Politecnico di Milano

Scuola di Ingegneria Industriale e Dell'Informazione
Corso di Laurea Magistrale in Ingegneria Aeronautica

Develop of a Novel Procedure for the Topological Optimization of an Engine Exhaust Mixer

Relatore: Prof. Alberto Matteo Attilio Guardone

Tutor Aziendale: Ph.D. Luca Patricelli

Elaborato di Laurea di:
Federico COLOMBO
Matr. 920340

Anno Accademico 2019/20

A mia Madre e a mio Padre

This research has received funding from the Clean Sky 2 Joint Undertaking (JU) under grant agreement No 945542. The JU receives support from the European Union's Horizon 2020 research and innovation programme and the Clean Sky 2 JU members other than the Union.

Ringraziamenti

Desidero anzitutto ringraziare Leonardo - Helicopters Division per avermi fornito questa possibilità di tesi all'interno della propria azienda. Un ringraziamento particolare va al mio Tutor aziendale Luca Patricelli, il quale mi ha seguito con costanza e dedizione in questo percorso, rendendosi sempre disponibile ad aiutarmi nello sviluppo del lavoro di tesi, e a Vincenzo Vitiello per avermi affiancato nelle analisi effettuate.

Un particolare ringraziamento anche a Flavio Cimolin, per l'enorme disponibilità dimostrata nel trasmettermi le sue conoscenze in ambito CFD e per il supporto tecnico fornitomi nell'utilizzo di STAR-CCM+.

Desidero ringraziare il Professor. Alberto Guardone per essersi mostrato sempre disponibile in questi anni al Politecnico e per aver accettato di segurimi in questo percorso in qualità di relatore.

Un ringraziamento particolare a mia Madre e a mio Padre, per avermi aiutato e sostenuto nelle mie scelte e per avermi permesso di studiare sempre con tranquillità e serenità, e a mio fratello Andrea, che oltre ad aiutarmi in momenti difficili di questo percorso si è sempre dimostrato una guida affidabile per me anche al di fuori del contesto accademico.

Ringrazio di cuore la mia ragazza Rachele, per avermi sopportato e supportato in questi anni, per avermi regalato gioie anche nei momenti per me più difficili e per essermi sempre stata accanto quando ne avevo bisogno.

Un ringraziamento speciale va a tutti i miei amici più stretti, Anna, Elena, Sharon, Veronica, Giacomo, Emanuele e Igor con cui condivido i momenti più importanti ormai da tantissimi anni e che rappresentano la mia seconda famiglia.

Un ringraziamento va ovviamemente ai miei compagni e colleghi di studi, tra i quali Andrea, Angelo, Alice, Alberto, Alessandra, Alessandro, Federico, Francesco, Gaia, Matteo Collura, Matteo Copercini e Sergio, con i quali ho inizato questo percorso più di 5 anni fa, con cui ho diviso e condiviso le giornate di studio e a cui devo gran parte dei risultati ottenuti.

Desidero ringraziare la mia famiglia tutta, i miei cugini, i miei zii e i miei nonni per l'affetto e il sostegno mostratomi ogni giorno e a tutti coloro i quali mi hanno accompagnato in questi anni, per lunghi o brevi periodi, e che, anche se qui non nominati, mi hanno permesso di costruire e raggiungere questo traguardo.

Federico Colombo
28 Aprile 2021

Sommario

Nell'affrontare le fasi di design o di progettazione, la ricerca della forma migliore di un o componente per il soddisfacimento di una determinata performance ha da sempre rappresentato uno dei problemi principali nel campo ingegneristico. L'ottimizzazione topologica nasce alla fine degli anni 80 per fornire una risposta a questo tipo di problema. Nonostante si sia sviluppata nel campo strutturale, dove è ormai ben consolidata, nel corso degli anni è stata ampliata a maggior varietà di campi fisici tra cui la fluidodinamica, nel quale tuttavia è una metodologia ancora acerba.

Lo scopo di questa tesi è lo sviluppo di una procedura solida e facilmente replicabile per l'ottimizzazione topologica di componenti aerodinamici in campi fluidodinamici complessi. Il modello utilizzato per lo sviluppo delle analisi è basato sul modello di penalizzazione di Brinkman, il quale permette di simulare il dominio solido all'interno di quello fluido attraverso un mezzo poroso, definito implicitamente tramite l'interfaccia di una funzione di livello (LSF).

Dopo una breve introduzione all'ottimizzazione topologica e ai vari modelli ad oggi sviluppati, il modello sopra descritto viene presentato nel suo formalismo matematico. Tramite semplici test case vengono analizzate le sue performance e l'errore introdotto dal mezzo poroso, assieme allo sviluppo della procedura per la corretta definizione dei vincoli e della funzione obiettivo tramite l'analisi delle sensitività. Viene inoltre presentata una prima correzione al campo turbolento generato all'interno del mezzo poroso. L'impatto e l'efficacia della metodologia sviluppata sono infine testati sullo scarico primario del motore del Next-Generation Civil TiltRotor sviluppato da Leonardo Helicopter Division nell'ambito del progetto Clean Sky 2 - Fast Rotor Craft. La topologia ottenuta è dunque confrontata con quella base, sviluppata con metodi ingegneristici classici, per confrontarne le prestazioni e i miglioramenti apportati dal nuovo modello.

La tesi è stata sviluppata in collaborazione con Leonardo Helicopter Division. Tutte le analisi sono state effettuate tramite il software commerciale Siemens PLM STAR-CCM+ (v. 2020.3).

Parole chiave: Ottimizzazione Topologica; Metodi Level-set; Modello di Penalizzazione di Brinkman; Adjoint; Analisi di Sensitività.

Abstract

In facing the design of a generic engineering component, the search for the best shape and material distribution that could guarantee the best performances has always been one of the greatest problems in the engineering field. Topology optimization born in the late 1980s to try to give an answer at such a problem. Though it was born for the structural optimization, where it is now well-established, the basic ideas are applicable to all those physics where the governing equations can be described by a set of partial differential equations (PDEs), which therefore enlarges its applications to a high variety of different physics such as acoustics, electromagnetism, heat conduction and fluid flow, where however it is still immature.

The aim of this thesis is the development of a robust and easy-to-replicate procedure for the topology optimization of aerodynamic component in a complex fluid-dynamic scenario. The mathematical model on which the analysis are based is the Brinkman penalization model, which simulates the solid domain through a porous media, implicitly defined through a Level-Set function (LSF) interface.

Since the topology optimization has been only recently introduced in Computational Fluid Dynamics tools, this work aims to investigate all the steps required for the optimization. These include the choice of the constraints and of the objective function based on the sensitivity analysis, together with analysis of the error introduced by the Brinkman Penalization model and the introduction of a turbulence correction. To prove the efficiency of the developed procedure, the topology optimization is at the end applied to a non-trivial aerodynamic problem: primary exhaust of the tilt-rotor demonstrator developed by Leonardo Helicopter Division within the Clean Sky 2 Fast Rotor Craft framework. The obtained typologies are then compared with the primary exhaust developed by using standard methodologies.

This thesis was developed in collaboration with Leonardo Helicopter Division. All simulations were performed using Siemens PLM commercial software STAR-CCM+ (v. 2020.3).

Keywords: Topology Optimization; Level-set Method; Brinkman Penalization; Adjoint; Sensitivity analysis.

Contents

1 Introduction **1**

1.1 Next-Generation Civil TiltRotor (NGCTR) and the Clean Sky project 2

1.2 Objective of the Thesis 5

1.3 Structure of the Thesis 6

2 Topology Optimization **9**

2.1 State of the art 11

2.2 Introduction to Shape Optimization 13

2.3 Topology optimization methods 15

2.3.1 Density-Based Method (SIMP) 16

2.3.2 Level-set methods 17

2.3.3 Adjoint Topology Optimization Model 18

2.3.3.1 Discrete Adjoint model 19

2.3.3.2 Adjoint topology optimization model 24

3 STAR-CCM+ Environment and Workflow **29**

3.1 Geometry definition 29

3.2 Mesh models and parameters 30

3.3 Physics models for the Adjoint Topology optimization 32

3.4 Adjoint Topology Optimization setup 36

3.4.1 Adjoint Setup 36

3.4.2 Report definition 37

3.4.3 Topology Optimization Loop 40

3.4.4 Stopping Criteria for the Topology Optimization 41

4 Procedure Development and Test Cases **43**

4.1 Quasi-2D mass flow splitting 43

4.1.1 Boundary conditions 44

4.1.2 Mesh Analysis 45

4.1.2.1 Mesh models and size influence on topology optimization 46

4.1.3 Penalty parameter 50

4.1.4 StepSize parameter 53

4.1.5 Manual Normalization and Sensitivity analysis 56
4.1.6 Optimization results and validations 63
4.2 Application on a real 3D case: Inlet Particle Separator 67
4.2.1 Problem introduction and geometry definition 67
4.2.2 Optimization Results 71
4.2.3 Topology physic error analysis 75

5 Topology Optimization of the Primal Exhaust of the NGCTR Engine 79
5.1 Definition and Setup of the Problem 79
5.1.1 Engine base geometry 80
5.1.2 Problem Setup and Mesh Analysis 81
5.2 Mass Flow Secondary Exhaust optimization 87
5.2.1 Optimization with a quarter of the domain 89
5.2.2 Mid-Domain optimization 95
5.2.3 Full-Domain optimization 96
5.3 Turbine Pressure optimization 102
5.4 Velocity field comparison 105

6 Conclusions 109
6.1 Future Developments 111

Appendix 113

A Jet Pump Effect 115

B Mass Flow Splitting Test Cases - Images 119
B.1 Penalty analysis 119
B.2 StepSize Analysis 120

List of Figures

1.1 Example of structures developed with the topology optimization approach . . . 2

1.2 Example of Fluid-dynamic Topology Optimization - fuel injector . . . 2

1.3 Example of Topology Optimization for thermal problems . . . 2

1.4 Clean Sky 2 Logo . . . 3

1.5 Airbus RACER compound helicopter . . . 4

1.6 Next-Generation Civil TiltRotor rendering. Note the new idea of tilting system with respect to the one of the AW609 . . . 4

1.7 AW609 during flight test. Note the tilting system that was designed for rotating all the engine bay . . . 5

1.8 example of an exhaust mixer for aeronautical purpose . . . 6

2.1 Example of topology optimization results . . . 10

2.2 Topology optimization typical workflow . . . 11

2.3 Example of topology optimization for a fluid domain . . . 12

2.4 Example of a Level Set function [48] . . . 18

2.5 Differences between Standard Gradient Descent (right) and Stochastic Gradient Descent (left) . . . 26

2.6 Difference of convergence for a SGD with (right) and without (left) momentum . . . 26

3.1 Example of a Trimmed Mesh [41] . . . 31

3.2 Examples of a Polyhedral Mesh [41] . . . 31

3.3 TVRScale function . . . 34

3.4 Turbulence viscosity ratio field with TVRScale effect. The isoline inside the domain highlights the porous media . . . 35

3.5 Effect of turbulence correction on the reduction of the error . . . 35

3.6 Difference on Pressure Drop computation . . . 38

3.7 Example of a Pressure Cost Function. $\alpha_p = -1$, $\beta_p = 10$, $P_{max} = 100Pa$, $P_{norm} = 500Pa$ . . . 38

3.8 Example of a SVR Cost Function for a request of a minimum SVR value. $\alpha_{svr} = 1$, $\beta_{svr} = 0.001$, $SVR_{min} = 0.15$, $SVR_{norm} = 0.05$ . . . 40

3.9 Topology optimization Loop . . . 41

4.1 Initial domain of the mass flow splitting test case . . . 44

4.2 Difference from a mesh generated with or without the alignment tool 46
4.3 Mesh convergence analysis - Pressure Drop. Values are normalized with their converged value 47
4.4 Mesh convergence analysis - Mass flow ratio - Mass flow splitting test case. 47
4.5 Influence of the mesh model and mesh size on the topology optimization result 48
4.6 Difference between the introduction of the Adaptive Mesh model. The outline depicts the level set boundary. 50
4.7 Mass Flow behaviour comparison - Penalty analysis 52
4.8 SVR behaviour comparison - Penalty analysis 52
4.9 Pressure Drop behaviour comparison - Penalty analysis 53
4.10 Final topology - Penalty analysis 53
4.11 SVR comparison - Step Size Analysis - Penalty = 1000 54
4.12 Pressure drop comparison - Step Size Analysis - Penalty = 1000 . . . 54
4.13 SVR behaviour comparison - Step Size Analysis - Penalty = 10000 . . 55
4.14 Final Topology comparison - Step Size Analysis - Penalty = 10000 . . 55
4.15 Sensitivity Loop inside the general TO Loop 57
4.16 Mass flow Report Normalization - Case A VS Case B. Both reports are defined with $\dot{m}_{norm} = 0.001kg/s$, but they differ for the normalization: $\alpha_{\dot{m}_A} = 500$ while $\alpha_{\dot{m}_B} = 5000$. 58
4.17 Sensitivity of the Pressure Drop w.r.t the material indicator at iteration 1 (a) and 66 (b) for Case A 60
4.18 Mass Flow behaviour during the optimization process: while Case A (a) managed to satisfy the constraint, Case B (b) diverged. 60
4.19 Optimization results - Mass flow splitting test case. 61
4.20 Pressure Drop behaviour during the optimization loop - Mass flow splitting test case 62
4.22 Optimized geometry reconstruction - Trim (a) VS Poly (b) mesh models 63
4.23 Difference in the velocity field for the reconstructed geometry - Trim (a) VS poly (b) 64
4.24 Turbulence viscosity ratio field comparison: without TVR (a), with TVR (b) and real geometry (c) 65
4.25 Turbulence kinetic energy field comparison: without TVR (a), with TVR (b) and real geometry (c) 65
4.26 Difference between the values of the Sensitivity of the Mass flow w.r.t the material indicator - Case A VS Case B. 66
4.27 Reference geometry of the IPS and starter ducts (in green) 67
4.28 Mesh convergence analysis for the base geometry of the IPS test case. Values are normalized with the converged value 68
4.29 Streamlines in the connection zone. A huge circulation is generated after the junction. Blue indicates lower values 69
4.30 Section of the turbulent viscosity ratio. Values are normalized according to the maximum value 69

4.31 Surface sensitivity of the boundaries with respect to the starter pressure. Arrows point in the direction of increasing the objective function, so to reduce the pressure an opposite modification is required. . 69

4.32 Geometry for the topology optimization of the IPS case. Light blue is used to highlight the TO-Region but only the dark blue boundaries are editable 70

4.33 IPS and Starter pressure behaviour during the optimization iterations: Manual normalization case. Values are normalized with the one at 1^{st} iteration. 72

4.34 Behaviour of an automatic normalization optimization with a too high penalty value. Values are normalized with the one at 1^{st} iteration. . . 72

4.35 Comparison of the final topology between automatic (a) and manual (b) report defintion and normalization 73

4.36 Streamline on the Starter duct before the junction. Note the birth of a rotational component in the flow 73

4.37 Effect of the two support-like structures on the flow behaviour 73

4.38 Final topology for the maximization of the mass flow case. Highly 3D structures are evident in the new topology that help the natural flow development 74

4.39 Flow behaviour in the optimized geometry. Note how the geometry follow the swirl of the starter before the mixing. 74

4.40 Temperature field inside the porous media. The convection is simulated with an effective thermal conductivity. Values are scaled w.r.t the maximum value 76

5.1 Engine bay (in yellow) with 3 of the 4 inlets (in purple) and the secondary exhaust (in red) 80

5.2 Internal components of the engine bay: In gray all the engine block with the actual geometry of the primal exhaust in orange 80

5.3 Part of the mounting system of the Primal Exhaust 83

5.4 Initial Condition for the Material Indicator (in red). In green is highlighted the TO-Region 84

5.5 Example of the result of the *InitialConditionFunction* with a too coarse mesh 84

5.6 Mesh convergence analysis - The reports are normalized with their asymptotic value 85

5.7 Normalized Velocity profile at outlet section - Mesh convergence analysis: Data refer to Base Size 86

5.8 Normalized Temperature profile at outlet section - Mesh convergence analysis: Data refer to Base Size 86

5.9 Detail of the Prism layer reduction in the mesh. Here, the Starter pipe reduces the fluid domain and the prism layer general settings cannot fit correctly 87

5.10 Minimum backpressure coefficient allowable for the turbine exhaust section 88

5.11 Quarter of the Engine bay: in blue the symmetry boundaries 89

5.12 Comparison of the flow direction and magnitude between the Full Domain (a) and the Quarter Domain (b) at the secondary exhaust section. Blue indicates lower values. 90

5.13 Pressure turbine report definition: $\alpha_p = -0.1$, $\beta_p = 20$, $P_{norm} = 0.257$, $P_{max} = 2.32$ (w.r.t the normalized value). 91

5.14 SVR report definition: $\alpha_{svr} = 1$, $\beta_{svr} = 0.1$, $SVR_{norm} = 0.01$, $SVR_{min} = 0.11$ 91

5.15 Sensitivity values for the objective (a) and pressure constraint (b) report. Values are normalized w.r.t the greatest value. 92

5.16 Optimized topology - Quarter of domain 92

5.17 Convergence plot - objective function and constraint reports 92

5.18 Comparison between Base (a), topology reconstructed (b) and CAD (c) geometry of the primal exhaust 92

5.19 Comparison of the flow behaviour between the Base exhaust (a) and the optimized one (b) (c). Values are normalized with the greatest one. 93

5.20 Comparison between the turbulence viscosity field with the exhaust simulated as porous media (in black) (a) and the one of the real geometry (b). Values are normalized with the greatest one. 94

5.21 Comparison between the turbulence kinetic energy with the exhaust simulated as porous media (in black) (a) and the one of the real geometry (b). Values are normalized with the greatest one. 94

5.22 Final topology - Mid Domain optimization 95

5.23 Velocity field in the lobed region. Comparison between the outer and the middle lobes - Mid Domain case. Values are normalized w.r.t the greatest one. 96

5.24 Streamlines comparison between the outer (a) and the middle (b) lobes - Mid Domain case. Note how the secondary flow enters the primal exhaust in case (b). Values are normalized w.r.t the greatest one. 96

5.25 Defeatured engine bay for the Full-Domain analysis - In purple the new engine bay inlet boundary replace the air intakes 97

5.26 Final topology - Full Domain Mass-Flow-secondary maximization case 97

5.27 Comparison between the Quart-Domain (in transparency) and the Full-Domain topology result 98

5.28 Convergence plot - objective function and constraint reports. The solution converged after 82 iterations ($\approx 4hrs$ on 48 cores) 99

5.29 Sensitivity values for the objective (a) and pressure constraint (b) report at the first iteration - Full Domain case. Values are normalized with the greatest one. 99

5.30 Isosurface of the porous media after 1 optimization iteration compared with the base geometry of the exhaust 99

5.31 Final topology - hand-made CAD model based on the topology result 100

5.32 Differences in the velocity field near the horn-like shapes with the final CAD (a) and with the reconstructed topology of the Quart-Domain (b). 101
5.33 Flow behaviour with the optimized primal exhaust - Mass Flow optimization case. Note the huge recirculation generated in the secondary exhaust- 101
5.34 Convergence plot - Pressure turbine minimization case - objective function and constraint reports. The solution stopped after 130 iterations ($\approx 7hrs$ on 48 cores) 102
5.35 Final topology obtained - Pressure minimization case 103
5.36 Detail of the final topology - Pressure minimization case. 103
5.37 CAD model of the optimal solution - Pressure minimization case. . . 103
5.38 Velocity field near the A-Shaped feature of the optimized exhaust . . 104
5.39 Effect of the triangular hole on the primal flow 104
5.40 Velocity profile at outlet section - Comparison between the obtained configurations 106
5.41 Temperature profile at outlet section - Comparison between the obtained configuration 106
5.42 Comparison of the flow field in different sections of the exhaust - Base geometry, Mass-flow optimized and Pressure optimized. Values are normalized with respect to the biggest one. 107

A.1 Basic scheme of the jet pump principle [47] 115
A.2 Jet Pump scheme 116

B.1 Mass Flow behaviour - Penalty Analysis 119
B.2 Pressure Drop behaviour - Penalty Analysis 119
B.3 SVR behaviour - Penalty Analysis 120
B.4 final topology - Penalty Analysis 120
B.5 Mass Flow comparison - StepSize Analysis - Penalty = 1000 120
B.6 Final Topology - StepSize Analysis - Penalty = 1000 121
B.7 Mass Flow behaviour comparison - Step Size Analysis - Penalty = 10000 121
B.8 Pressure Drop behaviour comparison - Step Size Analysis - Penalty = 10000 122

List of Tables

3.1 Physical models. In bold the models required for the TO 33

4.1 Mesh convergence analysis - Mass flow splitting test cases 46
4.2 Mesh parameters: Mass flow splitting test case. Percentages refer to the Base Size 50
4.3 Effect of the Penalty parameter on the solution. In red are characterized the values obtained in the last iteration of the non converged optimizations 51
4.4 Comparison of the results of the optimization between the mesh models and the reconstructed (smoothed) geometry 63
4.5 Comparison of the results predicted by the topology optimization with or without the TVRScale function 65
4.6 Mesh parameters: IPS Test case. Percentage referees to the Base Size 68
4.7 Comparison between the improvement given from the automatic normalization or the manual one. Values are normalized with respect to the maximum IPS constraint value 73
4.8 Comparison between the results of the IPS optimization cases. In bold are characterized the boundary conditions. Pressures are normalized w.r.t the maximum IPS constraint value, while mass flows w.r.t the starter boundary condition 75
4.9 Error introduced by the Topology physic model - Case 1. Pressure values are normalized w.r.t the maximum IPS constraint value, while temperatures w.r.t the Starter boundary condition 76
4.10 Error introduced by the Topology physic model - Case 2. Pressure values are normalized w.r.t the maximum IPS constraint value, while mass flows and temperatures w.r.t the Starter boundary condition . . 76

5.1 Mesh parameters: Engine primal exhaust optimization. Percentage refer to the Base Size 86
5.2 Comparison of the main parameters of the base primal exhaust between the real component and the porous-made one - Quart-Domain case. Mass flows and Pressures are normalized with the Base Geometry values 90
5.3 Results comparison: Quart Domain - Mass flow maximization case. Values are normalized with the Base Geometry values. 93
5.4 Results comparison: Mid Domain - Mass flow maximization case. Values are normalized according to the Quart Domain case. 96

5.5 Results of the optimization - Mass Flow Secondary maximization. Values are normalized according to the Quart Domain case. In red the results of the topology physics, since referred to the defeatured geometry 100

5.6 Results of the Pressure turbine minimization optimization. Values are normalized according to the previous cases. 103

1 Introduction

Topology optimization was born at the end of the 1980s as an alternative design technique for structural components to the classical methods and the already developed methods of shape and dimension optimization. The great advantage introduced by this methodology lies in the fact that there is no need for a predefined shape, but only in the available design space. Topological optimization is posed as a material arrangement technique but it brings with it two major advantages: the first, obviously, lies in identifying the best material distribution for satisfying the various constraints introduced and the objective function. The second resides in how such distribution is obtained and, above all, in how much time: through the topological optimization it is in fact possible to obtain the best form in a very short amount of time, avoiding other longer methodologies such as the procedures of trial and error or those based on experiences of the designer. However, one of the problems introduced by this methodology lies in the fact that the geometry obtained in this way is "natural" and, therefore, difficult to replicate with the standard methodologies of CAD design and manufacturing. An aid to the development of this technology comes from the parallel development of Additive Manufacturing technologies, which have greatly expanded the production capabilities of more complex shapes and geometries and have opened up the possibility of generating and testing geometries obtained from topological optimization as they are and not a simplified replica of them.

The great advantages brought to design by topological optimization in the structural field have led to its application to various other fields, including fluid dynamics, thermal (and thermo-fluid dynamics), acoustic, electromagnetic, etc. In such fields its development has been rendered but more complex and of difficult implementation, in how much to the base of one good success of the optimization there is a correct simulation of the physical field in examination. If in the structural field this does not represent a problem, in the field under examination (that is the fluid-dynamic one) it is a great limitation to the possibilities of development: the simulation of a solid inside a fluid must be able to replicate its effects, not only in terms of not passage of fluid, but also and above all in terms of development of the turbulence.

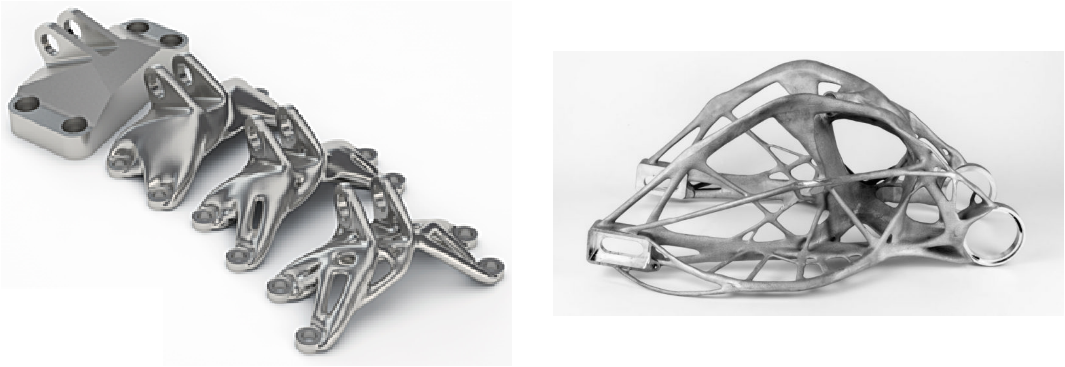


Figure 1.1: Example of structures developed with the topology optimization approach

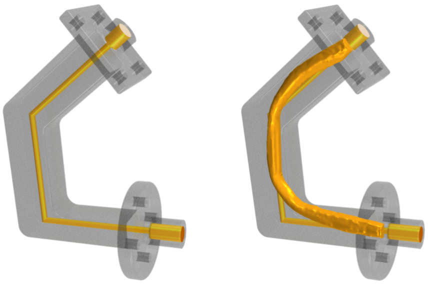


Figure 1.2: Example of Fluid-dynamic Topology Optimization - fuel injector

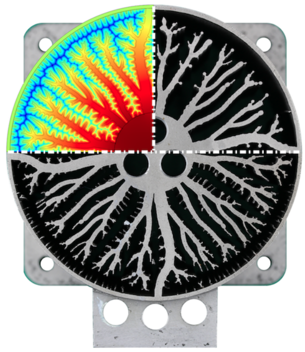


Figure 1.3: Example of Topology Optimization for thermal problems

These issues are still present in the models used for the implementation of this optimization methodology, thus they limit considerably the applications: most of them are in fact related to the study of the optimal geometry of ducts or piping systems for the reduction of pressure drops or for an uniformity of flow output, but it can not go beyond these applications. In this thesis we want to investigate these issues and these aspects, acquiring notions on the physical limits introduced by the optimization models (illustrated in chapter. 2) and some possible corrections to them, with the aim of applying topological optimization to a more complex aerodynamic component: the primary exhaust of an aircraft engine, in particular in the GE CT7-2E1 installed on the Next-Generation Civil TiltRotor developed by Leonardo Helicopters.

