\delta b_n} d\Omega \\
&+ \int_{\Omega^F} q_\mu \frac{\partial v_{i,\nu}}{\partial x_k} \frac{\partial}{\partial x_i} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{7.60a}$$

$$\begin{aligned}
\frac{\delta}{\delta b_n} \int_{\Omega^F} u_{i,\mu} v_{j,\nu} \frac{\partial v_{i,o}}{\partial x_j} d\Omega = &\int_{\Omega^F} u_{j,\mu} \frac{\partial v_{j,\nu}}{\partial x_i} \frac{\delta v_i}{\delta b_n} d\Omega + \int_{S^F} u_{i,\mu} v_{j,\nu} n_j \frac{\delta v_{i,o}}{\delta b_n} dS \\
&- \int_{\Omega^F} \frac{\partial (u_{i,\mu} v_{j,\nu})}{\partial x_j} \frac{\delta v_{i,o}}{\delta b_n} d\Omega \\
&- \int_{\Omega^F} u_{i,\mu} v_{j,\nu} \frac{\partial v_{i,o}}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{7.60b}$$

$$\begin{aligned}
\frac{\delta}{\delta b_n} \int_{\Omega^F} u_{i,\mu} \frac{\partial p_o}{\partial x_j} d\Omega = &\int_{S^F} u_{i,\mu} n_i \frac{\delta p_o}{\delta b_n} dS - \int_{\Omega^F} \frac{\partial u_{i,\mu}}{\partial x_i} \frac{\delta p_o}{\delta b_n} d\Omega \\
&- \int_{\Omega^F} u_{i,\mu} \frac{\partial p_o}{\partial x_k} \frac{\partial}{\partial x_i} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{7.60c}$$

$$\begin{aligned}
\frac{\delta}{\delta b_n} \int_{\Omega^F} u_{i,\mu} \tau_{ij,\nu} d\Omega = &-\int_{S^F} u_{i,\mu} n_j \frac{\delta \tau_{ij,\nu}}{\delta b_n} dS + \int_{\Omega^F} u_{i,\mu} \frac{\partial \tau_{ij,\nu}}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
&+ \int_{S^F} \nu_{eff} \left(\frac{\partial u_{i,\mu}}{\partial x_j} + \frac{\partial u_{j,\nu}}{\partial x_i} \right) n_j \frac{\delta v_{i,\nu}}{\delta b_n} dS \\
&- \int_{\Omega^F} \frac{\partial}{\partial x_j} \left[\nu_{eff} \left(\frac{\partial u_{i,\mu}}{\partial x_j} + \frac{\partial u_{j,\nu}}{\partial x_i} \right) \right] \frac{\delta v_{i,\nu}}{\delta b_n} d\Omega \\
&- \int_{\Omega^F} \nu_{eff} \left(\frac{\partial u_{i,\mu}}{\partial x_j} + \frac{\partial u_{j,\mu}}{\partial x_i} \right) \frac{\partial v_{i,\nu}}{\partial x_k} \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega
\end{aligned} \tag{7.60d}$$

$$\frac{\delta}{\delta b_n}\int_{\Omega^F}\rho^F c_p T^F_{a,\mu} v_{i,\nu}\frac{\partial T^F_o}{\partial x_i}d\Omega = \int_{\Omega^F}\rho^F c_p T^F_{a,\mu}\frac{\partial T^F_o}{\partial x_i}\frac{\delta v_{i,\nu}}{\delta b_n}d\Omega + \int_{S^F}\rho^F c_p T^F_{a,\mu} v_{i,\nu} n_i \frac{\delta T^F_o}{\delta b_n}dS$$
$$-\int_{\Omega^F}\frac{\partial}{\partial x_i}\left(\rho^F c_p T^F_{a,\mu} v_{i,\nu}\right)\frac{\delta T^F_o}{\delta b_n}d\Omega$$
$$-\int_{\Omega^F}\rho^F c_p T^F_{a,\mu} v_j \frac{\partial T^F_o}{\partial x_k}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \tag{7.60e}$$

$$\frac{\delta}{\delta b_n}\int_{\Omega^F} T^F_{a,\mu}\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T^F_\nu}{\partial x_j}\right)d\Omega = \int_{\Omega^F}\left[-k^F\frac{\partial T^F_{a,\mu}}{\partial x_j}\frac{\partial T^F_\nu}{\partial x_k} + T^F_{a,\mu}\frac{\partial}{\partial x_k}\left(k^F\frac{\partial T^F_\nu}{\partial x_j}\right)\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega$$
$$-\int_{S^F} T^F_{a,\mu} n_j \frac{\delta}{\delta b_n}\left(k^F\frac{\partial T^F_\nu}{\partial x_j}\right)dS + \int_{S^F} k^F\frac{\partial T^F_{a,\mu}}{\partial n}\frac{\delta T^F_\nu}{\delta b_n}dS$$
$$-\int_{\Omega^F}\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T^F_{a,\mu}}{\partial x_j}\right)\frac{\delta T^F_\nu}{\delta b_n}d\Omega \tag{7.60f}$$

$$\frac{\delta}{\delta b_n}\int_{\Omega^S} T^S_{a,\mu}\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T^S_\nu}{\partial x_j}\right)d\Omega = -\int_{S^S} T^S_{a,\mu} n_j \frac{\delta}{\delta b_n}\left(k^S\frac{\partial T^S_\nu}{\partial x_j}\right)dS + \int_{S^S} k^S\frac{\partial T^S_{a,\mu}}{\partial n}\frac{\delta T^S_\nu}{\delta b_n}dS$$
$$+\int_{\Omega^S}\left[T^S_{a,\mu}\frac{\partial}{\partial x_k}\left(k^S\frac{\partial T^S_\nu}{\partial x_j}\right) - k^S\frac{\partial T^S_\nu}{\partial x_k}\frac{\partial T^S_{a,\mu}}{\partial x_j}\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega$$
$$-\int_{\Omega^S}\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T^S_{a,\mu}}{\partial x_j}\right)\frac{\delta T^S_\nu}{\delta b_n}d\Omega \tag{7.60g}$$

To derive the field adjoint equations, boundary conditions and SDs expressions, the procedure is the same as for the CHT problems without uncertainties. By using the formulas for the inner products of the multivariate Hermite polynomials presented in Appendix D and zeroing, inside field integrals, the derivatives of the iPCE flow variables w.r.t. b_n, the following PDEs arise

- $\underline{k=0}$:

$${}^0R^q = -\frac{\partial u_{j,0}}{\partial x_j} + j^{F,p}_{\Omega,0} = 0 \tag{7.61a}$$

$${}^0R^u = u_{j,0}\frac{\partial v_{i,0}}{\partial x_j} + u_{j,1}\frac{\partial v_{i,1}}{\partial x_j} + u_{j,2}\frac{\partial v_{i,2}}{\partial x_j} + u_{j,3}\frac{\partial v_{i,3}}{\partial x_j} + 2u_{j,4}\frac{\partial v_{i,4}}{\partial x_j} + 2u_{j,5}\frac{\partial v_{i,5}}{\partial x_j}$$
$$-\frac{\partial\left(v_{j,0}u_{i,0}\right)}{\partial x_j} - \frac{\partial\left(v_{j,1}u_{i,1}\right)}{\partial x_j} - \frac{\partial\left(v_{j,2}u_{i,2}\right)}{\partial x_j} - \frac{\partial\left(v_{j,3}u_{i,3}\right)}{\partial x_j} - 2\frac{\partial\left(v_{j,4}u_{i,4}\right)}{\partial x_j} - 2\frac{\partial\left(v_{j,5}u_{i,5}\right)}{\partial x_j}$$
$$+\rho^F c_p\left(T^F_{a,0}\frac{\partial T^F_0}{\partial x_i} + T^F_{a,1}\frac{\partial T^F_1}{\partial x_i} + T^F_{a,2}\frac{\partial T^F_2}{\partial x_i} + T^F_{a,3}\frac{\partial T^F_3}{\partial x_i} + 2T^F_{a,4}\frac{\partial T^F_4}{\partial x_i} + 2T^F_{a,5}\frac{\partial T^F_5}{\partial x_i}\right)$$
$$+\frac{\partial q_0}{\partial x_i} - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial u_{i,0}}{\partial x_j} + \frac{\partial u_{j,0}}{\partial x_i}\right)\right] + j^{F,v}_{\Omega,i,0} = 0, \quad i = 1,2(,3) \tag{7.61b}$$

$$
\begin{aligned}
{}^{0}R^{T_a^F} = &-\rho^F c_p \left[\frac{\partial \left(v_{j,0} T_{a,0}^F \right)}{\partial x_j} + \frac{\partial \left(v_{j,1} T_{a,1}^F \right)}{\partial x_j} + \frac{\partial \left(v_{j,2} T_{a,2}^F \right)}{\partial x_j} + \frac{\partial \left(v_{j,3} T_{a,3}^F \right)}{\partial x_j} \right. \\
&\left. + 2\frac{\partial \left(v_{j,4} T_{a,4}^F \right)}{\partial x_j} + 2\frac{\partial \left(v_{j,5} T_{a,5}^F \right)}{\partial x_j} \right] - \frac{\partial}{\partial x_j} \left(k^F \frac{\partial T_{a,0}^F}{\partial x_j} \right) + j_{\Omega,0}^{F,T} = 0
\end{aligned} \tag{7.61c}
$$

$$
{}^{0}R^{T_a^S} = -\frac{\partial}{\partial x_j} \left(k^S \frac{\partial T_{a,0}^S}{\partial x_j} \right) + j_{\Omega,0}^{S,T} = 0 \tag{7.61d}
$$

- $\underline{k=1:}$

$$
{}^{1}R^{q} = -\frac{\partial u_{j,1}}{\partial x_j} + j_{\Omega,1}^{F,p} = 0 \tag{7.62a}
$$

$$
\begin{aligned}
{}^{1}R^{u} = &\, u_{j,1} \frac{\partial v_{i,0}}{\partial x_j} + \left(u_{j,0} + 2u_{j,4} \right) \frac{\partial v_{i,1}}{\partial x_j} + u_{j,3} \frac{\partial v_{i,2}}{\partial x_j} + u_{j,2} \frac{\partial v_{i,3}}{\partial x_j} + 2u_{j,1} \frac{\partial v_{i,4}}{\partial x_j} \\
&- \frac{\partial \left(v_{j,1} u_{i,0} \right)}{\partial x_j} - \frac{\partial \left(\left(v_{j,0} + 2v_{j,4} \right) u_{i,1} \right)}{\partial x_j} - \frac{\partial \left(v_{j,3} u_{i,2} \right)}{\partial x_j} - \frac{\partial \left(v_{j,2} u_{i,3} \right)}{\partial x_j} - 2\frac{\partial \left(v_{j,1} u_{i,4} \right)}{\partial x_j} \\
&+ \rho^F c_p \left(T_{a,1}^F \frac{\partial T_0^F}{\partial x_i} + T_{a,0}^F \frac{\partial T_1^F}{\partial x_i} + T_{a,3}^F \frac{\partial T_2^F}{\partial x_i} + T_{a,2}^F \frac{\partial T_3^F}{\partial x_i} + 2T_{a,1}^F \frac{\partial T_4^F}{\partial x_i} + 2T_{a,4}^F \frac{\partial T_1^F}{\partial x_i} \right) \\
&+ \frac{\partial q_1}{\partial x_i} - \frac{\partial}{\partial x_j} \left[\nu \left(\frac{\partial u_{i,1}}{\partial x_j} + \frac{\partial u_{j,1}}{\partial x_i} \right) \right] + j_{\Omega,i,1}^{F,v} = 0, \quad i = 1, 2(, 3)
\end{aligned} \tag{7.62b}
$$

$$
\begin{aligned}
{}^{1}R^{T_a^F} = &-\rho^F c_p \left[\frac{\partial \left(v_{j,1} T_0^F \right)}{\partial x_j} + \frac{\partial \left(\left(v_{j,0} + 2v_{j,4} \right) T_1^F \right)}{\partial x_j} + \frac{\partial \left(v_{j,3} T_2^F \right)}{\partial x_j} + \frac{\partial \left(v_{j,2} T_3^F \right)}{\partial x_j} \right. \\
&\left. + 2\frac{\partial \left(v_{j,1} T_4^F \right)}{\partial x_j} \right] - \frac{\partial}{\partial x_j} \left(k^F \frac{\partial T_{a,1}^F}{\partial x_j} \right) + j_{\Omega,1}^{F,T} = 0
\end{aligned} \tag{7.62c}
$$

$$
{}^{1}R^{T_a^S} = -\frac{\partial}{\partial x_j} \left(k^S \frac{\partial T_{a,1}^S}{\partial x_j} \right) + j_{\Omega,1}^{S,T} = 0 \tag{7.62d}
$$

- $\underline{k=2:}$

$$
{}^{2}R^{q} = -\frac{\partial u_{j,2}}{\partial x_j} + j_{\Omega,2}^{F,p} = 0 \tag{7.63a}
$$

$$
\begin{aligned}
{}^{2}R^{u} = &\, u_{j,2} \frac{\partial v_{i,0}}{\partial x_j} + u_{j,3} \frac{\partial v_{i,1}}{\partial x_j} + \left(2u_{j,5} + u_{j,0} \right) \frac{\partial v_{i,2}}{\partial x_j} + u_{j,1} \frac{\partial v_{i,3}}{\partial x_j} + 2u_{j,2} \frac{\partial v_{i,5}}{\partial x_j} \\
&- \frac{\partial \left(v_{j,2} u_{i,0} \right)}{\partial x_j} - \frac{\partial \left(v_{j,3} u_{i,1} \right)}{\partial x_j} - \frac{\partial \left(\left(2v_{j,5} + v_{j,0} \right) u_{i,2} \right)}{\partial x_j} - \frac{\partial \left(v_{j,1} u_{i,3} \right)}{\partial x_j} - 2\frac{\partial \left(v_{j,2} u_{i,5} \right)}{\partial x_j} \\
&+ \rho^F c_p \left(T_{a,0}^F \frac{\partial T_2^F}{\partial x_i} + T_{a,1}^F \frac{\partial T_3^F}{\partial x_i} + T_{a,2}^F \frac{\partial T_0^F}{\partial x_i} + 2T_{a,2}^F \frac{\partial T_5^F}{\partial x_i} + T_{a,3}^F \frac{\partial T_1^F}{\partial x_i} + 2T_{a,5}^F \frac{\partial T_2^F}{\partial x_i} \right)
\end{aligned}
$$

$$+\frac{\partial q_2}{\partial x_i}-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial u_{i,2}}{\partial x_j}+\frac{\partial u_{j,2}}{\partial x_i}\right)\right]+j^{F,v}_{\Omega,i,2}=0,\quad i=1,2(,3) \tag{7.63b}$$

$$\begin{aligned}{}^2R^{T^F_a}=&-\rho^F c_p\left[\frac{\partial\left(v_{j,2}T^F_{a,0}\right)}{\partial x_j}+\frac{\partial\left(v_{j,3}T^F_{a,1}\right)}{\partial x_j}+\frac{\partial\left(\left(2v_{j,5}+v_{j,0}\right)T^F_{a,2}\right)}{\partial x_j}+\frac{\partial\left(v_{j,1}T^F_{a,3}\right)}{\partial x_j}\right.\\&\left.+2\frac{\partial\left(v_{j,2}T^F_{a,5}\right)}{\partial x_j}\right]-\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T^F_{a,2}}{\partial x_j}\right)+j^{F,T}_{\Omega,2}=0\end{aligned} \tag{7.63c}$$

$${}^2R^{T^S_a}=-\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T^S_{a,2}}{\partial x_j}\right)+j^{S,T}_{\Omega,2}=0 \tag{7.63d}$$

- $\underline{k=3}$:

$${}^3R^q=-\frac{\partial u_{j,3}}{\partial x_j}+j^{F,p}_{\Omega,3}=0 \tag{7.64a}$$

$$\begin{aligned}{}^3R^u=&u_{j,3}\frac{\partial v_{i,0}}{\partial x_j}+u_{j,2}\frac{\partial v_{i,1}}{\partial x_j}+u_{j,1}\frac{\partial v_{i,2}}{\partial x_j}+\left(u_{j,0}+2u_{j,4}+2u_{j,5}\right)\frac{\partial v_{i,3}}{\partial x_j}+2u_{j,3}\frac{\partial v_{i,4}}{\partial x_j}+2u_{j,3}\frac{\partial v_{i,5}}{\partial x_j}\\&-\frac{\partial\left(v_{j,3}u_{i,0}\right)}{\partial x_j}-\frac{\partial\left(v_{j,2}u_{i,1}\right)}{\partial x_j}-\frac{\partial\left(v_{j,1}u_{i,2}\right)}{\partial x_j}-\frac{\partial\left(\left(v_{j,0}+2v_{j,4}+2v_{j,5}\right)u_{i,3}\right)}{\partial x_j}-2\frac{\partial\left(v_{j,3}u_{i,4}\right)}{\partial x_j}\\&+\rho^F c_p\left(T^F_{a,0}\frac{\partial T^F_3}{\partial x_i}+T^F_{a,1}\frac{\partial T^F_2}{\partial x_i}+T^F_{a,2}\frac{\partial T^F_1}{\partial x_i}+T^F_{a,3}\frac{\partial T^F_0}{\partial x_i}+2T^F_{a,3}\frac{\partial T^F_4}{\partial x_i}\right.\\&\left.+2T^F_{a,3}\frac{\partial T^F_5}{\partial x_i}+2T^F_{a,4}\frac{\partial T^F_3}{\partial x_i}+2T^F_{a,5}\frac{\partial T^F_3}{\partial x_i}\right)-2\frac{\partial\left(v_{j,3}u_{i,5}\right)}{\partial x_j}+\frac{\partial q_3}{\partial x_i}\\&-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial u_{i,3}}{\partial x_j}+\frac{\partial u_{j,3}}{\partial x_i}\right)\right]+j^{F,v}_{\Omega,i,3}=0,\quad i=1,2(,3)\end{aligned} \tag{7.64b}$$

$$\begin{aligned}{}^3R^{T^F_a}=&-\rho^F c_p\left[\frac{\partial\left(v_{j,3}T^F_{a,0}\right)}{\partial x_j}+\frac{\partial\left(v_{j,2}T^F_{a,1}\right)}{\partial x_j}+\frac{\partial\left(v_{j,1}T^F_{a,2}\right)}{\partial x_j}+2\frac{\partial\left(v_{j,3}T^F_{a,4}\right)}{\partial x_j}\right.\\&\left.+\frac{\partial\left(\left(v_{j,0}+2v_{j,4}+2v_{j,5}\right)T^F_{a,3}\right)}{\partial x_j}+2\frac{\partial\left(v_{j,3}T^F_{a,5}\right)}{\partial x_j}\right]-\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T^F_{a,3}}{\partial x_j}\right)+j^{F,T}_{\Omega,3}=0\end{aligned} \tag{7.64c}$$

$${}^3R^{T^S_a}=-\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T^S_{a,3}}{\partial x_j}\right)+j^{S,T}_{\Omega,3}=0 \tag{7.64d}$$

- $\underline{k=4}$:

$${}^4R^q=-\frac{\partial u_{j,4}}{\partial x_j}+j^{F,p}_{\Omega,4}=0 \tag{7.65a}$$

$${}^4R^u=u_{j,4}\frac{\partial v_{i,0}}{\partial x_j}+u_{j,1}\frac{\partial v_{i,1}}{\partial x_j}+u_{j,3}\frac{\partial v_{i,3}}{\partial x_j}+\left(v_{j,0}+4v_{j,4}\right)\frac{\partial v_{i,4}}{\partial x_j}$$

$$
\begin{aligned}
&-\frac{\partial\left(v_{j,4}u_{i,0}\right)}{\partial x_j}-\frac{\partial\left(v_{j,1}u_{i,1}\right)}{\partial x_j}-\frac{\partial\left(v_{j,3}u_{i,3}\right)}{\partial x_j}-\frac{\partial\left(\left(v_{j,0}+4v_{j,4}\right)u_{i,4}\right)}{\partial x_j}\\
&+\rho^F c_p\left(T^F_{a,0}\frac{\partial T^F_4}{\partial x_i}+T^F_{a,1}\frac{\partial T^F_1}{\partial x_i}+T^F_{a,3}\frac{\partial T^F_3}{\partial x_i}+T^F_{a,4}\frac{\partial T^F_0}{\partial x_i}+4T^F_{a,4}\frac{\partial T^F_4}{\partial x_i}\right)\\
&+\frac{\partial q_4}{\partial x_i}-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial u_{i,4}}{\partial x_j}+\frac{\partial u_{j,4}}{\partial x_i}\right)\right]+j^{F,v}_{\Omega,i,4}=0,\quad i=1,2(,3)
\end{aligned}
\tag{7.65b}
$$

$$
\begin{aligned}
{}^4R^{T^F_a}=&-\rho^F c_p\left[\frac{\partial\left(v_{j,4}T^F_{a,0}\right)}{\partial x_j}+\frac{\partial\left(v_{j,1}T^F_{a,1}\right)}{\partial x_j}+\frac{\partial\left(v_{j,3}T^F_{a,3}\right)}{\partial x_j}+\frac{\partial\left(\left(v_{j,0}+4v_{j,4}\right)T^F_{a,4}\right)}{\partial x_j}\right]\\
&-\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T^F_{a,4}}{\partial x_j}\right)+j^{F,T}_{\Omega,4}=0
\end{aligned}
\tag{7.65c}
$$

$$
{}^4R^{T^S_a}=-\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T^S_{a,4}}{\partial x_j}\right)+j^{S,T}_{\Omega,4}=0
\tag{7.65d}
$$

- $\underline{k=5:}$

$$
{}^5R^{q}=-\frac{\partial u_{j,5}}{\partial x_j}+j^{F,p}_{\Omega,5}=0
\tag{7.66a}
$$

$$
\begin{aligned}
{}^5R^{u}=&u_{j,5}\frac{\partial v_{i,0}}{\partial x_j}+u_{j,2}\frac{\partial v_{i,2}}{\partial x_j}+u_{j,3}\frac{\partial v_{i,3}}{\partial x_j}+\left(v_{j,0}+4v_{j,5}\right)\frac{\partial v_{i,5}}{\partial x_j}\\
&-\frac{\partial\left(v_{j,5}u_{i,0}\right)}{\partial x_j}-\frac{\partial\left(v_{j,2}u_{i,2}\right)}{\partial x_j}-\frac{\partial\left(v_{j,3}u_{i,3}\right)}{\partial x_j}-\frac{\partial\left(\left(v_{j,0}+4v_{j,5}\right)u_{i,5}\right)}{\partial x_j}\\
&+\rho^F c_p\left(T^F_{a,0}\frac{\partial T^F_5}{\partial x_i}+T^F_{a,2}\frac{\partial T^F_2}{\partial x_i}+T^F_{a,3}\frac{\partial T^F_3}{\partial x_i}+T^F_{a,5}\frac{\partial T^F_0}{\partial x_i}+4T^F_{a,5}\frac{\partial T^F_5}{\partial x_i}\right)\\
&+\frac{\partial q_5}{\partial x_i}-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial u_{i,5}}{\partial x_j}+\frac{\partial u_{j,5}}{\partial x_i}\right)\right]+j^{F,v}_{\Omega,i,4}=0,\quad i=1,2(,3)
\end{aligned}
\tag{7.66b}
$$

$$
\begin{aligned}
{}^5R^{T^F_a}=&-\rho^F c_p\left[\frac{\partial\left(v_{j,5}T^F_{a,0}\right)}{\partial x_j}+\frac{\partial\left(v_{j,2}T^F_{a,2}\right)}{\partial x_j}+\frac{\partial\left(v_{j,3}T^F_{a,3}\right)}{\partial x_j}+\frac{\partial\left(\left(v_{j,0}+4v_{j,5}\right)T^F_{a,5}\right)}{\partial x_j}\right]\\
&-\frac{\partial}{\partial x_j}\left(k^F\frac{\partial T^F_{a,5}}{\partial x_j}\right)+j^{F,T}_{\Omega,5}=0
\end{aligned}
\tag{7.66c}
$$

$$
{}^5R^{T^S_a}=-\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T^S_{a,5}}{\partial x_j}\right)+j^{S,T}_{\Omega,5}=0
\tag{7.66d}
$$

The iPCE adjoint PDEs are solved using similar discretization and numerical schemes to the primal PDEs. The difference is that the iPCE adjoint continuity and momentum PDEs can be solved after making the iPCE heat transfer PDEs

converge, instead of solving all of them simultaneously, since the iPCE heat transfer PDEs do not include any PCE adjoint velocity or pressure coefficient fields.

7.2.3.2 iPCE Adjoint Boundary Conditions

The boundary conditions for the PCE coefficient fields arise after zeroing the multipliers of the variations of the PCE fields included inside the surface integrals.

Along S_I^F,

$$u^I_{\langle t\rangle,m} = \frac{\partial j_{S_{I,k,m}}}{\partial \tau_{ij}} n_k t^I_i n_j + \frac{\partial j_{S_{I,k,m}}}{\partial \tau_{ij}} n_k t^I_j n_i \tag{7.67a}$$

$$u^{II}_{\langle t\rangle,m} = \frac{\partial j_{S_{I,k,m}}}{\partial \tau_{ij}} n_k t^{II}_i n_j + \frac{\partial j_{S_{I,k,m}}}{\partial \tau_{ij}} n_k t^{II}_j n_i \tag{7.67b}$$

Also, $T^F_{a,m} = \frac{\partial j^{'F}_S}{\partial Q_m}$ and $\partial q_m/\partial n = 0$, $m = 0, ..., 5$.

Along S_O^F, the boundary conditions for $q_m,\ u_{i,m}$ and $T^F_{a,m}$ read

- $\underline{k=0}$:

$$\begin{aligned} q_0 &= u_{\langle n\rangle,0} v_{\langle n\rangle,0} + u_{\langle n\rangle,1} v_{\langle n\rangle,1} + u_{\langle n\rangle,2} v_{\langle n\rangle,2} + u_{\langle n\rangle,3} v_{\langle n\rangle,3} + 2u_{\langle n\rangle,4} v_{\langle n\rangle,4} + 2u_{\langle n\rangle,5} v_{\langle n\rangle,5} \\ &+ 2\nu \frac{\partial u_{\langle n\rangle,0}}{\partial n} + \frac{\partial j_{S_{O,j},0}}{\partial v_i} n_j n_i + j^v_{S_{O,i},0} n_i = 0 \end{aligned} \tag{7.68a}$$

$$\begin{aligned} &u^l_{\langle t\rangle,0} v_{\langle n\rangle,0} + u^l_{\langle t\rangle,1} v_{\langle n\rangle,1} + u^l_{\langle t\rangle,2} v_{\langle n\rangle,2} + u^l_{\langle t\rangle,3} v_{\langle n\rangle,3} + 2u^l_{\langle t\rangle,4} v_{\langle n\rangle,4} + 2u^l_{\langle t\rangle,5} v_{\langle n\rangle,5} \\ &+ \nu \left(\frac{\partial u^l_{\langle t\rangle,0}}{\partial n} + \frac{\partial u_{\langle n\rangle,0}}{\partial t^l} \right) + \frac{\partial j_{S_{O,j},0}}{\partial v_i} n_j t^l_i + j^v_{S_{O,i},0} t^l_i = 0 \end{aligned} \tag{7.68b}$$

$$\begin{aligned} &\rho^F c_p \left(T^F_{a,0} v_{\langle n\rangle,0} + T^F_{a,1} v_{\langle n\rangle,1} + T^F_{a,2} v_{\langle n\rangle,2} + T^F_{a,3} v_{\langle n\rangle,3} + 2T^F_{a,4} v_{\langle n\rangle,4} + 2T^F_{a,5} v_{\langle n\rangle,5} \right) \\ &+ \rho^F \alpha_{eff} c_p \frac{\partial T^F_{a,0}}{\partial n} + \frac{\partial j^F_{S_i,0}}{\partial T^F} n_i + j^{F,T}_{S,0} = 0 \end{aligned} \tag{7.68c}$$

- $\underline{k=1}$:

$$\begin{aligned} q_1 &= u_{\langle n\rangle,0} v_{\langle n\rangle,1} + u_{\langle n\rangle,1} v_{\langle n\rangle,0} + u_{\langle n\rangle,2} v_{\langle n\rangle,3} + u_{\langle n\rangle,3} v_{\langle n\rangle,2} + 2u_{\langle n\rangle,4} v_{\langle n\rangle,1} + 2u_{\langle n\rangle,1} v_{\langle n\rangle,4} \\ &+ 2\nu \frac{\partial u_{\langle n\rangle,1}}{\partial n} + \frac{\partial j_{S_{O,j},1}}{\partial v_i} n_j n_i + j^v_{S_{O,i},1} n_i = 0 \end{aligned} \tag{7.69a}$$

$$u^l_{\langle t\rangle,0} v_{\langle n\rangle,1} + u^l_{\langle t\rangle,1} v_{\langle n\rangle,0} + u^l_{\langle t\rangle,2} v_{\langle n\rangle,3} + u^l_{\langle t\rangle,3} v_{\langle n\rangle,2} + 2u^l_{\langle t\rangle,4} v_{\langle n\rangle,1} + 2u^l_{\langle t\rangle,1} v_{\langle n\rangle,4} \tag{7.69b}$$

$$\rho^F c_p \left(T^F_{a,0} v_{\langle n\rangle,1} + T^F_{a,1} v_{\langle n\rangle,0} + T^F_{a,2} v_{\langle n\rangle,3} + T^F_{a,3} v_{\langle n\rangle,2} + 2T^F_{a,4} v_{\langle n\rangle,1} + 2T^F_{a,1} v_{\langle n\rangle,4} \right)$$

$$+\rho^F \alpha_{eff} c_p \frac{\partial T^F_{a,1}}{\partial n} + \frac{\partial j^F_{S_i,1}}{\partial T^F} n_i + j^{F,T}_{S,1} = 0 \tag{7.69c}$$

- $\underline{k=2:}$

$$\begin{aligned} q_2 &= u_{\langle n\rangle,2} v_{\langle n\rangle,0} + u_{\langle n\rangle,3} v_{\langle n\rangle,1} + u_{\langle n\rangle,0} v_{\langle n\rangle,2} + 2u_{\langle n\rangle,5} v_{\langle n\rangle,3} + u_{\langle n\rangle,1} v_{\langle n\rangle,4} + 2u_{\langle n\rangle,2} v_{\langle n\rangle,5} \\ &\quad + 2\nu \frac{\partial u_{\langle n\rangle,2}}{\partial n} + \frac{\partial j_{S_{O,j},2}}{\partial v_i} n_j n_i + j^v_{S_{O,i},2} n_i = 0 \end{aligned} \tag{7.70a}$$

$$\begin{aligned} &u^l_{\langle t\rangle,2} v_{\langle n\rangle,0} + u^l_{\langle t\rangle,3} v_{\langle n\rangle,1} + u^l_{\langle t\rangle,0} v_{\langle n\rangle,2} + 2u^l_{\langle t\rangle,5} v_{\langle n\rangle,3} + u^l_{\langle t\rangle,1} v_{\langle n\rangle,4} + 2u^l_{\langle t\rangle,2} v_{\langle n\rangle,5} \\ &+ \nu \left(\frac{\partial u^l_{\langle t\rangle,2}}{\partial n} + \frac{\partial u_{\langle n\rangle,2}}{\partial t^l} \right) + \frac{\partial j_{S_{O,j},2}}{\partial v_i} n_j t^l_i + j^v_{S_{O,i},2} t^l_i = 0 \end{aligned} \tag{7.70b}$$

$$\begin{aligned} &\rho^F c_p \left(T^F_{a,2} v_{\langle n\rangle,0} + T^F_{a,3} v_{\langle n\rangle,1} + T^F_{a,0} v_{\langle n\rangle,2} + 2T^F_{a,5} v_{\langle n\rangle,3} + T^F_{a,1} v_{\langle n\rangle,4} + 2T^F_{a,2} v_{\langle n\rangle,5} \right) \\ &+ \rho^F \alpha_{eff} c_p \frac{\partial T^F_{a,2}}{\partial n} + \frac{\partial j^F_{S_i,2}}{\partial T^F} n_i + j^{F,T}_{S,2} = 0 \end{aligned} \tag{7.70c}$$

- $\underline{k=3:}$

$$\begin{aligned} q_3 &= u_{\langle n\rangle,3} v_{\langle n\rangle,0} + u_{\langle n\rangle,2} v_{\langle n\rangle,1} + u_{\langle n\rangle,1} v_{\langle n\rangle,2} + u_{\langle n\rangle,0} v_{\langle n\rangle,3} + 2u_{\langle n\rangle,4} v_{\langle n\rangle,3} + 2u_{\langle n\rangle,5} v_{\langle n\rangle,3} \\ &\quad + 2u_{\langle n\rangle,3} v_{\langle n\rangle,4} + 2u_{\langle n\rangle,3} v_{\langle n\rangle,5} + 2\nu \frac{\partial u_{\langle n\rangle,3}}{\partial n} + \frac{\partial j_{S_{O,j},3}}{\partial v_i} n_j n_i + j^v_{S_{O,i},3} n_i = 0 \end{aligned} \tag{7.71a}$$

$$\begin{aligned} &u^l_{\langle t\rangle,3} v_{\langle n\rangle,0} + u^l_{\langle t\rangle,2} v_{\langle n\rangle,1} + u^l_{\langle t\rangle,1} v_{\langle n\rangle,2} + u^l_{\langle t\rangle,0} v_{\langle n\rangle,3} + 2u^l_{\langle t\rangle,4} v_{\langle n\rangle,3} + 2u^l_{\langle t\rangle,5} v_{\langle n\rangle,3} \\ &+ 2u^l_{\langle t\rangle,3} v_{\langle n\rangle,4} + 2u^l_{\langle t\rangle,3} v_{\langle n\rangle,5} + \nu \left(\frac{\partial u^l_{\langle t\rangle,3}}{\partial n} + \frac{\partial u_{\langle n\rangle,3}}{\partial t^l} \right) + \frac{\partial j_{S_{O,j},3}}{\partial v_i} n_j t^l_i + j^v_{S_{O,i},3} t^l_i = 0 \end{aligned} \tag{7.71b}$$

$$\begin{aligned} &\rho^F c_p \big(T^F_{a,3} v_{\langle n\rangle,0} + T^F_{a,2} v_{\langle n\rangle,1} + T^F_{a,1} v_{\langle n\rangle,2} + T^F_{a,0} v_{\langle n\rangle,3} + 2T^F_{a,4} v_{\langle n\rangle,3} + 2T^F_{a,5} v_{\langle n\rangle,3} \\ &+ 2T^F_{a,3} v_{\langle n\rangle,4} + 2T^F_{a,3} v_{\langle n\rangle,5} \big) + \rho^F \alpha_{eff} c_p \frac{\partial T^F_{a,3}}{\partial n} + \frac{\partial j^F_{S_i,3}}{\partial T^F} n_i + j^{F,T}_{S,3} = 0 \end{aligned} \tag{7.71c}$$

- $\underline{k=4:}$

$$\begin{aligned} q_4 &= u_{\langle n\rangle,0} v_{\langle n\rangle,0} + u_{\langle n\rangle,1} v_{\langle n\rangle,1} + u_{\langle n\rangle,3} v_{\langle n\rangle,3} + u_{\langle n\rangle,0} v_{\langle n\rangle,4} + 4u_{\langle n\rangle,4} v_{\langle n\rangle,4} \\ &\quad + 2\nu \frac{\partial u_{\langle n\rangle,4}}{\partial n} + \frac{\partial j_{S_{O,j},4}}{\partial v_i} n_j n_i + j^v_{S_{O,i},4} n_i = 0 \end{aligned} \tag{7.72a}$$

$$\begin{aligned} &u^l_{\langle t\rangle,0} v_{\langle n\rangle,0} + u^l_{\langle t\rangle,1} v_{\langle n\rangle,1} + u^l_{\langle t\rangle,3} v_{\langle n\rangle,3} + u^l_{\langle t\rangle,0} v_{\langle n\rangle,4} + 2u^l_{\langle t\rangle,4} v_{\langle n\rangle,4} \\ &+ \nu \left(\frac{\partial u^l_{\langle t\rangle,4}}{\partial n} + \frac{\partial u_{\langle n\rangle,4}}{\partial t^l} \right) + \frac{\partial j_{S_{O,j},4}}{\partial v_i} n_j t^l_i + j^v_{S_{O,i},4} t^l_i = 0 \end{aligned} \tag{7.72b}$$

$$\rho^F c_p \left(T^F_{a,0} v_{\langle n\rangle,0} + T^F_{a,1} v_{\langle n\rangle,1} + T^F_{a,3} v_{\langle n\rangle,3} + T^F_{a,0} v_{\langle n\rangle,4} + 2T^F_{a,4} v_{\langle n\rangle,4}\right)$$
$$+\rho^F \alpha_{eff} c_p \frac{\partial T^F_{a,4}}{\partial n} + \frac{\partial j^F_{S_i,4}}{\partial T^F} n_i + j^{F,T}_{S,4} = 0 \tag{7.72c}$$

- $\underline{k=5}$:

$$q_5 = u_{\langle n\rangle,5} v_{\langle n\rangle,0} + u_{\langle n\rangle,2} v_{\langle n\rangle,2} + u_{\langle n\rangle,3} v_{\langle n\rangle,3} + u_{\langle n\rangle,0} v_{\langle n\rangle,5} + 4u_{\langle n\rangle,5} v_{\langle n\rangle,5}$$
$$+2\nu \frac{\partial u_{\langle n\rangle,5}}{\partial n} + \frac{\partial j_{S_{O,j},5}}{\partial v_i} n_j n_i + j^v_{S_{O,i},5} n_i = 0 \tag{7.73a}$$
$$u^l_{\langle t\rangle,5} v_{\langle n\rangle,0} + u^l_{\langle t\rangle,2} v_{\langle n\rangle,2} + u^l_{\langle t\rangle,3} v_{\langle n\rangle,3} + u^l_{\langle t\rangle,0} v_{\langle n\rangle,5} + 4u^l_{\langle t\rangle,5} v_{\langle n\rangle,5}$$
$$+\nu \left(\frac{\partial u^l_{\langle t\rangle,5}}{\partial n} + \frac{\partial u_{\langle n\rangle,5}}{\partial t^l}\right) + \frac{\partial j_{S_{O,j},5}}{\partial v_i} n_j t^l_i + j^v_{S_{O,i},5} t^l_i = 0 \tag{7.73b}$$
$$\rho^F c_p \left(T^F_{a,5} v_{\langle n\rangle,0} + T^F_{a,2} v_{\langle n\rangle,2} + T^F_{a,3} v_{\langle n\rangle,3} + T^F_{a,0} v_{\langle n\rangle,5} + 2T^F_{a,5} v_{\langle n\rangle,5}\right)$$
$$+\rho^F \alpha_{eff} c_p \frac{\partial T^F_{a,5}}{\partial n} + \frac{\partial j^F_{S_i,5}}{\partial T^F} n_i + j^{F,T}_{S,5} = 0 \tag{7.73c}$$

Finally, the boundary condition for the normal component of the adjoint velocity PCE coefficients reads $\partial u_{\langle n\rangle,m}/\partial n = 0,\ m = 0, ..., M$.

Along S^F_W, the boundary conditions for $v_{i,k}$ and q_k are the same as for the fluid inlet. Along $S^F_{W,D}$ and $S^S_{W,D}$, $T^F_{a,k} = \frac{\partial j'^F_S}{\partial Q^F_k},\ k = 0, ..., 5$ and $T^S_{a,k} = \frac{\partial j'^S_S}{\partial Q^S_k},\ k = 0, ..., 5$ and along $S^F_{W,Fl}$ and $S^S_{W,Fl}$, $k^F \frac{\partial T^F_{a,k}}{\partial n} + \frac{\partial j'^F_{S,i}}{\partial T^F_{a,k}} n_i + j'^{F,T}_{S,k} = 0$ and $k^S \frac{\partial T^S_{a,k}}{\partial n} + \frac{\partial j'^S_{S,i}}{\partial T^S_{a,k}} n_i + j'^{S,T}_{S,k} = 0,\ k = 0, ..., 5$.

