
# National Technical University of Athens

**School of Mechanical Engineering**
**Fluids Department**
**Laboratory of Thermal Turbomachines**
**Parallel CFD & Optimization Unit**


# Application of the Continuous Adjoint Method in the Design-Optimization of Automotive HVAC Ducts

Diploma Thesis
by


**Andreas Stefanos Margetis**


Advisor:
Kyriakos C. Giannakoglou, Professor NTUA

Athens, July 2019

# Acknowledgements

First and foremost, I would like to express my deep graditude to my parents, Ioannis and Panagiota, for their continuous support all these years and for the fact that they are always present when I need them. Their contribution to all stages of my life, during my school and student years, has been decisive. I thank them for all kind of sacrifices they made for me, for the ethical values they gave me, that were the foundation of the effort that led to this diploma thesis.

I would also like to thank my professor Kyriakos Giannakoglou for the time he devoted in making corrections and suggestions, but above all for trusting me and giving me the opportunity to work on my diploma thesis in industrial environment. I also thank him because as a professor during the courses he taught, he caught my interest to the subject of his work which made me want to deepen my knowledge in it.

I should also like to thank Dr. Evangelos Papoutsis Kiachagias and PhD candidate Konstantinos Gkaragkounis for their help during the work I did in the previous semesters, during my theoretical preparation at the start of my diploma thesis and for their continuous support till its completion. I thank them for their patience and their willingness to help despite their heavy workload.

Last but not least, I would like to thank my BMW supervisors Dr. Majid Hojjat and Dr. Steffen Jahnke for giving me the opportunity of a 6-month internship in the automotive industry.

# Ευχαριστίες

Πρώτα απ΄ όλα, θα ήθελα να ευχαριστήσω τους γονείς μου, Ιωάννη και Παναγιώτα, για τη συνεχή υποστήριξη τους εδώ και 23 χρόνια, για το γεγονός ότι είναι πάντα ¨εκεί¨. Η συμβολή τους υπήρξε καθοριστική σε όλα τα στάδια της ζωής μου, στα σχολικά και στα φοιτητικά μου χρόνια. Τους ευχαριστώ για τις θυσίες, κάθε είδους, που έκαναν και κάνουν για εμένα, για τα ψυχικά και ηθικά εφόδια που μου έδωσαν και αποτέλεσαν τα θεμέλια μιας προσπάθειας που επιστεγάζεται με την παρούσα διπλωματική εργασία.

Θα ήθελα ακόμη να ευχαριστήσω τον καθηγητή μου κ. Κυριάκο Γιαννάκογλου για το χρόνο που αφιέρωσε σε διορθώσεις και υποδείξεις και πάνω απ΄ όλα για την εμπιστοσύνη που έδειξε στο πρόσωπό μου, δίνοντάς μου τη δυνατότητα να εκπονήσω τη διπλωματική μου εργασία στην αυτοκινητοβιομηχανία. Τον ευχαριστώ ακόμη γιατί ως καθηγητής στα αμφιθέατρα με έκανε να αγαπήσω το αντικείμενό εργασίας του και να θέλω να ασχοληθώ με αυτό σε μεγαλύτερο βάθος.

Δε θα μπορούσα να μην ευχαριστήσω τον Δρ. Ευάγγελο Παπουτσή Κιαχαγιά και τον υποψήφιο Διδάκτορα Κωνσταντίνο Γκαραγκούνη, για τη βοήθεια που μου παρείχαν κατά την εκπόνηση του υπολογιστικού θέματος ένα χρόνο πριν, κατά τη θεωρητική μου προετοιμασία στην έναρξη της διπλωματικής εργασίας και για τη συνεχή υποστήριξή τους μέχρι την ολοκλήρωσή της. Τους ευχαριστώ για την υπομονή και την παντοτινή προθυμία τους να βοηθήσουν παρά τον αυξημένο φόρτο εργασίας τους.

Τέλος, θα ήθελα να ευχαριστήσω τους επιβλέποντές μου στη BMW Δρ. Majid Hojjat και Δρ. Steffen Jahnke που συνέβαλαν ώστε να εργαστώ για 6 μήνες στην αυτοκινητοβιομηχανία.

**National Technical University of Athens**
**School of Mechanical Engineering**
**Fluids Department**
**Parallel CFD & Optimization Unit**

# Application of the Continuous Adjoint Method in the Design-Optimization of Automotive HVAC Ducts

Diploma Thesis

**Andreas Stefanos Margetis**

Advisor: Kyriakos C. Giannakoglou, Professor NTUA

Athens, July 2019

## Abstract

This thesis presents the use of the continuous adjoint method for the shape optimization of HVAC (Heating, Ventilation and Air Conditioning) ducts of a passenger vehicle and the main part of the presented work was carried out at the BMW premises in Munich. The flow and the continuous adjoint solver used for this purpose have been developed by the Parallel CFD & Optimization Unit (PCOpt) of NTUA in the OpenFOAM environment. This software computes the sensitivity derivatives of the objective function chosen by the engineer, with respect to the surface nodes coordinates of the geometry being optimized. Using these derivatives, the surface nodes move in order to improve the objective function value as computed by the BMW optimization software 'ShapeModule', whose operation is based on the node-based optimization method known as Vertex Morphing. As part of this thesis, the communication software between the flow and adjoint solver of PCOpt/NTUA and ShapeModule was reimplemented, basing its functionality on reading and writing of the necessary files (File-IO coupling) rather than on the previously used MPI communication protocol. Furthermore, the communication software was expanded to support the separate use of the sensitivity derivatives computed by each adjoint solver. After the successful coupling of the two softwares, an HVAC duct was designed and optimized. Given that the initial geometry was a box with one inlet and three outlets, topology optimization was applied at first, in order to produce a first shape of the duct that was then optimized further using shape optimization. Subsequently, the produced geometry was smoothed using the Inspire software and the ANSA software. At the final stage, the duct was optimized with the coupled PCOpt/NTUA and ShapeModule software, using two objective functions. The first one aims at the minimization of the total pressure losses between inlet and outlets,

whereas the second one is a surrogate, noise-related objective function, which aims at the minimization of the turbulence related noise by minimizing the integral of the squared turbulent viscosity in specific parts of the duct.

Furthermore, the continuous adjoint method for an objective function that describes the flow uniformity at the outlet of the domain to be optimized was developed. This objective function was used on a different HVAC duct with one inlet and one outlet in a single-objective optimization problem, but also together with the total pressure losses objective function in a double-objective optimization problem. These simulations were conducted by using exclusively the PCOpt/NTUA software.

**Εθνικό Μετσόβιο Πολυτεχνείο**
**Σχολή Μηχανολόγων Μηχανικών**
**Τομέας Ρευστών**
**Μονάδα Παράλληλης Υπολογιστικής Ρευστοδυναμικής & Βελτιστοποίησης**

# Εφαρμογή της Συνεχούς Συζυγούς Μεθόδου στη Βελτιστοποίηση Μορφής Αεραγωγών Κλιματισμού Επιβατικού Αυτοκινήτου

Διπλωματική εργασία

**Ανδρέας Στέφανος Μαργέτης**

Επιβλέπων: Κυριάκος Χ. Γιαννάκογλου, Καθηγητής ΕΜΠ

Αθήνα, Ιούλιος 2019

Η διπλωματική εργασία ασχολείται με την εφαρμογή της συνεχούς συζυγούς μεθόδου στη βελτιστοποίηση της μορφής αγωγών κλιματισμού επιβατικού αυτοκινήτου με το μεγαλύτερο μέρος της εργασίας να πραγματοποιήθηκε στις εγκαταστάσεις της BMW στο Μόναχο. Ο επιλύτης της ροής και ο συνεχής συζυγής επιλύτης που χρησιμοποιήθηκε για το σκοπό αυτό έχει αναπτυχθεί από τη Μονάδα Παράλληλης Υπολογιστικής Ρευστοδυναμικής & Βελτιστοποίησης (ΜΠΥΡ&Β) του ΕΜΠ σε περιβάλλον OpenFOAM. Το λογισμικό αυτό υπολογίζει τις παραγώγους ευαισθησίας της επιλεγείσας από τον μηχανικό αντικειμενικής συνάρτησης ως προς τις συντεταγμένες των επιφανειακών κόμβων της υπό βελτιστοποίηση γεωμετρίας. Με γνωστές αυτές τις παραγώγους, οι οριακοί κόμβοι μετακινούνται, βελτιώνοντας την τιμή της αντικειμενικής συνάρτησης, εφαρμόζοντας το λογισμικό βελτιστοποίησης μορφής 'ShapeModule' της BMW, το οποίο βασίζεται στην τεχνική μορφοποίησης των κόμβων του πλέγματος (Vertex Morphing). Στο πλαίσιο της διπλωματικής εργασίας προγραμματίσθηκε λογισμικό για την επικοινωνία του επιλύτη του ευθέος και συζυγούς προβλήματος της ΜΠΥΡ&Β/ΕΜΠ με το ShapeModule, το οποίο περιλαμβάνει την εγγραφή και ανάγνωση των απαραίτητων αρχείων επικοινωνίας (File-IO coupling) αντικαθιστώντας το πρότερα χρησιμοποιούμενο πρωτόκολλο επικοινωνίας MPI. Συγχρόνως, επεκτάθηκε το λογισμικό επικοινωνίας ώστε να μπορεί να υποστηρίζει το ξεχωριστό χειρισμό των παραγώγων ευαισθησίας κάθε συζυγούς επιλύτη. Μετά την επιτυχή σύζευξη των δύο κωδίκων, σχεδιάστηκε και βελτιστοποιήθηκε αγωγός κλιματισμού της καμπίνας επιβατικού αυτοκινήτου (HVAC, Heating, Ventilation and Air Conditioning). Δεδομένου ότι η αρχική γεωμετρία ήταν ένα κουτί με μία είσοδο και τρεις εξόδους, πραγματοποιήθηκε αρχικά βελτιστοποίηση τοπολογίας προκειμένου να παραχθεί ένας αρχικός αγωγός ο οποίος θα μπορούσε να χρησιμοποιηθεί στη βελτιστοποίηση μορφής. Στη συνέχεια η παραχθείσα γεωμετρία εξομαλύνθηκε χρησιμοποιώντας κατά σειρά τα λογισμικά In-

spire της εταιρείας Altair και ANSA της εταιρείας BETA CAE Systems. Στο τελικό στάδιο, ο αγωγός βελτιστοποιήθηκε με το συζευγμένο λογισμικό της ΜΠΥΡ&Β/ΕΜΠ και ShapeModule χρησιμοποιώντας δύο αντικειμενικές συναρτήσεις. Η πρώτη εξ αυτών στοχεύει στην ελαχιστοποίηση των απωλειών ολικής πίεσης μεταξύ εισόδου και εξόδων, ενώ η δεύτερη στην ελαχιστοποίηση του παραγόμενου θορύβου στις εξόδους του αγωγού μέσω της μείωσης της τυρβώδους συνεκτικότητας σε επιλεγμένα τμήματά του.

Παράλληλα, αναπτύχθηκε η συνεχής συζυγής μέθοδος για αντικειμενική συνάρτηση που εκφράζει την ομοιομορφία της ροής στην έξοδο της υπό βελτιστοποίηση γεωμετρίας. Η αντικειμενική αυτή συνάρτηση χρησιμοποιήθηκε σε ένα διαφορετικό αγωγό κλιματισμού επιβατικού αυτοκινήτου με μία είσοδο και μία έξοδο τόσο για την μονοκριτηριακή βελτιστοποίησή του όσο και σε συνδυασμό με την αντικειμενική συνάρτηση των απωλειών ολικής πίεσης σε δικριτηριακή βελτιστοποίηση με στάθμιση βαρών και με περιορισμούς. Στις προσομοιώσεις αυτές χρησιμοποιήθηκε αποκλειστικά το λογισμικό της ΜΠΥΡ&Β/ΕΜΠ.

# Acronyms

| | |
|---|---|
| CFD | Computational Fluid Dynamics |
| NTUA | National Technical University of Athens |
| PCOpt | Parallel CFD & Optimization unit |
| HVAC | Heating, Ventilation and Air Conditioning |
| GBM | Gradient Based Method |
| CPU | Central Processing Unit |
| RANS | Reynolds-Averaged Navier-Stokes |
| SD | Sensitivity Derivatives |
| FAE | Field Adjoint Equations |
| PDE | Partial Differential Equation |
| gdPDE | grid displacement Partial Differential Equation |
| RHS | Right Hand Side |
| w.r.t. | with respect to |

---

| | |
|---|---|
| ΕΜΠ | Εθνικό Μετσόβιο Πολυτεχνείο |
| ΕΘΣ | Εργαστήριο Θερμικών Στροβιλομηχανών |
| ΜΠΥΡ&Β | Μονάδα Παράλληλης Υπολογιστικής Ρευστοδυναμικής & Βελτιστοποίησης |
| ΥΡΔ | Υπολογιστική Ρευστοδυναμική |
| ΜΔΕ | Μερική Διαφορική εξίσωση |

# Contents

# Chapter 1

# Introduction

The improvement of computational methods through years of research and development in conjunction with the ever increasing performance of the modern computer systems has led to a widespread use of computational fluid dynamics (CFD) in a significant range of industrial applications. The availability of CFD codes that can simulate the flow field around or inside the shape under consideration and the trustworthy results that they can produce nowadays has led to their integration in the shape optimization procedure. As a result, the necessary experiments that have to be conducted to evaluate new engineering designs have been limited in comparison to the past, leading to a significant cost reduction.

## 1.1 Shape Optimization

According to the control theory adapted for the needs of CFD-based optimization, the shape to be optimized is controlled by a number of variables, referred to as *design* or *optimization variables*. For instance, these could be the control point coordinates of volumetric B-splines polynomials parameterizing the shape under consideration. The quality of the shape to be optimized is evaluated by computing a usually integral quantity, known as the *objective function*. The objective function can be defined either on the boundaries, such as the total pressure losses which is defined at the inlet and outlet of the geometry, or in a volume inside the geometry, such as the noise induced by the turbulent viscosity. Goal of the optimization is to compute the values of the design variables that minimize or maximize the objective function.

CFD-based optimization methods can be classified into two main categories, according to the way the optimal set of design variables is computed: *stochastic* and

*deterministic*. This diploma thesis focuses exclusively on the second category. The deterministic optimization algorithms improve a given geometry by computing the gradient of the objective function in question w.r.t. the design variables, also known as the *sensitivity derivatives* (SD). A shape update is thereafter computed based on the direction dictated by the sensitivity derivatives. Subsequently, the flow field, the objective function value and the new SD field are computed based on the updated geometry. This process is repeated until either the objective function has converged to its minimum value or the user-defined maximum number of optimization cycles is reached. Since the values of the sensitivity derivatives tend to be zeroed in areas of local minima, the entrapment to a local optimum solution is highly possible. As a result, the user of the algorithm will get an optimized rather than an optimal solution, which is the main disadvantage of a gradient-based method (GBM).

The efficiency of GBMs is highly dependent on how the sensitivity derivatives are computed. The most straightforward method of computing the SDs is by using *finite differences* (FD). According to FD, each of the design variables $b_n, n \in [1, N]$, where $N$ is the total number of them, is perturbated by an infinitesimally small quantity, $\epsilon$, and the objective function is re-evaluated using the perturbated design. For a second-order FD scheme, we have

$$\frac{\delta F}{\delta b_n} = \frac{F(b_1, ..., b_n + \epsilon, ..., b_N) - F(b_1, ..., b_n - \epsilon, ..., b_N)}{2\epsilon} \tag{1.1}$$

Despite its simple implementation, since it requires only the re-computation of the value of the objective function, this method poses great concerns because of its two main drawbacks. First and foremost, the cost of the FD method scales linearly with the number of the design variables, $N$, as it requires $2N$ evaluations of $F$ by solving the flow equations, making it infeasible for large scale optimization problems. The second downside is the dependence of the computed derivatives from $\epsilon$, the value of which cannot be determined a priori. The use of a too "small" value is not always the answer to the aforementioned problem as it can introduce round-off errors. Moreover, the flow equations must be fully converged as the subtraction performed in eq. 1.1 is between two very close values of $F$.

An alternative method for the computation of the sensitivity derivatives is the *complex variable* (CV) method according to which the sensitivity derivatives are computed by

$$\frac{\delta F}{\delta b_n} = \frac{Im[F(b_1, ..., b_n + i\epsilon, ..., b_N)]}{\epsilon} \tag{1.2}$$

where $Im$ is the imaginary part of the complex function $F$ and $i = \sqrt{-1}$. As reflected in eq. 1.2, the round-off errors cease to exist, since there is no subtraction of two very close values as in the case of FD. Subsequently, the method is $\epsilon$-independent and there is no need for the flow equations to be fully converged. Nevertheless,

the cost of the complex variable method still scales linearly with $N$ and in fact it requires $N$ evaluations of the objective function.

Another alternative to FDs is the *direct differentiation* (DD) method, according to which the flow equations are differentiated w.r.t. $\mathbf{b}$ and the $N$ linear systems that arise are solved to define the derivatives of the flow variables w.r.t. to the design variables. Given that the SDs are expressed in terms of these fields, their final computation is straightforward. DD is harder to implement than FD (since a new flow solver has to be written) and its cost still scales with $N$, making it inadequate for large scale simulations.

## 1.2 The Adjoint Method

The *adjoint* method of computing the sensitivity derivatives required by GBMs is the alternative that has a cost practically independent from the number of the design variables $N$. As a result, this method is a perfect choice for large industrial optimization problems. In order to achieve this independence, an augmented objective function is defined, by adding the volume integrals of the residuals of the flow equation (also referred to as the *primal* or *state* equations), multiplied by the adjoint (or *co-state* or *dual*) variable fields, to $F$. Considering that the residuals of the primal equations must be zero, $F \equiv F_{aug}$. After differentiating the augmented objective function and re-arranging the resulting terms, the system of adjoint equations and adjoint boundary conditions is formulated, the numerical solution of which leads to a N-independent computation of the SDs.

There are two different approaches [1] on how the aforementioned adjoint method can be applied, that differ from each other in the sequence that the differentiation of the objective function and the discretization of the flow equations happen. In the *discrete* adjoint approach, the residuals of the primal equations that are added to the objective function are in their discrete form and the resulting system of adjoint equations and adjoint boundary conditions after the differentiation is already discretized and ready to be numerically solved. On the other hand, in the *continuous* adjoint approach, the residuals of the primal equations that are added to the objective function are in their continuous form and the resulting system of adjoint equations and the boundary conditions have to be discretized, in order to be numerically solved

There is a general consensus, that both discrete and continuous adjoint methods can produce sensitivity derivatives with sufficient accuracy to be used in common optimization problems. Nevertheless, the discrete approach is more accurate in computing the SD especially on coarse meshes, since it takes the primal discretization schemes into consideration, although its implementation can become cumbersome

when higher discretization schemes are used. On the other hand, the continuous adjoint outweighs the discrete one in terms of CPU cost and memory requirements per iteration. Continuous approach also leads to better physical understanding of the adjoint system, since closed-form expressions exist for the field adjoint equations, their boundary conditions and the sensitivity derivatives expression.

This diploma thesis focuses on the applications of the continuous adjoint method in industrial internal-aerodynamics automotive applications. The mathematical formulation and software programming of the adjoint solver for incompressible fluid flows has been performed by the PCOpt/NTUA within the OpenFOAM©environment.

# Chapter 2

# CFD Analysis and Optimization Process

In this chapter, the mathematical formulation of the optimization procedure for a steady state incompressible flow over a given domain is presented.

## 2.1 Formulation of the Primal Problem

### 2.1.1 Flow Modelling

This thesis is concerned with flows that are highly turbulent, as this is the case in most industrial applications. A turbulent flow is characterized by unsteady fluctuations of the pressure and velocity, which are not completely random but they develop coherent structures, called eddies. In order to take into account the whole range of spatial and temporal scales of the turbulence an unbearably small cell size is necessary, as well as an infinitesimally small discretization of the time domain. The cost of such simulations, known as direct numerical simulations (DNS), in which the Navier-Stokes equations are numerically solved without any turbulence modelling, is practically unacceptable with the computational resources available in the industry today, especially when the flow simulation is part of a larger optimization problem.

The most widely used approach for calculating industrial flows is by the Reynolds-Averaged Navier-Stokes models (RANS). The equations that are solved are the ensemble (or time or Reynolds) averaged Navier-Stokes and turbulence models are

used to take the random fluctuations of the flow into account. The idea that was suggested by Reynolds, already since 1895, and in its core is still valid today, is the decomposition of the flow variables into the mean and the fluctuating components [2, 3].

The system of the RANS equations for an incompressible fluid flow in Cartesian coordinates is shown below, where repeated indices denote summation, according to the Einstein's convention.

The conservation of mass or continuity equation is

$$\frac{\partial \bar{v}_j}{\partial x_j} = 0 \tag{2.1}$$

and the conservation of momentum equation in non-conservative form is

$$\bar{v}_j \frac{\partial \bar{v}_i}{\partial x_j} + \frac{\partial \bar{p}}{\partial x_i} - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial \bar{v}_i}{\partial x_j} + \frac{\partial \bar{v}_j}{\partial x_i}\right)\right] - \frac{\partial}{\partial x_j}\left(-\overline{v_i' v_j'}\right) = 0 \ , \qquad i = 1,2,3 \tag{2.2}$$

where $\bar{v}_i$ are the velocity components, $\bar{p}$ is the static pressure divided by the constant fluid density. The '$-$' symbol is used to denote the mean value of a variable, whereas '$'$' symbol to denote the fluctuation of a variable. The new term appearing in eq. 2.2 is the turbulent shear stress or Reynolds stress tensor $\tau_{ij}'/\rho = -\overline{v_i' v_j'}$. To obtain equations containing only the mean velocity and pressure, the Reynolds stresses term needs to be modelled as a function of the mean flow quantities, removing any reference to the fluctuating part of the velocity. In this thesis, this is achieved through the Boussinesq hypothesis, proposed by Boussinesq in 1877 [3, 4].

$$-\overline{v_i' v_j'} = \nu_t\left(\frac{\partial \bar{v}_i}{\partial x_j} + \frac{\partial \bar{v}_j}{\partial x_i}\right) - \frac{2}{3} k \delta_i^j \tag{2.3}$$

where $\delta_i^j$ is the Kronecker delta, $k$ is the turbulent kinetic energy (TKE)

$$k = \frac{1}{2}\overline{v_i' v_i'} \tag{2.4}$$

and $\nu_t$ is the kinematic turbulent -or eddy- viscosity. $\nu_t$ has the same units as the kinematic viscosity of the fluid, $m^2/s$.
Taking these into consideration, eq. 2.2 becomes

$$R_i^v = v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial}{\partial x_j}\left[(\nu + \nu_t)\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] + \frac{\partial p}{\partial x_i} = 0 \ , \qquad i = 1,2,3 \tag{2.5}$$

where the '$^-$' symbol is omitted for simplicity since all flow variable correspond to the mean flow. This convection is followed for the rest of this thesis.

The introduction of the $\nu_t$ field into the RANS equations yields a new complication for the solution of the problem, since the necessity of more equations to close the system arises. This issue, also referred to as the 'closure problem', is overcome with the introduction of the turbulence models, which attempt to predict the turbulence evolution using extensive experimental data and analysis of the past decades. Turbulence models that are using the Boussinesq hypothesis are known as eddy viscosity models or EVM's. In this category belongs the Spalart-Allmaras turbulence model [5], which is used in the analysis performed within this diploma thesis and is discussed below.

### 2.1.2 The Spalart-Allmaras Turbulence Model

Spalart-Allmaras [5] is a low-cost one equation mixing-length turbulence model for incompressible flows. It was initially designed for aerospace applications involving wall-bounded flows and has been shown to give good results for boundary layers subjected to adverse pressure gradients. It uses the following single transport PDE for a modified eddy viscosity, known as the Spalart-Allmaras variable, or $\widetilde{\nu}$.

$$R^{\widetilde{\nu}} = v_j \frac{\partial \widetilde{\nu}}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu}}{\partial x_j}\right] - \frac{c_{b2}}{\sigma}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right)^2 - \widetilde{\nu}\, P\left(\widetilde{\nu}\right) + \widetilde{\nu}\, D\left(\widetilde{\nu}\right) = 0 \tag{2.6}$$

Eddy viscosity $\nu_t$ relates to $\widetilde{\nu}$ with the following expression.

$$\nu_t = \widetilde{\nu} f_{v_1} \tag{2.7}$$

The production and dissipation terms are given by

$$P(\widetilde{\nu}) = c_{b1}\widetilde{Y}\ , \qquad D(\widetilde{\nu}) = c_{w1} f_w(\widetilde{Y})\frac{\widetilde{\nu}}{\Delta^2} \tag{2.8}$$

where $\widetilde{Y}$ is computed through

$$\widetilde{Y} = Y f_{v_3} + \frac{\widetilde{\nu}}{\Delta^2 \kappa^2} f_{v_2}\ , \qquad Y = \left| e_{ijk}\frac{\partial v_k}{\partial x_j}\right| \tag{2.9}$$

with $Y$ standing for the vorticity magnitude and $\Delta$ being the distance of cell-centres from the wall boundaries (since OpenFOAM uses a finite volume cell-centered dis-

cretization of the domain [6]). The model functions read

$$\begin{aligned} f_{v_1} &= \frac{\chi^3}{\chi^3 + c_{v_1}^3} \ , \quad f_{v_2} = \frac{1}{\left(1 + \frac{\chi}{c_{v_2}}\right)^3} \\ f_{v_3} &= \frac{(1 + \chi f_{v1})}{c_{v2}} \left[ 3\left(1 + \frac{\chi}{c_{v2}}\right) + \left(\frac{\chi}{c_{v2}}\right)^2 \right] \left(1 + \frac{\chi}{c_{v2}}\right)^{-3} \\ \chi &= \frac{\widetilde{\nu}}{\nu} \ , \quad f_w = g\left(\frac{1 + c_{w_3}^6}{g^6 + c_{w_3}^6}\right)^{1/6} \\ g &= r + c_{w_2}(r^6 - r) \ , \quad r = \frac{\widetilde{\nu}}{\widetilde{Y}\kappa^2\Delta^2}. \end{aligned} \tag{2.10}$$

The constants of the model 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 $Y$, is

$$e_{ijk} = \begin{cases} +1 & (i,j,k) \in (1,2,3),\ (2,3,1),\ (3,1,2) \\ -1 & (i,j,k) \in (1,3,2),\ (3,2,1),\ (2,1,3) \\ 0 & i = j,\ j = k,\ k = i \end{cases} \tag{2.11}$$

### 2.1.3 Wall Treatment

The turbulence model as defined till this point is valid only in the area where turbulence is fully developed and does not perform well in the area close to the wall. In order to deal with the near-wall, viscosity-affected flow region, two strategies can be followed.