1.1 Next-Generation Civil TiltRotor (NGCTR) and the Clean Sky project

In the last decades, the development of new technologies suitable for increasing the efficiency and reducing consumption and emissions of CO_2 and NO_x has become more and more important. For this purpose, the Clean Sky Joint Undertaking (CSJU), a public-private partnership between the European Commission and the European aeronautics industry, was born. The main goal of the CSJU is to coordinate and fund research activities to deliver significantly quieter and more environmentally friendly aircraft [44] [37]. Following the success of the initial Clean Sky Programme, its successor, Clean Sky 2, was launched in 2014 as part of the Commission's Horizon 2020 Research and Innovation Programme. Clean Sky 2 aims to be the main contributor to the Commission's Flightpath 2050 goals set by Advisory Council for Aeronautics Research in Europe (ACARE), which are more ambitious than those of the initial Clean Sky Programme. These goals are:

- a 75% reduction in carbon dioxide (CO2) emissions;
- a 90 % reduction in mono-nitrogen oxides (NOX);
- a noise reduction of flying aircraft of 65 %;
- mitigate the environmental impact of the life-cycle of aircraft and related products by designing and manufacturing aircraft to be recyclable.

Figure 1.4: Clean Sky 2 Logo

Among all the innovative technologies promoted, there are the study of new aircraft configurations and capabilities and advances in aerodynamic efficiency. For this reason, the EU funds two high-speed rotorcraft: the Airbus RACER compound helicopter [17] and the Leonardo Next-Generation Civil Tiltrotor (NGCTR)[18]. The two projects, despite having the same goal of creating new VTOL aircraft, are completely different. The Airbus Racer projects is based on the Airbus Eurocopter X^3 and it concerns the study of a "fast helicopter". It is, indeed, an helicopter with the addition of two Prandtl-like box wings with two propellers alongside the main rotor. This configuration should provide thrust and lift during the cruising phase, reducing the main rotor angular velocity and, consequently, the noise produced while increasing the speed.

On the other hand, the project developed by Leonardo aims to create an aircraft that is capable of taking off and landing like an helicopter, i.e. a new generation of tiltrotor aircraft. This idea comes from the long-time experience in tiltrotor that Leonardo acquired with the development of the AW609. There are many advantages in the tiltrotor concept: it combines the speed and long-range capability of an aircraft without the necessity of airport infrastructures. According to the required specifications, it has a MTOW of 11 tons, it can carry up to 25 passengers with a cruising speed of 300 knots, a ceiling of 25000 feet and a 500 nautical miles range, numbers that, all together, are unattainable for normal civil helicopters. Moreover,

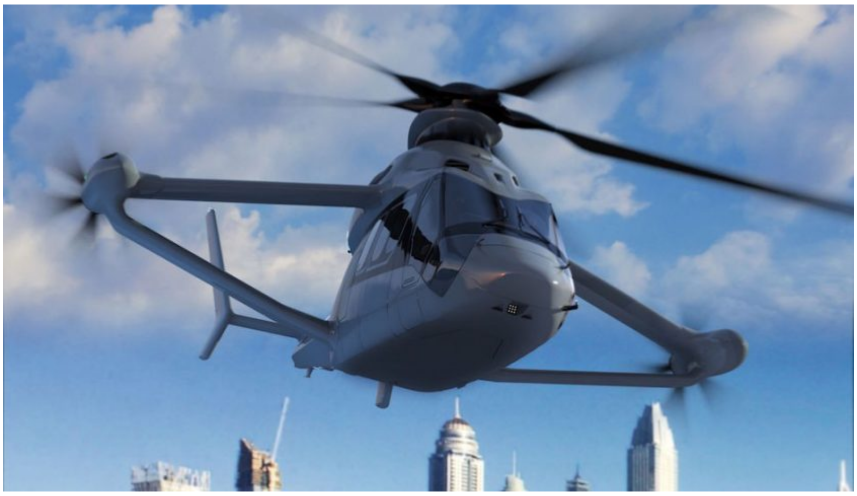


Figure 1.5: Airbus RACER compound helicopter

beyond civil air transport, tiltrotor aircraft are ideal for a range of missions that serve specialised societal needs, such as medical evacuation, search and rescue, and other utility roles where airport infrastructure is limited or unavailable, like for offshore transports. The Next Generation Civil Tiltrotor project aims to introduce and develop new technologies for the design, production and definition of a new architecture of the aircraft and of its systems to increase the performances and the operational capabilities with respect to traditional helicopters or VTOL aircraft. The ambitious environmental goals for the NextGenCTR, which are relative to the Leonardo AW139 helicopter, aim to cut CO_2 emission by half, to lower NO_x by 14%, and to mitigate noise by 30%.

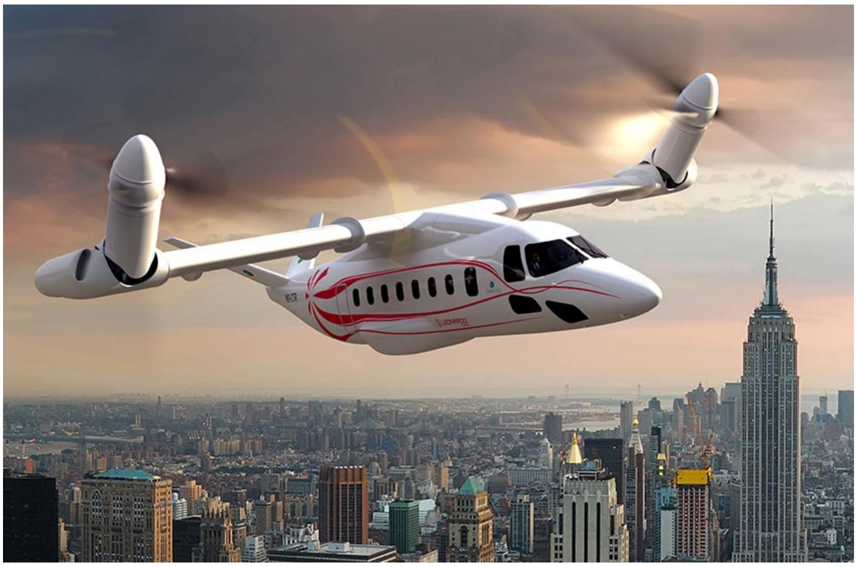


Figure 1.6: Next-Generation Civil TiltRotor rendering. Note the new idea of tilting system with respect to the one of the AW609

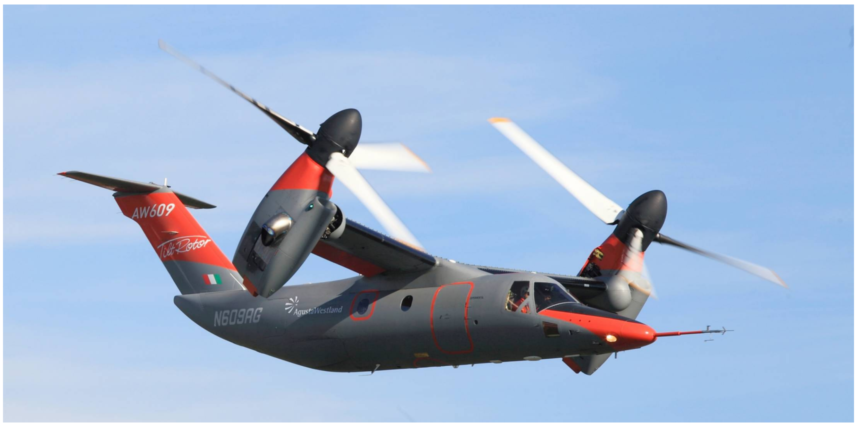


Figure 1.7: AW609 during flight test. Note the tilting system that was designed for rotating all the engine bay

The continuous innovation led to a completely new design of the nacelle and of the tilting system, along with the introduction of a new V-shaped tail. Compared with the AW609, in the NGCTR it is no longer the whole nacelle that tilts but it is only the propeller section: this solutions ensure a lighter tilting system and avoid the tilting of the engine, that remains horizontal.
Together with these new solutions, according with the Clean Sky promotion in the search and develop of new and more efficient technologies, the NGCTR project aims to investigate new design and production systems of the components. It is precisely in this area that we set the study for a procedure for the topology optimization: this tool not only can provide new features, but it can create design-based optimized geometry for a high variety of components in a faster and more efficient way with respect to the traditional "manual" optimization.

1.2 Objective of the Thesis

The goal of this thesis is to investigate how topology optimization works for a non-trivial fluid dynamical problem and to develop a procedure that is suitable for a huge variety of components that have predominantly a fluid dynamic function. Topology optimization is a well-developed tool for structural components and it is verified that it can provide optimal design solutions, but it is not so deepened in the fluid dynamic environment. The procedure here proposed is based on the analysis of some simple test cases that helped in the analysis of the different problematic introduced in the solution from the Brinkman Penalization model. The procedure developed was then tested along with the study of the optimization of the exhaust mixer (an example in fig. 1.8) of the NGCTR engine, the General Electric CT7-2E1. To test the robustness of the procedure two different optimization with two different goals were performed: the first had the primal purpose of increasing the Jet Pump effect that it generates inside the engine bay, to increase and maximize the air flow circulating in the bay and improving thus its cooling, while keeping a maximum pressure on the turbine exhaust section. The second optimization had the purpose of reducing the pressure losses of the flow exiting the turbine stages, keeping as much as possible the same amount of mass flow in the engine bay. The different results were then analyzed to understand not only the differences in the topology but also how these reflect on their performances.

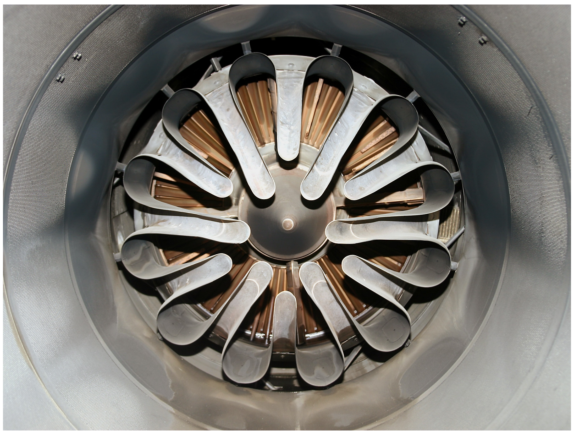


Figure 1.8: example of an exhaust mixer for aeronautical purpose

The analysis here proposed were performed with Siemens STAR-CCM+, which in its newest release (v. 2020.3, available from November 2020) introduced the Adjoint Topology Optimization model (cap. 2.3.3.2).

1.3 Structure of the Thesis

After a brief explanation of the theory behind the topology optimization and its state of the art, a focus on the physical models and how this optimization is included in the work-frame of STAR-CCM+ is presented. Then, some simple test cases are analyzed to understand how the optimization is performed and to identify the main passages required for the correct setup of this type of analysis. Then the application on the exhaust mixer is presented, with its setup, analysis and main results. In the last chapter some conclusions are reported, together with some considerations on the future of this tool for fluid-dynamic problems.

Here is presented a brief description of each chapter and its content:

Chapter 2: in this chapter the problem of topology optimization is proposed, focusing on the differences with shape optimization. There will be an introduction to the theory behind topology optimization, at first concerning the basic concept of structural optimization (which is the area in which TO firstly developed) and later with a deeper focus on fluid-dynamic topology optimization with the adjoint solver as it is implemented in STAR-CCM+;

Chapter 3: in this chapter the STAR-CCM+ environment is presented, together with the general setup of a topology optimization problem. Here there will be a brief discussion on the physical models necessary for the adjoint solver, with a focus on the turbulence models, a description of the mesh parameters and, lastly, an in-depth analysis of the definition of the objective function and the constraints for the optimization.

Chapter 4: in this chapter the topology optimization model is applied on simple but meaningful problems: starting from a quasi-2D problem, the model is studied in all its details, focusing on the problems and weaknesses of the current formulation.

The model is therefore improved with a new sensitivity normalization procedure and with an implementation of a turbulence correction. The new model is then compared with the base one in a real 3D problem, to investigate how the new features behave in more complex geometries. These cases represent the application of topology optimization in its simplest way, the one for which until today TO was used in fluid-dynamic environment.

Chapter 5: in this chapter the new topology optimization procedure is applied on the exhaust mixer of the GE CT7-2E1. After a brief description of the problem and of the geometry, the optimization analysis are presented. An in-depth analysis is reserved for the validation of the new geometry and the analysis of the difference in the flow behaviour between the basic and the two optimized configurations together with the verification of the satisfaction of the constraints required by the type certification of the engine.

Chapter 6: in this chapter are reported the conclusions based on the acquired knowledge of the optimizations performed. Moreover, some considerations on the next steps in the topology optimization and some possible future development are proposed.

2
Topology Optimization

Topology optimization originates in the structural and solid mechanics field as an alternative to shape and sizing optimizations. The basic concepts of this technique were posed by Bendsøe and Kikuchi [5] in the late 1980s with the introduction of a homogenization method. Topology optimization poses itself as a material distribution technique that, in its structural formulation, aims to find the optimal distribution that can improve the performance of a structural component under specific constraints; in other words, topology optimization aims to give an answer to the question “where should solid material be placed?” or, alternatively, “where should the holes be inserted?" [25].

Even thought topology optimization (hereafter referred to also as TO) was developed a few years after shape optimization (hereafter referred to also as SO), in the flowchart of the development of a structure it ideally come first since it does not need an initial structure defined a priori, but moves in a wide design space; the output of TO is indeed a specific material distribution, that *may* be furthermore analyzed with the SO tools if necessary. Today, topology optimization for solid mechanics is a well-developed and widely available technology and can be applied to a huge variety of problems.

The basic idea of the original methodology is applicable to all those physics in which the governing equations can be described by a set of partial differential equations (PDEs), which therefore enlarges its applications to a high variety of different physics such as acoustics, electromagnetism, heat conduction, fluid flow, etc. [13]. It is in the latter field that the work of this thesis is developed: when applied to a fluid field, the above question should be changed in “where should the fluid flow?”. Although the basic concepts are also valid for this field, dealing with the Navier-Stokes equations introduces some advanced considerations, especially on turbulence modeling and in near-wall treatments.

Since the first mathematical formulation for topology optimization was developed in the structural fields, it is based on a structural language and such will be used here for its description. In the following chapters the fluid-dynamic description and modeling is proposed with an in-depth analysis of the model used for this analysis:

the discrete adjoint of the compressible Navier-Stokes equations coupled with the level-set method.

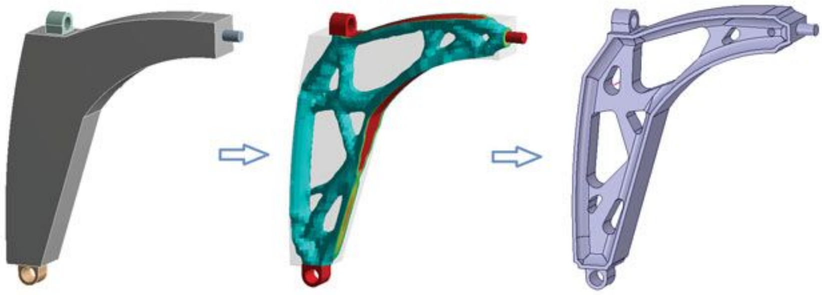


Figure 2.1: Example of topology optimization results

The general workflow of a Topology optimization analysis, illustrated in figure 2.2, consists of three fundamental steps: the Finite Element Analysis, the Sensitivity analysis and the modification of the material distribution. FEA is the starting point, since it provides the strain and stress fields on which the sensitivities are computed; it is clear that a correct and compliant FEA is necessary for the optimization to succeed. The real backbone of the TO is however the sensitivity analysis, since, based on its field, a cell is characterized as favorable or counterproductive for the chosen objective function. As previously sad, topology optimization can handle several constraints, which also need their own sensitivity analysis: the latter is combined with the one computed for the base function, which is modified accordingly (i.e. the sensitivity is filtered with the constraints one). The filtered sensitivity field is therefore used to modify the material distribution; after such a modification, a new FEA is needed to analyze the improvements and to compute the new stresses and strains, and the loop is repeated until the performance is satisfied.
These analysis can be performed in different ways, depending on the method chosen; in the following sections different methods are introduced and illustrated, with a particular focus on the one used for the topology optimization performed in this thesis.

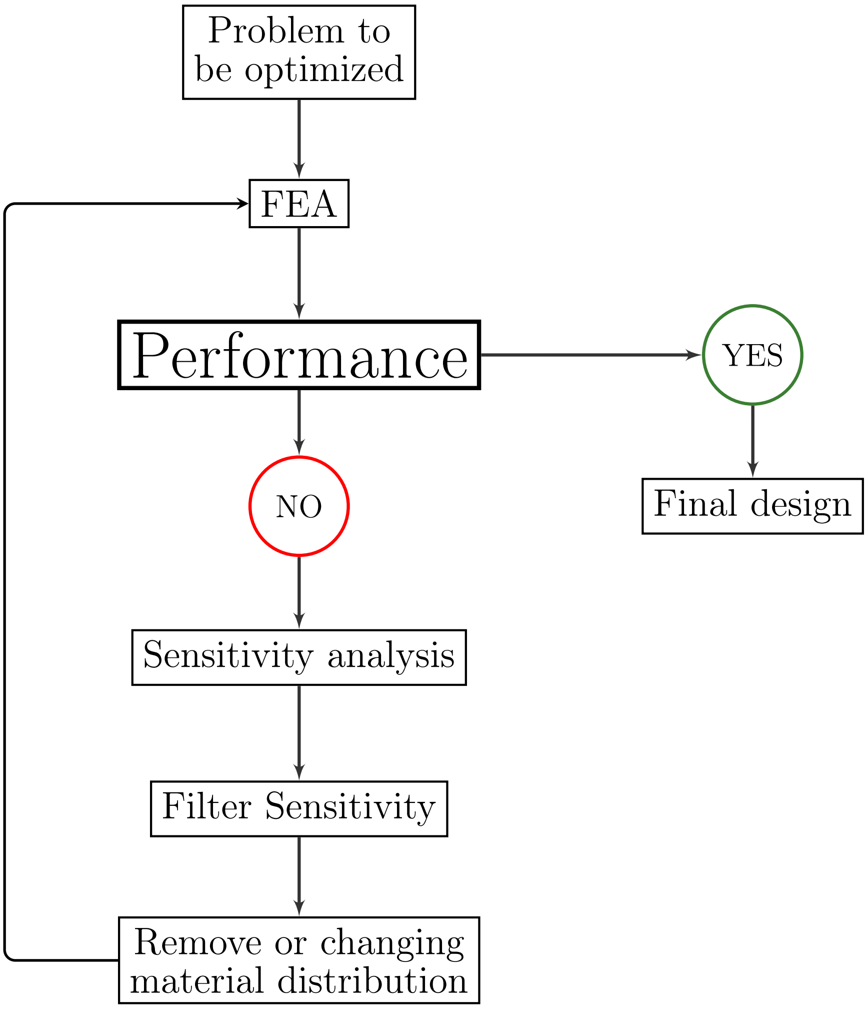


Figure 2.2: Topology optimization typical workflow

2.1 State of the art

Although the foundations of topological optimisation were only laid in the late 1980s, it has undergone a rapid development because of its versatility and the improvement it can introduce into the design of a structural component. Nowadays, structural topology optimization is capable of solving a huge variety of problems, with multiple constraints, multi-objects, multi-materials or multi-phase physics, even under manufacturing or performance constraints. Topology optimization usually generates "natural" and uncommon shapes: for the common and traditional CAD modeling and manufacturing procedure these represented a problem in the early stages of its development, but with the introduction and the improvement of the Additive Manufacturing technologies its implementations have increased rapidly. To fully understand the capabilities that nowadays TO have, and what it is capable of, the easiest way is to analyze the huge variety of constraints that it can handle; some of theme are here presented here, but more can be found, for example, in [3] and [35]:

- **Minimum feature size**: ensure a minimal thickness for each geometric feature.

- **Retained surfaces**: ensure a surface not to be modified by the optimization.
- **Draw direction constraints**: prevent the formation of holes and cavities along a given direction.
- **Cyclic symmetry**: to impose cyclic symmetry for circular geometries.

It is evident that the geometries that can be developed with the topology optimization coupled with Additive Manufacturing have, ideally, no limits, and the ease with which are obtained depicts this analysis as the new starting point for the structural design.
The application of topology optimization for fluid-dynamic aims, in its complete formulation, to find the optimal geometry that can account both for aerodynamic and structural parameters. In the past years. the search for model that can account for fluid-structure interaction, i.e that can couple the topology optimization problem with the aeroelastic one, has led to some simplified models: initially, linear flow theories and simplified structural models were applied for predicting the aerodynamic loads in the aeroelastic design procedure [15], but in the last years were developed some more complex models that allow for detailed linear and non-linear structural finite element models and non-linear flow theories, such as Euler and Navier–Stokes flows [20] [14] [32]. Despite the great results that are achievable, these model are based on the topology optimization as far as the structural model is concerned, while for the aerodynamic problem a shape optimization is performed and, therefore, they are limited to well-defined geometry such as wings; in the general context of fluid-dynamic topology optimization it is not yet fully developed. In the following analysis, we will focus on the fluid-dynamic problem only, without considering the structural one; The major applications of TO for fully fluid-dynamic problems were, by now, internal flows and piping systems, with the goal of reducing the pressure losses, increasing the mass flow or optimizing the velocity uniformity (fig. 2.3).
The goal of this thesis is to analyze the different aspects and problematic of the Level-set approach for TO (explained in chapter 2.3.2) and to find a possible procedure for its correct application also a more complex aerodynamic problems, such as the topology of an aeronautical engine exhaust.

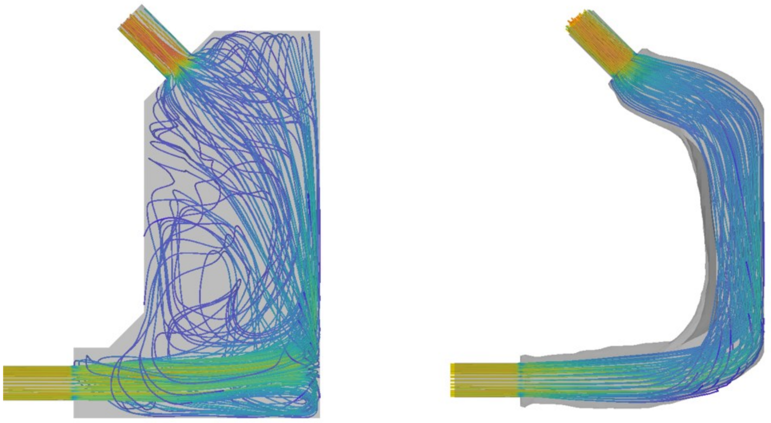


Figure 2.3: Example of topology optimization for a fluid domain

2.2 Introduction to Shape Optimization

Quite often topology optimization is confused or mistaken for shape optimization. Although they are closely related, SO aims to maximize performance (or reduce cost) by acting on design parameters associated with a CAD model, for examples the thickness of a component or the radius of a cylinder, i.e. by changing only the *shape* of the design, without introducing new features. On the other hand, TO can work in a generic design space and does not always need for an initial geometry to start from; in other words, it has more freedom in finding the optimal solution. Ideally, TO and SO can be linked and coupled for an in-depth optimization: the final geometry obtained with topology optimization is usually rough and "natural", therefore it can be used as an indication of the best geometry; this can then be subjected to shape optimisation for some minor modifications and improvements.
As mentioned before, SO and TO were firstly developed for structural optimizations, so the basic mathematical concepts relate to the structural languages. In this brief introduction to SO and TO, this particular language will be used, although the following analyses refer to a fluid-dynamic environment.

Let us consider the objective of reducing the volume of a generic component while imposing a constraint on its stiffness. Since the research of a optimum usually refers to the minimization of an objective function, we will consider the inverse of the stiffness, namely the compliance, that is defined as:

$$J = \mathbf{f}^T \cdot \mathbf{u} \tag{2.1}$$

where $\mathbf{f}$ and $\mathbf{u}$ are, respectively, the finite element discretized force and displacement vectors.
It is now possible to pose two meaningful shape optimization problems:

- Starting from an initial design D_0 with volume V_0 and compliance J_0, find the shape parameters that minimize the compliance with a constraint on the volume:

  $$\begin{aligned} &\underset{\mathbf{s}}{Minimize} \;\; J \\ &V \leq V_0 \\ &\mathbf{s}_{min} < \mathbf{s} < \mathbf{s}_{max} \end{aligned} \tag{2.2}$$

  where $\mathbf{s}$ is the vector of the shape parameters.

- Starting from an initial design D_0 with volume V_0 and compliance J_0, find the shape parameters that minimize the volume with a constraint on the compliance:

  $$\begin{aligned} &\underset{\mathbf{s}}{Minimize} \;\; V \\ &J \leq J_0 \\ &\mathbf{s}_{min} < \mathbf{s} < \mathbf{s}_{max} \end{aligned} \tag{2.3}$$

Although these two formulations may seem similar, there is a huge difference between them: the first one aims at minimizing the compliance with a constraint on the volume, i.e. without adding material, while the latter aims to the opposite, i.e. removing as much material as possible while keeping the compliance below its initial value. It is obvious, anyway, that if a random search of the optimum is performed, both these two optimization will lie on the Pareto front.

Today, there are different methods for solving equations 2.2 and 2.3, but the most popular ones are the first order optimization methods and, among them, the non-linear conjugate gradients method. Without going into too much detail about this method, the direction of searching for the minimum of the objective function is based on the gradient of the same. The minimum is found when the gradients are equal to zero.

This method is quite powerful because it can simply handle constraints by the use of the *Lagrangian* $\mathscr{L}$.

Considering a function $f(\mathbf{D})$, where $\mathbf{D}$ are the design variables, subject to equality constraints $\mathbf{g(D)} = \mathbf{0}$ and inequality ones $\mathbf{h(D)} \leq \mathbf{0}$, the corresponding *Lagrangian* will be:

$$\mathscr{L} = f(\mathbf{D}) + \lambda \cdot \mathbf{g(D)} + \mu \cdot \mathbf{h(D)} \tag{2.4}$$

where λ and μ are the Lagrange multipliers vectors.