Regarding $\overline{S}^F$ and $\overline{S}^S$,

$$T^F_{a,k}\big|_{\overline{S}^F} - \frac{\partial j'^F_S}{\partial Q^F_k} = T^S_{a,k}\big|_{\overline{S}^S} - \frac{\partial j'^S_S}{\partial Q^S_k} \tag{7.74a}$$
$$\left(k^F \frac{\partial T^F_{a,k}}{\partial n} + \frac{\partial j^F_{S_{i,k}}}{\partial T^F} n_i + j^{F,T}_{S,k}\right)\Big|_{\overline{S}^F} = -\left(k^S \frac{\partial T^S_{a,k}}{\partial n} + \frac{\partial j_{S^S_{i,k}}}{\partial T^S} n_i + j^{S,T}_{S,k}\right)\Big|_{\overline{S}^S} \tag{7.74b}$$

Regarding the boundary conditions along the FSI boundaries with thermal insulation thickness, the following conditions are derived

- $\underline{k=0}$:

$$\mu_L \frac{\partial T^F_{a,0}}{\partial n} + \sigma_L \frac{\partial T^F_{a,2}}{\partial n} = -k^c \left(\Delta T^F_{a,0} + \frac{\partial j'^F_S}{\partial Q^F_0} - \frac{\partial j'^S_S}{\partial Q^S_0}\right) \tag{7.75}$$

- $\underline{k=1:}$

$$\mu_L \frac{\partial T^F_{a,1}}{\partial n} + \sigma_L \frac{\partial T^F_{a,3}}{\partial n} = -k^c \left(\Delta T^F_{a,1} + \frac{\partial j'^F_S}{\partial Q^F_1} - \frac{\partial j'^S_S}{\partial Q^S_1} \right) \tag{7.76}$$

- $\underline{k=2:}$

$$\mu_L \frac{\partial T^F_{a,2}}{\partial n} + \sigma_L \left(\frac{\partial T^F_{a,0}}{\partial n} + 2\frac{\partial T^F_{a,5}}{\partial n} \right) = -k^c \left(\Delta T^F_{a,2} + \frac{\partial j'^F_S}{\partial Q^F_2} - \frac{\partial j'^S_S}{\partial Q^S_2} \right) \tag{7.77}$$

- $\underline{k=3:}$

$$\sum_{m=0}^{M} \mu_L \frac{\partial T^F_{a,3}}{\partial n} + \sigma_L \frac{\partial T^F_{a,1}}{\partial n} = -k^c \left(\Delta T^F_{a,3} + \frac{\partial j'^F_S}{\partial Q^F_3} - \frac{\partial j'^S_S}{\partial Q^S_3} \right) \tag{7.78}$$

- $\underline{k=4:}$

$$\mu_L \frac{\partial T^F_{a,4}}{\partial n} = -k^c \left(\Delta T^F_{a,4} + \frac{\partial j'^F_S}{\partial Q^F_4} - \frac{\partial j'^S_S}{\partial Q^S_4} \right) \tag{7.79}$$

- $\underline{k=5:}$

$$\mu_L \frac{\partial T^F_{a,5}}{\partial n} + \sigma_L \frac{\partial T^F_{a,2}}{\partial n} = -k^c \left(\Delta T^F_{a,5} + \frac{\partial j'^F_S}{\partial Q^F_5} - \frac{\partial j'^S_S}{\partial Q^S_5} \right) \tag{7.80}$$

7.2.3.3 iPCE Adjoint SDs

The adjoint SD expressions are derived similarly to the SDs for CHT problems without uncertainties; the following expressions are obtained

$$\begin{aligned}
\left. \frac{\delta J'}{\delta b_n} \right|_{FI} = \int_{\Omega^F} \Big[& (u_{i,0}v_{j,0} + u_{i,1}v_{j,1} + u_{i,2}v_{j,2} + u_{i,3}v_{j,3} + 2u_{i,4}v_{j,4} + 2u_{i,5}v_{j,5}) \frac{\partial v_{i,0}}{\partial x_k} \\
& + (u_{i,0}v_{j,1} + u_{i,1}v_{j,0} + 2u_{i,1}v_{j,4} + u_{i,2}v_{j,3} + u_{i,3}v_{j,2} + 2u_{i,4}v_{j,1}) \frac{\partial v_{i,1}}{\partial x_k} \\
& + (u_{i,0}v_{j,2} + u_{i,1}v_{j,3} + u_{i,2}v_{j,0} + 2u_{i,2}v_{j,5} + u_{i,3}v_{j,1} + 2u_{i,5}v_{j,2}) \frac{\partial v_{i,2}}{\partial x_k}
\end{aligned}$$

$$
\begin{aligned}
&+\Bigg(u_{i,0}v_{j,3}+u_{i,1}v_{j,2}+u_{i,2}v_{j,1}+u_{i,3}v_{j,0}+2u_{i,3}v_{j,4}+2u_{i,3}v_{j,5}+2u_{i,4}v_{j,3}\\
&+2u_{i,5}v_{j,3}\Bigg)\frac{\partial v_{i,3}}{\partial x_k}+(2u_{i,0}v_{j,4}+2u_{i,1}v_{j,1}+2u_{i,3}v_{j,3}+2u_{i,4}v_{j,0}+8u_{i,4}v_{j,4})\frac{\partial v_{i,4}}{\partial x_k}\\
&+(2u_{i,0}v_{j,5}+2u_{i,2}v_{j,2}+2u_{i,3}v_{j,3}+2u_{i,5}v_{j,0}+8u_{i,5}v_{j,5})\frac{\partial v_{i,5}}{\partial x_k}\\
&+\left(T^F_{a,0}v_{j,0}+T^F_{a,1}v_{j,1}+T^F_{a,2}v_{j,2}+T^F_{a,3}v_{j,3}+2T^F_{a,4}v_{j,4}+2T^F_{a,5}v_{j,5}\right)\frac{\partial T^F_0}{\partial x_k}\\
&+\left(T^F_{a,0}v_{j,1}+T^F_{a,1}v_{j,0}+2T^F_{a,1}v_{j,4}+T^F_{a,2}v_{j,3}+T^F_{a,3}v_{j,2}+2T^F_{a,4}v_{j,1}\right)\frac{\partial T^F_1}{\partial x_k}\\
&+\left(T^F_{a,0}v_{j,2}+T^F_{a,1}v_{j,3}+T^F_{a,2}v_{j,0}+2T^F_{a,2}v_{j,5}+T^F_{a,3}v_{j,1}+2T^F_{a,5}v_{j,2}\right)\frac{\partial T^F_2}{\partial x_k}\\
&+\Bigg(T^F_{a,0}v_{j,3}+T^F_{a,1}v_{j,2}+T^F_{a,2}v_{j,1}+T^F_{a,3}v_{j,0}+2T^F_{a,3}v_{j,4}+2T^F_{a,3}v_{j,5}+2T^F_{a,4}v_{j,3}\\
&+2T^F_{a,5}v_{j,3}\Bigg)\frac{\partial T^F_3}{\partial x_k}+\left(2T^F_{a,0}v_{j,4}+2T^F_{a,1}v_{j,1}+2T^F_{a,3}v_{j,3}+2T^F_{a,4}v_{j,0}+8T^F_{a,4}v_{j,4}\right)\frac{\partial T^F_4}{\partial x_k}\\
&+\left(2T^F_{a,0}v_{j,5}+2T^F_{a,2}v_{j,2}+2T^F_{a,3}v_{j,3}+2T^F_{a,5}v_{j,0}+8T^F_{a,5}v_{j,5}\right)\frac{\partial T^F_5}{\partial x_k}\Bigg]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\
&+\sum_{l=0}^{5}c_l\Bigg\{\int_{\Omega^F}-u_{j,l}\frac{\partial p_l}{\partial x_k}-\tau^{\alpha}_{ij,l}\frac{\partial v_{i,l}}{\partial x_k}+u_{i,l}\frac{\partial \tau_{ij,l}}{\partial x_k}+q_l\frac{\partial v_{j,l}}{\partial x_k}-k^F\frac{\partial T^F_{a,l}}{\partial x_j}\frac{\partial T^F_l}{\partial x_k}\\
&+T^F_{a,l}\frac{\partial}{\partial x_k}\left(k^F\frac{\partial T^F_l}{\partial x_j}\right)\Bigg]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\
&+\int_{\Omega^S}\left[-k^S\frac{\partial T^S_{a,l}}{\partial x_j}\frac{\partial T^S_l}{\partial x_k}+T^S_{a,l}\frac{\partial}{\partial x_k}\left(k^S\frac{\partial T^S_l}{\partial x_j}\right)\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\
&-\int_{S^F_{W_p}}\left[\frac{\partial j'^F_{S_{W_{p,k}}}}{\partial \tau_{mo,l}}n_k t^I_m t^I_o \tau_{ij,l}\frac{\delta(t^I_i t^I_j)}{\delta b_n}\right]dS-\int_{S^F_{W_p}}\left[\frac{\partial j'^F_{S_{W_{p,k}}}}{\partial \tau_{mo,l}}n_k t^{II}_m t^{II}_o \tau_{ij,l}\frac{\delta(t^{II}_i t^{II}_j)}{\delta b_n}\right]dS\\
&-\int_{S^F_{W_p}}\left[\left(-u_{\langle n\rangle,l}+\frac{\partial j'^F_{S_{W_{p,k}}}}{\partial \tau_{mo,l}}n_k n_m n_o\right)\tau_{ij,l}\frac{\delta(n_i n_j)}{\delta b_n}\right]dS\\
&-\int_{S^F_{W_p}}\left[\left(\frac{\partial j'^F_{S_{W_{p,k}}}}{\partial \tau_{mo,l}}n_k(t^{II}_m t^I_o+t^I_m t^{II}_o)\right)\tau_{ij,l}\frac{\delta(t^{II}_i t^I_j)}{\delta b_n}\right]dS+\int_{\Omega^S}j'^{S,g}_{\Omega}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\
&+\sum_{D=F,S}\Bigg[\int_{S^D_{W_p}}j'^{D,g}_{S_i}\frac{\delta x_k}{\delta b_n}n_i dS+\int_{S^D_{W_p}}j'^{D}_{S_i}\frac{\delta(n_i dS)}{\delta b_n}+\int_{\Omega^D}j'^{D}_{\Omega}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\
&+\int_{S^D_{W_p}}j'^{D}_{S}\frac{\delta(dS)}{\delta b_n}+\int_{\overline{S}^D_p\cup S^D_{W,D,p}\cup S^D_{W,Fl,p}\cup\overline{S}^D_{Ins,p}}k^D T^D_{a,l}\frac{\partial T^D_l}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS\Bigg]\Bigg\}
\end{aligned}
\tag{7.81}
$$

where $\overline{S}^{D}_{Ins,p}$ denote FSI boundaries with thermal insulation thickness and

$$c_l = \begin{cases} 1, & if \quad l = 0,1,2,3 \\ 2, & if \quad l = 4,5 \end{cases} \tag{7.82}$$

7.3 The non-intrusive PCE Method

7.3.1 The niPCE Method for any Distribution

In the niPCE method, the PCE coefficients of the QoI are computed either by performing Galerkin projections [32, 31], as in this thesis, or through the Linear Regression method [101]. The normalized orthogonal polynomials $\widetilde{\psi}_m(\boldsymbol{\xi})$, with $||\widetilde{\psi}_m(\boldsymbol{\xi})||_w = 1$, are used to expand J as

$$J = \sum_{m=0}^{M} J_m \widetilde{\psi}_m(\boldsymbol{\xi}) \tag{7.83}$$

The mean value of J is given by eq. 7.50 for $\nu = 1$, i.e.

$$\mu_J = \langle y^1 \rangle = \int_{\boldsymbol{\mathcal{D}}} \left[\sum_{m=0}^{M} J_m \widetilde{\psi}_m(\boldsymbol{\xi}) \right] W(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{7.84}$$

Since $\widetilde{\psi}_0(\boldsymbol{\xi}) = 1$ for all distributions, eq. 7.84 becomes

$$\mu_J = \sum_{m=0}^{M} J_m \left[\int_{\boldsymbol{\mathcal{D}}} \widetilde{\psi}_m(\boldsymbol{\xi}) \widetilde{\psi}_0(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} \right] = J_m ||\widetilde{\psi}_0(\boldsymbol{\xi})||_w \delta_0^m = J_0 \tag{7.85}$$

The variance σ_J^2 is computed as

$$\sigma_J^2 = \langle y^2 \rangle - \mu_J^2 \tag{7.86}$$

where

$$\langle y^2 \rangle = \int_{\boldsymbol{\mathcal{D}}} J^2\, W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \int_{\boldsymbol{\mathcal{D}}} \left[\sum_{m=0}^{M} J_m \widetilde{\psi}_m(\boldsymbol{\xi}) \right]^2 W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$

$$= \sum_{m_1=0}^{M} \sum_{m_2=0}^{M} J_{m_1} J_{m_2} \int_{\mathcal{D}} \widetilde{\psi}_{m_1}(\boldsymbol{\xi}) \widetilde{\psi}_{m_2}(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$
$$= \sum_{m_1=0}^{M} \sum_{m_2=0}^{M} J_{m_1} J_{m_2} ||\widetilde{\psi}_{m_1}(\boldsymbol{\xi})||_w \delta_{m_1}^{m_2} = \sum_{m=0}^{M} J_m^2 ||\widetilde{\psi}_m(\boldsymbol{\xi})||_w \tag{7.87}$$

Therefore, the standard deviation of J is given by

$$\sigma_J = \sqrt{\sum_{m=0}^{M} J_m^2 - J_0^2} = \sqrt{\sum_{m=1}^{M} J_m^2} \tag{7.88}$$

By comparing eqs. 7.85 and 7.88, in contrast to μ_J, the accuracy of σ_J is affected by the chaos order ϖ, since the latter is used to compute M. To compute both μ_J and σ_J, the PCE coefficients of the QoI, $J_m,\ m=0,...,M$, are needed. To compute them, Galerkin projections of J on each $\widetilde{\psi}_i(\boldsymbol{\xi}),\ i=0,...,M$ are performed, i.e.

$$\langle J, \widetilde{\psi}_i(\boldsymbol{\xi}) \rangle_w = \int_{\mathcal{D}} J \widetilde{\psi}_i(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \int_{\mathcal{D}} \left[\sum_{m=0}^{M} J_m \widetilde{\psi}_m(\boldsymbol{\xi}) \right] \widetilde{\psi}_i(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = J_i \tag{7.89a}$$

(7.89b)

So, the PCE coefficients are given by

$$J_i = \int_{\mathcal{D}} J\ \widetilde{\psi}_i(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{7.90}$$

where the integral is computed by the Gauss Quadrature (GQ) method [86], i.e.

$$J_i = \sum_{j=1}^{N_{GQ}} J\big|_{\mathbf{z}^{(j)}} \widetilde{\psi}_i\big(\mathbf{z}^{(j)}\big) \prod_{l=0}^{K} w_l^{(j)} \tag{7.91}$$

where $N_{GQ} = (\varpi + 1)^K$. To compute J_i, the QoI J has to be evaluated at each combination $\mathbf{z}^{(j)}$ of the roots $z_l^{(j)}$ of the l-th univariate polynomial of degree $\varpi+1$, with $\mathbf{z}^{(\mathbf{j})} = \left(z_1^{(j)}, ..., z_K^{(j)}\right),\ j = 1, ..., N_{GQ}$. This means that to compute μ_J and σ_J, the governing equations have to be solved $N_{GQ} = (\varpi + 1)^K$ times. For the same chaos order, the cost of the iPCE method is M EFS, which is smaller that the N_{GQ} EFS required by the niPCE method, for $\varpi > 1$. By increasing either K or ϖ, the cost of computing μ_J and σ_J increases exponentially. This can be avoided

by using the Smolyak Sparse Grids [65, 166]; in this thesis, it doesn't make any sense to use them, since computation gains appear for problems with higher ϖ and K than those in this thesis. Each weight $w_l^{(j)}$ is computed as

$$w_l^{(j)} = \frac{A_{q+1}}{A_q} \frac{||p_q(z_l^{(j)})||_w}{p'_{q+1}(z_l^{(j)}) p_q(z_l^{(j)})} \tag{7.92}$$

where A_q is the multiplier of z_l^q in $\widetilde{p}_q(z_l)$ and p'_q is the derivative of p_q w.r.t. $z_l^{(j)}$ [115].

7.3.2 The niPCE Method for Normal Distributions

Assuming that all uncertain variants undergo normal distributions, the univariate orthonormal Hermite polynomials are used, which are given by

$$\widetilde{He}_k(\xi_i) = \frac{He_k(\xi_i)}{\|He_k\|_w} = \frac{He_k(\xi_i)}{\sqrt{k!}} \tag{7.93}$$

the w-norm of which is equal to 1. The PCE coefficients of J are computed as

$$J_i = \frac{1}{\sqrt{2\pi}^K} \int_{\mathcal{D}} J(\xi_i) \widetilde{\psi}_i(\boldsymbol{\xi}) W(\boldsymbol{\xi})\, d\boldsymbol{\xi} = \sum_{j=1}^{N_{GQ}} \widetilde{\psi}_i(\mathbf{z}^{(j)}) \left(\prod_{l=0}^{K} w_l^{(i)} \right) J(\boldsymbol{\xi}^{(j)}) \tag{7.94}$$

where $\mathbf{z}^{(j)}$ stands for j-th Gaussian node, the components of which are uncertain variables following the standard normal distribution, i.e. with $\mathcal{N}(0,1)$. To compute each $\xi_i^{(j)}$, the following relationship is used

$$\xi_i^{(j)} = \sigma\, z_i^{(j)} + \mu \text{ and } d\xi_i^{(j)} = \sigma\, dz_i^{(j)} \tag{7.95}$$

which is a transformation from the standard normal distribution to the normal distribution $\mathcal{N}(\mu, \sigma^2)$.

7.3.3 A niPCE Algorithm for Computing Moments of J

Here, an algorithm for computing the mean value and standard deviation of J, with uncertain variables undergoing normal distributions, is presented. In this thesis, the algorithm is applied to CHT problems; of course it can be also used to deal with problems that are not CHT.

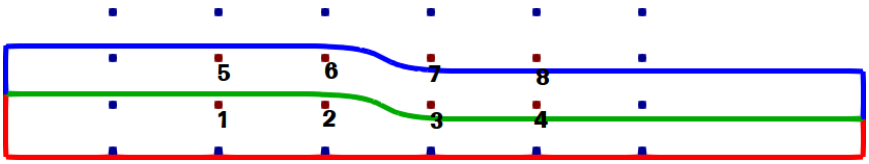


Figure 7.1: Geometry and parameterization in Cases 1 and 2. The CPs in blue are kept fixed and the red ones are free to move.

1. Having decided the chaos order ϖ, M is computed from eq. 7.2.
2. Gaussian nodes $\mathbf{z}^{(i)},\ i = 1, ..., N_{GQ}$, are selected and the weights are computed by eq. 7.92. Roots and weights are tabulated in Appendix E. Then, $\boldsymbol{\xi}_j^{(i)}$ are computed by eq. 7.95.
3. For each $\boldsymbol{\xi}^{(i)}$, eqs. 7.14 to 7.17 are solved and $J(\boldsymbol{\xi}^{(i)})$ are computed. Also, $\widetilde{\psi}_k(\mathbf{z}^{(i)})$ is computed from 7.13, respectively.
4. Each PCE coefficient J_i is computed from eq. 7.94.
5. The mean value and standard deviation of J are computed from eqs. 7.85 and 7.88.

7.4 Applications

In the two cases of this section, a CHT problem involving an S-shaped duct, being in contact with a solid body, is investigated. The geometry is presented in fig. 7.1.

7.4.1 Case 13: Uncertain Inlet Temperature and Velocity Magnitude

In Case 13, the QoI is the solid mean temperature J^{meanT}. Uncertain variables are the magnitude of the inlet velocity and the inlet fluid temperature; undergoing normal distributions with mean values $1\ m/s$ and $310\ K$ and standard deviations $0.1\ m/s$ and $20\ K$. The upper wall of the S-shaped duct is adiabatic. At the lower wall of Ω^S, a constant heat flux $Q = 4800\ W/m^2$ is provided and the rest of the non-FSI walls are adiabatic. Also, $c_p = 1006\ J/kg/K$, $\rho^F = 1.21\ kg/m^3$, $\mathrm{Pr} = 0.7$ and $k^S = 60\ J/m/s/K$. The mean value of the Reynolds number based on the velocity inlet magnitude is $Re = 909$.

First, the primal iPCE method and programmed software are verified against Monte Carlo and niPCE. The values of μ_J and σ_J, computed with the various

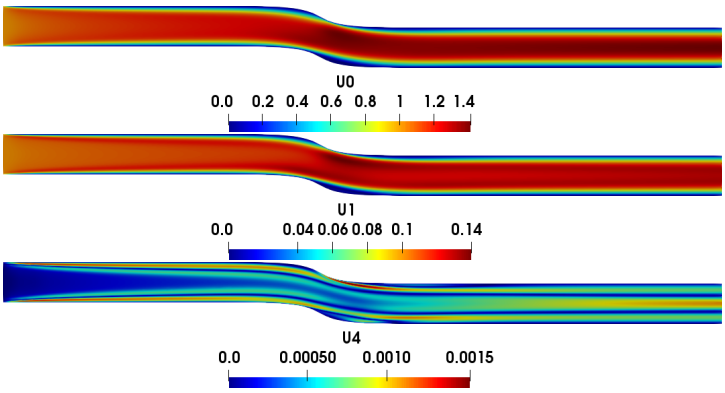


Figure 7.2: Case 13: Magnitude of PCE velocity fields over the initial geometry. Fields regarding the PCE coefficients indexed by $2, 3, 5$ are not plotted as their values are practically zero.

methods, are presented in table 7.2. The cost of the iPCE method is 0.006% and 67% of that required by the Monte Carlo and niPCE methods, respectively. The difference in the μ_J and σ_J values between the iPCE method and the Monte Carlo is 0.1% and 0.3%, respectively. These differences between the iPCE and the niPCE methods are 0.01% and 0.1%, respectively. Under the assumption that the Monte Carlo method (with a sufficient number of replicates) computes reference values of μ_J and σ_J, it is concluded that, for chaos order $\varpi = 2$, accurate statistical moments are computed by both PCE variants, with iPCE requiring 33 % less EFS than niPCE. In figs. 7.2 to 7.4, the non-zero iPCE primal coefficients fields, as

Method	μ_J $[K]$	σ_J $[K]$	**EFS**
Monte Carlo	770.7	27.88	1000
niPCE	772.1	27.83	9
iPCE	771.5	27.80	6

Table 7.2: Case 13: Computed μ_J and σ_J of J^{meanT} with the Monte Carlo, niPCE and iPCE methods and computational cost, in EFS.

computed after solving the iPCE primal PDEs, are presented. By comparing the primal deterministic and PCE fields indexed by 0, in fig. 7.5, small differences between them can be seen, yielding that the PCE fields indexed by 0 are the dominant ones when expanding the corresponding deterministic fields. On the other hand, this does not happen when comparing the adjoint deterministic and PCE fields indexed by 0 for $J' = \mu_J$, as seen in eqs. 7.10; smaller differences occur for $J' = \sigma_J$, see fig. 7.14.

In fig. 7.6, adjoint SDs of μ_J and σ_J w.r.t. the x and y coordinates of the red colored CPs of fig. 7.1 are computed and verified against FDs; the differences between the two curves are practically negligible. In figs. 7.7 to fig. 7.13, the PCE

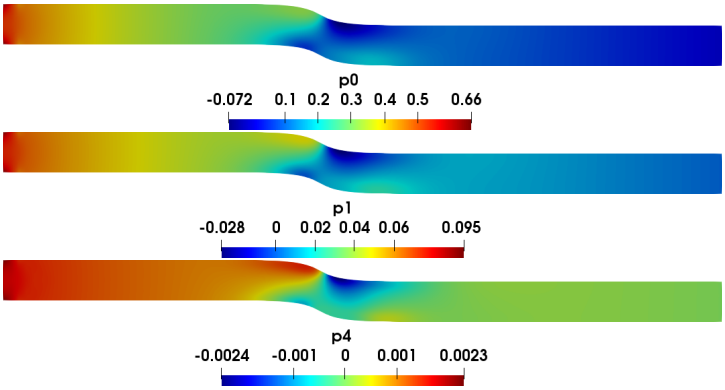


Figure 7.3: Case 13: PCE pressure fields over the initial geometry. Fields regarding the PCE coefficients indexed by $2, 3, 5$ are not plotted as their values are practically zero.

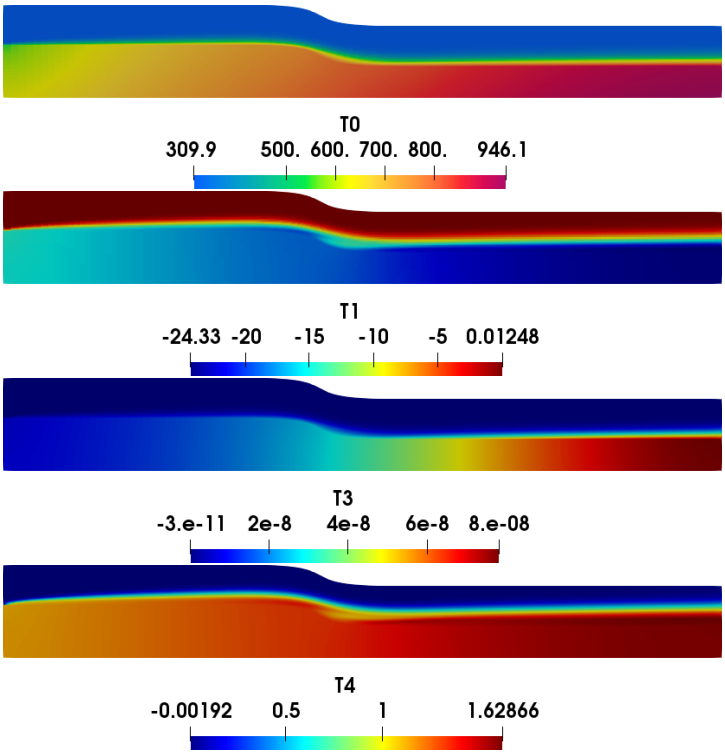


Figure 7.4: Case 13: PCE temperature fields over the fluid and solid domains of the initial geometry. Fields regarding the PCE coefficients indexed by $1, 4, 5$ are not plotted as their values are practically zero.

adjoint fields are presented; in the first three, the PCE adjoint fields have been computed for $J' = \mu_J$ and in the last three, for $J' = \sigma_J$.

Having verified the primal and adjoint iPCE solvers, two independent shape optimizations are performed. In the first one, the goal is to minimize the mean

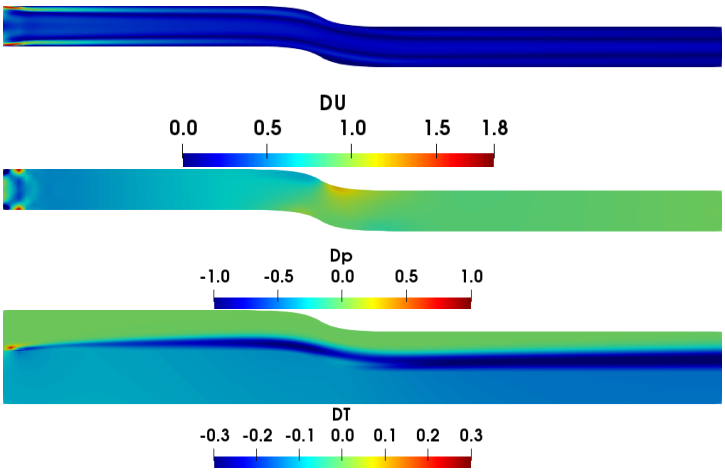


Figure 7.5: Case 13: Difference (%) of the primal deterministic fields and PCE fields indexed by 0, over the initial geometry, for the velocity magnitude, pressure and temperature fields, from top to bottom, respectively. Top and middle: Absolute differences normalized with the max. value of the corresponding deterministic field. Bottom: Relative difference.

Optimizations	μ_J	σ_J
min. μ_J	-12.6%	-5.9%
min. σ_J	-6.6%	-7.2%

Table 7.3: Case 13: Results of the optimizations for min. μ_J and min. σ_J. For each optimization, both of them tabulated.

value of J^{meanT}. In the second, the standard deviation is minimized. Even if minimizing just the standard deviation does not practically make any sense, as long as the mean value of J remains high, this test is very useful in the assessment of the developed iPCE method. Having seen the way an optimization loop behaves for min. mean value and min. standard deviation is all we need in view of real applications in which the objective function is a weighted combination of both. Setting either μ_J or σ_J as targets could reveal whether there is not a new design minimizing both of them at the same time. Results of the two optimizations are tabulated in table 7.3. The optimized geometries pertaining to min. μ_J and min. σ_J are presented in fig. 7.15. Similar trends in the geometry deformation are observed in both optimized geometries. When performing a SOO for one of the two quantities, the other gets reduced but not as much as when being the target in a SOO. As seen in fig. 7.15, to minimize both μ_J and σ_J, the S-shaped duct becomes more curved and the cross-sectional area reduces.

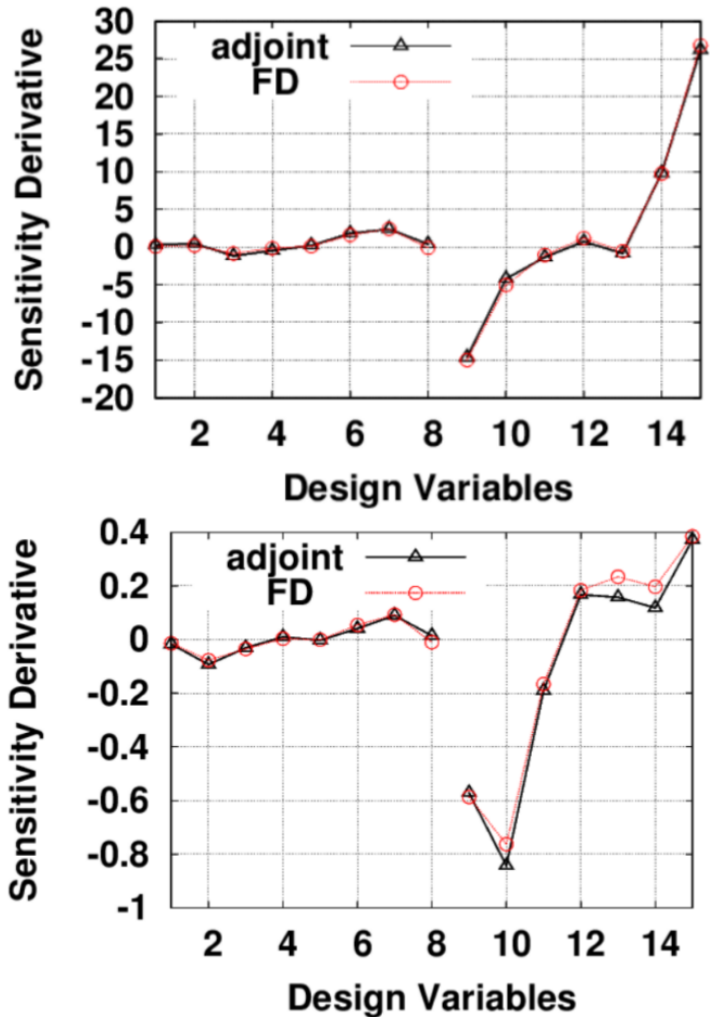


Figure 7.6: Case 13: Adjoint SDs of μ_J (top) and σ_J (bottom) compared with FDs. First half of the SDs correspond to the x and the second half to the y coordinates of the red colored CPs of fig. 7.1.

7.4.2 Case 14: Uncertain Inlet Velocity Magnitude and Insulation Thickness

In Case 14, the QoI is the heat flux J^{HF} passing from Ω^F to Ω^S. Uncertain variables are the inlet velocity magnitude and the thickness of insulation between the two domains, with mean values $1\ m/s$ and $1\ mm$ and standard deviations $0.1\ m/s$ and $0.1\ /mm$ respectively. The upper wall of the S-shaped duct is adiabatic. At S_I^F, $T^F = 700\ K$. Along the lower wall of Ω^S, $T^S = 300\ K$ and the rest of the non-FSI walls are adiabatic. Also, the mean value of Reynolds number is $Re = 909$, $c_p = 1006\ J/kg/K$, $\rho^F = 1.21\ kg/m^3$, $\mathrm{Pr} = 0.7$, $k^S = 60\ J/m/s/K$ and

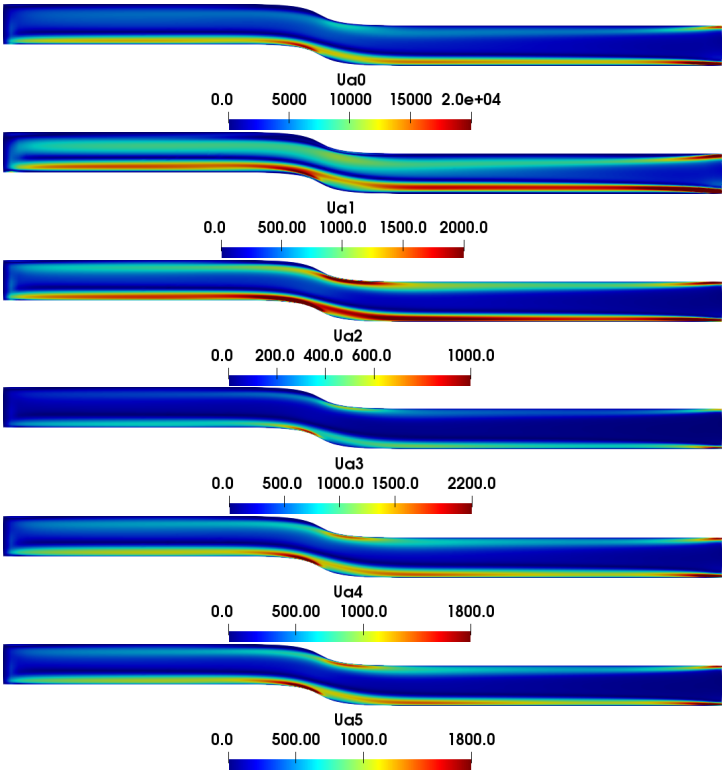


Figure 7.7: Case 13: Magnitude of PCE adjoint velocity fields over the initial geometry, for $J' = \mu_J$.

$k^c = 0.1\ J/m/s/K$.

The niPCE and iPCE methods are used to compute the mean value and standard deviation of the heat flux J^{HF} from the fluid to the solid. The results are tabulated in table 7.4. The relative difference between the two methods is below $0.08\ \%$, with the iPCE method requiring $2/3$ of the cost of the niPCE method.

In fig. 7.19, a comparison is made between the primal deterministic field and those corresponding to the PCE coefficients indexed by 0. As in Case 13, it is observed that the difference between them is very small.

Method	$\mu_j\ [W]$	$\sigma_j\ [W]$	**EFS**
niPCE	2288.11	85.54	9
iPCE	2287.17	85.47	6
Difference [%]	0.04	0.08	50

Table 7.4: Case 14: μ_J and σ_J of J^{HF} computed using the niPCE and iPCE methods. Computational cost, in EFS, is also provided.

In figs. 7.16 to 7.18, the non-zero fields of the iPCE primal coefficients are presented.

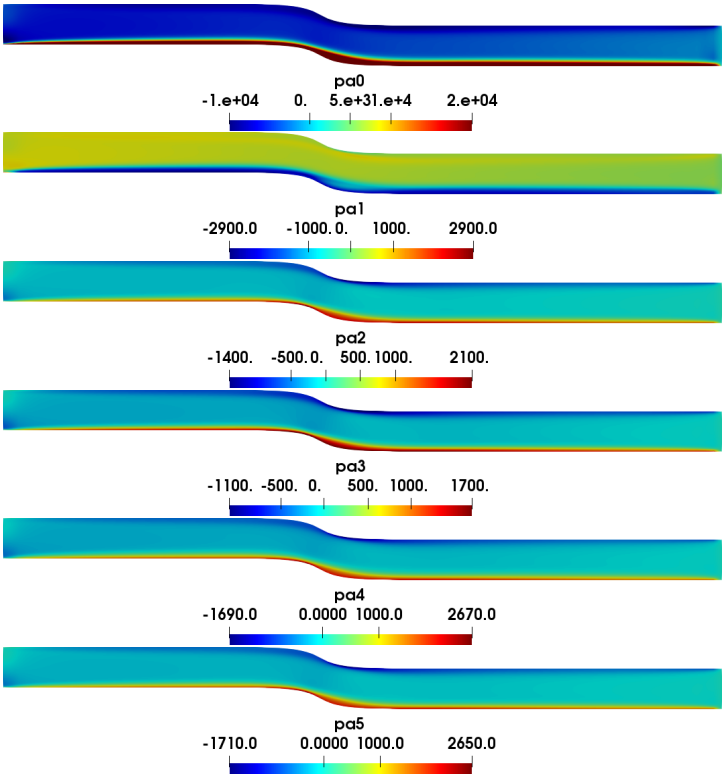


Figure 7.8: Case 13: PCE adjoint pressure fields over the initial geometry, for $J' = \mu_J$.

7.5 Conclusions

In this chapter, the intrusive Polynomial Chaos Expansion (iPCE) method was developed, programmed and applied for the primal CHT equations and laminar flows; the iPCE primal PDEs were differentiated w.r.t. the design variables to acquire iPCE adjoint equations. Extension to turbulent flows is possible, though this chapter is dealing only with laminar flows. The mathematical development was made for $K = 2$ independent uncertain variables, by assuming that these undergo normal distributions, and chaos order $\varpi = 2$. Uncertainties were related to the magnitude of the inlet velocity, the inlet fluid temperature and the thickness of thermal insulation. In the latter case, the iPCE primal FSI conditions couple PCE temperature fields T_k^F, T_k^S of different values of $k = 0, ..., M$, in contrast the rest of the iPCE boundary conditions. A disadvantage of the iPCE method is that, by changing either ϖ or K, the necessary number $M+1$ of the PCE coefficient fields changes and thus, the deterministic primal and adjoint variables expansions change. This would lead to new iPCE primal and adjoint PDEs, boundary conditions and adjoint SDs expressions and, requires new mathematical development and programming. In terms of computational cost, though, the iPCE method is superior to the niPCE one; both of them are far cheaper than the Monte

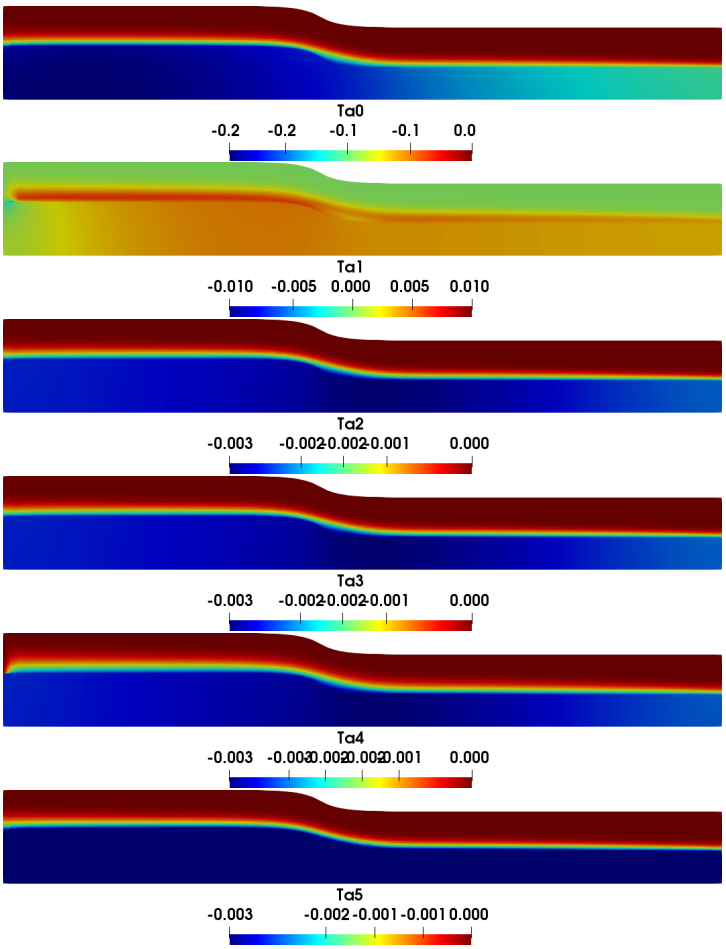


Figure 7.9: Case 13: PCE adjoint temperature fields over the fluid and solid domains of the initial geometry, for $J' = \mu_J$.

Carlo method. By selecting chaos order $\varpi = 2$ and $K = 2$, the cost of solving the iPCE equations is equal to $M = 6$ EFS, whereas the niPCE method required $(\varpi + 1)^K = 9$ EFS. Comparisons with the Monte Carlo method yielded that, in CHT problems with two uncertain variables, $K = 2$, selecting chaos order $\varpi = 2$ was enough to guarantee high accuracy in the computed μ_J and σ_J by the iPCE and niPCE methods. Differences between the iPCE and Monte Carlo methods were found to be below 0.1 % and 0.3 %, respectively; differences between the iPCE and niPCE methods were smaller than 0.1 %. In order to reduce the cost of shape optimization under uncertainties, compared to the niPCE method based on Galerkin projections, the iPCE adjoint PDEs, boundary conditions and SDs expressions were derived; the latter have been verified against FDs. In the studied cases, it was observed that the differences between the primal deterministic and PCE fields indexed by 0 were quite small, yielding that the latter are the dominant PCE fields when expanding the deterministic fields; this was not the case how-

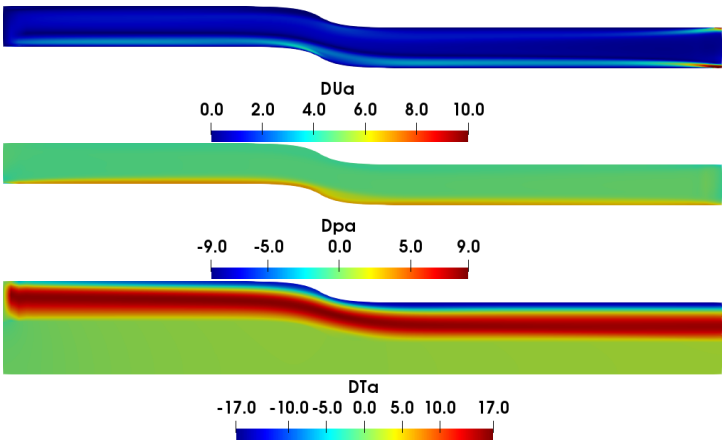


Figure 7.10: Case 13: Absolute differences with the max. value of the deterministic field, in (%) of the adjoint deterministic fields and PCE fields indexed by 0, computed for $J' = \mu_J$, over the initial geometry, for the adjoint velocity magnitude, pressure and temperature fields, from top to bottom, respectively.

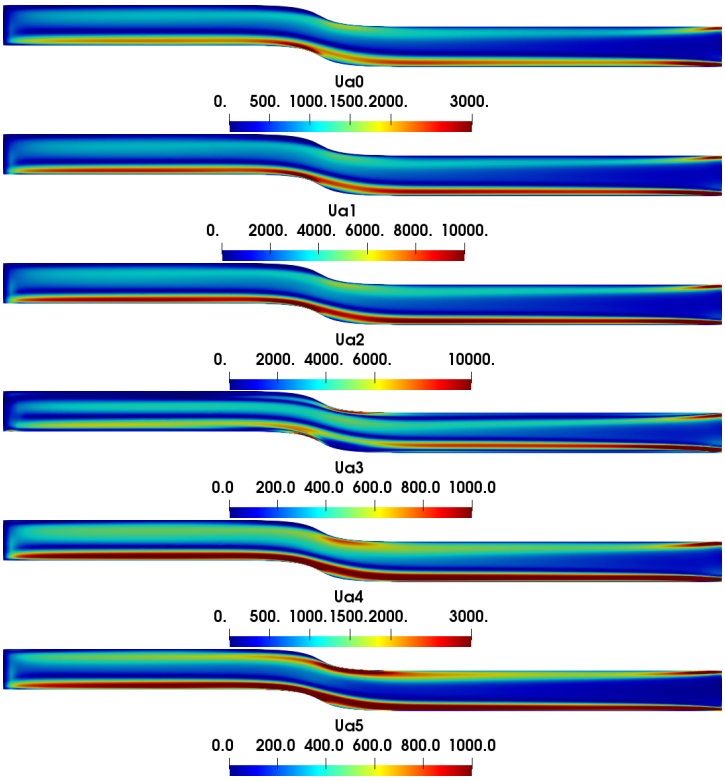


Figure 7.11: Case 13: Magnitude of PCE adjoint velocity fields over the initial geometry, for $J' = \sigma_J$.

ever for their adjoint counterparts. Shape optimizations targeting min. μ_J and σ_J of J^{meanT} were performed for an S-shaped duct attached to a solid body. To

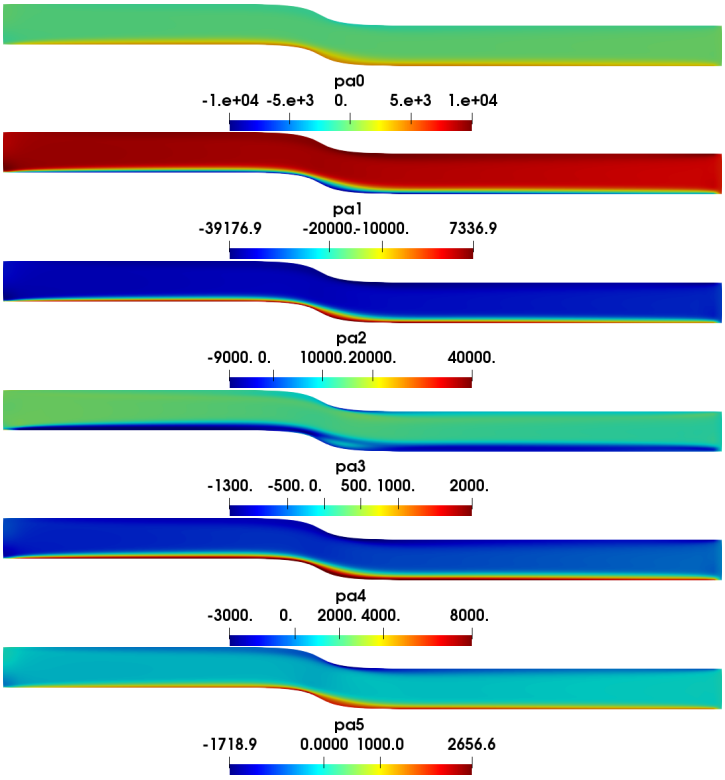


Figure 7.12: Case 13: PCE adjoint pressure fields over the initial geometry, for $J' = \sigma_J$.

reach the minimum values of μ_J and σ_J, similar trends in the deformation of the geometry were observed.

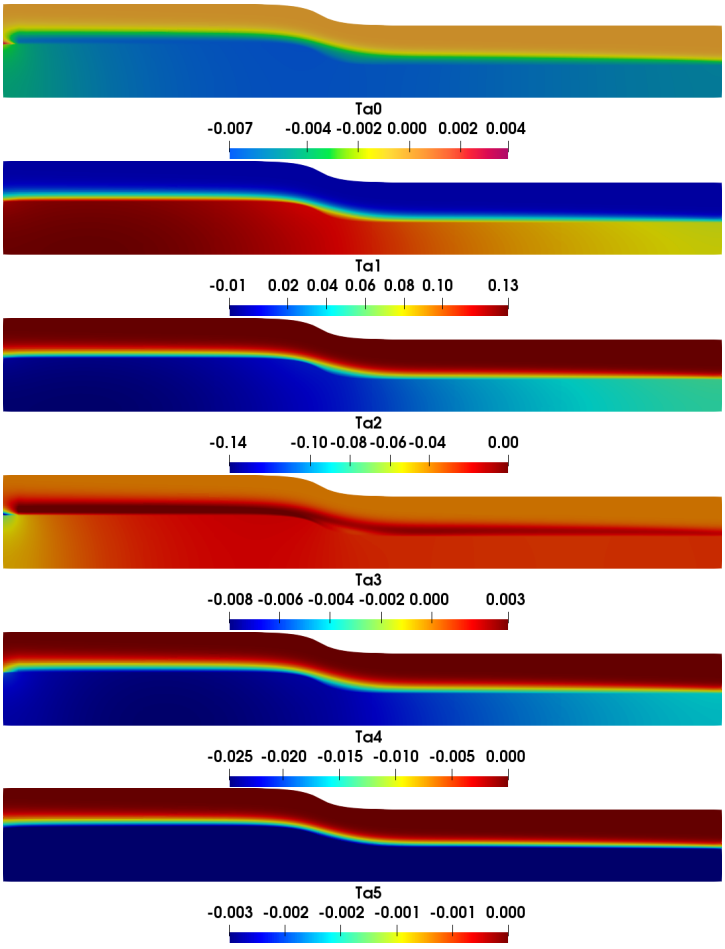


Figure 7.13: Case 13: PCE adjoint temperature fields over the fluid and solid domains of the initial geometry, for $J' = \sigma_J$.

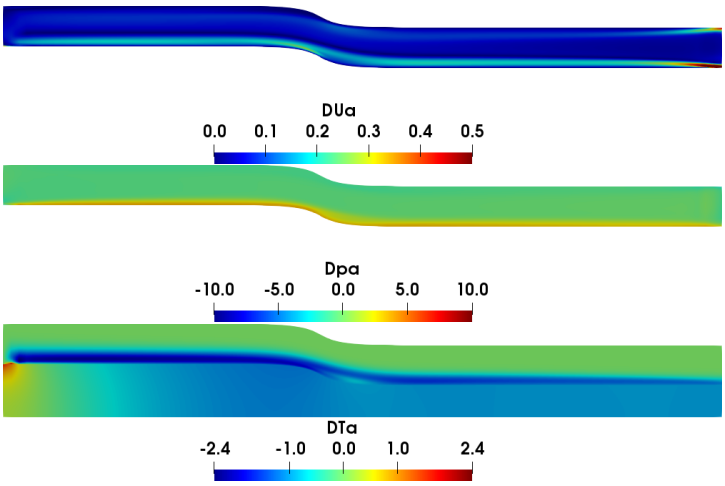
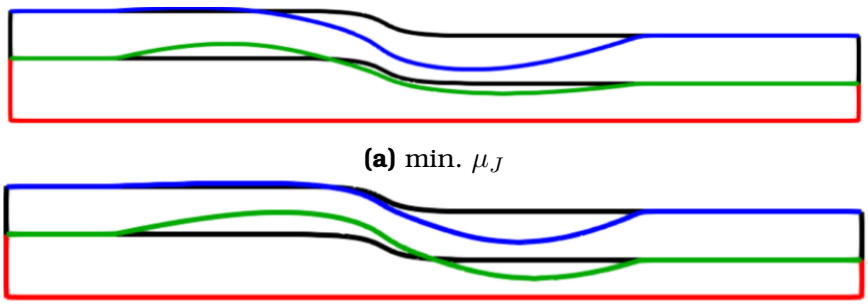


Figure 7.14: Case 13: Absolute differences with the max. value of the deterministic field, in (%) of the adjoint deterministic fields and PCE fields indexed by 0, computed for $J' = \sigma_J$, over the initial geometry.