According to *Low Reynolds number of Turbulence* modelling [3], the viscous sublayer shown in figure 2.1 is resolved numerically and viscous effects are included in the turbulence model, often by including 'near-wall damping' terms and other source terms in the model transport equation 2.6. Since very steep gradients of velocity (and turbulence statistics) occur across the viscous sublayer, very fine grids are needed to provide adequate numerical resolution. Typically, this requires ensuring that the first cell center lies inside the viscous sublayer, at a non-dimensional distance of around $y^+ < 1$. $y^+$ is defined as $y^+ = \frac{v_\tau y}{\nu}$, where $v_\tau = \sqrt{\tau_w}$ is the friction velocity, $\tau_w$ the wall friction and y the distance of the cell center from the wall. In large 3D simulations, employing such fine grids demands significant computational resources that are not always available in the industry. This leads to the alternative that is outlined below.

According to *High Reynolds number of Turbulence* modelling, the viscous sublayer

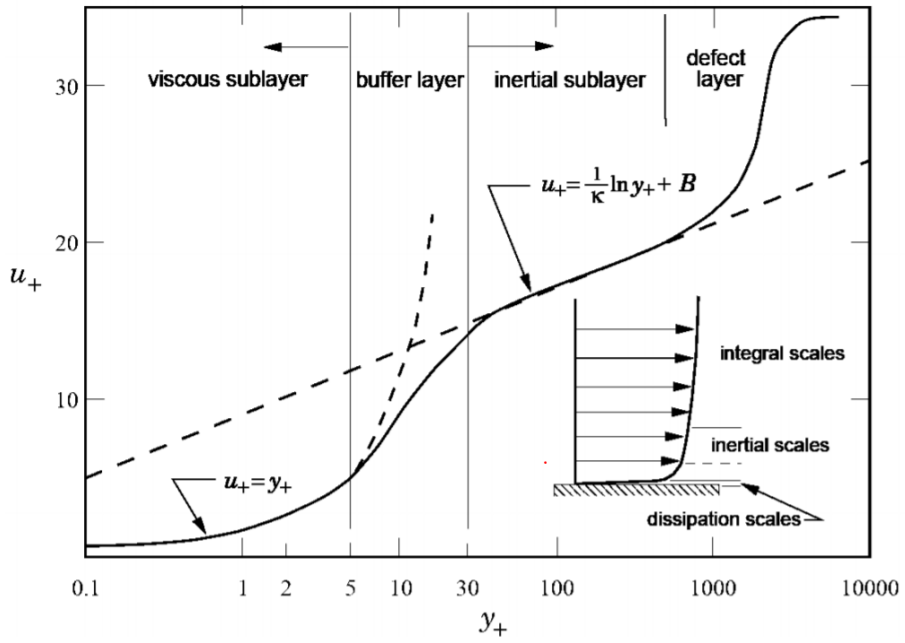



**Figure 2.1:** *Velocity profile in the characteristic regions of a developed turbulent boundary layer.*

is not resolved, but approximations are introduced to account for the flow behaviour across it. By using empirical information, analytical expressions for the mean velocity distribution across the near-wall region of the flow are developed, the so called *Wall Functions*. In High-Re models, the first cell center form the wall lies in the log-law region with the maximum accepted value for $y^+$ to be about 100 without to compromise the accuracy of the simulation. Since the viscous sublayer is not numerically resolved, a much smaller mesh size can be used, making wall functions the preferred choice for the industry. For this reason, in the scope of this thesis the wall functions approach is used.

In the Spalart-Allmaras model, wall functions are used to approximate the value of $\nu_t$ at the cell closest to the wall. $\nu_t$ is computed according to $\nu_t = \frac{u_\tau^2}{\partial U/\partial n}$, where $u_\tau$ is computed based on the $y^+$ value. The formulation for the computation of $y^+$, as programmed in OpenFOAM©, follows Spalding's Law [7]. Spalding's Law models the inner sublayer and the logarithmic region of the boundary layer with one equation

$$y^+ = v^+ + e^{-\kappa B}\left[e^{\kappa v^+} - 1 - \kappa v^+ - \frac{(\kappa v^+)^2}{2} - \frac{(\kappa v^+)^3}{6}\right] \tag{2.12}$$

where $\kappa$ is the von-Karman constant equal to 0.41 and $B \approx 5.5$.

This equation came as a result of best fit between the curve of $y^+ = u^+$ which is valid in the viscous sublayer and $u^+ = Ey^+/\kappa$ which is valid in the logarithmic region. $E$ is an empirical constant equal to $9.793 = \left(e^{-\kappa B}\right)$.

### 2.1.4 Primal Equations and Primal Boundary Conditions

According to the previous analysis, the primal problem for steady-state flow of an incompressible fluid consists of the following set of partial differential equations.

$$R^p = -\frac{\partial v_j}{\partial x_j} = 0 \tag{2.13a}$$

$$R_i^v = v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial}{\partial x_j}\left[(\nu + \nu_t)\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] + \frac{\partial p}{\partial x_i} = 0 \ , \qquad i = 1, 2, 3 \tag{2.13b}$$

$$R^{\widetilde{\nu}} = v_j \frac{\partial \widetilde{\nu}}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu}}{\partial x_j}\right] - \frac{c_{b2}}{\sigma}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right)^2 - \widetilde{\nu}\, P\left(\widetilde{\nu}\right) + \widetilde{\nu}\, D\left(\widetilde{\nu}\right) = 0 \tag{2.13c}$$

The set of the boundary conditions needed to close the primal problem are the following.

**• Inlet**

At the inlet of the computational domain, a Dirichlet boundary condition is imposed on $v_i$ and a zero Neumann on $p$. For the Spalart-Allmaras variable $\widetilde{\nu}$, a Dirichlet boundary condition is used with the exact value depending on the value of the kinematic viscosity $\nu$ of the fluid.

**• Outlet**

At the outlet of the computational domain, a zero Neumann boundary condition is imposed on $v_i$ and $\widetilde{\nu}$ and a zero Dirichlet on $p$.

**• Wall Boundaries**

At the wall boundaries of the computational domain, a no-slip, i.e. zero Dirichlet, boundary condition is imposed on $v_i$ and a zero Neumann on $p$. A zero Dirichlet boundary condition is imposed on $\widetilde{\nu}$.

## 2.2 Formulation of the Adjoint Problem

As described in the introduction, in order to compute the sensitivity derivatives of the objective function $F$ w.r.t. the design variables at a cost that does not depend on N, where $N$ is the number of the design variables, an augmented objective function, $F_{aug}$, must be defined. This is done by adding to $F$ the volume integrals of the state PDEs, multiplied by the adjoint variable fields. At this point of the analysis, it is common in the literature of continuous adjoint to neglect variations in the turbulence viscosity by assuming that changes in the shape of the aerodynamic body affect only the mean flow quantities. This is referred to as the "frozen turbulence" assumption and leads to a system of adjoint equations which doesn't include the adjoint to the turbulence model PDE(s). As stated in [8] and [9], the so-computed sensitivities are occasionally even wrongly signed and this may seriously affect the optimization process. A much more rigorous approach includes the differentiation of the turbulence model equation(s) w.r.t. the design variables and requires the formulation and solution of the adjoint to the turbulence model PDE(s). This is one of the key feautures of the adjoint solver developed by PCOpt/NTUA.

### 2.2.1 Three Continuous Adjoint Formulations: FI - SI - ESI

Based on the literature, the continuous adjoint method can be formulated in three different ways. All three formulations result in the same field adjoint equations and the same adjoint boundary conditions. What distinguishes them from one another is the final expression for the sensitivity derivatives (SD) of the objective function w.r.t. the design variables. The first chronologically formulation results in SD expressions including both boundary and field integrals (FI approach). The FI approach is characterised by high accuracy, but increased computational cost, due to integrating over the entire domain and the need of computing $\delta x_k/\delta b_n$, the so-called grid, or mesh, sensitivities. The second formulation leads to SD expressions containing only boundary, i.e. surface integrals (SI approach). It, thus, has low computational cost but can lack accuracy especially when used with coarse grids. Finally, the Enhanced SI (E-SI) approach, which also leads to SD expressions depending only on surface integrals, combines the advantages of the aforementioned methods, being as fast as the standard SI and as accurate as the FI one [10].

Next, follows a brief presentation of the three methods, with a more thorough analysis to be found in [11]. In order to make the following analysis valid for all formulations and keep the resulting expressions as simple as possible, we define a

generalised augmented objective function,

$$F_{aug} = F + \int_{\Omega} \Psi_i R_i d\Omega \tag{2.14}$$

where $F$ is the objective function, $R_i = 0$ are the residuals of the state equations, $i = 1, ..., E$ (E is the number of the state equations), $\Psi_i$ are the adjoint variables and $\Omega$ the computational domain. Differentiation of eq. 2.14 w.r.t. $b_n$ yields

$$\frac{\delta F_{aug}}{\delta b_n} = \frac{\delta F}{\delta b_n} + \frac{\delta}{\delta b_n} \int_{\Omega} \Psi_i R_i d\Omega \tag{2.15}$$

The different formulations of FI and SI (standard or enhanced) result from the different ways the $\delta / \delta b_n$ integral of eq. 2.15 is expanded.

According to the FI approach the term is developed as follows

$$\frac{\delta}{\delta b_n} \int_{\Omega} \Psi_i R_i d\Omega = \int_{\Omega} \Psi_i \frac{\delta R_i}{\delta b_n} d\Omega + \int_{\Omega} \Psi_i R_i \frac{\delta (d\Omega)}{\delta b_n} \tag{2.16}$$

Using the relation $\frac{\delta (d\Omega)}{\delta b_n} = \frac{\partial}{\partial x_k} \left( \frac{\delta x_k}{\delta b_n} \right) d\Omega$, whose proof can be found in [8], eq. 2.15 becomes

$$\frac{\delta F_{aug}}{\delta b_n} = \frac{\delta F}{\delta b_n} + \int_{\Omega} \Psi_i \frac{\delta R_i}{\delta b_n} d\Omega + \int_{\Omega} \Psi_i R_i \frac{\partial}{\partial x_k} \left( \frac{\delta x_k}{\delta b_n} \right) d\Omega \tag{2.17}$$

By utilising equation $\frac{\delta}{\delta b_n} \left( \frac{\partial \Phi}{\partial x_j} \right) = \frac{\partial}{\partial x_j} \left( \frac{\delta \Phi}{\delta b_n} \right) - \frac{\partial \Phi}{\partial x_k} \frac{\partial}{\partial x_j} \left( \frac{\delta x_k}{\delta b_n} \right)$ (proof also in [8]) and the Green-Gauss theorem, the final FI sensitivity derivatives expression containing the field variations of $x_k$ is received. This is the basis of the so-called *FI* adjoint formulation.

The appearance of the grid-sensitivities can be circumvented by using the SI approach. In this case, the Leibniz theorem for the differentiation of volume integrals with variable boundaries is used. Leibniz theorem yields

$$\frac{\delta}{\delta b_n} \int_{\Omega} \Phi \delta \Omega = \int_{\Omega} \frac{\partial \Phi}{\partial b_n} \delta \Omega + \int_{S} \Phi n_k \frac{\delta x_k}{\delta b_n} \delta S \tag{2.18}$$

where $\Phi$ is an arbitrary quantity and $S = S(\mathbf{b}) = \partial \Omega$.

Eq. 2.15 becomes

$$\frac{\delta F_{aug}}{\delta b_n} = \frac{\delta F}{\delta b_n} + \int_\Omega \Psi_i \frac{\partial R_i}{\partial b_n} d\Omega + \underbrace{\int_S \Psi_i R_i n_k \frac{\delta x_k}{\delta b_n} dS}_{LBterm} \tag{2.19}$$

which is the origin of the so-called *SI* formulation. The last integral on the RHS of eq. 2.19 is the so-called Leibniz term (LBterm in short). In literature, the LBterm is usually ignored [11] under the assumption that the flow equations are satisfied not only inside the flow domain but also along the boundaries. Indeed, on very fine grids, where the flow PDEs are satisfied very close to the wall and their residuals along S are for this reason very small, neglecting LBterm does not affect the accuracy of SD computed by SI. However, depending on the case and the grid coarseness, the residuals of the primal equations might not be negligible along the boundary, making LBterm very important for the accurate computation of the SD. Nevertheless, its straightforward inclusion in the SD expression in not the appropriate treatment, because of the numerical difficulties related with the computation of the residuals of the primal PDEs along the boundary.

As proved in [11], LBterm can be replaced by a volume integral containing the grid sensitivities, which is much more easily computable.

$$\int_S \Psi_i R_i n_k \frac{\delta x_k}{\delta b_n} dS = -\int_\Omega \frac{\partial}{\partial x_j} \left\{ -u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} u_i \frac{\partial \tau_{ij}}{\partial x_k} + q \frac{\partial v_j}{\partial x_k} \right\} \frac{\delta x_k}{\delta b_n} \delta\Omega \tag{2.20}$$

where $\tau_{ij} = \nu \left( \frac{\partial v_i}{\partial x_j} \frac{\partial v_j}{\partial x_i} \right)$ is the stress tensor and $\tau_{ij}^a = \nu \left( \frac{\partial u_i}{\partial x_j} \frac{\partial u_j}{\partial x_i} \right)$ is the adjoint stress tensor.

However, the cost of computing this field integral scales linearly with the number of design variables, because of the requirement for $\delta x_k / \delta b_n$ in $\Omega$ for all design variables. This leads to a computational cost similar to that of the FI approach. The Enhanced SI (E-SI) formulation aims at the elimination of the field integrals of $\delta x_k / \delta b_n$ in the SD expression by solving the adjoint to a hypothetical grid displacement PDE. The grid displacement PDE (gdPDE) used in this thesis is a Laplace equation, which reads

$$R_i^m = \frac{\partial^2 m_i}{\partial x_j^2} = 0 \tag{2.21}$$

where $m_i$ are the Cartesian displacements of the grid nodes. Along the boundary, $m_i$ represents the displacement of the boundary points.

Following the adjoint methodology, to derive the adjoint gdPDE, a new term is added to the augmented function of eq. 2.14, containing the field integral of the

laplacian grid displacement PDE multiplied by the adjoint to $m_i$ variable. The resulting $\delta/\delta b_n$ field integrals are expanded using the Leibniz theorem. A thorough analysis for the E-SI approach that is used in this thesis is done in the following subsection.

### 2.2.2 The E-SI Continuous Adjoint Method

The generalised augmented objective function, eq. 2.14, is specified based on the primal problem described in section 2.1. The extra field integral of the laplacian grid displacement PDE is also included since the analysis is based on the E-SI continuous adjoint approach.

$$F_{aug} = F + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega + \int_\Omega \widetilde{\nu_a} R^{\widetilde{\nu}} d\Omega + \int_\Omega m_i^a R_i^m d\Omega \tag{2.22}$$

where $\Omega$ is the computational domain, $u_i$ the adjoint velocity, $q$ the adjoint pressure, $\widetilde{\nu_a}$ the adjoint turbulence (or adjoint Spalart-Allmaras) variable and $m_i^a$ the adjoint to $m_i$ variable. It should be noted that the third integral of eq. 2.22 would be excluded if the 'frozen turbulence' assumption were made, whereas the fourth integral would be eliminated if the FI or the SI approach was adopted.

By employing the Leibniz and the Green-Gauss theorem we receive

$$\frac{\delta F_{aug}}{\delta b_n} = \frac{\delta F}{\delta b_n} + \frac{\delta}{\delta b_n} \int_\Omega \left( u_i R_i^v + q R^p + \widetilde{\nu_a} R^{\widetilde{\nu}} + m_i^a R_i^m \right) d\Omega \tag{2.23}$$

$$\begin{aligned} &= \frac{\delta F}{\delta b_n} + \int_\Omega u_i \frac{\partial R_i^v}{\partial b_n} d\Omega + \int_\Omega q \frac{\partial R^p}{\partial b_n} d\Omega + \int_\Omega \widetilde{\nu_a} \frac{\partial R^{\widetilde{\nu}}}{\partial b_n} d\Omega \\ &+ \int_S m_i^a n_j \frac{\partial}{\partial x_j} \left( \frac{\delta x_i}{\delta b_n} \right) dS - \int_{S_W} \frac{\partial m_i^a}{\partial x_j} n_j \frac{\delta x_i}{\delta b_n} dS + \int_\Omega \frac{\partial^2 m_i^a}{\partial x_j^2} \frac{\delta x_i}{\delta b_n} \delta\Omega \\ &+ \int_S \left( u_i R_i^v + q R^p + \widetilde{\nu_a} R^{\widetilde{\nu}} + m_i^a R_i^m \right) n_k \frac{\delta x_k}{\delta b_n} dS \end{aligned} \tag{2.24}$$

where $S$ is the boundary of the computational domain, $S = S_I \cup S_O \cup S_W \cup S_{W_P}$. The boundaries $S_I, S_O, S_W$ and $S_{W_P}$ refer to the inlet, outlet, fixed and controlled (thus parameterized) wall boundaries of the domain, respectively. Also, $n_k$ stands for the components of the unit outward vector which is normal to the surface. Since the only parameterized boundary is $S_{W_P}$ and for the non-controlled boundaries $\delta x_k/\delta b_n = 0$,

we have

$$\begin{aligned}\frac{\delta F_{aug}}{\delta b_n} = &\frac{\delta F}{\delta b_n} + \int_\Omega u_i \frac{\partial R_i^v}{\partial b_n} d\Omega + \int_\Omega q \frac{\partial R^p}{\partial b_n} d\Omega + \int_\Omega \widetilde{\nu_a} \frac{\partial R^{\widetilde{\nu}}}{\partial b_n} d\Omega \\ &+ \int_{S_{W_P}} m_i^a n_j \frac{\partial}{\partial x_j}\left(\frac{\delta x_i}{\delta b_n}\right) dS - \int_{S_{W_P}} \frac{\partial m_i^a}{\partial x_j} n_j \frac{\delta x_i}{\delta b_n} dS + \int_\Omega \frac{\partial^2 m_i^a}{\partial x_j^2} \frac{\delta x_i}{\delta b_n} \delta\Omega \\ &+ \int_{S_{W_P}} \left(u_i R_i^v + q R^p + \widetilde{\nu_a} R^{\widetilde{\nu}} + m_i^a R_i^m\right) n_k \frac{\delta x_k}{\delta b_n} dS \end{aligned} \tag{2.25}$$

Since the residuals of the primal equations must be zero over the whole domain, $F_{aug} = F$ and consequently $\frac{\delta F_{aug}}{\delta b_n} = \frac{\delta F}{\delta b_n}$.

A sharp distinction between symbols $\delta()/\delta b_n$ and $\partial()/\partial b_n$ must be made. $\delta\Phi/\delta b_n$ denotes the total (or material) derivative of an arbitrary quantity $\Phi$ and represents the total change in $\Phi$ by varying $b_n$, whereas $\partial\Phi/\partial b_n$ denotes the partial derivative of $\Phi$ and represents the variation in $\Phi$ due to changes in the flow variables excluding the contributions from the space deformation. $\delta\Phi/\delta b_n$ and $\partial\Phi/\partial b_n$ are related with the following expression depending if they are computed on the interior of $\Omega$ or on the boundary of $\Omega$.

Interior of $\Omega$

$$\frac{\delta\Phi}{\delta b_n} = \frac{\partial\Phi}{\partial b_n} + \frac{\partial\Phi}{\partial x_k}\frac{\delta x_k}{\delta b_n} \tag{2.26}$$

Surface - Boundary of $\Omega$

$$\frac{\delta_s\Phi}{\delta b_n} = \frac{\partial\Phi}{\partial b_n} + \frac{\partial\Phi}{\partial x_k} n_k \frac{\delta x_m}{\delta b_n} n_m \tag{2.27}$$

Before proceeding with analysing the integrals appearing on the RHS of eq. 2.25 the following observation must be made. Since $\partial()/\partial b_n$ takes into account only changes in the flow variables and excludes changes in the shape/volume of the flow domain, spatial differentiation and partial differentiation w.r.t. the design variables can commute, i.e.

$$\frac{\partial}{\partial b_n}\left(\frac{\partial\phi}{\partial x_j}\right) = \frac{\partial}{\partial x_j}\left(\frac{\partial\phi}{\partial b_n}\right) \tag{2.28}$$

In general, this is not valid for the total derivative, i.e.

$$\frac{\delta}{\delta b_n}\left(\frac{\partial\phi}{\partial x_j}\right) \neq \frac{\partial}{\partial x_j}\left(\frac{\delta\phi}{\delta b_n}\right) \tag{2.29}$$

### 2.2.3 Differentiation of the Objective Function

In the present study three different objective functions are used that are defined either on a volume of the computational domain or on specific surfaces of it. A general objective function that comprises both surface and volume integrals can be expressed as

$$F = \int_S F_{S_i} n_i dS + \int_\Omega F_\Omega d\Omega \tag{2.30}$$

where $F_{S_i}$ and $F_\Omega$ are the integrands on the boundary and the volume of the computational domain, respectively. Differentiation of $F$ w.r.t. $b_n$ gives

$$\frac{\delta F}{\delta b_n} = \frac{\delta}{\delta b_n} \int_S F_{S_i} n_i dS + \frac{\delta}{\delta b_n} \int_\Omega F_\Omega d\Omega \tag{2.31}$$

The detailed derivation of the expressions for the surface and volume integral on the RHS of eq. 2.31 can be found in [8]. The surface integral is written as

$$\begin{aligned}\frac{\delta}{\delta b_n} \int_S F_{S_i} n_i dS = &\int_S \frac{\partial F_{S_i}}{\partial v_k} n_i \frac{\partial v_k}{\partial b_n} dS + \int_S \frac{\partial F_{S_i}}{\partial p} n_i \frac{\partial p}{\partial b_n} dS + \int_S \frac{\partial F_{S_i}}{\partial \tau_{kj}} n_i \frac{\partial \tau_{kj}}{\partial b_n} dS \\ &+ \int_S \frac{\partial F_{S_i}}{\partial \widetilde{\nu}} n_i \frac{\partial \widetilde{\nu}}{\partial b_n} dS + \int_S n_i \frac{\partial F_{S_i}}{\partial x_m} \frac{\delta x_k}{\delta b_n} n_k dS \\ &+ \int_S F_{S_i} \frac{\delta n_i}{\delta b_n} dS + \int_S F_{S_i} n_i \frac{\delta (dS)}{\delta b_n}\end{aligned} \tag{2.32}$$

The volume integral, after applying the Leibniz theorem for the differentiation of volume integrals with moving boundaries, becomes

$$\frac{\delta}{\delta b_n} \int_\Omega F_\Omega d\Omega = \int_\Omega \frac{\partial F_\Omega}{\partial b_n} d\Omega + \int_S F_\Omega n_k \frac{\delta x_k}{\delta b_n} dS \tag{2.33}$$

Eq. 2.33 can be expanded if the dependency of $F$ on the flow variables is taken into account. In general, F may contain the flow variables as well as differential operators of these quantities. Considering these, eq. 2.33 becomes

$$\begin{aligned}\frac{\delta}{\delta b_n} \int_\Omega F_\Omega d\Omega = &\int_\Omega \acute{F}^v_{\Omega,i} \frac{\partial v_i}{\partial b_n} d\Omega + \int_\Omega \acute{F}^p_\Omega \frac{\partial p}{\partial b_n} d\Omega + \int_\Omega \acute{F}^{\widetilde{\nu}}_\Omega \frac{\partial \widetilde{\nu}}{\partial b_n} d\Omega + \int_S \acute{F}^v_{S,i} \frac{\partial v_i}{\partial b_n} dS \\ &+ \int_S \acute{F}^p_S \frac{\partial p}{\partial b_n} dS \int_S \acute{F}^{\widetilde{\nu}}_S \frac{\partial \widetilde{\nu}}{\partial b_n} dS \int_S F_\Omega n_k \frac{\delta x_k}{\delta b_n} dS\end{aligned} \tag{2.34}$$

where $\acute{F}^\Phi_\Omega$ includes the partial derivatives $\partial F_\Omega / \partial \Phi$ plus any term that might result from the use of the Green-Gauss theorem for integrals of the form $\int_\Omega \frac{\partial}{\partial b_n} \left( \frac{\partial \Phi}{\partial x_j} \right) d\Omega$.