As stated before, it is required the computation of the gradient of the *Lagrangian*, that will be as follow:

$$\nabla_{\mathbf{D}}\mathscr{L} = \nabla f(\mathbf{D}) + \lambda \cdot \nabla\mathbf{g(D)} + \mu \cdot \nabla\mathbf{h(D)} \tag{2.5}$$

The concept of the *Lagrangian* will be used later in the text.

It is clear that to solve a TO problem it is necessary to compute the gradients, i.e. perform a *sensitivity analysis*. While the sensitivity of the bound constraints and the volume one (or objective) are easy to handle, the real problem is the gradient of the compliance. There are two different approaches: Direct Finite Difference and Indirect Finite Difference.

- **Direct Finite Difference**: computing the sensitivity of the compliance via a 1-st order finite differences (ex. forward finite difference):

$$\frac{\partial J}{\partial D_i} = \frac{J(\mathbf{D} + \Delta\mathbf{D}_j) - J(\mathbf{D})}{\Delta\mathbf{D}_j} \tag{2.6}$$

  where

$$\Delta\mathbf{D}_j = \begin{cases} \Delta D_i, & \text{if } i = j \\ 0, & \text{else} \end{cases} \tag{2.7}$$

  Unfortunately, a direct finite difference method suffers from numerical problems within a FEA context, and would probably lead to the failure of the optimization process.

- **Indirect Finite Difference**: let's consider the definition of compliance:

$$\mathbf{Ku} = \mathbf{f} \tag{2.8}$$

$$J = \mathbf{f}^T\mathbf{u} \tag{2.9}$$

taking the derivative of both equations with respect to shape parameters, remembering that the vector of the forces does not depend on them, we obtain:

$$\mathbf{K}\mathbf{u}' = -\mathbf{K}'\mathbf{u} \tag{2.10}$$

$$J' = \mathbf{f}^T\mathbf{u}' \tag{2.11}$$

where:

$$(.)' \equiv \frac{\partial}{\partial D_i} \tag{2.12}$$

So we obtained the displacement sensitivity ($\mathbf{u}'$) and the stiffness matrix sensitivity ($\mathbf{K}'$). The advantage is that the latter is computationally efficient to calculate using mesh morphing methods. Once we know the stiffness matrix sensitivity, is possible to compute via eq 2.10 the displacement sensitivity, which via eq 2.11 let us finally compute the compliance sensitivity.

Based on the gradient, the algorithms chooses where to move in the domain. For example, a classical gradient-descending method computes the new design variables at the iteration $k+1$ as follow:

$$\mathbf{D}^{k+1} = \mathbf{D}^k - \alpha J'|_{\mathbf{D}=\mathbf{D}^k} \tag{2.13}$$

where α is an algorithm parameter, usually constant.

2.3 Topology optimization methods

The theory of topology optimization generalizes the one of shape optimization since it allows new features to be introduced into the domain. In this chapter, after the statement of the mathematical TO problem, the two most common methods will be briefly discussed, while the one used for the here performed analyses is discussed in depth in chapter 2.3.3.

The generic TO problem can be formally stated as:

$$\begin{aligned} &\underset{\Omega\subset D}{Minimize}\;\; J \\ &\text{Subject to} \\ &V \leq V_0 \\ &\mathbf{K}\mathbf{u} = \mathbf{f} \end{aligned} \tag{2.14}$$

where Ω is the topology to be computed and D is the topological domain. It is important to compare the above formulation with the one of SO (eq. 2.2 or eq. 2.3) to clearly understand the differences: in the topology optimization formulation no design parameters are considered, but the request is to determine the topological distribution of the material in the domain, i.e. where to insert or remove material. Nowadays, the two most common methods to parameterize the topology are:

1. density-based methods;
2. Level-set methods.

The first and fundamental difference between them is in the computation of the sensitivities, as it is explained in the next sections.

2.3.1 Density-Based Method (SIMP)

The density-based approach is the most popular TO method, since it has a solid mathematical foundation, it can handle various constraints and objective functions and it is quite easy to implement in a FE environment. This method is also known as Solid Isotropic Material with Penalization (SIMP) and was originally proposed by *Bendsoe and Kikuchi* (1988) and *Rozvany and Zhou* (1992). According to Bendsoe (1989): *"Shape optimization in a general setting requires the determination of the optimal spatial material distribution for given loads and boundary conditions. Every point in space is thus a material point or a void and the optimization problem is a discrete variable one."* (note: even though Bendsoe refers to "shape optimization", this method was developed for TO, but at the time no distinctions where still made). The idea behind the finite element based SIMP method is that it is possible to associate a pseudo-density variable ρ to every single element, called isotropic solid micro-structure, that allows to parameterize the topology. In the domain, each single cell is characterised as either full of material ($\rho = 1$) or empty ($\rho = 0$), which will lead to a discrete material distribution. The introduction of a continuous density distribution function avoids the binary nature of the problem: for each element, ρ_e can vary between a minimum value ρ_{min} (different from zero to avoid numerical problems) that identifies the empty domain, and 1, thus allowing the assignment of intermediate densities for the cells.

The TO problem therefore can be reformulated as:

$$\begin{aligned} &\underset{\rho_e}{Minimize} \;\; J \\ &\sum_{e=1}^{E} \rho_e \nu_e - V_0 = 0 \\ &\mathbf{Ku} = \mathbf{f} \\ &\rho_{min} \leq \rho_e \leq 1 \end{aligned} \tag{2.15}$$

where ν_e is the finite element volume and E is the total number of finite elements. To find the optimal values of ρ_e it is necessary to carry out a sensitivity analysis. Since the relative density of the material can now vary continuously, the Young's modulus of the material in each cell must also does the same. A simple but powerful relationship between them is a power law function (*Sigmund 2001*; *Zhou et al. 2001*):

$$E(\rho_e) = (\rho_e)^p E_0 \tag{2.16}$$

Where p is the penalty factor, which diminishes the contribution of elements with intermediate densities to the total stiffness. From numerical experiments, in a 3D problem $p = 3$ was found to be a suitable value.

In order to compute the impact of the variation of material densities on the stiffness (or compliance) matrix, it is necessary to perform a sensitivity analysis. Recalling the general definition of the *Lagrangian* (eq. 2.4), with the SIMP method it can be reformulated as:

$$\mathscr{L}(\mathbf{u}, \rho) = \mathbf{u}^T\mathbf{Ku} + \mu_\alpha \left[\sum_{e=1}^{E} \rho_e \nu_e - V_0 = 0 \right] + \mu_f^T(\mathbf{Ku} - \mathbf{f}) \tag{2.17}$$

which, considering the gradients with respect to **u** and $\boldsymbol{\rho}$ leads to the sensitivity expression:

$$\mu_\alpha \nu_e = p\rho_e^{p-1}\mathbf{u}^T\mathbf{K}_0(e)\mathbf{u}_e \tag{2.18}$$

The computation of the Lagrange multiplier μ_α can be performed, for example, via the bi-section method [3].
We can therefore summarize the SIMP algorithm in the following steps :

1. Initialize the densities to the specific volume fraction $\rho_e = \alpha$;
2. With the specified densities, the static problem $\mathbf{Ku} = \mathbf{f}$ via FE method is solved;
3. The sensitivity analysis is performed and, via the bi-section method, the densities are updated;
4. If convergence is not reached, return to step 2 with the new densities. If convergence is reached, the topology obtained is the required one.

2.3.2 Level-set methods

The other most common popular TO approaches are the Level-set Methods (LSMs), which define the interfaces between material phases implicitly using isolines of a Level-set Function (LSF) [23] [2]. Unlike density-based ones, this kind of representation can create crisp interfaces between material phases, which can lead to greater accuracy in correctly capturing the mechanical response near the boundaries, avoiding ambiguities of intermediate material phases. Although early versions of LSMs were not able to insert holes in the domain, this problem has now been solved with the introduction of the concept of *topological derivatives.*
The LSF is used to describe the boundaries of a structure occupying the domain according to the following definition:

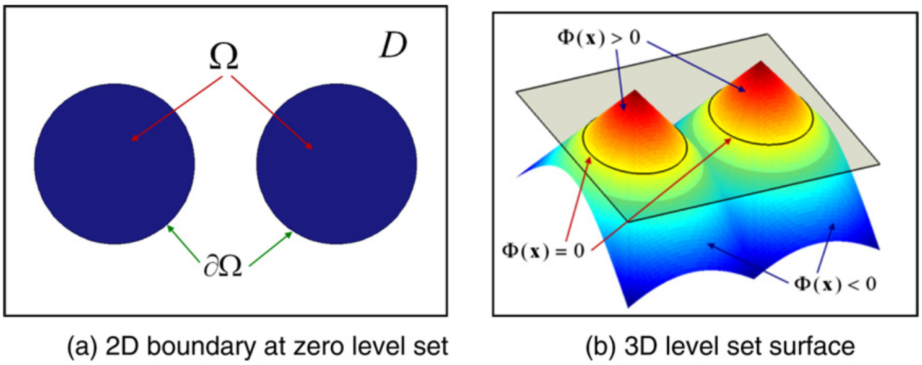
$$\phi(x,y) = \begin{cases} < 0, & \text{if } (x,y) \in \Omega \\ = 0, & \text{if}(x,y) \in \partial\Omega \\ > 0, & \text{else} \end{cases} \tag{2.19}$$

where (x,y) is any point in the domain. An example of a LSF is shown in figure 2.4.

The shape and the topology of the domain change during the optimization according to the Level set function ϕ, which evolves according to the following Hamilton-Jacobi transport equation:

$$\frac{\partial\phi}{\partial\tau} + \mathbf{V}|\nabla\phi| = 0 \tag{2.20}$$

where τ is a fictitious time parameter $\tau > 0$ which corresponds to descent stepping and V is the interface velocity between two phases (see chapter 2.3.3.2).
Topological derivatives concern a further discussion, since they lead to the possibility of generating holes in the domain. This possibility is quite important and fundamental in a 2D domain, while in 3D this possibility must be used with care, especially if the domain is a fluid one, since it can generate floating portions of solid



(a) 2D boundary at zero level set (b) 3D level set surface

Figure 2.4: Example of a Level Set function [48]

inside the domain, leading to an unfeasible design.
For simplicity, let's consider a full solid domain and insert a hypothetical hole of radius r in a point p. The topological derivative is defined as follow:

$$\mathscr{T}(p) \equiv \lim_{r \to 0} \frac{\phi(r;p) - \phi}{\pi r^2} \tag{2.21}$$

where ϕ is a generic scalar quantity of interest. Let's now introduce an infinitesimal increment in the radius throughout a parameter τ. The shape sensitivity of ϕ with respect to τ is the following:

$$\chi(r) \equiv \frac{d\phi(r)}{d\tau}|_{\tau=0} \tag{2.22}$$

We can now relate topological sensitivity with shape sensitivity as follows:

$$\mathscr{T}(p) = \lim_{r \to 0} \frac{\chi(r)}{2\pi r} \tag{2.23}$$

It is possible to derive a closed-form expression for the topological derivative, using adjoint methods and evaluating stresses and strains in the point p before hole is inserted. Level set methods are the one used in STAR-CCM+, so there will be a further discussion later (see chapter 2.3.3.2).
So far, all the models presented where referred to the early theory of structural topology optimization. From now on, we will focus on a fluid dynamic topology optimization performed by means of the adjoint model, which is based on the Level-set method discussed above. In the following sections the discrete adjoint of the compressible Navier-Stokes equations coupled with the level-set method is presented and discussed in all its components, since it is the chosen model for the performed analyses.

2.3.3 Adjoint Topology Optimization Model

As mentioned before, the fundamental step in TO is the computation of the sensitivity of the objective function with respect to the design variables. In the gradient-based algorithms, as shown in eq. 2.6, the direct finite difference method not only can lead to instability but, for a fluid dynamic problem, it can also be computationally infeasible, due to the high number of design variables. Considering

L(**D**) the objective function, each evaluation of **L**(**D**) needs the state system to be solved as first, which requires N+1 solver calls (where N is the number of design variables): one **L**(**D**+Δ**D**$_j$) for each Δ**D**$_j$ in the direction of each design variable plus one **L**(**D**).
On the other hand, the adjoint approach, which consists of computing the gradient of the cost function using the *adjoint state*, can compute the whole sensitivity with just one solution of the *primal solver* and one of the *adjoint solver*, requiring in this way just two calls, *regardless of the number of the design parameters*.
Since this type of optimization requires the *adjoint model*, here the fundamental theory behind the adjoint sensitivity is briefly discussed.

2.3.3.1 Discrete Adjoint model

Let **R(Q,X)** be the vector of the residuals of the discrete Navier-Stokes problem, where **X** is the mesh domain, and the steady-state solution **R(Q,X)**=0. The sensitivity of the cost function is therefore given by the chain rule:

$$\frac{\partial \mathbf{L}}{\partial \mathbf{D}} = \left[\frac{\partial \mathbf{L}}{\partial \mathbf{X}} + \frac{\partial \mathbf{L}}{\partial \mathbf{Q}}\frac{\partial \mathbf{Q}}{\partial \mathbf{X}}\right]\frac{d\mathbf{X}}{d\mathbf{D}} \tag{2.24}$$

Focusing on the rightmost term, considering M points on the mesh domain, it turn out to be a (N x M) matrix. For a large number of variables, like in a generic fluid dynamic problem, the computation of this term can be too expensive. To avoid this problem, it is possible to decompose the derivatives:

$$\frac{\partial \mathbf{L}}{\partial \mathbf{D}_i} = \left[\frac{\partial \mathbf{L}}{\partial \mathbf{X}} + \frac{\partial \mathbf{L}}{\partial \mathbf{Q}}\frac{\partial \mathbf{Q}}{\partial \mathbf{X}}\right]\frac{d\mathbf{X}}{d\mathbf{D}_i} \tag{2.25}$$

for $i = 1, 2, ..., N$ columns.
It is worth noticing that, while the design variables may be in a large number, the outputs required are often limited (e.g. pressure, mass flow, forces, etc..). The computational cost can therefore be improved taking the transpose of the derivatives, which gives a system consisting of only a few equations:

$$\frac{\partial L_j}{\partial \mathbf{D}}^T = \frac{d\mathbf{X}}{d\mathbf{D}}^T \left[\frac{\partial L_j}{\partial \mathbf{X}}^T + \frac{\partial \mathbf{Q}}{\partial \mathbf{X}}^T \frac{\partial L_j}{\partial \mathbf{Q}}^T\right] \tag{2.26}$$

Equation 2.26 represent also how sensitivity is performed in STAR-CCM+.
By analysing the terms that make up the equation described above, it is possible to identify three different sensitivities:

- $\frac{\partial L_j}{\partial \mathbf{Q}}^T$: this term is the sensitivity of the cost function with respect to the solution **Q**
- $\frac{\partial L_j}{\partial \mathbf{X}}^T$: this term represents the direct sensitivity of the cost function with respect to the mesh
- $\frac{\partial \mathbf{Q}}{\partial \mathbf{X}}^T$: this term is the sensitivity of the solution with resepct to the mesh

In order to solve equation 2.26, it is necessary to compute the sensitivity with respect to the solution, which also gives the one for the mesh. Subsequently it is possible to evaluate the term in bracket, that represents the sensitivity of the report objectives with respect to the mesh, i.e. $\frac{dL_j}{d\mathbf{X}}^T$. Lastly, perform the product between the sensitivity of the mesh with respect to the design parameter $\frac{d\mathbf{X}}{d\mathbf{D}}^T$ and the result of the previous step. It is so obtained the sensitivity of the cost function with respect to the design parameters, i.e. the wanted sensitivity.
The steps required for evaluating these sensitivities can be briefly summarized as [41]:

- **Sensitivity with respect to the solution**. First, the sensitivity of the cost function w.r.t the solution and the direct sensitivity w.r.t the mesh are computed. These sensitivities are evaluated once the physical solver has converged or reached a stop criterion, so these operations are only performed once.

- **Sensitivity with respect to the mesh**. The core of this analysis is the computation of the sensitivity with respect to the mesh, i.e. the term in brackets in eq. 2.26.
  To calculate the required sensitivity, the adjoint solution for the sensitivity of the cost function with respect to the residuals of the governing equations, i.e. $\frac{dL}{d\mathbf{R}}^T$, is performed.
  Suppose there are **N** iterative solvers, each for a subset of variables, such that particular governing equations are satisfied. Let us consider $\mathbf{Q}_i$ the set of variables and $\mathbf{R}_i$ the residual forms of the governing equations associated to the i^{th} solver. Since a coupled flow model is required for the adjoint solver, the solution of the flow field can be written as a system of coupled non-linear equations:

  $$\mathbf{R}_i(\mathbf{Q}_1, \mathbf{Q}_2, ..\mathbf{Q}_N, \mathbf{X}) = 0 \quad for\ i = 1 : N \tag{2.27}$$

  Since the adjoint can be performed only in a time-steady solution, each solver constructs its residual based on the previous values and advances thanks to a local preconditioner $[P_i]$, which is an approximate linearization of the Navier-Stokes equations with respect to the variables:

  $$\begin{aligned} \mathbf{R}_i^k &= \mathbf{R}_i(\mathbf{Q}_1^{k+1}, \mathbf{Q}_2^{k+1}, ..., \mathbf{Q}_i^k, ..., \mathbf{Q}_N^k) \\ \mathbf{Q}_i^{k+1} &= \mathbf{Q}_i^k - [P_i]^-1\mathbf{R}_i^k \end{aligned}$$

  To obtain the adjoint solution, it is necessary to perform the derivatives with respect to **X** for every i^{th} solver, which results in a fully coupled linear system:

  $$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right]\frac{\partial \mathbf{Q}}{\partial \mathbf{X}} = -\frac{\partial \mathbf{R}}{\partial \mathbf{X}} \tag{2.28}$$

  If we invert for the sensitivity of the variables, we obtain a new expression for the sensitivity of the cost function with respect to the mesh:

  $$\frac{dL}{d\mathbf{X}}^T = \frac{\partial L}{\partial \mathbf{X}}^T - \frac{\partial \mathbf{R}}{\partial \mathbf{X}}^T\left[\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right]^{-T}\frac{\partial L}{\partial \mathbf{Q}}^T \tag{2.29}$$

From this equation we can define the *adjoint solution variable* $\Lambda_{\mathbf{Q}}$:

$$\Lambda_{\mathbf{Q}} = \left[\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right]^{-T} \frac{\partial L}{\partial \mathbf{Q}}^{T} \tag{2.30}$$

To avoid the computation and assembly of a large and complex matrix, the adjoint solution process uses algorithms that only require products between matrices and vectors, namely Restarted GMRES and Defect correction [24]. The defect correction can be written as:

$$[\mathbf{P}]\delta\Lambda_{\mathbf{Q}} = \frac{\partial L}{\partial \mathbf{Q}} - \left[\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right]\Lambda_{\mathbf{Q}} \tag{2.31}$$

The advantage in solving the adjoint with this algorithm is that the system is solved in a Jacobi-like manner, saving lot of memory during the computation. The final forms of the equations for the computation of the residual are the following:

$$\begin{gathered} \mathbf{R}^{\mathbf{k}}_{\Lambda_{\mathbf{Q},\mathbf{i}}} = \frac{\partial \mathbf{R}_1}{\partial \mathbf{Q}_i}\Lambda^{\mathbf{k}}_{\mathbf{Q},\mathbf{1}} + \frac{\partial \mathbf{R}_2}{\partial \mathbf{Q}_i}\Lambda^{\mathbf{k}}_{\mathbf{Q},\mathbf{2}} + \cdots + \frac{\partial \mathbf{R}_i}{\partial \mathbf{Q}_i}\Lambda^{\mathbf{k}}_{\mathbf{Q},\mathbf{i}} + \cdots + \frac{\partial \mathbf{R}_N}{\partial \mathbf{Q}_i}\Lambda^{\mathbf{k}}_{\mathbf{Q},\mathbf{N}} \\ \Lambda^{\mathbf{k+1}}_{\mathbf{Q},\mathbf{i}} = \Lambda^{\mathbf{k}}_{\mathbf{Q},\mathbf{i}} - [\mathbf{P}_i]^{-T}\mathbf{R}^{\mathbf{k}}_{\Lambda_{\mathbf{Q},i}} \end{gathered} \tag{2.32}$$

This system of equations also include the solver for the turbulence models, which are expensive and can lead to instability. To enhance stability it is possible to eliminate the column representing the variables of that solver using the STAR-CCM+ so-called *adjoint frozen turbulence model*, which consists in considering fixed the turbulent quantities obtained in the primal solution during the adjoint iteration.

- **Sensitivity with respect to design parameters**. The last important sensitivity is the one w.r.t to the design parameters $\frac{dL_j}{d\mathbf{D}}^T$, i.e. to the mesh deformation. This term is the product between the result of the previous step ($\frac{dL_j}{d\mathbf{X}}^T$) and the sensitivity of the mesh with respect to the design parameter $\frac{d\mathbf{X}}{d\mathbf{D}}^T$. The output of this sensitivity usually is used to compute the effect that a change in the mesh would have on the solution, in order to perform an optimization via a mesh deformation; this sensitivity is therefore usually used for shape optimization problems since it requires an initial geometry from which start from, but for sake for completeness it is analyzed anyway.
The different mesh deformation techniques developed over the years can be summarized in two great categories: physical analogy based techniques and interpolation based techniques. While the former describes the fluid mesh deformation according to a physical process that can be modeled using numerical methods, the latter uses an interpolation function to transfer the prescribed displacements of the boundary points to the fluid mesh [39] [40] [28]. In STAR-CCM+, the mesh deformation algorithm that is used is the *radial basis function (RBF) morpher algorithm* [6], which falls into the category based on physical analogy, and which compute the sensitivity w.r.t. the design parameter as:

$$\frac{dL}{d\mathbf{D}}^T = \frac{d\mathbf{X}}{d\mathbf{D}}^T \frac{dL}{d\mathbf{X}}^T \tag{2.33}$$

The RBF allows to create a scalar field that interpolates a function defined on a set of source points: considering a reference position $\mathbf{x}^0$, the interpolated position $\mathbf{x}$ is given by:

$$\mathbf{x}(\mathbf{x}^0) = \boldsymbol{\alpha} + \sum_{j=1}^{N} \boldsymbol{\beta}_j \phi_j(r_j(\mathbf{x}^0)) \tag{2.34}$$

There exist two types of RBFs, the ones with compact support [45] and the ones with global support. In the latter case, the following multiquadratic biharmonics function $\phi(r)$ is used:

$$\phi_j(r_j) = \sqrt{r_j^2 + c_j^2}, \quad \xrightarrow[\forall j]{c_j=0} \quad \phi_j(r_j) = |r_j| \tag{2.35}$$

where

$$r_j(\mathbf{x}^0) = ||\mathbf{x}^0 - \mathbf{x}_j^0|| \tag{2.36}$$

Considering a discrete domain, with N reference positions, the M interpolation positions becomes:

$$\begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_M \end{bmatrix} = \begin{bmatrix} 1_1 \\ \vdots \\ 1_M \end{bmatrix} \boldsymbol{\alpha} + \begin{bmatrix} \phi_{1,1} & \dots & \phi_{1,N} \\ \vdots & \ddots & \vdots \\ \phi_{M,1} & \dots & \phi_{M,N} \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta}_1 \\ \vdots \\ \boldsymbol{\beta}_N \end{bmatrix} \tag{2.37}$$

In order to obtain the unknown coefficients, it is necessary to solve the following system:

$$\begin{bmatrix} \phi_{1,1} & \dots & \phi_{1,N} & 1_1 \\ \vdots & \ddots & \vdots & \vdots \\ \phi_{M,1} & \dots & \phi_{M,N} & 1_N \\ 1_1 & \dots & 1_N & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta}_1 \\ \vdots \\ \boldsymbol{\beta}_N \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_N \\ 0 \end{bmatrix} \tag{2.38}$$

with the additional constraint

$$\sum_{j=1}^{N} \boldsymbol{\beta}_j = 0 \tag{2.39}$$

It is now possible to derive the expression for the required sensitivity. From equation 2.38 one can obtain the derivatives of the coefficients α and β and inserting in the 2.37 from which is possible to derive $\frac{\partial \mathbf{X}}{\partial \mathbf{d}_j}$.
Since the aim is the calculation of equation 2.33, considering that the second term has been already computed, the adjoint only has to perform the product:

$$\frac{\partial L}{\partial \alpha} = \begin{bmatrix} 1_1 \\ \vdots \\ 1_M \end{bmatrix} \begin{bmatrix} \frac{\partial L}{\partial \mathbf{x}_1} \\ \vdots \\ \frac{\partial L}{\partial \mathbf{x}_M} \end{bmatrix} \tag{2.40}$$

which gives

$$\begin{bmatrix} \frac{\partial L}{\partial \boldsymbol{\beta}_1} \\ \vdots \\ \frac{\partial L}{\partial \boldsymbol{\beta}_M} \end{bmatrix} = \begin{bmatrix} \phi_{1,1} & \dots & \phi_{1,N} \\ \vdots & \ddots & \vdots \\ \phi_{M,1} & \dots & \phi_{M,N} \end{bmatrix}^T \begin{bmatrix} \frac{\partial L}{\partial \mathbf{x}_1} \\ \vdots \\ \frac{\partial L}{\partial \mathbf{x}_M} \end{bmatrix} \tag{2.41}$$

and, finallly,

$$\begin{bmatrix} \phi_{1,1} & \dots & \phi_{1,N} & 1_1 \\ \vdots & \ddots & \vdots & \vdots \\ \phi_{M,1} & \dots & \phi_{M,N} & 1_N \\ 1_1 & \dots & 1_N & 0 \end{bmatrix}^T \begin{bmatrix} \frac{\partial L}{\partial \mathbf{d}_1} \\ \vdots \\ \frac{\partial L}{\partial \mathbf{d}_M} \\ \mathbf{t} \end{bmatrix} = \begin{bmatrix} \frac{\partial L}{\partial \beta_1} \\ \vdots \\ \frac{\partial L}{\partial \beta_M} \\ \frac{\partial L}{\partial \alpha} \end{bmatrix} \tag{2.42}$$

where $\mathbf{t}$ is rejected.
A similar approach to the mesh sensitivity is the *Surface sensitivity*; although it is conceptually similar to the former, the latter allows for a quicker calculation of the sensitivities with respect to the boundary parameters as it provides a large-scale trend and simplifies or guides the placement of the control points in the domain (for a shape optimization problem). In surface sensitivity analysis the goal is to give an indication of morphing or displacement and not to modify the mesh. However, in order to compute a consistent displacement between the cells, avoiding discontinuities or unrealisable deformations, one can use the so-called mesh deformation algorithm *spring-analogy* [46]: with this approach, each edge of the mesh is replaced by a tension spring with a spring stiffness k_{ij} taken as inversely proportional to the edge length.
To solve the cell displacement, applying Hook's law to the nodes displacements, we impose the equilibrium of the forces between the i^{th} cell and the adjacent j^{th} ones. Considering δx_i the displacement of the centroid of the cell, the equilibrium is given by:

$$\sum_j k_{ij}(\delta x_j - \delta x_i) = 0 \tag{2.43}$$

For cells at the boundary, the displacement is known, so the equilibrium can be re-written:

$$\sum_{j,interior} k_{ij}(\delta x_j - \delta x_i) - k_i b \delta x_i = -k_{ib}\delta x_b \tag{2.44}$$

inverting for the displacement:

$$\delta \mathbf{x} = \mathbf{K}^{-1}B \;\; where \;\; B = \begin{cases} 0, & \text{for interior cells} \\ -k_{ib}\delta x_b, & \text{for cells on boundaries} \end{cases} \tag{2.45}$$

The following step is the computation of the adjoint equations. However, since surface sensitivity has not been used in the continue of the analysis, the development is not reported here but it can be found in [41].