(a) min. μ_J

(b) min. σ_J

Figure 7.15: Case 13: Optimized geometries for (a) min. μ_J and (b) min. σ_J, along with the initial geometry (in black). For min. μ_J, a reduction of 12.6 % was achieved, with a reduction appearing in the σ_J value, by 6.6 %. For min. σ_J, a reduction by almost 7.2 % was achieved together with a reduction by 5.9 % in the value of μ_J.

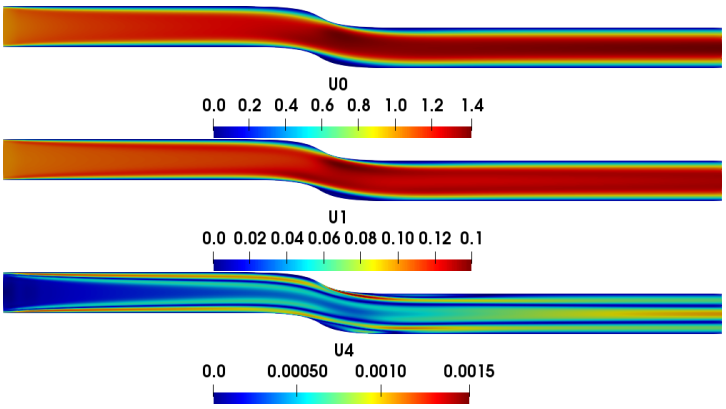


Figure 7.16: Case 14: Magnitude of PCE velocity fields over the initial geometry. Fields regarding the PCE coefficients indexed by $2, 3, 5$ are not plotted as their values are practically zero.

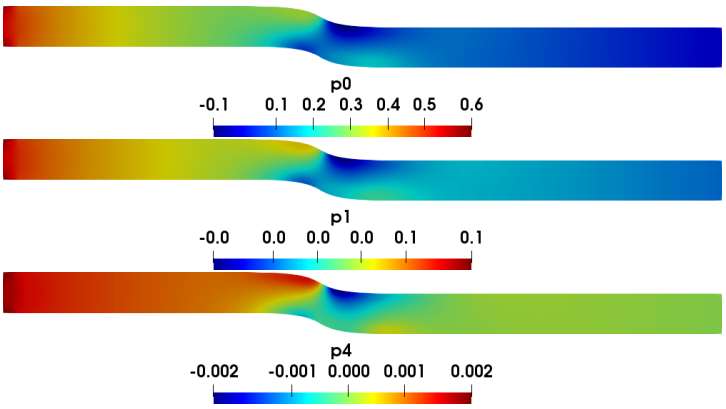


Figure 7.17: Case 14: PCE pressure fields over the initial geometry. Fields regarding the PCE coefficients indexed by $2, 3, 5$ are not plotted as their values are practically zero.

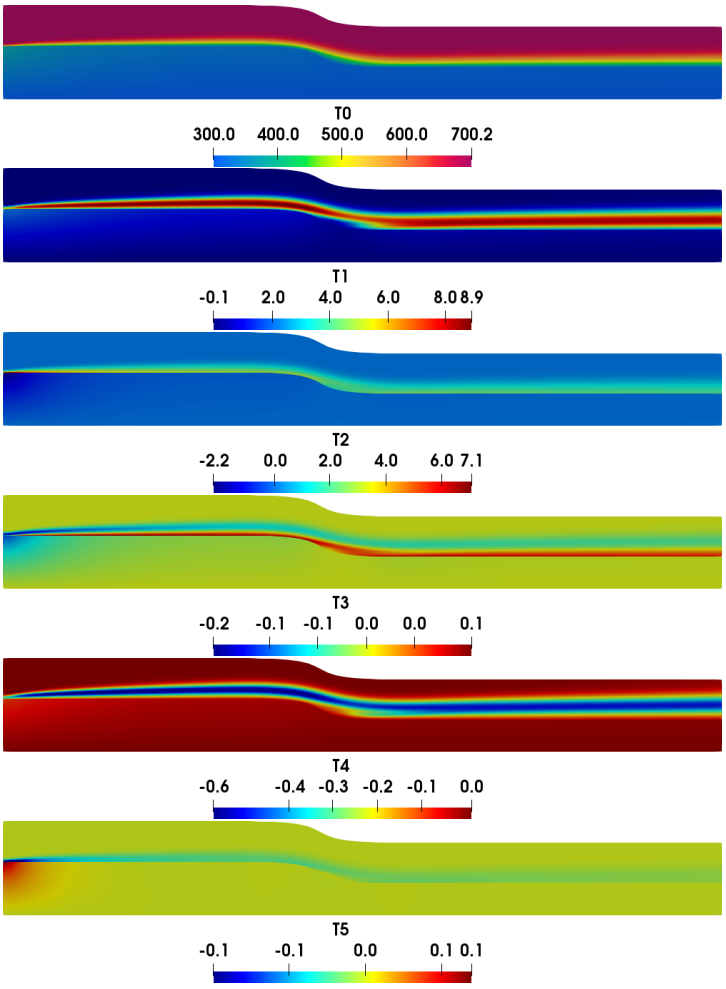


Figure 7.18: Case 14: PCE temperature fields over the fluid and solid domains of the initial geometry.

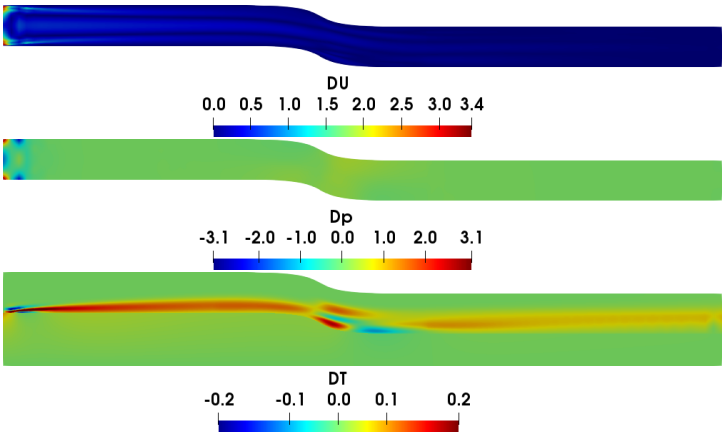


Figure 7.19: Case 14: Difference (%) of the primal deterministic fields and PCE fields indexed by 0, over the initial geometry. Top and middle: Absolute differences for velocity magnitude and pressure fields, respectively, normalized with the max. value of the corresponding deterministic field. Bottom: Relative difference for the temperature.

Chapter 8

Closure

8.1 Conclusions

The purpose of this thesis was to extend and further develop/program adjoint-assisted shape optimization methods and tools for pure fluid flow and CHT problems. In addition, methods for multi-objective problems, with or without the presence of uncertainties were developed. Steady-state flows of incompressible fluids were studied and turbulence was considered by using the Spalart–Allmaras model. The know-how acquired from pure fluid flow problems for computing accurate SDs at low computational cost, such as by differentiating the turbulence model equation [175, 125] and developing adjoint grid displacement equations [70], was extended to CHT problems. The developed methods were first tested in 2D academic cases and, then, applied to 3D ones. The relevant software has been programmed in the open-source CFD toolbox OpenFOAM©. In the next paragraphs, the methods developed in this thesis along with the conclusions drawn from each chapter are summarized.

- First, the derivation of the *FI* and *E-SI* continuous adjoint methods was presented, by differentiating the Spalart–Allmaras and Eikonal (Hamilton-Jacoby) equations. The less accurate (Severed) *SI* adjoint method was occasionally used too and comparisons were made. In contrast to the (Severed) *SI* adjoint, which assumes that the internal grid nodes are fixed, in the *FI* and *E-SI* adjoints, internal grid displacement is considered in different ways, leading to more accurate SDs. The *FI* adjoint led to SDs with field integrals of the grid nodes variations w.r.t. the design variables, a.k.a. the "grid sensitivities". To compute the latter, PDEs arising after differentiating w.r.t. the design variables the Laplace PDEs, which are assumed to govern the internal grid displacement, are solved. The relevant cost is N equivalent grid displacements, with N being the number of the design variables. In the *E-SI* adjoint, the adjoint to the Laplace PDEs, which governs the internal grid

displacement, was derived to provide the adjoint grid displacement field. In this way, the costly computation of grid sensitivities is avoided. Results known from [70] were reconfirmed in two new applications, in internal and external aerodynamics. Both the *FI* and *E-SI* adjoints are able to reproduce the SDs computed by FDs, whereas discrepancies appeared when the (Severed) *SI* method is used. The differences between the *FI* and *E-SI* adjoints are so small that they could not affect the outcome of shape optimizations driven by the BFGS method; it is important to keep in mind that the latter is known to be quite sensitive to the SDs accuracy, since it relies on Hessian matrices approximated by differences.

- The *FI* and *E-SI* continuous adjoint methods with turbulent flows were extended from pure fluid flows to CHT problems. The differentiation of the fluid energy PDE resulted into new terms in the adjoint momentum and turbulence model equations; regarding the *E-SI* adjoint, also the adjoint grid displacement PDEs in the fluid were enriched by new terms. In the solid, the differentiation of the primal PDEs led to the adjoint heat conduction and grid displacement equations. When solving the adjoint PDEs, the "coupling" between the fluid and solid disciplines follows FSI conditions similar to the primal problem; additional terms in the adjoint FSI conditions emerge only when objective functions are defined along the FSI. SDs computed by the *FI* and *E-SI* adjoints for the mean solid temperature and the volume of the solid within a user-defined temperature threshold reproduced the FDs. On the other hand, the (Severed) *SI* adjoint led to wrongly signed SDs and two orders of magnitude off, compared to FDs. Regarding the cost, the *E-SI* adjoint was cheaper than the *FI* one, since the computation of grid sensitivities over the whole domain is avoided, by solving the adjoint grid displacement PDEs. These results are in full agreement with those reached for pure fluid flows in [70] and reconfirmed in two new cases in Chapter 2. Moreover, it was demonstrated that the exact differentiation of turbulence model equations is crucial for computing accurate SDs.

- By using the adjoint method for CHT problems with turbulent flows, a series of 2D and 3D optimization problems, with or without constraints, were investigated; applications vary from a 2D turbine cascade vane and cooling ducts to 3D cooling serpentines, cooling fins and a car engine cylinder head. The cooling efficiency in internal cooling systems, expressed by either the heat flux absorbed by the coolant or the max. solid temperatures, was improved by increasing both the FSI area and the flow velocity magnitude of the coolant since, in this way, the heat conduction and temperature convection were enhanced. Also, the effect on the total pressure losses of the coolant flowing inside the cooling system of a car-engine cylinder-head, when minimizing max. T^S values, and vice-versa, was investigated. It was

observed that a single design of the cooling channel cannot minimize both targets at the same time. This was justified by the opposite trends in the change of the velocity magnitude. Regarding the cascade vane, by reducing the distance between the cooling holes and the cascade contour, especially close to the overheated leading edge, the cooling was improved and the mean solid temperature reduced.

- It was demonstrated that the choice of the GDM has negligible effect on the computed SDs. For internal and external aerodynamics and CHT problems, four different GDMs, based on Laplace PDEs, a volumetric B-Splines morpher, Delaunay Graph and the IDW method were considered and FDs yielded similar SDs. Assuming that FDs compute reference SDs, this outcome is equally significant for both continuous and discrete adjoint methods. A term-by-term analysis on the *FI*-based SDs shed light into the findings. It was revealed that close to the parameterized/changing boundaries: (a) grid sensitivities are practically the same, irrelevant of the GDM in use, and (b) the term (that depends on primal and adjoint fields) which multiplies the grid sensitivities inside field integrals takes on its highest values; that term decays rapidly as moving away from the wall. As a result, almost identical SD values arose for the various GDMs, despite large differences in $\delta x_i/\delta b_n$, for some GDMs, away from the walls. Differences in the adjoint SDs were so small that the outcome of shape optimizations was practically the same, even when the BFGS method, which is sensitive to the SDs accuracy, was used. Based on these findings, it was concluded for the cases investigated in this chapter, that the SD values are not affected by the GDM choice. Also, the *E-SI* adjoint which avoids the computation of grid sensitivities by making the convenient assumption of a Laplace GDM, can safely be used to compute SDs at low cost, irrespective of the GDM used in the optimization.

- Three variants of a prediction-correction algorithm conceptually based on [148] were developed, in order to compute Pareto fronts of non-dominated solutions with gradient-based algorithms. The three variants were initialized by a Pareto front member, acquired through a SOO run. After ensuring that the three variants compute similar fronts, they were compared in terms of cost. In all variants, a common feature is that the computation of the exact Hessian on the l.h.s. of the prediction and correction steps systems is avoided. Instead, a linear-Restarted GMRES solver assisted by Hessian-vector products and the damped BFGS method providing Hessian approximations through adjoint SDs, were used. To compute Hessian-vector products at low cost, DD was applied to the primal and adjoint equations. First, the three variants were compared for computing Pareto fronts on the drag-lift plane of two isolated airfoils, while preserving the airfoil area. The variant that proved to be the less expensive was the one approximating the

Hessian through the BFGS method in both the prediction and correction steps. The variant omitting the prediction step and solving constrained optimization problems through the SQP method proved to be more expensive. It required $\sim 20\%$ more EFS, since more optimization cycles and also more objective function re-evaluations due to the use of line-search were both needed, for the same number of PPs. Also, at the first step, this variant often failed to lead close to the next PP. Thus, including the prediction step has led to gain in computational cost. Computing Hessian-vector products in the prediction step required the solution of additional PDEs and as a result, led to a variant with higher cost. To keep the cost as low as possible, the min. number of bases were used in the GMRES solver, requiring the solution of the newly derived PDEs only once. This did not prevent though the prediction step from computing a point close to the next PP. The less expensive prediction-correction variant was also 2 to 3 times less expensive than the weighted sum method, with the additional advantage of controlling the distance between the PPs in the objective space. The prediction-correction variant that consistently proved to be the less expensive, was used to compute the Pareto front for a 3D CHT problem with a turbulent flow. The total pressure losses of the coolant and max. solid temperature were the contradicting targets and 10 Pareto front members were computed at the affordable cost of 150 EFS.

- To perform UQ at low cost, the development of the iPCE method for the primal CHT problems with laminar flows was carried out, with uncertainties at the inlet fluid conditions and the thermal insulation thickness. Associating uncertainties with the inlet fluid conditions leads to uncoupled boundary conditions, for the iPCE coefficients. On the other hand, regarding the thermal insulation thickness, coupled iPCE boundary conditions emerged. In terms of accuracy, the differences between the iPCE-based estimations of the mean value and standard deviation and those computed by the Monte Carlo and niPCE methods, were below 0.3 %. Comparisons between the deterministic primal fields and the PCE ones of zero index yielded that the latter are the significant components of the expanded deterministic variables. However, larger differences occurred in their adjoint counterparts. A comparison between the iPCE method with the niPCE yielded that the former requires 67 % of the cost of the latter, to estimate the mean value and standard deviation of a QoI, for the same chaos order; both required orders of magnitude less EFS compared to the Monte Carlo method which was also implemented.

- To perform OuU at reasonable computational cost, new adjoint equations, boundary conditions and SDs expressions were derived after differentiating w.r.t. the design variables the iPCE primal equations. Instead of solving the

deterministic adjoint PDEs multiple times, at various Gaussian nodes, to compute the mean value and standard deviation through the niPCE method at higher computational cost, the iPCE adjoint PDEs were solved and the so-computed SDs proved to be accurate, compared with FDs. Adjoint iPCE SDs were used to assist independent shape optimizations for min. mean value and standard deviation of the mean solid temperature; similar shape modifications emerged for both targets.

8.2 Novel Contributions

- The continuous adjoint method for CHT shape optimization was developed. The adjoint turbulent equations for CHT problems were derived for the first time in the literature. Also, the adjoint grid displacement PDEs for the fluid and solid regions, to assist the *E-SI* adjoint method, were developed for CHT problems for the first time.
- The effect of the selected GDM on the SDs values was investigated for a variety of problems, including internal and external pure fluid flows and CHT problems. The similarities among the SDs computed for different GDMs is a clear indication that there is no need to derive new adjoint grid displacement PDEs, in the *E-SI* adjoint, when GDMs different than the Laplace PDEs are used. A term-by-term analysis of the *FI*-based SDs shed light into the reasons of the similarities of the SDs for different GDMs.
- Three variants of a prediction-correction algorithm conceptually based on [148] were developed and applied in Pareto front computations for aerodynamic and CHT shape optimization problems, for the first time. To compute Hessian-vector products at the lowest possible cost and solve the prediction step equations by means of a Krylov subspace method, DD was applied to the primal and adjoint equations. Also, a comparison between the least expensive prediction-correction variant and the weighted-sum minimization method was carried out.
- The iPCE method was developed and applied for the first time in CHT problems, allowing the efficient estimation of the mean value and standard deviation of QoI, as well as their derivatives. To do so, new primal and adjoint PDEs and boundary conditions for environmental and manufacturing uncertainties were derived. Also, expansions of objective functions defined either in the fluid or solid domains were made.

8.3 Publications and Conference Presentations

Journal Publications:

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. "The continuous adjoint method for shape optimization in conjugate heat transfer problems with turbulent incompressible flows". Applied Thermal Engineering, 140:351-362, 2018.

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, A.G. Tsolovikos, K.C Giannakoglou. "Effect of Grid Displacement Models on Sensitivity Derivatives computed by Continuous Adjoint in Aerodynamic and Conjugate Heat Transfer Shape Optimization". Engineering Optimization (doi 10.1080/0305215X.2020.1796998).

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, K.C Giannakoglou. "Adjoint-assisted Pareto Front Tracing in Aerodynamic and Conjugate Heat Transfer Shape Optimization". Computers and Fluids (accepted).

Conference Papers:

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, K.C. Giannakoglou, "Conjugate heat transfer shape optimization of internal cooling systems using continuous adjoint in OpenFOAM". 7th OpenFOAM Conference, Berlin, Germany, 2019.

- K. Gkaragkounis and E. Papoutsis-Kiachagias and V. Asouti and K. Giannakoglou. "Adjoint-based Pareto front tracing in aerodynamic shape optimization". In ICCFD10-2018-322, Barcelona, Spain, 2018.

- E.M. Papoutsis-Kiachagias, J. Koch, K. Gkaragkounis, K. Giannakoglou. "A Continuous Adjoint Framework for Shape and Topology Optimization and their Synergistic Use". 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Florida, USA, 2018.

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, K.C. Giannakoglou . "Conjugate Heat Transfer shape optimization based on the continuous adjoint method". VII International Conference on Coupled Problems in Science and Engineering, Rhodes,Greece, 2017.

- K.C. Giannakoglou, E.M. Papoutsis-Kiachagias, I.S. Kavvadias, K.T. Gkaragkounis. "Continuous Adjoint in Shape & Topology Optimization-Recent Developments & Applications". Seminar on Adjoint CFD Methods in Industry and Research, NAFEMS, 2016.

Invited Lectures:

- K.C. Giannakoglou, E.M. Papoutsis-Kiachagias, K.T. Gkaragkounis. "The Continuous Adjoint Method in Aero/Hydrodynamic Optimization, including Conjugate Heat Transfer Problems". Von Karman Institute Lecture Series, 2020.

- K.C. Giannakoglou, E.M. Papoutsis-Kiachagias, K.T. Gkaragkounis, K.D. Samouchos, C.A. Vezyris, C.A. J.R.L. Koch. "The continuous adjoint method in aero/hydrodynamic optimization" . Von Karman Institute Lecture Series, 2019.

Awards:

- **Academic Best Paper Award:** K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, K.C. Giannakoglou, "Conjugate heat transfer shape optimization of internal cooling systems using continuous adjoint in OpenFOAM". 7th OpenFOAM Conference, Berlin, Germany, 2019.

8.4 Future Work

- Continuous adjoint methods for steady-state CHT problems could be extended to unsteady ones. To do so, the steady-state adjoint PDEs should include terms arising from the differentiation of the transient terms. Since the unsteady adjoint PDEs must be solved backwards in time, the computational overhead regarding the memory and storage requirements could be lifted by using either checkpointing methods [79], reduced-order models such as [162] or efficient algorithms for data compression [168]. Recent research in the PCOpt/NTUA regarding these techniques can be found in [146, 96].

- Adjoint methods for CHT problems may be enriched to account for the dependence of the specific heat transfer coefficient, thermal conductivity and viscosity on temperature. Buoyancy effects could also be considered by developing the adjoint to the Boussinesq approximation [18].

- The adjoint to other turbulence models than the Spalart–Allmaras one, such as the $k-\omega$ SST [71] and the k-ϵ [130] ones, could be extended from pure fluid flow to CHT problems.

- The Pareto front tracing methods presented in this thesis can be extended to deal with problems with more than two objective functions. This requires the development of an algorithm providing the target values $\hat{J}_i$ of the various objective functions, so that a well-distributed Pareto front is computed at low computational cost.

- The iPCE method for the primal and adjoint CHT problems developed in this thesis can be extended to turbulent flows.

- Pareto front tracing methods developed in this thesis could be extended to topology optimization for pure fluid flow and CHT problems. The number

of design variables in topology optimization is equal to the number of the cells of the fluid and solid regions. Thus, computational gains are expected to arise by applying gradient-based methods, instead of evolutionary algorithms, in this kind of problems. On the other hand, when using gradient-based methods, BFGS turns to be inefficient in terms of computational cost, due to the necessity of storing dense square matrices, with dimensions equal to the cell number. Thus, to deal with topology optimization problems, gradient-based methods presented in this thesis could be extended, by computing Hessian-vector products in the prediction step with methods inspired by the truncated Newton methods for topology optimization in [125] or by approximating the Hessian with the Limited-Memory BFGS (L-BFGS) method, [113]. Also, the developed prediction-correction algorithms could be enriched by using the MMA method [154] at the correction step.

Appendix A

Derivation of Adjoint Boundary Conditions in Pure Flow Problems

In this Appendix, the detailed derivation of the adjoint boundary conditions for steady-state flows of turbulent incompressible fluids is presented. The derivation of the adjoint boundary conditions, presented for the *FI* and *E-SI* adjoints, has many similarities with [125], which were derived for developing the *SI* adjoint method. The main difference with [125] is that, here, adjoint boundary conditions are derived based on the material derivative $\delta()/\delta b_n$ of the primal variables instead of $\partial()/\partial b_n$, as in [125], in which the Leibniz theorem for integral variations is used to derive the adjoint method. After satisfying the field adjoint equations in eq. 2.46, the remaining terms in the derivative of L

$$
\begin{aligned}
\frac{\delta L}{\delta b_n} = & \int_S \left(u_i v_j n_j + \tau_{ij}^a n_j - q n_i + \widetilde{\nu}_a \widetilde{\nu} \frac{C_Y}{Y} e_{jql} \frac{\partial v_l}{\partial x_q} e_{jki} n_k + \frac{\partial j_{S_k}}{\partial v_i} n_k + j_{S,i}^v \right) \frac{\delta v_i}{\delta b_n} dS \\
& + \int_S \left(u_i n_i + \frac{\partial j_{S_i}}{\partial p} n_i + j_S^p \right) \frac{\delta p}{\delta b_n} dS + \int_S \left(-u_i n_j + \frac{\partial j_{S_k}}{\partial \tau_{ij}} n_k \right) \frac{\delta \tau_{ij}}{\delta b_n} dS \\
& + \int_S \left[\widetilde{\nu}_a v_j n_j - \frac{\widetilde{\nu}_a}{\sigma} \frac{\partial \widetilde{\nu}}{\partial n} + \frac{\partial \widetilde{\nu}_a}{\partial n} \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) - 2 \widetilde{\nu}_a \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial n} + j_S^{\widetilde{\nu}} \right] \frac{\delta \widetilde{\nu}}{\delta b_n} dS \\
& - \int_S \widetilde{\nu}_a n_j \left(\frac{\nu + \widetilde{\nu}}{\sigma} \right) \frac{\delta}{\delta b_n} \left(\frac{\partial \widetilde{\nu}}{\partial x_j} \right) dS + \int_S \left[\Delta^a \frac{\partial \Delta}{\partial n} + \Delta^a c_j n_j \right] \frac{\delta \Delta}{\delta b_n} dS \\
& + \int_\Omega \left(-u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q \frac{\partial v_j}{\partial x_k} + \Theta_{jk} \right) \frac{\partial}{\partial x_j} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega \\
& + \int_\Omega j_\Omega \frac{\partial}{\partial x_k} \left(\frac{\delta x_k}{\delta b_n} \right) d\Omega + \int_{S_{W_p}} j_{S_i,k}^g \frac{\delta x_k}{\delta b_n} n_i dS + \int_{S_{W_p}} j_{S,i} \frac{\delta \left(n_i dS \right)}{\delta b_n}
\end{aligned}
\tag{A.1}
$$

are used to derive the adjoint boundary conditions.

A.1 Inlet Boundaries, S_I

Along S_I, Dirichlet conditions are imposed on v_i and $\widetilde{\nu}$ and a zero Neumann condition on p. Thus, $\delta v_i/\delta b_n = 0$ and $\delta\widetilde{\nu}/\delta b_n = 0$. As a result, the first and fourth integrals in eq. A.1, written along S_I, vanish.

The second and third integral in eq. A.1 are eliminated by setting

$$u_j n_j = u_{\langle n\rangle} = -\frac{\partial j_{S_{I,i}}}{\partial p} n_i - j^p_{S_I} \tag{A.2a}$$

$$u^I_{\langle t\rangle} = \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^I_i n_j + \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^I_j n_i \tag{A.2b}$$

$$u^{II}_{\langle t\rangle} = \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^{II}_i n_j + \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^{II}_j n_i \tag{A.2c}$$

where t^I_i, t^{II}_i are the components of the tangent to the surface unit vectors. The first tangent vector t^I_i is an arbitrarily selected unit vector parallel to S_I whereas t^{II}_i results from $t^{II}_i = e_{ijk} n_j t^I_k$, with e_{ijk} being the permutation symbol. Quantities $u^I_{\langle t\rangle}$ and $u^{II}_{\langle t\rangle}$ are the components of the adjoint velocity in the t^I_i, t^{II}_i directions, respectively. It should be noted that if J is not defined at S_I (for instance, the drag or the lift objective function), all the adjoint velocity components are zero along S_I.

Since no boundary condition for q results from the need to eliminate any of the boundary integrals in eq. A.1, a zero Neumann condition is imposed on it. Finally, to eliminate the fifth and sixth integrals in eq. A.1 written along S_I, $\widetilde{\nu_a} = 0$ and $\Delta^a = 0$.

Proof of eqs. A.2b and A.2c

The third integral on the r.h.s. of eq. A.1, written along S_I, is repeated below for the sake of completeness

$$\mathcal{T} = \int_{S_I} T_I dS = \int_{S_I} (-u_i n_j + \phi_{ij}) \frac{\delta \tau_{ij}}{\delta b_n} dS \tag{A.3}$$

where $\phi_{ij} = \frac{\partial j_{S_{I_k}}}{\partial \tau_{ij}} n_k$. In order to further develop the r.h.s. of eq. A.3, terms included in its integrand are expressed in the Frenet-Serret frame, which is an orthogonal system defined at each point of the inlet by the outward normal unit $\mathbf{n}$ and two tangential unit vectors $\mathbf{t^I}$, $\mathbf{t^{II}}$. Decomposition of vectors into a normal and two tangential components is performed by projecting them on the three unit vectors.

Term ϕ_{ij} is expressed in terms of its Frenet-Serret components, as follows

$$\begin{aligned}\phi_{ij} = &\phi_{\langle n\rangle\langle n\rangle} n_i n_j + \phi_{\langle t^I\rangle\langle n\rangle} t^I_i n_j + \phi_{\langle t^{II}\rangle\langle n\rangle} t^{II}_i n_j \\ &+ \phi_{\langle n\rangle\langle t^I\rangle} n_i t^I_j + \phi_{\langle t^I\rangle\langle t^I\rangle} t^I_i t^I_j + \phi_{\langle t^{II}\rangle\langle t^I\rangle} t^{II}_i t^I_j \\ &+ \phi_{\langle n\rangle\langle t^{II}\rangle} n_i t^{II}_j + \phi_{\langle t^I\rangle\langle t^{II}\rangle} t^I_i t^{II}_j + \phi_{\langle t^{II}\rangle\langle t^{II}\rangle} t^{II}_i t^{II}_j\end{aligned} \tag{A.4}$$

where $\phi_{\langle n\rangle\langle n\rangle} = \phi_{lm} n_l n_m$, $\phi_{\langle t^I\rangle\langle n\rangle} = \phi_{lm} t^I_l n_m$ and so forth.

The integrand of eq. A.3 is written as

$$T_I = -u_{\langle n\rangle} n_i n_j \frac{\delta \tau_{ij}}{\delta b_n} + (-u^I_{\langle t\rangle} t^I_i n_j - u^{II}_{\langle t\rangle} t^{II}_i n_j + \phi_{ij}) \frac{\delta \tau_{ij}}{\delta b_n} \tag{A.5}$$

Since S_I is fixed during the optimization, the first term on the r.h.s. of eq. A.5 is developed as

$$\begin{aligned}-u_{\langle n\rangle} n_i n_j \frac{\delta \tau_{ij}}{\delta b_n} &= -u_{\langle n\rangle} \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) n_i n_j \right] \\ &= -2 u_{\langle n\rangle} \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \frac{\partial v_{\langle n\rangle}}{\partial n} \right]\end{aligned} \tag{A.6}$$

Also,

$$\frac{\partial v_i}{\partial x_i} = \frac{\partial}{\partial x_i} \left(v_{\langle n\rangle} n_i + v^I_{\langle t\rangle} t^I_i + v^{II}_{\langle t\rangle} t^{II}_i \right) \approx \frac{\partial v_{\langle n\rangle}}{\partial n} + \frac{\partial v^I_{\langle t\rangle}}{\partial t^I} + \frac{\partial v^{II}_{\langle t\rangle}}{\partial t^{II}} = 0 \tag{A.7}$$

Then, the r.h.s. term in eq. A.6 is written as

$$-2 u_{\langle n\rangle} \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \frac{\partial v_{\langle n\rangle}}{\partial n} \right] = 2 u_{\langle n\rangle} \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \left(\frac{\partial v^I_{\langle t\rangle}}{\partial t^I} + \frac{\partial v^{II}_{\langle t\rangle}}{\partial t^{II}} \right) \right] \tag{A.8}$$

The term on the r.h.s. of eq. A.8 vanishes automatically for flows with a uniform tangential velocity profile at the inlet, which is the case for all flows examined in this thesis.

After using eq. A.4, the remaining terms in the integrand of eq. A.5 are

$$T_I = \Bigg[\underbrace{(-u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle}) t^I_i n_j}_{T1} + \underbrace{(-u^{II}_{\langle t\rangle} + \phi_{\langle t^{II}\rangle\langle n\rangle}) t^{II}_i n_j}_{T2} + \underbrace{\phi_{\langle n\rangle\langle n\rangle} n_i n_j}_{T3}$$

$$+\underbrace{\phi_{\langle n\rangle\langle t^I\rangle}n_i t_j^I}_{T4}+\underbrace{\phi_{\langle t^I\rangle\langle t^I\rangle}t_i^I t_j^I}_{T5}+\underbrace{\phi_{\langle t^{II}\rangle\langle t^I\rangle}t_i^{II}t_j^I}_{T6}+\underbrace{\phi_{\langle n\rangle\langle t^{II}\rangle}n_i t_j^{II}}_{T7}$$
$$+\underbrace{\phi_{\langle t^I\rangle\langle t^{II}\rangle}t_i^I t_j^{II}}_{T8}+\underbrace{\phi_{\langle t^{II}\rangle\langle t^{II}\rangle}t_i^{II}t_j^{II}}_{T9}\Bigg]\frac{\delta\tau_{ij}}{\delta b_n} \tag{A.9}$$

Following a procedure similar to that used for the first term in eq. A.5 (eqs. A.6 to A.8) and assuming a tangentially uniform inlet velocity profile

$$T_3\frac{\delta\tau_{ij}}{\delta b_n}=0 \tag{A.10}$$

Under the same assumption, a similar mathematical development for T_5, T_6, T_8, and T_9 yields

$$T_5\frac{\delta\tau_{ij}}{\delta b_n}=T_6\frac{\delta\tau_{ij}}{\delta b_n}=T_8\frac{\delta\tau_{ij}}{\delta b_n}=T_9\frac{\delta\tau_{ij}}{\delta b_n}=0 \tag{A.11}$$

since these terms depend only on the tangential derivatives of the tangential velocity components, assumed to be uniformly distributed.

Terms $T_1,\ T_2,\ T_4$ and T_7 are developed as

$$T_1\frac{\delta\tau_{ij}}{\delta b_n}=(-u^I_{\langle t\rangle}+\phi_{\langle t^I\rangle\langle n\rangle})t_i^I n_j\frac{\delta\tau_{ij}}{\delta b_n}=(-u^I_{\langle t\rangle}+\phi_{\langle t^I\rangle\langle n\rangle})\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)t_i^I n_j\right]$$
$$=(-u^I_{\langle t\rangle}+\phi_{\langle t^I\rangle\langle n\rangle})\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v^I_{\langle t\rangle}}{\partial n}+\frac{\partial v_{\langle n\rangle}}{\partial t^I}\right)\right] \tag{A.12}$$

$$T_2\frac{\delta\tau_{ij}}{\delta b_n}=(-u^{II}_{\langle t\rangle}+\phi_{\langle t^{II}\rangle\langle n\rangle})t_i^{II} n_j\frac{\delta\tau_{ij}}{\delta b_n}=(-u^{II}_{\langle t\rangle}+\phi_{\langle t^{II}\rangle\langle n\rangle})\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)t_i^{II} n_j\right]$$
$$=(-u^{II}_{\langle t\rangle}+\phi_{\langle t^{II}\rangle\langle n\rangle})\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v^{II}_{\langle t\rangle}}{\partial n}+\frac{\partial v_{\langle n\rangle}}{\partial t^{II}}\right)\right] \tag{A.13}$$

$$T_4\frac{\delta\tau_{ij}}{\delta b_n}=\phi_{\langle n\rangle\langle t^I\rangle}n_i t_j^I\frac{\delta\tau_{ij}}{\delta b_n}=\phi_{\langle n\rangle\langle t^I\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)n_i t_j^I\right]$$
$$=\phi_{\langle n\rangle\langle t^I\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v_{\langle n\rangle}}{\partial t^I}+\frac{\partial v^I_{\langle t\rangle}}{\partial n}\right)\right] \tag{A.14}$$

$$T_7\frac{\delta\tau_{ij}}{\delta b_n}=\phi_{\langle n\rangle\langle t^{II}\rangle}n_i t_j^{II}\frac{\delta\tau_{ij}}{\delta b_n}=\phi_{\langle n\rangle\langle t^{II}\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right)n_i t_j^{II}\right]$$
$$=\phi_{\langle n\rangle\langle t^{II}\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v_{\langle n\rangle}}{\partial t^{II}}+\frac{\partial v^{II}_{\langle t\rangle}}{\partial n}\right)\right] \tag{A.15}$$

Taking eqs. A.10 to A.15 into consideration, eq. A.9 becomes

$$T_I=(-u^I_{\langle t\rangle}+\phi_{\langle t^I\rangle\langle n\rangle}+\phi_{\langle n\rangle\langle t^I\rangle})\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v^I_{\langle t\rangle}}{\partial n}+\frac{\partial v_{\langle n\rangle}}{\partial t^I}\right)\right]$$
$$+(-u^{II}_{\langle t\rangle}+\phi_{\langle t^{II}\rangle\langle n\rangle}+\phi_{\langle n\rangle\langle t^{II}\rangle})\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v^{II}_{\langle t\rangle}}{\partial n}+\frac{\partial v_{\langle n\rangle}}{\partial t^{II}}\right)\right] \tag{A.16}$$

In order to make T_I independent of variations in stresses, their multipliers are set to zero, i.e.

$$u^I_{\langle t\rangle}=\phi_{\langle t^I\rangle\langle n\rangle}+\phi_{\langle n\rangle\langle t^I\rangle}=\phi_{ij}t_i^I n_j+\phi_{ij}n_i t_j^I$$
$$u^{II}_{\langle t\rangle}=\phi_{\langle t^{II}\rangle\langle n\rangle}+\phi_{\langle n\rangle\langle t^{II}\rangle}=\phi_{ij}t_i^{II} n_j+\phi_{ij}n_i t_j^{II}$$

giving rise to eqs. A.2b and A.2c. ■

A.2 Outlet Boundaries, S_O

Along S_O, Dirichlet condition is imposed on p and zero Neumann conditions on v_i and $\widetilde{\nu}$. With this in mind and since S_O is fixed, $\delta p/\delta b_n=0$ holds and the second integral in eq. A.1, written along S_O, vanishes. Also, $\delta n_j/\delta b_n=0$ and the fifth integral of eq. A.1 vanishes. An almost uniform velocity profile is assumed along S_O. Hence, the third term on the r.h.s. of eq. A.1 is neglected. In order to eliminate the first integral in the same equation, its integrand must be zeroed

$$u_i v_j n_j+\tau^a_{ij}n_j-qn_i+\frac{\partial j_{S_{O,k}}}{\partial v_i}n_k+\widetilde{\nu}_a\widetilde{\nu}\frac{C_Y}{Y}e_{jql}\frac{\partial v_l}{\partial x_q}e_{jki}n_k+j^v_{S_{O,i}}=0$$
$$i=1,2(,3) \tag{A.17}$$

In 3D flows, eq. A.17 includes four unknown quantities (the three components of the adjoint velocity and pressure. Therefore, one of them must be extrapolated from the interior of the domain. This is chosen to be the normal component of the adjoint velocity, $u_{\langle n\rangle}$. By multiplying eq. A.17 with the normal to the surface

vector, n_i, the adjoint pressure is determined

$$q = u_{\langle n\rangle} v_{\langle n\rangle} + 2(\nu+\nu_t)\frac{\partial u_{\langle n\rangle}}{\partial n} + \widetilde{\nu}_a\widetilde{\nu}\frac{C_Y}{Y} e_{jql}\frac{\partial v_l}{\partial x_q} e_{jki} n_k n_i + \frac{\partial j_{S_{O,k}}}{\partial v_i} n_k n_i + j^v_{S_{O,i}} n_i \tag{A.18}$$

The tangential adjoint velocity components is obtained by multiplying eq. A.17 with the tangent to the surface unit vectors, $t^l_i, l=1,2$

$$v_n u^l_{\langle t\rangle} + (\nu+\nu_t)\left(\frac{\partial u^l_{\langle t\rangle}}{\partial n} + \frac{\partial u_{\langle n\rangle}}{\partial t^l}\right) + \widetilde{\nu}_a\widetilde{\nu}\frac{C_Y}{Y} e_{jql}\frac{\partial v_l}{\partial x_q} e_{jki} n_k t^l_i + \frac{\partial j_{S_{O,k}}}{\partial v_i} n_k t^l_i + j^v_{S_{O,i}} t^l_i = 0 \tag{A.19}$$

and imposing the resulting Robin type condition.

Eq. A.17 is also treated in different ways to obtain the adjoint outlet conditions. For example, instead of using eqs. A.18 and A.19, the whole vector of the adjoint velocity is computed through eq. A.17, after imposing a zero Neumann condition on q. Numerical experiments have shown that the different formulations have a negligible impact on the computed SDs.

In order to eliminate the fourth integral in eq. A.1, the following condition

$$\widetilde{\nu}_a v_j n_j + \frac{\partial \widetilde{\nu}_a}{\partial n}\left(\frac{\nu+\widetilde{\nu}}{\sigma}\right) + j^{\widetilde{\nu}}_S = 0 \tag{A.20}$$

results for $\widetilde{\nu}_a$ along S_O. Finally, to eliminate the sixth integral of eq. A.1, written along S_O, $\Delta^a = 0$.

A.3 Symmetry Planes, S_S

Along the symmetry planes, if any, the normal velocity and the normal gradient of the tangential velocity components are zero ($v_{\langle n\rangle} = 0, \partial v^l_{\langle t\rangle}/\partial n = 0$) whereas zero Neumann conditions are imposed on p and $\widetilde{\nu}$, resulting for the latter to the elimination of the fifth integral of eq. A.1 along S_S. Since the objective function is never defined along the symmetry planes of the domain, quantities depending on J do not exist in the integrals of eq. A.1. In order to eliminate the second integral in the same equation, a zero normal adjoint velocity is imposed

$$u_{\langle n\rangle} = 0 \tag{A.21}$$

The integrand of the third integral in eq. A.1

$$\begin{aligned}
-u_i n_j \frac{\delta\tau_{ij}}{\delta b_n} &= -u_{\langle n\rangle} n_i n_j - u^l_{\langle t\rangle} t^l_i n_j)\frac{\delta\tau_{ij}}{\delta b_n} = -u^l_{\langle t\rangle} t^l_i n_j \frac{\delta\tau_{ij}}{\delta b_n} \\
&= u^l_{\langle t\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right) n_j t^l_i\right] \\
&= u^l_{\langle t\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v^l_{\langle t\rangle}}{\partial n}+\frac{\partial v_{\langle n\rangle}}{\partial t^l}\right)\right] = 0
\end{aligned}$$

vanishes automatically since $\partial v^l_{\langle t\rangle}/\partial n = 0$ and $\partial v_{\langle n\rangle}/\partial t^l = 0$ for the entire surface.

The first integrand in eq. A.1 can be expressed as

$$\begin{aligned}
\left(u_i v_{\langle n\rangle} + \tau^a_{ij} n_j - q n_i\right)\frac{\delta v_i}{\delta b_n} &= \left[(\nu+\nu_t)\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) n_j - q n_i\right]\frac{\delta v^l_{\langle t\rangle}}{\delta b_n} t^l_i \\
&= (\nu+\nu_t)\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) n_j t^l_i \frac{\delta v^l_{\langle t\rangle}}{\delta b_n} \\
&= (\nu+\nu_t)\left(\frac{\partial u_{\langle n\rangle}}{\partial t^l}+\frac{\partial u^l_{\langle t\rangle}}{\partial n}\right)\frac{\delta v^l_{\langle t\rangle}}{\delta b_n} \\
&= (\nu+\nu_t)\frac{\partial u^l_{\langle t\rangle}}{\partial n}\frac{\delta v^l_{\langle t\rangle}}{\delta b_n}
\end{aligned} \tag{A.22}$$

In order to eliminate this term in eq. A.22, the normal derivative of the tangential adjoint velocity is set to zero

$$\frac{\partial u^l_{\langle t\rangle}}{\partial n} = 0 \tag{A.23}$$

Eqs. A.21 and A.23 comprise the set of symmetry plane conditions for the adjoint velocity which are equivalent to those of the primal problem.

Similar to S_I, since no boundary condition for q results from eq. A.1, a zero Neumann condition is imposed. Regarding the fourth integral of the same equation, all of the integrated terms, are zero, since $v_n = 0$ and $\partial\widetilde{\nu}/\partial n = 0$, except for the third term, which is eliminated by imposing $\partial\widetilde{\nu}_a/\partial n = 0$. Also, along S_S, for the same reasons as for S_I and S_O, $\Delta^a = 0$.

A.4 Wall Boundaries, S_W

Since $v_i = 0$ and $\widetilde{\nu} = 0$ are imposed on S_{W_p}, $\delta v_i/\delta b_n = 0$ and $\delta\widetilde{\nu}/\delta b_n = 0$ and thus, the first and fourth integrals on the r.h.s. of eq. A.1 vanish. The second and third

integrals in eq. A.1 are eliminated by setting

$$u_{\langle n\rangle} = -\frac{\partial j_{S_{W_p,i}}}{\partial p} n_i - j^p_{S_{W_p}} \tag{A.24a}$$

$$u^I_{\langle t\rangle} = \frac{\partial j_{S_{W_p,k}}}{\partial \tau_{ij}} n_k t^I_i n_j + \frac{\partial j_{S_{W_p,k}}}{\partial \tau_{ij}} n_k t^I_j n_i \tag{A.24b}$$

$$u^{II}_{\langle t\rangle} = \frac{\partial j_{S_{W_p,k}}}{\partial \tau_{ij}} n_k t^{II}_i n_j + \frac{\partial j_{S_{W_p,k}}}{\partial \tau_{ij}} n_k t^{II}_j n_i \tag{A.24c}$$

It is convenient to chose t^I_i to be parallel to the primal velocity vector at the first cell centre adjacent to the wall.

The proof of eqs. A.24 is similar to that of eqs. A.2 for S_I, with a significant difference though: along the controlled wall boundaries S_{W_p}, $\delta x_k/\delta b_n \neq 0$. This gives rise to integrals contributing to the SDs.

To eliminate the fifth integral written along S_{W_p}, $\widetilde{\nu_a} = 0$. Along S_W, $\Delta = 0$ and the sixth integral in eq. A.1 is set to zero. Since no boundary condition for Δ^a results from the elimination of any of the boundary integrals in eq. A.1, a zero Neumann condition is imposed.

Proof of eqs. A.24

In order to eliminate the second integral in eq. A.1, written along S_{W_p}, its integrand is set to zero, giving rise to eq. A.24a.