By substituting the above expressions in eq. 2.31 the final expression of $\delta F/\delta b_n$ is reached

$$\begin{aligned}\frac{\delta F}{\delta b_n} =& \int_\Omega \acute{F}^v_{\Omega,i}\frac{\partial v_i}{\partial b_n}d\Omega + \int_\Omega \acute{F}^p_\Omega \frac{\partial p}{\partial b_n}d\Omega + \int_\Omega \acute{F}^{\widetilde{\nu}}_\Omega \frac{\partial \widetilde{\nu}}{\partial b_n}d\Omega + \int_S \left(\acute{F}^v_{S,i} + \frac{\partial F_{S_k}}{\partial v_i}n_k\right)\frac{\partial v_i}{\partial b_n}dS \\ &+ \int_S \left(\acute{F}^p_S + \frac{\partial F_{S_i}}{\partial p}n_i\right)\frac{\partial p}{\partial b_n}dS + \int_S \left(\acute{F}^{\widetilde{\nu}}_S + \frac{\partial F_{S_i}}{\partial \widetilde{\nu}}n_i\right)\frac{\partial \widetilde{\nu}}{\partial b_n}dS \\ &+ \int_S \frac{\partial F_{S_k}}{\partial \tau_{ij}}n_k\frac{\partial \tau_{ij}}{\partial b_n}dS + \int_{S_{W_p}} F_\Omega n_k \frac{\delta x_k}{\delta b_n}dS + \int_{S_{W_p}} n_i \frac{\partial F_{S_{W_p,i}}}{\partial x_m}n_m\frac{\delta x_k}{\delta b_n}n_k dS \\ &+ \int_{S_{W_p}} F_{S_{W_p,i}}\frac{\delta n_i}{\delta b_n}dS + \int_{S_{W_p}} F_{S_{W_p,i}}n_i\frac{\delta(dS)}{\delta b_n}\end{aligned} \tag{2.35}$$

The above equation includes the partial derivatives of the flow variables w.r.t. the design variables, the computation of which requires the solution of N systems of equations similar to the Navier-Stokes equations. Aim of the adjoint method is to circumvent this expensive computation by solving the adjoint equations.

The previous analysis can be specified for each of the three objective functions used in this thesis.

### Total Pressure Losses

The objective function used to minimize the total pressure losses between the inlet and outlet boundaries of the fluid domain is given by

$$F_{P_t} = -\int_{S_{I,O}} \left(p + \frac{1}{2}v_i^2\right) v_j n_j dS \tag{2.36}$$

where $n_i$ are the components of the outwards pointing unit normal vector. $S_{I,O}$ denotes the inlet and outlet boundaries of the domain. As defined in eq. 2.36, the units of $F_{P_t}$ are $m^5/s^3$, i.e. power losses per unit of density.

Differentiation of the above objective function yields

$$\begin{aligned}\frac{\delta F_{P_t}}{\delta b_n} =& -\int_{S_{I,O}} \frac{\delta}{\delta b_n}\left(p + \frac{1}{2}v_i^2\right) v_j n_j dS - \int_{S_{I,O}} \left(p + \frac{1}{2}v_i^2\right)\frac{\delta v_j}{\delta b_n}n_j dS \\ &- \int_{S_{I,O}} \left(p + \frac{1}{2}v_i^2\right) v_j \frac{\delta(n_j dS)}{\delta b_n}\end{aligned} \tag{2.37}$$

Because the integrals of eq. 2.37 are defined over the domain inlet and outlet boundaries, which are fixed during the optimization, the third integral of eq. 2.37 is zero, since $\frac{\delta(n_j dS)}{\delta b_n} = 0$ over non-controlled patches.

Therefore, we have

$$\begin{aligned}
\frac{\delta F_{P_t}}{\delta b_n} &= -\int_{S_{I,O}} \frac{\delta}{\delta b_n}\left(p + \frac{1}{2}v_i^2\right) v_j n_j dS - \int_{S_{I,O}} \left(p + \frac{1}{2}v_i^2\right) \frac{\delta v_j}{\delta b_n} n_j dS \\
&= -\int_{S_{I,O}} \frac{\delta p}{\delta b_n} v_j n_j dS - \int_{S_{I,O}} \frac{\delta}{\delta b_n}\left(\frac{1}{2}v_i^2\right) v_j n_j dS \\
&- \int_{S_{I,O}} \left(p + \frac{1}{2}v_i^2\right) \frac{\delta v_j}{\delta b_n} n_j dS \\
&= \int_{S_{I,O}} \left(-v_j n_j\right) \frac{\delta p}{\delta b_n} dS + \int_{S_{I,O}} \left[-v_i v_j n_j - \left(p + \frac{1}{2}v_i^2\right) n_i\right] \frac{\delta v_i}{\delta b_n} dS
\end{aligned} \tag{2.38}$$

**Noise**

In order to minimise the noise received by an observer at the outlet of the domain, a surrogate noise function is defined. Based on industrial experience, an appropriate objective function for this purpose depends exclusively on the turbulence viscosity and is given by the following formula [12].

$$F_{\nu_t} = \int_{\Omega'} \nu_t^2 d\Omega \tag{2.39}$$

where $\nu_t$ is the turbulent viscosity and $\Omega'$ is a volume area, where it is desired to minimize the turbulence viscosity. Since the noise reduction is achieved by a flow solver without any coupling with an acoustic one, the selection of the appropriate volume area $\Omega'$ is crucial, so that the minimization of the objective function leads also to minimization of the produced noise.

It should be noted that obtaining adjoint-based sensitivities for this function would not have been possible without differentiating the turbulence model since the acquired sensitivities would have zero values irrespective of the body shape.

Differentiation of the above objective function yields

$$\frac{\delta F_{\nu_t}}{\delta b_n} = \frac{\delta}{\delta b_n}\int_{\Omega'} \nu_t^2 d\Omega = \int_{\Omega'} \frac{\partial \nu_t^2}{\partial b_n} d\Omega + \int_{S'} \nu_t^2 n_k \frac{\delta x_k}{\delta b_n} dS$$
$$= \int_{\Omega'} 2\nu_t \frac{\partial \nu_t}{\partial \widetilde{\nu}} \frac{\partial \widetilde{\nu}}{\partial b_n} d\Omega + \int_{S'} \nu_t^2 n_k \frac{\delta x_k}{\delta b_n} dS \tag{2.40}$$

where the Leibniz theorem is used.

The volume integral of eq. 2.40 contributes to the Field Adjoint Equations and more specifically to the adjoint turbulence model equation, whereas the surface integral to the Sensitivity Derivatives.

**Flow Uniformity**

The flow uniformity objective is used to drive the velocity at an outlet patch of the domain to the mean value over the same patch, by minimizing the standard deviation of each velocity component over the outlet patch. The objective function that has to be minimized in order to achieve that is

$$F_\gamma = \frac{1}{2}\int_{S_O} \left(v_i - \bar{v}_i\right)^2 dS \tag{2.41}$$

where $\bar{v}_i$ is the mean value of the velocity component $v_i$ over the outlet of the domain $S_O$.

Differentiation of eq. 2.41 yields

$$\frac{\delta F_\gamma}{\delta b_n} = \frac{1}{2}\int_{S_O} \frac{\delta}{\delta b_n}\left(v_i - \bar{v}_i\right)^2 dS + \frac{1}{2}\int_{S_O} \left(v_i - \bar{v}_i\right)^2 \frac{\delta (dS)}{\delta b_n} \tag{2.42}$$

Since the outlet of the domain is non-controlled, $\frac{\delta (dS)}{\delta b_n} = 0$ and the second integral of eq. 2.42 is zero.

Therefore, we have

$$\frac{\delta F_\gamma}{\delta b_n} = \int_{S_O} \left(v_i - \bar{v}_i\right) \frac{\delta v_i}{\delta b_n} dS - \int_{S_O} \left(v_i - \bar{v}_i\right) \frac{\delta \bar{v}_i}{\delta b_n} dS \tag{2.43}$$

Since term $\bar{v}_i$ is constant in $S_O$, eq. 2.43 becomes

$$\frac{\delta F_\gamma}{\delta b_n} = \int_{S_O} (v_i - \bar{v}_i) \frac{\delta v_i}{\delta b_n} dS - \frac{\delta \bar{v}_i}{\delta b_n} \int_{S_O} (v_i - \bar{v}_i)\, dS \tag{2.44}$$

where the second surface integral is by default equal to zero. Hence, the above equation is simplified to the following one.

$$\frac{\delta F_\gamma}{\delta b_n} = \int_{S_O} (v_i - \bar{v}_i) \frac{\delta v_i}{\delta b_n} dS \tag{2.45}$$

A more intuitive scale of the outlet flow uniformity is provided by the *uniformity index*, defined as:

$$\gamma = 1 - \frac{\int_{S_O} ||v| - |\bar{v}||\, dS}{2\,|\bar{v}|\, S_O} \tag{2.46}$$

The uniformity index is a dimensionless quantity and has a value that varies between 0 and 1, with 1 indicating perfect uniformity.

### 2.2.4 Differentiation of the Primal Equations

In order to obtain the final expression of $\delta F_{aug}/\delta b_n$ from eq. 2.25, the partial derivatives of the primal equations w.r.t. the design variables have to be analysed, i.e. $\partial R^p/\partial b_n$, $\partial R^v_i/\partial b_n$ and $\partial R^{\widetilde{\nu}}/\partial b_n$.

Differentiation of the continuity and momentum equations, eqs 2.13a and 2.13b respectively, yield

$$\frac{\partial R^p}{\partial b_n} = -\frac{\partial}{\partial x_j}\left(\frac{\partial v_j}{\partial b_n}\right) \tag{2.47}$$

and

$$\begin{aligned}\frac{\partial R^v_i}{\partial b_n} = {} & \frac{\partial v_j}{\partial b_n}\frac{\partial v_i}{\partial x_j} + v_j \frac{\partial}{\partial x_j}\left(\frac{\partial v_i}{\partial b_n}\right) + \frac{\partial}{\partial x_i}\left(\frac{\partial p}{\partial b_n}\right) \\ & - \frac{\partial}{\partial x_j}\left[(\nu + \nu_t)\frac{\partial}{\partial b_n}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] - \frac{\partial}{\partial x_j}\left[\frac{\partial \nu_t}{\partial b_n}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right]\end{aligned} \tag{2.48}$$

where $\partial\nu_t/\partial b_n$ can be computed as follows

$$\frac{\partial \nu_t}{\partial b_n} = \frac{\partial \nu_t}{\partial \widetilde{\nu}} \frac{\partial \widetilde{\nu}}{\partial b_n} \quad , with \quad \frac{\partial \nu_t}{\partial \widetilde{\nu}} = f_{v_1} + \widetilde{\nu} \frac{\partial f_{v_1}}{\partial \widetilde{\nu}} = f_{v_1} + \frac{3c_{v_1}^3 \chi^2}{\left(\chi^3 + c_{v_1}^3\right)^2} \tag{2.49}$$

Differentiation of the turbulence model equation, eq. 2.13c, yields

$$\begin{aligned} \frac{\delta R^{\widetilde{\nu}}}{\delta b_n} = & \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{\partial v_j}{\partial b_n} + v_j \frac{\partial}{\partial x_j} \left( \frac{\partial \widetilde{\nu}}{\partial b_n} \right) - \frac{\partial}{\partial x_j} \left[ \left( \nu + \frac{\widetilde{\nu}}{\sigma} \right) \frac{\partial}{\partial x_j} \frac{\partial \widetilde{\nu}}{\partial b_n} \right] \\ & - \frac{1}{\sigma} \frac{\partial}{\partial x_j} \left( \frac{\partial \widetilde{\nu}}{\partial b_n} \frac{\partial \widetilde{\nu}}{\partial x_j} \right) - 2 \frac{c_{b2}}{\sigma} \frac{\partial \widetilde{\nu}}{\partial x_j} \frac{\partial}{\partial x_j} \left( \frac{\partial \widetilde{\nu}}{\partial b_n} \right) \\ & + \widetilde{\nu} \left( -\frac{\partial P}{\partial b_n} + \frac{\partial D}{\partial b_n} \right) + (-P + D) \frac{\partial \widetilde{\nu}}{\partial x_j} \end{aligned} \tag{2.50}$$

The differentiation of the production and dissipation terms, eq. 2.8, yields

$$-\frac{\partial P}{\partial b_n} + \frac{\partial D}{\partial b_n} = \mathcal{C}_{\widetilde{\nu}} \frac{\partial \widetilde{\nu}}{\partial b_n} + \mathcal{C}_{\Delta} \frac{\partial \Delta}{\partial b_n} + \mathcal{C}_Y \frac{1}{Y} e_{mjk} \frac{\partial v_k}{\partial x_j} e_{mli} \frac{\partial}{\partial b_n} \left( \frac{\partial v_i}{\partial x_l} \right) \tag{2.51}$$

where

$$\mathcal{C}_Y = \left( -c_{b_1} - c_{w_1} \mathcal{C} \frac{r}{\widetilde{Y}} \right) f_{v_3} \tag{2.52}$$

$$\mathcal{C}_{\Delta} = -\frac{2}{\Delta^3} \left[ c_{w_1} r \mathcal{C} \left( \Delta^2 - \frac{\widetilde{\nu} f_{v_2}}{\kappa^2 \widetilde{Y}} \right) + c_{w_1} f_w \widetilde{\nu} - c_{b_1} \frac{f_{v_2}}{\kappa^2} \widetilde{\nu} \right] \tag{2.53}$$

$$\mathcal{C}_{\widetilde{\nu}} = \left( -c_{b_1} - c_{w_1} \mathcal{C} \frac{r}{\widetilde{Y}} \right) \left( \frac{\partial f_{v_3}}{\partial \widetilde{\nu}} Y + \frac{f_{v_2}}{\kappa^2 \Delta^2} + \frac{\partial f_{v_2}}{\partial \widetilde{\nu}} \frac{\widetilde{\nu}}{\kappa^2 \Delta^2} \right) + c_{w_1} \mathcal{C} \frac{r}{\widetilde{\nu}} + c_{w_1} \frac{f_w}{\Delta^2} \tag{2.54}$$

$$C = \frac{c_{w_1} \widetilde{\nu}^2}{\Delta^2} \left[ 1 + c_{w_2} (6r^5 - 1) \right] \frac{c_{w_3}^6}{g^6 + c_{w_3}^6} \left( \frac{1 + c_{w_3}^6}{g^6 + c_{w_3}^6} \right)^{1/6} \tag{2.55}$$

$$\frac{\partial f_{v_2}}{\partial \widetilde{\nu}} = -\frac{3}{\nu c_{v_2}} \left( 1 + \frac{\chi}{c_{v_2}} \right)^{-4} \tag{2.56}$$

$$\begin{aligned} \frac{\partial f_{v_3}}{\partial \widetilde{\nu}} = & \frac{1}{c_{v_2}} \left( \frac{f_{v_1}}{\nu} + \chi \frac{\partial f_{v_1}}{\partial \nu} \right) \left[ 3 \left( 1 + \frac{\chi}{c_{v_2}} \right) + \left( \frac{\chi}{c_{v_2}} \right)^2 \right] \left( 1 + \frac{\chi}{c_{v_2}} \right)^{-3} \\ & + \frac{1}{\nu c_{v_2}^2} (1 + \chi f_{v_1}) \left( 3 + 2 \frac{\chi}{c_{v_2}} \right) c_{v_2}^2 \left( 1 + \frac{\chi}{c_{v_2}} \right)^{-3} \\ & - 3 \frac{(1 + \chi f_{v_1})}{\nu c_{v_2}^2} \left[ 3 \left( 1 + \frac{\chi}{c_{v_2}} \right) + \left( \frac{\chi}{c_{v_2}} \right)^2 \right] \left( 1 + \frac{\chi}{c_{v_2}} \right)^{-4} \end{aligned} \tag{2.57}$$

The continuous adjoint to the Spalart-Allmaras model can be found in much greater detail in [8, 13].

### 2.2.5 Field Adjoint Equations and Adjoint Boundary Conditions

By substituting eqs. 2.47, 2.48 and 2.50 into eq. 2.25 we receive the final expression of the material derivative of the augmented objective function w.r.t. the design variables.

$$\begin{aligned}\frac{\delta F_{aug}}{\delta b_n} = &\int_S \mathcal{BC}_i^u \frac{\partial v_i}{\partial b_n} dS + \int_S \mathcal{BC}^p \frac{\partial p}{\partial b_n} dS + \int_S \mathcal{BC}^{\widetilde{\nu_a}} \frac{\partial \widetilde{\nu}}{\partial b_n} dS + \int_S \mathcal{BC}^{m_a} \frac{\partial}{\partial x_j}\left(\frac{\delta x_i}{\delta b_n}\right) dS \\ &+ \int_S (-u_i n_j + \frac{\partial F_{S_k}}{\partial \tau_{ij}} n_k) \frac{\partial \tau_{ij}}{\partial b_n} dS - \int_S \widetilde{\nu}_a \left(\nu + \frac{\widetilde{\nu}}{\sigma}\right) \frac{\partial}{\partial b_n}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right) n_j dS \\ &+ \int_\Omega R_i^u \frac{\partial v_i}{\partial b_n} d\Omega + \int_\Omega R^q \frac{\partial p}{\partial b_n} d\Omega + \int_\Omega R^{\widetilde{\nu_a}} \frac{\partial \widetilde{\nu}}{\partial b_n} d\Omega + \int_\Omega R_k^{m^a} \frac{\delta x_k}{\delta b_n} d\Omega \\ &+ \int_{S_{W_p}} n_i \frac{\partial F_{S_{W_p,i}}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k dS + \int_{S_{W_p}} F_{S_{W_p,i}} \frac{\delta n_i}{\delta b_n} dS + \int_{S_{W_p}} F_{S_{W_p,i}} n_i \frac{\delta (dS)}{\delta b_n} \\ &- \int_{S_{W_P}} \frac{\partial m_i^a}{\partial x_j} n_j \frac{\delta x_i}{\delta b_n} dS + \int_\Omega \widetilde{\nu}\widetilde{\nu}_a \mathcal{C}_\Delta \frac{\partial \Delta}{\partial b_n} d\Omega + \int_S m_i^a R_i^m n_k \frac{\delta x_k}{\delta b_n} dS \end{aligned} \tag{2.58}$$

where

$$\begin{aligned}\mathcal{BC}_i^u = &u_i v_j n_j + (\nu + \nu_t)\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) n_j - q n_i + \widetilde{\nu}_a \widetilde{\nu} \frac{\mathcal{C}_Y}{Y} e_{mjk} \frac{\partial v_k}{\partial x_j} e_{mli} n_l \\ &+ \frac{\partial F_{S_k}}{\partial v_i} n_k + \acute{F}_{S,i}^v \end{aligned} \tag{2.59}$$

$$\mathcal{BC}^p = u_j n_j + \frac{\partial F_{S_i}}{\partial p} n_i + \acute{F}_S^p \tag{2.60}$$

$$\mathcal{BC}^{\widetilde{\nu_a}} = \widetilde{\nu}_a v_j n_j + \left(\nu + \frac{\widetilde{\nu}}{\sigma}\right) \frac{\partial \widetilde{\nu}_a}{\partial x_j} n_j - \frac{\widetilde{\nu}_a}{\sigma}(1 + 2c_{b_2}) \frac{\partial \widetilde{\nu}}{\partial x_j} n_j + \frac{\partial F_{S_k}}{\partial \widetilde{\nu}} n_k + \acute{F}_S^{\widetilde{\nu}} \tag{2.61}$$

$$\mathcal{BC}^{m_a} = m_i^a n_j \tag{2.62}$$

After setting the multipliers of $\partial v_i / \partial b_n$, $\partial p / \partial b_n$, $\partial \widetilde{\nu} / \partial b_n$ and $\delta x_k / \delta b_n$, in the volume integrals of eq. 2.58 to zero, the field adjoint equations are derived.

$$R^q = -\frac{\partial u_j}{\partial x_j} + \acute{F}^p_\Omega = 0 \tag{2.63}$$

$$R^u_i = u_j\frac{\partial v_j}{\partial x_i} - \frac{\partial (v_j u_i)}{\partial x_j} - \frac{\partial}{\partial x_j}\left[(\nu+\nu_t)\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right] + \frac{\partial q}{\partial x_i} + \acute{F}^v_{\Omega,i}$$
$$+\widetilde{\nu}_a\frac{\partial \widetilde{\nu}}{\partial x_i} - \frac{\partial}{\partial x_l}\left(\widetilde{\nu}_a\widetilde{\nu}\frac{\mathcal{C}_Y}{Y}e_{mjk}\frac{\partial v_k}{\partial x_j}e_{mli}\right) = 0\ ,\ i=1,2,3 \tag{2.64}$$

$$R^{\widetilde{\nu_a}} = -\frac{\partial (v_j\widetilde{\nu_a})}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\nu+\frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu_a}}{\partial x_j}\right] + \frac{1}{\sigma}\frac{\partial \widetilde{\nu_a}}{\partial x_j}\frac{\partial \widetilde{\nu}}{\partial x_j} + 2\frac{c_{b2}}{\sigma}\frac{\partial}{\partial x_j}\left(\widetilde{\nu_a}\frac{\partial \widetilde{\nu}}{\partial x_j}\right)$$
$$+\widetilde{\nu}_a\widetilde{\nu}\mathcal{C}_{\widetilde{\nu}} + \frac{\partial \nu_t}{\partial \widetilde{\nu}}\frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right) + (-P+D)\,\widetilde{\nu}_a + \acute{F}^{\widetilde{\nu}}_\Omega = 0 \tag{2.65}$$

$$R^{m^a}_k = \frac{\partial^2 m^a_k}{\partial x_j^2} + \frac{\partial}{\partial x_j}\left\{u_i v_j\frac{\partial v_i}{\partial x_k} + u_j\frac{\partial p}{\partial x_k} + \tau^a_{ij}\frac{\partial u_i}{\partial x_k} - u_i\frac{\partial \tau_{ij}}{\partial x_k} - q\frac{\partial v_j}{\partial x_k}\right\} = 0 \tag{2.66}$$

After satisfying the field adjoint equations, the remaining terms in eq. 2.58 are

$$\frac{\delta F_{aug}}{\delta b_n} = \int_S \mathcal{BC}^u_i\frac{\partial v_i}{\partial b_n}dS + \int_S \mathcal{BC}^p\frac{\partial p}{\partial b_n}dS + \int_S \mathcal{BC}^{\widetilde{\nu_a}}\frac{\partial \widetilde{\nu}}{\partial b_n}dS + \int_S \mathcal{BC}^{m_a}\frac{\partial}{\partial x_j}\left(\frac{\delta x_i}{\delta b_n}\right)dS$$
$$+\int_S(-u_i n_j + \frac{\partial F_{S_k}}{\partial \tau_{ij}}n_k)\frac{\partial \tau_{ij}}{\partial b_n}dS - \int_S \widetilde{\nu_a}\left(\nu+\frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial}{\partial b_n}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right)n_j dS$$
$$+\int_{S_{W_p}} n_i\frac{\partial F_{S_{W_p,i}}}{\partial x_m}n_m\frac{\delta x_k}{\delta b_n}n_k dS + \int_{S_{W_p}} F_{S_{W_p,i}}\frac{\delta n_i}{\delta b_n}dS + \int_{S_{W_p}} F_{S_{W_p,i}}n_i\frac{\delta(dS)}{\delta b_n}$$
$$-\int_{S_{W_P}}\frac{\partial m^a_i}{\partial x_j}n_j\frac{\delta x_i}{\delta b_n}dS + \int_\Omega \widetilde{\nu}\widetilde{\nu_a}\mathcal{C}_\Delta\frac{\partial \Delta}{\partial b_n}d\Omega + \int_S m^a_i R^m_i n_k\frac{\delta x_k}{\delta b_n}dS \tag{2.67}$$

The system of the field adjoint PDEs is closed with the adjoint boundary conditions. The adjoint boundary conditions (ABC) are imposed aiming to eliminate the surface integrals that contain the partial derivatives of the state variables w.r.t. the design variables, namely the first six and the last integral of eq. 2.67 which contain the surface integrals of $\partial v_i/\partial b_n$, $\partial p/\partial b_n$, $\partial v_i/\partial b_n$, $\partial\widetilde{\nu}/\partial b_n$,$\partial(\delta x_i/\delta b_n)/\partial x_j$, $\partial\tau_{ij}/\partial b_n$, $\partial(\partial\widetilde{\nu}/\partial x_j)/\partial b_n$ and $\delta x_i/\delta b_n$. For the sake of completeness these terms are rewritten as follows.

$$I_1 = \int_S \mathcal{BC}_i^u \frac{\partial v_i}{\partial b_n} dS \tag{2.68}$$

$$I_2 = \int_S \mathcal{BC}^p \frac{\partial p}{\partial b_n} dS \tag{2.69}$$

$$I_3 = \int_S \mathcal{BC}^{\widetilde{\nu_a}} \frac{\partial \widetilde{\nu}}{\partial b_n} dS \tag{2.70}$$

$$I_4 = \int_S \mathcal{BC}^{m_a} \frac{\partial}{\partial x_j} \left( \frac{\delta x_i}{\delta b_n} \right) dS \tag{2.71}$$

$$I_5 = \int_S (-u_i n_j + \frac{\partial F_{S_k}}{\partial \tau_{ij}} n_k) \frac{\partial \tau_{ij}}{\partial b_n} dS \tag{2.72}$$

$$I_6 = \int_S \widetilde{\nu_a} \left( \nu + \frac{\widetilde{\nu}}{\sigma} \right) \frac{\partial}{\partial b_n} \left( \frac{\partial \widetilde{\nu}}{\partial x_j} \right) n_j dS \tag{2.73}$$

$$I_7 = \int_S m_i^a R_i^m n_k \frac{\delta x_k}{\delta b_n} dS \tag{2.74}$$

At this point, the final expressions of the adjoint boundary conditions are presented, whereas their detailed deriviation can be found in [8]. The adjoint boundary condition of the adjoint gdPDEs is the same for all boundaries, namely $m_k^a = 0$, so that integral $I_4$ is eliminated. Also, since $m_k^a = 0$ along all boundaries, integral $I_7$, which is the equivalent of LBterm discussed in 2.2.1, also vanishes in all boundaries.