The above discussed sensitivities are the one used in the remainder of this thesis, but there are many others, such as the one with respect to boundary conditions; in the majority of the analyses the boundary conditions are fixed or there is a limited possibility of modifications, as in our analysis, which makes the computation of this sensitivity meaningless. The reader is referred to [34] [41] for more information.

2.3.3.2 Adjoint topology optimization model

Topology optimization, as mentioned earlier, aims to minimize (or maximize) the user defined cost function by adding new features in the design domain. If this is done by changing the mesh, it can lead to extremely high computational cost, s the mesh has to be recalculated at each iteration. The most established technique for simulating a solid inside a fluid domain is the use of a *porous medium*, which assigns a small porosity to the cells identified through the LSF interface; the optimal domain topology emerges from the final porosity distribution. To simulate the obstacle in the fluid domain, we use the Brinkman Penalization model [43], which adds a source term in the momentum equation that forces the velocity to zero in those cells:

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = -\nabla \cdot \sigma - \alpha(1-\chi)\mathbf{v} \tag{2.46}$$

where α is the *Brinkman Penalization magnitude.* This term must be high enough to force the velocity to zero in the "*solid*" domain. Ideally, the solid material is completely impermeable for both convective mass transport and pressure diffusion: while the former is easily simulated by forcing the velocity to zero, the Brinkman approach cannot prevent the latter and thus may lead to errors in flow prediction [4] [30]. Furthermore, the interfaces between fluid and solid are not characterized as no-slip walls, leading to an incorrect turbulence modeling. As we will show, these shortcomings in the model may lead to incorrect or even unphysical results in the optimization process. This error in inherent in the model and cannot be avoided, even with a very high value of the penalization magnitude which, in addition, can lead to convergence issues. In the past years, several methodologies have been developed to try to overcome this issues, focused on a different characterization of the interface and the enforcement of a no-slip condition [29] or on a completely new characterization of the solid domain, using the Extended Finite Element Model developed to studying crack propagation [33].

Analyzing the equation 2.46, another term is added to the momentum equation: χ, called "*Material Distribution Function*" which is the function that governs the material distribution within the domain and it is defined as in equation 2.48. $\chi = 1$ identify that the considered cell is a fluid-domain cell, $\chi = 0$ identifies the cell as a solid one. During the optimization process χ can assume all values between 0 and 1, according to the sensitivity analysis. The continuous transition from solid to fluid facilitates the use of gradient based optimization algorithms but at the end of the optimization it is necessary to revert to a Boolean approach (solid *or* fluid); the solid region can be then be identified with a user defined threshold on the Material Distribution Function, characterizing all the cells with $\chi < 0.5$ as solid. Therefore, it is quite obvious that the output of the topology optimization is geometry whose contours are characterized by a staircase shape and that the new solid domain must be slightly post-processed to reproduce a smoother geometry and to delete the smaller, unfeasible features produced by the algorithm.

To solve the energy equations, the inserted porous medium is treated as a solid in thermal equilibrium, so the only change in the basic equation is the addition of the thermal conductivity of the solid. The effective thermal conductivity is then given by:

$$k_{eff} = \chi k_{fluid} + (1-\chi) k_{solid} \tag{2.47}$$

This model works quite perfectly for Low-Re flows (compared to the resolution of the fully resolved geometry), while for High-Re ones, accuracy slowly decreased but withing an acceptable level of discrepancy.
As stated before, in order to solve the sensitivities that allow the function χ to distinguish between solid and fluid, a level-set method is used; χ is therefore a function of the LSF ϕ defined as:

$$\chi = 0.5 \cdot \left(1 + tanh\left(\frac{\phi}{\delta}\right)\right) \tag{2.48}$$

where δ is a parameter that defines the thickness of the interface. This parameter is usually set to 0.1.
In order to have a better control on the development of the porous media, allowing for the creation of solid domain no only from the interfaces but also in the middle of the fluid domain, it is possible to add a source term S. Equation 2.20 therefore becomes:

$$\frac{\partial \phi}{\partial \tau} + V||\nabla \phi|| = S \tag{2.49}$$

where ϕ can vary between -1 (that identify the solid phase) or 1 (fluid phase). As stated before, τ is a fictitious time step that is used to simulate the development of the LSF inside the domain. Although it has no physical meanings, its value must be chosen properly, since a too high value could lead to the divergence of the optimization, while a too low one could require too much iterations for the convergence (remark: TO usually has high computational costs).

When performing an optimization, we usually need to find the minimum (or a maximum) or the zero of a generic function, usually not achievable analytically. To solve such a problem, some different algorithm have been developed over the years: from the well-known root-finding algorithms as the Newton's method (which require a derivable function in order to apply the algorithm to the first derivative for finding minima) up to Gradient based algorithms, particle swarm methods, artificial neural networks, etc. In general, we can distinguish between gradient-based and non-gradient-based algorithms: the former are usually quite accurate and can converge faster then the latter, but may suffer for robustness and for the presence of multiple local minima. Among the Gradient-based, an algorithm that has stood out from the others for topology optimization is *stochastic gradient descent* (SDG) [42]: classic gradient descent works well for continuous and easily derivable functions but it deteriorates if the data-set size becomes too large, affecting the convergence as well; moreover, if the function has local minima, the algorithm might get stuck in one of them and never reach the global minimum. Stochastic gradient descent overcomes this problem by randomly selecting data sampling and replacing the exact value of the gradient with an estimated one based on these samples and updating the parameters accordingly to the sensitivity functions. This approach is faster and requires less memory usage during the computation, as demonstrated by [10] [31].

To avoid the algorithm from getting stuck in a local minima and to improve the convergence, SDG is updated with the *Adaptive Moment Estimation (ADAM)*

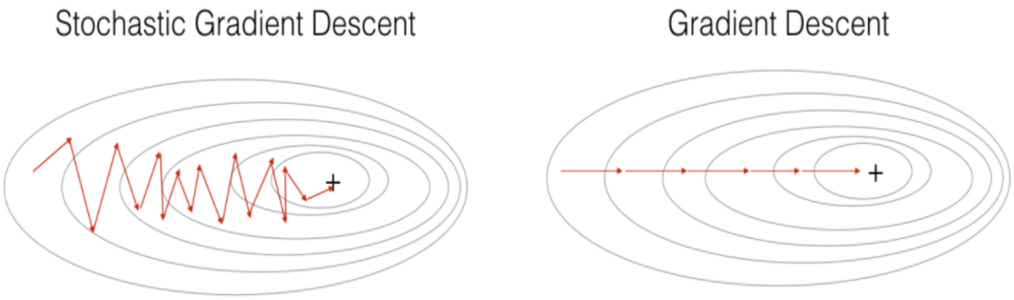


Figure 2.5: Differences between Standard Gradient Descent (right) and Stochastic Gradient Descent (left)

update rule [12], that gives to the searching direction a momentum (see fig. 2.6):

$$m_{k+1} = \beta_1 m_k + (1 - \beta_1)\frac{d\mathscr{L}}{d\chi} \tag{2.50}$$

$$v_{k+1} = \beta_2 v_k + (1 - \beta_2)\left(\frac{d\mathscr{L}}{d\chi}\right)^2 \tag{2.51}$$

$$\bar{v} = \frac{m_{k+1}}{\sqrt{v_{k+1} + \epsilon}} \tag{2.52}$$

where:

- β_1: constant that determines the momentum applied to the search direction. Higher values mean higher momentum, which means more difficulty for the gradient to change its direction. Usually, its value is set equal to 0.5;
- β_2: constant that determines the step-size velocity decay during the optimization. A higher value means a smooth but slow approach to the final result. Usually, its value is set equal to 0.5;
- ϵ: small constant to avoid a division by zero.

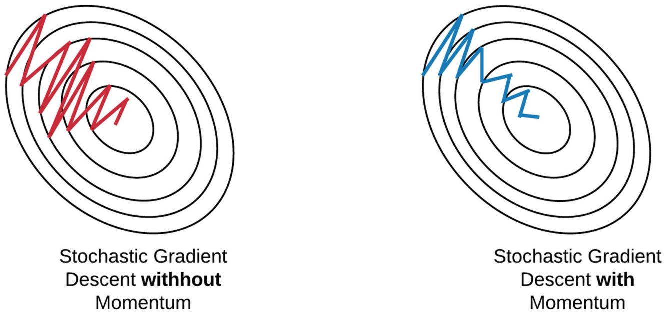

Stochastic Gradient Descent **withhout** Momentum
Stochastic Gradient Descent **with** Momentum

Figure 2.6: Difference of convergence for a SGD with (right) and without (left) momentum

It is now possible to define the velocity at the interface (recall equation 2.49) as a scaled value of the search direction:

$$V_i = \frac{\mathscr{V}_i \bar{v}_i}{\sum_k \left|\vec{A}_k\right|} \tag{2.53}$$

where $\mathscr{V}_i$ is the volume of the i^{th} cell and $\sum_k \left|\vec{A}_k\right|$ is the sum of the face areas of that cell.
Concerning the source term S in eq. 2.49, this is proportional to the search direction computed during the ADAM and it is defined as follows [41]:

$$S_i = -\omega[1 - sign(\bar{v}_i)\phi]\bar{v}_i \tag{2.54}$$

where ω is the source strength. This term is only considered ($\omega > 0$) when the formation of holes inside the domain is permitted, otherwise the solid can only grow from allowed boundary and $\omega = 0$.
While the ADAM rule helps the algorithm with the convergence, the choice of the direction is made by the sensitivity of the augmented Lagrangian function $\mathscr{L}$ with respect to χ. Considering the generic Lagrangian form as the one written in equation 2.4 and re-writing for our problem:

$$\mathscr{L}(x, \lambda) = f(x) + \sum_i \psi(c_i, \lambda_i^k, \mu) \tag{2.55}$$

where f(x) is the objective function, c is the constraint, λ is the Lagrange multiplier and μ the penalty parameter. The function ψ is defined as:

$$\psi(c, \lambda, \mu) = \begin{cases} \lambda c + \frac{\mu}{2}c^2 & if\ c \geq 0 \\ 0 & else \end{cases} \tag{2.56}$$

At each iteration, the Lagrange multiplier is updated:

$$\lambda_i^{k+1} = \begin{cases} 0 & if \quad c_i \leq 0 \\ \lambda_i^k + \mu c_i & else \end{cases} \tag{2.57}$$

The penalty parameter μ is one of the most important parameter in such a problems and assumes a very important role in the optimization, since it governs the weight that the constraints have on the optimization: the higher the penalty, the greater are the Lagrange multiplier and thus the importance of the constraint with respect to the objective function. An analysis on the importance of this parameter was performed in chapter 4.
For an optimization analysis to converge to the optimal solution, it is necessary that the Lagrangian are correctly scaled. Indeed, it is necessary that all the sensitivities of the constraints correctly represent the distance between the constraint itself and its desired value, in accordance with the value of the sensitivity of the objective function. Often the sensitivity fields of the objective function and of the constraints are very different one from the other, therefore a different scaling between them may be required. A possible approach is to define the constraints and the objective function is such a way that the obtained sensitivity fields are comparable, which is a not always that easy and may require some internal iterations of sensitivity analysis and corrections (For more information, see chapter 3.4.2).
Another possible solution is to normalize the objective and constraints by their greatest sensitivity value:

$$\hat{f}(x) = \frac{f(x)}{\left|\left|\frac{\partial f}{\partial \chi}\right|\right|_\infty} \tag{2.58}$$

$$\hat{c} = \frac{c}{\left|\left|\frac{\partial c}{\partial \chi}\right|\right|_{\infty}} \tag{2.59}$$

In the following analyses an in-depth comparison is made between optimization performed with or without such a normalization, in order to better understand the implications on the analyses themselves and on the results; as we shall see, the latter (usually referred to as "Automatic normalization") is not always able to correctly compare the different sensitivities of the objective function with respect to the constraints, leading to an incorrect final topology.
The theory reported here is just an introduction to all the existing ones concerning topology optimization and what lies behind it. Since the aim of this thesis is not the introduction of a new model or method for TO, its development stops here. However, all the theory necessary to understand the following of this text is contained in this chapter.

3
STAR-CCM+ Environment and Workflow

In this chapter, the STAR-CCM+ environment is presented with a focus on the most important aspects that will be used for the setup of the generic optimization problem. In particular, the mesh models that will be used and the physical models required for the Adjoint Topology Optimization (coupled flow, turbulence models, adjoint) will be discussed, together with the definition of the objective function and constraints. A fundamental step concerns the turbulence corrections required to reduce the error introduced in the solution from the Brinkman penalization model and its description of the *solid domain* through porous fluid cells.

3.1 Geometry definition

The first important consideration for the correct setting of a TO problem is the identification of the part of the fluid region that will take part in the optimization itself and, conversely, the one that cannot be modified. Although it is possible to consider the whole domain as a single fluid region, characterizing as "editable" only the boundaries from which the *solid* domain can grow, it is better, where possible, to distinguish between the inlet and outlet boundaries from the others. This ensures that solid domain cannot grow up to the inlet and outlet sections and, therefore, that these (that may be constrained by other external constraints not considered in the simulation) remain unchanged. Based on this considerations, from now on we will refer to the region that will take part in the optimization process as **TO-Region**, while the other will be called **Fixed-Region**. The differences between the single-region and the multi-region approach are purely logistical: First, it is more convenient not to include boundaries that do not need to be modified in the TO-Region, to simplify the boundary conditions (there is no need to set for each boundary a fixed constraint). On the other hand, this will lead to the creation of more interfaces between regions, but, if the mesh at the interface is set correctly,

the error introduced is negligible and therefore does not represent a problem for topology optimization (see chapter 5).
Another important aspect to considered is that topology optimization gives better results if the TO-Region is larger than the "base" geometry, since it has the more freedom and can generate a more varied topology. This consideration leads to the creation of a domain that *includes* the basic geometry and extends it, taking in account external constraints and limitations. On the other hand, if the external geometry is to be kept fixed, it is possible to study only an internal optimization starting from the basic geometry. Contrary to what may appear, an internal optimization is not in conflict with the concept of topology optimization (see chapter 4.2 and chapter 5).

3.2 Mesh models and parameters

The generation of the mesh is a fundamental step in all the CFD analysis, especially for a topology optimization problem that defines the solid domain through the porosity assigned to each individual cell. This in the TO-Region may require a finer mesh than the one that would have been enough to guarantee a convergence for the fluid field. In the following analysis, the optimizations are performed both with structured and unstructured meshes, in order to better understand the problems that their differences could introduce in the optimization. Here, the two mesh models that we will use later are briefly described:

- *Trimmed Mesh*: generates predominantly hexahedral cells. It can guarantee a high mesh quality but it can suffer from complex geometries with high curvatures. The mesh is generated by the intersection of the geometry with a outer grid and with a refinement near the boundaries where geometry intersect the grid itself. Although this mesh model generates cubic-shaped cells, it is not exactly a structured mesh if the geometry develops independently over the 3 dimensions or, on the contrary, if the grid is not perfectly aligned with it; it is possible, for simple geometries, to generate the grid according to the main direction of the domain and performing a full structured mesh. It is independent from the surface mesh quality but it cannot generate conformal mesh at the interfaces. An example is shown in figure 3.1.

- *Polyhedral Mesh*: generates polyhedral-shaped cells by connecting the centers of the cells generated by the surface mesh. These will be the model used for the unstructured mesh cases. This model is the preferred one for complex geometries or domain with different regions since it allows the generation of a mesh with conformal interfaces. The greatest advantage in using polyhedral is that a volume mesh generated with this model contains about five times fewer cells than an unstructured one generated, for example, using tetrahedra, for a given starting surface mesh [41], but it is still capable of achieving a high quality mesh. The polyhedral cells that are created typically have an average of 14 cell faces. An example is shown in figure 3.2.

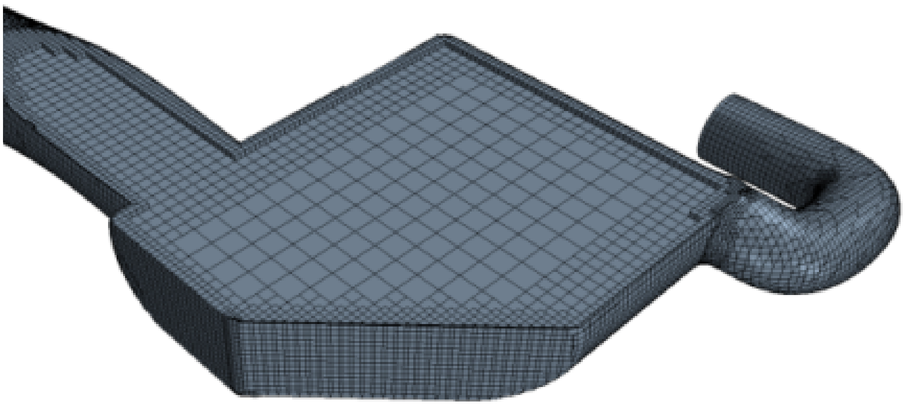


Figure 3.1: Example of a Trimmed Mesh [41]

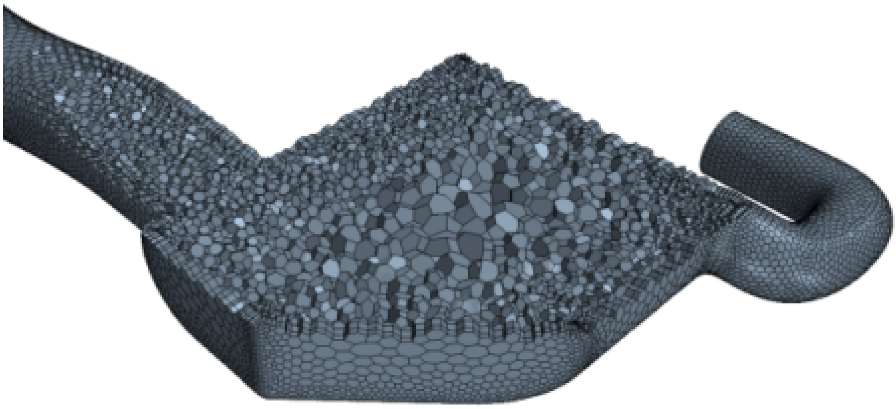


Figure 3.2: Examples of a Polyhedral Mesh [41]

The main parameter that we will use to distinguish the mesh size, for example for the mesh convergence analysis, is the **Base Size**. For in-depth mesh control, it is also possible to define, among many others, a minimum and a maximum size (relative or not to the Base Size), a growth rate (both for surface and volume mesh), a surface curvature refinement. Some of these parameters will be used later on.
As we already know, to correctly capture the near-wall behaviour and for a correct turbulence modeling, we need to obtain a Y^+ consistent with the chosen turbulence model; the thickness of the first layer is therefore a fundamental aspect in the mesh generation. As illustrated in the next chapter (ch. 3.1), the turbulence model used for the analysis can evaluate the Y^+ of each cell and chose accordingly whether to use or not the wall functions. Even though these can be less performing in the turbulence modeling, this model allows use to deal with complex geometries whit different physical scales, as the one illustrated in chapter 5.

It is clear that there is not a best mesh for all the simulations but the choice of the mesh model depends on the problem under analysis. For simple geometries (see chapter 4) without curvatures, the trimmed mesh model is more suitable, but for complex geometries with small features or large curvatures the polyhedral model can achieve a better quality with fewer cells. To better understand the differences

between the two mesh models, an analysis was performed with the test case described in chapter 4.1 performing a convergence analysis for the flow solution with both the trimmed mesh model and the polyhedral one.
But does the mesh model influences the topology optimization? And if it does, how? Does its result strictly depend on the size of the mesh? As mentioned, the topology optimization loop (fig. 3.9) has a very high computational cost, so it comes natural to investigate the influence of the mesh size on optimization solution. In particular it is important to understand whether the use of a mesh that is not strictly at convergence can affect the result and, if it does, how much is the error introduced. This analyses were performed again with the test case described in chapter 4.1 and the results are presented in a dedicated section (chapter 4.1.2.1) but before answering this question is necessary to introduce the physic of the problem and the error introduced by the adjoint topology optimization solver itself.

3.3 Physics models for the Adjoint Topology optimization

A correct and good-quality mesh is the starting point for a proper analysis, but it needs to be accompanied by a the correct description of the physics of the problem and, thus, with the appropriate choice of physics and turbulence models. Adjoint topology optimization forces some aspects of this description, requiring not only the resolution of a coupled flow model but also a stationary fluid flow. These constraints are due to the algorithm governing the adjoint solver and topology level set function (as described in chapter2.3.3.2). Although these "constraints" may appear to be restrictive, they actually reflect the idea behind an optimization: it is a different concept, in fact, to perform an optimization if the solution changes in time (in which case the optimisation will be based on the solution obtained as a time-averaged solution). In addiction, the coupled flow model (which, by the way, can achieve a higher accuracy on the solution than the segregated one) is necessary for the computation of the sensitivities with respect to the material indicator (see chapter 2.3.3.1).
We will not go into the details of each model that we will use in the analyses (gathered in tab 3.1), assuming them as previous and acquired knowledge.

The models that deserve the most attention are the turbulence ones. The one used in the simulations is the Menter's $K-\omega$ SST, as is the best solution for our cases since all of them concern internal flows and may suffer of adverse pressure gradients. Moreover, this model ensures a good robustness, which is crucial for a problem whose domain can change drastically during the iterations. The all-Y^+ wall treatment uses blended wall functions that emulate the low-Y^+ wall treatment for fine meshes, and the high-Y^+ wall treatment for coarse meshes. It also works well for meshes of intermediate resolution, i.e. when the centroid of the first cell falls in the buffer region of the boundary layer. Therefore, this wall treatment is suitable for a wide range of near-wall mesh densities and it is the best one for domain with different mesh scales.
The main problem is to capture the turbulence near the porous domain: for the computation of all the turbulence quantities and the application of the wall functions,

Table 3.1: Physical models. In bold the models required for the TO

Physics	Gas
	Three-Dimensional
	Ideal Gas
	Coupled flow
	Coupled energy
Time	**Steady**
Turbulence	RANS
	$K - \omega$ SST (Menter)
	Wall Distance
	All Y^+ Wall Treatment
	User-scaling Turbulent Viscosity
Adjoint	Adjoint Flow
	Adjoint Frozen Turbulence
	Adjoint Topology Optimization

it is necessary to know the distance of the nearest cell from the walls. As stated in the previous chapter, the All-Y^+ model can compute the distance of each near-wall cell and therefore compute its Y^+. However, this can only be performed with those cells **close to a non-slip wall**: this limitation is due to the the fact that there is no need to perform such computations for other types of boundary. It is therefore clear where an error can be introduced: even though we are simulating a *solid*, it is actually porous fluid cell and so it is not treated as such, there is no prism layer, no physical walls, no Y^+ calculation and no wall functions. In other words, there is no near-wall turbulence model applied near the porous medium. The error introduced is hence unavoidable: it is not possible to manually introduce the correct turbulence behaviour near the porous domain, since the mesh where solid growth is not defined to be a near-wall mesh and we usually do not know where the interface of the LSF will generate the solid boundaries. The final behaviour, on the contrary, is almost that of a rough surface. This error is intrinsic in the Brinkman penalization model and in all the models that use a fictitious porosity to simulate the solid domain. As a consequence of the incorrect turbulence modeling, two main considerations can be made: first, the solution predicted during the optimization itself is not correct and reliable from the physical point of view; second, the presence of non-zero turbulence field **inside** the porous media. As clearly described, the solid domain is, in fact, fluid domain with a very high porosity. Thus, the presence of turbulence (enhanced by the porous media itself) is possible, which is totally unrealistic if it were a real solid. This turbulence can also spread through the porous medium within the domain and reach areas of the fluid where it should not be, thus affecting the characteristics and behavior of the fluid field. The simplest (but efficient) way to destroy turbulence within the porous domain is to force the turbulence viscosity to zero in those cells. To identify them we use the *Material distribution function* χ defined as eq. 2.48, recalling that 0 means solid cells, while 1 means fluid cells. Since it is a continuous function, it can assume all the values between 0 and 1: by convention, the following is assumed:

$$\chi \begin{cases} \leq 0.5, & \text{Solid domain} \\ > 0.5, & \text{Fluid domain} \end{cases} \tag{3.1}$$

The turbulence correction is then achieved by introducing a new function called *TVRScale*, which, multiplied to the turbulent viscosity computed in the fluid field, destroys all the non-null turbulence inside the porous media. According to fig. 3.3, it is clear that the TVRScale function influences the value of μ_t only if the *Material indicator* value is below 0.5, while if it is greater than this value it is almost irrelevant. An example of the effect that this function has on the turbulence field is shown in figure 3.4, from which it is also possible to better understand the wrong turbulence field in the porous media: in fig. 3.4a the contour of the porous region is described by the black line inside the domain: it is quite evident that even if it is the *solid* one, in reality it is a fluid domain and so, even though the velocity is almost zero thanks to the Brinkman Penalization, a turbulence is developing inside, which can seriously influence the flow behaviour. Moreover, analysing the turbulent viscosity values, it is clear why there is inherently such an error in the model: both the optimization simulations (the one with $TVRScale$ and the one without) have a very high turbulence viscosity generation near the porous domain (also due to the roughness of the porous domain) compared to the one obtained with the correct turbulence analysis of the re-mesh surface, which is also of an order of magnitude lower (the new surface was not smoothed for a more consistent comparison).