For the sake of simplicity, let $\phi_{ij} = \frac{\partial j_{S_{W_p,k}}}{\partial \tau_{ij}} n_k$. The integrand of the third term on the r.h.s. of eq. A.1 is written as

$$T = (-u_i n_j + \phi_{ij}) \frac{\delta \tau_{ij}}{\delta b_n} = \underbrace{-u_{\langle n\rangle} n_i n_j \frac{\delta \tau_{ij}}{\delta b_n}}_{T_n} + \underbrace{(-u^I_{\langle t\rangle} t^I_i n_j - u^{II}_{\langle t\rangle} t^{II}_i n_j + \phi_{ij}) \frac{\delta \tau_{ij}}{\delta b_n}}_{T_t} \tag{A.25}$$

The first term on the r.h.s. of eq. A.25 is developed as

$$T_n = -u_{\langle n\rangle} n_i n_j \frac{\delta \tau_{ij}}{\delta b_n} = -u_{\langle n\rangle} \frac{\delta\left(\tau_{ij} n_i n_j\right)}{\delta b_n} + u_{\langle n\rangle} \tau_{ij} \frac{\delta (n_i n_j)}{\delta b_n} \tag{A.26}$$

Using eq. A.7 (this time written for S_{W_p}), the first term on the r.h.s. of the last equality in eq. A.26 is written as

$$-u_{\langle n\rangle} \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) n_i n_j \right] = 2 u_{\langle n\rangle} \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \frac{\partial v^l_{\langle t\rangle}}{\partial t^l} \right] \tag{A.27}$$

which vanishes, since a zero Dirichlet condition is imposed on the primal velocities along S_{W_p}. The two remaining terms in the same equation contribute to the SDs expression.

After taking eqs. A.26 and A.27 into consideration and decomposing ϕ_{ij} according to eq. A.4, eq. A.25 becomes

$$\begin{aligned} T = \Bigg[& \underbrace{(-u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle}) t^I_i n_j}_{T1} + \underbrace{(-u^{II}_{\langle t\rangle} + \phi_{\langle t^{II}\rangle\langle n\rangle}) t^{II}_i n_j}_{T2} + \underbrace{\phi_{\langle n\rangle\langle n\rangle} n_i n_j}_{T3} \\ & + \underbrace{\phi_{\langle n\rangle\langle t^I\rangle} n_i t^I_j}_{T4} + \underbrace{\phi_{\langle t^I\rangle\langle t^I\rangle} t^I_i t^I_j}_{T5} + \underbrace{\phi_{\langle t^{II}\rangle\langle t^I\rangle} t^{II}_i t^I_j}_{T6} + \underbrace{\phi_{\langle n\rangle\langle t^{II}\rangle} n_i t^{II}_j}_{T7} \\ & + \underbrace{\phi_{\langle t^I\rangle\langle t^{II}\rangle} t^I_i t^{II}_j}_{T8} + \underbrace{\phi_{\langle t^{II}\rangle\langle t^{II}\rangle} t^{II}_i t^{II}_j}_{T9} \Bigg] \frac{\delta \tau_{ij}}{\delta b_n} + u_{\langle n\rangle} \tau_{ij} \frac{\delta (n_i n_j)}{\delta b_n} \end{aligned} \tag{A.28}$$

The term depending on T_1 is developed as

$$\begin{aligned} T_1 \frac{\delta \tau_{ij}}{\delta b_n} &= (-u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle}) \frac{\delta \tau_{ij}}{\delta b_n} = (-u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle}) \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) t^I_i n_j \right] \\ & - (-u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle}) (\nu + \nu_t) \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \frac{\delta (t^I_i n_j)}{\delta b_n} \\ & = (-u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle}) \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \left(\frac{\partial v^I_{\langle t\rangle}}{\partial n} + \frac{\partial v_{\langle n\rangle}}{\partial t^I} \right) \right] \\ & - (-u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle}) \tau_{ij} \frac{\delta (t^I_i n_j)}{\delta b_n} \end{aligned} \tag{A.29}$$

Similarly, the rest of the terms on the r.h.s. of eq. A.28 read

$$\begin{aligned} T_2 \frac{\delta \tau_{ij}}{\delta b_n} &= (-u^{II}_{\langle t\rangle} + \phi_{\langle t^{II}\rangle\langle n\rangle}) \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \left(\frac{\partial v^{II}_{\langle t\rangle}}{\partial n} + \frac{\partial v_{\langle n\rangle}}{\partial t^{II}} \right) \right] \\ & - (-u^{II}_{\langle t\rangle} + \phi_{\langle t^{II}\rangle\langle n\rangle}) \tau_{ij} \frac{\delta (t^{II}_i n_j)}{\delta b_n} \end{aligned} \tag{A.30}$$

$$T_3 \frac{\delta \tau_{ij}}{\delta b_n} = \phi_{\langle n\rangle\langle n\rangle} n_i n_j \frac{\delta \tau_{ij}}{\delta b_n} = 2\phi_{\langle n\rangle\langle n\rangle} \frac{\delta}{\delta b_n} \left[(\nu + \nu_t) \frac{\partial v_{\langle n\rangle}}{\partial n} \right] - \phi_{\langle n\rangle\langle n\rangle} \tau_{ij} \frac{\delta (n_i n_j)}{\delta b_n}$$

$$= -\phi_{\langle n\rangle\langle n\rangle}\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n} \tag{A.31}$$

$$\begin{aligned} T_4\frac{\delta\tau_{ij}}{\delta b_n} &= \phi_{\langle n\rangle\langle t^I\rangle} n_i t_j^I \frac{\delta\tau_{ij}}{\delta b_n} \\ &= \phi_{\langle n\rangle\langle t^I\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v_{\langle n\rangle}}{\partial t^I} + \frac{\partial v^I_{\langle t\rangle}}{\partial n}\right)\right] - \phi_{\langle n\rangle\langle t^I\rangle}\tau_{ij}\frac{\delta(n_i t_j^I)}{\delta b_n} \end{aligned} \tag{A.32}$$

$$\begin{aligned} T_5\frac{\delta\tau_{ij}}{\delta b_n} &= \phi_{\langle t^I\rangle\langle t^I\rangle} t_i^I t_j^I \frac{\delta\tau_{ij}}{\delta b_n} = 2\phi_{\langle t^I\rangle\langle t^I\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\frac{\partial v^I_{\langle t\rangle}}{\partial t^I}\right] - \phi_{\langle t^I\rangle\langle t^I\rangle}\tau_{ij}\frac{\delta(t_i^I t_j^I)}{\delta b_n} \\ &= -\phi_{\langle t^I\rangle\langle t^I\rangle}\tau_{ij}\frac{\delta(t_i^I t_j^I)}{\delta b_n} \end{aligned} \tag{A.33}$$

$$\begin{aligned} T_6\frac{\delta\tau_{ij}}{\delta b_n} &= \phi_{\langle t^{II}\rangle\langle t^I\rangle} t_i^{II} t_j^I \frac{\delta\tau_{ij}}{\delta b_n} = \phi_{\langle t^{II}\rangle\langle t^I\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v^I_{\langle t\rangle}}{\partial t^{II}} + \frac{\partial v^{II}_{\langle t\rangle}}{\partial t^I}\right)\right] - \phi_{\langle t^{II}\rangle\langle t^I\rangle}\tau_{ij}\frac{\delta(t_i^{II} t_j^I)}{\delta b_n} \\ &= -\phi_{\langle t^{II}\rangle\langle t^I\rangle}\tau_{ij}\frac{\delta(t_i^{II} t_j^I)}{\delta b_n} \end{aligned} \tag{A.34}$$

$$\begin{aligned} T_7\frac{\delta\tau_{ij}}{\delta b_n} &= \phi_{\langle n\rangle\langle t^{II}\rangle} n_i t_j^{II} \frac{\delta\tau_{ij}}{\delta b_n} \\ &= \phi_{\langle n\rangle\langle t^{II}\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v_{\langle n\rangle}}{\partial t^{II}} + \frac{\partial v^{II}_{\langle t\rangle}}{\partial n}\right)\right] - \phi_{\langle n\rangle\langle t^{II}\rangle}\tau_{ij}\frac{\delta(n_i t_j^{II})}{\delta b_n} \end{aligned} \tag{A.35}$$

$$\begin{aligned} T_8\frac{\delta\tau_{ij}}{\delta b_n} &= \phi_{\langle t^I\rangle\langle t^{II}\rangle} t_i^I t_j^{II} \frac{\delta\tau_{ij}}{\delta b_n} \\ &= \phi_{\langle t^I\rangle\langle t^{II}\rangle}\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v^I_{\langle t\rangle}}{\partial t^{II}} + \frac{\partial v^{II}_{\langle t\rangle}}{\partial t^I}\right)\right] - \phi_{\langle t^I\rangle\langle t^{II}\rangle}\tau_{ij}\frac{\delta(t_i^I t_j^{II})}{\delta b_n} \\ &= -\phi_{\langle t^I\rangle\langle t^{II}\rangle}\tau_{ij}\frac{\delta(t_i^I t_j^{II})}{\delta b_n} \end{aligned} \tag{A.36}$$

$$
\begin{aligned}
T_9\frac{\delta\tau_{ij}}{\delta b_n} &= \phi_{\langle t^{II}\rangle\langle t^{II}\rangle} t_i^{II} t_j^{II} \frac{\delta\tau_{ij}}{\delta b_n} \\
&= 2\phi_{\langle t^{II}\rangle\langle t^{II}\rangle} \frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\frac{\partial v^{II}_{\langle t\rangle}}{\partial t^{II}}\right] - \phi_{\langle t^{II}\rangle\langle t^{II}\rangle}\tau_{ij}\frac{\delta(t_i^{II}t_j^{II})}{\delta b_n} \\
&= -\phi_{\langle t^{II}\rangle\langle t^{II}\rangle}\tau_{ij}\frac{\delta(t_i^{II}t_j^{II})}{\delta b_n}
\end{aligned}
\tag{A.37}
$$

After taking eqs. A.29 to A.37 into consideration, eq. A.28 takes the following form

$$
\begin{aligned}
T = &\left(-u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle} + \phi_{\langle n\rangle\langle t^I\rangle}\right)\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v^I_{\langle t\rangle}}{\partial n} + \frac{\partial v_{\langle n\rangle}}{\partial t^I}\right)\right] \\
&+\left(-u^{II}_{\langle t\rangle} + \phi_{\langle t^{II}\rangle\langle n\rangle} + \phi_{\langle n\rangle\langle t^{II}\rangle}\right)\frac{\delta}{\delta b_n}\left[(\nu+\nu_t)\left(\frac{\partial v^{II}_{\langle t\rangle}}{\partial n} + \frac{\partial v_{\langle n\rangle}}{\partial t^{II}}\right)\right] \\
&-\left(-u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle} + \phi_{\langle n\rangle\langle t^I\rangle}\right)\tau_{ij}\frac{\delta(t_i^I n_j)}{\delta b_n} - \left(-u^{II}_{\langle t\rangle} + \phi_{\langle t^{II}\rangle\langle n\rangle} + \phi_{\langle n\rangle\langle t^{II}\rangle}\right)\tau_{ij}\frac{\delta(t_i^{II} n_j)}{\delta b_n} \\
&-\left(-u_{\langle n\rangle} + \phi_{\langle n\rangle\langle n\rangle}\right)\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n} - \phi_{\langle t^I\rangle\langle t^I\rangle}\tau_{ij}\frac{\delta(t_i^I t_j^I)}{\delta b_n} - \left(\phi_{\langle t^{II}\rangle\langle t^I\rangle} + \phi_{\langle t^I\rangle\langle t^{II}\rangle}\right)\tau_{ij}\frac{\delta(t_i^{II} t_j^I)}{\delta b_n} \\
&-\phi_{\langle t^{II}\rangle\langle t^{II}\rangle}\tau_{ij}\frac{\delta(t_i^{II}t_j^{II})}{\delta b_n}
\end{aligned}
\tag{A.38}
$$

In order to make eq. A.38 independent of variations in stresses, the following conditions,

$$
\begin{aligned}
u^I_{\langle t\rangle} &= \phi_{\langle t^I\rangle\langle n\rangle} + \phi_{\langle n\rangle\langle t^I\rangle} \\
u^{II}_{\langle t\rangle} &= \phi_{\langle t^{II}\rangle\langle n\rangle} + \phi_{\langle n\rangle\langle t^{II}\rangle}
\end{aligned}
$$

must be met, giving rise to eqs. A.24b and A.24c. The remaining terms in eq. A.28

$$
\begin{aligned}
T = &-\left(-u_{\langle n\rangle} + \phi_{\langle n\rangle\langle n\rangle}\right)\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n} - \phi_{\langle t^I\rangle\langle t^I\rangle}\tau_{ij}\frac{\delta(t_i^I t_j^I)}{\delta b_n} \\
&-\left(\phi_{\langle t^{II}\rangle\langle t^I\rangle} + \phi_{\langle t^I\rangle\langle t^{II}\rangle}\right)\tau_{ij}\frac{\delta(t_i^{II} t_j^I)}{\delta b_n} - \phi_{\langle t^{II}\rangle\langle t^{II}\rangle}\tau_{ij}\frac{\delta(t_i^{II} t_j^{II})}{\delta b_n}
\end{aligned}
\tag{A.39}
$$

can be computed since they include quantities depending on the primal and adjoint variables as well as variations in the geometry, contributing to the SDs expression. These expressions are zero along uncontrolled boundaries $S_{W_{np}}$. ■

Appendix B

Objective Functions and their Differentiation

Below, the detailed differentiation of the objective functions J used in this PhD thesis is presented. The mathematical expressions of the integrands included in eqs. 3.17 and 3.18 corresponding to a generalized objective function, are also given below.

B.1 Forces Component

The function expressing the forces component along a specified direction $\mathbf{r}$, reads

$$J^{forces} = \int_{S_W} \left(p\delta_i^j - \tau_{ij}\right) r_i n_j dS \tag{B.1}$$

To compute the drag or the lift, $\mathbf{r}$ is defined to be parallel and perpendicular to the flow, respectively. The derivative of J^{forces} w.r.t. b_n is

$$\frac{\delta J^{forces}}{\delta b_n} = \int_{S_W} \left(\frac{\delta p}{\delta b_n}\delta_i^j - \frac{\delta \tau_{ij}}{\delta b_n}\right) r_i n_j dS + \int_{S_W} \left(p\delta_i^j - \tau_{ij}\right) r_i \frac{\delta\left(n_j dS\right)}{\delta b_n} \tag{B.2}$$

The last integral on the r.h.s. of eq. B.2 contributes to the SDs, whereas contributions to the adjoint boundary conditions arise from the first integral, where

$$\frac{\partial j_{Sj}}{\partial p} n_j = \delta_i^j r_i n_j, \qquad \frac{\partial j_{Sj}}{\partial \tau_{il}} n_j = -\delta_j^l r_i n_j,$$

and δ_i^j is the Kronecker delta.

B.2 Volume-Averaged Total Pressure Losses

The volume-weighted total pressure losses between the inlet S_I and the outlet S_O of the flow domain are defined as

$$J^{pt} = -\int_{S_{I,O}} \left(p + \frac{1}{2}v_j^2\right) v_i n_i dS \tag{B.3}$$

Differentiating eq. B.3 w.r.t. b_n yields

$$\frac{\delta J^{pt}}{\delta b_n} = -\int_{S_{I,O}} v_i n_i \frac{\delta p}{\delta b_n} dS - \int_{S_{I,O}} \left[v_i v_j n_j + \left(p + \frac{1}{2}v_j^2\right) n_i\right] \frac{\delta v_i}{\delta b_n} dS \tag{B.4}$$

The first surface integral on the r.h.s. of eq. B.4 contributes to the boundary condition of the normal to S_I component of the adjoint velocity, which reads

$$\frac{\partial j_{S_{I,i}}}{\partial p} n_i = v_i n_i$$

The second integral along S_I is equal to zero, since a Dirichlet condition is imposed on v_i and, thus, $\delta v_i / \delta b_n = 0$. Along S_O, the first integral is eliminated, since $p = 0$ and thus, $\delta p / \delta b_n = 0$. The second integral contributes to q and $v = u_i$ boundary conditions along S_O through

$$\frac{\partial j_{S_{O,j}}}{\partial v_i} n_j = v_i v_j n_j + \left(p + \frac{1}{2}v_k^2\right) n_j \delta_i^j$$

.

B.3 Solid Overheating Function, J^{penT}

J^{penT} is defined on the solid domain in an CHT problem and is expressed as

$$J^{penT} = \int_{\Omega^S} j_\Omega d\Omega \tag{B.5}$$

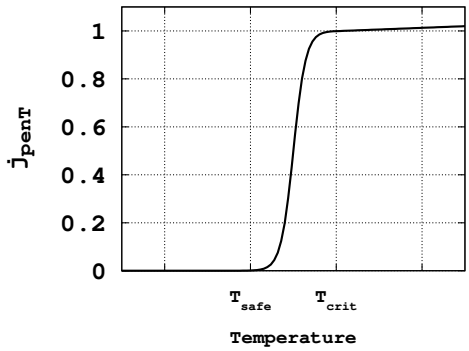


Figure B.1: Graph of eq. B.6 for $f_{min}=0.001$ and $f_{max}=0.999$.

where $j_{\Omega}^{S,T}=\frac{j^{penT}}{V_S}$, and

$$j^{penT}=\begin{cases}\left[1-\frac{1}{1+e^{k_2(T^S-T_{crit})+k_1}}\right] & ,if\ T^S\leq T_{crit}\\ \alpha\left(T^S-T_{crit}\right)+\beta & ,if\ T^S>T_{crit}\end{cases} \tag{B.6}$$

$$V^S=\int_{\Omega^S}d\Omega \tag{B.7}$$

and $k_1=log\left(\frac{1}{1-f_{max}}-1\right)$, $k_2=\frac{log\left(\frac{1}{1-f_{mim}}-1\right)-k_1}{T_{safe}-T_{crit}}$, $\alpha=\frac{k_2e^{k_1}}{(1+e^{k_1})^2}$, $\beta=\frac{e^{k_1}}{1+e^{k_1}}$ with f_{min}, f_{max} being two user-defined constants and T_{safe} and T_{crit} ($T_{safe}<T_{crit}$) two temperature thresholds. $T^S>T_{crit}$ corresponds to absolutely undesirable local overheating whereas T_{safe} pertains to the temperature below which safe operation is guaranteed. The choice of these values depends on the thermal properties of the solid body material. The goal of using eq. B.6 is to suppress areas where the temperature values exceed T_{crit}, without resorting to non-differentiable step functions. The graph of j^{penT} is illustrated in fig. B.1. The reason for which a two-branch function is chosen can be understood by looking at the contribution of the differentiated objective function T_{pen}^S to the adjoint solid heat conduction equation (last term in eq. 3.25f). If j^{penT} was given only by the first branch of eq. B.6, T^S values higher than T_{crit} would contribute to J^S without practically affecting $\partial j_{\Omega}^{S,T}/\partial T^S$. Instead, with the two-branch function, eq. B.6, the last term in eq. 3.25f takes on large values for $T^S>T_{crit}$, with the optimization process trying to reduce them. The derivative of J^{penT} w.r.t. b_n is

$$\frac{\delta J^{penT}}{\delta b_n}=\frac{1}{V^S}\left[\int_{\Omega^S}\frac{\partial j^{penT}}{\partial T^S}\frac{\delta T^S}{\delta b_n}d\Omega+\int_{\Omega^S}j^{penT}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega-J^{penT}\int_{\Omega^S}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\right] \tag{B.8}$$

where

$$\frac{\partial j^{penT}}{\partial T^S} = \begin{cases} \frac{e^{k_2 T^S} k_2}{\left(1+e^{k_2(T^S - T_{crit})+k_1}\right)^2} & , if\ T^S \leq T_{crit} \\ \alpha & , if\ T^S > T_{crit} \end{cases} \tag{B.9}$$

Following the generalized definition of objective functions presented in section 3.2.1, $j_\Omega^{S,T} = \frac{j^{penT}}{V^S}$. $j_\Omega^{S,g} = -\frac{J^{penT}}{V^S}$.

B.4 Solid Mean Temperature, J^{meanT}

In CHT problems, the mean temperature of Ω^S, J^{meanT}, is expressed as

$$J^{meanT} = \frac{\int_{\Omega^S} T^S d\Omega}{V^S} \tag{B.10}$$

The derivative of J^{meanT} w.r.t. b_n is

$$\frac{\delta J^{meanT}}{\delta b_n} = \frac{1}{V^S}\left[\int_{\Omega^S} \frac{\delta T^S}{\delta b_n} d\Omega + \int_{\Omega^S} T^S \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega - J^{meanT} \int_{\Omega^S} \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega\right] \tag{B.11}$$

Following the generalized definition of objective functions presented in section 3.2.1, $j_\Omega^{S,T} = \frac{T^S}{V^S}$. $j_\Omega^{S,g} = -\frac{J^{meanT}}{V^S}$.

B.5 FSI Heat Flux, J^{HF}

This objective function can be defined either along $\overline{S}^F$ or $\overline{S}^S$; here, the former is chosen and J^{HF} is defined as

$$J^{HF} = -\int_{\overline{S}^F} k^F \frac{\partial T^F}{\partial n} dS \tag{B.12}$$

and its derivatives w.r.t. b_n are given by

$$\frac{\delta J^{HF}}{\delta b_n} = -\int_{\overline{S}^F} \frac{\delta}{\delta b_n}\left(k^F \frac{\partial T^F}{\partial n}\right) dS - \int_{\overline{S}^F} k^F \frac{\partial T^F}{\partial n} \frac{\delta(dS)}{\delta b_n} \tag{B.13}$$

According to the generalized definition of objective functions presented in section 3.2.1, $j_{S,i}^F = -k^F \frac{\partial T^F}{\partial x_j}$ and $\partial j_S^F / \partial Q^F = -1$.

Appendix C

The Projected Direct Differentiation Method for the Primal & Adjoint Equations

The derivation of the Projected Direct Differentiation (PDD) method for the primal and adjoint equations for laminar fluid flows is presented. The developed method is of use when computing Hessian-vector products in Variant 1 for the Pareto front tracing, in Chapter 6. To compute Hessian-vector products, first the adjoint SDs are differentiated w.r.t. the design variables and, then, are multiplied with a vector $\mathbf{S}$. The arising expressions include the projected variations of the primal and adjoint fields which are computed by deriving new PDEs and boundary conditions, as presented below.

Let $\overline{\Phi}$ be defined as

$$\overline{\Phi} = \frac{\delta \Phi}{\delta b_m} s_m \tag{C.1}$$

where Φ is any field. Based on

$$\frac{\delta}{\delta b_n}\left(\frac{\partial \Phi}{\partial x_i}\right) = \frac{\partial}{\partial x_i}\left(\frac{\delta \Phi}{\delta b_n}\right) - \frac{\partial \Phi}{\partial x_k}\frac{\partial}{\partial x_i}\left(\frac{\delta x_k}{\delta b_n}\right) \tag{C.2}$$

[116], then

$$\overline{\frac{\partial \Phi}{\partial x_i}} = \frac{\delta}{\delta b_m}\left(\frac{\partial \Phi}{\partial x_i}\right) s_m = \frac{\partial}{\partial x_i}\left(\frac{\delta \Phi}{\delta b_m}\right) s_m - \frac{\partial \Phi}{\partial x_k}\frac{\partial}{\partial x_i}\left(\frac{\delta x_k}{\delta b_n}\right) s_m \tag{C.3}$$

or

$$\overline{\frac{\partial \Phi}{\partial x_i}} = \frac{\partial \overline{\Phi}}{\partial x_i} - \frac{\partial \Phi}{\partial x_k}\frac{\partial \overline{x_k}}{\partial x_i} \tag{C.4}$$

According to the *FI* adjoint, the projected second derivative of an objective function J for laminar flows is derived by differentiating eq. 2.55 w.r.t. b_m and multiplying

it with s_m, i.e.

$$\begin{aligned}\left.\frac{\delta^2 J}{\delta b_n \delta b_m}\right|_{FI} s_m = &\underbrace{\frac{\delta}{\delta b_m}\left[\int_\Omega A_{jk}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\right]s_m}_{I_1} + \underbrace{\frac{\delta}{\delta b_m}\left[\int_\Omega j_\Omega \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\right]s_m}_{I_2}\\ &+\frac{\delta}{\delta b_m}\left(\int_{S_{W_p}} j^g_{S_{i,k}}\frac{\delta x_k}{\delta b_n}n_i dS\right)s_m + \frac{\delta}{\delta b_m}\left[\int_{S_{W_p}} j_{S,i}\frac{\delta\left(n_i dS\right)}{\delta b_n}\right]s_m\\ &-\frac{\delta}{\delta b_m}\left\{\int_{S_{W_p}}\left[\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}}n_k t^I_l t^I_m \tau_{ij}\frac{\delta(t^I_i t^I_j)}{\delta b_n}\right]dS\right\}s_m\\ &-\frac{\delta}{\delta b_m}\left\{\int_{S_{W_p}}\left[\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}}n_k t^{II}_l t^{II}_m \tau_{ij}\frac{\delta(t^{II}_i t^{II}_j)}{\delta b_n}\right]dS\right\}s_m\\ &-\frac{\delta}{\delta b_m}\left\{\int_{S_{W_p}}\left[\left(-u_{\langle n\rangle}+\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}}n_k n_l n_m\right)\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n}\right]dS\right\}s_m\\ &-\frac{\delta}{\delta b_m}\left\{\int_{S_{W_p}}\left[\left(\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}}n_k(t^{II}_l t^I_m + t^I_l t^{II}_m)\right)\tau_{ij}\frac{\delta(t^{II}_i t^I_j)}{\delta b_n}\right]dS\right\}s_m \end{aligned} \tag{C.5}$$

where $A_{jk} = -u_i v_j \frac{\partial v_i}{\partial x_k} - u_j\frac{\partial p}{\partial x_k} - \tau^a_{ij}\frac{\partial v_i}{\partial x_k} + u_i\frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k}$. In the above equation, I_1 is further developed as

$$\begin{aligned} I_1 &= \int_\Omega \overline{A_{jk}}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega + \int_\Omega A_{jk}\frac{\delta}{\delta b_m}\left[\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)\right]s_m d\Omega + \int_\Omega A_{jk}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\delta(d\Omega)}{\delta b_m}s_m\\ &= \int_\Omega \overline{A_{jk}}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega + \int_\Omega A_{jk}\frac{\partial}{\partial x_j}\left(\frac{\delta^2 x_k}{\delta b_n \delta b_m}\right)s_m d\Omega - \int_\Omega A_{jk}\frac{\partial}{\partial x_\lambda}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial \overline{x_\lambda}}{\partial x_j}d\Omega\\ &+\int_\Omega A_{jk}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial \overline{x_\lambda}}{\partial x_\lambda}d\Omega \end{aligned} \tag{C.6}$$

$\overline{A_{jk}}$ is developed as

$$\begin{aligned}\overline{A_{jk}} &= \frac{\delta}{\delta b_m}\left(-u_i v_j\frac{\partial v_i}{\partial x_k} - u_j\frac{\partial p}{\partial x_k} - \tau^a_{ij}\frac{\partial v_i}{\partial x_k} + u_i\frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k}\right)s_m\\ &= -\overline{u_i}v_j\frac{\partial v_i}{\partial x_k} - u_i\overline{v_j}\frac{\partial v_i}{\partial x_k} - u_i v_j\overline{\frac{\partial v_i}{\partial x_k}} - \overline{u_j}\frac{\partial p}{\partial x_k} - u_j\overline{\frac{\partial p}{\partial x_k}} - \nu\left(\overline{\frac{\partial u_i}{\partial x_j}} + \overline{\frac{\partial u_j}{\partial x_i}}\right)\frac{\partial v_i}{\partial x_k}\\ &-\tau^a_{ij}\overline{\frac{\partial v_i}{\partial x_k}} + \overline{u_i}\frac{\partial \tau_{ij}}{\partial x_k} + u_i\overline{\frac{\partial \tau_{ij}}{\partial x_k}} + \overline{q}\frac{\partial v_j}{\partial x_k} + q\overline{\frac{\partial v_j}{\partial x_k}}\\ &= -\overline{u_i}v_j\frac{\partial v_i}{\partial x_k} - u_i\overline{v_j}\frac{\partial v_i}{\partial x_k} - u_i v_j\frac{\partial \overline{v_i}}{\partial x_k} + u_i v_j\frac{\partial v_i}{\partial x_\lambda}\frac{\partial \overline{x_\lambda}}{\partial x_k} - \overline{u_j}\frac{\partial p}{\partial x_k} - u_j\frac{\partial \overline{p}}{\partial x_k} + u_j\frac{\partial p}{\partial x_\lambda}\frac{\partial \overline{x_\lambda}}{\partial x_k}\end{aligned}$$

$$
\begin{aligned}
&-\nu\left(\frac{\partial\overline{u_i}}{\partial x_j}+\frac{\partial\overline{u_j}}{\partial x_i}-\frac{\partial u_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_j}-\frac{\partial u_j}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_i}\right)\frac{\partial v_i}{\partial x_k}-\tau_{ij}^a\left(\frac{\partial\overline{v_i}}{\partial x_k}-\frac{\partial v_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}\right)\\
&+\overline{u_i}\frac{\partial\tau_{ij}}{\partial x_k}+u_i\frac{\partial}{\partial x_k}\left[\nu\left(\overline{\frac{\partial v_i}{\partial x_j}}+\overline{\frac{\partial v_j}{\partial x_i}}\right)\right]-u_i\frac{\partial\tau_{ij}}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}+\overline{q}\frac{\partial v_j}{\partial x_k}+q\frac{\partial\overline{v_j}}{\partial x_k}-q\frac{\partial v_j}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}\\
&=-\overline{u_i}v_j\frac{\partial v_i}{\partial x_k}-u_i\overline{v_j}\frac{\partial v_i}{\partial x_k}-u_iv_j\frac{\partial\overline{v_i}}{\partial x_k}+u_iv_j\frac{\partial v_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}-\overline{u_j}\frac{\partial p}{\partial x_k}-u_j\frac{\partial\overline{p}}{\partial x_k}+u_j\frac{\partial p}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}\\
&-\nu\left(\frac{\partial\overline{u_i}}{\partial x_j}+\frac{\partial\overline{u_j}}{\partial x_i}\right)\frac{\partial v_i}{\partial x_k}+\nu\left(\frac{\partial u_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_j}+\frac{\partial u_j}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_i}\right)\frac{\partial v_i}{\partial x_k}-\tau_{ij}^a\frac{\partial\overline{v_i}}{\partial x_k}+\tau_{ij}^a\frac{\partial v_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}\\
&+\overline{u_i}\frac{\partial\tau_{ij}}{\partial x_k}+u_i\frac{\partial}{\partial x_k}\left[\nu\left(\frac{\partial\overline{v_i}}{\partial x_j}+\frac{\partial\overline{v_j}}{\partial x_i}\right)\right]-u_i\frac{\partial}{\partial x_k}\left[\nu\left(\frac{\partial v_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_j}+\frac{\partial v_j}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_i}\right)\right]\\
&-u_i\frac{\partial\tau_{ij}}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}+\overline{q}\frac{\partial v_j}{\partial x_k}+q\frac{\partial\overline{v_j}}{\partial x_k}-q\frac{\partial v_j}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}
\end{aligned}
\tag{C.7}
$$

Also, in eq. C.5, I_2 is further developed as

$$
\begin{aligned}
I_2&=\int_\Omega\overline{j_\Omega}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega+\int_\Omega j_\Omega\frac{\delta}{\delta b_m}\left[\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)\right]s_m d\Omega+\int_\Omega j_\Omega\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\delta(d\Omega)}{\delta b_m}s_m\\
&=\int_\Omega\overline{j_\Omega}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega+\int_\Omega j_\Omega\frac{\partial}{\partial x_k}\left(\frac{\delta^2 x_k}{\delta b_n\delta b_m}\right)s_m d\Omega-\int_\Omega j_\Omega\frac{\partial}{\partial x_\lambda}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial\overline{x_\lambda}}{\partial x_k}d\Omega\\
&+\int_\Omega j_\Omega\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial\overline{x_\lambda}}{\partial x_\lambda}d\Omega
\end{aligned}
\tag{C.8}
$$

By substituting eqs. C.6, eqs. C.7 and eqs. C.8 into eqs. C.5 and by applying the rule of product diffentiation to all terms except for the I_1 and I_2 integrals, the projected second derivative of J is computed as

$$
\begin{aligned}
\left.\frac{\delta^2 J}{\delta b_n\delta b_m}\right|_{FI}^{s_m}=\int_\Omega\Bigg\{&-\overline{u_i}v_j\frac{\partial v_i}{\partial x_k}-u_i\overline{v_j}\frac{\partial v_i}{\partial x_k}-u_iv_j\frac{\partial\overline{v_i}}{\partial x_k}+u_iv_j\frac{\partial v_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}-\overline{u_j}\frac{\partial p}{\partial x_k}-u_j\frac{\partial\overline{p}}{\partial x_k}\\
&+u_j\frac{\partial p}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}-\nu\left(\frac{\partial\overline{u_i}}{\partial x_j}+\frac{\partial\overline{u_j}}{\partial x_i}\right)\frac{\partial v_i}{\partial x_k}+\nu\left(\frac{\partial u_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_j}+\frac{\partial u_j}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_i}\right)\frac{\partial v_i}{\partial x_k}-\tau_{ij}^a\frac{\partial\overline{v_i}}{\partial x_k}\\
&+\tau_{ij}^a\frac{\partial v_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}+\overline{u_i}\frac{\partial\tau_{ij}}{\partial x_k}+u_i\frac{\partial}{\partial x_k}\left[\nu\left(\frac{\partial\overline{v_i}}{\partial x_j}+\frac{\partial\overline{v_j}}{\partial x_i}\right)\right]-u_i\frac{\partial}{\partial x_k}\left[\nu\left(\frac{\partial v_i}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_j}+\frac{\partial v_j}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_i}\right)\right]\\
&-u_i\frac{\partial\tau_{ij}}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}+\overline{q}\frac{\partial v_j}{\partial x_k}+q\frac{\partial\overline{v_j}}{\partial x_k}-q\frac{\partial v_j}{\partial x_\lambda}\frac{\partial\overline{x_\lambda}}{\partial x_k}\Bigg\}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega\\
&+\int_\Omega\left(-u_iv_j\frac{\partial v_i}{\partial x_k}-u_j\frac{\partial p}{\partial x_k}-\tau_{ij}^a\frac{\partial v_i}{\partial x_k}+u_i\frac{\partial\tau_{ij}}{\partial x_k}+q\frac{\partial v_j}{\partial x_k}\right)\left[\frac{\partial}{\partial x_j}\left(\frac{\delta^2 x_k}{\delta b_n\delta b_m}\right)s_m\right.\\
&\left.-\frac{\partial}{\partial x_\lambda}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial\overline{x_\lambda}}{\partial x_j}+\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial\overline{x_\lambda}}{\partial x_\lambda}\right]d\Omega+\int_\Omega\overline{j_\Omega}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega
\end{aligned}
$$

$$
\begin{aligned}
&+\int_{\Omega} j_{\Omega}\frac{\partial}{\partial x_k}\left(\frac{\delta^2 x_k}{\delta b_n \delta b_m}\right) s_m d\Omega - \int_{\Omega} j_{\Omega}\frac{\partial}{\partial x_\lambda}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial \overline{x_\lambda}}{\partial x_k} d\Omega \\
&+\int_{\Omega} j_{\Omega}\frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right)\frac{\partial \overline{x_\lambda}}{\partial x_\lambda} d\Omega + \int_{S_{W_p}}\left(\overline{j^g_{S_{i,k}}}\frac{\delta x_k}{\delta b_n} + j^g_{S_{i,k}}\frac{\delta^2 x_k}{\delta b_n \delta b_m} s_m\right) n_i dS \\
&+\int_{S_{W_p}} j^g_{S_{i,k}}\frac{\delta x_k}{\delta b_n} s_m \frac{\delta(n_i dS)}{\delta b_m} + \int_{S_{W_p}} \overline{j_{S,i}}\frac{\delta\left(n_i dS\right)}{\delta b_n} + \int_{S_{W_p}} j_{S,i} s_m \frac{\delta^2\left(n_i dS\right)}{\delta b_n \delta b_m} \\
&-\int_{S_{W_p}}\Bigg[\overline{\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}}} n_k t^I_l t^I_o \tau_{ij}\frac{\delta(t^I_i t^I_j)}{\delta b_n} + \frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}}\frac{\delta(n_k t^I_l t^I_o)}{\delta b_m} s_m \tau_{ij}\frac{\delta(t^I_i t^I_j)}{\delta b_n} \\
&+\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k t^I_l t^I_o \overline{\tau_{ij}}\frac{\delta(t^I_i t^I_j)}{\delta b_n} + \frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k t^I_l t^I_o \tau_{ij}\frac{\delta^2(t^I_i t^I_j)}{\delta b_n \delta b_m} s_m\Bigg] dS \\
&-\int_{S_{W_p}}\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k t^I_l t^I_o \tau_{ij}\frac{\delta(t^I_i t^I_j)}{\delta b_n} s_m \frac{\delta(dS)}{\delta b_m} \\
&-\int_{S_{W_p}}\Bigg[\overline{\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}}} n_k t^{II}_l t^{II}_o \tau_{ij}\frac{\delta(t^{II}_i t^{II}_j)}{\delta b_n} + \frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}}\frac{\delta(n_k t^{II}_l t^{II}_o)}{\delta b_m} s_m \tau_{ij}\frac{\delta(t^{II}_i t^{II}_j)}{\delta b_n} \\
&+\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k t^{II}_l t^{II}_o \overline{\tau_{ij}}\frac{\delta(t^{II}_i t^{II}_j)}{\delta b_n} + \frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k t^{II}_l t^{II}_o \tau_{ij}\frac{\delta^2(t^{II}_i t^{II}_j)}{\delta b_n \delta b_m} s_m\Bigg] dS \\
&-\int_{S_{W_p}}\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k t^{II}_l t^{II}_o \tau_{ij}\frac{\delta(t^{II}_i t^{II}_j)}{\delta b_n} s_m \frac{\delta(dS)}{\delta b_m} \\
&-\int_{S_{W_p}}\left[\left(-\overline{u}_{\langle n\rangle} + \overline{\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}}} n_k n_l n_o + \frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}}\frac{\delta(n_k n_l n_o)}{\delta b_m} s_m\right)\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n}\right] dS \\
&-\int_{S_{W_p}}\left[\left(-u_{\langle n\rangle} + \frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k n_l n_o\right)\left(\overline{\tau_{ij}}\frac{\delta(n_i n_j)}{\delta b_n} + \tau_{ij}\frac{\delta^2(n_i n_j)}{\delta b_n \delta b_m} s_m\right)\right] dS \\
&-\int_{S_{W_p}}\left[\left(-u_{\langle n\rangle} + \frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k n_l n_o\right)\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n}\right] s_m \frac{\delta(dS)}{\delta b_m} \\
&-\int_{S_{W_p}}\Bigg[\overline{\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}}} n_k (t^{II}_l t^I_o + t^I_l t^{II}_o)\tau_{ij}\frac{\delta(t^{II}_i t^I_j)}{\delta b_n} \\
&+\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}}\overline{n_k}(t^{II}_l t^I_o + t^I_l t^{II}_o)\tau_{ij}\frac{\delta(t^{II}_i t^I_j)}{\delta b_n} \\
&+\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k(\overline{t^{II}_l} t^I_o + t^{II}_l \overline{t^I_o} + \overline{t^I_l} t^{II}_o + t^I_l \overline{t^{II}_o})\tau_{ij}\frac{\delta(t^{II}_i t^I_j)}{\delta b_n} \\
&+\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k(t^{II}_l t^I_o + t^I_l t^{II}_o)\overline{\tau_{ij}}\frac{\delta(t^{II}_i t^I_j)}{\delta b_n} \\
&+\frac{\partial j_{S_{W_{p,k}}}}{\partial \tau_{lo}} n_k(t^{II}_l t^I_o + t^I_l t^{II}_o)\tau_{ij}\frac{\delta^2(t^{II}_i t^I_j)}{\delta b_n \delta b_m} s_m\Bigg] dS
\end{aligned}
$$

$$-\int_{S_{W_p}}\left[\frac{\partial j_{S_{W_{p,k}}}}{\partial\tau_{lo}}n_k(t_l^{II}t_o^I+t_l^I t_o^{II})\tau_{ij}\frac{\delta(t_i^{II}t_j^I)}{\delta b_n}\right]s_m\frac{\delta(dS)}{\delta b_m} \tag{C.9}$$

Eq. C.9 includes $\overline{v_i}$, $\mathbf{p}$, $\mathbf{u_i}$ and $\mathbf{q}$. To compute them, new PDEs and boundary conditions are derived, see below. Also, $\overline{x_i}$ included in eq. C.9 is computed by applying the DD method on the GDM, see section 5.1, and multiplying the arising equations with s_m.

C.1 The PDD Method for the Primal PDEs

To compute $\overline{v_i}$, $\mathbf{p}$ included in eq. C.9, eqs. 2.1a and 2.1b are differentiated w.r.t. b_m and multiplied with s_m, yielding

$$\overline{R^p}=\frac{\delta R^p}{\delta b_m}s_m=-\overline{\frac{\partial v_i}{\partial x_i}}=-\frac{\partial\overline{v_i}}{\partial x_i}+\frac{\partial v_i}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}=0 \tag{C.10}$$

and

$$\begin{aligned}\overline{R_i^v}&=\frac{\delta R_i^v}{\delta b_m}s_m=\frac{\delta}{\delta b_m}\left[\frac{\partial(v_jv_i)}{\partial x_j}\right]s_m+\frac{\delta}{\delta b_m}\left(\frac{\partial p}{\partial x_i}\right)s_m-\frac{\delta}{\delta b_m}\left(\frac{\partial\tau_{ij}}{\partial x_j}\right)s_m\\&=\frac{\partial\left(\overline{v_j}v_i+v_j\overline{v_i}\right)}{\partial x_j}-\frac{\partial\left(v_jv_i\right)}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\frac{\partial\overline{p}}{\partial x_i}-\frac{\partial p}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\\&-\frac{\partial}{\partial x_i}\left\{\nu\left[\frac{\delta}{\delta b_m}\left(\frac{\partial v_i}{\partial x_j}\right)+\frac{\delta}{\delta b_m}\left(\frac{\partial v_j}{\partial x_i}\right)\right]s_m\right\}+\frac{\partial\tau_{ij}}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}\\&=\frac{\partial\left(\overline{v_j}v_i+v_j\overline{v_i}\right)}{\partial x_j}-\frac{\partial\left(v_jv_i\right)}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\frac{\partial\overline{p}}{\partial x_i}-\frac{\partial p}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\\&-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial\overline{v_i}}{\partial x_j}+\frac{\partial\overline{v_j}}{\partial x_i}-\frac{\partial v_i}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}-\frac{\partial v_j}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\right)\right]+\frac{\partial\tau_{ij}}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}=0\end{aligned} \tag{C.11a}$$

The boundary conditions in laminar flows are either of Dirichlet or zero Neumann type. For any variable Φ, applying the PDD on the Dirichlet conditions of the form $\Phi=c$, yields

$$\overline{\Phi}=\frac{\delta\Phi}{\delta b_m}s_m=\frac{\delta c}{\delta b_m}s_m=0 \tag{C.12}$$

where c is a constant. Regarding the zero Neumann conditions $\frac{\partial\Phi}{\partial n}=0$,

$$\frac{\delta}{\delta b_m}\left(\frac{\partial\Phi}{\partial n}\right)s_m=\frac{\delta}{\delta b_m}\left(\frac{\partial\Phi}{\partial x_j}n_j\right)s_m=\frac{\delta}{\delta b_m}\left(\frac{\partial\Phi}{\partial x_j}\right)s_mn_j+\frac{\partial\Phi}{\partial x_j}\frac{\delta n_j}{\delta b_m}s_m=0 \tag{C.13}$$

Thus,

$$\frac{\partial\overline{\Phi}}{\partial n}-\frac{\partial\Phi}{\partial x_k}\frac{\partial\overline{x_k}}{\partial n}+\frac{\partial\Phi}{\partial x_j}\overline{n_j}=0 \tag{C.14}$$

In case the boundary is not parameterized, $\frac{\delta x_k}{\delta b_m}=\frac{\delta n_j}{\delta b_m}=0$ and, thus, the last two terms of eq. C.14 are zero.