**Inlet Boundaries** $S_I$

At the inlet boundaries since Dirichlet boundary conditions are imposed on $v_i$ and $\widetilde{\nu}$, $\delta v_i / \delta b_n = 0$ and $\delta \widetilde{\nu} / \delta b_n = 0$. Since $S_I$ in a non-controlled boundary, $\delta x_k / \delta b_n = 0$ and taking into consideration eq. 2.27, $\partial v_i / \partial b_n = 0$ and $\partial \widetilde{\nu} / \partial b_n = 0$. This means that $I_1 = I_3 = 0$.

Integrals $I_2$ and $I_5$ are eliminated by demanding

$$u_{\langle n \rangle} = -\frac{\partial F_{S_I,j}}{\partial p} n_j \tag{2.75a}$$

$$u^I_{\langle t \rangle} = \frac{\partial F_{S_I,k}}{\partial \tau_{ij}} n_k t_i^I n_j + \frac{\partial F_{S_I,k}}{\partial \tau_{ij}} n_k t_j^I n_i \tag{2.75b}$$

$$u^{II}_{\langle t \rangle} = \frac{\partial F_{S_I,k}}{\partial \tau_{ij}} n_k t_i^{II} n_j + \frac{\partial F_{S_I,k}}{\partial \tau_{ij}} n_k t_j^{II} n_i \tag{2.75c}$$

where $t_i^I, t_i^{II}$ are the components of the tangent to the surface unit vectors. The first tangent vector $t_i^I$ can be defined as an arbitrary unit vector parallel to $S_I$, whereas

$t_i^{II}$ forms an orthogonal system with $n$ and $t_i^I$. Quantities $u_{\langle t\rangle}^I$ and $u_{\langle t\rangle}^{II}$ are the components of the adjoint velocity in the $t_i^I$, $t_i^{II}$ directions respectively. It should be noted that if $F$ is not defined at the inlet of the computational domain, the adjoint velocity components are zero along $S_I$. Integral $I_6$ is zeroed by imposing a zero Dirichlet condition to $\widetilde{\nu_a}$, i.e. $\widetilde{\nu_a} = 0$.

Finally, since no boundary condition for $q$ results from the elimination of any of the seven boundary integrals already discussed, a zero Neumann boundary condition is employed.

**Outlet Boundaries** $S_O$

At the outlet boundaries since a Dirichlet boundary condition is imposed on $p$, $\delta p/\delta b_n = 0$. Since $S_O$ is fixed, $\delta x_k/\delta b_n = 0$ and taking into consideration eq. 2.27, $\partial p/\partial b_n = 0$. As a result, integral $I_2$ vanishes automatically. Due to the distance of the outlet boundary from the controlled area, an almost uniform velocity profile can be assumed along $S_O$, meaning that $\delta\tau_{ij}/\delta b_n = 0$ along $S_O$. Hence, integral $I_5$ can be neglected.

In order to eliminate $I_1$, its integrand quantity is set equal to zero, i.e.

$$\begin{aligned}\mathcal{BC}_i^u = & u_i v_j n_j + (\nu+\nu_t)\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)n_j - q n_i + \widetilde{\nu_a}\widetilde{\nu}\frac{\mathcal{C}_Y}{Y}e_{mjk}\frac{\partial v_k}{\partial x_j}e_{mli}n_l \\ & + \frac{\partial F_{S_k}}{\partial v_i}n_k + \acute{F}_{S,i}^v = 0\end{aligned} \tag{2.76}$$

Eq. 2.76, which can be analysed in three scalar equations, $i = 1, 2, 3$, includes four unknown quantities (the adjoint pressure $q$ and the three components of the adjoint velocity $u_i$). Therefore, one of them may take on an arbitrary value. This is chosen to be the normal component of the adjoint velocity $u_{\langle n\rangle}$, on which a zero Neumann boundary condition is imposed. By multiplying equation 2.76 with $n_i$ a Dirichlet condition for the adjoint pressure is derived

$$\begin{aligned}q = & u_{\langle n\rangle}v_{\langle n\rangle} + 2(\nu+\nu_t)\frac{\partial u_{\langle n\rangle}}{\partial n} + \frac{\partial F_{S_{O,k}}}{\partial v_i}n_i n_k + \acute{F}_{S_{O,i}}^v n_i \\ & + \widetilde{\nu_a}\widetilde{\nu}\frac{\mathcal{C}_Y}{Y}e_{mjk}\frac{\partial v_k}{\partial x_j}e_{mli}n_l n_i = 0\end{aligned} \tag{2.77}$$

The outlet adjoint tangential velocity is computed by multiplying eq. 2.76 with the tangent to the surface vectors $t_i^l$ , $l=1,2$.

$$0=v_{\langle t\rangle}u^l_{\langle t\rangle}+(\nu+\nu_t)\left(\frac{\partial u^l_{\langle t\rangle}}{\partial n}+\frac{\partial u_{\langle n\rangle}}{\partial t^l}\right)+\frac{\partial F_{S_{O,k}}}{\partial v_i}n_k t^l_i+\acute{F}^v_{S_{O,i}}t^l_i$$
$$-\widetilde{\nu_a}\widetilde{\nu}\frac{\mathcal{C}_Y}{Y}e_{mjk}\frac{\partial v_k}{\partial x_j}e_{mzi}n_z t^l_i \ , \ l=1,2 \tag{2.78}$$

Finally, a Robin-type boundary condition is imposed on $\widetilde{\nu_a}$ in order to eliminate integral $I_3$.

$$BC^{\widetilde{\nu_a}}=\widetilde{\nu_a}v_j n_j+\left(\nu+\frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial\widetilde{\nu_a}}{\partial x_j}n_j+\frac{\partial F_{S_{O,k}}}{\partial\widetilde{\nu}}n_k+\acute{F}^{\widetilde{\nu}}_{S_O}=0 \tag{2.79}$$

It must be noted that term $\frac{\widetilde{\nu_a}}{\sigma}\left(1+2c_{b_2}\right)\frac{\partial\widetilde{\nu}}{\partial x_j}n_j$ has been eliminated from eq. 2.79 with regards to eq. 2.61 where $BC^{\widetilde{\nu_a}}$ was originally defined, because of the zero Neumann boundary condition imposed on $\widetilde{\nu}$ for the outlet boundaries.

**Unparameterized/Fixed Wall Boundaries** $S_W$

Since $\widetilde{\nu}$ is equal to zero on the wall boundaries, integral $I_3$ vanishes. However, this is not the case for the gradient of $\widetilde{\nu}$ and in order to eliminate integral $I_6$ a zero Dirichlet boundary condition imposed on $\widetilde{\nu_a}$. The boundary conditions imposed on the adjoint velocity conditions are derived following the same procedure presented for the inlet boundaries. For the sake of completeness these boundary conditions are

$$u_{\langle n\rangle}=-\frac{\partial F_{S_W,j}}{\partial p}n_j \tag{2.80a}$$
$$u^I_{\langle t\rangle}=\frac{\partial F_{S_W,k}}{\partial\tau_{ij}}n_k t^I_i n_j+\frac{\partial F_{S_W,k}}{\partial\tau_{ij}}n_k t^I_j n_i \tag{2.80b}$$
$$u^{II}_{\langle t\rangle}=\frac{\partial F_{S_W,k}}{\partial\tau_{ij}}n_k t^{II}_i n_j+\frac{\partial F_{S_W,k}}{\partial\tau_{ij}}n_k t^{II}_j n_i \tag{2.80c}$$

Finally, a zero Neumann boundary condition is imposed on $q$.

**Parameterized/Controlled Wall Boundaries** $S_{W_P}$

The main difference between parameterized and non-parameterized wall boundaries is the fact that the parameterized boundaries may change during the optimization. Thus, $\delta x_k/\delta b_n\neq 0$ and the total and partial derivatives of the flow quantities are

different and are linked through eq. 2.27. In addition, the total variations in the normal and tangent surface vectors are not zero, contributing extra terms during the formulation of the adjoint boundary conditions [8].

### 2.2.6 Adjoint to the Distance Equation

After satisfying the field adjoint equations along with their adjoint boundary conditions, eq. 2.67 takes the form of eq. 2.81. To this equation are included some extra terms that arise from the derivation of the adjoint boundary conditions at the controlled boundaries [8, 9].

$$
\begin{aligned}
\frac{\delta F_{aug}}{\delta b_n} =& T_{SD}^{WF} - \int_{S_{W_p}} \mathcal{SD}_1 \frac{\partial \tau_{ij}}{\partial x_m} n_j t_i^I n_m n_k \frac{\delta x_k}{\delta b_n} dS - \int_{S_{W_p}} \mathcal{SD}_1 \tau_{ij} \frac{\delta (n_j t_i^I)}{\delta b_n} \frac{\delta x_k}{\delta b_n} dS \\
&+ \int_{S_{W_p}} \mathcal{SD}_{2,i} v_{\langle t \rangle}^I \frac{\delta t_i^I}{\delta b_n} dS - \int_{S_{W_p}} \mathcal{SD}_{2,i} \frac{\partial v_i}{\partial x_m} n_m n_k \frac{\delta x_k}{\delta b_n} dS \\
&- \int_{S_{W_p}} \left[ \left( \nu + \frac{\widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu}_a}{\partial x_j} n_j + \frac{\partial F_{S_z}}{\partial \widetilde{\nu}} n_z + \acute{F}_S^{\widetilde{\nu}} \right] \frac{\partial \widetilde{\nu}}{\partial x_m} n_m n_k \frac{\delta x_k}{\delta b_n} dS \\
&- \int_{S_{W_p}} (-u_{\langle n \rangle} + \phi_{\langle n \rangle \langle n \rangle}) \left( \tau_{ij} \frac{\delta (n_i n_j)}{\delta b_n} + \frac{\partial \tau_{ij}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k n_i n_j \right) dS \\
&- \int_{S_{W_p}} \phi_{\langle t^I \rangle \langle t^I \rangle} \left( \tau_{ij} \frac{\delta (t_i^I t_j^I)}{\delta b_n} + \frac{\partial \tau_{ij}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k t_i^I t_j^I \right) dS \\
&- \int_{S_{W_p}} (\phi_{\langle t^{II} \rangle \langle t^I \rangle} + \phi_{\langle t^I \rangle \langle t^{II} \rangle}) \left( \tau_{ij} \frac{\delta (t_i^{II} t_j^I)}{\delta b_n} + \frac{\partial \tau_{ij}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k t_i^{II} t_j^I \right) dS \\
&- \int_{S_{W_p}} \phi_{\langle t^{II} \rangle \langle t^{II} \rangle} \left( \tau_{ij} \frac{\delta (t_i^{II} t_j^{II})}{\delta b_n} + \frac{\partial \tau_{ij}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k t_i^{II} t_j^{II} \right) dS \\
&+ \int_{S_{W_p}} n_i \frac{\partial F_{S_{W_p,i}}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k dS + \int_{S_{W_p}} F_{S_{W_p,i}} \frac{\delta n_i}{\delta b_n} dS + \int_{S_{W_p}} F_{S_{W_p,i}} n_i \frac{\delta (dS)}{\delta b_n} \\
&- \int_{S_{W_P}} \frac{\partial m_i^a}{\partial x_j} n_j \frac{\delta x_i}{\delta b_n} dS + \int_{S_{W_p}} \mathcal{A}_\Delta^{WF} \frac{\partial \Delta^P}{\partial b_n} dS + \int_{S_W} \mathcal{A}_\Delta^{WF} \frac{\partial \Delta^P}{\partial b_n} dS \\
&+ \int_\Omega \widetilde{\nu} \widetilde{\nu_a} \mathcal{C}_\Delta \frac{\partial \Delta}{\partial b_n} d\Omega
\end{aligned}
\tag{2.81}
$$

where

$$\mathcal{SD}_1 = -u^I_{\langle t \rangle} + \phi_{\langle t^I \rangle \langle n \rangle} + \phi_{\langle n \rangle \langle t^I \rangle} \tag{2.82}$$

$$\mathcal{SD}_{2,i} = (\nu + \nu_t)\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) n_j - q n_i + \frac{\partial F_{S_{W_p,k}}}{\partial v_i} n_k + \acute{F}^v_{S_{W_p,i}} \tag{2.83}$$

$$\phi_{ij} = \frac{\partial F_{S_{W_p,k}}}{\partial \tau_{ij}} n_k \tag{2.84}$$

$T^{WF}_{SD}$, $\int_{S_{W_P}} \mathcal{A}^{WF}_{\Delta} \frac{\partial \Delta^P}{\partial b_n} dS$ and $\int_{S_W} \mathcal{A}^{WF}_{\Delta} \frac{\partial \Delta^P}{\partial b_n} dS$ summarize the contribution of the wall functions differentiation to the sensitivity derivatives [14]. As can be seen, all but the last term of eq. 2.81 are surface integrals, which can be computed at a cost that is, practically, negligible when compared to the solution of the primal or the adjoint equations. However, this is not the case for the last field integral which contains the distance variation for the entire domain w.r.t. the design variables. The simplest way to compute this variation is through finite differences, i.e. by perturbating each of the design variables by an infinitesimally small quantity $\epsilon$ in the positive and negative directions and re-computing nodal distances for the entire domain. Then, the total distance variation would be

$$\frac{\delta \Delta}{\delta b_n} = \frac{\Delta(b_n + \epsilon) - \Delta(b_n - \epsilon)}{2\epsilon} \tag{2.85}$$

Having computed the total distance variation, the partial variation of $\Delta$ appearing in the last field integral of 2.81 can be calculated through eq. 2.26 as follows

$$\frac{\partial \Delta}{\partial b_n} = \frac{\delta \Delta}{\delta b_n} - \frac{\partial \Delta}{\partial x_k} \frac{\delta x_k}{\delta b_n}$$

Nevertheless, the finite differences method has the same issues as the ones described in the introduction of this diploma thesis, namely the requirement to make $2N$ computations of the distance field (for instance by an exhaustive search of all cell centers with all boundary faces) and the sensitivity of the result from the value of $\epsilon$.

An alternative and more cost-effective way to deal with $\partial\Delta/\partial b_n$ is to apply the adjoint methodology in order to eliminate the term containing this variation. There are various PDEs that can be used to compute the distances field $\Delta$. Hamilton-Jacobi equation has shown to produce a very good approximation to the Euclidean distance field and to be numerically robust [8, 15]. Hamilton-Jacobi equation reads

$$R^{\Delta} = \frac{\partial \left(c_j \Delta\right)}{\partial x_j} - \Delta \frac{\partial^2 \Delta}{\partial x_j^2} - 1 = 0 \tag{2.86}$$

where $c_j = \partial\Delta/\partial x_j$. The boundary conditions of eq. 2.86 consist of a zero Dirichlet condition for the solid wall boundaries and $\frac{\partial\Delta}{\partial x_i} n_i = 1$ for the rest of the domain boundaries. This equation can be viewed as an additional primal PDE to be solved, meaning that it should be added to the augmented objective function, eq. 2.22 which now becomes

$$F_{aug} = \underbrace{F + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega + \int_\Omega \widetilde{\nu}_a R^{\widetilde{\nu}} d\Omega + \int_\Omega m_i^a R_i^m d\Omega}_{T_1} + \underbrace{\int_\Omega \Delta_\alpha R^\Delta d\Omega}_{T_2} \quad (2.87)$$

where $\Delta_\alpha$ is the adjoint to the distance field variable. The differentiation of $F_{aug}$ follows the same methodology presented in section 2.2.2. We have

$$\frac{\delta F_{aug}}{\delta b_n} = \frac{\delta T_1}{\delta b_n} + \frac{\delta T_2}{\delta b_n} \quad (2.88)$$

The development of $\delta T_1/\delta b_n$ led to eq. 2.81. $\delta T_2/\delta b_n$ is developed using the Leibniz theorem, as follows

$$\frac{\delta T_2}{\delta b_n} = \frac{\delta}{\delta b_n} \int_\Omega \Delta_a R^\Delta d\Omega = \int_\Omega \Delta_a \frac{\partial R^\Delta}{\partial b_n} d\Omega + \int_{S_{W_p}} \Delta_a R^\Delta n_k \frac{\delta x_k}{\delta b_n} dS \quad (2.89)$$

After differentiating the Hamilton-Jacobi equation and substituting the result in eq. 2.89, we receive

$$\frac{\delta T_2}{\delta b_n} = \int_S 2\Delta_a \frac{\partial\Delta}{\partial x_j} n_j \frac{\partial\Delta}{\partial b_n} dS + \int_{S_{W_p}} \Delta_a R^\Delta n_k \frac{\delta x_k}{\delta b_n} dS - \int_\Omega 2\frac{\partial}{\partial x_j}\left(\Delta_a \frac{\partial\Delta}{\partial x_j}\right)\frac{\partial\Delta}{\partial b_n} d\Omega \quad (2.90)$$

By integrating eq. 2.90 into 2.81, the expression where the multiplier of $\partial\Delta/\partial b_n$ in the resulting volume integrals should be set to zero, is derived. Thus, the adjoint to the distance field equation is derived

$$R^{\Delta_\alpha} = -2\frac{\partial}{\partial x_j}\left(\Delta_\alpha \frac{\partial\Delta}{\partial x_j}\right) + \widetilde{\nu}\widetilde{\nu}_a C_\Delta = 0 \quad (2.91)$$

where the first of the terms in the RHS of eq. 2.91 is contributed by the differentiation of the Hamilton-Jacobi equation 2.86, whereas the second one from the differentiation of the Spalart-Allmaras equation.

Having satisfied the field adjoint distance equation along with the proper boundary condition [9], the terms that should be added to the sensitivity derivatives expression

replacing the last field integral $\int_\Omega \widetilde{\nu}\widetilde{\nu_a}\mathcal{C}_\Delta \frac{\partial \Delta}{\partial b_n} d\Omega$ of equation 2.81 are

$$\int_\Omega \widetilde{\nu}\widetilde{\nu_a}\mathcal{C}_\Delta \frac{\partial \Delta}{\partial b_n} d\Omega = \int_{S_{W_p}} \Delta_a R^\Delta n_k \frac{\delta x_k}{\delta b_n} dS - \int_{S_{W_p}} 2\Delta_a \frac{\partial \Delta}{\partial x_j} n_j \frac{\partial \Delta}{\partial x_m} n_m n_k \frac{\delta x_k}{\delta b_n} dS \tag{2.92}$$

### 2.2.7 Final Expression of the Sensitivity Derivatives

Taking everything into consideration, the final expression for the sensitivity derivatives reads

$$\begin{aligned}
\frac{\delta F_{aug}}{\delta b_n} =& T_{SD}^{WF} - \int_{S_{W_p}} \mathcal{SD}_1 \frac{\partial \tau_{ij}}{\partial x_m} n_j t_i^I n_m n_k \frac{\delta x_k}{\delta b_n} dS - \int_{S_{W_p}} \mathcal{SD}_1 \tau_{ij} \frac{\delta (n_j t_i^I)}{\delta b_n} \frac{\delta x_k}{\delta b_n} dS \\
&+ \int_{S_{W_p}} \mathcal{SD}_{2,i} v_{\langle t \rangle}^I \frac{\delta t_i^I}{\delta b_n} dS - \int_{S_{W_p}} \mathcal{SD}_{2,i} \frac{\partial v_i}{\partial x_m} n_m n_k \frac{\delta x_k}{\delta b_n} dS \\
&- \int_{S_{W_p}} \left[ \left( \nu + \frac{\widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu_a}}{\partial x_j} n_j + \frac{\partial F_{S_z}}{\partial \widetilde{\nu}} n_z + \acute{F}_S^{\widetilde{\nu}} \right] \frac{\partial \widetilde{\nu}}{\partial x_m} n_m n_k \frac{\delta x_k}{\delta b_n} dS \\
&- \int_{S_{W_p}} \left( -u_{\langle n \rangle} + \phi_{\langle n \rangle \langle n \rangle} \right) \left( \tau_{ij} \frac{\delta (n_i n_j)}{\delta b_n} + \frac{\partial \tau_{ij}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k n_i n_j \right) dS \\
&- \int_{S_{W_p}} \phi_{\langle t^I \rangle \langle t^I \rangle} \left( \tau_{ij} \frac{\delta (t_i^I t_j^I)}{\delta b_n} + \frac{\partial \tau_{ij}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k t_i^I t_j^I \right) dS \\
&- \int_{S_{W_p}} \left( \phi_{\langle t^{II} \rangle \langle t^I \rangle} + \phi_{\langle t^I \rangle \langle t^{II} \rangle} \right) \left( \tau_{ij} \frac{\delta (t_i^{II} t_j^I)}{\delta b_n} + \frac{\partial \tau_{ij}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k t_i^{II} t_j^I \right) dS \\
&- \int_{S_{W_p}} \phi_{\langle t^{II} \rangle \langle t^{II} \rangle} \left( \tau_{ij} \frac{\delta (t_i^{II} t_j^{II})}{\delta b_n} + \frac{\partial \tau_{ij}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k t_i^{II} t_j^{II} \right) dS \\
&+ \int_{S_{W_p}} n_i \frac{\partial F_{S_{W_p,i}}}{\partial x_m} n_m \frac{\delta x_k}{\delta b_n} n_k dS + \int_{S_{W_p}} F_{S_{W_p,i}} \frac{\delta n_i}{\delta b_n} dS + \int_{S_{W_p}} F_{S_{W_p,i}} n_i \frac{\delta (dS)}{\delta b_n} \\
&- \int_{S_{W_P}} \frac{\partial m_i^a}{\partial x_j} n_j \frac{\delta x_i}{\delta b_n} dS + \int_{S_{W_p}} \mathcal{A}_\Delta^{WF} \frac{\partial \Delta^P}{\partial b_n} dS + \int_{S_W} \mathcal{A}_\Delta^{WF} \frac{\partial \Delta^P}{\partial b_n} dS \\
&+ \int_{S_{W_p}} \Delta_a R^\Delta n_k \frac{\delta x_k}{\delta b_n} dS - \int_{S_{W_p}} 2\Delta_a \frac{\partial \Delta}{\partial x_j} n_j \frac{\partial \Delta}{\partial x_m} n_m n_k \frac{\delta x_k}{\delta b_n} dS
\end{aligned} \tag{2.93}$$

## 2.3 Grid Displacement Strategy

The sensitivity derivatives of the objective function w.r.t. the design variables can be utilised to produce an improved geometry, i.e. a shape update of the design surface, which leads to an improved objective function value. This can be implemented by importing the sensitivities into an automated morphing tool. The tool needs to have the ability to morph the geometry according to the direction suggested by the sensitivities, from which point the grid can either be regenerated according to the newly suggested shape or be deformed without the need for remeshing the geometry. Since the former highly increases the computational cost of each shape deformation, the latter option is utilized.

In this diploma thesis, two different grid morphing techniques are employed. At the simulations conducted with the BMW shape optimization software, the Vertex Morphing method for node-based shape optimization is utilised, whereas at the simulations that are executed with the complete PCOpt/NTUA software, the computational grid is parameterized using volumetric B-splines.

In the following subsections the basic features of the two aforementioned morphing methods are presented.

### 2.3.1 Vertex Morphing

Vertex Morphing [16, 17] is a shape control method for node-based shape optimization, which assumes the location of each surface point, as a control parameter (design variable). In other words, unlike "morphing box" techniques, Vertex Morphing uses the design surface node positions as the design variables of the optimization problem. Since all available degrees of freedom are used as design variables, node-based methods lead to the richest design space possible for optimization of a discrete geometry, which means that they can lead to any non-intuitive shape.

A disadvantage of node-based methods is the possible shape update irregularities. This problem is prevented by applying a distance-dependent smoothing filter on the resulting sensitivities and displacements of each node, which weighs the sensitivities and displacements of the node with those of its neighbours.