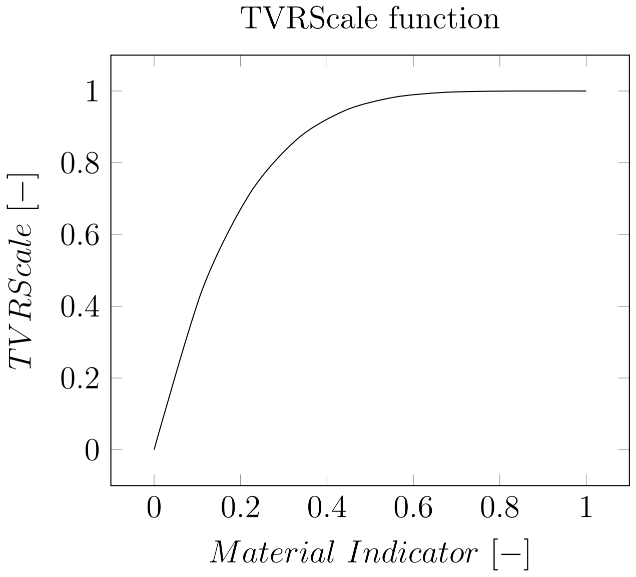
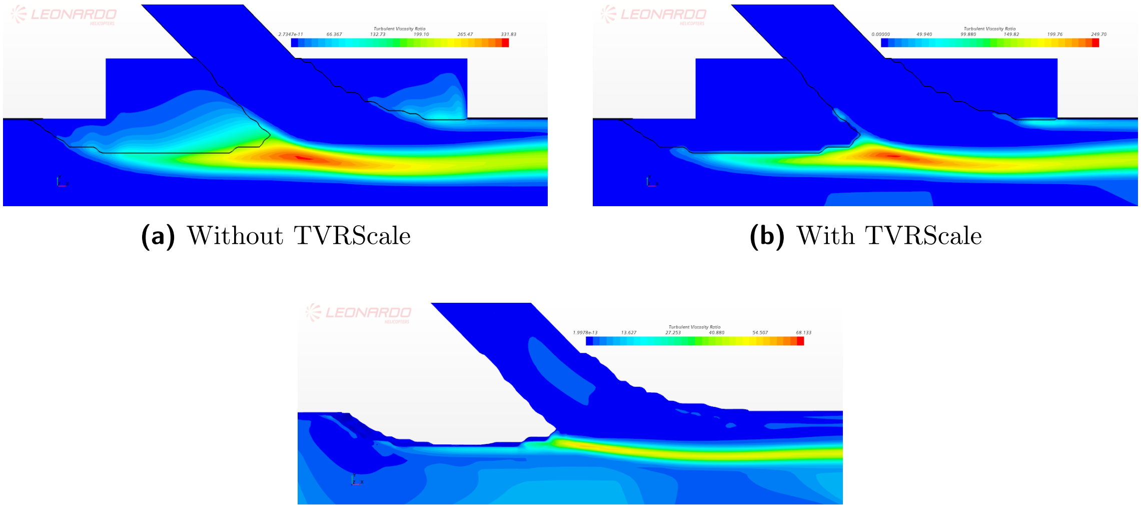


Figure 3.3: TVRScale function

Analyzing the results of the optimization performed on a simplified domain of the test case described in chapter 4.2, it is evident that there is (and always will be) an error, even a major one, during the optimization iterations, for which the results must be taken as an *indication* of an improvement. From the results reported here, the *TVRScale* function was able to reduce the error, for example for the total pressure drop, from the 7.5% to a 4%, or for the P_{IPS} from a 9% to a 4%. However the error must be taken into account when performing the optimization, since this



(a) Without TVRScale

(b) With TVRScale

(c) Reconstructed domain (correct prism layer mesh everywhere)

Figure 3.4: Turbulence viscosity ratio field with TVRScale effect. The isoline inside the domain highlights the porous media

reflects on the choice of the constraints value. Based on these considerations, a reconstruction of the geometry with a new simulation to validate the domain is always necessary and essential.

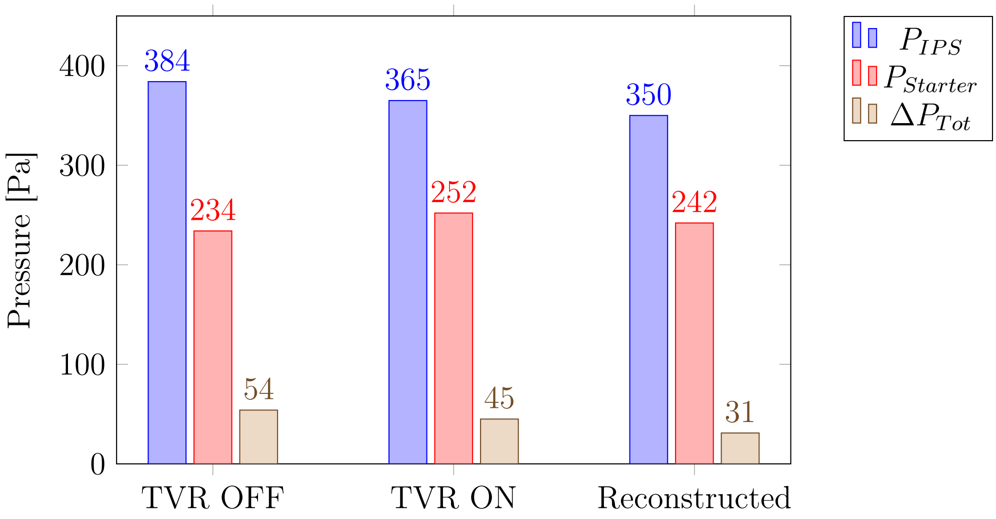


Figure 3.5: Effect of turbulence correction on the reduction of the error

3.4 Adjoint Topology Optimization setup

In this section, a focus on the setup for the Adjoint Topology Optimization is proposed, starting from the settings of the adjoint solver itself up to the construction of the objective function and the constraints. The latter are reported here with their definition and with a brief explanation of their role in the sensitivity analysis, but the reasons for such a definition are analyzed in depth in chapter 4. The reader is then invited to take note and become familiar with them, in order to better understand their influence on the analyses performed later.

3.4.1 Adjoint Setup

The topology optimization process does not need a high level of accuracy in the computation of the adjoint variables, since these are taken as an *indication* of where material should be placed to improve the cost function and/or to satisfy the constraints. The domain can change drastically between two iterations, especially at the beginning of the optimization, so a too high accuracy in the adjoint computation would not introduce any beneficial effects in the solution, but would only increase the computational time. For this reason, the adjoint solver is solved at the 1^{st} order with only one iteration, with a Krylov space dimension of 50 [1] and a relative tolerance target of $1e-10$. This tolerance is almost always reached before all the 50 iterations of the Krylov space are computed.
These settings are sufficient for the topology optimization since more accuracy would increase the computational cost with almost no additional information. It is important to remember that the optimization process is made of a sequence of primal and adjoint iterations, which requires a (good) convergence of the primal solution and then the adjoint computation: this not only leads to high computational costs but also to override the previously computed adjoint quantities. At the very beginning of the optimization, the changes in the domain may be enough to completely change the fluid field and all the adjoint variables, so a high precision of the latter is unnecessary.
The most important output of the adjoint analysis for this kind of optimization is the Sensitivity of the considered physical quantity with respect to the Material indicator (one for each constraint plus the one of the objective function). These indicate to the solver how the desired quantities would change if material indicator changes in the cells of the domain. Knowing how far the constraint is from the desired value, based on the sensitivity values, the LFS interface velocity changes to modify the solid domain, in order to increase the constraint satisfaction while, at the same time, increasing (or minimally decreasing) the objective function (or vice versa if the goal is the minimization of the objective). If the constraints and the objective function are related to the same physical quantity (pressure, mass flow, velocity, etc.) and are of the same order of magnitude, the computed sensitivities can already be used for the analysis. Otherwise, one of the constraint can prevail over the others, leading the optimization in a wrong direction (see chapter 4.1.3). This is why it is so important a new definition of the reports with a different weight and normalization for each constraints.

3.4.2 Report definition

Solving an optimization problem that takes into account multiple constraints means looking for the best tradeoff between improving the objective function and satisfying the constraints themselves. This compromise must be chosen by the user, who decides the importance of the satisfaction of each single constraint. It is not necessary that the constraints are not violated during the process, but it is essential that they are satisfied at the end. In developing the following optimizations, three different physical quantities will be mainly considered: Mass flow, Pressure Drop and Solid Volume Ratio (SVR). While the first two are well known, the last one must be properly understood. Its physical meaning is quite simple and obvious: SVR simply indicates the fraction of solid cells within the TO region. The reason for this constraint, on the other hand, is less obvious: while the requirement for a solid maximum may be consistent (for reasons of space and/or weight), the request for a minimum may appear counter-intuitive, as it is generally a quantity that is not known or desired as an output of the optimization itself. However, this constraint is necessary since the Topology Level Set method, for its definition, *needs* an external input to force the interface to enter in the domain, especially at the beginning of the optimization itself. Choosing a minimum SVR can be problematic in some cases, since its value is often unknown or not required at all for the purpose of the optimization, so in some cases different optimizations with different values of SVR may be needed. Recently, some topology optimization methods called "Performance-based methods" have been developed to try to overcome this issue [21].
From the analysis performed in chapter 4, it appears that the basic approach with the normalization of the sensitivities as described in eq. 2.58 may suffers for convergence issues. Therefore, a new approach is needed, which is able to give different weights to each constraints and to correctly reproduce the distance between the constraint itself and its limit value. Moreover, it appears necessary to ensure an adequate sensitivity not always when the constraint is violated but also when it is satisfied: this is the reason why the new constraints are defined as a continuous and derivable function, with different behaviours before and after their limit value. Here below the new definition of the reports are described:

- Pressure Drop: this constraint computes the total pressure drop between two sections. It works well for problems with simple geometries, but for multi-inlet ones (as the one in chapter 4.2 e chapter 5) the total pressure drop between the desired section and the outlet depends on the mass flow from the other inlets. For the topology optimization to work properly, it is crucial that the report definition is unique, otherwise it can be lead in a different direction than the desired one. It is therefore clear that forcing a maximum value on the pressure drop asking for the maximization of the mass flow on the other inlet sections (as done for some cases later on) can generate an inconsistent constraint. To better understand the problem we can consider the scheme shown in figure 3.6: while in case 3.6a the pressure drop is uniquely defined, for example for the section 2, in case 3.6b the correctly computed pressure drop should take into account both the inlet sections. Since the mass flow rate coming from inlet 1 influences the pressure drop of the whole system, if we want to reduce the pressure of the inlet section 2 we cannot use the pressure drop of the whole system.

To overcome this issue, since in our problem the mass flows of the sections whit a constraint on the maximum pressure are defined as a boundary condition, the constraint is imposed on the static (or total) pressure on that boundary. Following the considerations made before on the continuity of the constraint function, here we give the general definition of that function, where P denotes either the static (or total) pressure or the pressure drop.

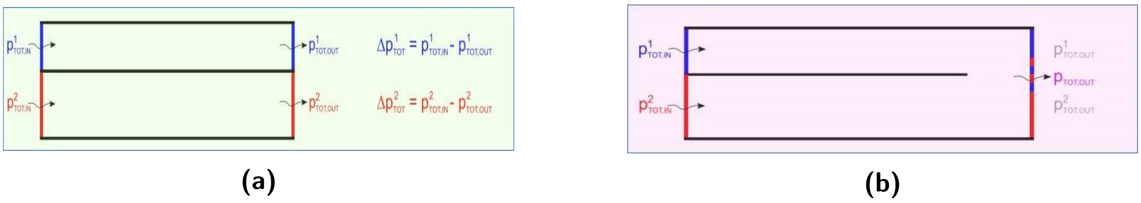


Figure 3.6: Difference on Pressure Drop computation

$$P_{CF} = \begin{cases} \alpha_p \cdot \left[\frac{P-P_{max}}{P_{norm}}\right]^2 & P \leq P_{max} \\ \beta_p \cdot \left[\frac{P-P_{max}}{P_{norm}}\right]^2 & P > P_{max} \end{cases} \tag{3.2}$$

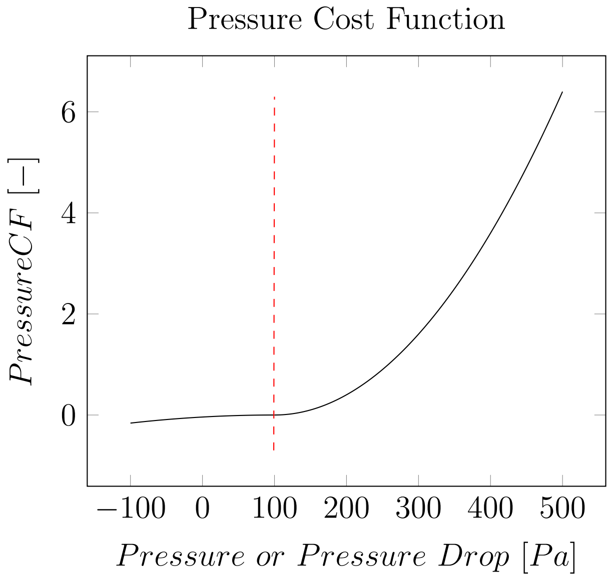


Figure 3.7: Example of a Pressure Cost Function. $\alpha_p = -1$, $\beta_p = 10$, $P_{max} = 100Pa$, $P_{norm} = 500Pa$

Observing the cost function (fig. 3.7), it is equal to zero if the pressure is equal to the maximum value, while it increases for greater values. Thinking about the nature of this constraint, it is clear that if the final pressure is lower than the maximum value is a benefit of the optimization, so the constraint will be

$$PressureCF_{max} = 0 \tag{3.3}$$

For lower values of the pressure the solver has to know the value of the cost function and its derivatives, so we must define a derivable function. However,

a too high derivative may force the solver to decrease the pressure, as if it was the objective function. Since we do not want such a behaviour, the values for $P < P_{max}$ are negative but at least a order of magnitude smaller than the values for $P > P_{max}$. This behaviour is obtained with $\alpha_p < 0$ and $\beta_p > 0$ (for a *minimum* constraint they are reversed).
If the pressure is the objective of the optimization, it can not be scaled as depicted since we will give a minimum to the function, but it is simply scaled with a normalization value to adjust the sensitivity:

$$P_{obj} = \frac{P}{P_{norm}} \tag{3.4}$$

- Mass Flow: this constraint computes the mass flow rate trough a boundary (or the sum if computed on multiple boundaries). Usually this is a boundary condition or an objective function, but in particular cases (as the one in chapter 4.1) it can be a constraint. As described for the pressure constraint, if it is the objective function, it is simply defined as

$$\dot{M}_{obj} = \frac{\dot{m}}{\dot{m}_{norm}} \tag{3.5}$$

  while if it act like a constraint, it is defined as a continuous and derivable function:

$$\dot{M}_{CF} = \begin{cases} \alpha_{\dot{m}} \cdot \left[\frac{\dot{m}-\dot{m}_{max}}{\dot{m}_{norm}}\right]^2 & \dot{m} \leq \dot{m}_{max} \\ \beta_{\dot{m}} \cdot \left[\frac{\dot{m}-\dot{m}_{max}}{\dot{m}_{norm}}\right]^2 & \dot{m} > \dot{m}_{max} \end{cases} \tag{3.6}$$

- Solid Volume Ratio (SVR): as explained before, this constraints defines the ratio between the cells defined as solid over the other cells of the TO-Region. Similar considerations made for the above constraints can be made here, with a slightly difference in the behaviour after the constraint violation:we cannot impose that a volume greater than the minimum required is a positive feature, as this could lead to the introduction of solid only and exclusively for an improvement in the constraint without a real need, but we must still define a continuous and derivable function. The behaviour after the constraint *satisfaction* is therefore a (small) growth in the cost function itself. Analytically, the expression is the same as the one used for the Pressure Drop report, but the coefficients α_{svr} and β_{svr} are both positive:

$$SVR_{obj} = \begin{cases} \alpha_{svr} \cdot \left[\frac{SVR-SVR_{min}}{SVR_{norm}}\right]^2 & SVR \leq SVR_{min} \\ \beta_{svr} \cdot \left[\frac{SVR-SVR_{min}}{SVR_{norm}}\right]^2 & SVR > SVR_{min} \end{cases} \tag{3.7}$$

$$SVRCF_{max} = 0 \tag{3.8}$$

It is fundamental to understand that these reports are just a user-defined characterization of the problem and the minimum value of the cost functions does not represent the optimal solution for the problem: those functions, indeed, are just a scaling of the desired physical quantities. On the contrary, the objective function is

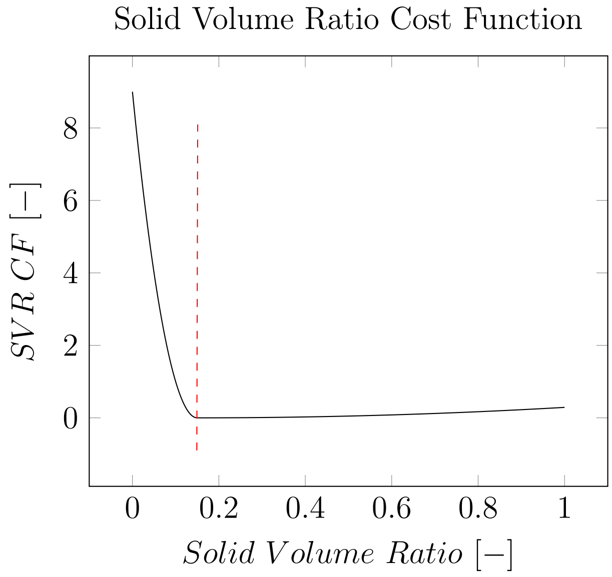


Figure 3.8: Example of a SVR Cost Function for a request of a minimum SVR value. $\alpha_{svr} = 1$, $\beta_{svr} = 0.001$, $SVR_{min} = 0.15$, $SVR_{norm} = 0.05$

a linear function of the physical quantities, so the minimum value is not user-defined and it depends on the solution.

It is also essential to understand the difference between the different weight given with this approach and the penalty parameter introduced in eq 2.55: the latter indeed is a single parameter, equal for all the constraints, that is used for the Lagrange update (eq. 2.57) to take into account the constraints satisfaction, but it cannot distinguish between them. With the automatic normalization approach, the penalty parameter is **also** used for weighing the constraints with respect to the objective function. With the manual approach, this task is entrusted to the different sensitivities obtained with the new constraints definition and the penalty value is used only to introduce the constraint in the Lagrange multiplier updating. The consequence (as analyzed in chapter 4.1.5) is that in the manual approach a smaller value can be used and its influence is less relevant for the constraint satisfaction. An in-depth analysis of the influence of the *Penalty* parameter, together with the one of the *Stepsize* (which governs the pseudo-time τ in eq. 2.49) are proposed in cap. 4.1.

3.4.3 Topology Optimization Loop

The topology optimization solver, unlike other adjoint problems, benefits from frequent recalculations of the primal and adjoint flow and may require many loops to reach the optimal solution. We have already seen in figure 2.2 the basic workflow for a correct TO analysis from the structural point of view; here, the same is translated in the fluid-dynamic language (fig. 3.9). The most important distinction that we have to made is on the difference between "Base Solution" and "Primal Analysis": both of them refer to the solution of the flow-field, so formally speaking they are the same. However there is a difference in the TO context:

- Base Solution: this is the resolution of the primal flow that represents my reference condition and starting point for the analysis. A *good* convergence for this step is required since it must ensure the proper flow behaviour, regardless of the initial condition.

- Primal analysis: as said, this is again the resolution of the primal flow, but it does not require the same high level of convergence of the Base Solution, since it is just a step in the optimization process and it is needed to update the LSF interface and for the new computation of the adjoint quantities. Once the solution converged to a defined topology, the convergence of the solution itself increase automatically, since the domain remain nearly unchanged as the loop proceeds. Usually, a fixed number of iterations that guarantee a reduction in the residuals in the order of $10^{-3} \div 10^{-4}$ is enough.

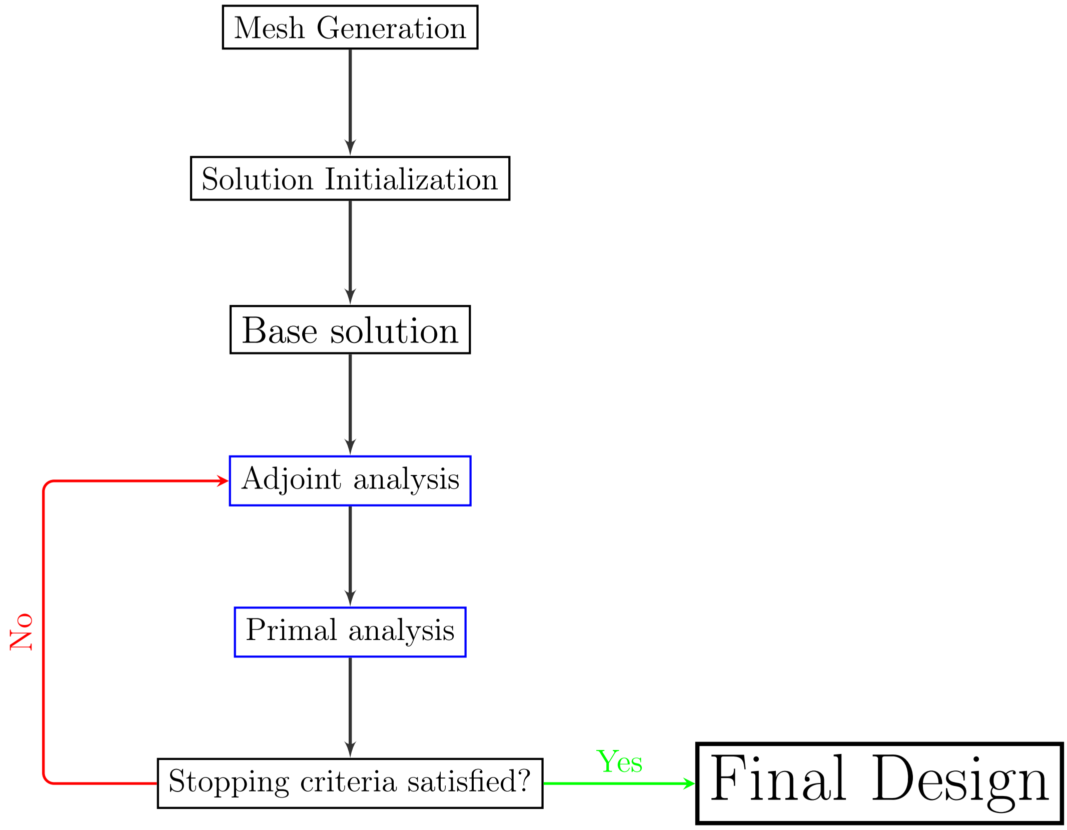


Figure 3.9: Topology optimization Loop

3.4.4 Stopping Criteria for the Topology Optimization

When performing an optimization problem, some simple questions arise spontaneously: when can we consider the optimization converged to a final solution? When and how much can we accept a constraint violation if it results in a higher value of the cost function? When are the oscillations acceptable? By analyzing the plots of the constraints (see for example fig. 4.20), it is clear that the solver aims to satisfy the constraints even if this means increasing the cost function and then reducing it as long as possible until it starts to oscillate close to the optimal solution. If these oscillations are small enough and the topology does not change significantly, the solution can be considered as the optimal one and the solver cannot get closer than that: the solution has converged. It is not necessary for the solution to be perfectly stationary on a final topology: first, the level set boundary is driven by the sensitivities and its interface moves continuously, second, as mentioned before, the geometry *must* be regenerated and cleaned, so small oscillations are irrelevant. Finally, as described in cap 4.1.2.1, the final description of the geometry depends on the generated mesh: finer features are not captured if the mesh is not small enough, and thus the solution may oscillate around two near-optimal configurations. In order to stop the topology optimization loop even with small oscillations, a particular stopping criteria called **Optimization Convergence Criteria** was set, along with two other criteria that prevent the optimization from moving forward if it is diverging (note: the primal solution can converge also if the optimization is diverging):

- Maximum Optimization iterations: this first stopping criteria avoid the optimization to go on forever, normally is set to 150.
- Solid Volume Ratio asymptotic criteria: this criteria stops the iterations if the change in the SVR from one optimization iteration to the other is lower than 0.1%. If this criteria is satisfied before the Optimization Convergence Criteria, it means that the optimization cannot improve in the modification of the geometry (the level set function boundary cannot move anymore) but the constraints are not satisfied yet. A modification in the
- Optimization Convergence Criteria: this criteria is simply made of a scalar parameter which is 0 at the beginning. This parameter is defined as follow:

$$OCC^0 = 0;\ OCC^{i+1} = \begin{cases} min(OCC^i + 1,\ 10) & \forall\ Constraints \leq 0.1 \\ max(OCC^i - 1,\ 0) & \exists\ Constraint > 0.1 \end{cases} \tag{3.9}$$

  If all the constraint functions have a value lower than a user-defined threshold, for example 0.1 but can be different for each constraint if needed, (remember, 0 means that the report equals the constraint), then the parameter value increase by 1 at each optimization iteration, otherwise it decreases by 1 (down to 0). The optimization is considered at convergence if the Optimization Convergence Criteria reaches the value of 10 (which means that in the last iterations the constraints were sufficiently satisfied).

These constraints are the ones that govern the TO loop depicted in figure 3.9.

4
Procedure Development and Test Cases

This chapter aims to develop the required knowledge and a robust procedure for the correct implementation of the adjoint topology optimization model as presented in the previous chapter. Starting from a simple Quasi 2D analysis of a mass flow partitioning up to a full-fledged 3D problem, the main aspects of a CFD simulation are analyzed and tested, with an eye on the topology optimization results and performances. With the simplest Quasi-2D problem, the dependence of the topology with respect to the main governing parameters is analyzed, starting from that with respect to the mesh model and mesh size, up to the one with respect to the *Penalty* and the *StepSize* values, concluding with the differences between the automatic and the manual normalization of the sensitivities. In the 3-D case the latter is resumed and compared to the one of the 2-D case, along with the analysis of the effect that the 3-D structures have in the final topological description. In both 2D and 3D cases, the effect that the porous medium has in the flow development is analyzed, along with the analysis of the error that it introduces into the solution and the beneficial effect brought by the *TVRScale* function.