C.2 The PDD Method for the Adjoint Equations

The computation of $\overline{u_i}$ and $\overline{q}$ is done by differentiating eq. 2.48a w.r.t. b_m and multiplying with s_m, i.e.

$$\overline{R^q}=\frac{\delta R^q}{\delta b_m}s_m=-\overline{\frac{\partial u_i}{\partial x_i}}+\overline{j^p_\Omega}=-\frac{\partial\overline{u_i}}{\partial x_i}+\overline{\frac{\partial u_i}{\partial x_k}\frac{\partial x_k}{\partial x_i}}+\overline{j^p_\Omega}=0 \tag{C.15}$$

Similarly, for eqs. 2.48b,

$$\begin{aligned}
\overline{R^u_i}=&\frac{\delta R^u_i}{\delta b_m}s_m=\frac{\delta}{\delta b_m}\left(u_j\frac{\partial v_j}{\partial x_i}\right)s_m-\frac{\delta}{\delta b_m}\left[\frac{\partial(v_ju_i)}{\partial x_j}\right]s_m+\frac{\delta}{\delta b_m}\left(\frac{\partial q}{\partial x_i}\right)s_m\\
&-\frac{\delta}{\delta b_m}\left(\frac{\partial\tau^a_{ij}}{\partial x_j}\right)s_m+\overline{j^v_{\Omega,i}}\\
=&\overline{u_j}\frac{\partial v_j}{\partial x_i}+u_j\frac{\delta}{\delta b_m}\left(\frac{\partial v_j}{\partial x_i}\right)s_m-\frac{\partial\left(\overline{v_j}u_i+v_j\overline{u_i}\right)}{\partial x_j}+\frac{\partial\left(v_ju_i\right)}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\frac{\partial\overline{q}}{\partial x_i}-\frac{\partial q}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\\
&-\frac{\partial}{\partial x_i}\left\{\nu\left[\frac{\delta}{\delta b_m}\left(\frac{\partial u_i}{\partial x_j}\right)+\frac{\delta}{\delta b_m}\left(\frac{\partial u_j}{\partial x_i}\right)\right]s_m\right\}+\frac{\partial\tau^a_{ij}}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\overline{j^v_{\Omega,i}}\\
=&\overline{u_j}\frac{\partial v_j}{\partial x_i}+u_j\frac{\partial\overline{v_j}}{\partial x_i}-u_j\frac{\partial v_j}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}-\frac{\partial\left(\overline{v_j}u_i+v_j\overline{u_i}\right)}{\partial x_j}+\frac{\partial\left(v_ju_i\right)}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\frac{\partial\overline{q}}{\partial x_i}-\frac{\partial q}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\\
&-\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial\overline{u_i}}{\partial x_j}+\frac{\partial\overline{u_j}}{\partial x_i}-\frac{\partial u_i}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}-\frac{\partial u_j}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_i}\right)\right]+\frac{\partial\tau^a_{ij}}{\partial x_k}\frac{\partial\overline{x_k}}{\partial x_j}+\overline{j^v_{\Omega,i}}=0
\end{aligned} \tag{C.16a}$$

Along S_I and $S_{W_{np}}$, $\frac{\delta n_j}{\delta b_m}=\frac{\delta t^I_i}{\delta b_m}=\frac{\delta t^{II}_i}{\delta b_m}=0$ and the PDD for eqs. 2.49 leads to

$$\overline{u}_{\langle n\rangle}=-\overline{\frac{\partial j_{S_{I,i}}}{\partial p}}n_i-\overline{j^p_{S_I}} \tag{C.17a}$$

$$\overline{u^l}_{\langle t\rangle}=\overline{\frac{\partial j_{S_{I,k}}}{\partial\tau_{ij}}}n_kt^l_in_j+\overline{\frac{\partial j_{S_{I,k}}}{\partial\tau_{ij}}}n_kt^l_jn_i,\ l=I,II \tag{C.17b}$$

and, due to eq. C.14, $\partial\overline{q}/\partial n=0$

Along S_{W_p},

$$\overline{u_{\langle n\rangle}} = \frac{\delta}{\delta b_m}(u_i n_i)\, s_m = -\frac{\delta}{\delta b_m}\left(\frac{\partial j_{S_{I,i}}}{\partial p} n_i\right) s_m - \frac{\delta}{\delta b_m}\left(j^p_{S_I}\right) s_m$$

$$\Rightarrow \overline{u}_{\langle n\rangle} = -u_i\overline{n_i} - \overline{\frac{\partial j_{S_{I,i}}}{\partial p}} n_i - \frac{\partial j_{S_{I,i}}}{\partial p}\overline{n_i} - \overline{j^p_{S_I}} \tag{C.18a}$$

$$\overline{u^l_{\langle t\rangle}} = \frac{\delta}{\delta b_m}\left(u_i t^l_i\right) s_m = -\frac{\delta}{\delta b_m}\left(\frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^l_i n_j\right) s_m - \frac{\delta}{\delta b_m}\left(\frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}} n_k t^l_j n_i\right) s_m$$

$$\Rightarrow \overline{u}^l_{\langle t\rangle} = -u_i\overline{t^l_i} - \overline{\frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}}} n_k t^l_i n_j - \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}}\left(\overline{n_k} t^l_i n_j + n_k\overline{t^l_i} n_j + n_k t^l_i \overline{n_j}\right)$$
$$- \overline{\frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}}} n_k t^l_j n_i - \frac{\partial j_{S_{I,k}}}{\partial \tau_{ij}}\left(\overline{n_k} t^l_j n_i + n_k\overline{t^l_j} n_i + n_k t^l_j \overline{n_i}\right),\ l = I, II \tag{C.18b}$$

and due to eq. C.14, $\frac{\partial \overline{q}}{\partial n} = \frac{\partial q}{\partial x_k}\frac{\partial \overline{x_k}}{\partial n} - \frac{\partial q}{\partial x_j}\overline{n_j}$.

Along S_O, since $\frac{\delta n_j}{\delta b_m} = \frac{\delta t^I_i}{\delta b_m} = \frac{\delta t^{II}_i}{\delta b_m} = 0$, the PDD for eq. 2.50 and 2.51 leads to

$$\overline{q} = \overline{u}_{\langle n\rangle} v_{\langle n\rangle} + u_{\langle n\rangle}\overline{v}_{\langle n\rangle} + 2\nu\frac{\partial \overline{u}_{\langle n\rangle}}{\partial n} + \overline{\frac{\partial j_{S_{O,k}}}{\partial v_i}} n_k n_i + \overline{j^v_{S_{O,i}}} n_i = 0 \tag{C.19a}$$

$$\overline{v}_{\langle n\rangle} u^l_{\langle t\rangle} + v_{\langle n\rangle}\overline{u}^l_{\langle t\rangle} + \nu\left(\frac{\partial \overline{u}^l_{\langle t\rangle}}{\partial n} + \frac{\partial \overline{u}_{\langle n\rangle}}{\partial t^l}\right) + \overline{\frac{\partial j_{S_{O,k}}}{\partial v_i}} n_k t^l_i + \overline{j^v_{S_{O,i}}} t^l_i = 0 \tag{C.19b}$$

Finally, the boundary condition on the normal component of the adjoint velocity is $\partial \overline{v}_{\langle n\rangle}/\partial n = 0$.

Over the symmetry planes S_S, applying the PDD method on eqs. 2.53 and 2.54 leads to

$$\overline{u}_{\langle n\rangle} = 0 \tag{C.20a}$$

$$\frac{\partial \overline{u}^l_{\langle t\rangle}}{\partial n} = 0 \tag{C.20b}$$

Also, $\partial \overline{q}/\partial n = 0$.

Both the PDD primal (eqs. C.15 and C.15) and adjoint (eqs. C.15 and eqs. C.15) systems are solved in this thesis by using the SIMPLE algorithm [132]. Second-order upwind schemes are used to discretize the convection terms, central schemes including a correction for non-orthogonality are used for diffusion fluxes and spatial gradients are computed through the Gauss divergence scheme, using a linear interpolation from the cell-centers to the cell-faces.

Appendix D

Computation of Hermite Polynomials Inner Products

When applying the iPCE method in problems with uncertain variables ξ following the normal distribution, Galerkin projections result into integrals of the form

$$\int_{-\infty}^{\infty} \left(\xi^k + \alpha_{k-1}\xi^{k-1} + \cdots a_0\right) w\left(\xi\right) d\xi \tag{D.1}$$

where

$$w(\xi) = \frac{1}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}} \tag{D.2}$$

is the weighting function. In eq. D.1, k is an integer and $\alpha_i,\ i = 0, ..., k-1$ is a real number multiplying ξ^i. For the mathematical development required in the iPCE method, a general rule for computing integrals of the form

$$\int_{-\infty}^{\infty} \xi^k w\left(\xi\right) d\xi = \int_{-\infty}^{\infty} \xi^k \frac{1}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}} d\xi \tag{D.3}$$

is required. Let l be an integer, then, regarding the value of k, two cases exist:

- k is odd, i.e. $k = 2l - 1$:

The integrand in eq. D.3 is an odd function and, thus, the integral is zero.

- k is even, i.e. $k = 2l$:

Since the integrand in eq. D.3 is an even function

$$\int_{-\infty}^{\infty} \xi^{2l} \frac{1}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}} d\xi = (2l-1)!! \tag{D.4}$$

where

$$l!!=\begin{cases}\prod_{k=1}^{l/2}(2l)=l\,(l-2)\cdots 2, & \text{if l is even}\\ \prod_{k=1}^{(l+1)/2}(2l-1)=l\,(l-2)\cdots 1, & \text{if l is odd}\end{cases} \tag{D.5}$$

Thus,

$$\begin{aligned}&\int_{-\infty}^{\infty} w(\xi)d\xi=1\\ &\int_{-\infty}^{\infty} \xi^2 w(\xi)d\xi=1\\ &\int_{-\infty}^{\infty} \xi^4 w(\xi)d\xi=3\\ &\int_{-\infty}^{\infty} \xi^6 w(\xi)d\xi=15\\ &\int_{-\infty}^{\infty} \xi^8 w(\xi)d\xi=105\\ &\cdots\end{aligned}$$

These rules are applied to the computation of the inner product of the multivariate Hermite polynomials. Below, the non-zero inner products of the multivariate polynomials are presented.

Double inner products:

$$\begin{aligned}\langle\psi_0(\boldsymbol{\xi}),\psi_0(\boldsymbol{\xi})\rangle_w&=1\\ \langle\psi_1(\boldsymbol{\xi}),\psi_1(\boldsymbol{\xi})\rangle_w&=1\\ \langle\psi_2(\boldsymbol{\xi}),\psi_2(\boldsymbol{\xi})\rangle_w&=1\\ \langle\psi_3(\boldsymbol{\xi}),\psi_3(\boldsymbol{\xi})\rangle_w&=1\\ \langle\psi_4(\boldsymbol{\xi}),\psi_4(\boldsymbol{\xi})\rangle_w&=2\\ \langle\psi_5(\boldsymbol{\xi}),\psi_5(\boldsymbol{\xi})\rangle_w&=2\end{aligned}$$

Indicatively, the proof of the last relationship is as follows:

$$\langle\psi_5(\boldsymbol{\xi}),\psi_5(\boldsymbol{\xi})\rangle_w=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(\xi_2-1)^2\,W(\boldsymbol{\xi})\,d\xi_1 d\xi_2=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(\xi_2^4-2\xi_2^2+1\right)W(\boldsymbol{\xi})\,d\xi_1 d\xi_2$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \xi_2^4 W(\boldsymbol{\xi})\, d\xi_1 d\xi_2 - \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} 2\xi_2^2 W(\boldsymbol{\xi})\, d\xi_1 d\xi_2 + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} W(\boldsymbol{\xi})\, d\xi_1 d\xi_2$$
$$= 4-2+1=2 \tag{D.7}$$

Triple inner products:

$$\langle \psi_0(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}) \rangle_w = 1$$
$$\langle \psi_1(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}) \rangle_w = \langle \psi_1(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}) \rangle_w = \langle \psi_0(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}) \rangle_w = 1$$
$$\langle \psi_2(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}) \rangle_w = \langle \psi_2(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}) \rangle_w = \langle \psi_0(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}) \rangle_w = 1$$
$$\langle \psi_3(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}) \rangle_w = \langle \psi_3(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}) \rangle_w = \langle \psi_0(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}) \rangle_w = 1$$
$$\langle \psi_4(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}) \rangle_w = \langle \psi_4(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi}) \rangle_w = \langle \psi_0(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi}) \rangle_w = 2$$
$$\langle \psi_5(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}) \rangle_w = \langle \psi_5(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}) \rangle_w = \langle \psi_0(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}) \rangle_w = 2$$
$$\langle \psi_5(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}) \rangle_w = 8$$
$$\langle \psi_1(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi}) \rangle_w = \langle \psi_1(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}) \rangle_w = \langle \psi_4(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}) \rangle_w = 2$$
$$\langle \psi_2(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}) \rangle_w = \langle \psi_2(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}) \rangle_w = \langle \psi_5(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}) \rangle_w = 2$$
$$\langle \psi_3(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi}) \rangle_w = \langle \psi_3(\boldsymbol{\xi}), \psi_4(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}) \rangle_w = \langle \psi_4(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}) \rangle_w = 2$$
$$\langle \psi_3(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}) \rangle_w = \langle \psi_3(\boldsymbol{\xi}), \psi_5(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}) \rangle_w = \langle \psi_5(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}) \rangle_w = 2$$
$$\langle \psi_1(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}) \rangle_w = \langle \psi_3(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}) \rangle_w = \langle \psi_2(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}) \rangle_w$$
$$= \langle \psi_3(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}) \rangle_w = \langle \psi_1(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}) \rangle_w = \langle \psi_2(\boldsymbol{\xi}), \psi_1(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}) \rangle_w = 1$$

The proof of the first six relationships is straightforward; the Galerkin projection of the non-zero double inner products on $\psi_0(\boldsymbol{\xi}) = 1$ is required. Among the rest of them, indicatively the last relationship is proven, as

$$\langle \psi_1(\boldsymbol{\xi}), \psi_2(\boldsymbol{\xi}), \psi_3(\boldsymbol{\xi}) \rangle_w = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \xi_1\xi_2\, (\xi_1\xi_2)\, W(\boldsymbol{\xi})\, d\xi_1 d\xi_2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \xi_1^2\xi_2^2 W(\boldsymbol{\xi})\, d\xi_1 d\xi_2 = 1 \tag{D.9}$$

Appendix E

Roots and Weights for Univariate Hermite Polynomials

In table E.1, the roots and the weights for univariate Hermite polynomials are tabulated, for $N_{GQ} = 1, ..., 6$, for use in the Gauss Quadrature integration method computing the PCE coefficients of the QoI, in the niPCE method.

N_{GQ}	He_n		$\widetilde{He}_n$	
	Gaussian nodes	**Weights**	**Gaussian nodes**	**Weights**
1	0	2.506628275	0	1
2	1	1.25331414	1	0.5
	−1	1.25331414	−1	0.5
3	1.732050808	0.417771379	1.732050808	0.1666666667
	0	1.671085516	0	0.6666666667
	−1.732050808	0.417771379	−1.732050808	0.1666666667
4	2.334414218	0.11499371	2.334414218	0.04587585477
	0.7419637843	1.13832042	0.7419637843	0.4541241452
	−0.7419637843	1.13832042	−0.7419637843	0.4541241452
	−2.334414218	0.11499371	−2.334414218	0.04587585477
5	2.856970014	0.028218146	2.856970014	0.01125741133
	1.35562618	0.55666179	1.35562618	0.222075922
	0	1.336868413	0	0.5333333333
	−1.35562618	0.55666179	−1.35562618	0.222075922
	−2.856970014	0.028218146	−2.856970014	0.01125741133
6	3.324257434	0.0064064014	3.324257434	0.002555784402
	1.889175878	0.22212673	1.889175878	0.08861574604
	0.6167065902	1.02478100	0.6167065902	0.4088284696
	−0.6167065902	1.02478100	−0.6167065902	0.4088284696
	−1.889175878	0.22212673	−1.889175878	0.08861574604
	−3.324257434	0.0064064014	−3.324257434	0.002555784402

Table E.1: Gaussian nodes and weights to be used in Gauss Quadrature integration method, for the univariate non-orthonormal and orthonormal Hermite polynomials.

Bibliography

[1] A. Mousavi and S.K. Nadarajah. Heat transfer optimization of gas turbine blades using an adjoint approach. In *AIAA Paper 2010-9048*, 2010.

[2] A. Mousavi and S.K. Nadarajah. Adjoint-based multidisciplinary design optimization of cooled gas turbine blades. In *AIAA Paper 2011-1131*, 2011.

[3] A.G. Fedorov and R. Viskanta. Three-dimensional conjugate heat transfer in the microchannel heat sink for electronic packaging. *International Journal of Heat and Mass Transfer*, 43:399–415, 2000.

[4] E. Alba and B. Dorronsoro. *Cellular genetic algorithms*. Springer-Verlag, 2008.

[5] E. Alba and J.M. Troya. Analyzing synchronous and asynchronous parallel distributed genetic algorithms. *Future Generation Computer Systems*, 17(4):451–465, 2001.

[6] W.K. Anderson and D.L. Bonhaus. Aerodynamic design on unstructured grids for turbulent flows. NASA Technical Memorandum 112867, 1997.

[7] W.K. Anderson and D.L. Bonhaus. Airfoil design on unstructured grids for turbulent flows. *AIAA Journal*, 37(2):185–191, 1999.

[8] W.K. Anderson and V. Venkatakrishnan. Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation. *Computers & Fluids*, 28(4-5):443–480, 1999.

[9] R. Askey and J. Wilson. *Some basic hypergeometric orthogonal polynomials that generalize Jacobi polynomials*. Memoirs AMS 319. Amer Mathematical Society, 1985.

[10] V.G. Asouti. *Aerodynamic analysis and design methods at high and low speed flows, on multiprocessor platforms*. PhD thesis, National Technical University of Athens, 2009.

[11] V.G. Asouti and K.C. Giannakoglou. Aerodynamic optimization using a parallel asynchronous evolutionary algorithm controlled by strongly interacting demes. *Engineering Optimization*, 41(3):241–257, 2009.

[12] B. Wang, W. Zhang, G. Xie, Y. Xu and M. Xiao. Multiconfiguration shape optimization of internal cooling systems of a turbine guide vane based on thermomechanical and conjugate heat transfer analysis. *Journal of Heat Transfer*, 137(6), 2015.

[13] T. Bäck. *Evolutionary algorithms in theory and practice. Evolution strategies, evolutionary programming, genetic algorithms*. Oxford University Press, 1996.

[14] D.P. Bertsekas. *Constrained optimization and Lagrange multiplier methods, 1st edition*. Athena Scientific, 1996.

[15] D.P. Bertsekas. *Nonlinear programming, 2nd edition*. Athena Scientific, 1999.

[16] B.H. Dennis, I.N. Egorov, G.S. Dulikravich and S. Yoshimura. Optimization of a large number of coolant passages located close to the surface of a turbine blade. In *ASME Paper GT2003-38051*, 2003.

[17] H. Bijl, D. Lucor, S. Mishra, and C. Schwab. *Uncertainty quantification in computational fluid dynamics*. Lecture Notes in Computational Science and Engineering. Springer, 2013.

[18] J. Boussinesq. *Theorie de l'ecoulement tourbillonnant et tumulueux des liquides and les lits rectilignes a grande section*. Gauthier-Villars et Fils, Des Comptes Rendus des Seances de L'academie des Sciences, 1897.

[19] J. Brezillon and R. Dwight. Discrete adjoint of the Navier-Stokes equations for aerodynamic shape optimization. In *Evolutionary and Deterministic Methods for Design, EUROGEN*, Munich, 2005.

[20] A. Bueno-Orovio, C. Castro, F. Palacios, and E. ZuaZua. Continuous adjoint approach for the Spalart–Allmaras model in aerodynamic optimization. *AIAA Journal*, 50(3):631–646, 2012.

[21] G.W. Burgreen and O. Baysal. Three-dimensional aerodynamic shape optimization using discrete sensitivity analysis. *AIAA Journal*, 34(9):1761–1770, 1996.

[22] I. Das and J.E. Dennis. A closer look at the drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Structural Optimization*, 14:63–69, 1997.

[23] I. Das and J.E. Dennis. Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization*, 8(3):631–657, 1998.

[24] L. Davis. *Handbook of genetic algorithms*. Van Nostrand Reinhold, 1991.

[25] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, April 2002.

[26] C. Dinescu, S. Smirnov, C. Hirsch, and C. Lacor. Assessment of intrusive and non-intrusive non-deterministic CFD methodologies based on polynomial chaos expansions. *International Journal of Engineering Systems Modelling and Simulation*, 2:87–98, 2010.

[27] R. Dwight, J. Brezillon, and D. Vollmer. Efficient algorithms for solution of the adjoint compressible Navier-Stokes equations with applications. In *Proceedings of the ONERA-DLR Aerospace Symposium (ODAS)*, Toulouse, 2006.

[28] R.P. Dwight. Robust mesh deformation using the linear elasticity equations. In Herman Deconinck and E. Dick, editors, *Computational Fluid Dynamics 2006*, pages 401–406, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[29] R.P. Dwight and J. Brezillon. Effect of approximations of the discrete adjoint on gradient–based optimization. *AIAA Journal*, 44(12):3022–3031, 2006.

[30] M.A. El-Beltagy, P.B. Nair, and A.J. Keane. Metamodeling techniques for evolutionary optimization of computationally expensive problems: Promises and limitations. In *Genetic and Evolutionary Computation Conference - GECCO 1999*. Morgan Kaufmann, San Fransisco, USA, 27-30 June 1999.

[31] M. Eldred and J. Burkardt. Comparison of non-intrusive polynomial chaos and stochastic collocation methods for uncertainty quantification. In *AIAA Paper 2009-976, 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition*, Orlando, Florida, USA, 5-8 January 2009.

[32] M. Eldred, C. Webster, and P. Constantine. Design under uncertainty employing stochastic expansion methods. In *AIAA Paper 2008-6001*, Victoria, British Columbia, Canada, 10-12 September 2008.

[33] J. Elliot and J. Peraire. Aerodynamic design using unstructured meshes. In *AIAA Paper 1996-1941, 27th Fluid Dynamics Conference*, New Orleans, LA, 17-20 June 1996.

[34] J. Elliot and J. Peraire. Practical 3D aerodynamic design and optimization using unstructured meshes. *AIAA Journal*, 35(9):1479–1485, 1997.

[35] J. Elliott and J. Peraire. Aerodynamic optimization on unstructured meshes with viscous effects. *AIAA Paper*, pages 97–1849, 1997.

[36] M. Ferlauto. An inverse method of designing the cooling passages of turbine blades based on the heat adjoint equation. *Journal of Power and Energy*, 228(3):328–339, 2014.

[37] J. Fliege and B.F. Svaiter. Steepest descent methods for multicriteria optimization. *Mathematical Methods of Operations Research*, 51:479–494, 2000.

[38] K.B. Fragkos, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. pFOSM: An efficient algorithm for aerodynamic robust design based on continuous adjoint and matrix-vector products. *Computers & Fluids*, 181:57 – 66, 2019.

[39] G. Nowak and W. Wroblewski. Cooling system optimisation of turbine guide vane. *Applied Thermal Engineering*, 29:567–572, 2009.

[40] F. Gembicki and Y. Haimes. Approach to performance and sensitivity multiobjective optimization: The goal attainment method. *IEEE Transactions on Automatic Control*, 20(6):769–771, 1975.

[41] H.A. Georgopoulou. *Optimization techniques for committing combined cycle power plants and designing their components*. PhD thesis, National Technical University of Athens, 2009.

[42] M. Ghavami Nejad, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. Aerodynamic shape optimization using the truncated Newton method and continuous adjoint. In *ECCOMAS Congress 2016, VII European Congress on Computational Methods in Applied Sciences and Engineering*, Crete island, Greece, June 5-10 2016.

[43] K.C. Giannakoglou. Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence. *Progress in Aerospace Sciences*, 38(1):43–76, 2002.

[44] M. Giles, M. Duta, and J. Muller. Adjoint code developments using the exact discrete approach. In *AIAA Paper 2001-2596, 15th Computational Fluid Dynamics Conference*, Anaheim, CA, 2001.

[45] M. Giles and N. Pierce. An introduction to the adjoint approach to design. *Flow, Turbulence and Combustion*, 65:393–415, 2000.

[46] M.B. Giles, M.C. Duta, J.D. Muller, and N.A. Pierce. Algorithm developments for discrete adjoint methods. *AIAA Journal*, 41(2), 2003.

[47] P.E. Gill, W. Murray, and M.H. Wright. *Practical optimization*. Academic Press, 1981.

[48] A.P. Giotis. *Application of evolutionary algorithms, computational intelligence and advanced computational fluid dynamics techniques to the optimization-inverse design of turbomachinery cascades, using parallel processing*. PhD thesis, National Technical University of Athens, 2003.

[49] K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. The continuous adjoint method for shape optimization in conjugate heat transfer problems with turbulent incompressible flows. *Applied Thermal Engineering*, 140:351–362, 2018.

[50] P. Grinfield. Hadamard's formula inside and out. *Journal of optimization theory and applications*, 146:654–690, 2010.

[51] H. Narten, C. Correia, C. Othmer and R. Radespiel. Adjoint-based cooling efficiency optimization of turbulent ducted flows. In *THERMACOMP2014*, Lake Bled, Slovenia, June 2-4 2014.

[52] F. Herrera, M. Lozano, and C. Moraga. Hierarchical distributed genetic algorithms. *International Journal of of Intelligent Systems*, 14(9):1099–1121, 1999.

[53] S. Hosder, R. Walters, and R. Perez. A non-intrusive polynomial chaos method for uncertainty propagation in CFD simulations. In *AIAA Paper 2006-891, 44th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, 9-12 January 2006.

[54] Z. Huan, G. Zhenghong, X. Fang, and Z. Yidian. Review of robust aerodynamic design optimization for air vehicles. *Archives of Computational Methods in Engineering*, 26:685–732, 2019.

[55] J. Iseler and T.J. Martin. Multi-disciplinary analyses for the design of a high pressure turbine blade tip. In *ASME Paper GT2016-57794*, 2016.

[56] S. Jakobsson and O. Amoignon. Mesh deformation using radial basis functions for gradient-based aerodynamic shape optimization. *Computers & Fluids*, 36:1119–1136, 2007.

[57] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3:233–260, 1988.

[58] A. Jameson. Optimum aerodynamic design using CFD and control theory. In *AIAA Paper 1995-1729, 12th Computational Fluid Dynamics Conference*, 1995.

[59] A. Jameson, J.J. Alonso, J.J. Reuther, L. Martinelli, and J.C. Vassberg. Aerodynamic shape optimization techniques based on control theory. In *AIAA Paper 1998-2358*, 1998.

[60] A. Jameson and S. Kim. Reduction of the adjoint gradient formula in the continuous limit. In *AIAA Paper 2003-0040, 41th Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, January 2003.

[61] A. Jameson, N.A. Pierce, and L. Martinelli. Optimum aerodynamic design using the Navier-Stokes equations. In *AIAA Paper 97-0101, 35th Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, 6-9 January 1997.

[62] A. Jameson and J. Reuther. Control theory based airfoil design using the Euler equations. In *AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. Panama City Beach, September 1994.

[63] H. Jasak and Ž. Tuković. Automatic mesh motion for the unstructured finite volume method. *Transactions of FAMENA*, 30(2):1–18, 2007.

[64] J.H.K. Haertel and G.F. Nelli. A fully developed flow thermofluid model for topology optimization of 3D-printed air-cooled heat exchangers. *Applied Thermal Engineering*, 119:10–24, 2017.

[65] K.L. Judd, L. Maliar, S. Maliar, and R. Valero. Smolyak method for solving dynamic economic models: Lagrange interpolation, anisotropic grid and adaptive domain. *Journal of Economic Dynamics & Control*, 44:92–123, 2014.

[66] D.H. Jung and B.C Lee. Development of a simple and efficient method for robust optimization. *International Journal for Numerical Methods in Engineering*, 53(9):2201–2215, 2002.

[67] K. Gkaragkounis and E. Papoutsis-Kiachagias and V. Asouti and K. Giannakoglou. Adjoint-based Pareto front tracing in aerodynamic shape optimization. In *ICCFD10-2018-322*, Barcelona, Spain, 9-13 July 2018.

[68] I.C. Kampolis. *Parallel, multilevel algorithms for the aerodynamic optimization in turbomachines*. PhD thesis, National Technical University of Athens, 2009.

[69] M.K. Karakasis. *Hierarchical, distributed evolutionary algorithms and computational intelligence in aerodynamic shape optimization, on multiprocessing systems*. PhD thesis, National Technical University of Athens, 2006.

[70] I.S. Kavvadias. *Continuous adjoint methods for steady and unsteady turbulent flows with emphasis on the accuracy of sensitivity derivatives*. PhD thesis, National Technical University of Athens, 2016.

[71] I.S. Kavvadias, E.M. Papoutsis-Kiachagias, G. Dimitrakopoulos, and K.C. Giannakoglou. The continuous adjoint approach to the k–ω SST turbulence

model with applications in shape optimization. *Engineering Optimization*, 47(11):1523–1542, 2015.

[72] 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. *Journal of Computational Physics*, 301:1–18, 2015.

[73] Z. Ke and J. Wang. Conjugate heat transfer simulations of pulsed film cooling on an entire turbine vane. *Applied Thermal Engineering*, 109:600–609, 2016.

[74] H. Keller. RPM: A remedy for instability. In *Collected Lectures on the Preservation of Stability under Discretization, SIAM Proceedings in Applied Mathematics 109*, pages 185–196. 2002.

[75] C. Kim, C. Kim, and O. Rho. Sensitivity analysis for the Navier-Stokes equations with two equations turbulence models. *AIAA Journal*, 39(5):838–845, 2001.

[76] C. Kim, C. Kim, and O. Rho. Effects of constant eddy viscosity assumption on gradient-based design optimization. In *AIAA Paper 2002-0262, 40th Aerospace Sciences Meeting & Exhibit*, Reno, Nevada, 2002.

[77] I.Y. Kim and O.I. de Weck. Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Structural and Multidisciplinary Optimization*, 29:149–158, 2005.

[78] K.K. Ambatipudi and M.M. Rahman. Analysis of conjugate heat transfer in microchannel heat sinks. *Numerical Heat Transfer: Part A: Applications*, 37:711–731, 2010.

[79] K. Koichi. A Fortran77 preprocessor for reverse mode automatic differentiation with recursive checkpointing. *Optimization Methods and Software*, 10(2):319–335, 1998.

[80] E.A. Kontoleontos. *Designing thermo-fluid systems using gradient-based optimization methods and evolutionary algorithms*. PhD thesis, National Technical University of Athens, 2012.

[81] D.P. Kroese, T. Taimre, and Z.I. Botev. *Handbook of Monte Carlo methods*. Wiley Series in Probability and Statistics, 2011.

[82] K.S. Ho, J.S. Liu, T. Elliott and B. Aguilar. Conjugate heat transfer analysis for gas turbine film-cooled blade. In *ASME Paper GT2016-56688*, 2016.

[83] S. Kyriacou. *Evolutionary algorithm-based design-optimization methods in turbomachinery*. PhD thesis, National Technical University of Athens, 2013.

[84] L. Mangani, M. Cerutti, M. Maritano and M, Spiel. Conjugate heat transfer analysis of NASA C3X film cooled vane with an object-oriented CFD code. In *ASME Paper GT2010-23458*, 2010.

[85] L.D. Landau and E.M. Lifshitz. *Fluid Mechanics, Volume 6 of Course of Theoretical Physics*. Pergamon Press, 1987.

[86] S.H. Lee, W. Chen, and B.M Kwak. Robust design with arbitrary distributions using Gauss-type quadrature formula. *Structural and Multidisciplinary Optimization*, 39:227–243, 2009.

[87] D. Lim, Y.S. Ong, Y. Jin, B. Sendhoff, and B.S. Lee. Efficient hierarchical parallel genetic algorithms using grid computing. *Future Generation Computer Systems*, 23(4):658–670, 2007.

[88] J.L. Lions. *Optimal control of systems governed by partial differential equations*. Springer-Verlag, New York, 1971.

[89] X. Liu, N. Qin, and H. Xia. Fast dynamic grid deformation based on Delaunay graph mapping. *Journal of Computational Physics*, 211(2):405 – 423, 2006.

[90] C. Lozano, E. Andrés, M. Martin, and P. Bitrián. Domain versus boundary computation of flow sensitivities with the continuous adjoint method for aerodynamic shape optimization problems. *Numerical Methods in Fluids*, 70(10):1305–1323, 2012.

[91] D.G. Luenberger. *Linear and nonlinear programming, 2nd edition*. Kluwer Academic Publishers, 2003.

[92] E. Luke, E. Collins, and E. Blades. A fast mesh deformation method using explicit interpolation. *Journal of Computational Physics*, 231(2):586 – 601, 2012.

[93] M. Pietropaoli, R. Ahlfeld, F. Montomoli, A. Ciani and M.D' Ercole. Design for additive manufacturing: internal channel optimization. In *ASME Paper GT2016-57318*, 2016.

[94] M. Zeinalpour and K. Mazaheri. Entropy minimization in turbine cascade using continuous adjoint formulation. *Journal of Engineering Optimization*, 48:213–{230, 2015.

[95] M. Zeinalpour, K. Mazaheri and K.C. Kiani. A coupled adjoint formulation for non-cooled and internally cooled turbine blade optimization. *Applied Thermal Engineering*, 105:327{–335, 2016.

[96] A. Margetis. *The unsteady continuous adjoint method for the aeroacoustic optimization in OpenFOAM, (in progress)*. PhD thesis, National Technical University of Athens.

[97] A.C. Marta and S. Shankaran. On the handling of turbulence equations in RANS adjoint solvers. *Computers & Fluids*, 74:102–113, 2013.

[98] D. Maruyama, D. Bailly, and G. Carrier. High-quality mesh deformation using quaternions for orthogonality preservation. *AIAA Journal*, 52(12):2712–2729, 2014.

[99] D. Mavriplis. Discrete adjoint-based approach for optimization problems on three-dimensional unstructured meshes. *AIAA Journal*, 45(4):740–750, 2007.

[100] Z. Michalewicz. *Genetic algorithms + data structures = evolution programs, 2nd edition*. Springer-Verlag, Berlin Heidelberg, 1994.

[101] J. Miranda, D. Kumar, and C. Lacor. Adjoint-based robust optimization using polynomial chaos expansions. In *ECCOMAS Congress 2016, VII European Congress on Computational Methods in Applied Sciences and Engineering*, Crete Island, Greece, June 2018.

[102] B. Mohammadi and O. Pironneau. *Applied shape optimization for fluids*. Oxford University Press, 2001.

[103] G.L. Mura, B.L. Hinchliffe, N. Qin, and J. Brezillon. Effect of non-consistent mesh movements and sensitivities on a discrete adjoint based aerodynamic optimization. In *AIAA 2017-0461, 55th AIAA Aerospace Sciences Meeting, AIAA SciTech Forum*, Grapevine, Texas, 9-13 January 2017 2017.

[104] G.L. Mura, B.L. Hinchliffe, N. Qin, and J. Brezillon. Efficient method to eliminate mesh sensitivity in adjoint-based optimization. *AIAA Journal*, 55(4):1140–1151, 2017.

[105] G.L. Mura, B.L. Hinchliffe, N. Qin, and J. Brezillon. Nonconsistent mesh movement and sensitivity calculation on adjoint aerodynamic optimization. *AIAA Journal*, 56(4):1541–1553, 2018.

[106] S. Nadarajah and A. Jameson. A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization. In *AIAA Paper 2000-0667*, 2000.

[107] S. Nadarajah and A. Jameson. Studies of the continuous and discrete adjoint approaches to viscous automatic aerodynamic shape optimization. In *AIAA Paper 2001-2530, 15th Computational Fluid Dynamics Conference*, Anaheim, CA, 2001.

[108] N. Nemec and D. Zingg. Towards efficient aerodynamic shape optimization based on the Navier-Stokes equations. In *AIAA Paper 2001-2532, 15th Computational Fluid Dynamics Conference*, Anaheim, CA, 2001.

[109] J.C. Newman, W.K. Anderson, and D.L. Whitfield. Multidisciplinary sensitivity derivatives using complex variables. *MSSU-COE-ERC-98-08*, 1998.

[110] E. Nielsen and W. Anderson. Aerodynamic design optimization on unstructured meshes using the Navier-Stokes equations. *AIAA Journal*, 37(11):185–191, 1999.

[111] E. Nielsen and M. Park. Using an adjoint approach to eliminate mesh sensitivities in aerodynamic design. *AIAA Journal*, 44(5):948–953, 2006.

[112] E.J. Nielsen and W.L. Kleb. Efficient construction of discrete adjoint operators on unstructured grids by using complex variables. In *AIAA Paper 2005-0324, 43rd AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, 2005.

[113] J. Nocedal and S.J. Wright. *Numerical optimization*. Springer, New York, 1999.

[114] G. Onorato, G. Loeven, G.J.A. Ghorbaniasl, and C. Lacor. Comparison of intrusive and non-intrusive polynomial chaos methods for CFD applications in aeronautics. In *ECCOMAS CFD 2010, V European Congress on Computational Methods in Applied Sciences and Engineering*, Lisbon, Portugal, June 2010.

[115] G. Pampalis. *Non-Intrusive Polynomial Chaos Expansion in Aerodynamic Robust Design*. Diploma Thesis, PCOpt/NTUA, in greek, 2015.

[116] D. Papadimitriou and K. Giannakoglou. A continuous adjoint method for the minimization of losses in cascade viscous flows. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, 9-12 January 2012.

[117] D.I. Papadimitriou. *Adjoint formulations for the analysis and design of turbomachinery cascades and optimal grid adaptation using a posteriori error analysis*. PhD thesis, National Technical University of Athens, 2007.

[118] D.I. Papadimitriou and K.C. Giannakoglou. A continuous adjoint method with objective function derivatives based on boundary integrals for inviscid and viscous flows. *Journal of Computers & Fluids*, 36(2):325–341, 2007.

[119] D.I. Papadimitriou and K.C. Giannakoglou. A continuous adjoint method with objective function derivatives based on boundary integrals for inviscid and viscous flows. *Computers & Fluids*, 36(2):325–341, 2007.

[120] D.I. Papadimitriou and K.C. Giannakoglou. Direct, adjoint and mixed approaches for the computation of Hessian in airfoil design problems. *International Journal for Numerical Methods in Fluids*, 56(10):1929–1943, 2007.

[121] D.I. Papadimitriou and K.C. Giannakoglou. Direct, adjoint and mixed approaches for the computation of Hessian in airfoil design problems. *International Journal for Numerical Methods in Fluids*, 56(10):1929–1943, 2008.

[122] D.I. Papadimitriou and K.C. Giannakoglou. Aerodynamic design using the truncated Newton algorithm and the continuous adjoint approach. *International Journal for Numerical Methods in Fluids*, 68(6):724–739, 2012.

[123] D.I. Papadimitriou and K.C. Giannakoglou. Third-order sensitivity analysis for robust aerodynamic design using continuous adjoint. *International Journal for Numerical Methods in Fluids*, 71(5):652–670, 2013.

[124] A.K. Papageorgiou, K.B. Fragkos, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. Uncertainty quantification and robust design for aerodynamic applications, using continuous adjoint methods. In *6th European Conference on Computational Mechanics (ECCM 6) - 7th European Conference on Computational Fluid Dynamics (ECFD 7)*, Glasgow, UK, June 11-15 2018.

[125] E.M. Papoutsis-Kiachagias. *Adjoint Methods for Turbulent Flows, Applied to Shape or Topology Optimization and Robust Design*. PhD thesis, National Technical University of Athens, 2013.

[126] E.M. Papoutsis-Kiachagias, Papadimitriou D.I, and K.C. Giannakoglou. Robust design in aerodynamics using 3rd-order sensitivity analysis based on discrete adjoint. application to Quasi-1D flows. *International Journal for Numerical Methods in Fluids*, 69(3):691–709, 2012.

[127] E.M. Papoutsis-Kiachagias and K.C. Giannakoglou. Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: Industrial applications. *Archives of Computational Methods in Engineering*, 23(2):255–299, 2016.

[128] E.M. Papoutsis-Kiachagias, S.A. Kyriacou, and K.C. Giannakoglou. The continuous adjoint method for the design of hydraulic turbomachines. *Computer Methods in Applied Mechanics and Engineering*, 278:621–639, 2014.

[129] E.M. Papoutsis-Kiachagias, N. Magoulas, J. Mueller, C. Othmer, and K.C. Giannakoglou. Noise reduction in car aerodynamics using a surrogate objective function and the continuous adjoint method with wall functions. *Computers & Fluids*, 122:223–232, 2015.

[130] E.M. Papoutsis-Kiachagias, A.S. Zymaris, I.S. Kavvadias, D.I. Papadimitriou, and K.C. Giannakoglou. The continuous adjoint approach to the k–ϵ turbulence model for shape optimization and optimal active control of turbulent flows. *Engineering Optimization*, 47(3):370–389, 2015.

[131] E.M. Papoutsis-Kiachagias, A.S. Zymaris, I.S. Kavvadias, D.I. Papadimitriou, and K.C. Giannakoglou. The continuous adjoint approach to the k–ε turbulence model for shape optimization and optimal active control of turbulent flows. *Engineering Optimization*, 47(3):370–389, 2015.

[132] S.V. Patankar and D.B. Spalding. Calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal of Heat and Mass Transfer*, 15:1787–1806, 1972.

[133] J. Periaux and G. Winter. *Genetic algorithms in engineering and computer science*. John Wiley & Sons, 1995.

[134] J. Peter and R.P. Dwight. Numerical sensitivity analysis for aerodynamic optimization: A survey of approaches. *Computers & Fluids*, 39(3):373–391, 2010.

[135] J. Peter and J. Mayeur. Improving accuracy and robustness of a discrete direct differentiation method and discrete adjoint method for aerodynamic shape optimization. In *Proceedings of ECCOMAS*, Egmond ann Zee, 2006.

[136] O. Pironneau. *Optimal shape design for elliptic systems*. Springer-Verlag, New York, 1984.

[137] D. Poljak. Stochastic collocation applications in computational electromagnetics. *Mathematical Problems in Engineering, Article ID 1917439*, 2018.

[138] M.M. Putko, A.C. III Taylor, P.A. Newman, and L.L. Green. Approach for input uncertainty propagation and robust design in CFD using sensitivity derivatives. *Journal of Fluids Engineering*, 124(1):60–69, 11 2001.

[139] J. Reuther, J.J. Alonso, M.J. Rimlinger, and A. Jameson. Aerodynamic shape optimization of supersonic aircraft configurations via an adjoint formulation on distributed memory parallel computers. In *AIAA, NASA and ISSMO, Symposium on Multidisciplinary Analysis and Optimization*, 6th, Bellevue, Wa, 4-6 September 1996.

[140] J. Reuther and A. Jameson. Control theory based airfoil design for potential flow and a finite volume discretization. In *AIAA Paper 91-499, 32th Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, June 1994.

[141] J. Reuther and A. Jameson. Aerodynamic shape optimization of wing and wing-body configurations using control theory. In *AIAA Paper 95-0213, 33rd Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, June 1995.

[142] J. Reuther, A. Jameson, J. Farmer, L. Martinelli, and D. Saunders. Aerodynamic shape optimization of complex aircraft configurations via an adjoint formulation. In *AIAA Paper 1996-0094, 34th Aerospace Sciences Meeting*, Reno, Nevada, 1996.

[143] J. Reuther, A. Jameson, J. Reuther, and A. Jamesont. Control theory based airfoil design using the euler equations. In *AIAA paper 94-4272, 5th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Panama City Beach, FL*, 1994.

[144] Y. Saad and M.H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.

[145] J.A. Samareh. Application of quaternions for mesh deformation. *NASA/TM-2002-211646*, April 2002.

[146] K. Samouchos. *Adjoint methods for aerodynamic optimization problems solved with the Cut-Cell technique (in progress)*. PhD thesis, National Technical University of Athens.

[147] C. Schillings, S. Schmidt, and V. Schulz. Efficient shape optimization for certain and uncertain aerodynamic design. *Computers & Fluids*, 46:78–87, 2011.

[148] S. Schmidt and V. Schulz. Pareto-curve continuation in multi-objective optimization. *Pacific Journal of Optimization*, 4(2):243–257, 2008.

[149] G. Schroff and H. Keller. Stabilization of unstable procedures: The recursive projection method. *SIAM Journal of Numerical Analysis*, 30(4):1099–1120, 1993.

[150] A.A. Schy and D.P Giesy. Multiobjective insensitive design of airplane control systems with uncertain parameters. In *AIAA Paper 81-1818, Guidance and Control Conference*, Albuquerque, NM, USA, 1981.

[151] A.A. Schy and D.P Giesy. Tradeoff studies in multiobjective insensitive design of airplane control systems. In *AIAA Paper 83-2273, Guidance and Control Conference*, Gatlinburg, TN, USA, 1983.

[152] S. Shankaran and B. Barr. Efficient gradient-based algorithms for the construction of Pareto fronts. In *ASME Paper GT2011-45069*, Vancouver, Canada, 6-10 June 2011.

[153] S. Shankaran and A. Jameson. Robust optimal control using polynomial chaos and adjoints for systems with uncertain inputs. In *20th AIAA Computational Fluid Dynamics Conference*, Honolulu, Hawaii, 27-30 June 2011.

[154] K. Svanberg. The method of moving asymptotes|a new method for structural optimization. *International Journal for Numerical Methods in Engineering*, 24(2):359–373, 1987.

[155] T. Van Oevelen, M. Baelmans. Application of topology optimization in a conjugate heat transfer problem. In *Proceedings of Engineering and Applied Sciences Optimization*, Kos Island, Greece, June 4-6 2014.

[156] T.C. Hung, W.-M. Yan and W.P. Li. Analysis of heat transfer characteristics of double-layered microchannel heat sink. *International Journal of Heat and Mass Transfer*, 55:3090–3099, 2012.

[157] D. Thévenin and G. Janiga. *Optimization and computational fluid dynamics*. Springer, 2008.

[158] K. Tsiakas. *Development of optimization methods for use on Graphics Processing Units, with turbomachinery applications*. PhD thesis, National Technical University of Athens, 2019.

[159] P.G. Tucker. Differential equation-based wall distance computation for DES and RANS. *Journal of Computational Physics*, 190:229–248, 2003.

[160] I. Vasilopoulos, V.G. Asouti, K.C. Giannakoglou, and M. Meyer. Gradient-based Pareto front approximation applied to turbomachinery shape optimization. *Engineering with Computers*, 2019.

[161] I. Vassilopoulos. *CAD-based and CAD-free aerodynamic optimization of geometrically complex turbomachienry components*. PhD thesis.

[162] C. Vezyris, E. Papoutsis-Kiachagias, and K. Giannakoglou. On the incremental singular value decomposition method to support unsteady adjoint-based optimization. *International Journal for Numerical Methods in Fluids*, 91(7):315–331, 2019.

[163] G. Wang, L. Xu, Ch. Li, and Z. Ye. Influence of mesh sensitivities on computational-fluid-dynamics-based derivative calculation. *AIAA Journal*, 54(12):3717–3726, 2016.

[164] J. Witteveen and H. Bijl. Explicit mesh deformation using inverse distance weighting interpolation. In *19th AIAA Computational Fluid Dynamics*, San Antonio, Texas, 22-25 June 2009.