### 2.3.2 Volumetric B-Splines

On the contrary to node-based shape optimization, a Free Form Deformation (FFD) method can be used instead to parameterize and move the mesh of the geometry in question. To this category belongs the parameterization based upon B–splines defined in $3D$ space, the so-called volumetric B-Splines. The method, as implemented by the PCOpt/NTUA [18] and coupled with the corresponding adjoint solver, uses a set of control points in 3D space, in the form of a structured control grid. Since the displacement of the B-Splines control points leads to a reallocation of the CFD mesh points residing inside the boundaries of the control grid, the Cartesian X,Y and Z coordinates of the B-Splines control points can be used as the design variables of the optimization problem.

The theoretical background of the volumetric B-Splines representation and morphing method is briefly discussed below.

Let $b_m^{ijk}, m \in [1,3], i \in [0,I], j \in [0,J], k \in [0,K]$ be the Cartesian coordinates of the $ijk-th$ control point of the 3D structured control grid, where $I$, $J$ and $K$ are the number of control points per control grid direction. The Cartesian coordinates $\mathbf{x} = [x_1, x_2, x_3]^T = [x, y, z]^T$ of a CFD mesh point residing within the boundaries defined by the control grid are related to the parametric ones with the following expression.

$$x_m(u,v,w) = U_{I,pu}(u) V_{J,pv}(v) W_{L,pw}(w) b_m^{ijk} \quad , \quad m = 1,2,3 \tag{2.94}$$

where $\mathbf{u} = [u_1, u_2, u_3]^T = [u, v, w]^T$ are the mesh point parametric coordinates, $U, V, W$ are the B-splines polynomial basis functions in each of the 3 directions and $pu, pv, pw$ their respective degrees.

In order to compute the mesh parametric coordinates, a mapping from $\Re^3(x,y,z) \rightarrow \Re^3(u,v,w)$ is required. Given the control points position, the knot vectors and the basis functions degrees, the parametric coordinates $(u,v,w)$ of a point with Cartesian coordinates $r = [x_r, y_r, z_r]^T$ can be computed by demanding the volumetric B-splines to be able to reproduce its actual Cartesian coordinates, i.e. by solving the system of equations

$$\mathbf{R}(u,v,w) = \begin{bmatrix} x(u,v,w) - x_r = 0 \\ y(u,v,w) - y_r = 0 \\ z(u,v,w) - z_r = 0 \end{bmatrix} \tag{2.95}$$

where $x_m(u,v,w), \quad m = 1,2,3$ are computed through eq. 2.94, based on the given $\mathbf{b}$ values. The $3 \times 3$ system of eq. 2.95 can be solved independently for each parameterized mesh point using for example the Newton-Raphson method, after computing

and inverting the Jacobian $\partial x_m/\partial u_j$ , $m, j \in [1,3]$. The Jacobian matrix is computed analytically through a closed form expression resulting by differentiating eq. 2.94 w.r.t. the components of $\mathbf{u}$. Since the evaluation of the parametric coordinates of each point is independent from the rest of the grid points, this phase may run in parallel.

The aforementioned mapping process has to be done only once at the beginning of the optimization and thereafter eq. 2.94 can be used to compute the CFD mesh coordinates according to the values of the parametric ones at a negligible computational cost. In addition, since $x_m$ depends only on $(u, v, w)$ and $\mathbf{b}$, the deformed meshes are step-independent. This means that, for a given final control points position, the same mesh quality will be obtained independent of the number of steps taken to reach that position.

## 2.4 Design Variables Update Methods

There are various methods to update the design variables according to the computed sensitivity derivatives. In this thesis two of these methods are utilised in the applications presented in chapter 3, *steepest descent* and *BFGS* method. Both methods are briefly described in this section, whereas more detailed analysis can be found in [19].

In both approaches the design variables $b_n, n \in [1, N]$ are updated according to the following expression.

$$b_n^{new} = b_n^{old} + \eta p_n \tag{2.96}$$

What distinguishes them from one another is the calculation of $p_n$. The $\eta$ value defines the *step length* of the descent and is computed through eq. 2.97.

$$\eta = \frac{\Delta b^{max}}{\Delta b^{act}} \tag{2.97}$$

where $\Delta b^{max}$ is the maximum allowed displacement of the control points $b_n$ as defined by the user and $\Delta b^{act}$ is the maximum displacement of the control points as calculated from the optimization for $\eta = 1$.

It must be noted that for the simulations that are ran exclusively with the PCOpt/NTUA software the $\eta$ value is computed once at the end of the first optimization cycle and kept constant for the rest of the optimization. On the contrary, for the simulations that are ran with the PCOpt/NTUA code coupled with the ShapeModule software from BMW, $\eta$ is reevaluated at the end of each optimization cycle to ensure that the displacement of the surface mesh nodes does not exceed the maximum value determined by the user.

### 2.4.1 Steepest Descent

Steepest descent is a simple gradient method with a straightforward implementation. Although it is often not the most efficient method, it is an absolutely essential tool to prototype optimization algorithms and for preliminary testing of models. Once the model formulation is stable, one might want to invest more in considering improved optimization methods, especially 2nd order ones such as the BFGS method.

The gradient $\frac{\delta J}{\delta b_n}$ at location $b_n^{old}$ points towards a direction where the function increases. Since the adjoint code, that has been developed by PCOpt/NTUA, incorporates optimization problems as minimization ones, the so-called *steepest descent direction* is $p_n = -\delta J/\delta b_n$. Eventually, equation 2.96 takes the form,

$$b_n^{new} = b_n^{old} - \eta \frac{\delta J}{\delta b_n} \tag{2.98}$$

### 2.4.2 BFGS

Newton methods utilise the second derivatives of the objective function w.r.t. design variables, also known as the Hessian matrix $\nabla^2_{b_n} J$, to define the search direction $p_n$.

$$p_n = -\left[\left(\frac{\delta^2 F}{\delta b_n \delta b_m}\right)^{-1} \frac{\delta F}{\delta b_m}\right]^{old} \tag{2.99}$$

However, the exact computation of the inverse of the hessian matrix can become challenging and this is what quasi-Newton methods try to address. The most popular quasi-Newton algorithm is the BFGS method [19], named after its discoverers Broyden - Fletcher - Golfsrb - Shanno. It belongs to the second-derivative line search family method and is considered to be one of the most powerful methods to solve unconstrained optimization problems. BFGS method poses superlinear convergence but tends to get more affected by inexact sensitivities in comparison to simple gradient methods. Like other quasi-Newton methods, BFGS approximates the Hessian matrix $\nabla^2_{b_n} F$ with a $B_{nm}$ matrix, based on the gradient only. In this case, the search direction becomes

$$p_n = -\left[B_{nm}^{-1} \frac{\delta F}{\delta b_m}\right]^{old} \tag{2.100}$$

Eventually, equation 2.96 becomes

$$b_n^{new} = b_n^{old} - \eta \left[B_{nm}^{-1} \frac{\delta F}{\delta b_m}\right]^{old} \tag{2.101}$$

Approximation matrix $B_k$ (the nm index is omitted for simplicity) can be updated in each iteration $k$ with equation 2.102 which takes the Sherman–Morrison–Woodbury formula into consideration.

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k} \tag{2.102}$$

where $s_k = x_{k+1} - x_k$ and $y_k = \nabla F_{k+1} - \nabla F_k$.

The implementation of eq. 2.102 is computationally expensive, because it requires the system $B_k p_k = \nabla F_k$ to be solved for the step $p_k$. A solution to this problem comes through eq. 2.103.

$$B_{k+1}^{-1} = \left(I - \rho_k s_k y_k^T\right) B_k^{-1} \left(I - \rho_k y_k^T s_k\right) + \rho_k s_k s_k^T \tag{2.103}$$

where $\rho_k = \frac{1}{s_k y_k^T}$.

The appropriate initial approximation $B_0$ depends on the case. The most simple solution is to set it to be the identity matrix, or a multiple of the identity matrix, where the multiple is chosen to reflect the scaling of the variables. In the case that $B_k = I$, eq. 2.101 is simplified to eq. 2.98.

The $\eta$ value can be computed through line search methods or using the procedure that includes the maximum allowed displacement as previously described.

# Chapter 3

# Applications

In the first part of this chapter the integration of PCOpt/NTUA solvers into the BMW optimization software 'ShapeModule' is presented. Thereafter various simulations concerning the shape optimization of two different HVAC ducts are presented.

## 3.1 Integration of the PCOpt Solvers into ShapeModule

ShapeModule is the BMW shape optimization framework based on Vertex Morphing Method, 2.3.1, into which the PCOpt/NTUA primal and adjoint solvers are integrated. ShapeModule provides an interface that allows the user to connect to various tools or solvers like the one developed by PCOpt/NTUA. These solvers are responsible for computing the primal and adjoint solution and the sensitivity derivatives of the objective function w.r.t. the surface nodes coordinates of the geometry being optimized. The necessary geometry, i.e. the controlled wall boundaries, and the SDs are, then, communicated to ShapeModule, where the necessary geometry updates for the optimization are computed. Subsequently, the solver receives the geometry update, according to which it deforms the volume mesh of the domain and proceeds to the next optimization cycle.

ShapeModule is functionally divided into three different sections:

• **Core**: It includes the database and Vertex Morphing implementation and is developed in C++ programming language.

• **Algorithmic**: It includes the optimization algorithms, such as steepest descent, and is developed in Python programming language.

• **Interface**: It is responsible for receiving the geometry and the sensitivities and returning the shape update to the solver to which ShapeModule is coupled. This part is developed both with C++ and Python languages.

The shape optimization simulation is configured by setting up a .json (JavaScript Object Notation) file to control the shape optimization technique and, also, by providing the relevant primal and adjoint solvers with the proper configuration settings. More specifically, the most important settings in the .json file are the optimization algorithm to be used, the list of the objectives with their corresponding weights, the constraints, the maximum allowed displacement per optimization cycle (step size), the file path from where shapeModule will read the geometry and the sensitivities and write the displacements file, the filtering type and radius used on the sensitivities and displacements and the distance from the edges of the design surface to which the vertex morphing methodology will not be implemented (damping distance).

As part of this thesis the communication software between the flow and adjoint solver of PCOpt/NTUA and ShapeModule was reimplemented, basing its functionality on reading and writing of the necessary data files (File-IO coupling) rather than on the previously used MPI communication protocol. Furthermore, the communication software was expanded to support the separate use of the SDs computed by each adjoint solver defined by the user, making it capable of communicating multiple sensitivity files from the PCOpt/NTUA software into ShapeModule. As a result, some of the objective functions may be used as constraints, which would not be feasible if a single sensitivity vector list, i.e. a single sensitivity file, was computed. An additional benefit of the new implementation is that the weighting between the objective functions may be done now, if necessary, inside shapeModule according to the settings defined in .json file.

The simplified optimization workflow is shown in figure 3.2, whereas the simplified coupling diagram is presented in figure 3.1.

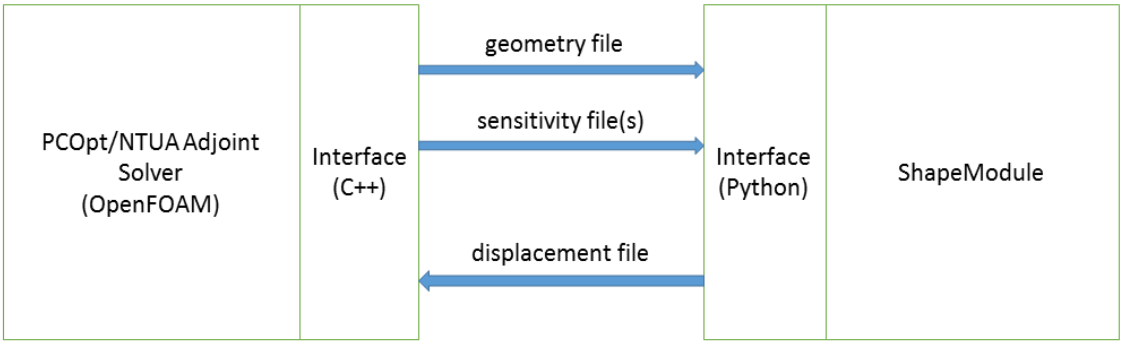



**Figure 3.1:** *The coupling of PCOpt/NTUA software with ShapeModule. Each software has an interface, through which the communication between them is achieved.*

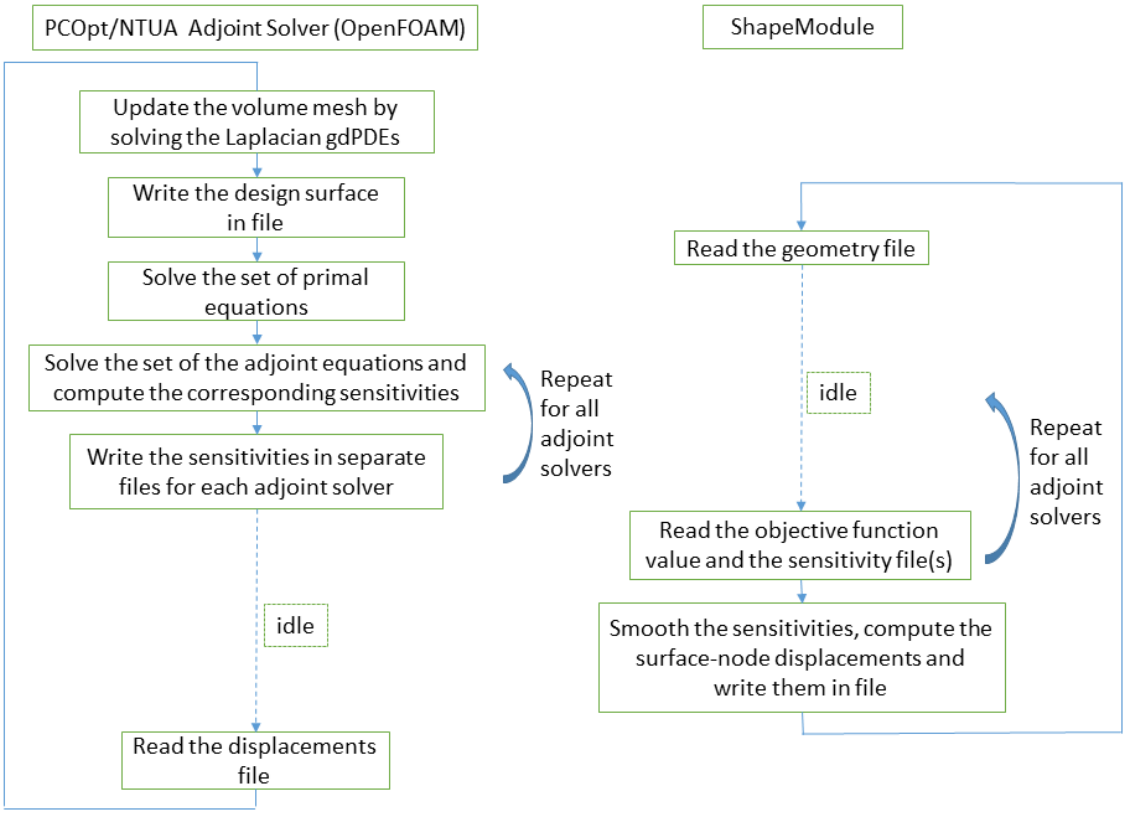



**Figure 3.2:** *The simplified diagram of the optimization procedure implemented in BMW.*

The communication files has been chosen to have a .csv format with the following structure.

• **geometry file**:

# nodeID , Cartesian coordinates
nodeID1 , x-coordinate , y-coordinate , z-coordinate
nodeID2 , x-coordinate , y-coordinate , z-coordinate
...
nodeIDN , x-coordinate , y-coordinate , z-coordinate
# faces , nodeIDs belonging to the face
faceID1 , nodeID1 , nodeID2 , ...
faceID2 , nodeID1 , nodeID2 , ...
...
faceIDM , nodeID1 , nodeID2 , ...
# EOF

• **sensitivity file**:

# nodeID , sensitivity vector
nodeID1 , x-component , y-component , z-component
nodeID2 , x-component , y-component , z-component
...
nodeIDN , x-component , y-component , z-component
# EOF

• **displacements file**:

# nodeID , displacements in each direction
nodeID1 , x-displacement , y-displacement , z-displacement
nodeID2 , x-displacement , y-displacement , z-displacement
...
nodeIDN , x-displacement , y-displacement , z-displacement
# EOF

where $N$ is the total number of nodes and $M$ the total number of faces of the design surface.

## 3.2 Simulations conducted with the coupled software of PCOpt/NTUA and ShapeModule

The coupling of the PCOpt/NTUA software with ShapeModule was first tested on an S-Shaped 3D Duct with a circular cross-section, due to its low computational

cost. The duct was optimized by using the pressure drop and the 'noise' objectives separately as well as combined in a multi-objective optimization. Since the aforementioned case was a demo one to make sure that the two softwares cooperate as expected, the results are not presented in this thesis.

Thereafter, the coupled software was used to design an HVAC duct, i.e. a defrosting duct. The initial geometry was a 3D box with one inlet and three outlets, as can be seen in fig. 3.3 The optimization workflow that was adopted is the following:

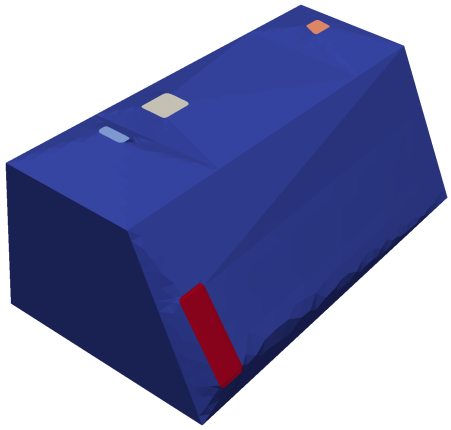


**Figure 3.3:** *Initial Box at the beginning of the optimization. Wall boundaries (blue), inlet (red) and the three outlets at the top of the box.*

- **Step 1**: Topology Optimization for total pressure losses minimization.
- **Step 2**: Smoothing of the geometry produced in Step 1.
- **Step 3**: Shape Optimization for total pressure losses and noise minimization.

The duct that was generated by applying topology optimization with the minimization of the total pressure losses as objective function, can be seen in fig. 3.4. The mesh is comprised of 81000 cells and the Reynolds number is $Re = 5500$, based on the hydraulic diameter of the rectangular inlet patch.

It is apparent from fig. 3.4 that the surface of the duct resulting from the topology optimization needs to be preprocessed before it can be inserted in the shape optimization software, in order to fill the appearing surface gaps and smooth the surface, especially near the inlet and outlets of the duct. This is achieved by using at first the Inspire CAD software by Altair to smooth the main part of the duct excluding the surfaces near the inlet and the outlets, and then by using the ANSA software by BETA CAE Systems to make sure that the inlet and outlet surfaces remain the same as those appearing in the initial box, fig. 3.3, close any gaps still appearing

in the geometry and make the final surface smoothing. The result of this procedure can be seen in fig. 3.5.

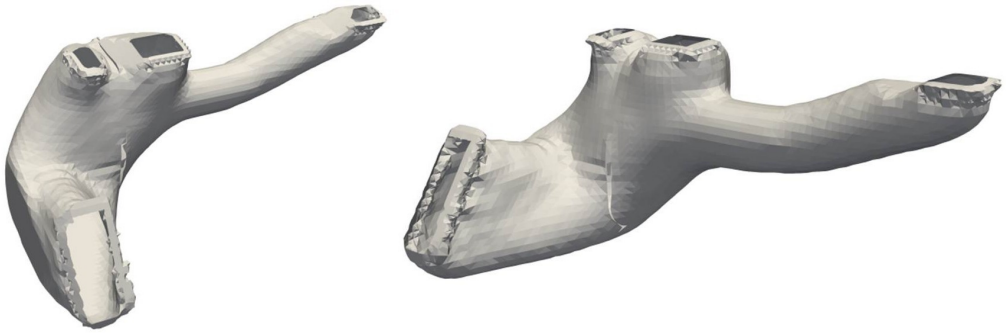


**Figure 3.4:** *The duct that was generated through topology optimization based on the box of fig. 3.3 from different angles of sight.*

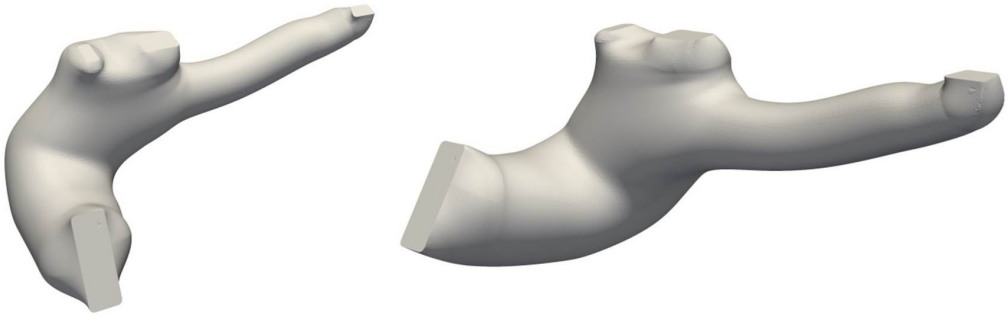


**Figure 3.5:** *The duct after the smoothing the surface geometry topology optimization produced, fig. 3.4.*

The duct of fig. 3.5 was inserted in the OpenFOAM environment, where the volume grid was generated using snappyHexMesh utility. The grid is comprised of 600.000 cells and $y^+$ ranges from 0.05 to 15 along the duct walls. Reynolds number remains the same as that chosen for the topology optimization, $Re = 5500$. Based on the smoothed geometry, fig. 3.5, the following simulations were conducted.

### 3.2.1 Minimization of 'Noise' in the HVAC Duct

The first shape optimization simulation, that was conducted by using the coupled PCOpt/NTUA and ShapeModule software, had as target the minimization of the noise produced at the outlets of the duct by minimizing the surrogate "noise" objective function presented in subsection 2.2.3. As initial geometry was taken the duct that was smoothed after the topology optimization, fig. 3.5. The design surface is

the whole solid surface of the duct, excluding apparently the inlet and the outlets of the domain. The volume where the objective is defined has been chosen to be near the outlets of the duct, since these are the areas that are most directly related to the generated noise. This volume consists of all duct cells that reside inside the boxes of fig. 3.6.

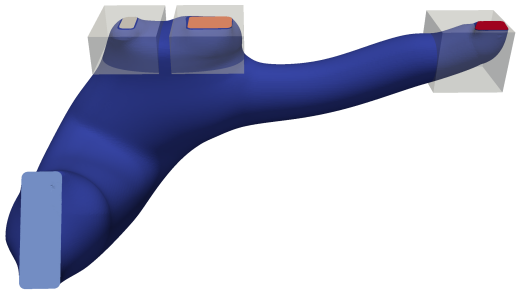


**Figure 3.6:** *The duct geometry before shape optimization and the semi-transparent boxes inside of which the objective function is defined.*

The optimization ran for 35 cycles after which the simulation stopped due to mesh worsening. The convergence of the objective function can be seen in fig. 3.7, whereas the comparison between the initial and optimized shape is presented in fig. 3.8. As a more characteristic magnitude, the average value of $\nu_t$ at the three outlets of the duct has been computed. The results appear on table 3.1.

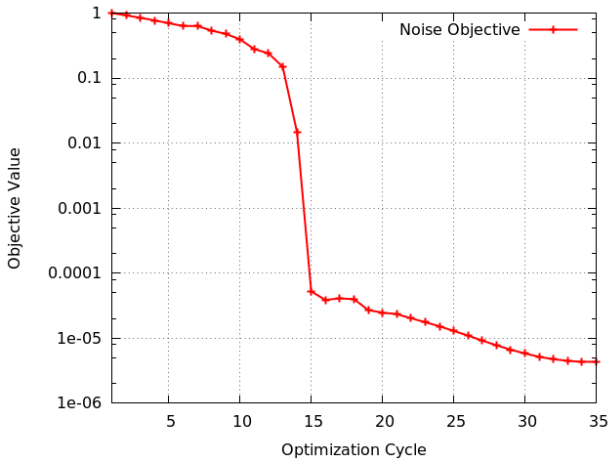



**Figure 3.7:** *Convergence rate of the "noise" objective function. All values are dimensionless, i.e. divided by the objective function value at the beginning of the optimization. The objective function reduced over 5 orders of magnitude over the 35 optimization cycles, i.e. over* 99.99%.