4.1 Quasi-2D mass flow splitting

The first case that has been studied is a simple Quasi-2D flow division (fig. 4.1). The goal of this optimization was to find the optimal topology that could provide an equal mass flow splitting between the two outlet sections, while minimizing the system pressure drop. Since this simulation is a simple test case, with no external constraints, only one fluid region has been created, but all the boundaries of the inlet and outlet sections are constrained not to create a solid domain (see chapter 4.1.1 for in-depth analysis). The squared section represent the entire topological domain, within which the generation of the porous domain will occur. The Quasi 2D formulation was obtained with a much smaller z-component of the domain than the others and a characterization of the z-normal boundaries as periodic boundaries.

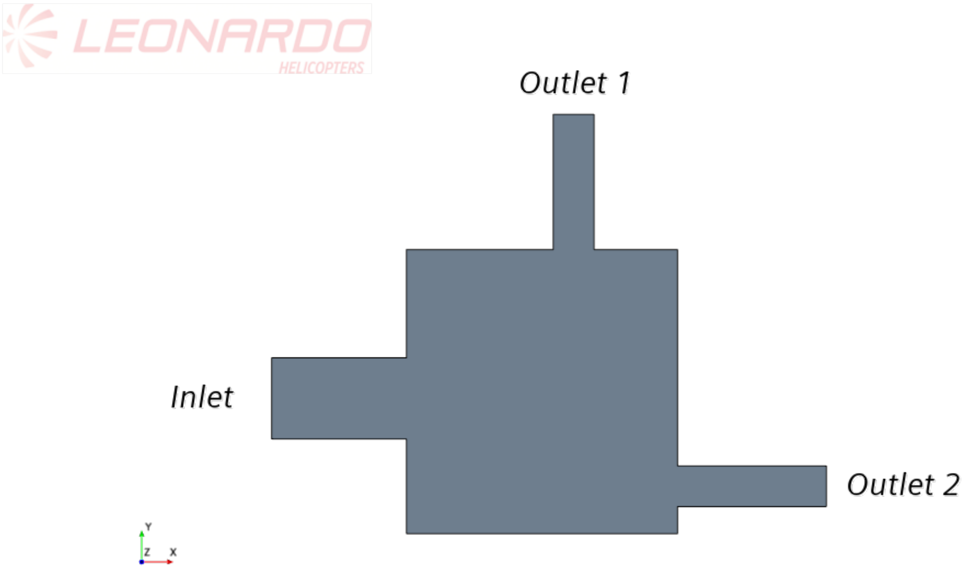


Figure 4.1: Initial domain of the mass flow splitting test case

4.1.1 Boundary conditions

This case represents a simple division of the mass flow into two different ducts, with one inlet section and two outlet sections. The first one, that is $0.06m$ wide, is set to be a mass flow inlet with $\dot{m}_{in} = 0.071kg/s$ (which gives a $Re \approx 30000$), while the other, that are $0.03m$ wide, are pressure outlets. In order to replicate a quasi 2D flow, the z-normal boundaries are characterized with a periodic interface, while the remaining boundaries are non-slip walls. The most important boundary conditions for the latter (from the optimization point of view) are those for the Material Indicator function, since these govern the growth of the solid: it is possible, in fact, to set the value of the Material Indication function to tell the solver whether that particular boundary should be fluid (imposing a constant value of 1) or solid (imposing a constant value of 0). Since the Material Indicator function is a continuous function, all the values between 0 and 1 are allowed, but not recommended since its physical meaning would be lost. Characterizing the boundary as fluid does not prevent the solid phase from reaching that boundary if necessary, but only prevent it to be a source for the solid material. Therefore, this condition is not suitable if there are external geometric constraints that do not allow a different geometry (e.g., manifolds), for which it is advised to separate that region from the TO-Region. It is necessary for at least one boundary to have a Material Indicator value of 0 since it is necessary for the porous medium to have some source of solid from which to start. For obvious reasons, this condition only applies if the boundary is a non-slip wall. A more freedom can be given to the boundary without choosing any value of the Material indicator function and leaving the level-set free to move within the domain; this choice is suitable for all the boundaries that can be modified if necessary but that are not the ones that I need to be modified.

Another key aspect is the Initial condition of the Material Indicator function:

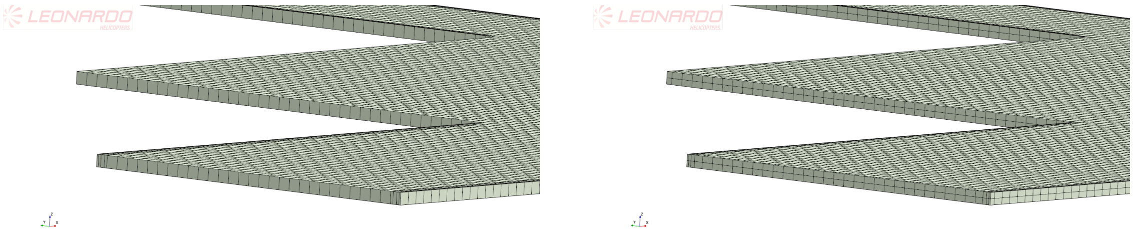
normally, the optimization should start with all the domain filled with fluid, i.e. the Material Indicator function is equal to 1 in the whole domain. However, since we are performing an optimization analysis, the search for the global minimum may be affected by the starting point of the search itself. Although the ADAM rule should prevent the algorithm from getting stuck in local minima, guessing a favourable initial condition can improve the convergence and reduce the number of iterations required. In addition, inserting a porous medium into the domain can help the solver find more complex topology since it has pre-made porous media from which the solid can grow (the porous medium can only grow from specified boundaries or cells with a material indicator of 0). An example of this implementation can be found in the optimization of the primal exhaust in chapter 5, where a particular initial condition is used.

4.1.2 Mesh Analysis

In the generation of a Quasi-2D problem, the *Trimmed* mesh model is the more suitable one since it allows the generation of a volume mesh with only 1 cell in the z-direction (fig. 4.2a). Due to a better control over the cell size and the alignment with the boundary of the domain, this type of mesh can better represent the two dimensional behaviour (remember: the trimmed mesh, as stated in chapter 3.2, is generated by crossing a mesh grid with the domain. By aligning this grid with a particular boundary of the domain and not with the origin of the reference frame, the intersection in the z-direction can be avoided). As mentioned in the previous chapter, an analysis of the mesh is necessary to understand how the optimization is influenced by its size and whether a fully converged mesh is needed to obtain a good result, or whether a coarser one can be generated, reducing the computational cost of the analysis. Since the mesh size strictly influences the dimension of the smallest features that can be outlined, it is interesting to instigate whether the optimization can still converge, even if with a rougher geometry. In the case of positive convergence, the optimization may forego the possibility of finer modification to the topology for a faster convergence. The mesh size also strictly influences the Solid Volume Ratio constraint, as it governs the minimum percentage increment given by a single cell. Therefore, the choice of its size influences the tolerance on the satisfaction of the constraint and, if the mesh is too coarse, it can lead to oscillations on the final solution around different (near-optimal) configurations. It is clear than the larger the domain, the less influence the individual cell has on this constraint, so the choice of a finer mesh depends on the problem in analysis. It is important to remember that topology optimization is a high-computational cost operation per se, so generating a mesh that is too fine can drastically increase the time required for the optimization and reduce the performances, so it is not always the best choice. In addition to this, the geometry obtained by topology optimization is a rough discretized geometry, which must be smoothed and cleaned after the optimization itself. Therefore, the best result that the topology optimization can give is an indication of the best geometry and not an off-the-shelf one. Based on these consideration, it is clear that the choice of a too fine mesh can be counterproductive, increasing the computational time without adding any more information to the solution.

4.1.2.1 Mesh models and size influence on topology optimization

In this section the different mesh models are analyzed in relation to the topology optimization loop, to understand not only if they can affect the solution but also how much and in which way. Initially, a mesh convergence analysis was performed on the base domain with both the trimmed and polyhedral models to not only discover the mesh size needed for the correct primary fluid solution but also to investigate the difference in the given solution between the mesh models. As mentioned earlier, the advantage of the trimmed mesh for quasi 2-d problem is given by the possibility of generating a mesh aligned with the domain, which results in the inclusion of only one cell in the z-component of the domain (fig.4.2): the difference between the mesh generated with or without this tool is analyzed as well. It is important to understand that this difference leads to an advantage only if the dimension of the $BaseSize$ is *at least* least equal to that of the domain in the z component; for a lower value, indeed, this advantage cannot be obtained since there is no possibility for generating only one cell and the mesh is therefore not different from the one obtained without this tool. All meshes are generated with the same parameters, with the only change of the $BaseSize$ parameter: from $5mm$ to $2mm$. From the results of the analysis, which are shown in figure 4.3 and 4.3, it is clearly evident that the two mesh models give a slightly different result ($10.3Pa$ for the Poly mesh respect to $9.1Pa$ for the trimmed not aligned) but both reach a convergence almost at 20000 elements, with a base size of $2.5mm$. Focusing on the difference between the aligned and the non-aligned mesh, the most important result is that the convergence of the analysis (which gives the same result: $9.1Pa$ vs $9Pa$) is achieved whit a number of elements that is almost half of the one required by the standard mesh, but with a Base size that is the same ($2.5mm$): from the optimization point of view, this is the best possible result, since it ensures the same geometry definition with a much lower computational cost.



(a) Mesh generated with the alignment tool **(b)** Mesh generated without the alignment tool

Figure 4.2: Difference from a mesh generated with or without the alignment tool

Table 4.1: Mesh convergence analysis - Mass flow splitting test cases

	Pressure Drop [Pa]			Mass Flow		
Base Size [mm]	Poly	Trim	Trim Aligned	Poly	Trim	Trim Aligned
5	12.5	11	10.4	1.4	1.50	1.50
4	11.8	10.5	10	1.47	1.46	1.55
3	10.5	10	9.5	1.6	1.6	1.63
2.5	10.3	9.3	9.1	1.63	1.65	1.66
2	10.3	9.2	9	1.63	1.65	1.66

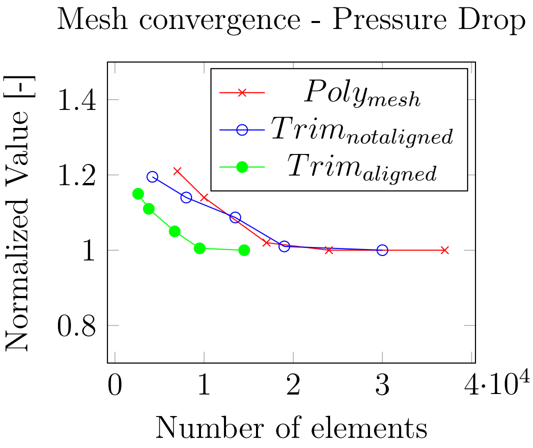


Figure 4.3: Mesh convergence analysis - Pressure Drop. Values are normalized with their converged value

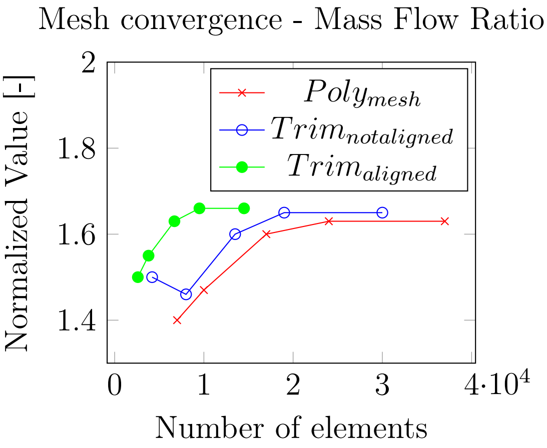
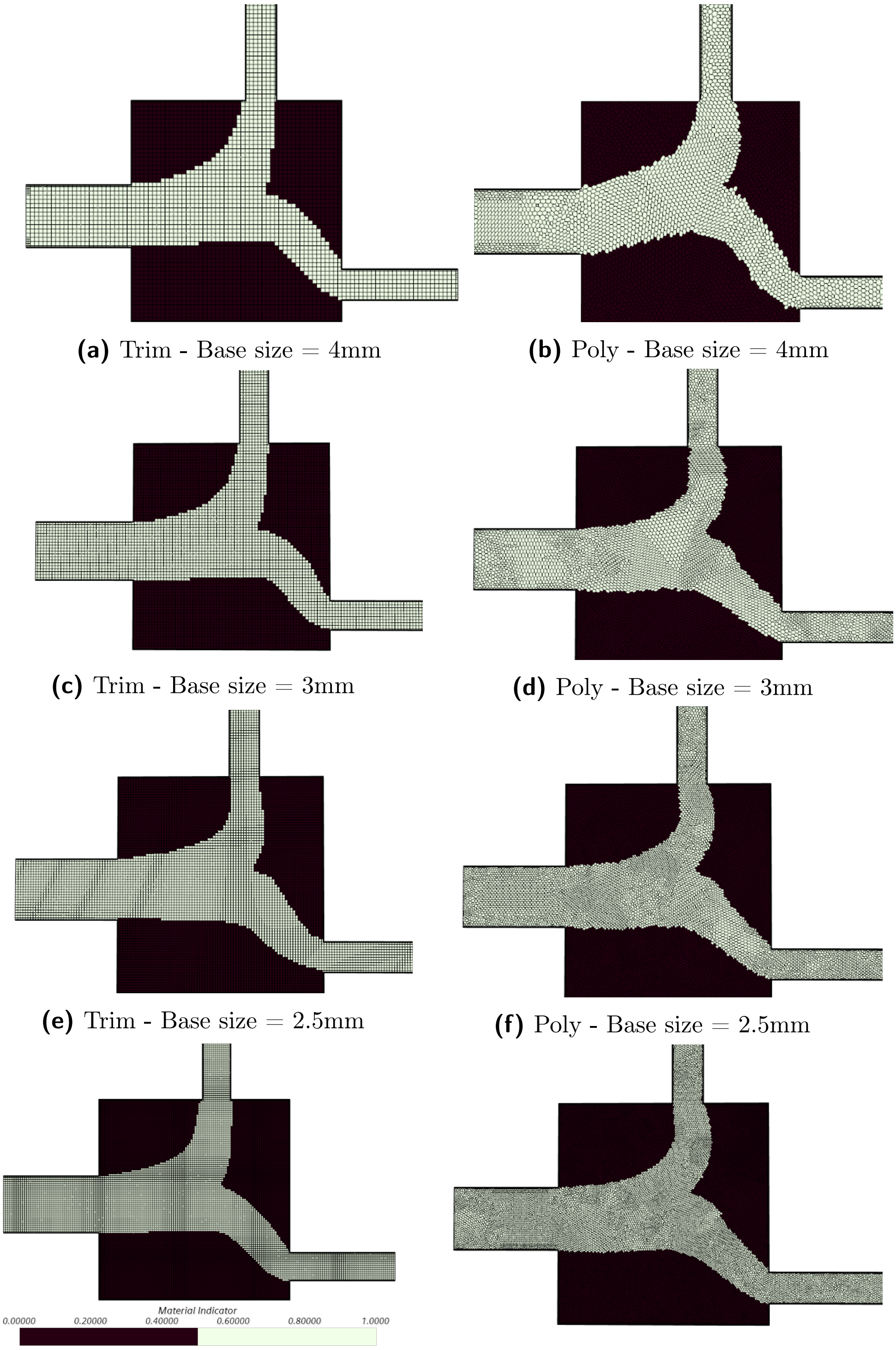


Figure 4.4: Mesh convergence analysis - Mass flow ratio - Mass flow splitting test case.

It is clearly evident that for this case, which is made of a simple plane domain, the trimmed mesh model seems to require less cells for the primal analysis and may be the best choice, but it is important to understand how the mesh model reflects on the optimization. To explore this dependency, the topology optimization was performed with both mesh models, with the aligned one for the case with the trimmed model. By analyzing the results, reported in figure 4.5, it appears that the optimization was able to converge to the same result (from the topology point of view) also with a coarser mesh than the one at convergence, since the geometry is nearly the same aside from small features that would likely be eliminated in the geometry reconstruction process. In addition, it seems that the different mesh model has a slight influence on the topology (the upper boundary of the duct is more curved in the poly mesh model, fig. 4.5h, than the one obtained by the trim model, fig. 4.5g), but this is a small difference that is conceivable since the numerical result is also slightly different between the two models. However, from the topology perspective, the optimization seems to converge to a very similar result, so the choice of the mesh model seems not to influence the topology optimization. Even though this result might seems obvious, it gives the possibility to change the mesh model according to the problem in analysis, without affecting the optimization itself. Concerning the mesh size, the results are very interesting since it seems that it is not mandatory to use a complete converged mesh, but it is possible to slightly reduce the number of the elements and obtaining the same topology: this is clearly an advantage from the computational cost point of view since it is possible to reduce the computation time giving up a more precise geometry.



(a) Trim - Base size = 4mm

(b) Poly - Base size = 4mm

(c) Trim - Base size = 3mm

(d) Poly - Base size = 3mm

(e) Trim - Base size = 2.5mm

(f) Poly - Base size = 2.5mm

(g) Trim - Base size = 2mm

(h) Poly - Base size = 2mm

Figure 4.5: Influence of the mesh model and mesh size on the topology optimization result

If the mesh is judged not fine enough for the required geometry definition but you do not want to refine the entire domain, it is possible to locally increase the mesh resolution. In order to do so, it is possible to generate a mesh control function which reduce the mesh size according with the value of a chosen function. For a topology optimization problem, the driving function can be the gradient of the material indicator function: although it is a continuous function and can assume all the value between 0 and 1, the porous domain is often characterized with a low value of the Material Indicator, while the fluid one have a value that is close to (or equal to) 1. This discontinuity would lead to a high gradient value, which can be used for the refinement of the mesh. Within STAR-CCM+, this can be easily performed with the following control function definition [41] (disclaimer: field functions works with JAVA programming language):

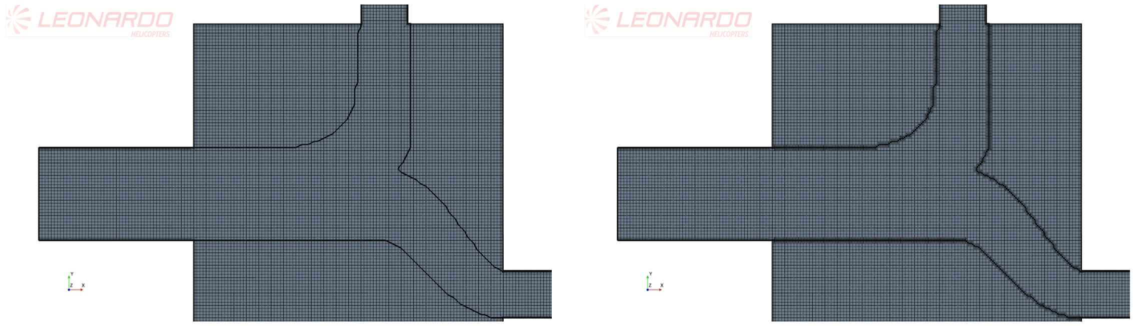
$$AMR = alternateValue(mag(grad(\$MaterialIndicator)) * \$AdaptionCellSize, 0) \tag{4.1}$$

which translates in the following criterion:

$$|\nabla\chi| \cdot \Delta x_{adapt} \begin{cases} < 0.001 & Coarsen \\ > 0.1 & Refine \\ Else & Keep\ the\ actual\ size \end{cases} \tag{4.2}$$

where χ is the Material indicator function and Δx is the size of the cell that, in case, will be modified (see fig. 4.6 for an example of its implementation). To refine the mesh, a child cell is generated from each edge of the selected cell, which together with the others generated from the same cell will replace the parent one. For this reason, this method works perfectly with the trimmed mesher but can generate very bad quality mesh with the polyhedral one. However, there is another major problem with using such a refinement: since the Level set interface moves during the optimization **and** also during the primal solution, it is possible that the mesh is changed multiple times during the Primal analysis iterations. This can be a huge problem from the convergence point of view, since we are using a coupled flow model that, especially for the continuity, is more sensitive to a mesh change during the iterations than a segregated solver. Even though the local refinement can give a more depicted geometry with a low number of cells, to guarantee a correct convergence the required primal iterations may be increased by 2 or 3 times, filling the time gap with the optimization performed with a finer mesh. Moreover, we must remember that the final topology is described by a threshold on the material indicator function which must be undergo post-processing improvements to generate a feasible geometry, so a too high resolution can be unnecessary. For all these reasons, it is preferable to use a finer mesh from the beginning, although this would require higher computational power and time.

When comparing the results of the optimization given by different mesh sizes, it may come naturally to ask why the obtained topology is not so dependent on it. One answer to this question may come from the fact that the error introduced by a coarser mesh, if it is small enough, is overridden by the error introduced by the physical model with which the topology itself is described (hereafter also called "topology physics") and, therefore, the solver does not feel such a difference in the optimization loop. Hence, it becomes important to understand and try to quantify this error, comparing the numerical results obtained with the porous media with



(a) Optimization performed without AMR **(b)** Optimization performed with AMR

Figure 4.6: Difference between the introduction of the Adaptive Mesh model. The outline depicts the level set boundary.

those obtained with a reconstruction of the geometry and a proper boundary layer mesh around it.
The following analysis will investigate the effect of two of the most important parameters: *Penalty* and *StepSize*. In order to eliminate the error introduced by the mesh itself and to focus on the other parameters, the mesh used for these analyses is the finer one analyzed in the convergence analysis, and their parameters are reported in tab 4.2.

Table 4.2: Mesh parameters: Mass flow splitting test case. Percentages refer to the Base Size

Mesh parameters	Base Size [m]	0.002
	Minimum Surface Size [-]	100 %
	Maximum Cell Size [-]	100 %
Prism Layer	Stretching factor [-]	1.5
	Number of layers [-]	3
	Total thickness [m]	0.001
Total Cell Number		**≈ 15500**

4.1.3 Penalty parameter

In this section the influence of the Penalty is analyzed. Recalling the equations 2.56 and 2.57, the Penalty is the parameter that multiplies the i^{th} constraint in both the definition of ψ and in the update of the Lagrange multipliers, thus governing the weight that they have in the solution: the greater is the penalty value, the greater is the influence of the constraints. For this reason, it is interesting to investigate the typical values that it can assumes and how the solution proceeds with values that are too high or too low. In order to perform these analyses, five different penalty values were chosen while all the other parameters are held constant. All the analyses presented here were performed using the automated normalization approach.
From the results of the optimizations the importance and influence of the penalty is clearly visible: with low values (10 and 100) the constraints are hardly satisfied, while

the objective function is the real driving force of the optimization. The final topology is indeed well performing from the pressure point of view, but the constraints are very far from being satisfied, since a greater satisfaction would lead to an increase on the objective function. This is a problem, since we do not want the lowest possible value of the objective function, but we need the lowest possible value **according to the given constraints**. With higher values (5000 and 10000) the constraints are quickly satisfied but than the solution starts to oscillate and cannot converge to the optimal one: the objective function is not perceived correctly by the optimization while the constraints seems to plays a role only when they are violated, while once satisfied seems to be no longer considered (this behaviour is clearly visible in the SVR plot in fig. 4.8). As a result, the solution either failed to converge to an optimal solution (for the case with a penalty of 5000, the optimization was stopped after 830 iterations) or got stuck in a local minimum (such as the case with a penalty value of 10000). It is however interesting to notice that the final topology obtained with a penalty of 5000 seems to be consistent with the optimal one (fig. 4.10), even if the solution did not converged. For what concerns the middle value, i.e. 1000, in this case the optimization suffered from some oscillations at the beginning but managed to converge in the end, with the final solution providing a huge reduction of the pressure drop and the SVR constraint is fully satisfied (as reported in tab 4.3); only the Mass Flow constraint is not completely satisfied but the violation can be acceptable and, in any case, the obtained geometry should be re-meshed to confirm the values.

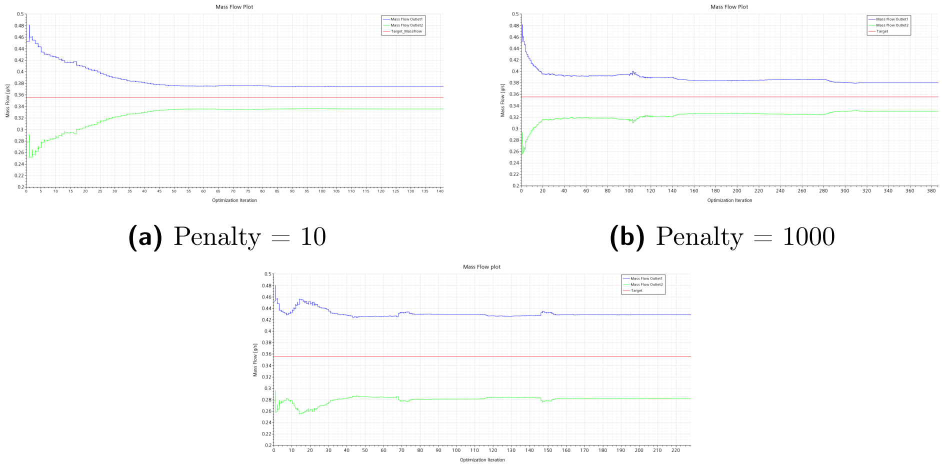
Analysing the numerical results, a strange behaviour of the constraints can be found: with the highest penalty values, which should weight the constraints more, at the end of the simulations the Mass Flow constraint turns out to be more satisfied in the typologies obtained with the lowest value of the penalties. This behaviour is due to the weighting that is unique to all constraints: the mass flow constraint is like overridden by the SVR and thus is not properly felt. On the other way, with a low value of the penalty the greatest satisfaction could be a consequence of the minimization of the pressure drop, which generates such a flow separation.

Numerical results aside, it is important to understand how the optimization behave as it develops; here below are reported just some of the most significant plots that clarify the concepts above discussed, while all the others can be found in Appendix B.