[165] X. Qian and E.M. Dede. Topology optimization of a coupled thermal-fluid system under a tangential thermal gradient constraint. *Structral and Multidisciplinary Optimization*, 54:531{–551, 2016.

[166] D. Xiu and J.S. Hesthaven. High-order collocation methods for differential equations with random inputs. *SIAM Journal on Scientific Computing*, 27(3):1118–1139, 2005.

[167] D. Xiu and G.E. Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24:619–644, 2006.

[168] L. Yang and S. Nadarajah. Data Compression Algorithms for Adjoint Based Sensitivity Studies of Unsteady Flows. volume Volume 2: Development and Applications in Computational Fluid Dynamics; Industrial and Environmental Applications of Fluid Mechanics; Fluid Measurement and Instrumentation; Cavitation and Phase Change of *Fluids Engineering Division Summer Meeting*, 07 2018.

[169] W. Yao, X. Chen, W. Luo, M. van Tooren, and J. Guo. Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles. *Progress in Aerospace Sciences*, 47(6):450 – 479, 2011.

[170] T. Zervogiannis. *Optimization methods in aerodynamics and turbomachinery based on the adjoint technique, hybrid grids and the exact Hessian matrix*. PhD thesis, National Technical University of Athens, 2011.

[171] P. Zhang, J. Lu, Z. Wang, L. Song, and Z. Feng. Adjoint-based optimization method with linearized SST turbulence model and a frozen gamma-theta transition model approach for turbomachinery design. In *Proceedings of ASME Turbo Expo 2015: Turbine Technical Conference and Exposition,*, GT2015-42582, page V02BT39A019, Montreal, Canada, 2015.

[172] H. Zhao, Z. Gao, Y. Gao, and C. Wang. Effective robust design of high lift NLF airfoil under multi-parameter uncertainty. *Aerospace Science and Technology*, 68:530–542, 2017.

[173] D.W. Zingg, M. Nemec, and T.H. Pulliam. A comparative evaluation of genetic and gradient- based algorithms applied to aerodynamic optimization. *European Journal of Computational Mechanics*, 17:103–126, 2012.

[174] A.S. Zymaris. *Adjoint methods for the design of shapes with optimal aerodynamic performance in laminar and turbulent flows*. PhD thesis, National Technical University of Athens, 2010.

[175] A.S. Zymaris, D.I. Papadimitriou, K.C. Giannakoglou, and C. Othmer. Continuous adjoint approach to the Spalart-Allmaras turbulence model for incompressible flows. *Computers & Fluids*, 38(8):1528–1538, 2009.

[176] A.S. Zymaris, D.I. Papadimitriou, K.C. Giannakoglou, and C. Othmer. Adjoint wall functions: A new concept for use in aerodynamic shape optimization. *Journal of Computational Physics*, 229(13):5228–5245, 2010.

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Μηχανολόγων Μηχανικών
Εργαστήριο Θερμικών Στροβιλομηχανών
Μονάδα Παράλληλης Υπολογιστικής Ρευστοδυναμικής & Βελτιστοποίησης

Η Συνεχής Συζυγής Μέθοδος για τη Βελτιστοποίηση Μορφής στην Αεροδυναμική και τη Συζευγμένη Μεταφορά Θερμότητας, με Τυρβώδεις Ροές

Διδακτορική Διατριβή

Κωνσταντίνος Θ. Γκαραγκούνης

Επιβλέπων: Κυριάκος Χ. Γιαννάκογλου
Καθηγητής ΕΜΠ

Αθήνα, 2020

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Μηχανολόγων Μηχανικών
Εργαστήριο Θερμικών Στροβιλομηχανών
Μονάδα Παράλληλης Υπολογιστικής Ρευστοδυναμικής & Βελτιστοποίησης

Η Συνεχής Συζυγής Μέθοδος για τη Βελτιστοποίηση Μορφής στην Αεροδυναμική και τη Συζευγμένη Μεταφορά Θερμότητας, με Τυρβώδεις Ροές

Διδακτορική Διατριβή

Κωνσταντίνος Θ. Γκαραγκούνης

Εξεταστική Επιτροπή:

1. Κ.Χ. Γιαννάκογλου (Επιβλέπων)*
   Καθηγητής ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών
2. Σ. Βουτσινάς*
   Καθηγητής ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών
3. Κ. Μαθιουδάκης*
   Καθηγητής ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών
4. Ι. Αναγνωστόπουλος
   Καθηγητής ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών
5. Λ. Καικτσής
   Καθηγητής ΕΜΠ, Σχολή Ναυπηγών Μηχανολόγων Μηχανικών
6. Κ. Μπελιμπασάκης
   Καθηγητής ΕΜΠ, Σχολή Ναυπηγών Μηχανολόγων Μηχανικών
7. Β. Ριζιώτης
   Επίκουρος Καθηγητής ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών

* Μέλος της Συμβουλευτικής Επιτροπής.

Αθήνα, 2020

Περίληψη

Η διδακτορική διατριβή στοχεύει στην ανάπτυξη, προγραμματισμό και πιστοποίηση μεθόδων βελτιστοποίησης μορφής (shape optimization) οι οποίες χαρακτηρίζονται από υψηλή ακρίβεια και χαμηλό υπολογιστικό κόστος και στηρίζονται στη συνεχή συζυγή μέθοδο (continuous adjoint method). Οι μέθοδοι που αναπτύσσονται αντιμετωπίζουν προβλήματα είτε (μόνο) ροής ρευστού είτε συζευγμένης μεταφοράς θερμότητας (Conjugate Heat Transfer, CHT). Επιπλέον, διερευνώνται μέθοδοι υπολογισμού του μετώπου μη-κυριαρχούμενων λύσεων (Pareto) σε προβλήματα πολυκριτηριακής βελτιστοποίησης, οι οποίες βασίζονται στην κλίση των συναρτήσεων-στόχου. Τέλος, αναπτύσσεται η επεμβατική μέθοδος αναπτύγματος πολυωνυμικού χάους (intrusive Polynomial Chaos Expansion, iPCE) για προβλήματα βελτιστοποίησης CHT με αβεβαιότητες. Προκειμένου να διαμορφωθούν οι προαναφερθείσες μέθοδοι, επεκτάθηκαν και εμπλουτίστηκαν συζυγείς μέθοδοι που είχαν παρουσιαστεί σε προηγούμενες διδακτορικές διατριβές στην ΜΠΥΡ&Β ΕΜΠ [50, 40, 27] και είχαν επιτυχώς εφαρμοστεί σε προβλήματα αεροδυναμικής και υδροδυναμικής βελτιστοποίησης μορφής. Η διδακτορική διατριβή κινείται σε τέσσερις άξονες:

Ο πρώτος άξονας της διατριβής αφορά στη διαχείριση των παραγώγων ευαισθησίας πλέγματος, δηλαδή των παραγώγων της θέσης των κόμβων του πλέγματος ως προς τις μεταβλητές σχεδιασμού, στη συνεχή συζυγή μέθοδο (Κεφάλαιο 5). Μέχρι τώρα, οι μετατοπίσεις των εσωτερικών κόμβων του πλέγματος λαμβανόταν υπόψη κατά στην κλίση της συνάρτησης-στόχου, στις οποίες εμφανίζονται οι παράγωγοι ευαισθησίας πλέγματος μέσα σε Χωρικά Ολοκληρώματα (Field Integrals - συζυγής διατύπωση *FI*). Ο υπολογισμός των παραγώγων ευαισθησίας πλέγματος στο εσωτερικό του χωρίου έχει κόστος ίσο με τη μετατόπιση του πλέγματος τόσες φορές όσες είναι και οι μεταβλητές σχεδιασμού. Αποφεύγεται δε διατυπώνοντας το συζυγές πρόβλημα του μοντέλου μετατόπισης πλέγματος [27]. Έτσι προκύπτουν παράγωγοι που αποτελούνται μόνο από επιφανειακά ολοκληρώματα και η μέθοδος ονομάζεται συζυγής διατύπωση *E-SI* (Enhanced Surface Integrals). Με τον όρο 'Enhanced' διαφοροποιείται από προγενέστερές της στις οποίες η μετατόπιση των εσωτερικών κόμβων του πλέγματος δεν λαμβάνεται υπόψη (Severed *SI*). Η ανάπτυξη της μεθόδου *E-SI* βασίστηκε στην υπόθεση ότι τη μετατόπιση των κόμβων του πλέγματος διέπουν εξισώσεις Laplace. Εδώ εξετάζεται, εάν αυτή η παραδοχή είναι ασφαλής ακόμα και όταν χρησιμοποιούνται διαφορετικά μοντέλα μετατόπισης πλέγματος. Συγκρίνονται παράγωγοι που υπολογίζονται με βάση άλλα, ευρέως διαδεδομένα, μοντέλα μετατόπισης (ογκομετρικές B-Splines, Γράφοι Delaunay, μέθοδος Inverse Distance Weighting), δείχνοντας ότι το είδος του μοντέλου μετατόπισης δεν επιδρά στις τιμές των παραγώγων. Συνεπώς, δεν απαιτείται η ανάπτυξη της συνεχούς συζυγούς διατύπωσης *E-SI* εξαρχής, κάθε φορά που αλλάζει το μοντέλο μετατόπισης πλέγματος το οποίο υποστηρίζει μεθόδους βελτιστοποίησης μορφής.

Στον δεύτερο άξονα της διατριβής επεκτείνονται οι συζυγείς διατυπώσεις *E-SI* και *FI* (παρουσιάζονται στο Κεφάλαιο 2) που είχαν αναπτυχθεί σε προηγούμενες διδακτορικές διατριβές στην ΜΠΥΡ&Β ΕΜΠ για προβλήματα μόνο ροής, σε προβλήματα

CHT (Κεφάλαιο 3). Ο υπολογισμός παραγώγων με ακρίβεια εξασφαλίζεται μέσω της διαφόρισης του μοντέλου τύρβης, που στη διατριβή αυτή είναι το Spalart–Allmaras. Στα προβλήματα CHT, εκτός από τις εξισώσεις συνέχειας, ορμής και μοντέλου τύρβης, επιλύονται η ενεργειακή εξίσωση στο ρευστό και η εξίσωση αγωγής της θερμότητας στο στερεό, λαμβάνοντας υπόψη και τη συναλλαγή θερμότητας μέσω της διεπιφάνειας ρευστού-στερεού. Επιβεβαιώθηκε ότι, και σε προβλήματα CHT μειώνεται η ακρίβεια των παραγώγων, όταν υιοθετείται η παραδοχή της 'παγωμένης τύρβης', με την οποία θεωρείται αμελητέα η εξάρτηση της τυρβώδους συνεκτικότητας από τις μεταβλητές σχεδιασμού. Επιπλέον, επεκτείνεται η κατάλληλη διαχείριση των παραγώγων ευαισθησίας πλέγματος σε προβλήματα CHT διατυπώνοντας συζυγείς εξισώσεις μετατόπισης για το στερεό πέρα από το ρευστό. Συγκρίσεις ανάμεσα στις μεθόδους *FI* και *E-SI* επιβεβαιώνουν ότι και σε προβλήματα CHT η δεύτερη μέθοδος είναι οικονομικότερη. Επίσης, ο υπολογισμός παραγώγων μέσω της διατύπωσης (Severed) *SI* συχνά βλάπτει την ακρίβεια των παραγώγων. Τέλος, και σε προβλήματα CHT είναι αμελητέα η επίδραση της επιλογής του μοντέλου μετατόπισης πλέγματος στις παραγώγους.

Στον τρίτο άξονα της διατριβής, παραλλαγές ενός αλγορίθμου πρόβλεψης-διόρθωσης οι οποίες βασίζονται στη συζυγή μέθοδο, υπολογίζουν το μέτωπο Pareto για προβλήματα πολυκριτηριακής βελτιστοποίησης (Κεφάλαιο 6) . Οι παραλλαγές του αλγορίθμου πρόβλεψης-διόρθωσης αρχικοποιούνται εντοπίζοντας ένα σημείο του μετώπου Pareto μέσω βελτιστοποίησης για μία μόνο από τις αντικρουόμενες συναρτήσεις-στόχους. Στα συστήματα, για την ανανέωση των μεταβλητών σχεδιασμού, στα βήματα πρόβλεψης και διόρθωσης, αποφεύγεται ο ακριβός υπολογισμός του ακριβούς Εσσιανού μητρώου των συναρτήσεων-στόχου, είτε με: (α) τον υπολογισμό γινομένων Εσσιανού-διανυσμάτων, που χρησιμοποιούνται σε έναν αλγόριθμο επίλυσης εξισώσεων τύπου Krylov είτε με (β) την προσέγγιση του Εσσιανού μητρώου μέσω της μεθόδου BFGS. Προκειμένου να υπολογιστούν οικονομικά τα γινόμενα Εσσιανού-διανυσμάτων, διατυπώνονται νέες εξισώσεις μέσω της Ευθείας Διαφόρισης των πρωτευουσών και συζυγών εξισώσεων. Συγκρίνοντας τις δύο εναλλακτικές, αυτή που προσεγγίζει το Εσσιανό μητρώο στο βήμα πρόβλεψης έχει το χαμηλότερο κόστος ενώ δείχνεται ότι η παράλειψη του βήματος πρόβλεψης οδηγεί στην αύξηση του υπολογιστικού κόστους. Περαιτέρω συγκρίσεις κόστους μεταξύ του πιο οικονομικού αλγορίθμου πρόβλεψης-διόρθωσης και μεθόδων βελτιστοποίησης με χρήση βαρών δείχνουν την υπεροχή της πρώτης μεθόδου.

Ο τελευταίος άξονας αφορά στη βελτιστοποίηση μορφής σε προβλήματα CHT με αβεβαιότητες, με τη μέθοδο iPCE (Κεφάλαιο 7). Εξετάζονται προβλήματα με δύο αβέβαιες μεταβλητές που ακολουθούν κανονική κατανομή. Οι μεταβλητές στις πρωτεύουσες και συζυγείς εξισώσεις χωρίς αβεβαιότητες και οι αντίστοιχες οριακές συνθήκες αναλύονται σε γρα$\mu\mu$ικούς συνδυασμούς ορθογωνίων πολυωνύμων για τάξη χάους ίση με δύο. Προκύπτουν νέα συστήματα εξισώσεων, που επιλύονται για να υπολογιστούν οι στατιστικές ροπές και οι παράγωγοί τους. Η ακρίβεια των στατιστικών ροπών, που προσεγγίζονται με τη μέθοδο iPCE, πιστοποιείται χρησιμοποιώντας πιο

κοστοβόρες μεθόδους, τη μη-επεμβατική μέθοδο PCE (non-intrusive PCE-niPCE) και τη Monte Carlo. Επιπλέον, λαμβάνουν χώρα βελτιστοποιήσεις μορφής με αβεβαιότητες στις συνθήκες εισόδου του ρευστού και το πάχος θερμικής μόνωσης στη διεπιφάνεια ρευστού-στερεού.

Οι προαναφερθείσες μέθοδοι πιστοποιούνται αρχικά σε 2Δ προβλήματα μόνο ροής, για ροές γύρω από αεροτομές και μέσα σε αγωγούς σχήματος 'S' και σε προβλήματα CHT με αγωγό σχήματος 'S' σε επαφή με στερεό. Σε προβλήματα ροής γύρω από αεροτομές υπολογίζονται μέτωπα Pareto για αντικρουόμενους στόχους την άνωση και την αεροδυναμική αντίσταση. Σε πρόβλημα CHT για ένα εσωτερικά ψυχόμενα 2Δ πτερύγια σταθερής πτερύγωσης στροβίλου, ελαχιστοποιείται η μέση θερμοκρασία του, μεταβάλλοντας το σχήμα του και τις θέσεις των οπών εσωτερικής ψύξης. Επίσης, βελτιστοποιείται 2Δ αγωγός σε επαφή με στερεό, για ελάχιστες απώλειες ολικής πίεσης του ρευστού, επιβάλλοντας περιορισμούς που αποτρέπουν την υπερθέρμανση του στερεού. Επιπλέον, πραγματοποιούνται βελτιστοποιήσεις σε αγωγό ψύξης μέσα σε στερεό, θέτοντας ως στόχους τη μεγιστοποίηση της θερμορροής από το στερεό προς το ψυκτικό και την ελαχιστοποίηση του ποσοστού του όγκου του στερεού με υψηλές θερμοκρασίες. Επίσης, βελτιστοποιείται ο αγωγός ψύξης κυλινδροκεφαλής σε μηχανή εσωτερικής καύσης ενός αυτοκινήτου, με δύο διαφορετικούς στόχους: (α) ελάχιστες μέγιστες θερμοκρασίες της κυλινδροκεφαλής και (β) ελάχιστες απώλειες ολικής πίεσης στο ψυκτικό. Τέλος, υπολογίζεται μέτωπο Pareto σε πτερύγια ψύξης, με τις δύο προαναφερθείσες συναρτήσεις ως τους δύο στόχους.

Οι προαναφερθείσες μέθοδοι προγραμματίστηκαν στο περιβάλλον ανοιχτού κώδικα OpenFOAM©, το οποίο παρέχει υποδομή που βασίζεται στην κεντροκυψελική διατύπωση πεπερασμένων όγκων.

Η ερευνητική εργασία υποστηρίχτηκε από το Ελληνικό Ίδρυμα Έρευνας και Καινοτομίας (ΕΛ.ΙΔ.Ε.Κ.) και από τη Γενική Γραμματεία Έρευνας και Τεχνολογίας (ΓΓΕΤ), στο πλαίσιο της Δράσης «Υποτροφίες ΕΛΙΔΕΚ Υποψηφίων Διδακτόρων» (αρ.Σύμβασης 1796).

Λέξεις κλειδιά: Συζευγμένη Μεταφορά Θερμότητας, Αεροδυναμική Βελτιστοποίηση Μορφής, Συνεχής Συζυγής Μέθοδος, Συζυγές Μοντέλο Τύρβης, Παράγωγοι Ευαισθησίας Πλέγματος, Μοντέλο Μετατόπισης Πλέγματος, Μέτωπα Pareto, Ποσοτικοποίηση Αβεβαιοτήτων, Βελτιστοποίηση υπό Αβεβαιότητες, Επεμβατικό Ανάπτυγμα Πολυωνυμικού Χάους

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγή

Η ραγδαία αύξηση της υπολογιστικής ισχύος και η ανάπτυξη υπολογιστικών μεθόδων οδήγησαν στην ευρεία χρήση της υπολογιστικής ρευστοδυναμικής (Computational Fluid Dynamics, CFD) για προλέξεις ροών σε προβλήματα αερο- και υδροδυναμικής. Επόμενο της διαθεσιμότητας λογισμικών CFD ήταν η δημιουργία μεθόδων βελτιστοποίησης στη ρευστοδυναμική (CFD-based optimization). Αυτή η διδακτορική διατριβή συνεισφέρει σε αυτό πεδίο, επεκτείνοντας τις μεθόδους και σε προβλήματα Συζευγμένης Μεταφοράς Θερμότητας (Conjugate Heat Transfer - CHT).

Στη βελτιστοποίηση μορφής οι γεωμετρίες προς βελτιστοποίηση ελέγχονται από τις μεταβλητές σχεδιασμού $b_n,\ n = 1, ..., N$. Η συσχέτιση των κόμβων της γεωμετρίας με τις μεταβλητές σχεδιασμού ονομάζεται παραμετροποίηση. Η ροϊκή επίδοση της γεωμετρίας προς βελτιστοποίηση ποσοτικοποιείται από τις συναρτήσεις-στόχους J (objective functions), που υπολογίζονται μέσω της επίλυσης των εξισώσεων που διέπουν το φυσικό πρόβλημα ('πρωτεύουσες εξισώσεις/primal equations'). Όλα τα προβλήματα ροής που διερευνώνται σε αυτήν τη διατριβή διέπονται από τις Reynolds-Averaged Navier Stokes (RANS) εξισώσεις για μόνιμες ροές ασυμπίεστων ρευστών.

Οι μέθοδοι βελτιστοποίησης στην αεροδυναμική διακρίνονται σε στοχαστικές [11, 36, 44, 7] και αιτιοκρατικές [18, 8, 9, 38, 34, 37, 48]. Κύριοι εκπρόσωποι των πρώτων είναι οι εξελικτικοί αλγόριθμοι (Evolutionary Algorithms, EAs), στις οποίες προγραμματίζονται αλγόριθμοι που μιμούνται διαδικασίες φυσικής εξέλιξης, ώστε να ανανεώνονται σε κάθε κύκλο βελτιστοποίησης οι μεταβλητές σχεδιασμού. Πλεονεκτήματά τους είναι η δυνατότητα εύρεσης ολικού ακροτάτου της συνάρτησης-στόχου, η ευελιξία χρήσης τους αντιμετωπίζοντας το λογισμικό CFD με τη λογική του 'μαύρου κουτιού' και η δυνατότητα υπολογισμού του μετώπου μη-κυριαρχούμενων λύσεων (μέτωπο Pareto) σε προβλήματα πολλών στόχων. Μειονέκτημα είναι η εξάρτηση του κόστους από το πλήθος των μεταβλητών σχεδιασμού. Με σκοπό τη μείωση του υπολογιστικού κόστους των EAs αναπτύχθηκαν οι Παράλληλοι [20, 1], Ασύγχρονοι [2, 6] και οι Ιεραρχικοί EAs, ενώ ιδιαιτέρως διαδεδομένη είναι και η χρήση Μεταπροτύπων (υποκατάστατων του CFD μοντέλων αξιολόγησης) [13, 16].

Οι αιτιοκρατικές μέθοδοι βασίζονται στην κλίση της συνάρτησης-στόχου (Gradient-

Based Methods, GBMs). Η ανανέωση των μεταβλητών σχεδιασμού γίνεται με βάση τις παραγώγους της συνάρτησης-στόχου ως προς τις μεταβλητές σχεδιασμού, που αναφέρονται ως παράγωγοι ευαισθησίας ('Sensitivity Derivatives'). Πλεονέκτημα αυτών των μεθόδων είναι το μικρότερο κόστος σε σχέση με τους EAs σε προβλήματα βελτιστοποίησης ενός στόχου, αν και είναι πιθανός ο εγκλωβισμός σε τοπικό ακρότατο. Η απόδοση των GBMs εξαρτάται από τη μέθοδο υπολογισμού των παραγώγων ευαισθησίας. Η απλούστερη μέθοδος είναι οι Πεπερασμένες Διαφορές (Finite Differences, FDs), στις οποίες επιβάλλονται απειροστά μικρές αλλαγές $+/-\epsilon$ σε κάθε μεταβλητή σχεδιασμού προκειμένου να υπολογιστούν οι παράγωγοι ευαισθησίας, με αποτέλεσμα να στοιχίζει συνολικά $2N$ 'ισοδύναμες' επιλύσεις των εξισώσεων ροής (Equivalent Flow Solutions, EFS), όταν χρησιμοποιούνται κεντρικές διαφορές. Οι FDs υπολογίζουν παραγώγους αναφοράς και χρησιμοποιούνται για την πιστοποίηση της ακρίβειας των παραγώγων που υπολογίζονται με τις συζυγείς μεθόδους. Μία εναλλακτική των FDs είναι η μέθοδος της Ευθείας Διαφόρισης (Direct Differentiation, DD). Σε αυτήν, η J παραγωγίζεται ως προς b_n και εμφανίζονται οι παράγωγοι των ροϊκών μεταβλητών ως προς b_n. Για να υπολογιστούν οι τελευταίες ποσότητες, οι πρωτεύουσες εξισώσεις διαφορίζονται ως προς b_n και προκύπτει σύστημα N εξισώσεων με N αγνώστους, με κόστος επίλυσης N EFS.

Οι συζυγείς μέθοδοι (adjoint methods) χαρακτηρίζονται από το μικρότερο κόστος υπολογισμού των παραγώγων ευαισθησίας. Με αυτές αποφεύγεται ο υπολογισμός των παραγώγων των ροϊκών ποσοτήτων ως προς b_n που εμφανίζεται στη μέθοδο DD. Για να γίνει αυτό δυνατό, εισάγονται τόσα πεδία νέων βαθμών ελευθερίας ή πολλαπλασιαστών Lagrange, όσες και οι πρωτεύουσες εξισώσεις και διαμορφώνεται η επαυξημένη συνάρτηση L η οποία ισούται με τη J συν τα υπόλοιπα των εξισώσεων ροής πολλαπλασιασμένα με τα πεδία των συζυγών μεταβλητών. Με την παραγώγιση της L ως προς b_n μηδενίζονται οι πολλαπλασιαστές των παραγώγων των ροϊκών μεγεθών και προκύπτουν οι συζυγείς εξισώσεις και οριακές συνθήκες. Οι εναπομείναντες όροι αποτελούν τις παραγώγους ευαισθησίας. Το κόστος υπολογισμού των παραγώγων είναι ίσο με την επίλυση των πρωτευουσών και συζυγών εξισώσεων, δηλαδή περίπου 2 EFS σε προβλήματα βελτιστοποίησης χωρίς περιορισμούς.

Ανάλογα με τη σειρά διαφόρισης και διακριτοποίησης οι συζυγείς μέθοδοι κατηγοριοποιούνται στη συνεχή (continuous adjoint) [21, 24, 22, 5] και τη διακριτή (discrete adjoint) [14, 3, 4, 17]. Αυτή η διατριβή ασχολείται αποκλειστικά με τη συνεχή συζυγή μέθοδο, η οποία διατυπώνεται με τους εξής τρεις τρόπους:

1. Η ανάπτυξη της συνεχούς συζυγούς μεθόδου σύμφωνα με τους [21, 33] οδηγεί σε παραγώγους ευαισθησίας που περιλαμβάνουν Χωρικά Ολοκληρώματα Field Integrals· γι' αυτό καλείται αυτή η μέθοδος ως η διατύπωση *FI*. Η μετατόπιση των εσωτερικών κόμβων του πλέγματος λαμβάνεται υπόψη μέσω των 'παραγώγων ευαισθησίας πλέγματος' $\delta x_i/\delta b_n$, (grid sensitivities). Ο υπολογισμός αυτών των ποσοτήτων γίνεται είτε μέσω FDs, με κόστος ίσο με $2N$ μετατοπίσεις πλέγματος, είτε μέσω της διαφόρισης των εξισώσεων του μοντέλου μετατόπισης πλέγματος (Grid Displacement Model, GDM). Στην τελευταία περίπτωση,

προκύπτουν μαθηματικές εκφράσεις παρόμοιες με τις εξισώσεις του GDM, που επιτρέπουν τον υπολογισμό των παραγώγων ευαισθησίας πλέγματος με κόστος N μετατοπίσεις πλέγματος. Ανάλογα με το μέγεθος του πλέγματος και το είδος του GDM, το κόστος υπολογισμού των $\delta x_i/\delta b_n$ μπορεί να πλησιάζει το 1 EFS. Πλεονέκτημα της διατύπωσης *FI* είναι η υψηλή ακρίβεια των παραγώγων ευαισθησίας, [21, 33, 29].

2. Στις [29, 27], προκειμένου να αποφευχθεί ο υπολογισμός των $\delta x_i/\delta b_n$ με κόστος που εξαρτάται από το πλήθος των μεταβλητών σχεδιασμού, αναπτύχθηκαν συζυγείς εξισώσεις ως προς το GDM, το οποίο έγινε η υπόθεση ότι διέπεται από εξισώσεις Laplace. Αναπτύσσοντας με αυτόν τον τρόπο τη συνεχή συζυγή μέθοδο, οι παράγωγοι ευαισθησίας αποτελούνται αποκλειστικά από επιφανειακά ολοκληρώματα (Surface Integrals). Αυτό έχει ως αποτέλεσμα το υπολογιστικό κόστος να είναι μικρότερο της συζυγούς διατύπωσης *FI*. Αυτή η μέθοδος καλείται συζυγής διατύπωση Enhanced *SI* (*E-SI*), επειδή αποτελεί βελτίωση μεθόδων που παρουσιάστηκαν παλαιότερα στη βιβλιογραφία [23, 5] και οδηγούσαν σε παραγώγους αποτελούμενες επίσης μόνο από επιφανειακά ολοκληρώματα, ωστόσο χαμηλότερης ακρίβειας, γιατί δεν λαμβανόταν υπόψη η μετατόπιση των εσωτερικών κόμβων.

3. Η προαναφερθείσα συζυγής διατύπωση που οδηγεί σε παραγώγους που αποτελούνται από επιφανειακά ολοκληρώματα αλλά δεν λαμβάνει υπόψη το GDM, ονομάζεται διατύπωση "Severed" *SI* και χαρακτηρίζεται από χαμηλότερη ακρίβεια παραγώγων ευαισθησίας.

Εκτενής βιβλιογραφική επισκόπηση των μεθόδων που αναπτύσσονται σε αυτήν τη διατριβή καθώς και λεπτομερής αναφορά στις συζυγείς μεθόδους που αναπτύχθηκαν στην ΜΠΥΡ&Β ΕΜΠ παρατίθενται στο πλήρες κείμενο της διατριβής.

Κεφάλαιο 2

Η Συνεχής Συζυγής Μέθοδος για Τυρβώδεις Ροές Ασυμπίεστου Ρευστού

Σε αυτό το κεφάλαιο παρουσιάζονται οι διατυπώσεις *FI*, (Severed) *SI* και *E-SI* της συνεχούς συζυγούς μεθόδου, για μόνιμες τυρβώδεις ροές ασυμπίεστων ρευστών. Η μοντελοποίηση της τύρβης γίνεται με το μοντέλο Spalart–Allmaras. Για τον υπολογισμό παραγώγων με ακρίβεια αποφεύγεται η παραδοχή της 'παγωμένης τύρβης' και παραγωγίζεται το μοντέλο τύρβης [40, 27, 50].

2.1 Εξισώσεις του Πρωτεύοντος Προβλήματος

Οι εξισώσεις για μόνιμες τυρβώδεις ροές ασυμπίεστων ρευστών είναι οι εξής [30]:

$$R^p = -\frac{\partial v_i}{\partial x_i} = 0 \tag{2.1α'}$$

$$R^v_i = v_j \frac{\partial v_i}{\partial x_j} + \frac{\partial p}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} = 0 \tag{2.1β'}$$

$$R^{\widetilde{\nu}} = v_j \frac{\partial \widetilde{\nu}}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu}}{\partial x_j}\right] - \frac{c_{b2}}{\sigma}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right)^2 - \widetilde{\nu}\mathcal{P}(\widetilde{\nu}) + \widetilde{\nu}\mathcal{D}(\widetilde{\nu}) = 0 \tag{2.1γ'}$$

$$R^{\Delta} = \frac{\partial\left(c_j \Delta\right)}{\partial x_j} - \Delta\frac{\partial^2 \Delta}{\partial x_j^2} - 1 = 0, \quad c_j = \frac{\partial \Delta}{\partial x_j} \tag{2.1δ'}$$

οι οποίες αντιστοιχούν στις εξισώσεις συνέχειας, ορμής, του μοντέλου Spalart–Allmaras και της Hamilton-Jacobi (Eikonal) που επιλύονται με τον αλγόριθμο SIMPLE [43]. Επαναλαμβανόμενοι δείκτες υποδηλώνουν άθροιση. Στις παραπάνω εξισώσεις, p είναι το πεδίο της στατικής πίεσης διαιρεμένο με τη σταθερή πυκνότητα, v_i οι συνιστώσες της ταχύτητας, $\tau_{ij} = (\nu + \nu_t)\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)$ ο τανυστής των τάσεων, ν και ν_t η κινηματική και η τυρβώδης συνεκτικότητα, $\widetilde{\nu}$ η μεταβλητή του μοντέλου τύρβης και

Δ το πεδίο της απόστασης κάθε σημείου από το πλησιέστερο όριο.

2.2 Διατυπώσεις της Συνεχούς Συζυγούς Μεθόδου

Για την ανάπτυξη της συζυγούς μεθόδου ορίζεται η επαυξημένη συνάρτηση L

$$L = J + \int_\Omega q R^p d\Omega + \int_\Omega u_i R_i^v d\Omega + \int_\Omega \widetilde{\nu_a} R^{\widetilde{\nu}} d\Omega + \int_\Omega \Delta^a R^\Delta d\Omega \tag{2.2}$$

όπου J η συνάρτηση-στόχος του προβλήματος βελτιστοποίησης, $q,\ u_i, \widetilde{\nu_a}$ και Δ^a τα πεδία της συζυγούς πίεσης, ταχύτητας, μεταβλητής μοντέλου τύρβης και απόστασης, αντίστοιχα. Η διαφόριση της εξ. 2.2, που παρουσιάζεται στο πλήρες κείμενο της διατριβής, οδηγεί στις παρακάτω συζυγείς εξισώσεις

$$R^q = -\frac{\partial u_i}{\partial x_i} + j_\Omega^p = 0 \tag{2.3α'}$$

$$\begin{aligned} R_i^u = {} & u_j \frac{\partial v_j}{\partial x_i} - \frac{\partial (v_j u_i)}{\partial x_j} - \frac{\partial \tau_{ij}^a}{\partial x_j} + \frac{\partial q}{\partial x_i} + \widetilde{\nu_a} \frac{\partial \widetilde{\nu}}{\partial x_i} - \frac{\partial}{\partial x_l}\left(\widetilde{\nu_a}\widetilde{\nu}\frac{C_Y}{Y} e_{mjk}\frac{\partial v_k}{\partial x_j} e_{mli}\right) \\ & + j_{\Omega,i}^v = 0,\ i = 1, 2(, 3) \end{aligned} \tag{2.3β'}$$

$$\begin{aligned} R^{\widetilde{\nu_a}} = {} & -\frac{\partial (v_j \widetilde{\nu_a})}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu_a}}{\partial x_j}\right] + \frac{1}{\sigma}\frac{\partial \widetilde{\nu_a}}{\partial x_j}\frac{\partial \widetilde{\nu}}{\partial x_j} + 2\frac{c_{b2}}{\sigma}\frac{\partial}{\partial x_j}\left(\widetilde{\nu_a}\frac{\partial \widetilde{\nu}}{\partial x_j}\right) \\ & + \widetilde{\nu_a}\widetilde{\nu} C_{\widetilde{\nu}} - (\mathcal{P} - \mathcal{D})\,\widetilde{\nu_a} + \frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\frac{\delta \nu_t}{\delta \widetilde{\nu}} + j_\Omega^{\widetilde{\nu}} = 0 \end{aligned} \tag{2.3γ'}$$

$$R^{\Delta^a} = -2\frac{\partial}{\partial x_j}\left(\Delta^a \frac{\partial \Delta}{\partial x_j}\right) + \widetilde{\nu}\widetilde{\nu_a} C_\Delta = 0 \tag{2.3δ'}$$

Οι παράγωγοι της J ως προς b_n είναι οι εξής

$$\begin{aligned} \left.\frac{\delta J}{\delta b_n}\right|_{\text{FI}} = {} & \int_\Omega \left(-u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right)\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \\ & + \int_\Omega j_\Omega \frac{\partial}{\partial x_k}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega + \int_{S_{W_p}} j_{S_i,k}^g \frac{\delta x_k}{\delta b_n} n_i dS + \int_{S_{W_p}} j_{S,i} \frac{\delta\,(n_i dS)}{\delta b_n} \\ & - \int_{S_{W_p}} \left[\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k t_l^I t_m^I \tau_{ij} \frac{\delta (t_i^I t_j^I)}{\delta b_n}\right] dS - \int_{S_{W_p}} \left[\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k t_l^{II} t_m^{II} \tau_{ij} \frac{\delta (t_i^{II} t_j^{II})}{\delta b_n}\right] dS \\ & - \int_{S_{W_p}} \left[\left(-u_{\langle n \rangle} + \frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k n_l n_m\right)\tau_{ij} \frac{\delta (n_i n_j)}{\delta b_n}\right] dS \end{aligned}$$

$$-\int_{S_{W_p}}\left[\left(\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k (t_l^{II} t_m^I + t_l^I t_m^{II})\right) \tau_{ij} \frac{\delta (t_i^{II} t_j^I)}{\delta b_n}\right] dS \tag{2.4}$$

Λόγω της παρουσίας Χωρικών Ολοκληρωμάτων (Field Integrals) στην εξ. 2.4, αυτή η διατύπωση της συζυγούς μεθόδου ονομάζεται *FI*. Προκειμένου να υπολογιστεί το ογκικό ολοκλήρωμα στην εξ. 2.4, είναι αναγκαίος ο υπολογισμός των παραγώγων ευαισθησίας πλέγματος, $\delta x_i / \delta b_n$ (grid sensitivities). Αυτό κοστίζει όσο η μετατόπιση του υπολογιστικού πλέγματος N φορές, κάτι που αυξάνει το κόστος υπολογισμού των παραγώγων της J.

Για να αποφευχθεί ο υπολογισμός των παραγώγων ευαισθησίας πλέγματος και να μειωθεί το υπολογιστικό κόστος, χωρίς να αμεληθεί η μετατόπιση του πλέγματος, αναπτύχθηκε για πρώτη φορά από τους [27, 29], στο πλαίσιο διατριβής στην ΜΠΥΡ&Β ΕΜΠ, η συνεχής συζυγής μέθοδος για τις εξισώσεις μετατόπισης πλέγματος. Γίνεται η υπόθεση ότι εξισώσεις Laplace R_i^m διέπουν τη μετατόπιση του πλέγματος και η επαυξημένη συνάρτηση ορίζεται πλέον ως

$$L = J + \int_\Omega q R^p d\Omega + \int_\Omega u_i R_i^v d\Omega + \int_\Omega \widetilde{\nu_a} R^{\widetilde{\nu}} d\Omega + \int_\Omega \Delta^a R^\Delta d\Omega + \int_\Omega m_i^a R_i^m d\Omega \tag{2.5}$$

όπου m_i^a αντιστοιχεί στο πεδίο της συζυγούς μετατόπισης. Μετά από μαθηματική ανάπτυξη προκύπτουν οι ίδιες συζυγείς εξισώσεις ροής και οριακές συνθήκες όπως στη διατύπωση *FI*. Επιπλέον, προκύπτουν οι συζυγείς εξισώσεις μετατόπισης πλέγματος

$$\begin{aligned}&\frac{\partial^2 m_k^a}{\partial {x_j}^2} - \frac{\partial}{\partial x_j}\left(-u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q \frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right)\\ &- \frac{\partial j_\Omega}{\partial x_k} = 0,\ \ k = 1, 2(, 3)\end{aligned} \tag{2.6}$$

και η έκφραση της παραγώγου της J ως προς b_n δίνεται από την ακόλουθη σχέση

$$\begin{aligned}\left.\frac{\delta J}{\delta b_n}\right|_{\text{E-SI}} =& \int_{S_{W_p}}\left(-u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q \frac{\partial v_j}{\partial x_k} + \Theta_{jk}\right) n_j \frac{\delta x_k}{\delta b_n} dS\\ &+ \int_{S_{W_p}} j_\Omega n_k \frac{\delta x_k}{\delta b_n} d\Omega + \int_{S_{W_p}} j_{S_i,k}^g \frac{\delta x_k}{\delta b_n} n_i dS + \int_{S_{W_p}} j_{S,i} \frac{\delta\left(n_i dS\right)}{\delta b_n} - \int_{S_{W_p}} \frac{\partial m_i^a}{\partial n} \frac{\delta x_i}{\delta b_n} dS\\ &- \int_{S_{W_p}}\left[\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k t_l^I t_m^I \tau_{ij} \frac{\delta (t_i^I t_j^I)}{\delta b_n}\right] dS - \int_{S_{W_p}}\left[\frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k t_l^{II} t_m^{II} \tau_{ij} \frac{\delta (t_i^{II} t_j^{II})}{\delta b_n}\right] dS\\ &- \int_{S_{W_p}}\left[\left(-u_{\langle n \rangle} + \frac{\partial j_{S_{W_p,k}}}{\partial \tau_{lm}} n_k n_l n_m\right) \tau_{ij} \frac{\delta (n_i n_j)}{\delta b_n}\right] dS\end{aligned}$$

$$-\int_{S_{W_p}}\left[\left(\frac{\partial j_{S_{W_p},k}}{\partial \tau_{lm}} n_k (t_l^{II} t_m^I + t_l^I t_m^{II})\right) \tau_{ij} \frac{\delta (t_i^{II} t_j^I)}{\delta b_n}\right] dS \tag{2.7}$$

Σε αντίθεση με τη διατύπωση *FI*, η παράγωγος της J που δίνεται από την εξ. 2.7 αποτελείται μόνο από επιφανειακά ολοκληρώματα. Σε προγενέστερες μελέτες της [29], η μετατόπιση του πλέγματος δεν λαμβανόταν υπόψη [23, 5] και οι παράγωγοι της J υπολογίζονταν από την εξ. 2.7 χωρίς το 5ο ολοκλήρωμα, κάτι που μειώνει την ακρίβεια των παραγώγων [29, 27]. Διατυπώσεις της συζυγούς μεθόδου όπως αυτές που παρουσιάζονται από τους [23, 5] και οδηγούν σε παραγώγους με επιφανειακά ολοκληρώματα (Surface Integrals) ονομάζονται (Severed) *SI*. Αντίθετως, η ακριβέστερη (Enhanced) διατύπωση που οδηγεί στην εξ. 2.7 ονομάζεται *E-SI*.

2.3 Συγκρίσεις των Διατυπώσεων της Συζυγούς Μεθόδου

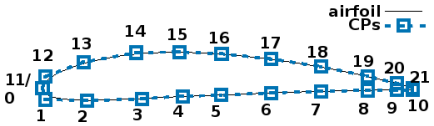
Εξετάζονται οι διαφορές ανάμεσα στις διατυπώσεις *FI*, *E-SI* και (Severed) *SI* μέσα από εφαρμογές. Στο Πρόβλημα 1 συγκρίνονται οι *FI*, (Severed) *SI* και *E-SI* διατυπώσεις για τη ροή γύρω από τη μεμονωμένη αεροτομή NACA4412, με $Re = 10^6$, γωνία της επ'άπειρον ροής $\alpha_\infty = 0^o$ και αδιαστατοποιημένη απόσταση του βαρυκέντρου του πρώτου κελιού από τον τοίχο $y^+ < 0.06$. Υπολογίζονται παράγωγοι της άνωσης

$$J^{forces} = \int_{S_{W_p}} \left(p\delta_i^j - \tau_{ij}\right) r_i n_j dS \tag{2.8}$$

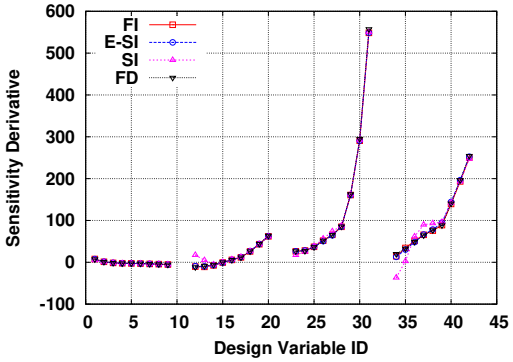
ως προς τις x και y συντεταγμένες των 18 σημείων ελέγχου NURBS που παραμετροποιούν το σχήμα της αεροτομής, σχ. 2.1α΄. Οι παράγωγοι που υπολογίζονται από τις *E-SI* και *FI* διατυπώσεις ταυτίζονται με αυτές που υπολογίζονται με πεπερασμένες διαφορές (Finite Differences, FD), σχ. 2.1β΄. Στο Πρόβλημα 2 συγκρίνονται οι συζυγείς διατυπώσεις *FI* και *E-SI* για τη τυρβώδη ροή ($Re = 10^5$, $y^+ < 1$) μέσα σε έναν αγωγό σχήματος S. Υπολογίζονται παράγωγοι για τις απώλειες ολικής πίεσης μεταξύ των διατομών εισόδου και εξόδου του αγωγού

$$J^{pt} = -\int_{S_{I,O}} \left(p + \frac{1}{2} v_k^2\right) v_j n_j dS \tag{2.9}$$

ως προς τις y συντεταγμένες 9 σημείων ελέγχου ογκομετρικών B-Splines που παραμετροποιούν τον αγωγό σχ. 2.2α΄. Οι παράγωγοι που υπολογίζονται από τις συζυγείς διατυπώσεις *FI* και *E-SI* ταυτίζονται με τις FDs. Οι διαφορές ανάμεσα στις δύο συζυγείς διατυπώσεις είναι τόσο μικρές που δεν επηρεάζουν την πορεία σύγκλισης βελτιστοποιήσεων οι οποίες βασίζονται στη μέθοδο BFGS [38], η οποία είναι ευαίσθητη στην ακρίβεια των παραγώγων [12]. σχ. 2.2γ΄. Στο τέλος κάθε βελτιστοποίησης, η J^{pt}

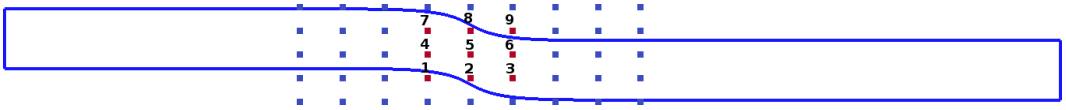


(α΄) Γεωμετρία και παραμετροποίηση

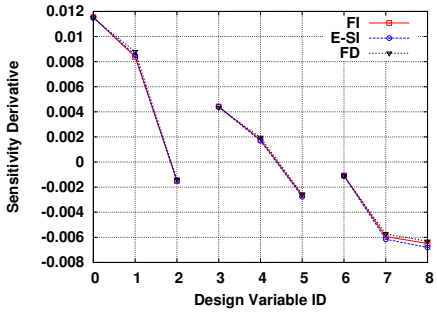


(β΄) Παράγωγοι της άνωσης

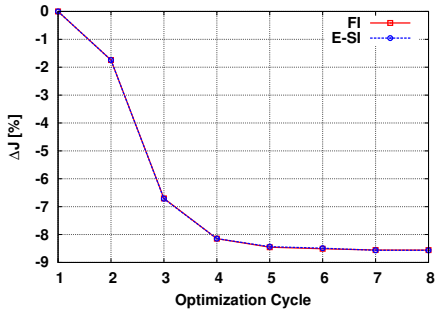
Σχήμα 2.1: Πρόβλημα 1: (α) 22 σημεία ελέγχου NURBS παραμετροποιούν το σχήμα της αεροτομής, με το πρώτο και τελευταίο για κάθε πλευρά της αεροτομής να παραμένουν ακίνητα. (β) Οι πρώτες 18 παράγωγοι αντιστοιχούν στις x και οι επόμενες 18 στις y συντεταγμένες των σημείων ελέγχου.