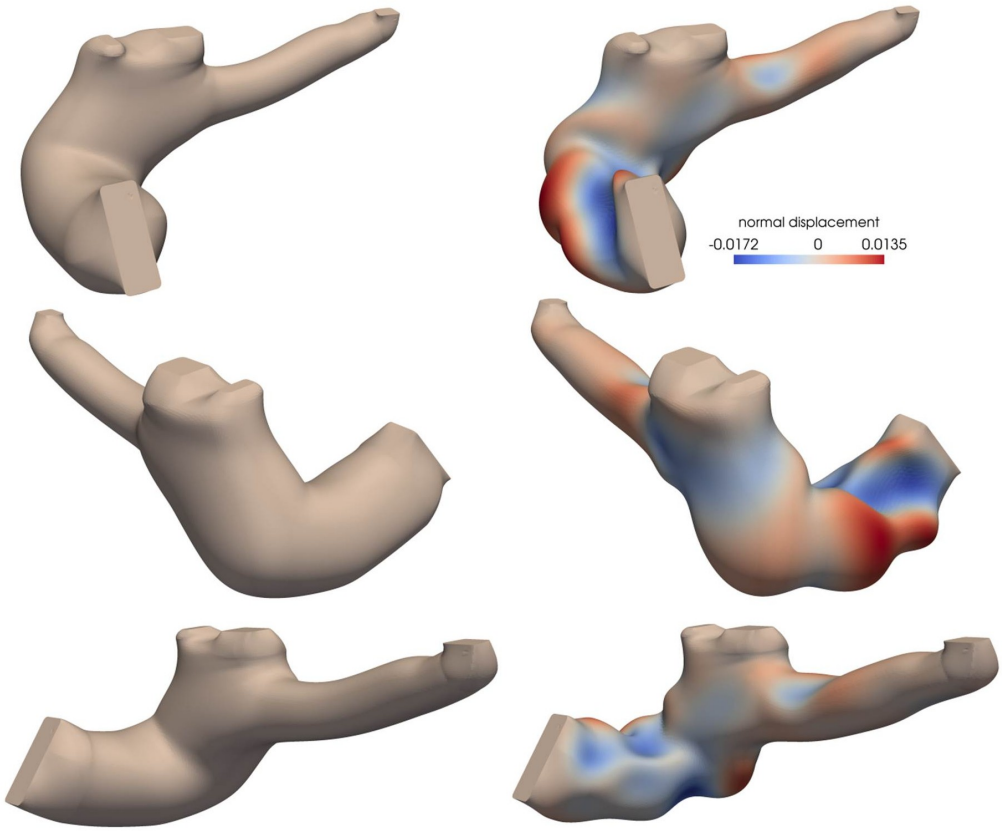



**Figure 3.8:** *Defrosting duct, $Re = 5500$, minimization of 'noise'. Initial (top) and optimized (bottom) geometry from different angles of sight. The signed cumulative normal displacement field indicates, also, the direction in which the duct surface points moved, either outwards (red) or inwards (blue).*

It must be noted that at the end of the $35^{th}$ optimization cycle, the total pressure losses objective is also reduced by 6.4% compared to its value at the beginning of the simulation, although it was not set as target of the optimization.

| | $InitialGeometry$ | $FinalGeometry$ |
|---|---|---|
| $\bar{\nu_t}$ *at the outlets* | $3.28\times10^{-5}$ | $1.47\times10^{-7}$ |

**Table 3.1:** *Average value of $\nu_t$ at the three outlets of the domain before and after the 'noise' optimization. $\bar{\nu_t}$ has reduced 220 times compared to its value at the beginning of the optimization.*

### 3.2.2 Minimization of Total Pressure Losses in the HVAC Duct

As initial geometry, was considered the duct that was smoothed after the topology optimization, fig. 3.5, similarly to the case of 'noise' minimization. The design surface is again the whole solid surface of the duct, excluding the inlet and the outlets of the domain.

The optimization ran for 13 cycles after which the simulation stopped due to mesh worsening. The convergence of the objective function can be seen in fig. 3.9, whereas the comparison between the initial and optimized shape is presented in fig. 3.10.

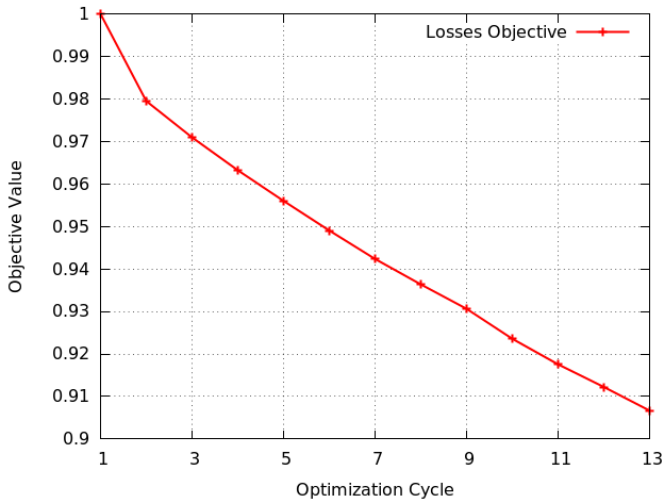



**Figure 3.9:** *Convergence rate of the total pressure losses objective function. All values are dimensionless, i.e. divided by the objective function value at the beginning of the optimization. The objective function has reduced by* 9.5% *over the 13 optimization cycles.*

It must be noted that at the end of the $13^{th}$ optimization cycle, the noise related objective function is also reduced by 80% compared to its value at the beginning of the simulation, although it was not set as the optimization target.

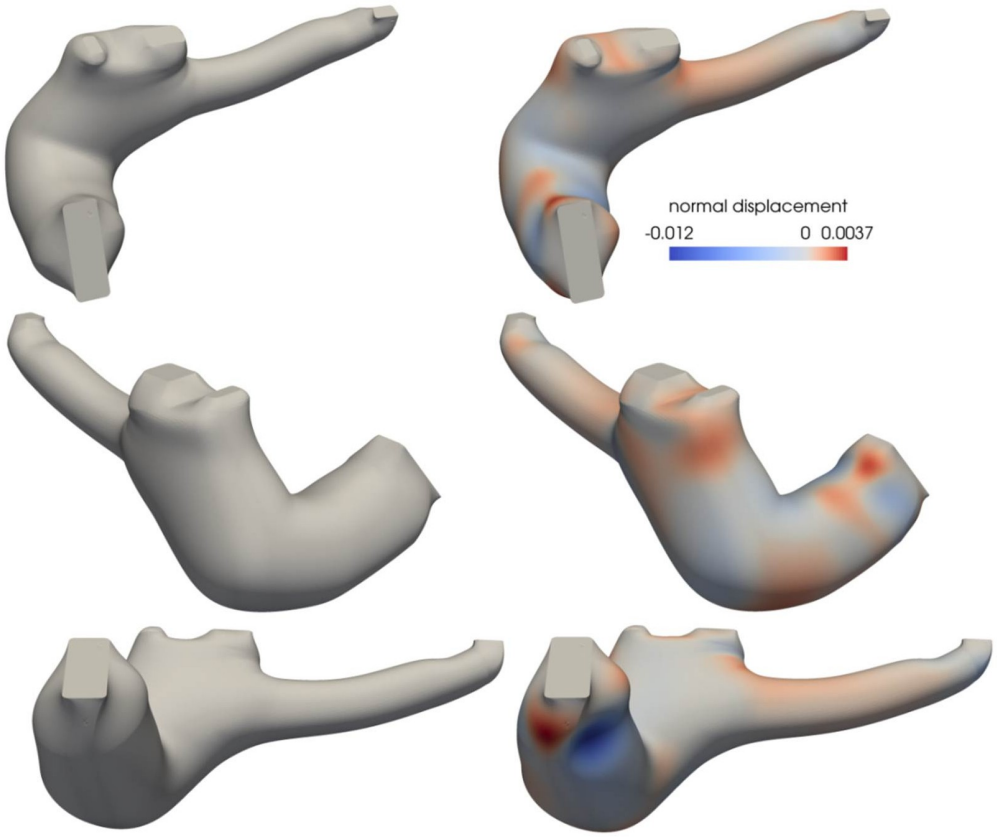



**Figure 3.10:** *Defrosting duct, $Re = 5500$, minimization of total pressure losses. Initial (top) and optimized (bottom) geometry from different angles of sight. The signed cumulative normal displacement field indicates, also, the direction in which the duct surface points moved, either outwards (red) or inwards (blue).*

The results of the two previous shape optimizations are summed up in the following table.

| | $F_{v_t}$ | $F_{P_t}$ |
|---|---|---|
| *'**Noise**' minimization* | $\mathbf{-99.99\%}$ | $-6.4\%$ |
| $\mathbf{\Delta P_t}$ *minimization* | $-80\%$ | $\mathbf{-9.5\%}$ |

**Table 3.2:** *Optimization results using the noise and total pressure losses objective functions. In bold are the objective function values reduction that was achieved by setting the corresponding objective functions as the optimization target.*

## 3.3 Simulations conducted with the complete optimization software of PCOpt/NTUA

The geometry that was optimized by using the complete optimization software of PCOpt/NTUA is that of an HVAC duct. In this case, the grid was parameterized by using volumetric B-Splines with 300 active control points (in two separate control boxes with 150 active CPs each). The grid is comprised of $2{\times}10^6$ cells and Reynolds number is $Re = 14600$, based on the hydraulic diameter of the inlet patch. The initial geometry and the two control boxes can be seen in fig. 3.11.

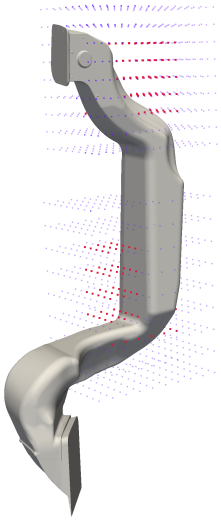


**Figure 3.11:** *The HVAC duct that is optimized exclusively with the PCOpt/NTUA software together with the two control boxes used to parameterize the CFD mesh residing inside them. The inlet is the lower end of the duct, whereas the outlet the upper end. Active control points (red) and non-active in (purple).*

### 3.3.1 Minimization of the Exit Velocity Profile Non-Uniformity

This optimization aims at minimizing the non-uniformity of the velocity profile at the outlet of the HVAC duct or, in other words, the standard deviation of each velocity component over the outlet patch, by using the corresponding objective function presented in subsection 2.2.3. The design variables were updated using steepest descent and BFGS. The convergence of the objective function is shown in fig. 3.12 and, as expected, BFGS converges faster. The total reduction in the objective function is about 22% and with steepest descent it is achieved in 19 optimization cycles, whereas with BFGS in just 5.

The comparison between the initial and optimized shape can be seen in fig. 3.14.

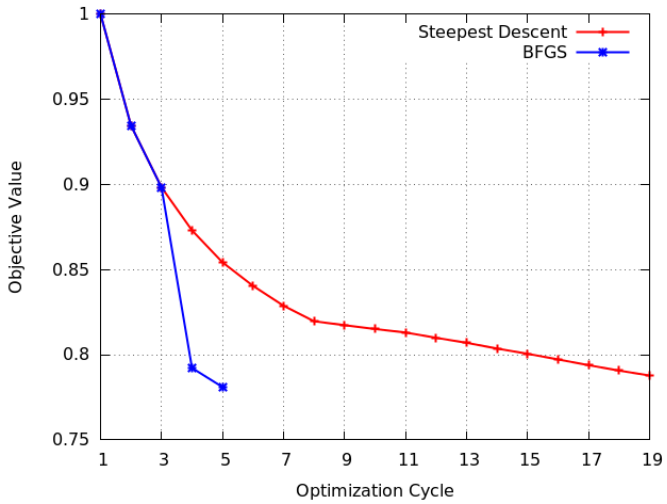



**Figure 3.12:** *HVAC duct, convergence rate of uniformity objective function. All values are divided by the objective function value at the beginning of the optimization. The objective function reduced by* 22% *over 5 optimization cycles with the BFGS method.*

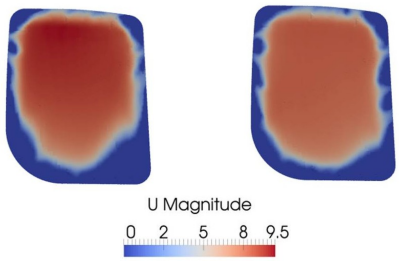



**Figure 3.13:** *Velocity magnitude at the outlet of the HVAC duct. The initial duct (left) and the optimized one (right).*

| *duct outlet* | *maximum velocity* | *average velocity* | $1-\gamma$ |
|---|---|---|---|
| *initial duct* | $9.42m/s$ | $4.98m/s$ | 0.2987 |
| *optimized duct* | $8.14m/s$ | $4.98m/s$ | 0.2729 |

**Table 3.3:** *Maximum velocity, average velocity and non-uniformity index* $1-\gamma$ *values compared between the initial and the optimized duct. Average velocity remains the same due to the continuity equation.*

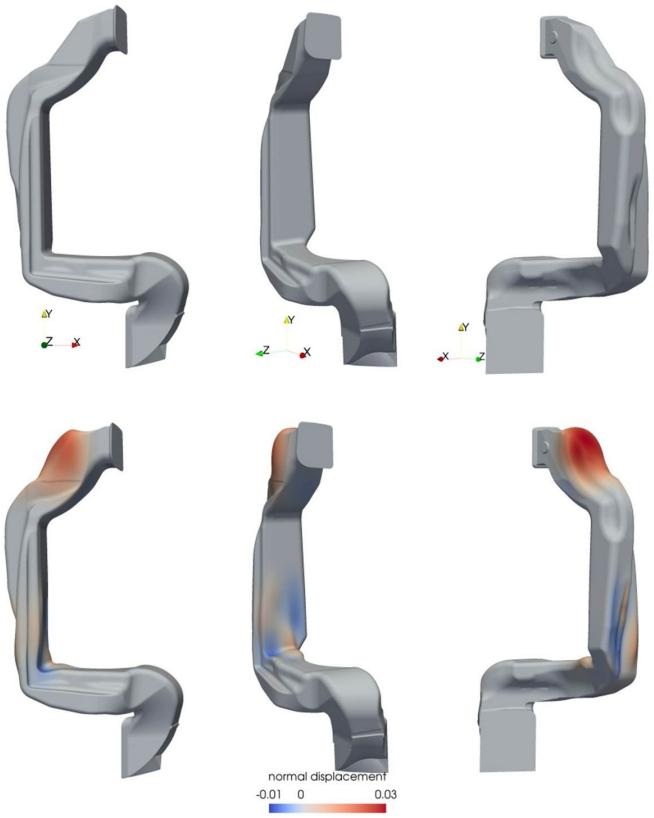



**Figure 3.14:** *HVAC duct, $Re = 14600$, minimization of non-uniformity at the outlet of the duct (upper end of the duct). Initial (top) and optimized (bottom) geometry from different angles of sight. The signed cumulative normal displacement field indicates, also, the direction in which the duct surface points moved, either outwards (red) or inwards (blue).*

### 3.3.2 Minimization of Total Pressure Losses and the Exit Velocity Profile Non-Uniformity

The same HVAC duct that was presented at the start of this section, fig. 3.11, was optimized by setting the minimization of both the outlet non-uniformity and the total pressure losses as target of the optimization. These goals are conflicting, as it can be seen in fig. 3.15, where the only target of the optimization was the $\Delta P_t$ minimization. Although $\Delta P_t$ drops by 23%, outlet non-uniformity has increased by 3.5%.

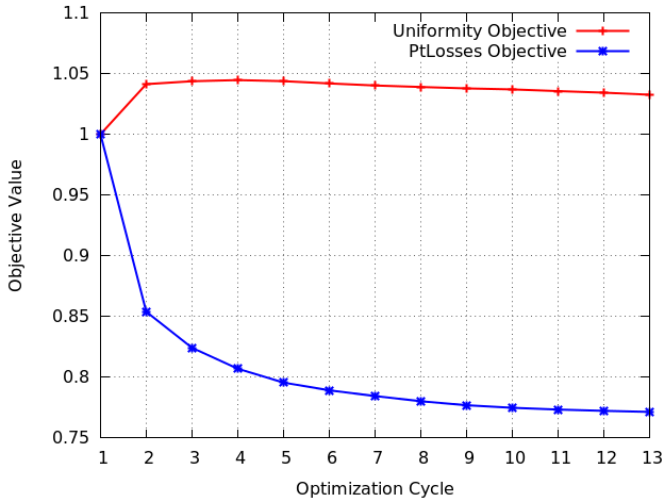



**Figure 3.15:** *HVAC duct, objective function values evolution, when the total pressure losses minimization is the target of the optimization.* $\Delta P_t$ *reduced by* 23% *after 13 optimization cycles, but the uniformity objective has increased by* 3.5%.

The case was approached with two methods:

- **Minimization using weights**

Each objective function was multiplied by a weight, with the uniformity objective to have a 250:1 advantage over $\Delta P_t$ objective function (ratio of the corresponding weights), due to their different units of measurement.

- **Minimization using equality constraint**

The minimization of $\Delta P_t$ objective was the target of the optimization, subject to the constraint of 10% reduction in the uniformity objective compared to its value at the beginning of the optimization. The SQP (Sequential Quadratic Programming) method for equality constraints was utilised.

The objective functions convergence can be seen in figures 3.16 and 3.17 respectively, whereas a comparison between the optimized geometries produced by each approach can be seen in fig. 3.18. The results from these simulations are summed up in table 3.4.

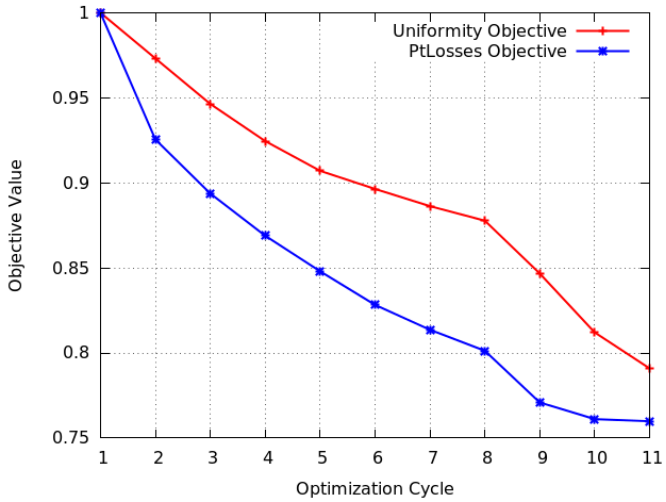



**Figure 3.16:** *HVAC duct, multi-objective with weighting, convergence rate of the uniformity and $\Delta P_t$ objective functions. All values are divided by the objective function value at the beginning of the optimization.*

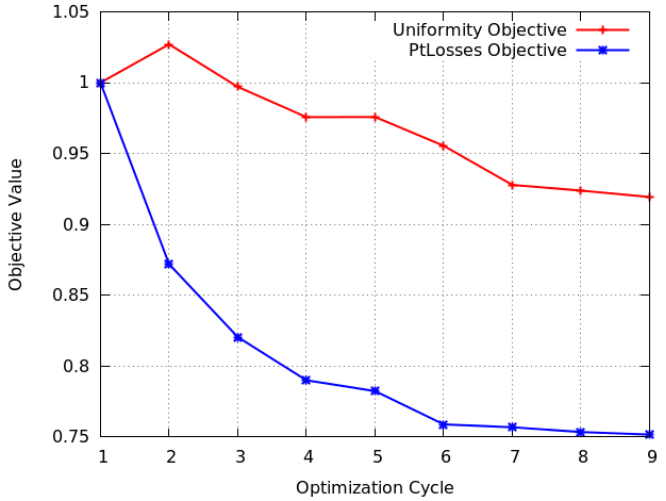



**Figure 3.17:** *HVAC duct, single-objective $\Delta P_t$ subject to the constraint of* 10% *reduction in the uniformity objective, convergence rate of the uniformity and $\Delta P_t$ objective functions. All values are divided by the objective function value at the beginning of the optimization.*

| | $F_\gamma$ | $F_{P_t}$ |
|---|---|---|
| *Non-uniformity minimization* | ***-22 %*** | *+5.5 %* |
| *Non-uniformity and $\Delta P_t$ minimization* ***using weights*** | ***-20 %*** | ***-24 %*** |
| *Non-uniformity and $\Delta P_t$ minimization* ***with constraint*** | ***-8 %*** | ***-25 %*** |

**Table 3.4:** *HVAC duct, $Re = 14600$, optimization results comparison.*

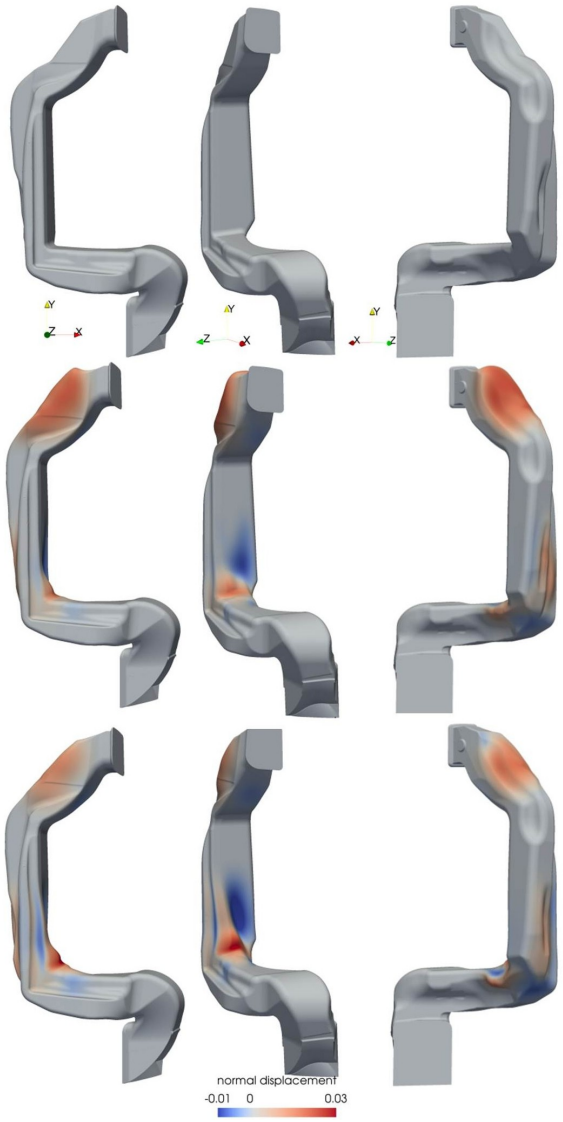



**Figure 3.18:** *HVAC duct, $Re = 14600$, optimization using the uniformity and the total pressure losses objectives. Initial geometry (top), optimized using weights (middle) and optimized using equality constraint for the uniformity objective (bottom), from different angles of sight. The signed cumulative normal displacement field indicates, also, the direction in which the duct surface points moved, either outwards (red) or inwards (blue).*

# Chapter 4

# Summary - Conclusions

In this diploma thesis, the continuous adjoint solver developed by PCOpt/NTUA in the OpenFOAM environment was integrated in the optimization workflow of BMW, the so-called 'ShapeModule'. The coupling between the two softwares is achieved with the File-IO communication method, replacing the previously used MPI communication protocol. Furthermore, the communication software was extended to support the separate use of the sensitivity derivatives computed by each adjoint solver defined by the user.

The adjoint method, that is developed by PCOpt/NTUA and used in this thesis, is formulated based on the Enhanced SI (E-SI) approach, which has a low computational cost as the standard SI method, but without compromising the accuracy of the computed sensitivities. The Spalart-Allmaras turbulence model was fully differentiated in order to take into account the effect of the design variable value changes in the flow turbulence, leading to more accurate SDs compared to the 'frozen turbulence' assumption.

After the integration was completed, the coupled software was used to design a defrosting duct of the cabin of a passenger car. Since the initial geometry was a box with specific inlet and outlets, topology optimization and surface smoothing were implemented as a preliminary stage for the shape optimization which followed. Targets of the optimization were the minimization of the total pressure losses between inlet and outlets and the minimization of the noise produced near the outlets of the duct. The later was achieved by using a surrogate, noise related, objective function.

Furthermore, the continuous adjoint method for an objective function that describes the flow uniformity at the outlet of the domain to be optimized was developed and programmed. Based on this formulation, an HVAC duct of a passenger vehicle was optimized with the uniformity objective as single target, but also in conjunc-

tion with the total pressure losses objective function in multi-objective optimization problems. In the second case, two approaches were examined, the weighting between the objectives and the optimization for $\Delta P_t$, subject to an equality constraint for the uniformity objective.

Concluding this thesis, the PCOpt/NTUA software has been successfully integrated in the BMW optimization workflow 'ShapeModule' and utilised to design a defrosting duct. A new objective function related to the flow uniformity was implemented in the PCOpt/NTUA code environment and utilised for the optimization of an HVAC duct.