Table 4.3: Effect of the Penalty parameter on the solution. In red are characterized the values obtained in the last iteration of the non converged optimizations

	10	100	1000	5000	10000
Pressure Drop [Pa]	4.1	4.43	5.05	5.05	6.7
Mass Flow Ratio [-]	1.12	1.15	1.15	1.19	5.45
SVR [-]	0.427	0.446	0.497	0.478	0.49
Optimization Iterations	140	150	385	830	230

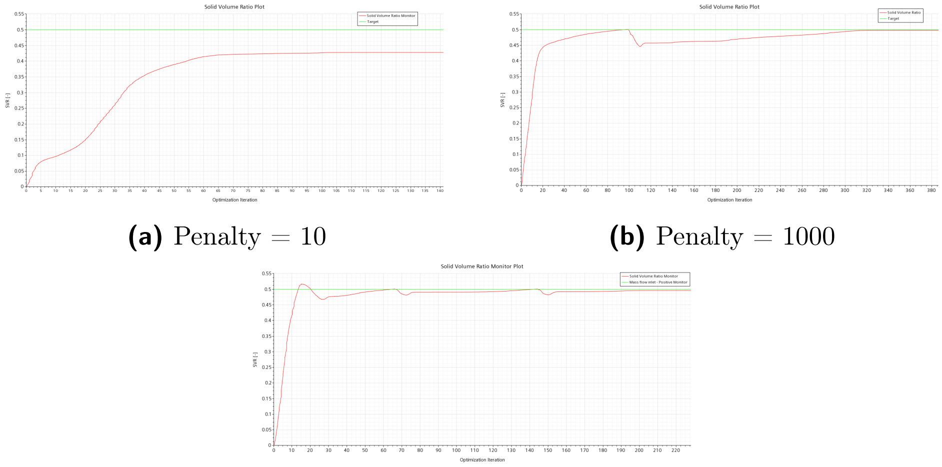
The last important consideration that can be made based on the above analysis is on the iterations required for the convergence of the analysis: while with low values of the penalty this is not a problem and the optimization converges quickly, increasing the penalty also results in an increase in the number of iterations required (which, by the way, do not guarantee the convergence). As we will see later in the chapter, a good convergence can be achieved with a far lower of iterations.



(a) Penalty = 10 **(b)** Penalty = 1000

(c) Penalty = 10000

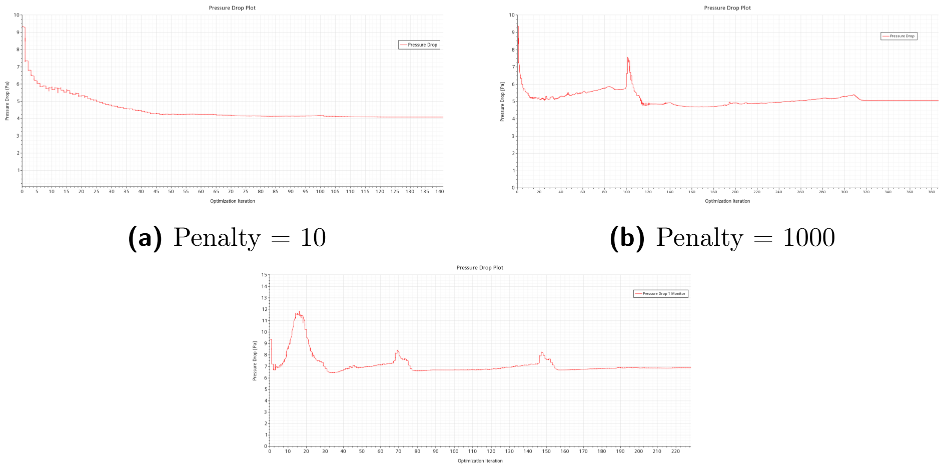
Figure 4.7: Mass Flow behaviour comparison - Penalty analysis



(a) Penalty = 10 **(b)** Penalty = 1000

(c) Penalty = 10000

Figure 4.8: SVR behaviour comparison - Penalty analysis

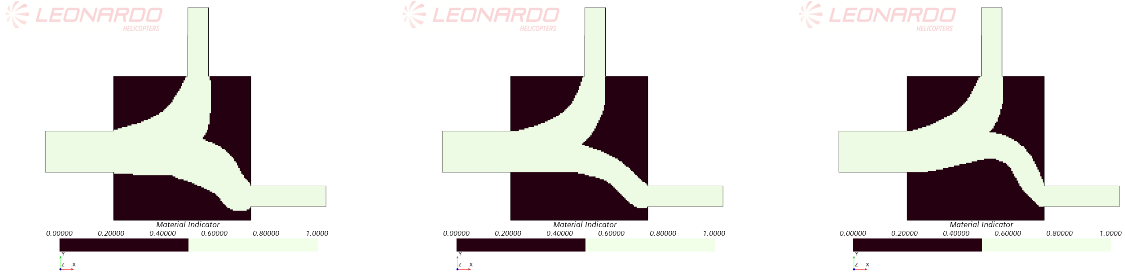


(a) Penalty = 10

(b) Penalty = 1000

(c) Penalty = 10000

Figure 4.9: Pressure Drop behaviour comparison - Penalty analysis



(a) Penalty = 10

(b) Penalty = 1000

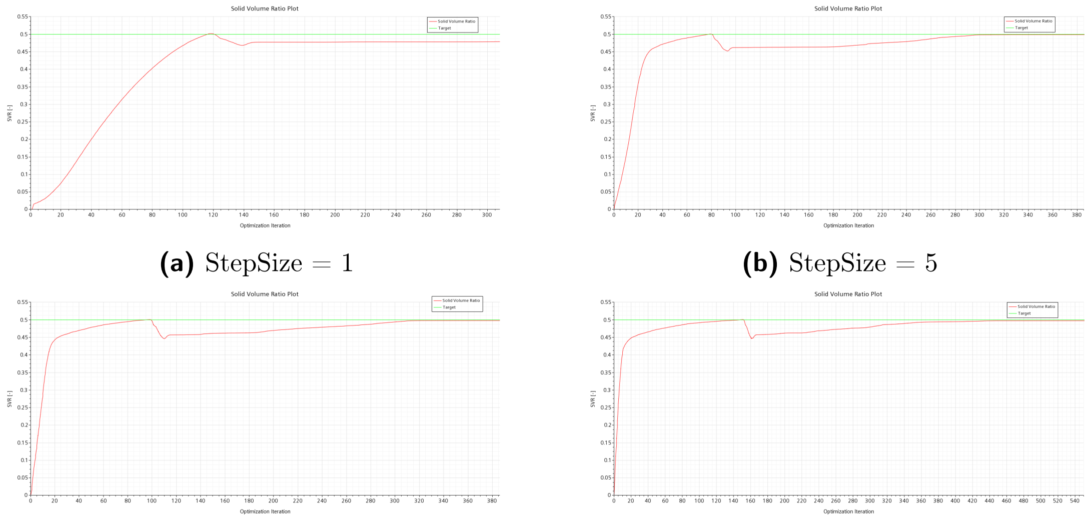
(c) Penalty = 10000

Figure 4.10: Final topology - Penalty analysis

4.1.4 StepSize parameter

Downstream of the results of the analysis on the *Penalty* parameter, it may come natural to wonder whether a different choice in the *Stepsize* could reduce the oscillations generated in the cases with the highest value of *Penalty*. Recalling the equation 2.49, we then wonder what effect the time step τ has on the solution; it is important to remember that we are solving for a steady state solution, so this is a fictitious time step that is needed to simulate the evolution of the LSF and its interface with the fluid domain. Since it has no physical meaning, it can be difficult to guess a good and consistent time step, so the analysis here performed aims to investigate the problems that can be generated when a wrong value is chosen.
Two different analyses were performed: the first one is related to the best solution obtained before, that is the one with a $Penalty = 1000$, while the second one is related to investigate a possible reduction in the oscillations, thus relative to the one with $Penalty = 10000$.
Let us focus on the first case. The analysis performed in the previous sections were obtained with a $StepSize = 10$, so both higher and lower values are tested, starting from 1 up to 20. According to [41] [12], with the ADAM update rule the optimization should be robust with respect to the step size and should be able to converge within a high range of StepSize. From the results reported in fig 4.11 and 4.12, this robustness seems to be confirmed: no major differences in the overall

behavior of the optimization are evident, with the solution approaches the optimum in almost the same way for all the simulations, even with a comparable number of iterations. The only major difference is in the analysis with the lower step size, since the analysis cannot reach the same level of constraint satisfaction. The reason can be found in the reduction of the efficiency of the ADAM update rule, since the momentum given to the interface velocity is not sufficient for overcoming the local minima, since it changes too quickly from one step to the other.



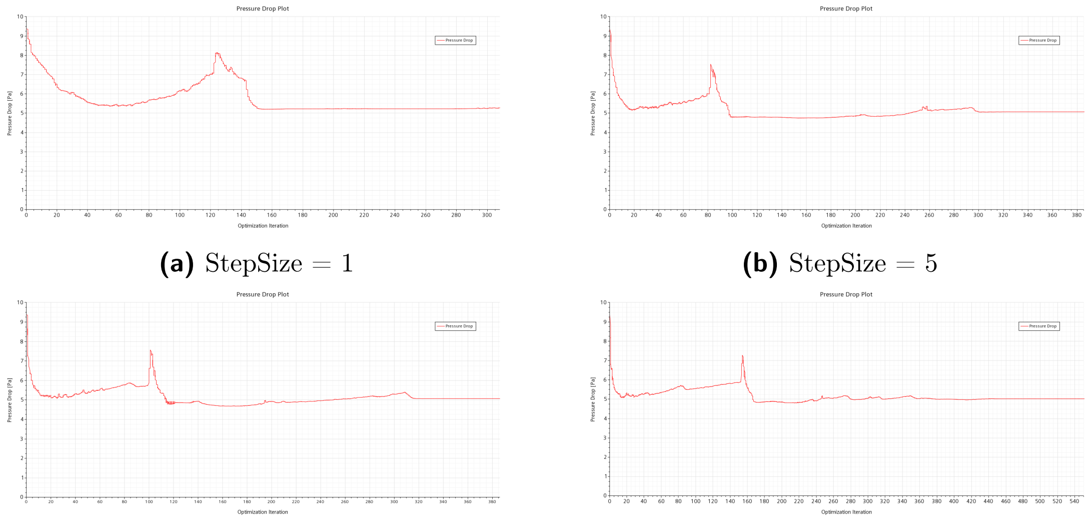
(a) StepSize = 1

(b) StepSize = 5

(c) StepSize = 10

(d) StepSize = 20

Figure 4.11: SVR comparison - Step Size Analysis - Penalty = 1000



(a) StepSize = 1

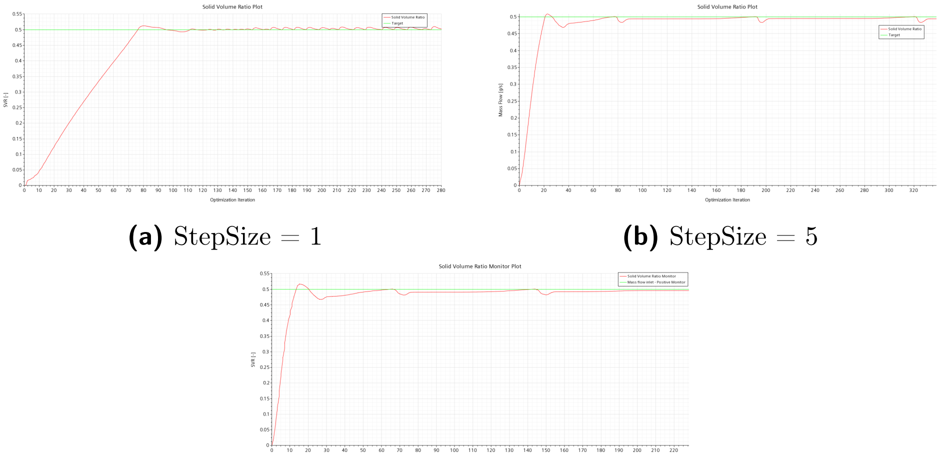
(b) StepSize = 5

(c) StepSize = 10

(d) StepSize = 20

Figure 4.12: Pressure drop comparison - Step Size Analysis - Penalty = 1000

A more interesting analysis is the one performed in the second case, where the reduction of the *StepSize* was intended to reduce the oscillations in the solution, to explore whether such a reduction could help the solution to converge to the optimum. In this case, for obvious reasons, only lower values of the *StepSize* were considered. From the results (fig. 4.13), it is clear that a lower *StepSize* reduced the oscillations, but this did not always turn into an improvement; indeed, with the lowest value the solution completely diverged, confirming that the ADAM update rule does not work well in such a situation. With an intermediate value ($StepSize = 5$), the solution always shows some oscillations but these are around the optimal solution, as confirmed not only by the final topology (4.14) but also from the values of the Pressure Drop and of the Mass flows. Here below are reported some of the most significant plots, while all the others are reported in Appendix B.

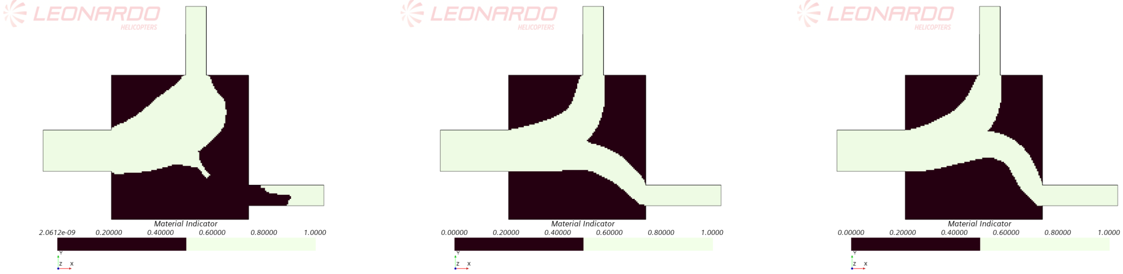


(a) StepSize = 1

(b) StepSize = 5

(c) StepSize = 10

Figure 4.13: SVR behaviour comparison - Step Size Analysis - Penalty = 10000



(a) StepSize = 1

(b) StepSize = 5

(c) StepSize = 10

Figure 4.14: Final Topology comparison - Step Size Analysis - Penalty = 10000

In conclusion, we can summarize the above performed analysis as following:

- The penalty value has a huge influence on the solution and on the convergence, but since it is the same for all the constraints, some of them can be less considered than others;
- The optimization performed with the automatic normalization of the sensitivities does not allow for a different weights between the constraints and most of

the time requires a huge number of iterations to converge to a solution, which not always is the optimum one;

- The StepSize does not influence so much the solution, if the penalty value is correctly set. Otherwise, acting of the StepSize to improve the simulation could not help in the convergence, leading to even more wrong solutions.

4.1.5 Manual Normalization and Sensitivity analysis

The test case analyzed here is a perfect example of what topology optimization for fluid-dynamic problem, until now, has been used for: the solver has a huge domain in which to move for the optimization, with the solid domain that grows to include the fluid one, to create the *best path* for the fluid flow. For this reason, and for the fact that is a very simple domain, is not necessary to pay too much attention in the definition of the objective function and constraints, since the optimization, if set up correctly, manages to improve and obtain the optimal topology also if the sensitivities of the constraints are different one from the others, leaving the solver to automatically normalize them (as in eq. 2.58). Thus, this is not the best case to better understand the differences in the results between the automatic and the manual normalization; an in-dept analysis is done with the test case described in chapter 4.2. On the other hand, precisely for its simplicity, it is the best case for understanding how these functions can be manually defined and normalized, to replicate this critical procedure for more complex geometries. With this approach, the value of $Penalty$ is no longer used to weigh the constraints against the objective function, since this is done manually; its value must therefore be small, since a too high one would lead (again) to an overconstrained optimization.Regarding the $StepSize$, as we have seen, the solution is not much affected by its value, as long as it is neither too small nor too high. For these simulations and for all the others analyzed with the manual normalization the chosen values are: $Penalty = 10$, $StepSize = 10$.
Analyzing the definition of the constraints, we notice that it is necessary to identify three parameters for the constraint functions, while for the objective function it is necessary only the normalization value. If, for simplicity, we assume that the normalization value is the order of magnitude of the physical quantity under consideration, the objective function is fully defined while the unknowns for each constraint are reduced to only two (or even one in this test case). It is important to remember that this reasoning does not apply to the SVR constraint, since it depends solely on the mesh size; its normalization will therefore have to be done a posteriori, based on the values obtained from the physical constraints.
The definition of the unknowns is therefore the core of the analysis, on which we focus for the definition of the developed procedure: starting from the fluid dynamic analysis, we evaluate the objective function and the imposed constraints, analyzing in particular the distance of these constraints from the imposed value. The greater the distance, the greater the sensitivity of these constraints. In general, it is good to remember that the sensitivities of the constraints should be smaller than that of the objective function, unless the constraint is largely unsatisfied. In the present case, the domain thus defined results in a $MassFlowRatio = 1.6$, so the constraint is unsatisfied but not largely so. The value of the sensitivity of this constraint must therefore be lower than that of the objective function, but still be considered:

the most consistent choice is therefore a sensitivity that has (as maximum absolute value) an order of magnitude lower than the maximum absolute value of the sensitivity of the objective function. In order to obtain this result, a cycle like the following can be introduced:

- Based on the result of the analysis and the distance of the constraints from the limit values, choose the order of magnitude of difference between the sensitivities, for example as the ratio of the maximum values.
- Guessing an initial value for each unknown;
- Perform the Sensitivity Analysis with the guessed values and evaluate the above defined ratio;
- If the ratio is greater than the desired one, increase the unknowns values, otherwise decrease;
- Repeat the loop until convergence.

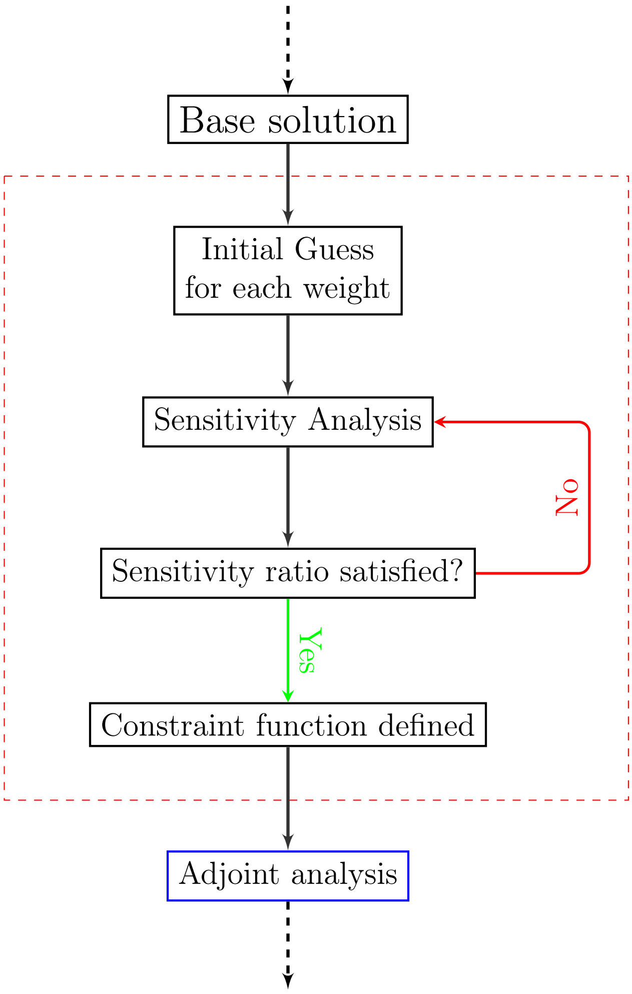

Base solution
Initial Guess for each weight
Sensitivity Analysis
No
Sensitivity ratio satisfied?
Yes
Constraint function defined
Adjoint analysis

Figure 4.15: Sensitivity Loop inside the general TO Loop

In order to better understand the problems that can arise from an incorrect definition of the functions, let's identify two different optimizations performed on the same meshed domain (to reduce the computational time, only the trimmed mesh with the parameters described in tab. 4.2 was used). Both cases are performed with a manual scaling of the constraints and with a definition of them as depicted in chapter 3.4.2; while the first one, called *Case A*, has a proper scaling (with an order of magnitude lower than the sensitivity of the objective function), the second one, *Case B*, has an incorrect normalization (same order of magnitude).
Since we need an equal division of the mass flow, we should re-define the report characterization:

$$\dot{M}_{CF} = \alpha_{\dot{m}} \cdot \left[\frac{\dot{m}_{out1} - \dot{m}_{out2}}{\dot{m}_{norm}}\right]^2 \overset{\dot{m}_{out1}=\dot{m}_{in}-\dot{m}_{out2}}{=} \alpha_{\dot{m}} \cdot \left[\frac{2 * \dot{m}_{out2} - \dot{m}_{in}}{\dot{m}_{norm}}\right]^2 \tag{4.3}$$

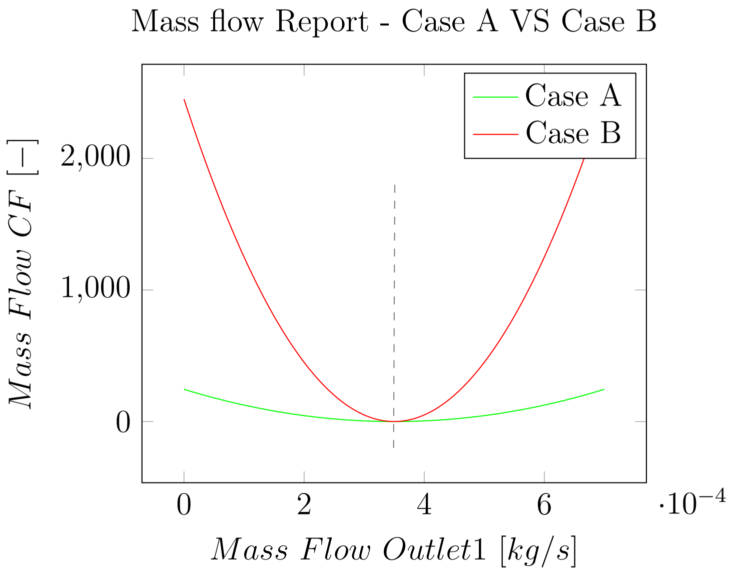
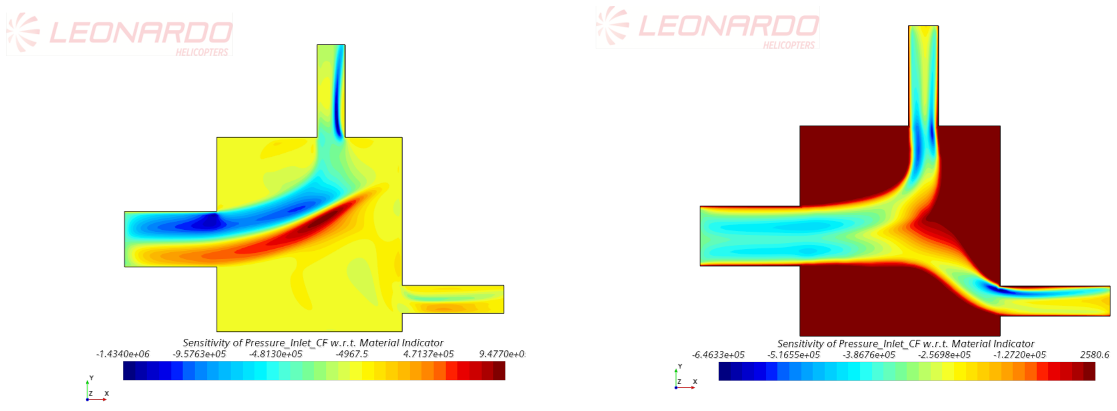


Figure 4.16: Mass flow Report Normalization - Case A VS Case B. Both reports are defined with $\dot{m}_{norm} = 0.001kg/s$, but they differ for the normalization: $\alpha_{\dot{m}_A} = 500$ while $\alpha_{\dot{m}_B} = 5000$.

Figure 4.16 shows the differences between the Mass Flow constraint function of Case A and B are illustrated: even thought the values of the constraints may seem extremely high, it is important to analyze the effect that these values have on the sensitivities. It is crucial to understand that when we talk about *Normalization* of the Sensitivity with respect to Material Indicator, we do not mean that the value should be near 1 but we mean that the sensitivity value should represent the distance between the actual value of the constraint and the desired one, assuming that similar distances equate to similar sensitivity values. As we can see in fig. 4.16, the two functions differ by an order of magnitude which translates (at the beginning, with the sensitivities performed on the exactly same domain) in sensitivity values that differ by an order of magnitude (fig. 4.26a and 4.26b). As the optimization proceeds, the sensitivity of *Case A* succeeds in leading it to the correct solution because it gives an indication of the "ideal path", without overcoming or contrasting the sensitivity of the objective function (fig. 4.17); on the contrary, the one of *Case B* is too high and it drives the optimization to the point of causing the solution to diverge in order to have an equal split of the mass fluxes (making them even almost zero) as can be seen from fig. 4.19b.

While the value of the constraint's function is expected to decrease down to zero during the optimization, and so their sensitivities, the value of the objective function is unknown and it can assume every possible value as the loop proceeds. Therefore, we do not expect the sensitivity of the objective function to decrease to zero, but we do expect it to "eliminate" the region of the domain that generates positive values (if we are performing a minimization of the objective function, vice-versa for a maximization), as it is clearly shown in fig. 4.17b where the regions for which the sensitivity is positive (dark red) are the ones with a material indicator value of 0. If we compare the sensitivity values of the pressure drop at the 66^{th} iteration with the ones of the mass flow constraint (fig 4.17b vs fig. 4.26g), it is clear that a trade off has been reached, since there are cells that may reduce the pressure but increase the mass flow ratio: if the sensitivities on that cell tell the solver that a change in the material indicator would produce an improvement, for example for the objective function, but a huge reduction in the satisfaction of the constraint, that cell is not modified, since there is not a clear advantage; this is the concept behind the *Sensitivity Normalization.* Based on these considerations, it is evident that there is not an unique definition of the functions, provided that the sensitivities are correctly scaled the one with the others.

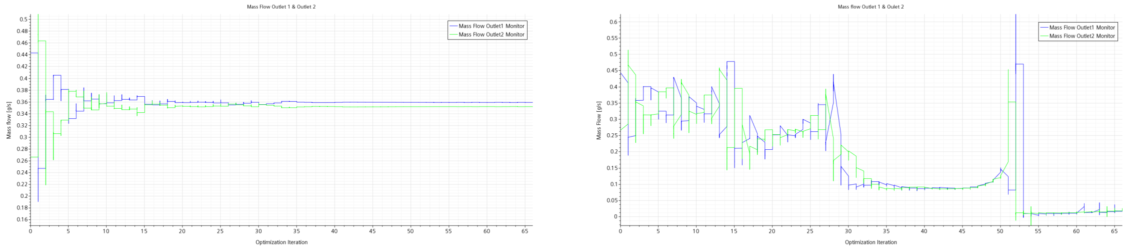
The sensitivity of the SVR deserves a separate discussion, since it is a function of the mesh size only: if the mesh does not change during the optimization (see. 4.1.2.1), the sensitivity remains constant during the entire loop, so its normalization is not related to the others based on physical quantities (Mass flow, pressure drop, velocity, etc.) and does not need to assume comparable values with the others (its value is of orders of magnitude lower then the others for obvious reasons). The value of its sensitivity is used for the solver to figure out how much solid is required to satisfy the constraint and the greater is the sensitivity the greater change on the SVR constraint is obtained if the Material indicator is changed on that cell. Although its value is not comparable to the others, a too low or too high value may lead to non satisfaction or instability, but it is impossible to establish this a priori: this consideration can be made by analyzing the convergence of the optimization. This is not a contradiction since this constraint is only an indication for the solver and an aid for the level set function to move within the domain.