(α΄) Γεωμετρία και παραμετροποίηση



(β΄) Παράγωγοι απωλειών ολικής πίεσης



(γ΄) Πορεία σύγκλισης βελτιστοποιήσεων

Σχήμα 2.2: Πρόβλημα 2: (α) Τα κόκκινα σημεία ελέγχου μετακινούνται ενώ τα μπλε παραμένουν ακίνητα. (β) Οι διατυπώσεις *FI* και *E-SI* αναπαράγουν τις παραγώγους των πεπερασμένων διαφορών. (γ) Βελτιστοποιήσεις με *FI* και *E-SI* παραγώγους.

μειώθηκε κατά $\sim 8.5\%$. Περισσότερες λεπτομέρειες και επιπρόσθετες διερευνήσεις περιλαμβάνονται στο πλήρες κείμενο της διατριβής. Με αυτές τις εφαρμογές επιβεβαιώθηκαν συμπεράσματα που ήταν γνωστά από τις εργασίες [29, 27], σύμφωνα με τα οποία η διατύπωση *E-SI* είναι τόσο ακριβής όσο και η υπολογιστικά ακριβότερη *FI*. Αντίθετα, οι παράγωγοι που υπολογίζονται με τη διατύπωση (Severed) *SI* είναι λιγότερο ακριβείς, επειδή αμελούν τη μετατόπιση του πλέγματος.

Κεφάλαιο 3

Η Συνεχής Συζυγής Μέθοδος για τη Συζευγμένη Μεταφορά Θερμότητας

Σε αυτό το κεφάλαιο επεκτείνονται οι διατυπώσεις *FI*, *E-SI* και Severed *SI* της συνεχούς συζυγούς μεθόδου σε προβλήματα Συζευγμένης Μεταφοράς Θερμότητας (Conjugate Heat Transfer, CHT), με τη ροή του ασυμπίεστου ρευστού να είναι χρονικά μόνιμη και τυρβώδης.

3.1 Πρωτεύον Πρόβλημα CHT

Οι ποσότητες που αναφέρονται στο ρευστό και στο στερεό συμβολίζονται με τους εκθέτες F και S αντίστοιχα. Σε προβλήματα CHT, εκτός από τις εξισώσεις ροής εξ. 2.1α′–2.1δ′, επιλύονται η ενεργειακή εξίσωση στο ρευστό και η εξίσωση αγωγής της θερμότητας στο στερεό

$$R^{T^F} = \rho^F v_j c_p \frac{\partial T^F}{\partial x_j} + \rho^F \frac{v_j}{2} \frac{\partial v_k^2}{\partial x_j} - \frac{\partial}{\partial x_j}\left(k^F \frac{\partial T^F}{\partial x_j}\right) = 0 \tag{3.1}$$

$$R^{T^S} = -\frac{\partial}{\partial x_j}\left[k^S \frac{\partial T^S}{\partial x_j}\right] = 0 \tag{3.2}$$

όπου $k^F = \rho^F c_p \alpha_{eff} = \rho^F c_p \left(\frac{\nu}{Pr} + \frac{\nu_t}{Pr_t}\right)$ και k^S είναι οι συντελεστές θερμικής αγωγιμότητας του ρευστού και στερεού, ρ^F η πυκνότητα του ρευστού, c_p ο συντελεστής μεταφοράς θερμότητας υπό σταθερή πίεση, Pr και Pr_t οι αριθμοί Prandtl για στρωτές και τυρβώδεις ροές. Επιπλέον, ορίζονται οι συντελεστές θερμικής διάχυσης για στρωτές και τυρβώδεις ροές α, α_t με α_{eff}, με $\alpha_{eff} = \alpha + \alpha_t$, $\alpha = \nu/Pr$, $\alpha_t = \nu_t/Pr_t$.

Σε κάθε σημείο στη διεπιφάνεια ρευστού-στερεού (Fluid-Solid Interface, FSI), $\overline{S}^F$ και $\overline{S}^S$, επιβάλλονται συνθήκες διατήρησης της θερμορροής και ισότητας της

θερμοκρασίας, ως εξής

$$T^S\big|_{\overline{S}_S} = T^F\big|_{\overline{S}_F} \tag{3.3}$$

$$k^S \frac{\partial T^S}{\partial n}\bigg|_{\overline{S}^S} = -k^F \frac{\partial T^F}{\partial n}\bigg|_{\overline{S}^F} \tag{3.4}$$

Λεπτομερής αναφορά στις υπόλοιπες οριακές συνθήκες γίνεται στο πλήρες κείμενο της διατριβής.

3.2 Η Συνεχής Συζυγής Μέθοδος για Προβλήματα CHT

Μετά από μαθηματική ανάπτυξη προκύπτουν οι συζυγείς εξισώσεις μεταφοράς θερμότητας του ρευστού και του στερεού

$$R_F^{T_a} = -\frac{\partial (v_j T_a^F)}{\partial x_j} - \frac{\partial}{\partial x_j}\left(\alpha_{eff}\frac{\partial T_a^F}{\partial x_j}\right) + j_\Omega^{F,T} = 0 \tag{3.5}$$

$$R^{T_a^S} = -\frac{\partial}{\partial x_j}\left(k^S \frac{\partial T_a^S}{\partial x_j}\right) + j_\Omega^{S,T} = 0 \tag{3.6}$$

όπου T_a^F, T_a^S τα πεδία των συζυγών θερμοκρασιών. Επιπλέον όροι προκύπτουν από τη διαφόριση της εξ. 3.1 στο αριστερό μέλος των συζυγών εξισώσεων ορμής εξ. 2.36′ και της συζυγούς εξίσωσης του μοντέλου τύρβης εξ. 2.3γ′. Οι συζυγείς οριακές συνθήκες στο FSI είναι οι εξής

$$T_a^F\big|_{\overline{S}^F} = T_a^S\big|_{\overline{S}^S} \tag{3.7α′}$$

$$\left(k^F\frac{\partial T_a^F}{\partial n} + \frac{\partial j_{S_i}^F}{\partial T^F}n_i + j_S^{F,T}\right)\bigg|_{\overline{S}^F} = -\left(k^S\frac{\partial T_a^S}{\partial n^S} + \frac{\partial j_{S,i}^S}{\partial T^S}n_i^S + j_S^{S,T}\right)\bigg|_{\overline{S}^S} \tag{3.7β′}$$

ενώ λεπτομερής αναφορά στις υπόλοιπες οριακές συνθήκες γίνεται στο πλήρες κείμενο της διατριβής. Οι παράγωγοι των συζυγών διατυπώσεων *FI* και *E-SI* για προβλήματα μόνο ροής επεκτείνονται στα προβλήματα CHT ως εξής

$$\begin{aligned}\frac{\delta J}{\delta b_n}\bigg|_{FI} = \frac{\delta J}{\delta b_n}\bigg|_{FI,inco} &+ \int_{\Omega^S}\left[-k^S\frac{\partial T_a^S}{\partial x_j}\frac{\partial T^S}{\partial x_k} + T_a^S\frac{\partial}{\partial x_k}\left(k^S\frac{\partial T^S}{\partial x_j}\right)\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega \\ &+ \int_{\Omega^F}\rho^F c_p\left[-T_a^F v_j\frac{\partial T^F}{\partial x_k} - T_a^F v_i v_j T_a^F\frac{\partial v_i}{\partial x_k} - \alpha_{eff}\frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_k}\right. \\ &\left. + T_a^F\frac{\partial}{\partial x_k}\left(\alpha_{eff}\frac{\partial T^F}{\partial x_j}\right)\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega + \int_{\overline{S}_p^F\cup S_{W,Fl,p}^F} k^F T_a^F\frac{\partial T^F}{\partial x_j}\frac{\delta n_j}{\delta b_n}dS\end{aligned}$$

$$+\int_{\overline{S}_p^S \cup S_{W,Fl_p}^S \cup S_{CH_p}^S} k^S T_a^S \frac{\partial T^S}{\partial x_j}\frac{\delta n_j}{\delta b_n} dS \tag{3.8}$$

και

$$\begin{aligned}\left.\frac{\delta J}{\delta b_n}\right|_{ESI} &= \left.\frac{\delta J}{\delta b_n}\right|_{ESI,inco} + \int_{S_{W_P}^F} \rho^F c_p \Bigg[-\alpha_{eff}\frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_k} + T_a^F \frac{\partial}{\partial x_k}\left(\alpha_{eff}\frac{\partial T^F}{\partial x_j}\right)\Bigg]\frac{\delta x_k}{\delta b_n} dS \\ &+ \int_{S_{W_p}^S} \left\{ \left[-k^S \frac{\partial T_a^S}{\partial x_j}\frac{\partial T^S}{\partial x_k} + T_a^S \frac{\partial}{\partial x_k}\left(k^S \frac{\partial T^S}{\partial x_j}\right) + j_{S_i,k}^{S,g}\right] n_j + \left(j_\Omega^S + j_\Omega^{S,g}\right) n_k \right\} \frac{\delta x_k}{\delta b_n} dS \\ &- \int_{S_{W_p}^F} \left[\left(\frac{\partial j_{S_{W_p},k}^F}{\partial \tau_{lm}} n_k (t_l^{II} t_m^I + t_l^I t_m^{II}) \right) \tau_{ij} \frac{\delta (t_i^{II} t_j^I)}{\delta b_n} \right] dS + \int_{\overline{S}_p^F \cup S_{W,Fl,p}^F} k^F T_a^F \frac{\partial T^F}{\partial x_j}\frac{\delta n_j}{\delta b_n} dS \\ &+ \int_{\overline{S}_p^S \cup S_{W,Fl_p}^S \cup S_{CH_p}^S} k^S T_a^S \frac{\partial T^S}{\partial x_j}\frac{\delta n_j}{\delta b_n} dS + \int_{S_{W_p}^S} j_{S,i}^S \frac{\delta\left(n_i dS\right)}{\delta b_n} - \int_{S^S} \frac{\partial m_i^{S,a}}{\partial n}\frac{\delta x_i}{\delta b_n} dS \end{aligned} \tag{3.9}$$

όπου οι όροι $\left.\frac{\delta J}{\delta b_n}\right|_{FI,inco}$ και $\left.\frac{\delta J}{\delta b_n}\right|_{ESI,inco}$ δίνονται από τις εξ. 3.8 και 2.7 αντίστοιχα. Τα πεδία των συζυγών μετατοπίσεων υπολογίζονται επιλύοντας τις ακόλουθες εξισώσεις

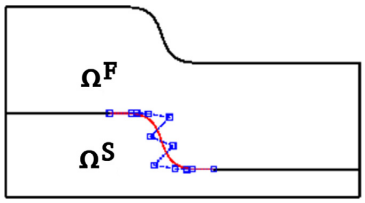
$$\begin{aligned}\frac{\partial^2 m_k^{F,a}}{\partial {x_j}^2} &= \frac{\partial}{\partial x_j}\Bigg[- u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q\frac{\partial v_j}{\partial x_k} + \Theta_{jk} - \rho^F c_p T_a^F v_j \frac{\partial T^F}{\partial x_k} \\ &- \rho^F T_a^F v_i v_j T_a^F \frac{\partial v_i}{\partial x_k} - \rho^F \alpha_{eff} c_p \frac{\partial T_a^F}{\partial x_j}\frac{\partial T^F}{\partial x_k} + T_a^F \frac{\partial}{\partial x_k}\left(\rho^F \alpha_{eff} c_p \frac{\partial T^F}{\partial x_j}\right)\Bigg] - \frac{\partial j_\Omega^F}{\partial x_k} \end{aligned} \tag{3.10α'}$$

$$\frac{\partial^2 m_k^{S,a}}{\partial {x_j}^2} = \frac{\partial}{\partial x_j}\left[-k^S \frac{\partial T_a^S}{\partial x_j}\frac{\partial T^S}{\partial x_k} + T_a^S \frac{\partial}{\partial x_k}\left(k^S \frac{\partial T^S}{\partial x_j}\right)\right] - \frac{\partial\left(j_\Omega^S + j_\Omega^{S,g}\right)}{\partial x_k} \tag{3.10β'}$$

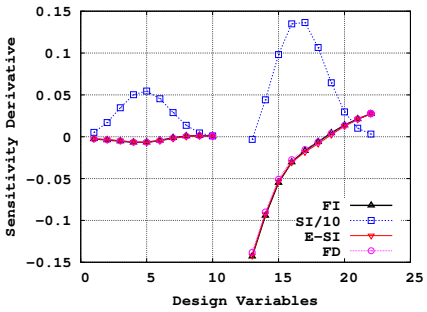
Στη διατύπωση *E-SI*, τα επιφανειακά ολοκληρώματα που περιλαμβάνουν το πεδίο της συζυγούς μετατόπισης πλέγματος παραλείπονται.

3.3 Εφαρμογές

Στο Πρόβλημα 3, σχ. 3.1, γίνεται μελέτη για μια περίπτωση CHT με τυρβώδη ροή ($Re = 10^4,\ y^+ < 0.018$). Στην είσοδο, $T^F = 291.2\ K$ και στα στερεά όρια που δεν ανήκουν στο FSI, $T^S = 300\ K$. Επίσης, $c_p = 4181\ J/kg/K$, $Pr = 0.1$ και $k^S = 60\ J/K/m/sec$. Υπολογίζονται παράγωγοι της συνάρτησης J^{penT} που δίνεται



(α′) Γεωμετρία και παραμετροποίηση



(β′) Παράγωγοι

Σχήμα 3.1: Πρόβλημα 3: (α) Το κόκκινο τμήμα του FSI παραμετροποιείται από 12 σημεία ελέγχου. (β) Παράγωγοι της J^{penT} υπολογισμένες από τις διατυπώσεις *FI*, *E-SI* και (Severed) *SI* συγκρίνονται με FDs.

από τη σχέση

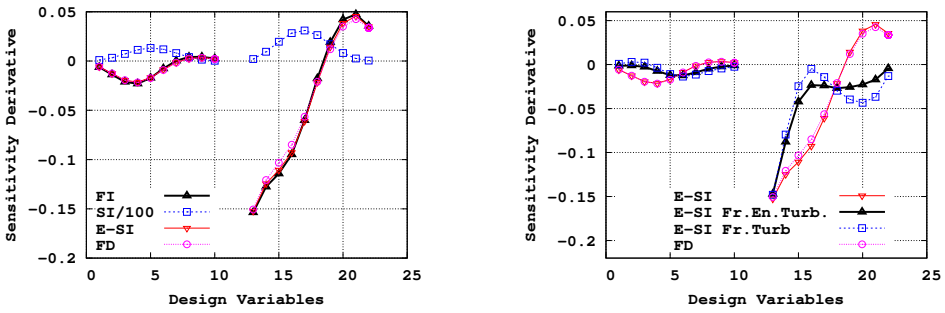
$$J^{penT} = \int_{\Omega^S} \frac{j^{penT}}{\int_{\Omega^S} d\Omega} d\Omega, \; j^{penT} = \begin{cases} \left[1 - \frac{1}{1+e^{k_2(T^S - T_{crit})+k_1}}\right] & , if \; T^S \leq T_{crit} \\ \alpha \left(T^S - T_{crit}\right) + \beta & , if \; T^S > T_{crit} \end{cases} \tag{3.11}$$

με $T_{safe} = 297 \; K$, $T_{crit} = 300 \; K$, ως προς τις x και y συντεταγμένες σημείων ελέγχου NURBS στο σχ. 3.1α′, με τις διατυπώσεις *FI*, *E-SI* και (Severed) *SI* και συγκρίνονται με FDs. Περισσότερες λεπτομέρειες για τη συνάρτηση J^{penT} βρίσκονται στο πλήρες κείμενο της διατριβής. Όπως φαίνεται στο σχ. 3.1, οι διατυπώσεις *FI* και *E-SI* υπολογίζουν ακριβείς παραγώγους, σε αντίθεση με την *SI*.

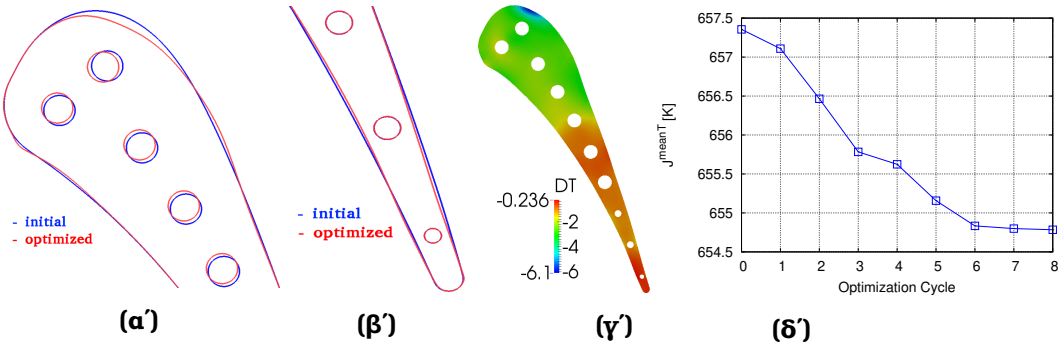
Για την ίδια γεωμετρία αλλά συνάρτηση-στόχο τη μέση θερμοκρασία του στερεού

$$J^{meanT} = \frac{\int_{\Omega^S} T^S d\Omega}{V^S} \tag{3.12}$$

υπολογίζονται παράγωγοι με τις διατυπώσεις *FI*, (Severed) *SI* και *E-SI*, σχ. 3.2. Και σε αυτήν την περίπτωση, φαίνονται τα πλεονεκτήματα από τη χρήση της διατύπωσης *E-SI*, η οποία με κόστος ίσο με το $2,67$ % της *FI* (υπολογισμοί σε 4 επεξεργαστές τύπου Intel(R) Xeon(R) CPU E5-2630 v2 @ 2.60GHz) υπολογίζει εξίσου ακριβείς παραγώγους. Αντίθετα, η διατύπωση *SI* οδηγεί σε παραγώγους αντίθετου προσήμου και μέτρου που διαφέρει κατά μια τάξη μεγέθους. Τα αποτελέσματα αυτά έρχονται σε πλήρη συμφωνία με αυτά που παρουσιάστηκαν στο κεφάλαιο 2 και στο [29, 27] για προβλήματα μόνο ροής. Επιπλέον, στο σχ. 3.2 φαίνεται ότι είτε η παραδοχή της 'παγωμένης τύρβης' είτε η επίλυση των εξισώσεων του συζυγούς μοντέλου τύρβης χωρίς ωστόσο να έχουν ληφθεί υπόψη οι συνεισφορές από την παραγώγιση της ενεργειακής εξίσωσης στο ρευστό έχουν σχεδόν την ίδια επίδραση στην ακρίβεια των παραγώγων.



Σχήμα 3.2: Αριστερά: Παράγωγοι της J^{meanT} υπολογισμένες με τις διατυπώσεις *FI*, *E-SI* και (Severed) *SI* συγκρίνονται με FDs. Δεξιά: Επίδραση της παραδοχής της 'παγωμένης τύρβης' (Fr. Turb.) και της χρήσης του συζυγούς μοντέλου τύρβης στις *E-SI* παραγώγους χωρίς να ληφθούν υπόψη οι συνεισφορές από την παραγώγιση της ενεργειακής εξίσωσης (Fr. En. Turb.).



Σχήμα 3.3: Βελτιστοποίηση 2Δ πτερυγίου σταθερής πτερύγωσης στροβίλου. (α) και (β) Σύγκριση αρχικής (μπλε) και βελτιστοποιημένης (κόκκινο) γεωμετρίας. (γ) Πεδίο διαφοράς θερμοκρασίας ανάμεσα σε αρχική και βελτιστοποιημένη γεωμετρία, σχεδιασμένο στη βελτιστοποιημένη γεωμετρία. (δ) Πορεία σύγκλισης βελτιστοποίησης.

Στο CHT πρόβλημα του σχ. 3.3 γίνεται ελαχιστοποίηση της J^{meanT} σε 2Δ πτερύγιο σταθερής πτερύγωσης στροβίλου, στο οποίο υπάρχουν 10 οπές εσωτερικής ψύξης (για το ψυκτικό μέσο $h_{coolant} = 2000\ J/m^2/sec/K$ και $T_{cool} = 432\ K$). Η ροή γύρω από το πτερύγιο είναι τυρβώδης ($Re = 53000$, $y^+ < 0.15$). Μεταβλητές σχεδιασμού είναι οι συντεταγμένες 62 σημείων ελέγχου NURBS που παραμετροποιούν το σχήμα του πτερυγίου και οι θέσεις των οπών ψύξης. Μετά από 8 κύκλους ψύξης, η μέση θερμοκρασία έχει ελαττωθεί κατά $2.7\ K$, σχ. 3.3δ΄. Όπως φαίνεται στο σχ. 3.3β΄, κατά την ελαχιστοποίηση της J^{meanT}, μειώθηκε η T^S σε κάθε σημείο του στερεού, σχ. 3.3γ΄, μειώνοντας της απόσταση μεταξύ των οπών ψύξης και του εξωτερικού τοιχώματος του πτερυγίου κοντά στην ακμή προσβολής και κοντά στην ακμή εκφυγής. Περισσότερες λεπτομέρειες για αυτό το πρόβλημα και άλλες εφαρμογές παρουσιάζονται στο πλήρες κείμενο της διατριβής.

Κεφάλαιο 4

3Δ Προβλήματα Βελτιστοποίησης Μορφής CHT

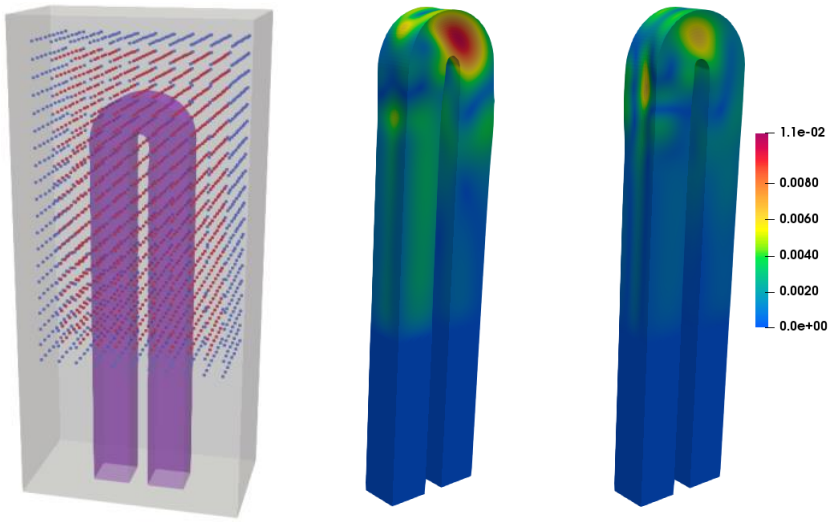
4.1 Αγωγός Ψύξης Σχήματος U

Σε αυτήν την εφαρμογή γίνεται η βελτιστοποίηση της ψύξης για το σύστημα εσωτερικής ψύξης που φαίνεται στο σχ. 4.1α΄. Στα τοιχώματα του στερεού που δεν είναι FSI, $T^S = 500\ K$, η ροή μέσα στον αγωγό είναι τυρβώδης ($Re = 84000$ υπολογισμένος με βάση την υδραυλική διάμετρο, μέσο $y^+ = 0.77$), με ταχύτητα στην είσοδο μέτρου $16.8 m/s$ και θερμοκρασία στην είσοδο $T^F\big|_{S_I^F} = 288\ K$. Επίσης, $c_p = 4181\ J/K/kg$, $Pr = 0.1$, $Pr_t = 1$ και $k^S = 60\ J/K/m/s$.

Πραγματοποιούνται δύο βελτιστοποιήσεις. Στη Βελτιστοποίηση 1, σκοπός είναι η ελαχιστοποίηση των υψηλότερων θερμοκρασιών που αναπτύσσονται στο στερεό. Αυτό επιτυγχάνεται μέσω της ελαχιστοποίησης της συνάρτησης-στόχου J^{penT}, εξ. 3.11, όπου $T_{safe} = 360\ K$ και $T_{crit} = 395\ K$. Στη Βελτιστοποίηση 2, στόχος είναι η μεγιστοποίηση της θερμορροής προς το ρευστό, η οποία δίνεται από τη σχέση

$$J^{HF} = \int_{\overline{S}^F} k^F \frac{\partial T^F}{\partial n} dS \tag{4.1}$$

Η παραμετροποίηση φαίνεται στο σχ. 4.1α΄, όπου οι x, y, και z συντεταγμένες των κόκκινων σημείων ελέγχου ογκομετρικών B-Splines είναι οι μεταβλητές σχεδιασμού. Στο τέλος της Βελτιστοποίησης 1, η J^{penT} έχει μειωθεί κατά $2.7\ \%$ και στο τέλος της Βελτιστοποίησης 2, η J^{HF} έχει αυξηθεί κατά $7.5\ \%$. Οι βελτιστοποιημένες γεωμετρίες συγκρίνονται στο σχ. 4.1. Και στις δύο περιπτώσεις, στη στροφή του αγωγού μειώνεται η διατομή του, ώστε να αυξηθεί η ταχύτητα και η συναγωγή της θερμοκρασίας, βελτιώνοντας με αυτόν τον τρόπο την ψύξη του στερεού.



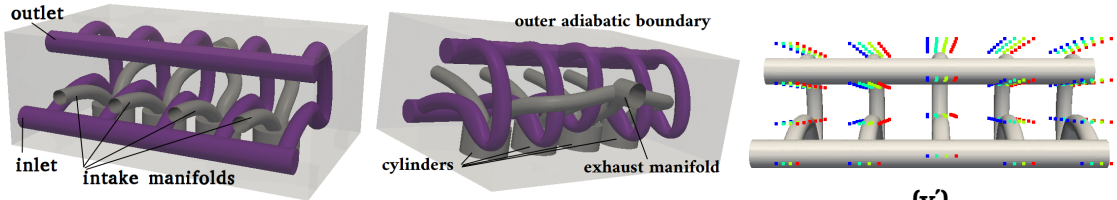
(α΄) Παραμετροποίηση **(β΄)** Βελτιστοποίηση 1 **(γ΄)** Βελτιστοποίηση 2

Σχήμα 4.1: Βελτιστοποίηση ψυκτικού καναλιού σχήματος U:(α) Παραμετροποίηση με πλέγμα $11 \times 20 \times 9$ σημείων ελέγχου ογκομετρικών B-Splines. (β) και (γ) Συνολική μετατόπιση του FSI για τις Βελτιστοποιήσεις 1 (β) και 2 (γ).

4.2 Κυλινδροκεφαλή Μηχανής Καύσης Αυτοκινήτου

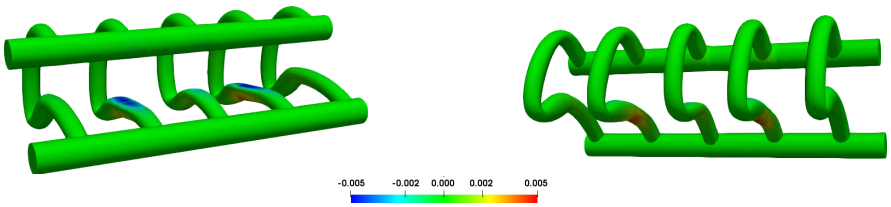
Σε αυτό το πρόβλημα διερευνάται η σχέση ανάμεσα στην ελαχιστοποίηση της μέγιστης θερμοκρασίας της κυλινδροκεφαλής μηχανής εσωτερικής καύσης (Βελτιστοποίηση 1, ελάχιστη τιμή της εξ. 3.11 με $T_{safe} = 410\ K$ και $T_{crit} = 480\ K$) και της ελαχιστοποίησης των απωλειών ολικής πίεσης J^{pt}, εξ. 2.9, (Βελτιστοποίηση 2) στο ψυκτικό ρευστό. Η γεωμετρία και η παραμετροποίηση παρουσιάζονται στο σχ. 4.2. Το εξωτερικό όριο της μηχανής είναι αδιαβατικό, οι κεφαλές των κυλίνδρων, οι αγωγοί εισαγωγής και εξαγωγής αέρα και καυσαερίων από τους κυλίνδρους παρέχουν σταθερή θερμορροή διαφορετικού μέτρου. Η ροή του ψυκτικού είναι τυρβώδης ($Re = 34000$), $c_p = 3554\ J/kg/K$, $k^S = 168\ J/m/sec/K$. Για κάθε συνάρτηση-στόχο πραγματοποιείται μια ξεχωριστή βελτιστοποίηση με μεταβλητές σχεδιασμού τις x, y, z συντεταγμένες των σημείων ελέγχου που φαίνονται στο σχ. 4.2.

Στο τέλος της Βελτιστοποίησης 1, η J^{penT} μειώθηκε κατά 15.9% και αν και δεν τέθηκε ως στόχος η J^{pt}, μειώθηκε και αυτή κατά 1.5%. Στη Βελτιστοποίηση 2, η J^{pt} μειώθηκε κατά 5.6% και η J^{penT} επίσης ελαττώθηκε κατά 0.8%, παρότι δεν τέθηκε ως στόχος. Φαίνεται ότι δεν υπάρχει σχεδιασμός της κυλινδροκεφαλής που να οδηγεί ταυτόχρονα στο ελάχιστο των J^{penT} και J^{pt}. Από το σχ. 4.3 παρατηρείται ότι για την ελαχιστοποίηση των μέγιστων θερμοκρασιών στο στερεό οι 5 κάθετοι αγωγοί ψύξης μετακινούνται προς τα κάτω, ώστε να βρίσκονται πιο κοντά στις κεφαλές των κυλίνδρων και να απάγουν περισσότερη θερμότητα. Για την ελαχιστοποίηση των απωλειών ολικής πίεσης, οι μεγαλύτερες αλλαγές παρατηρούνται στα σημεία ένωσης

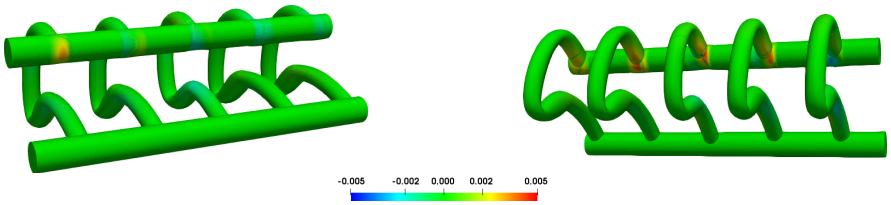


(α΄) **(β΄)** **(γ΄)**

Σχήμα 4.2: Βελτιστοποίηση κυλινδροκεφαλής: (α) και (β) Γεωμετρία. (γ) Παραμετροποίηση: Πλέγμα ελέγχου αποτελούμενο από $4 \times 4 \times 6$ σημεία ελέγχου ογκομετρικών B-Splines. Τα μπλε παραμένουν ακίνητα ενώ τα κόκκινα είναι ελεύθερα να μετακινηθούν.



(α΄) Βελτιστοποίηση 1



(β΄) Βελτιστοποίηση 2

Σχήμα 4.3: Βελτιστοποίηση κυλινδροκεφαλής: Όψη από μπροστά (αριστερά) και πίσω (δεξιά) των βελτιστοποιημένων καναλιών ψύξης μαζί με το πεδίο της συνολικής μετατόπισης, η οποία είναι προβεβλημένη στη κάθετη ως προς την επιφάνεια διεύθυνση. Οι χρωματισμένες με μπλε και κόκκινο περιοχές υποδηλώνουν μετακίνηση προς τα μέσα και προς τα έξω ως προς το χωρίο του ρευστού αντίστοιχα.

των πέντε κάθετων αγωγών και του πάνω οριζόντιου αγωγού.

Κεφάλαιο 5

Επίδραση του Είδους του Μοντέλου Μετατόπισης Πλέγματος στις Παραγώγους για τη Βελτιστοποίηση Μορφής

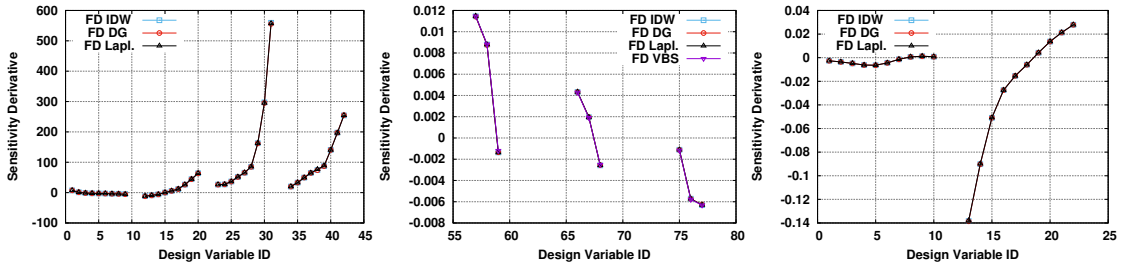
Η μετατόπιση του πλέγματος έχει φανεί ότι είναι σημαντικό να λαμβάνεται υπόψη προκειμένου να υπολογίζονται ακριβείς παράγωγοι, τόσο στη συνεχή [29, 19] όσο και τη διακριτή [49] συζυγή μέθοδο. Σε αυτό το κεφάλαιο το ενδιαφέρον εστιάζεται στο αν το είδος του μοντέλου μετατόπισης πλέγματος (Grid Displacement Model, GDM) επιδρά στην ακρίβεια των παραγώγων. Προκειμένου να εξασφαλιστεί η γενικότητα των αποτελεσμάτων, γίνεται μελέτη σε προβλήματα διαφορετικής φυσικής, συγκεκριμένα σε εξωτερικής, εσωτερικής αεροδυναμικής και CHT. Χρησιμοποιούνται τέσσερα διαφορετικά μοντέλα μετατόπισης, εξ. Laplace [25], Γράφοι Delaunay [32], ογκομετρικές B-Splines [41] και η μέθοδος Inverse Distance Weighting (IDW) [35]. Λεπτομέρειες για καθένα από αυτά παρατίθενται στο πλήρες κείμενο της διατριβής. Η διερεύνηση σε αυτό το κεφάλαιο γίνεται σε τέσσερα βήματα:

1. Υπολογίζονται παράγωγοι αναφοράς με FDs για τα διαφορετικά GDMs.

2. Υπολογίζονται παράγωγοι με τη συζυγή διατύπωση *E-SI*, στην οποία αναπτύσσονται συζυγείς εξισώσεις ως προς το GDM, κάνοντας την υπόθεση ότι αυτό βασίζεται σε επίλυση εξισώσεων Laplace.

3. Γίνεται ανάλυση των όρων που αποτελούν τις παραγώγους που υπολογίζονται από τη διατύπωση *FI*. Σε αυτές το είδος του GDM λαμβάνεται υπόψη μέσω των παραγώγων ευαισθησίας πλέγματος που περιέχονται μέσα σε χωρικά ολοκληρώματα. Μέσω αυτού του βήματος δικαιολογούνται τα ευρήματα των δύο παραπάνω βημάτων.

4. Διερευνάται η επίπτωση από τη χρήση διαφορετικών GDMs για τη μετατόπιση πλέγματος και τον υπολογισμό συζυγών παραγώγων στα αποτελέσματα της βελτιστοποίησης μορφής.



(α΄) Πρόβλημα 1 **(β΄)** Πρόβλημα 2 **(γ΄)** Πρόβλημα 3

Σχήμα 5.1: Σύγκριση FDs για διαφορετικά GDMs στα προβλήματα Προβλήματα 1-3.

5.1 Σύγκριση Παραγώγων για Διαφορετικά GDMs

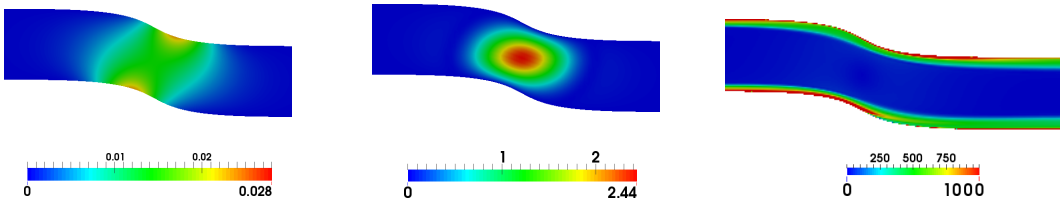
Στο σχ. 5.1 διαπιστώνεται ότι παράγωγοι που υπολογίζονται με FDs για διαφορετικά GDMs στα Προβλήματα 1-3 έχουν μεταξύ του αμελητέες διαφορές.

Στα κεφάλαια 2 και 3 πιστοποιήθηκε η ακρίβεια των παραγώγων που υπολογίζονται με τις διατυπώσεις *E-SI* και *FI*. Εφόσον οι παράγωγοι με FDs δεν διαφέρουν για τα διαφορετικά GDMs, αυτό σημαίνει ότι το είδος του GDM δεν επιδρά στην ακρίβεια των παραγώγων. Επομένως, η διατύπωση *E-SI*, στην οποία γίνεται η παραδοχή ότι εξισώσεις Laplace διέπουν τη μετατόπιση του πλέγματος, δεν χρειάζεται να αναπτυχθεί εκ νέου για κάθε διαφορετικό GDM. Στο πλήρες κείμενο της διατριβής παρατίθενται περισσότερες συγκρίσεις παραγώγων που υπολογίζονται με τη διατύπωση *FI* και FDs για διαφορετικά GDMs.

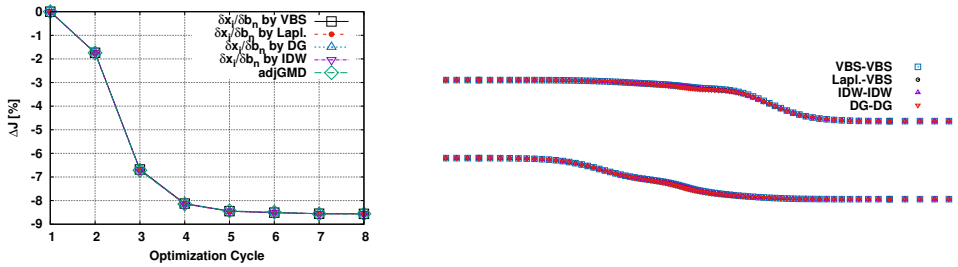
Προκειμένου να διερευνηθούν οι λόγοι για τους οποίους το είδος του GDM δεν επιδρά στις παραγώγους, γίνεται όρο-προς-όρο ανάλυση των παραγώγων που υπολογίζονται από τη διατύπωση *FI*, παρά το υψηλότερο κόστος σε σχέση με τη διατύπωση *E-SI*, γιατί δεν γίνεται η παραδοχή ότι εξισώσεις Laplace διέπουν τη μετατόπιση του πλέγματος. Αντιθέτως, οι παράγωγοι της διατύπωσης *FI*, που δίνονται συνοπτικά από τη σχέση

$$\left.\frac{\delta J}{\delta b_n}\right|_{\mathrm{FI}} = \underbrace{\int_\Omega \mathcal{F}\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)d\Omega}_{W} + \int_S \cdots dS \tag{5.1}$$

εξαρτώνται από το είδος του GDM, γιατί περιλαμβάνουν χωρικά ολοκληρώματα (όρος W) τα οποία περιέχουν τις παραγώγους ευαισθησίας πλέγματος. Στην παραπάνω σχέση, ο όρος $\mathcal{F}$ εξαρτάται από τα πρωτεύοντα και συζυγή ροϊκά μεγέθη και το επιφανειακό ολοκλήρωμα στο δεξί μέλος της σχέσης δεν επηρεάζεται από το είδος του GDM. Η ανάλυση εστιάζει στον όρο W και παρουσιάζεται ενδεικτικά για το Πρόβλημα 2, με τα υπόλοιπα δύο προβλήματα να αναλύονται με μεγάλη λεπτομέρεια στο πλήρες κείμενο. Στο σχ. 5.2 φαίνεται ότι ενώ κοντά στον τοίχο οι παράγωγοι ευαισθησίας πλέγματος διαφέρουν ελάχιστα, μακριά από αυτόν οι διαφορές αυξάνονται. Ωστόσο, μόνο κοντά στον τοίχο το μέτρο του $\mathcal{F}$ είναι μεγάλο, ενώ αυξάνοντας της



Σχήμα 5.2: Αριστερά: Μέτρο του $\delta \mathbf{x}/\delta b_n$ υπολογισμένο με βάση το Laplace GDM για το 5ο σημείο ελέγχου. Κέντρο: Διαφορά των μέτρων του $\delta \mathbf{x}/\delta b_n$ υπολογισμένο για τα Laplace και VBS GDMs, αδιαστοποιημένο με τη μέγιστη τιμή του μέτρου του $\delta \mathbf{x}/\delta b_n$ του τοιχώματος. Δεξιά: Μέτρο του $\mathcal{F}$.



Σχήμα 5.3: Αριστερά: Πορεία σύγκλισης της J^{pt}, λαμβάνοντας υπόψη διαφορετικά GDMs για τον υπολογισμό παραγώγων, ενώ η μετατόπιση πλέγματος γίνεται με χρήση ογκομετρικών B-Splines. Αντίστοιχα σχήματα για άλλα μοντέλα μετατόπισης βρίσκονται στο πλήρες κείμενο της διατριβής. Δεξιά: Βελτιστοποιημένες γεωμετρίες. Το πρώτο συνθετικό δηλώνει το GDM που χρησιμοποιήθηκε για τον υπολογισμό των παραγώγων και το 2ο το GDM για τη μετατόπιση του πλέγματος.

απόσταση από τον τοίχο το μέτρο του $\mathcal{F}$ μειώνεται. Συνεπώς, οι διαφορές ανάμεσα στις παραγώγους ευαισθησίας πλέγματος μακριά από τον τοίχο δεν οδηγούν σε διαφορετικές παραγώγους, επειδή εκεί μηδενίζεται ο όρος $\mathcal{F}$ που πολλαπλασιάζει τα διαφορετικού μέτρου $\delta x_k/\delta b_n$. Περισσότερες συγκρίσεις και λεπτομέρειες στο πλήρες κείμενο της διατριβής.

5.2 Επίδραση του είδους του GDM στη Βελτιστοποίηση

Ενδεικτικά για το Πρόβλημα 2 πραγματοποιούνται βελτιστοποιήσεις με διαφορετικά GDMs για τον υπολογισμό των παραγώγων και τη μετατόπιση πλέγματος. Χρησιμοποιείται η μέθοδος damped BFGS για την ανανέωση των μεταβλητών σχεδιασμού, μιας και η μέθοδος αυτή σε σύγκριση με άλλες είναι πιο ευαίσθητη στην ακρίβεια των παραγώγων [45]. Στο σχ. 5.3 διαπιστώνεται η απουσία διαφορών τόσο στην πορεία σύγκλισης της J^{pt} όσο και στις βελτιστοποιημένες γεωμετρίες.