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Μηχανολόγων Μηχανικών
Τομέας Ρευστών
Μονάδα Παράλληλης Υπολογιστικής Ρευστοδυναμικής
& Βελτιστοποίησης

# Εφαρμογή της Συνεχούς Συζυγούς Μεθόδου στη Βελτιστοποίηση Μορφής Αεραγωγών Κλιματισμού Επιβατικού Αυτοκινήτου

Εκτενής Περίληψη Διπλωματικής Εργασίας

**Ανδρέας Στέφανος Μαργέτης**

Επιβλέπων
Κυριάκος Χ. Γιαννάκογλου, Καθηγητής ΕΜΠ

Αθήνα, Ιούλιος 2019

### Διατύπωση του Συζυγούς Προβλήματος

Αφετηρία για τη διαμόρφωση του συζυγούς προβλήματος είναι η μαθηματική διατύπωση του πρωτεύοντος προβλήματος, το οποίο περιγράφεται από τις RANS εξισώσεις για ασυμπίεστη και χρονικά μόνιμη ροή συνδυασμένες με το μοντέλο τύρβης Spalart-Allmaras, [5]. Οι εξισώσεις αυτές, χρησιμοποιώντας τη σύμβαση του Einstein για τους επαναλαμβανόμενους δείκτες, γράφονται ως εξής

$$R^p = -\frac{\partial v_j}{\partial x_j} = 0 \tag{4.1α′}$$

$$R_i^v = v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial}{\partial x_j}\left[(\nu + \nu_t)\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] + \frac{\partial p}{\partial x_i} = 0\ , \qquad i = 1,2,3 \tag{4.1β′}$$

$$R^{\widetilde{\nu}} = v_j \frac{\partial \widetilde{\nu}}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu}}{\partial x_j}\right] - \frac{c_{b2}}{\sigma}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right)^2 - \widetilde{\nu}\, P\left(\widetilde{\nu}\right) + \widetilde{\nu}\, D\left(\widetilde{\nu}\right) = 0 \tag{4.1γ′}$$

όπου $v_i$ είναι η ταχύτητα του ρευστού, $\nu$ είναι η κινηματική συνεκτικότητα του ρευστού, $\nu_t$ η τυρβώδης κινηματική συνεκτικότητα και $p$ η στατική πίεση διαιρεμένη με την πυκνότητα του ρευστού. Η ανεξάρτητη μεταβλητή για την οποία επιλύεται το μοντέλο τύρβης είναι η $\widetilde{\nu}$ και η τυρβώδης συνεκτικότητα υπολογίζεται ως $\nu_t = \widetilde{\nu} f_{v_1}$. $P\left(\widetilde{\nu}\right)$ και $D\left(\widetilde{\nu}\right)$ είναι οι όροι παραγωγής και καταστροφής (βλ. κεφάλαιο 2).

Η συνεχής συζυγής μέθοδος μπορεί να αναπτυχθεί ακολουθώντας τρεις μεθοδολογίες, οι οποίες καταλήγουν στο ίδιο σύστημα συζυγών εξισώσεων και συζυγών οριακών συνθηκών, διαφορετικές όμως εκφράσεις παραγώγων ευαισθησίας, [10]. Η SI διατύπωση οδηγεί σε μια έκφραση για τις SDs η οποία περιλαμβάνει μόνο επιφανειακά ολοκληρώματα, έχει μικρό υπολογιστικό κόστος αλλά μπορεί κατά περίπτωση να υστερεί σε ακρίβεια. Η FI διατύπωση οδηγεί σε μια έκφραση για τις SDs η οποία περιλαμβάνει τόσο επιφανειακά όσο και χωρικά ολοκληρώματα, χαρακτηρίζεται από υψηλή ακρίβεια, αλλά και από υψηλό κόστος λόγω της ανάγκης υπολογισμού των παραγώγων ευαισθησίας του πλέγματος $\delta x_k / \delta b_n$ στον όγκο $\Omega$ του υπολογιστικού χωρίου. Η προσέγγιση Enhanced SI (E-SI), που πρόσφατα αναπτύχθηκε από τη ΜΠΥΡ&Β/ΕΜΠ, εξαλείφει την ανάγκη υπολογισμού του $\delta x_k / \delta b_n$ στο $\Omega$, οδηγώντας σε κόστος αντίστοιχο της SI και ακρίβεια αντίστοιχης της FI προσέγγισης.

Κατά την ανάπτυξη της συνεχούς συζυγούς μεθόδου είναι συνηθισμένη στη βιβλιογραφία η παραδοχή της ΄παγωμένης τύρβης΄, της θεώρησης δηλαδή ότι οι αλλαγές στη μορφή της υπό βελτιστοποίηση γεωμετρίας επηρεάζουν μόνο τις μέσες ροϊκές ποσότητες. Η παραδοχή αυτή οδηγεί σε αρκετές περιπτώσεις σε εσφαλμένο υπολογισμό των παραγώγων ευαισθησίας, οι οποίες μπορεί να έχουν ακόμη και λανθασμένο πρόσημο, [8, 9].

Τα δυό βασικά χαρακτηριστικά της συνεχούς συζυγούς μεθόδου που χρησιμοποιήθηκε στην παρούσα εργασία, όπως έχει αναπτυχθεί από τη ΜΠΥΡ&Β/ΕΜΠ, είναι ότι η μέθοδος βασίζεται στην E-SI διατύπωση και ότι γίνεται πλήρης διαφόριση του μοντέλου

τύρβης, εξασφαλίζοντας τον υπολογισμό των παραγώγων ευαισθησίας με αυξημένη ακρίβεια.

Η παρουσίαση της συζυγούς μεθόδου ακολουθεί μια γενική μεθοδολογία βασισμένη σε μια αντικειμενική συνάρτηση $F$ η οποία περιέχει τόσο χωρικά όσο και επιφανειακά ολοκληρώματα, ως εξής

$$F = \int_S F_{S_i} n_i dS + \int_\Omega F_\Omega d\Omega \tag{4.2}$$

όπου οι όροι $F_{S_i}$ και $F_\Omega$ είναι οι ολοκληρωματικές ποσότητες στο σύνορο και στο εσωτερικό του υπολογιστικού χωρίου, αντίστοιχα. Προκειμένου να εξασφαλιστεί η ανεξαρτησία του κόστους υπολογισμού των παραγώγων ευαισθησίας από το πλήθος N των μεταβλητών σχεδιασμού ορίζεται η επαυξημένη αντικειμενική συνάρτηση, όπως φαίνεται στη σχέση που ακολουθεί.

$$F_{aug} = F + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega + \int_\Omega \widetilde{\nu}_a R^{\widetilde{\nu}} d\Omega + \int_\Omega m_i^a R_i^m d\Omega \tag{4.3}$$

όπου $\Omega$ είναι το υπολογιστικό χωρίο, $u_i$ η συζυγής ταχύτητα, $q$ η συζυγής πίεση και $\widetilde{\nu}_a$ η συζυγής μεταβλητή του μοντέλου τύρβης και $m_i^a$ η συζυγής της μεταβλητής $m_i$. $m_i$ είναι οι Καρτεσιανές μετατοπίσεις των κόμβων του πλέγματος

Το τρίτο κατά σειρά ολοκλήρωμα της σχέσης 4.3 αμελείται στην περίπτωση που υιοθετείται η παραδοχή της ΄παγωμένης τύρβης΄. Το τέταρτο ολοκλήρωμα, το οποίο περιέχει το υπόλοιπο της εξίσωσης Laplace, $R_i^m = \frac{\partial^2 m_i}{\partial x_j^2} = 0$, που περιγράφει τη μετακίνηση του πλέγματος, έχει προστεθεί προκειμενου να αποφευχθεί ο υπολογισμός του $\delta x_k / \delta b_n$ στον όγκο $\Omega$ του υπολογιστικού χωρίου, για τον ίδιο λόγο που τα δύο πρώτα ολοκληρώματα έχουν προστεθεί προκειμένου να μη χρειαστεί να υπολογιστούν τα $\partial v_i / \partial b_n$ και $\partial p / \partial b_n$ στο $\Omega$ αντίστοιχα [11]. Δεδομένου ότι οι ροϊκές εξισώσεις ικανοποιούνται στο $\Omega$, $F_{aug} = F$ και $\delta F_{aug} / \delta b_n = \delta F / \delta b_n$.

Διαφορίζοντας την εξίσωση 4.3 και εν συνεχεία θέτοντας τους πολλαπλασιαστές των μεταβολών των ροϊκών ποσοτήτων ως προς τις μεταβλητές σχεδιασμού εντός των χωρικών ολοκληρωμάτων ίσους με το μηδέν, προκύπτουν οι συζυγείς πεδιακές εξισώσεις

$$R^q = -\frac{\partial u_j}{\partial x_j} + \acute{F}_\Omega^p = 0 \tag{4.4}$$

$$\begin{aligned} R_i^u = u_j \frac{\partial v_j}{\partial x_i} - \frac{\partial (v_j u_i)}{\partial x_j} - \frac{\partial}{\partial x_j} \left[ (\nu + \nu_t) \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] + \frac{\partial q}{\partial x_i} + \acute{F}_{\Omega,i}^v \\ + \widetilde{\nu}_a \frac{\partial \widetilde{\nu}}{\partial x_i} - \frac{\partial}{\partial x_l} \left( \widetilde{\nu}_a \widetilde{\nu} \frac{\mathcal{C}_Y}{Y} e_{mjk} \frac{\partial v_k}{\partial x_j} e_{mli} \right) = 0 \ , \ i = 1, 2, 3 \end{aligned} \tag{4.5}$$

$$R^{\widetilde{\nu_a}} = -\frac{\partial(v_j\widetilde{\nu}_a)}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\nu+\frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial\widetilde{\nu}_a}{\partial x_j}\right] + \frac{1}{\sigma}\frac{\partial\widetilde{\nu}_a}{\partial x_j}\frac{\partial\widetilde{\nu}}{\partial x_j} + 2\frac{c_{b2}}{\sigma}\frac{\partial}{\partial x_j}\left(\widetilde{\nu}_a\frac{\partial\widetilde{\nu}}{\partial x_j}\right)$$
$$+\widetilde{\nu}_a\widetilde{\nu}\mathcal{C}_{\widetilde{\nu}} + \frac{\partial\nu_t}{\partial\widetilde{\nu}}\frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right) + (-P+D)\,\widetilde{\nu}_a + \acute{F}^{\widetilde{\nu}}_{\Omega} = 0 \tag{4.6}$$

$$R^{m^a}_k = \frac{\partial^2 m^a_k}{\partial x_j^2} + \frac{\partial}{\partial x_j}\left\{u_i v_j\frac{\partial v_i}{\partial x_k} + u_j\frac{\partial p}{\partial x_k} + \tau^a_{ij}\frac{\partial u_i}{\partial x_k} - u_i\frac{\partial\tau_{ij}}{\partial x_k} - q\frac{\partial v_j}{\partial x_k}\right\} = 0 \tag{4.7}$$

Οι συζυγείς οριακές συνθήκες παρουσιάζονται στο κυρίως κείμενο στην αγγλική γλώσσα και παραλείπονται για λόγους συντομίας. Μετά την εξαγωγή των συζυγών εξισώσεων και των συζυγών οριακών συνθηκών, η τελική έκφραση των παραγώγων ευαισθησίας είναι

$$\begin{aligned}
\frac{\delta F_{aug}}{\delta b_n} &= T^{WF}_{SD} - \int_{S_{W_p}}\mathcal{SD}_1\frac{\partial\tau_{ij}}{\partial x_m}n_j t^I_i n_m n_k\frac{\delta x_k}{\delta b_n}dS - \int_{S_{W_p}}\mathcal{SD}_1\tau_{ij}\frac{\delta(n_j t^I_i)}{\delta b_n}\frac{\delta x_k}{\delta b_n}dS \\
&+ \int_{S_{W_p}}\mathcal{SD}_{2,i}v^I_{\langle t\rangle}\frac{\delta t^I_i}{\delta b_n}dS - \int_{S_{W_p}}\mathcal{SD}_{2,i}\frac{\partial v_i}{\partial x_m}n_m n_k\frac{\delta x_k}{\delta b_n}dS \\
&- \int_{S_{W_p}}\left[\left(\nu+\frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial\widetilde{\nu}_a}{\partial x_j}n_j + \frac{\partial F_{S_z}}{\partial\widetilde{\nu}}n_z + \acute{F}^{\widetilde{\nu}}_S\right]\frac{\partial\widetilde{\nu}}{\partial x_m}n_m n_k\frac{\delta x_k}{\delta b_n}dS \\
&- \int_{S_{W_p}}(-u_{\langle n\rangle}+\phi_{\langle n\rangle\langle n\rangle})\left(\tau_{ij}\frac{\delta(n_i n_j)}{\delta b_n} + \frac{\partial\tau_{ij}}{\partial x_m}n_m\frac{\delta x_k}{\delta b_n}n_k n_i n_j\right)dS \\
&- \int_{S_{W_p}}\phi_{\langle t^I\rangle\langle t^I\rangle}\left(\tau_{ij}\frac{\delta(t^I_i t^I_j)}{\delta b_n} + \frac{\partial\tau_{ij}}{\partial x_m}n_m\frac{\delta x_k}{\delta b_n}n_k t^I_i t^I_j\right)dS \\
&- \int_{S_{W_p}}(\phi_{\langle t^{II}\rangle\langle t^I\rangle}+\phi_{\langle t^I\rangle\langle t^{II}\rangle})\left(\tau_{ij}\frac{\delta(t^{II}_i t^I_j)}{\delta b_n} + \frac{\partial\tau_{ij}}{\partial x_m}n_m\frac{\delta x_k}{\delta b_n}n_k t^{II}_i t^I_j\right)dS \\
&- \int_{S_{W_p}}\phi_{\langle t^{II}\rangle\langle t^{II}\rangle}\left(\tau_{ij}\frac{\delta(t^{II}_i t^{II}_j)}{\delta b_n} + \frac{\partial\tau_{ij}}{\partial x_m}n_m\frac{\delta x_k}{\delta b_n}n_k t^{II}_i t^{II}_j\right)dS \\
&+ \int_{S_{W_p}}n_i\frac{\partial F_{S_{W_p,i}}}{\partial x_m}n_m\frac{\delta x_k}{\delta b_n}n_k dS + \int_{S_{W_p}}F_{S_{W_p,i}}\frac{\delta n_i}{\delta b_n}dS + \int_{S_{W_p}}F_{S_{W_p,i}}n_i\frac{\delta(dS)}{\delta b_n} \\
&- \int_{S_{W_P}}\frac{\partial m^a_i}{\partial x_j}n_j\frac{\delta x_i}{\delta b_n}dS + \int_{S_{W_p}}\mathcal{A}^{WF}_{\Delta}\frac{\partial\Delta^P}{\partial b_n}dS + \int_{S_W}\mathcal{A}^{WF}_{\Delta}\frac{\partial\Delta^P}{\partial b_n}dS \\
&+ \underbrace{\int_{S_{W_p}}\Delta_a R^{\Delta}n_k\frac{\delta x_k}{\delta b_n}dS - \int_{S_{W_p}}2\Delta_a\frac{\partial\Delta}{\partial x_j}n_j\frac{\partial\Delta}{\partial x_m}n_m n_k\frac{\delta x_k}{\delta b_n}dS}_{\Delta Term}
\end{aligned} \tag{4.8}$$

όπου

$$\mathcal{SD}_1 = -u^I_{\langle t\rangle} + \phi_{\langle t^I\rangle\langle n\rangle} + \phi_{\langle n\rangle\langle t^I\rangle} \tag{4.9}$$

$$\mathcal{SD}_{2,i} = (\nu+\nu_t)\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) n_j - q n_i + \frac{\partial F_{S_{W_p,k}}}{\partial v_i} n_k + \acute{F}^v_{S_{W_p,i}} \tag{4.10}$$

$$\phi_{ij} = \frac{\partial F_{S_{W_p,k}}}{\partial \tau_{ij}} n_k \tag{4.11}$$

$T^{WF}_{SD}$, $\int_{S_{W_P}} \mathcal{A}^{WF}_{\Delta} \frac{\partial \Delta^P}{\partial b_n} dS$ ανδ $\int_{S_W} \mathcal{A}^{WF}_{\Delta} \frac{\partial \Delta^P}{\partial b_n} dS$ συνοψίζουν τη συνεισφορά της διαφόρισης των συναρτήσεων τοίχου στις παραγώγους ευαισθησίας [14].

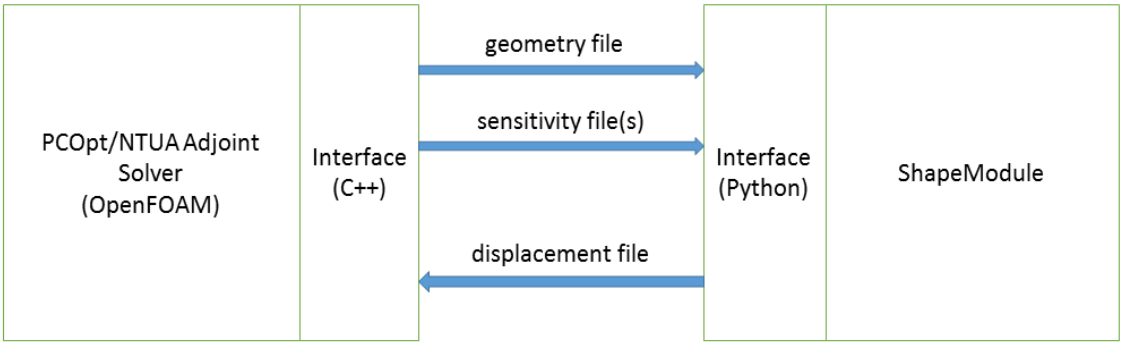
Ο όρος $\Delta Term$ της εξίσωσης 4.8 αντικατέστησε τον όρο $\int_\Omega \widetilde{\nu}\widetilde{\nu_a} \mathcal{C}_\Delta \frac{\partial \Delta}{\partial b_n} d\Omega$ που προκύπτει από τη διαφόριση του μοντέλου τύρβης, προκειμένου να μη χρειαστεί ο υπολογισμός των μεταβολών της απόστασης $\partial\Delta/\partial b_n$ στο $\Omega$ (βλ. υποκεφάλαιο 2.2.6). Αυτό κατέστη δυνατόν μέσω της προσθήκης του όρου $\int_\Omega \Delta_\alpha R^\Delta d\Omega$ στην έκφραση της επαυξημένης αντικειμενικής συνάρτησης, όπου $\Delta_\alpha$ είναι η συζυγής απόσταση και $R^\Delta = \frac{\partial (c_j \Delta)}{\partial x_j} - \Delta \frac{\partial^2 \Delta}{\partial x_j^2} - 1 = 0$ με $c_j = \partial\Delta/\partial x_j$, η ΜΔΕ Hamilton-Jacobi, η οποία δίνει μια πολύ καλή προσέγγιση του ευκλείδειου πεδίου απόστασης $\Delta$, [8, 15].

Αυτά ολοκληρώνουν τη θεωρία σχετικά με τη διατύπωση της συνεχούς συζυγούς μεθόδου που χρησιμοποιείται στις προσομοιώσεις που θα παρουσιαστούν στη συνέχεια.

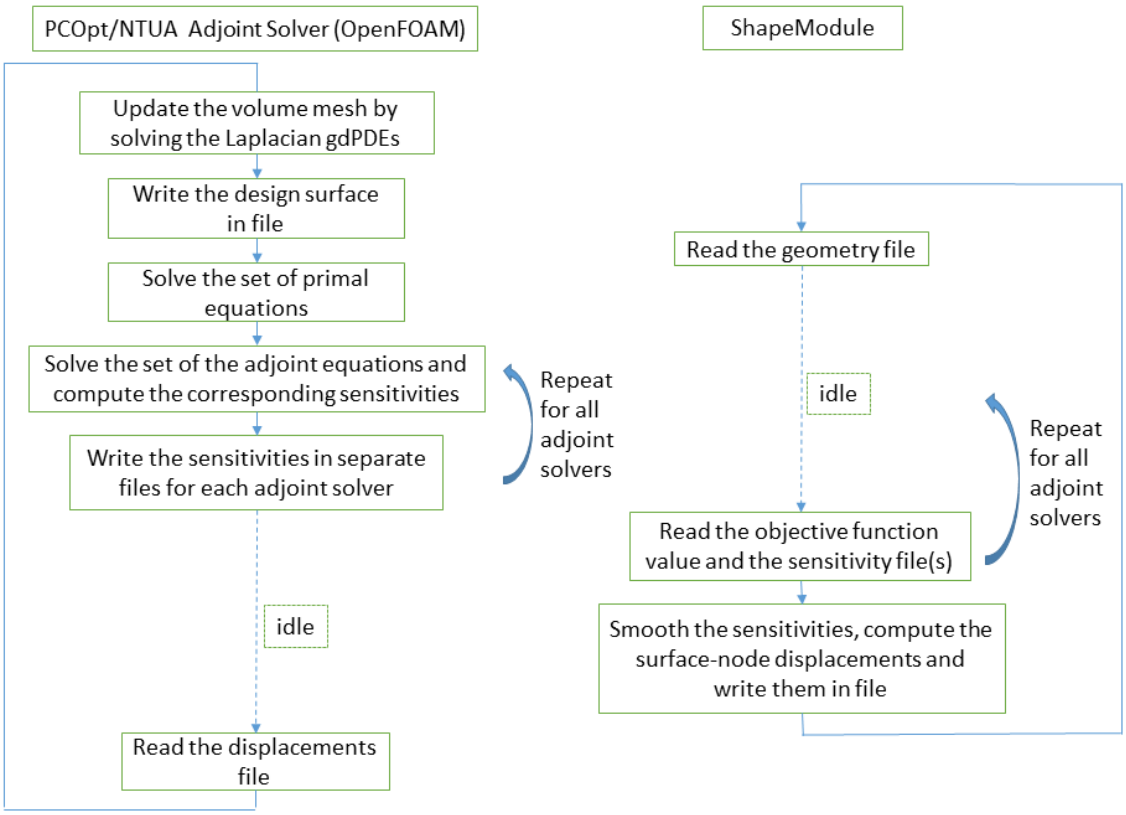
### Ενσωμάτωση του συζυγούς επιλύτη της ΜΠΥΡ&Β/ΕΜΠ στο λογισμικό βελτιστοποίησης μορφής της BMW

Στα πλαίσια της διπλωματικής εργασίας ο συνεχής συζυγής επιλύτης της ΜΠΥΡ&Β /ΕΜΠ ενσωματώθηκε στο λογισμικό βελτιστοποίησης μορφής ΄ShapeModule΄ της BMW. Το ShapeModule είναι φτιαγμένο ώστε να μπορεί να συνεργάζεται με διάφορους συζυγείς επιλύτες, οι οποίοι είναι υπεύθυνοι για τον υπολογισμό των παραγώγων ευαισθησίας της αντικειμενικής συνάρτησης ως προς τις συντεταγμένες των επιφανειακών κόμβων της υπό βελτιστοποίηση γεωμετρίας, ενώ το ίδιο υπολογίζει τις κομβικές μετατοπίσεις βάσει των παραγώγων αυτών εφαρμόζοντας την τεχνική Vertex Morphing. Η χρήση των επιφανειακών κόμβων ως μεταβλητών σχεδιασμού παρέχει το πλουσιότερο δυνατό χώρο σχεδιασμού για μια διακριτή γεωμετρία, αφού γίνεται χρήση όλων των διαθέσιμων βαθμών ελευθερίας της. Ωστόσο, είναι πιθανή η δημιουργία ανωμαλιών στη νέα γεωμετρία, πρόβλημα που αντιμετωπίζεται με την εφαρμογή ενός φίλτρου εξομάλυνσης πάνω στις παραγώγους ευαισθησίας και στις μετατοπίσεις κάθε κόμβου. Το συζευγμένο λογισμικό περιλαμβάνει δύο διεπαφές, με εκείνη που συνδέεται με το συζυγή επιλύτη να είναι γραμμένη σε γλώσσα προγραμματισμού C++, ενώ εκείνη που συνδέεται με το ShapeModule να είναι γραμμένη σε Python, σχήμα 4.1.

Το απλοποιημένο διάγραμμα της διαδικασίας βελτιστοποίησης, όπως αυτή εφαρμόζεται στη BMW φαίνεται στο σχήμα 4.2.



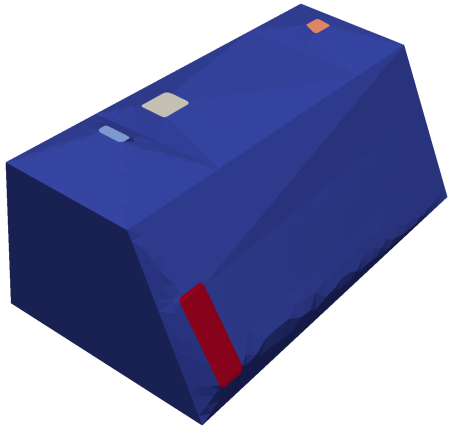

**Σχήμα 4.1:** *Η σύζευξη του λογισμικού της ΜΠΥΡ&Β/ΕΜΠ με το ShapeModule. Κάθε λογισμικό έχει μια διεπαφή (interface), μέσω της οποίας επιτυγχάνεται η επικοινωνία με το άλλο.*



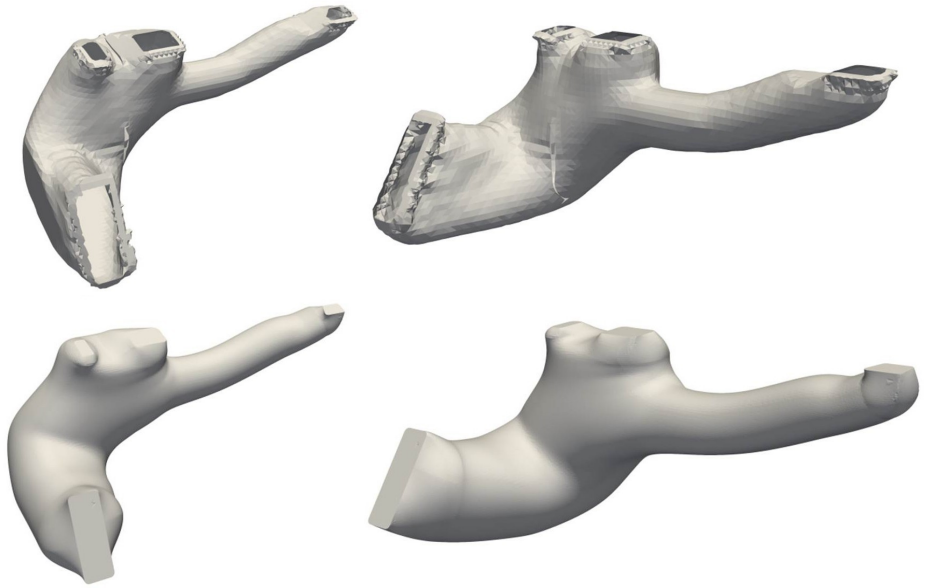

**Σχήμα 4.2:** *Το απλοποιημένο διάγραμμα της διαδιακασίας βελτιστοποίησης, όπως έχει εφαρμοστεί στη BMW.*

### Προσομειώσεις που πραγματοποιήθηκαν με το συζευγμένο λογισμικό ΜΠΥΡ&Β/ΕΜΠ και ShapeModule

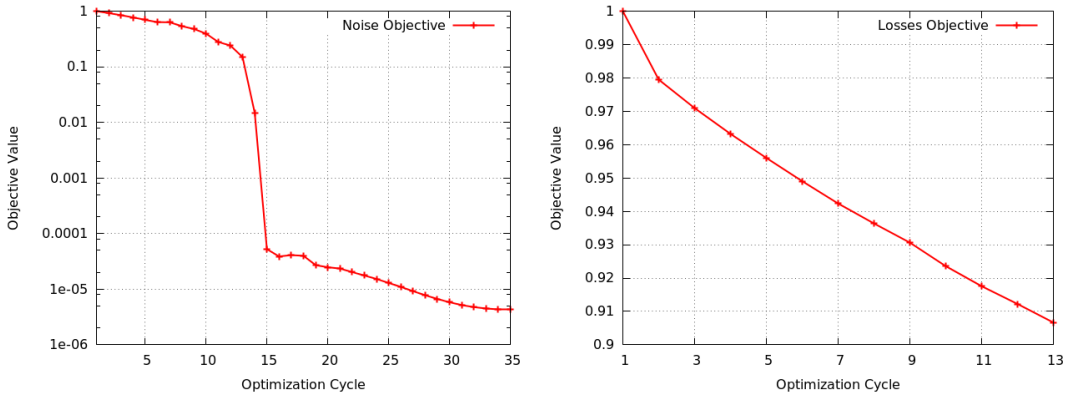
Το συζευγμένο λογισμικό της ΜΠΥΡ&Β/ΕΜΠ και του ShapeModule χρησιμοποιήθηκε για το σχεδιασμό αεραγωγού της καμπίνας επιβατικού αυτοκινήτου με τελικό στόχο την ελαχιστοποίηση του παραγόμενου θορύβου στις εξόδους του και των απωλειών ολικής πίεσης που προκαλεί στο διερχόμενο ρευστό. Οι αντικειμενικές συναρτήσεις που χρησιμοποιήθηκαν για το σκοπό είναι οι $F_{\nu_t} = \int_{\Omega'} \nu_t^2 d\Omega$ και $F_{P_t} = -\int_{S_{I,O}} \left(p + \frac{1}{2} v_i^2\right) v_j n_j dS$ αντίστοιχα. Αναλυτικότερη περιγραφή τους γίνεται στο υποκεφάλαιο 2.2.3 του αγγλικού μέρους. Δεδομένου ότι η αρχική γεωμετρία ήταν ένα κουτί με μία είσοδο και μια έξοδο, σχ. 4.3, έπρεπε αρχικά να πραγματοποιηθεί βελτιστοποίηση τοπολογίας με στόχο τη μείωση των απωλειών ολικής πίεσης προκειμένου να παραχθεί μια αρχική γεωμετρία πάνω στην οποία θα μπορούσε να εφαρμοστεί η βελτιστοποίηση μορφής. Στη συνέχεια έγινε εξομάλυνση της παραχθείσας επιφάνειας από τη βελτιστοποίηση τοπολογίας, σχ. 4.4. Έχοντας ως αφετηρία τη γεωμετρία που παράχθηκε μετά την εξομάλυνση, πραγματοποιήθηκαν 2 βελτιστοποιήσεις έχοντας ως στόχο είτε την ελαχιστοποίηση του θορύβου είτε την ελαχιστοποίηση του $\Delta P_t$. Το υπολογιστικό πλέγμα αποτελείται από 81000 κελιά και ο αριθμός Reynolds είναι $Re = 5500$, με βάση την υδραυλική διάμετρο της εισόδου του αγωγού. Η πορεία σύγκλισης της αντικειμενικής συναρτησης και για τις 2 περιπτώσεις φαίνεται στο σχήμα 4.5. Η τελική γεωμετρία παρουσιάζεται μόνο για την περίπτωση που αντικειμενική συνάρτηση ήταν ο θόρυβος, σχ. 4.6, ενώ παραλείπεται για λόγους συντομίας στην περίπτωση που αντικειμενική συνάρτηση ήταν οι απώλειες ολικής πίεσης.



**Σχήμα 4.3:** *Αρχικό κουτί κατά την έναρξη της βελτιστοποίησης. Τα στερεά όρια (μπλε), η είσοδος (κόκκινο) και οι τρεις έξοδοι στην πάνω πλευρά του κουτιού.*



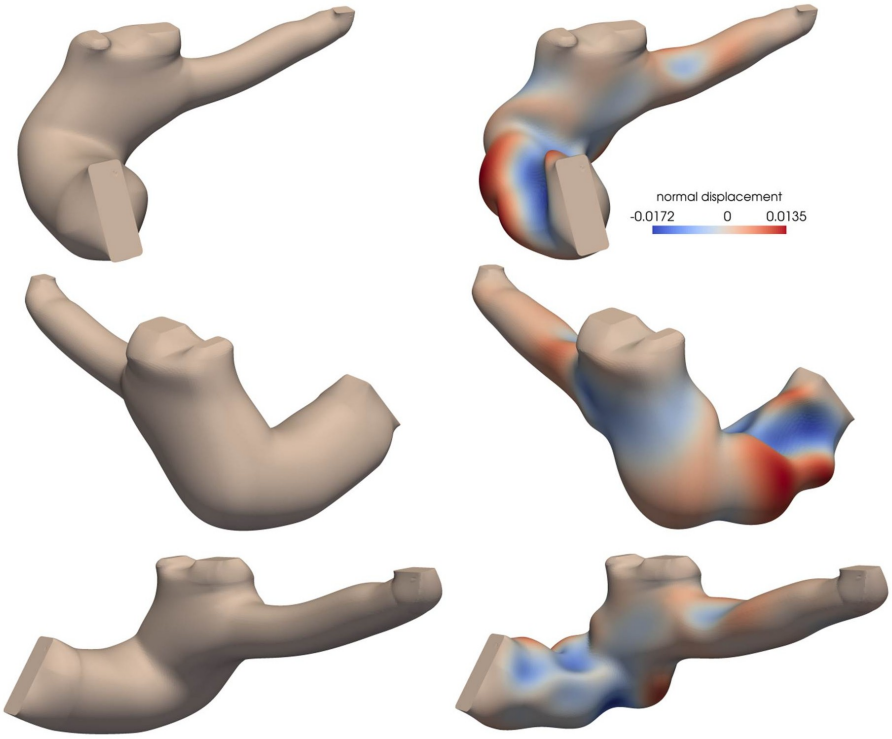
**Σχήμα 4.4:** *Ο αγωγός όπως προέκυψε από τη βελτιστοποίηση τοπολογίας του κουτιού του σχήματος 4.3 με στόχο την ελαχιστοποίηση των απωλειών ολικής πίεσης (άνω) και όπως διαμορφώθηκε μετά την εξομάλυνση επιφανείας (κάτω), υπό διαφορετικές γωνίες θέασης.*




**Σχήμα 4.5:** *Πορεία σύγκλισης της αντικειμενικής συνάρτησης του θορύβου (αριστερά) και του $\Delta P_t$ (δεξιά) σε ξεχωριστές περιπτώσεις βελτιστοποίησης.*

| | $F_{v_t}$ | $F_{P_t}$ |
|---|---|---|
| *Ελαχιστοποίηση '**Θορύβου**'* | $\mathbf{-99.99\%}$ | $-6.4\%$ |
| *Ελαχιστοποίηση* $\mathbf{\Delta P_t}$ | $-80\%$ | $\mathbf{-9.5\%}$ |

**Πίνακας 4.1:** *Συγκεντρωτικά αποτελέσματα. Με έντονη γραφή παρουσιάζεται η ποσοστιαία μείωση των αντικειμενικών συναρτήσεων που είχαν τεθεί ως στόχος της βελτιστοποίησης.*




**Σχήμα 4.6:** *Αεραγωγός επιβατικού αυτοκινήτου, $Re = 5500$, ελαχιστοποίηση θορύβου. Αρχική (αριστερά) και βελτιστοποίημένη γεωμετρία (δεξιά), υπό διαφορετικές γωνίες θέασης. Η αθροιστική κάθετη μετατόπιση είναι προσημασμένη και υποδεικνύει την κατεύθυνση κατά την οποία μετακινείται η επιφάνεια του αγωγού, είτε προς το εξωτερικό (κόκκινο) είτε προς το εσωτερικό (μπλε).*

### Προσομειώσεις που πραγματοποιήθηκαν με το πλήρες λογισμικό βελτιστοποίησης της ΜΠΥΡ&Β/ΕΜΠ

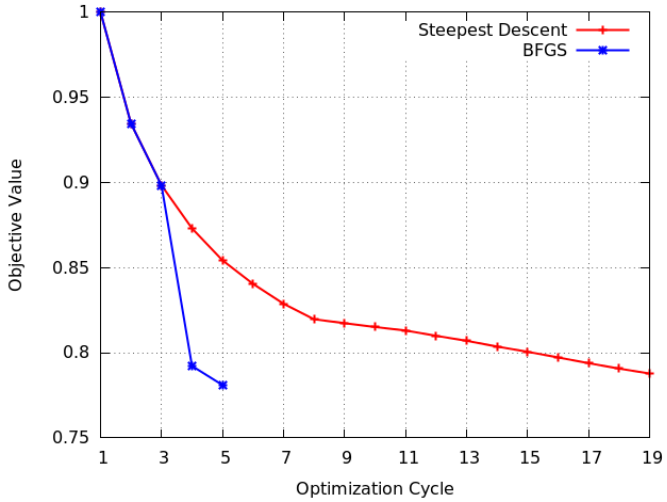
Επιπλέον των όσων περιγράφηκαν παραπάνω, στο πλαίσιο της διπλωματικής εργασίας αναπτύχθηκε η συνεχής συζυγής μέθοδος για αντικειμενική συνάρτηση, που εκφράζει την ομοιομορφία της ροής στην έξοδο της υπό βελτιστοποίηση γεωμετρίας, στο περιβάλλον του συζυγούς κώδικα της ΜΠΥΡ&Β/ΕΜΠ. Χρησιμοποιώντας αυτή την αντικειμενική συνάρτηση επιδιώκεται η ελαχιστοποίηση του αθροίσματος των τυπικών αποκλίσεων κάθε συνιστώσας της ταχύτητας από τη μέση τιμή της στην έξοδο του χωρίου, δηλαδή

$$J_\gamma = \frac{1}{2} \int_{S_O} (v_i - \bar{v}_i)^2 \, dS \tag{4.12}$$

ενώ ένα πιο απτό μέγεθος για την περιγραφή της ομοιομορφίας της ροής είναι ο αδιάστατος *δείκτης ομοιομορφίας*, ο οποίος δίνεται από τη σχέση $\gamma = 1 - \frac{\int_{S_O} ||v| - |\bar{v}|| dS}{2|\bar{v}|S_O}$ και λαμβάνει τιμές μεταξύ 0 και 1, με το 1 να εκφράζει την πλήρη ομοιομορφία της ροής.

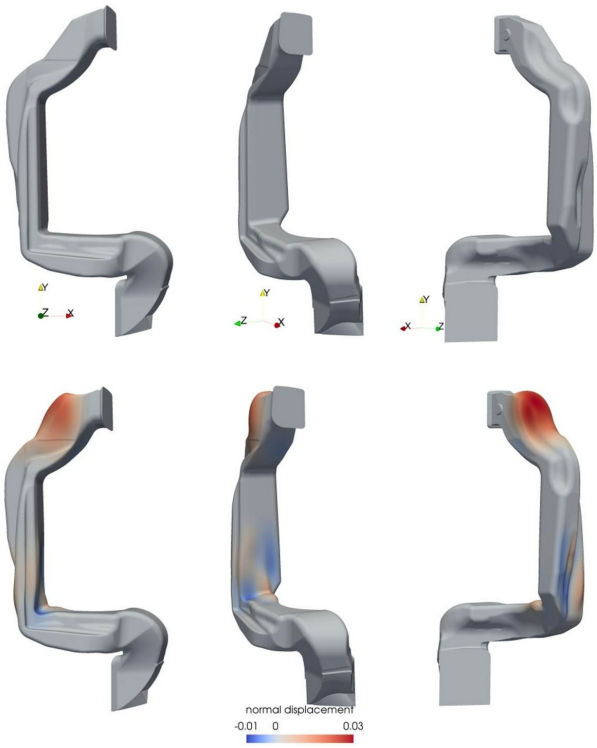
Χρησιμοποιώντας την αντικειμενική της ομοιομορφίας βελτιστοποιήθηκε αεραγωγός καμπίνας επιβατικού αυτοκινήτου. Σε αντίθεση με το λογισμικό βελτιστοποίησης μορφής της BMW, το λογισμικό της ΜΠΥΡ&Β/ΕΜΠ παραμετροποιεί το υπό βελτιστοποίηση τμήμα του αγωγού. Στη συγκεκριμένη προσομοίωση, χρησιμοποιήθηκαν ογκομετρικές B-Splines με 300 ενεργά σημεία ελέγχου. Το υπολογιστικό πλέγμα αποτελείται από $2{\times}10^6$ κελιά και ο αριθμός Reynolds είναι $Re = 14600$ με βάση την υδραυλική διάμετρο της εισόδου.

Χρησιμοποιώντας ως μοναδικό στόχο της βελτιστοποίησης την ελαχιστοποίηση της ανομοιομορφίας στην έξοδο του αγωγού, προέκυψαν τα αποτελέσματα των σχημάτων 4.7 και 4.8.

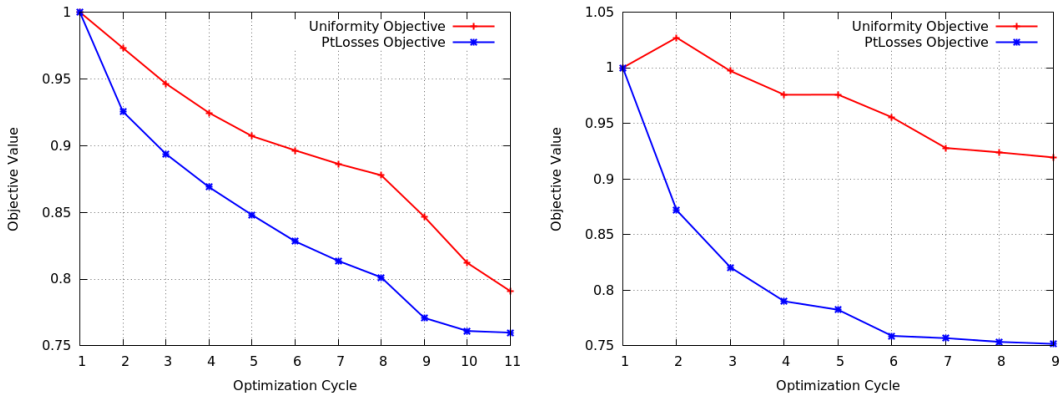



**Σχήμα 4.7:** *Πορεία σύγκλισης της αντικειμενικής της ομοιομορφίας. Σύγκριση της μεθόδου της απότομης καθόδου και της BFGS. Η αντικειμενική συνάρτηση μειώνεται κατά 22% σε 5 κύκλους βελτιστοποίησης με την BFGS μέθοδο.*

Στη συνέχεια, επιδιώχθηκε ο ίδιος αρχικός αγωγός να βελτιστοποιηθεί έχοντας ως στόχο την ελαχιστοποίηση τόσο της ανομοιομορφίας στην έξοδό του όσο και των απωλειών ολικής πίεσης που προκαλεί στο διερχόμενο ρευστό. Το πρόβλημα προσεγγίστηκε με δύο μεθόδους. Πρώτον, πραγματοποιώντας βελτιστοποίηση με στάθμιση των αντικειμενικών συναρτήσεων 250:1 υπέρ της αντικειμενικής της ομοιομορφίας, και δεύτερον, πραγματοποιώντας βελτιστοποίηση με αντικειμενική το $\Delta P_t$ υπό τον περιορισμό ισότητας η αντικειμενική της ανομοιομορφίας να μειωθεί κατά 10%. Στη βελτιστοποίηση με περιορισμό έγινε εφαρμογή της μεθόδου SQP (Sequential Quadratic Programming). Η σύγκλιση των αντικειμενικών συναρτήσεων για κάθε μία από τις προσεγγίσεις φαίνεται στο σχ. 4.9, ενώ η σύγκριση της αρχικής και βελτιστοποιημένων γεωμετριών φαίνεται στο σχ. 4.10.



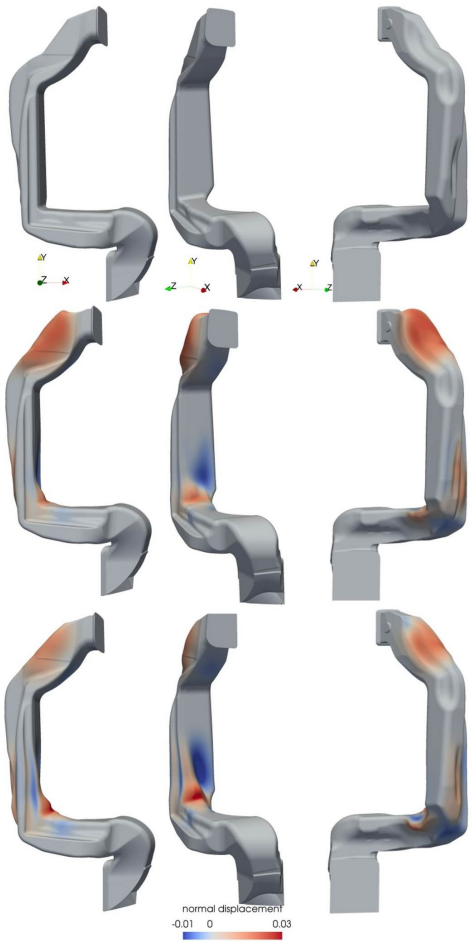

**Σχήμα 4.8:** *Αεραγωγός επιβατικού αυτοκινήτου, $Re = 14600$, ελαχιστοποίηση της ανομοιομορφίας στην έξοδό του (άνω άκρο). Αρχική (άνω) και βελτιστοποιημένη (κάτω) γεωμετρία ψπό διάφορες γωνίες θέασης. Η αθροιστική κάθετη μετατόπιση είναι προσημασμένη και υποδεικνύει την κατεύθυνση κατά την οποία μετακινείται η επιφάνεια του αγωγού, είτε προς το εξωτερικό (κόκκινο) είτε προς το εσωτερικό (μπλε).*




**Σχήμα 4.9:** *Πορεία σύγκλισης των αντικειμενικων συναρτήσεων της ομοιομορφίας και των απωλειών ολικής πίεσης, όταν πραγματοποιείται στάθμιση των αντικειμενικών συναρτήσεων (αριστερά) και όταν στόχος της βελτιστοποίησης είναι η μείωση του $\Delta P_t$ υπό τον περιορισμό η αντικειμενική συνάρτηση της ανομοιομορφίας να μειωθεί κατά 10% (δεξιά).*

| | $F_\gamma$ | $F_{P_t}$ |
|---|---|---|
| Ελαχιστοποίηση ανομοιομορφίας | **-22 %** | +5.5 % |
| Ελαχιστοποίηση ανομοιομορφίας και $\Delta P_t$ **με χρήση βαρών** | **-20 %** | **-24 %** |
| Ελαχιστοποίηση ανομοιομορφίας και $\Delta P_t$ **με περιορισμό ισότητας** | **-8 %** | **-25 %** |

**Πίνακας 4.2:** *Αεραγωγός επιβατικού αυτοκινήτου,* $Re = 14600$*, σύγκριση αποτελεσμάτων βελτιστοποίησης*




**Σχήμα 4.10:** *Αεραγωγός επιβατικού αυτοκινήτου,* $Re = 14600$*, βελτιστοποίηση με τις αντικειμενικές συναρτήσεις των απωλειών ολικής πίεσης και της ομοιομορφίας. Αρχική γεωμετρία (άνω), βελτιστοποίημένη με στάθμιση βαρών (μέση) και με περιορισμό για την αντικειμενική της ομοιομορφίας (κάτω). Η αθροιστική κάθετη μετατόπιση είναι προσημασμένη και υποδεικνύει την κατεύθυνση κατά την οποία μετακινείται η επιφάνεια του αγωγού, είτε προς το εξωτερικό (κόκκινο) είτε προς το εσωτερικό (μπλε)*

.

# Βιβλιογραφία


[1] Nadarajah, S.K. and Jameson, A.: *A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization.* AIAA Paper 2000-0667, AIAA 38th. Aerospace Sciences Meeting and Exhibit, Reno, NV, 2000.

[2] Τσαγγάρης, Σ.: *Μηχανική των Ρευστών.* Συμεών, Αθήνα, 2005.

[3] Γιαννάκογλου, Κ. Χ.: *Μέθοδοι Βελτιστοποίησης στην Αεροδυναμική.* Πανεπιστημιακές Εκδόσεις Ε.Μ.Π., Αθήνα, 2006.

[4] Giannakoglou, K.C., Papadimitriou, D.I., Papoutsis-Kiachagias, E.M., and Othmer, C.: *Adjoint methods in cfd-based optimization - gradient computation & beyond.* ECCOMAS 2012–European Congress on Computational Methods in Applied Sciences and Engineering, 2012.

[5] Spalart, P. and Allmaras, S.: *A one-equation turbulence model for aerodynamic flows.* AIAA Paper 1992-439, 30th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, USA, January 6–9 1992.

[6] Moukalled, F., Darwish, M., and Mangani, L.: *The Finite Volume Method in Computational Fluid Dynamics. An Advanced Introduction with OpenFOAM and MATLAB.* Springer, 2016.

[7] Spalding, D. B.: *A single formula for the law of the wall.* Journal of Applied Mechanics, 28:455–457, 1961.

[8] Papoutsis-Kiachagias, E.M.: *Adjoint methods for turbulent flows, applied to shape and topology optimization and robust design.* PhD thesis, Lab. of Thermal Turbomachines, N.T.U.A., Athens, 2013.

[9] Papoutsis-Kiachagias, E.M. and Giannakoglou, K.C.: *Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: Industrial applications.* Springer, 2014.

[10] Giannakoglou, K.C., Papoutsis-Kiachagias, E.M., Kavvadias, I.S., and Gkaragkounis, K.Th.: *Continuous Adjoint in Shape & Topology Optimization - Recent Developments & Applications.* Conference: Seminar on Adjoint CFD Methods in Industry and Research, At Wiesbaden, Germany, 2016.

[11] Kavvadias, I.S., Papoutsis-Kiachagias, E.M., and Giannakoglou, K.C.: *On the proper treatment of grid sensitivities in continuous adjoint methods for shape optimization*, volume 301. Journal of Computational Physics, 2015.

[12] Papoutsis-Kiachagias, E.M., Magoulas, N., Mueller, J., Othmer, C., and Giannakoglou, K.C.: *Noise reduction in car aerodynamics using a surrogate objective function and the continuous adjoint method with wall functions*, volume 122. Computers and Fluids, 2015.

[13] Zymaris, A.S., Papadimitriou, D.I., Giannakoglou, K.C., and Othmer, C.: *Continuous adjoint approach to the spalart-allmaras turbulence model for incompressible flows.* Computers & Fluids, 38(8):1528–1538, 2009.

[14] Zymaris, A.S., Papadimitriou, D.I., Giannakoglou, K.C., and Othmer, C.: *Adjoint wall-functions: A new concept for use in aerodynamic shape optimization.* Journal of Computational Physics, 229(13):5228–5245, 2010.

[15] Papoutsis-Kiachagias, E.M., Asouti, V.G., Giannakoglou, K.C., Gkagkas, K., Shimokawa, S., and Itakura, E.: *Multi-point aerodynamic shape optimization of cars based on continuous adjoint.* Structural and Multidisciplinary Optimization, 59(2):675–694, 2019.

[16] Bletzinger, Kai Uwe: *A consistent frame for sensitivity filtering and the vertex assigned morphing of optimal shape.* Springer, 2013.

[17] Hojjat, Majid, Stavropoulou, Electra, and Bletzinger, Kai Uwe: *The Vertex Morphing method for node-based shape optimization*, volume 268. Computer methods in applied mechanics and engineering, 2014.

[18] Papoutsis–Kiachagias, E.M. and Giannakoglou, K.C.: *A parameterization and mesh movement strategy based on volumetric B-splines. Applications to shape optimization.* NTUA/PCOpt/2015/01 REPORT, 2015.

[19] Nocedal, J. and Wright, S.J.: *Numerical Optimization.* Springer, 1999.