Κεφάλαιο 6

Η Συζυγής Μέθοδος για τον Υπολογισμό Μετώπου Pareto στη Βελτιστοποίηση Μορφής

Σε αυτό το κεφάλαιο γίνεται χρήση της συνεχούς συζυγούς μεθόδου για την αντιμετώπιση προβλημάτων βελτιστοποίησης αντικρουόμενων στόχων και τον υπολογισμού του μετώπου μη κυριαρχούμενων λύσεων (μέτωπο Pareto). Αναπτύσσονται παραλλαγές αλγορίθμου πρόβλεψης-διόρθωσης βασιζόμενου στην κλίση της συνάρτησης-στόχου, ο οποίος παρουσιάστηκε για πρώτη φορά στο [47] και διερευνήθηκε περαιτέρω στο [26]. Χαρακτηριστικό του αλγορίθμου είναι ο έλεγχος της απόστασης μεταξύ των σημείων του μετώπου Pareto (Pareto Point, PP). Η μετάβαση σε επόμενο PP ξεκινά από ένα ήδη υπολογισμένο PP. Για όλες τις συναρτήσεις-στόχους πλην μίας, τίθενται τιμές ως στόχοι που αντιστοιχούν στο επόμενο PP, διαμορφώνοντας έτσι περιορισμούς ισότητας. Στη συνέχεια, επιλύεται πρόβλημα βελτιστοποίησης με περιορισμούς για τη μετάβαση στο επόμενο PP. Εφόσον εντοπιστεί το νέο PP, οι στόχοι στους περιορισμούς ισότητας ανανεώνονται και ξεκινά η μετάβαση στο επόμενο PP. Για προβλήματα δύο στόχων J_1 και J_2 και ενός γεωμετρικού περιορισμού c_g ορίζεται η συνάρτηση Lagrangian ως

$$L\left(b_n, \lambda_1, \lambda_2, \hat{J}_2\right) = J_1\left(b_n\right) - \lambda_1 c_f\left(b_n, \hat{J}_2\right) - \lambda_2 c_g\left(b_n\right) \tag{6.1}$$

όπου $c_f = J_2 - \hat{J}_2$, με $\hat{J}_2$ να είναι η τιμή στόχος που αντιστοιχεί στη J_2 για το επόμενο PP και λ_1, λ_2 είναι πολλαπλασιαστές Lagrange. Στο βήμα πρόβλεψης, που χρησιμοποιείται για να μειωθεί το κόστος υπολογισμό του μετώπου Pareto, επιλύεται το

παρακάτω σύστημα εξισώσεων

$$\begin{bmatrix} \boldsymbol{\mathcal{H}} & -\frac{\delta c_f}{\delta \mathbf{b}}^T & -\frac{\delta c_g}{\delta \mathbf{b}}^T \\ \frac{\delta c_f}{\delta \mathbf{b}} & 0 & 0 \\ \frac{\delta c_g}{\delta \mathbf{b}} & 0 & 0 \end{bmatrix}_{old} \begin{bmatrix} \frac{\delta \mathbf{b}}{\delta \hat{J}_2} \\ \frac{\delta \lambda_1}{\delta \hat{J}_2} \\ \frac{\delta \lambda_2}{\delta \hat{J}_2} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \\ 0 \end{bmatrix} \tag{6.2}$$

με τα $\mathbf{b}$ και $\boldsymbol{\lambda}$ να ανανεώνονται μέσω των σχέσεων $\mathbf{b}^{new} = \mathbf{b}^{old} + \alpha \frac{\delta \mathbf{b}}{\delta \hat{J}_2} \left(\hat{J}_2^{\ new} - \hat{J}_2^{\ old} \right)$, $\boldsymbol{\lambda}^{new} = \boldsymbol{\lambda}^{old} + \frac{\delta \boldsymbol{\lambda}}{\delta \hat{J}_2} \left(\hat{J}_2^{\ new} - \hat{J}_2^{\ old} \right)$. Στις παραπάνω σχέσεις, οι δείκτες old και new δηλώνουν το τρέχον και νέο PP, αντίστοιχα. Επίσης, $\boldsymbol{\mathcal{H}}$ είναι το Εσσιανό Μητρώο της L ως προς τα $\mathbf{b}$ και α ο πολλαπλασιαστής του $\Delta \boldsymbol{b}$. Στο βήμα διόρθωσης χρησιμοποιείται η μέθοδος SQP, στην οποία επιλύεται το σύστημα

$$\begin{bmatrix} \boldsymbol{\mathcal{H}} & -\frac{\delta c_f}{\delta \mathbf{b}}^T & -\frac{\delta c_g}{\delta \mathbf{b}}^T \\ \frac{\delta c_f}{\delta \mathbf{b}} & 0 & 0 \\ \frac{\delta c_g}{\delta \mathbf{b}} & 0 & 0 \end{bmatrix}_{old} \begin{bmatrix} \boldsymbol{\Delta b} \\ \Delta \lambda_1 \\ \Delta \lambda_2 \end{bmatrix} = - \begin{bmatrix} \frac{\delta L}{\delta \mathbf{b}} \\ c_f \\ c_g \end{bmatrix}_{old} \tag{6.3}$$

προκειμένου να ανανεωθούν τα $\mathbf{b}$ και $\boldsymbol{\lambda}$ ως $\mathbf{b}^{new} = \mathbf{b}^{old} + \alpha \boldsymbol{\Delta} \mathbf{b}$, $\boldsymbol{\lambda}^{new} = \boldsymbol{\lambda}^{old} + \boldsymbol{\Delta \lambda}$. Σε αυτήν τη διατριβή, αναπτύσσονται τρεις παραλλαγές της μεθόδου πρόβλεψης-διόρθωσης, στις οποίες κοινό στοιχείο είναι η αποφυγή του υπολογισμού του ακριβούς Εσσιανού, λόγω του υψηλού υπολογιστικού κόστους:

1. Στην Παραλλαγή 1 (Variant 1) το σύστημα του βήματος πρόβλεψης επιλύεται με χρήση της μεθόδου GMRES [46], στην οποία τα απαραίτητα γινόμενα Εσσιανού-διανύσματος υπολογίζονται με εφαρμογή της μεθόδου ευθείας διαφόρισης στις πρωτεύουσες και συζυγείς εξισώσεις [15, 39]. Περισσότερες λεπτομέρειες παρατίθενται στο πλήρες κείμενο της διατριβής. Στο βήμα διόρθωσης, το Εσσιανό προσεγγίζεται με τη μέθοδο damped BFGS (DBFGS) [38].

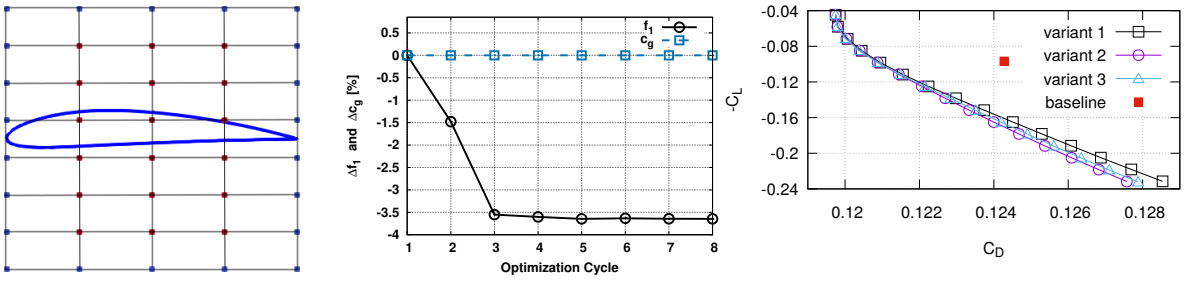
2. Στην Παραλλαγή 2 (Variant 2) η διαφορά με την Παραλλαγή 1 είναι ότι το Εσσιανό μητρώο στο αριστερό μέλος του συστήματος πρόβλεψης προσεγγίζεται με τη μέθοδο DBFGS.

3. Στην Παραλλαγή 3 (Variant 2) το βήμα πρόβλεψης παραλείπεται και η ανανέωση των $\mathbf{b}$, $\boldsymbol{\lambda}$ γίνεται μέσω της μεθόδου SQP. Με αυτόν τον τρόπο διερευνάται αν επιβραδύνεται ο υπολογισμός του μετώπου Pareto, παραλείποντας το βήμα πρόβλεψης.

Περισσότερες λεπτομέρειες για τις τρεις παραλλαγές παρατίθενται στο πλήρες κείμενο της διατριβής.

6.1 Εφαρμογή στην Αεροτομή NACA4412

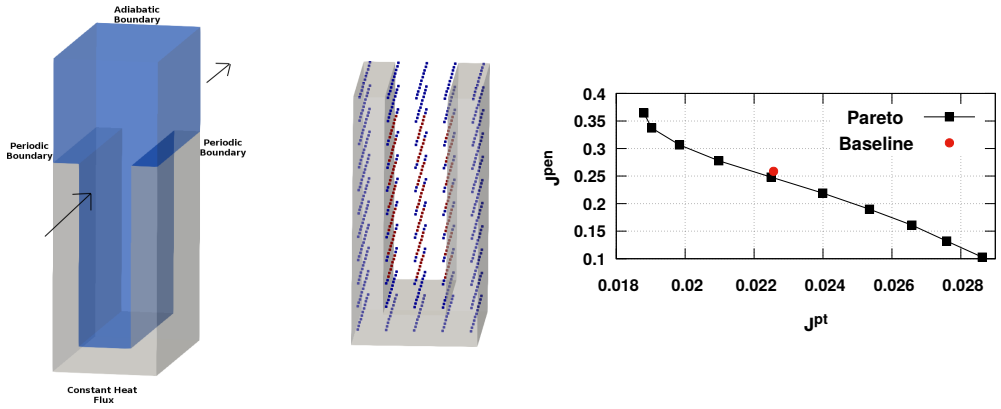
Οι τρεις παραλλαγές της μεθόδου πρόβλεψης-διόρθωσης συγκρίνονται μεταξύ τους κατά τον υπολογισμό του μετώπου Pareto στην αεροτομή NACA4412. Αντικρουόμενοι στόχοι είναι η ελαχιστοποίηση της αεροδυναμικής αντίστασης και η μεγιστοποίηση της άνωσης, με περιορισμό σταθερού εμβαδού που περικλείει η αεροτομή. Ο αριθμός Reynolds της ροής είναι $Re = 1000$ και η γωνία της επ' άπειρον ροής είναι $\alpha_\infty = 1^0$. Κάθε φορά που εντοπίζεται ένα νέο PP, η τιμή του $\hat{J}_2$ αλλάζει κατά 30%. Περισσότερες πληροφορίες παρατίθενται στο πλήρες κείμενο της διατριβής. Στο σχ. 6.1 παρουσιάζονται η παραμετροποίηση, η βελτιστοποίηση για το 1ο PP, στην οποία ελαχιστοποιήθηκε η αεροδυναμική αντίσταση με περιορισμό στην επιφάνεια που περικλείει η αεροτομή και το μέτωπο Pareto. Η Παραλλαγή 2 ήταν αυτή με το μικρότερο κόστος (50 EFS), ενώ η Παραλλαγή 1 είχε το υψηλότερο (71 EFS), λόγω της επίλυσης επιπλέον εξισώσεων για τον υπολογισμό των γινομένων Εσσιανού-διανύσματος. Η Παραλλαγή 3 (61 EFS) είχε υψηλότερο κόστος από την 2, κάτι που δείχνει πώς το βήμα πρόβλεψης όντως μειώνει το κόστος υπολογισμού του μετώπου Pareto. Στο πλήρες κείμενο της διατριβής η μελέτη αυτή επαναλαμβάνεται για την αεροτομή NACA0012. Επιπλέον, περιλαμβάνονται περισσότερες λεπτομέρειες όσον αφορά στη σύγκριση των τριών παραλλαγών. Τέλος, η Παραλλαγή 2, αυτή με το μικρότερο υπολογιστικό κόστος, φαίνεται ότι έχει χαμηλότερο κόστος και από τον υπολογισμό του μετώπου Pareto με χρήση χρήση βαρών για τις αντικρουόμενες συναρτήσεις-στόχους, όπως φαίνεται από διερευνήσεις που παρουσιάζονται στο πλήρες κείμενο της διατριβής.



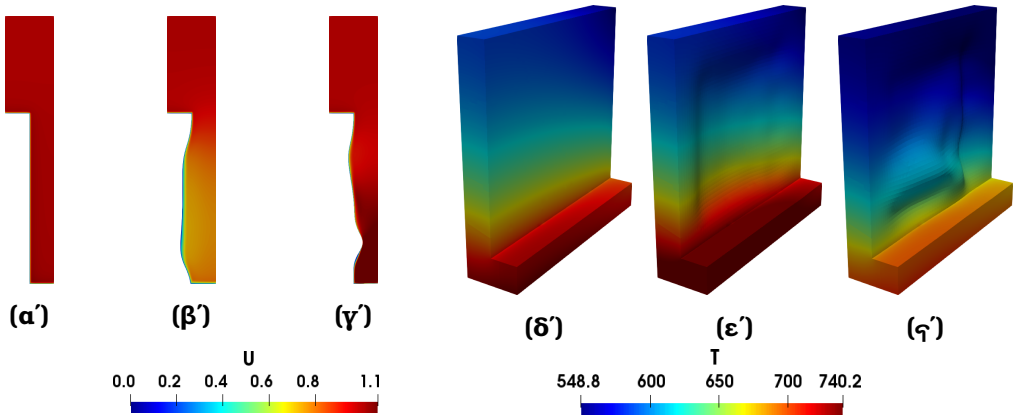
Σχήμα 6.1: Αριστερά: Παραμετροποίηση βασισμένη στις ογκομετρικές B-Splines. Τα κόκκινα σημεία ελέγχου είναι ελεύθερα να μετακινηθούν ενώ τα μπλε παραμένουν ακίνητα. Κέντρο: Σύγκλιση της βελτιστοποίησης για το 1ο PP. Δεξιά: Μέτωπο Pareto.

6.2 Εφαρμογή σε 3Δ Πρόβλημα CHT

Η Παραλλαγή 2 της μεθόδου πρόβλεψης-διόρθωσης χρησιμοποιείται για τον υπολογισμό του μετώπου Pareto σε ένα 3Δ πρόβλημα CHT που παρουσιάζεται στο σχ. 6.2.



Σχήμα 6.2: Αριστερά: Υπολογιστικά χωρία που αντιστοιχούν στο ρευστό (γαλάζιο) και στερεό (γκρι). Κέντρο: Παραμετροποίηση βασισμένη σε ογκομετρικές B-Splines. Τα κόκκινα σημεία ελέγχου μετακινούνται ελεύθερα κατά τη βελτιστοποίηση ενώ τα μπλε παραμένουν ακίνητα. Δεξιά: Μέτωπο Pareto.



Σχήμα 6.3: Μέτρο ταχύτητας στο ρευστό (α-γ) και θερμοκρασία στο στερεό (δ-ς), για την αρχική γεωμετρία (α,δ) και τις γεωμετρίες για ελάχιστες J^{pt} (β,ε) και J^{penT} (γ,ς).

Αντικρουόμενοι στόχοι είναι η ελαχιστοποίηση της J^{pt}, εξ. 2.9, στο ρευστό που ρέει ανάμεσα στα ψυκτικά πτερύγια και η ελαχιστοποίηση της J^{penT}, εξ. 3.11 με $T_{safe} = 650\ K$ και $T_{crit} = 730\ K$, στο στερεό. Η ροή είναι τυρβώδης $Re = 2.83 \times 10^5$, $y^+ < 0.6$, $Pr = 0.7$, $Pr_t = 1$, $c_p = 1006 J/kg/K$ και $k^S = 60 J/m/s/K$. Στην κάτω επιφάνεια του στερεού παρέχεται σταθερή θερμορροή $Q = 4800\ W/m^2$, τα πλευρικά όρια των Ω^F και Ω^S είναι περιοδικά και το πάνω όριο του ρευστού είναι αδιαβατικό.

Το 1ο PP υπολογίζεται ελαχιστοποιώντας την J^{pt}, η οποία μειώνεται κατά $\sim 18\%$ μέσα σε 12 κύκλους βελτιστοποίησης. Το μέτωπο Pareto, που παρουσιάζεται στο σχ. 6.2, υπολογίστηκε με κόστος ίσο με 120 EFS. Για να ελαχιστοποιηθεί η J^{pt}, η διατομή ανάμεσα στα πτερύγια ψύξης αυξήθηκε, με αποτέλεσμα να ελαττωθεί η συναγωγή της θερμοκρασίας στο ρευστό και να αυξηθεί συνεπώς η θερμοκρασία στο στερεό. Όπως φαίνεται στο σχ. 6.3, το αντίθετο συνέβη ώστε να ελαττωθεί η J^{penT}.

Κεφάλαιο 7

Επεμβατική Μέθοδος Αναπτύγματος Πολυωνυμικού Χάους για Προβλήματα CHT με Αβεβαιότητες

Σε αυτό το κεφάλαιο, η επεμβατική μέθοδος Αναπτύγματος Πολυωνυμικού Χάους (intrusive Polynomial Chaos Expansion, iPCE) αναπτύσσεται σε προβλήματα CHT με αβεβαιότητες, με σκοπό τον υπολογισμό στατιστικών ροπών (Uncertainty Quantification, UQ) της ποσότητας ενδιαφέροντος (Quantity of Interest, QoI), π.χ. τη μέση θερμοκρασία του στερεού J^{meanT}, τη θερμορροή στο FSI J^{HF} κ.ο.κ. και τη Βελτιστοποίηση με Αβεβαιότητες (Optimization under Uncertainties, OuU) με χαμηλό υπολογιστικό κόστος. Οι μεταβλητές των πρωτευουσών και συζυγών εξισώσεων χωρίς αβεβαιότητες αναπτύσσονται σε αθροίσματα ορθογωνίων πολυωνύμων με άγνωστους συντελεστές, μέσω της σχέσης

$$f = \sum_{m=0}^{M} f_m \psi_m(\boldsymbol{\xi}) \tag{7.1}$$

όπου f μια οποιαδήποτε συνάρτηση, $\boldsymbol{\xi}$ το διάνυσμα των N_{UV} αβέβαιων μεταβλητών, που σε αυτήν τη διατριβή είναι ανεξάρτητες μεταξύ τους και ακολουθούν κανονική κατανομή. Επίσης, $\psi_m(\boldsymbol{\xi})$ είναι τα πολυδιάστατα ορθογώνια πολυώνυμα Hermite, ενώ

$$M+1 = \binom{N_{UV}+N_{ch}}{N_{UV}} = \frac{(N_{UV}+N_{ch})!}{N_{UV}!N_{ch}!} \tag{7.2}$$

όπου N_{ch} είναι η τάξη χάους που καθορίζεται από το χρήστη.

Εφαρμόζοντας την εξ. 7.1 στις εξ. 2.1α′, 2.1β′, 3.1 και 3.2 για $N_{UV} = N_{ch} = 2$ και χρησιμοποιώντας προβολές Galerkin, προκύπτουν τα iPCE πρωτεύοντα και συζυγή συστήματα με 30 εξισώσεις για 30 άγνωστα πεδία iPCE συντελεστών (για 2Δ προβλήματα, 36 για 3Δ). Ενδεικτικά, οι εξισώσεις του πρωτεύοντος προβλήματος που

προκύπτουν μετά από προβολές Galerkin στο πολυώνυμο $\psi_0(\boldsymbol{\xi})=1$ είναι οι εξής

$$
\begin{aligned}
{}^0R^p &= -\frac{\partial v_{j,0}}{\partial x_j} = 0 && \text{(7.3α')}\\
{}^0R_i^v &= v_{j,0}\frac{\partial v_{i,0}}{\partial x_j} + v_{j,1}\frac{\partial v_{i,1}}{\partial x_j} + v_{j,2}\frac{\partial v_{i,2}}{\partial x_j} + v_{j,3}\frac{\partial v_{i,3}}{\partial x_j} + 2v_{j,4}\frac{\partial v_{i,4}}{\partial x_j} + 2v_{j,5}\frac{\partial v_{i,5}}{\partial x_j} + \frac{\partial p_0}{\partial x_j}\\
&\quad - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_{i,0}}{\partial x_j} + \frac{\partial v_{j,0}}{\partial x_i}\right)\right] = 0 && \text{(7.3β')}\\
{}^0R^{T^F} &= \rho^F c_p\left[v_{j,0}\frac{\partial T_0^F}{\partial x_j} + v_{j,1}\frac{\partial T_1^F}{\partial x_j} + v_{j,2}\frac{\partial T_2^F}{\partial x_j} + v_{j,3}\frac{\partial T_3^F}{\partial x_j} + 2v_{j,4}\frac{\partial T_4^F}{\partial x_j} + 2v_{j,5}\frac{\partial T_5^F}{\partial x_j}\right]\\
&\quad - \frac{\partial}{\partial x_j}\left(k^F\frac{\partial T_0^F}{\partial x_j}\right) = 0 && \text{(7.3γ')}\\
{}^0R^{T^S} &= -\frac{\partial}{\partial x_j}\left(k^S\frac{\partial T_0^S}{\partial x_j}\right) = 0 && \text{(7.3δ')}
\end{aligned}
$$

Μειονέκτημα της μεθόδου iPCE είναι ότι κάθε φορά που αλλάζουν τα N_{UV} ή/και N_{ch}, πρέπει να γίνει η μαθηματική ανάπτυξη των iPCE εξισώσεων και ο προγραμματισμός τους από την αρχή. Το κόστος επίλυσης των iPCE εξισώσεων είτε για το πρωτεύον είτε το συζυγές πρόβλημα είναι ίσο με $M = 6$ EFS. Οι μαθηματικές εκφράσεις για τη μέση τιμή και η τυπική απόκλιση της QoI δίνονται από την εξίσωση της ν-ιοστής στατιστικής ροπής της J

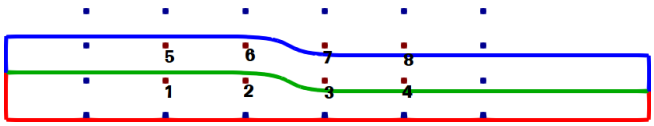
$$\langle J^\nu\rangle = \int_{\boldsymbol{\mathcal{D}}} J^\nu W(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{7.4}$$

θέτοντας $\nu=1$ και $\nu=2$, αντίστοιχα.

7.1 Η μη-Επεμβατική Μέθοδος Αναπτύγματος Πολυωνυμικού Χάους

Η μη-επεμβατική Μέθοδος Αναπτύγματος Πολυωνυμικού Χάους (non intrusive Polynomial Chaos Expansion, niPCE) χρησιμοποιείται σε αυτήν τη διατριβή για την πιστοποίηση της μεθόδου iPCE. Η βασική διαφορά με τη μέθοδο iPCE είναι ότι η εξ. 7.1 εφαρμόζεται στην QoI και όχι στα πρωτεύοντα και συζυγή πεδία. Αποδεικνύεται ότι η μέση τιμή και η τυπική απόκλιση υπολογίζονται από τις σχέσεις

$$\mu_J = J_0 \tag{7.5α'}$$



Σχήμα 7.1: Γεωμετρία και παραμετροποίηση. Τα κόκκινα σημεία ελέγχου μετακινούνται ελεύθερα ενώ τα μπλε παραμένουν ακίνητα.

$$\sigma_J = \sqrt{\sum_{m=1}^{M} J_m^2} \tag{7.56'}$$

Οι συντελεστές J_m υπολογίζονται μέσω της ολοκλήρωσης με τη μέθοδο Gauss Quadrature [31] από τη σχέση

$$J_m = \sum_{i=1}^{N_{GQ}} J\big|_{\mathbf{z}^{(i)}} \widetilde{\psi}_m\left(\mathbf{z}^{(i)}\right) \prod_{l=0}^{N_{UV}} w_l^{(i)} \tag{7.6}$$

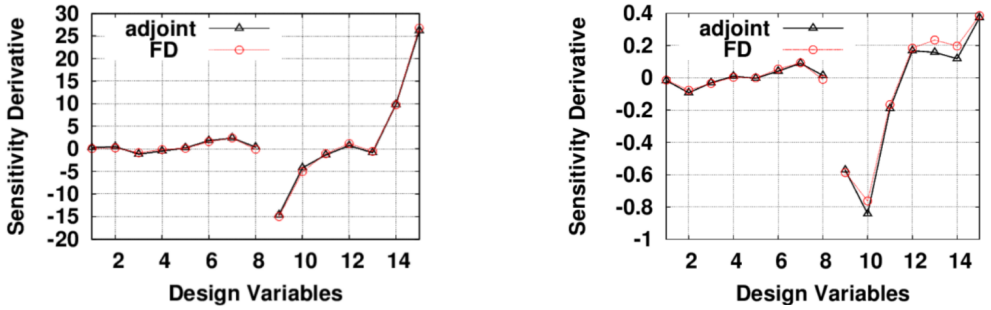
όπου $\widetilde{\psi}_m$ τα ορθοκανονικά πολυδιάστατα πολυώνυμα Hermite, $z^{(i)} = \left(z_1^{(i)}, ..., z_{N_{UV}}^{(i)}\right)$, $j = 1, ..., N_{GQ}$ ο i-οστός συνδυασμός των ριζών $z_l^{(i)}$ των μονοδιάστατων πολυωνύμων Hermite βαθμού l ή αλλιώς ο i-οστός κόμβος ολοκλήρωσης Gauss και $w_l^{(i)}$ τα βάρη της ολοκλήρωσης κατά Gauss Quadrature.

Για τον υπολογισμό των J_m απαιτείται η κλήση του λογισμικού επίλυσης των εξισώσεων CHT χωρίς αβεβαιότητες $N_{GQ} = (N_{ch} + 1)^{N_{UV}}$ φορές. Συνεπώς, το κόστος της μεθόδου niPCE είναι μεγαλύτερο από της μεθόδου iPCE (ίσο με M EFS), μιας και για $N_{ch} = N_{UV} = 2$, $N_{GQ} = 9$ ενώ $M = 6$. Ωστόσο, το πλεονέκτημα της μεθόδου niPCE είναι ότι δεν απαιτείται μαθηματική ανάπτυξη και προγραμματισμός νέων εξισώσεων, όπως στη μέθοδο iPCE.

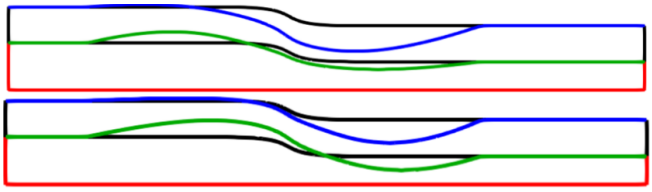
7.2 Εφαρμογή

Για το CHT πρόβλημα που παρουσιάζεται στο σχ. 7.1 πιστοποιείται η ακρίβεια υπολογισμού της μέσης τιμής και της τυπικής απόκλισης της μέσης θερμοκρασίας στο στερεό J^{meanT}, εξ. 3.12, καθώς και των παραγώγων τους, με τη μέθοδο iPCE. Στη συνέχεια πραγματοποιούνται βελτιστοποιήσεις. Αβέβαιες μεταβλητές είναι το μέτρο της ταχύτητας και η θερμοκρασία στην είσοδο του Ω^F, με μέση τιμή $1\ m/s$ και $300\ K$ και τυπική απόκλιση $0.1\ m/s$ και $20\ K$ αντίστοιχα. Θερμορροή σταθερού μέτρου $Q = 4800\ W/m^2$ παρέχεται στο Ω^S μέσω του κάτω τοιχώματός του, ενώ τα υπόλοιπα τοιχώματα του Ω^S και το άνω τοίχωμα του Ω^F είναι αδιαβατικά. Επίσης, $c_p = 1006\ J/kg/K$, $\rho^F = 1.21\ kg/m^3$, $\mathrm{Pr} = 0.7$ και $k^S = 60\ J/m/s/K$.

Στον πίνακα 7.1 φαίνεται ότι η μέθοδος iPCE για $N_{ch} = 2$ χαρακτηρίζεται από



Σχήμα 7.2: Πιστοποίηση συζυγών παραγώγων της μέσης τιμής (αριστερά) και τυπικής απόκλισης (δεξιά) της J^{meanT} με FDs.



Σχήμα 7.3: Γεωμετρίες για ελαχιστοποιημένη μέση τιμή (πάνω) και τυπική απόκλιση (κάτω) της J^{meanT}. Η αρχική γεωμετρία φαίνεται με μαύρο χρώμα.

υψηλή ακρίβεια όσον αφορά στον υπολογισμό των στατιστικών ροπών, ενώ έχει πολύ μικρότερο κόστος από τς μεθόδους Monte Carlo και niPCE. Στο σχ. 7.2 φαίνεται ότι

Μέθοδος	**Μέση Τιμή**	**Τυπική Απόκλιση**	**EFS**
Monte Carlo	770.7	27.88	1000
niPCE	772.1	27.83	9
iPCE	771.5	27.80	6

Πίνακας 7.1: Μέση τιμή και τυπική απόκλιση της J^{meanT}, υπολογισμένες με τις μεθόδους iPCE, niPCE και Monte Carlo και υπολογιστικό κόστος.

η μέθοδος iPCE στις συζυγείς εξισώσεις οδηγεί σε ακριβείς παραγώγους της μέσης τιμής και τυπικής απόκλισης της J^{meanT}.

Στο σχ. 7.3 φαίνονται τα αποτελέσματα από δύο διαφορετικές βελτιστοποιήσεις, μία με στόχο την ελάχιστη μέση τιμή της J^{meanT} και μια δεύτερη με στόχο την ελάχιστη τυπική απόκλιση της ίδιας QoI. Στην πρώτη βελτιστοποίηση, η μέση τιμή της J^{meanT} μειώθηκε κατά $12.6\ \%$ και προέκυψε και μείωση στην τιμή της τυπικής απόκλισης, αν και αυτή δεν τέθηκε ως στόχος, κατά $5.9\ \%$. Στη δεύτερη βελτιστοποίηση η τυπική απόκλιση μειώθηκε κατά $7.2\ \%$, ενώ η μέση τιμή μειώθηκε κατά $6.6\ \%$, αν και δεν τέθηκε ως στόχος.

Κεφάλαιο 8

Σύνοψη-Συμπεράσματα

Στόχος της διατριβής ήταν να αναπτυχθούν περαιτέρω και να επεκταθούν μέθοδοι που βασίζονται στη συνεχή συζυγή μέθοδο για προβλήματα μόνο ροής και CHT ενός ή περισσότερων στόχων, με ή χωρίς αβεβαιότητες. Στα προβλήματα που μελετήθηκαν οι ροές ήταν χρονικά μόνιμες, το ρευστό ασυμπίεστο και η τύρβη λήφθηκε υπόψη μέσω του μοντέλου Spalart–Allmaras. Η τεχνογνωσία που αποκτήθηκε από προβλήματα μόνο ροής για τον υπολογισμό παραγώγων με ακρίβεια και χαμηλό κόστος, όπως η διαφόριση της τύρβης [51, 40] και η κατάλληλη διαχείριση των παραγώγων ευαισθησίας πλέγματος [27] επεκτάθηκαν σε προβλήματα CHT. Οι μέθοδοι που αναπτύχθηκαν αρχικά δοκιμάστηκαν/πιστοποιήθηκαν σε 2Δ ακαδημαϊκά προβλήματα και, στη συνέχεια, εφαρμόστηκαν σε 3Δ εφαρμογές. Το λογισμικό για την υλοποίηση των αναπτυχθεισών μεθόδων προγραμματίστηκε στο περιβάλλον ανοιχτού κώδικα OpenFOAM©.

8.1 Στοιχεία Καινοτομίας και Σχετικές Δημοσιεύσεις

Συνοπτικά, οι καινοτομίες τις διατριβής είναι οι εξής:

- Αναπτύχθηκε η συνεχής συζυγής μέθοδος για τη βελτιστοποίηση μορφής σε προβλήματα CHT. Για πρώτη φορά στη βιβλιογραφία διαφορίστηκαν για προβλήματα CHT οι εξισώσεις του μοντέλου τύρβης. Επίσης, αναπτύχθηκαν συζυγείς εξισώσεις μετατόπισης πλέγματος, για το ρευστό και το στερεό, ώστε να διαμορφωθεί η συζυγής διατύπωση *E-SI* σε προβλήματα CHT.

- Διερευνήθηκε η επίδραση του GDM στις παραγώγους, σε προβλήματα εξωτερικής, εσωτερικής αεροδυναμικής και σε CHT. Η όρο-προς-όρο ανάλυση των παραγώγων έριξε φως στις αιτίες για τις οποίες δεν υπάρχουν διαφορές στις παραγώγους που αντιστοιχούν σε διαφορετικά GDMs.

- Αναπτύχθηκαν τρεις διαφορετικές παραλλαγές αλγορίθμου πρόβλεψης-διόρθωσης, για τον υπολογισμό μετώπου Pareto, βασιζόμενες στο [47]. Και στις τρεις πα-

ραλλαγές αποφεύχθηκε ο υπολογισμός του ακριβούς Εσσιανού μητρώου που εμφανίζεται στο αριστερό μέλος των προς επίλυση συστημάτων, λόγω του υψηλού κόστους. Για αυτόν το σκοπό, (α) υπολογίστηκαν γινόμενα Εσσιανού-διανυσμάτων μέσω του συνδυασμού της συζυγούς μεθόδου και της μεθόδου ευθείας διαφόρισης και χρησιμοποιήθηκαν ώστε να επιλυθεί το σύστημα στο βήμα πρόβλεψης με τη μέθοδο GMRES και (β) προσεγγίστηκε το Εσσιανό μητρώο μέσω της μεθόδου damped BFGS.

- Αναπτύχθηκε για πρώτη φορά η μέθοδος iPCE για τις πρωτεύουσες και συζυγείς εξισώσεις των προβλημάτων CHT, με σκοπό τον υπολογισμό των στατιστικών ροπών και των παραγώγων τους με όσο το δυνατόν χαμηλότερο κόστος.

Δημοσιεύσεις σε επιστημονικά περιοδικά:

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. "The continuous adjoint method for shape optimization in conjugate heat transfer problems with turbulent incompressible flows". Applied Thermal Engineering, 140:351{362, 2018.

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, A.G. Tsolovikos, K.C Giannakoglou. "Effect of Grid Displacement Models on Sensitivity Derivatives computed by Continuous Adjoint in Aerodynamic and Conjugate Heat Transfer Shape Optimization". Engineering Optimization, doi 10.1080/0305215X.2020.1796998.

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, K.C Giannakoglou. "Adjoint-assisted Pareto Front Tracing in Aerodynamic and Conjugate Heat Transfer Shape Optimization". Computers and Fluids (δεκτό προς δημοσίευση).

Δημοσιεύσεις σε συνέδρια:

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, K.C. Giannakoglou, "Conjugate heat transfer shape optimization of internal cooling systems using continuous adjoint in OpenFOAM". 7th OpenFOAM Conference, Berlin, Germany, 2019

- K. Gkaragkounis and E. Papoutsis-Kiachagias and V. Asouti and K. Giannakoglou. "Adjoint-based Pareto front tracing in aerodynamic shape optimization". In ICCFD10-2018-322, Barcelona, Spain, 9-13 July 2018.

- E.M. Papoutsis-Kiachagias, J. Koch, K. Gkaragkounis, K. Giannakoglou. "A Continuous Adjoint Framework for Shape and Topology Optimization and their Synergistic Use". 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Florida, USA, 2018

- K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, K.C. Giannakoglou . "Conjugate Heat Transfer shape optimization based on the continuous adjoint method". VII International Conference on Coupled Problems in Science and Engineering, Rhodes, Greece, 2017

- K.C. Giannakoglou, E.M. Papoutsis-Kiachagias, I.S. Kavvadias, K.T. Gkaragkounis. "Continuous Adjoint in Shape & Topology Optimization-Recent Developments & Applications". Seminar on Adjoint CFD Methods in Industry and Research, NAFEMS, 2016

Κεφάλαια σε βιβλία:

- K.C. Giannakoglou, E.M. Papoutsis-Kiachagias, K.T. Gkaragkounis, K.D. Samouchos, C.A. Vezyris, C.A. J.R.L. Koch. "The continuous adjoint method in aero/hydrodynamic optimization". Von Karman Institute Lecture Series, 2019

Βραβεία:

- **Academic Best Paper Award:** K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, K.C. Giannakoglou, "Conjugate heat transfer shape optimization of internal cooling systems using continuous adjoint in OpenFOAM". 7th OpenFOAM Conference, Berlin, Germany, 2019

8.2 Πιθανές Μελλοντικές Εργασίες

- Η συζυγής μέθοδος για μόνιμα προβλήματα CHT μπορεί να επεκταθεί σε μημόνιμα προβλήματα. Επειδή οι συζυγείς εξισώσεις λύνονται με αντίθετη φορά στο χρόνο σε σχέση με τις εξισώσεις του πρωτεύοντος προβλήματος, απαιτείται η χρήση ή ανάπτυξη μεθόδων με σκοπό να μειωθούν οι αυξημένες απαιτήσεις σε υπολογιστική μνήμη και αποθηκευτικό χώρο.

- Η συζυγής μέθοδος για προβλήματα CHT μπορεί να επεκταθεί για να ληφθούν υπόψη οι εξαρτήσεις των συντελεστών ειδικής μεταφοράς θερμότητας, θερμικής αγωγιμότητας και η συνεκτικότητα από τη θερμοκρασία. Επιπλέον, μπορεί να ληφθούν υπόψη και οι δυνάμεις λόγω άνωσης διαφορίζοντας τους όρους της προσέγγισης Boussinesq [10].

- Η διαφόριση των μοντέλων τύρβης για προβλήματα CHT μπορεί να επεκταθεί για άλλα μοντέλα τύρβης, όπως το $k-\omega$ SST [28], το κ-ϵ [42] κοκ.

- Ο αλγόριθμος πρόβλεψης-διόρθωσης για τον υπολογισμό μετώπου Pareto μπορεί να επεκταθεί σε προβλήματα με περισσότερους από δύο στόχους. Αυτό προϋποθέτει τη δημιουργία ενός αλγορίθμου που θα θέτει τιμές-στόχους $\hat{J}_i$ στις συναρτήσεις του πολυκριτηριακού προβλήματος βελτιστοποίησης, με σκοπό

την εξερεύνηση όλων περιοχών του μετώπου Pareto, τον έλεγχο της απόστασης των σημείων του μετώπου και τη διατήρηση του υπολογιστικού κόστους σε χαμηλό επίπεδο.

- Η μέθοδος iPCE για προβλήματα CHT μπορεί να επεκταθεί σε τυρβώδεις ροές.
- Σε πολυκριτηριακά προβλήματα βελτιστοποίησης τοπολογίας η χρήση εξελικτικών μεθόδων έχει πρακτικά απαγορευτικό κόστος, επειδή οι μεταβλητές σχεδιασμού είναι τόσες όσες οι κυψέλες του υπολογιστικό πλέγματος. Συνεπώς, ο αλγόριθμός πρόβλεψης-διόρθωσης αυτής της διατριβής που βασίζεται στην κλίση των συναρτήσεων-στόχου μπορεί να επεκταθεί και σε τέτοια προβλήματα ώστε να μειωθεί δραστικά το κόστος υπολογισμού μετώπου Pareto.

Bibliography

[1] E. Alba and B. Dorronsoro. Cellular Genetic Algorithms. Springer-Verlag, 2008.

[2] E. Alba and J.M. Troya. Analyzing synchronous and asynchronous parallel distributed genetic algorithms. Future Generation Computer Systems, 17(4):451–465, 2001.

[3] W.K. Anderson and D.L. Bonhaus. Aerodynamic design on unstructured grids for turbulent flows. NASA Technical Memorandum 112867, 1997.

[4] W.K. Anderson and D.L. Bonhaus. Airfoil design on unstructured grids for turbulent flows. AIAA Journal, 37(2):185–191, 1999.

[5] W.K. Anderson and V. Venkatakrishnan. Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation. Computers & Fluids, 28(4-5):443–480, 1999.

[6] V.G. Asouti and K.C. Giannakoglou. Aerodynamic optimization using a parallel asynchronous evolutionary algorithm controlled by strongly interacting demes. Engineering Optimization, 41(3):241–257, 2009.

[7] T. Bäck. Evolutionary Algorithms in Theory and Practice. Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford University Press, 1996.

[8] D.P. Bertsekas. Constrained Optimization and Lagrange Multiplier Methods, 1st edition. Athena Scientific, 1996.

[9] D.P. Bertsekas. Nonlinear Programming, 2nd edition. Athena Scientific, 1999.

[10] J. Boussinesq. Theorie de l'ecoulement tourbillonnant et tumulueux des liquides and les lits rectilignes a grande section. Gauthier-Villars et Fils, Des Comptes Rendus des Seances de L'academie des Sciences, 1897.

[11] L. Davis. Handbook of genetic algorithms. Van Nostrand Reinhold, 1991.

[12] R.P. Dwight and J. Brezillon. Effect of approximations of the discrete adjoint on gradient–based optimization. AIAA Journal, 44(12):3022–3031, 2006.

[13] M.A. El-Beltagy, P.B. Nair, and A.J. Keane. Metamodeling techniques for evolutionary optimization of computationally expensive problems: Promises and limitations. In Genetic and Evolutionary Computation Conference - GECCO 1999. Morgan Kaufmann, San Fransisco, USA, 27-30 June 1999.

[14] J. Elliot and J. Peraire. Aerodynamic design using unstructured meshes. In AIAA Paper 1996-1941, 27th Fluid Dynamics Conference, New Orleans, LA, 17-20 June 1996.

[15] M. Ghavami Nejad, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. Aerodynamic shape optimization using the truncated Newton method and continuous adjoint. In ECCOMAS Congress 2016, VII European Congress on Computational Methods in Applied Sciences and Engineering, Crete island, Greece, June 5-10 2016.

[16] K.C. Giannakoglou. Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence. Progress in Aerospace Sciences, 38(1):43–76, 2002.

[17] M.B. Giles, M.C. Duta, J.D. Muller, and N.A. Pierce. Algorithm developments for discrete adjoint methods. AIAA Journal, 41(2), 2003.

[18] P.E. Gill, W. Murray, and M.H. Wright. Practical Optimization. Academic Press, 1981.

[19] K.T. Gkaragkounis, E.M. Papoutsis-Kiachagias, and K.C. Giannakoglou. The continuous adjoint method for shape optimization in conjugate heat transfer problems with turbulent incompressible flows. Applied Thermal Engineering, 140:351–362, 2018.

[20] F. Herrera, M. Lozano, and C. Moraga. Hierarchical distributed genetic algorithms. International Journal of of Intelligent Systems, 14(9):1099–1121, 1999.

[21] A. Jameson. Aerodynamic design via control theory. Journal of Scientific Computing, 3:233–260, 1988.

[22] A. Jameson. Optimum aerodynamic design using CFD and control theory. In AIAA Paper 1995-1729, 12th Computational Fluid Dynamics Conference, 1995.

[23] A. Jameson and S. Kim. Reduction of the adjoint gradient formula in the continuous limit. In AIAA Paper 2003-0040, 41th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, USA, January 2003.

[24] A. Jameson and J. Reuther. Control theory based airfoil design using the Euler equations. In AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization. Panama City Beach, USA, September 1994.

[25] H. Jasak and Ž. Tuković. Automatic mesh motion for the unstructured finite volume method. Transactions of FAMENA, 30(2):1–18, 2007.

[26] K. Gkaragkounis and E. Papoutsis-Kiachagias and V. Asouti and K. Giannakoglou. Adjoint-based Pareto front tracing in aerodynamic shape optimization. In ICCFD10-2018-322, Barcelona, Spain, 9-13 July 2018.

[27] I.S. Kavvadias. Aerodynamic optimization for turbulent flows using adjoint methods and GPUs. PhD thesis, National Technical University of Athens, Greece, 2016.

[28] I.S. Kavvadias, E.M. Papoutsis-Kiachagias, G. Dimitrakopoulos, and K.C. Giannakoglou. The continuous adjoint approach to the k–ω SST turbulence model with applications in shape optimization. Engineering Optimization, 47(11):1523–1542, 2015.

[29] 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. Journal of Computational Physics, 301:1–18, 2015.

[30] L.D. Landau and E.M. Lifshitz. Fluid Mechanics, Volume 6 of Course of Theoretical Physics. Pergamon Press, 1987.

[31] S.H. Lee, W. Chen, and B.M Kwak. Robust design with arbitrary distributions using gauss-type quadrature formula. Structural and Multidisciplinary Optimization, 39:227–243, 2009.

[32] X. Liu, N. Qin, and H. Xia. Fast dynamic grid deformation based on Delaunay graph mapping. Journal of Computational Physics, 211(2):405 – 423, 2006.

[33] C. Lozano, E. Andrés, M. Martin, and P. Bitrián. Domain versus boundary computation of flow sensitivities with the continuous adjoint method for aerodynamic shape optimization problems. Numerical Methods in Fluids, 70(10):1305–1323, 2012.

[34] D.G. Luenberger. Linear and Nonlinear Programming, 2nd edition. Kluwer Academic Publishers, 2003.

[35] E. Luke, E. Collins, and E. Blades. A fast mesh deformation method using explicit interpolation. Journal of Computational Physics, 231(2):586 – 601, 2012.

[36] Z. Michalewicz. Genetic Algorithms + Data Structures = Evolution Programs, 2nd edition. Springer-Verlag, Berlin Heidelberg, Germany, 1994.

[37] B. Mohammadi and O. Pironneau. Applied shape optimization for fluids. Oxford University Press, 2001.

[38] J. Nocedal and S.J. Wright. Numerical Optimization. Springer, New York, 1999.

[39] D.I. Papadimitriou and K.C. Giannakoglou. Aerodynamic design using the truncated Newton algorithm and the continuous adjoint approach. International Journal for Numerical Methods in Fluids, 68(6):724–739, 2012.

[40] E.M. Papoutsis-Kiachagias. Adjoint Methods for Turbulent Flows, Applied to Shape or Topology Optimization and Robust Design. PhD thesis, National Technical University of Athens, Greece, 2013.

[41] E.M. Papoutsis-Kiachagias, N. Magoulas, J. Mueller, C. Othmer, and K.C. Giannakoglou. Noise reduction in car aerodynamics using a surrogate objective function and the continuous adjoint method with wall functions. Computers & Fluids, 122:223–232, 2015.

[42] E.M. Papoutsis-Kiachagias, A.S. Zymaris, I.S. Kavvadias, D.I. Papadimitriou, and K.C. Giannakoglou. The continuous adjoint approach to the k–ϵ turbulence model for shape optimization and optimal active control of turbulent flows. Engineering Optimization, 47(3):370–389, 2015.

[43] S.V. Patankar and D.B. Spalding. Calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. International Journal of Heat and Mass Transfer, 15:1787–1806, 1972.

[44] J. Periaux and G. Winter. Genetic Algorithms In Engineering And Computer Science. John Wiley & Sons, 1995.

[45] J. Peter and R.P. Dwight. Numerical sensitivity analysis for aerodynamic optimization: A survey of approaches. Computers & Fluids, 39(3):373–391, 2010.

[46] Y. Saad and M.H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM Journal on Scientific and Statistical Computing, 7(3):856–869, 1986.

[47] S. Schmidt and V. Schulz. Pareto-curve continuation in multi-objective optimization. Pacific Journal of Optimization, 4(2):243–257, 2008.

[48] D. Thévenin and G. Janiga. Optimization and Computational Fluid Dynamics. Springer, 2008.

[49] G. Wang, L. Xu, Ch. Li, and Z. Ye. Influence of mesh sensitivities on computational-fluid-dynamics-based derivative calculation. AIAA Journal, 54(12):3717–3726, 2016.

[50] A.S. Zymaris. Adjoint methods for the design of shapes with optimal aerodynamic performance in laminar and turbulent flows. PhD thesis, National Technical University of Athens, Greece, 2010.

[51] A.S. Zymaris, D.I. Papadimitriou, K.C. Giannakoglou, and C. Othmer. Continuous adjoint approach to the Spalart-Allmaras turbulence model for incompressible flows. Computers & Fluids, 38(8):1528–1538, 2009.