(a) **(b)**

Figure 4.17: Sensitivity of the Pressure Drop w.r.t the material indicator at iteration 1 (a) and 66 (b) for Case A

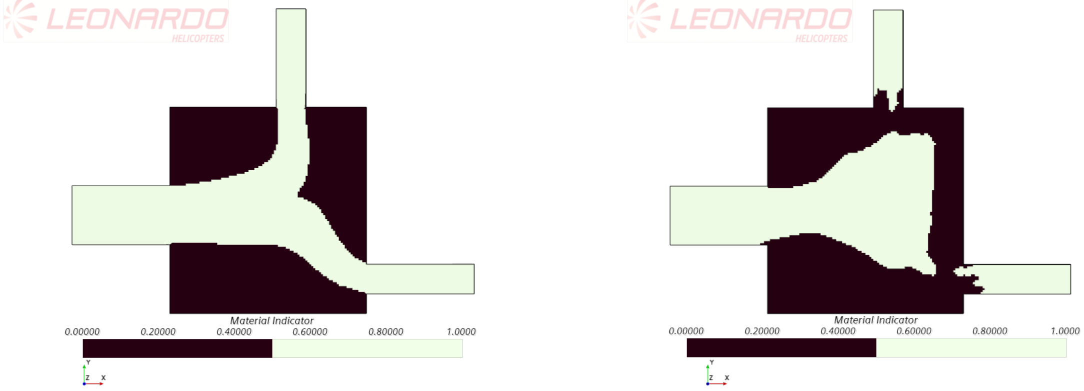
Analyzing the values of the mass flow as the optimization proceeds (fig. 4.18), it appears clear that Case B overestimates the importance of the constraint: although this may appear to satisfy it more stringently, in reality the optimization is completely blind to the objective function and physics of the problem; the consequence is that the mass flows of the outlets are equal just because they tend to zero, which is not what we wanted.



(a) **(b)**

Figure 4.18: Mass Flow behaviour during the optimization process: while Case A (a) managed to satisfy the constraint, Case B (b) diverged.

Case B highlights another aspect of the optimization and a problem that arises from the generation of porous media in the domain: with this characterization of the solid domain, the convergence of the continuity may suffer and the simulation may not converge, but it may not notice it on its own and will keep going through the optimization without stopping. The result of the optimization is completely wrong and this is noticeable since the first 10-20 iterations (this optimization was performed up to the 66th iteration just to compare it with the correct one, which stopped with the previously defined stopping criteria): the tendency to equalize the mass flow leads the solver to reduce their value down to zero, which results in a very high increase of the pressure drop of the system, but which is completely acceptable for the solver from the physical point of view. Even if the optimization would completely close the outlet sections, we must remember that the solid domain is fictitious and is actually a fluid-porous domain, so there is a fluid transition (the Brinkman penalty forces the velocity to zero but is never absolutely zero) that prevents the solver from diverging during the Primal Analysis solution. The result is a geometry through which the mass flow constraint is fully satisfied, with a ratio of 1.07, but both the mass flows are nearly 0 and the topology is absolutely non physical: the optimization has failed and cannot be accepted (4.19b). Does this mean that the iterations of the Primal Analysis may be increased? Perhaps, but not necessarily, for two reasons: first, as shown by *Case A* the optimization can be easily performed with the number of iterations chosen that are absolutely sufficient for the solver; second, the number of iterations required may be many times higher than those currently used, therefore the computational cost required will increase without any benefit. A test with 500 primal iterations (instead of 150) was performed to investigate this possibility, but the result was absolutely the same. For computational time reasons, no other tests were performed.



(a) Case A - correct optimization

(b) case B - wrong optimization

Figure 4.19: Optimization results - Mass flow splitting test case.

From these analyses the huge advantages that are introduced in the optimization with the manual normalization of the sensitivities are clear: not only the iterations required for the convergence are smaller (66 vs >300), but it is possible to give different weights at each constraint, which translates in a more consistent and correct optimization. The correct normalization on the sensitivity is made with an iterative process acting on the parameters that make up the new definition of the constraints functions, based on the constraint satisfaction at the first iteration.
An interesting analysis can be done by comparing the trend of the objective function during the optimization with the one of the constraints and with the one that they have with the automatic normalization (previously shown): from fig. 4.20 is reported the trend of the objective function as the optimization proceeds. It is clear that in the first part of the optimization the constraints (SVR and Mass flow ratio) are far from the satisfaction and therefore are predominant: the optimization is driven by them. As the optimization proceeds, the constraints tents to be satisfied more and more while the objective function starts to dominate in driving the Level set function. In the last iterations the solver reaches a convergence since a a tradeoff is achieved between satisfying the constraints and minimizing the objective function. The topology in the last iterations does not change drastically, but small changes are allowed since, as stated before, it is impossible for the boundary of the level set to be totally fixed. This leads to the fact the final value of the constraints may not be unique but may oscillates, therefore a definition of the stopping criteria as defined

in chapter 3.4.4 is necessary. It is important to distinguish between the oscillations introduced by the automatic normalization approach and the ones here illustrated: while in the first case the solution changed drastically during the oscillations, here the constraints are almost constant and the objective function oscillates around a single solution, which could probably be achieved if the time step would be iteratively reduced.

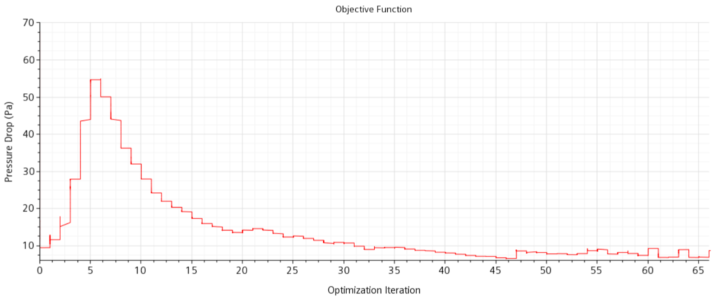
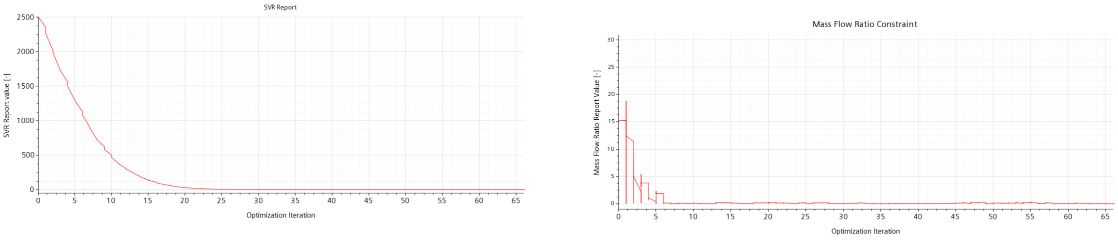


Figure 4.20: Pressure Drop behaviour during the optimization loop - Mass flow splitting test case



(a) SVR report convergence. $\alpha_{svr} = 100$, $\beta_{svr} = 0.1$, $SVR_{norm} = 0.01$, $SVR_{min} = 0.5$

(b) MF Ratio report convergence. $\alpha_{\dot{m}} = 500$, $\dot{m}_{norm} = 0.001 kg/s$

4.1.6 Optimization results and validations

With the acquired knowledge from the previous analysis it is now possible to investigate the error introduced by the topology model. As a further investigation of the mesh model influence of the mesh model on the optimization result, this analysis was performed both with the *Trim* and the *Poly* mesh (with the same parameters of tab. 4.2), since the final topology between the two models is very similar but not exactly the same. This is a simple 2-D test case: the geometry reconstruction was therefore trivial and based on a simple re-design of the domain (fig. 4.22); moreover, the mesh parameters used for the validation are exactly the same of the one used for the optimization loop. In table 4.4 it is possible to compare the results of the two parameters that were considered in the optimization process: the mass flow ratio between the two outlets and the pressure drop of the system.

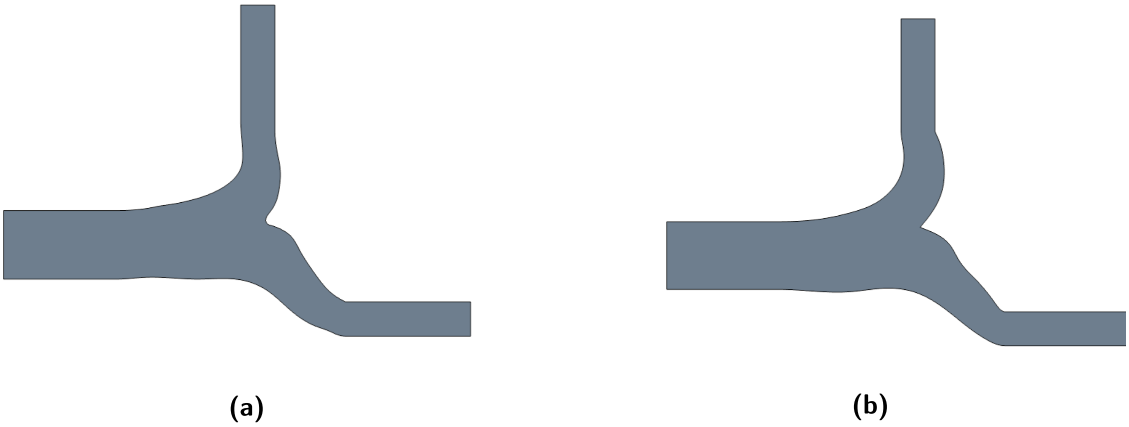


Figure 4.22: Optimized geometry reconstruction - Trim (a) VS Poly (b) mesh models

Table 4.4: Comparison of the results of the optimization between the mesh models and the reconstructed (smoothed) geometry

Optimization mesh		Mass flow ratio[-]	Pressure Drop[Pa]
Trim mesh	Topology physics	1.02	6.85
	Reconstructed	1.04	2.0
	Reconstructed (Poly)	1.03	2.0
Poly mesh	Topology physics	1.05	9.4
	Reconstructed	1.07	2.5
	Reconstructed (Trim)	1.09	2.4

Both the analysis were performed with both the models, to remove the dependence of the result of the reconstructed geometry from the mesh model itself. When analyzing the results of the reconstructed geometry, the topology found with the trimmed mesh model seems to perform slightly better than the one obtained with the polyhedral model, maybe for the greater curvature obtained in the upper section forcing the fluid to a more pronounced curvature, or for the larger stagnation zone near the ducts separation (fig. 4.23); the differences however are very small and cannot provide a definite answer. The topology, again, appear to be unaffected, which provide confidence that the best mesh can be chosen based on the problem and not forced by the need for optimization.

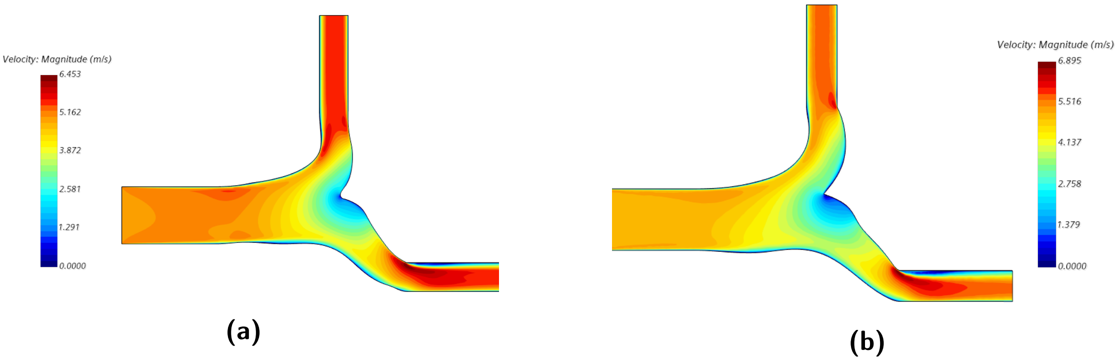
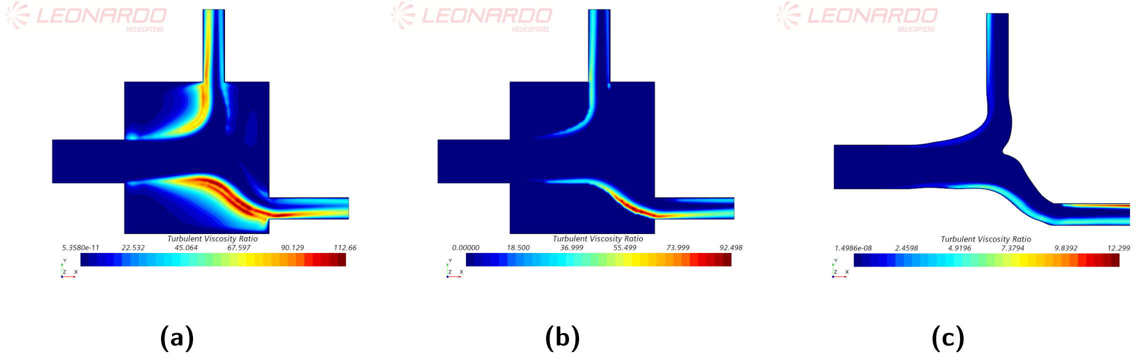


Figure 4.23: Difference in the velocity field for the reconstructed geometry - Trim (a) VS poly (b)

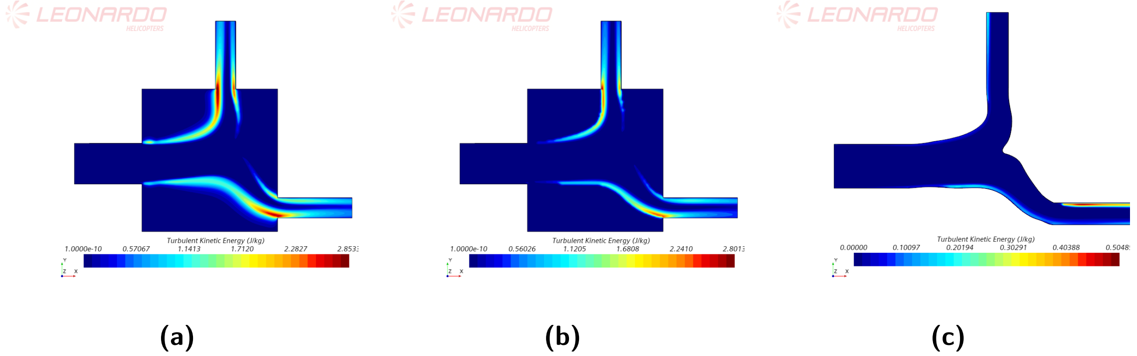
An important consideration must be made for the error introduced by the *topology physics*, which is quiet evident that it is not negligible, especially, in this case, for the computation of the pressure drop: in the trimmed model the value is 3 times higher and for the polyhedral near 4 times higher. This represents a huge problem for the correct setting of the optimization, especially for the value of the constraints that need to be imposed and therefore it is fundamental to look for corrections that help in its reduction. Since we are considering an optimization problem, the absolute value of the objective function is not predefined and is not as fundamental as the one of the constraints; the rebuilt geometry can indeed give us its real value. The problem arise when considering the constraints definition: not tacking into account the error in the computation of the physical quantities can lead to an incorrect value of the constraints, which would result in a over-constrained optimization (not capable of improving) or in a under-constrained one (the rebuilt geometry does not satisfy the constraint). In this test case, for example, if we had wanted to perform an optimization with the pressure drop as a constraint, imposing a maximum value of $3Pa$ could have led to instability in the solution or a non-satisfaction of the constraint, even though the re-constructed model ensures that it is possible to obtain such a pressure drop. Therefore, it is crucial to try to reduce as much as possible such an error or, where not possible, to figure out how to predict its influence on the solution. As mentioned in chapter 3.3, a major source of error is the presence of non-zero turbulence within the porous media. Even tough in the cells characterized with a Material Indicator different from 1 the velocity is forced to zero, thanks to the Brinkman penalization model (eq. 2.46), its value is never absolutely zero but is some order of magnitude smaller than the velocity scale of the problem. The consequence is that a turbulence may develop or propagate inside the porous media and influence the behaviour of the external fluid.

Let's consider the figures 4.24 and 4.25: it is clearly evident that there is a huge turbulence production in the cells near the Level-set interface and a not negligible one inside the porous media. Compared to the real geometry and a more correct turbulence modeling, such a production is not physic, as it is not the turbulence viscosity field. One of the possible and simplest solution identified is to destroy all of the turbulence present in those cells with a Material indicator value < 0.5, thanks



(a) (b) (c)

Figure 4.24: Turbulence viscosity ratio field comparison: without TVR (a), with TVR (b) and real geometry (c)



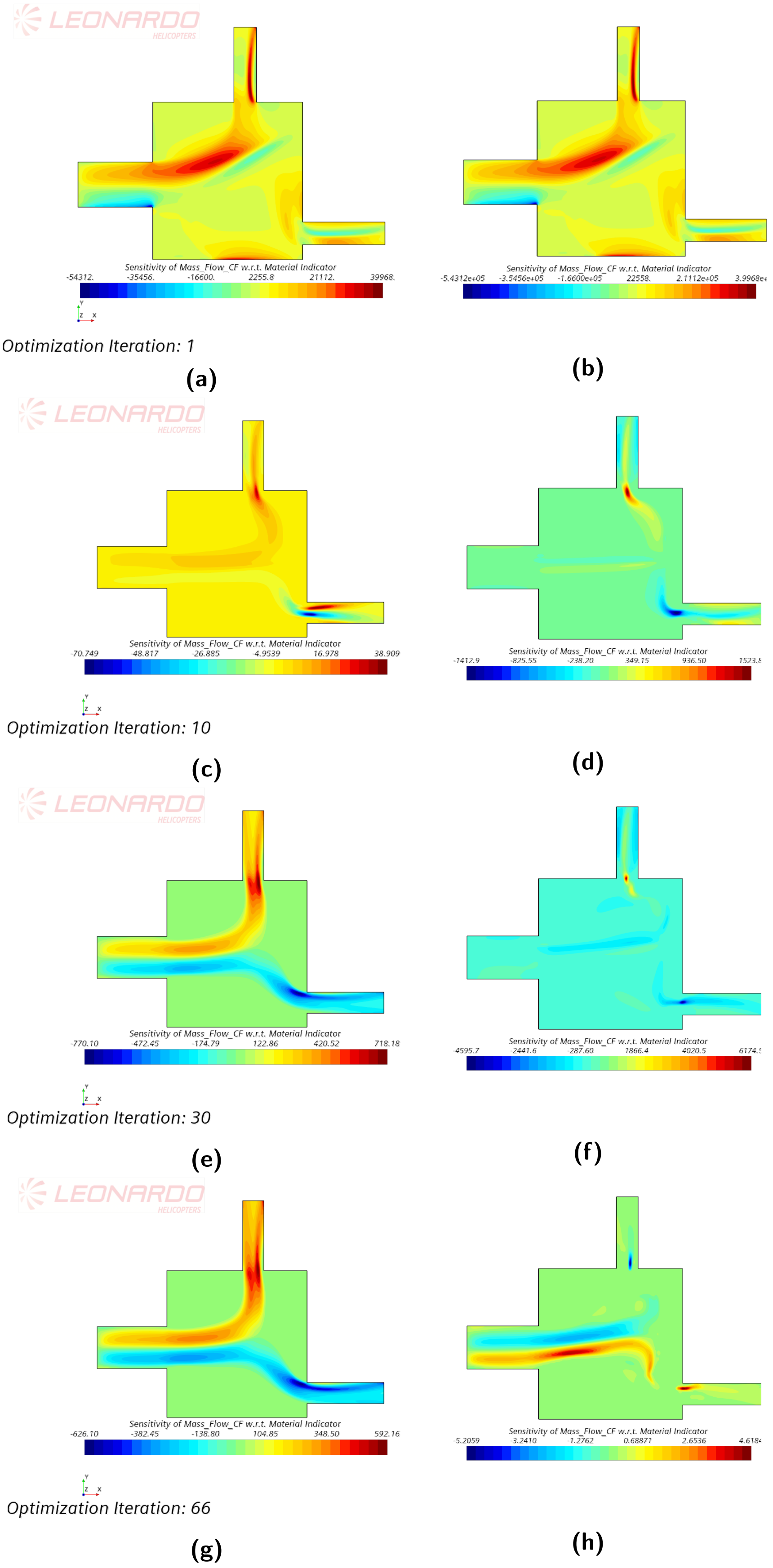
(a) (b) (c)

Figure 4.25: Turbulence kinetic energy field comparison: without TVR (a), with TVR (b) and real geometry (c)

to the TVRScale function described in fig. 3.3. Its effect on this simple test case is shown in figure 4.5, where one can appreciate not only the brutal destruction of turbulence but also the beneficial effect it has on its generic field. From the numerical point of view, its effect and the improvement obtained with the TVRScale function are reported in tab 4.5. Further consideration on this error and its influence on the solution are made for the next optimizations (chapter 4.2 and chapter 5).

Table 4.5: Comparison of the results predicted by the topology optimization with or without the TVRScale function

	Mass Flow Ratio [-]	Pressure Drop [Pa]
TVR_{OFF}	1.01	9.16
TVR_{ON}	1.02	6.85
Real Geometry	1.035	4.0
Smooth Geometry	1.04	2.0



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Figure 4.26: Difference between the values of the Sensitivity of the Mass flow w.r.t the material indicator - Case A VS Case B.

4.2 Application on a real 3D case: Inlet Particle Separator

In this section the new procedure for the Adjoint Topology Optimization is applied to a real 3D case that however represents a "standard" application case for fluid dynamics topology optimization: creating the best geometry for a duct system that would reduce the system pressure drop (or increase the mass flow). The base geometry is the one of the Inlet Particle Separator duct with the one of the Starter of the Engine of the NGCTR (see. cap 5 for in depth analysis). The goal of this analysis is to apply the knowledge gained from the 2-D case to a 3-D one, to analyze the possible differences. An in-depth discussion on the difference between the optimization performed with the automatic sensitivity normalization versus the new developed one is presented and the differences are analyzed, along with an analysis of the error introduced by the Topology Physics

4.2.1 Problem introduction and geometry definition

The Inlet Particle Separator (IPS) is a fundamental component in the engine system of an helicopter (or a tiltrotor) since it prevents solid particles (sand, dust, hailstorm or just in heavy rain condition) from entering the engine stages and contaminating the engine itself. These could damage the compressor or turbine blades, block the cooling systems or it could also generate vibrations and fatigue problems, affecting safety and engine life. A crucial parameter that governs the proper operation of this component is the pressure drop inside the duct, since the less the pressure the easier it is to suck in particles. The IPS duct in the CT7-2E1 engine is connected to the one of the engine starter (fig. 4.27) which also has some requirements and constraints for correct performances, like the mass flow or the pressure drop felt by the starter itself. The connection between them is therefore a source of turbulence and pressure drop, and it is precisely for this reason that it will be the area where optimization will be performed. According to the specifications given, the IPS and the Starter systems are characterized with the following conditions:

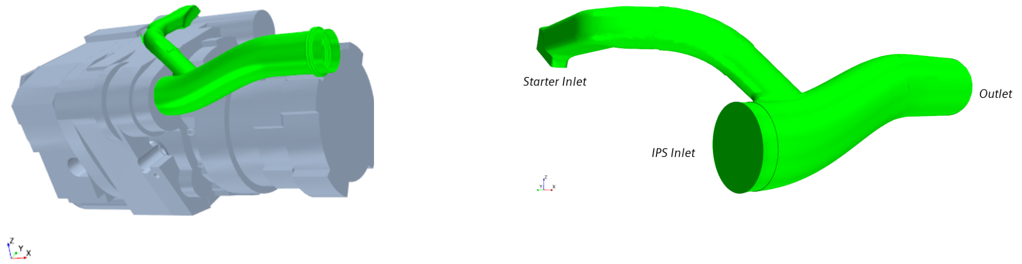


Figure 4.27: Reference geometry of the IPS and starter ducts (in green)

- $\frac{\dot{m}_{IPS}}{\dot{m}_{Starter}} = 7.7$;
- Maximum IPS pressure increment: $\Delta P_{max} = +35$ %;

- Starter pressure target = 25 %

with the given boundary conditions, on the main pipe we ended up with a $Re \approx 7 \cdot 10^5$.

Based on these conditions, the base geometry was simulated to quantify the pressure at the inlet sections and verify if they meet the imposed constraints. For obvious reasons, the simulation was performed only on the IPS and Starter ducts without all other engine components. For obvious reasons, the simulation was performed only on the IPS and Starter ducts without all the other engine components.
The topology optimization was performed after a mesh convergence analysis (fig. 4.28) with a trimmed mesh model: this choice was made with the topology optimization process in mind, since the trimmed mesh a works slightly better, as seen in the previous chapter it seems. To keep the number of cells as low as possible but to correctly capture the flow behavior, a refinement was introduced in the connection area between the IPS and the Starter ducts, with a mesh size that is equal to 25% of the Base size. The parameters that guarantee a converged mesh are reported in tab 4.6.

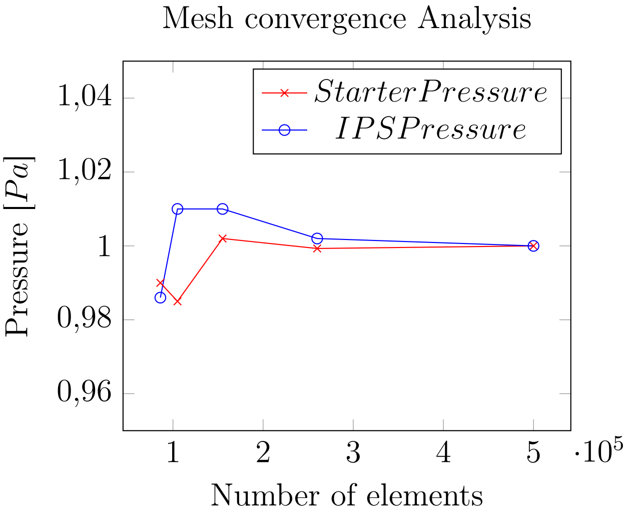


Figure 4.28: Mesh convergence analysis for the base geometry of the IPS test case. Values are normalized with the converged value

Table 4.6: Mesh parameters: IPS Test case. Percentage referees to the Base Size

Mesh parameters	Base Size [m]	$1.5 \cdot 10^{-2}$
	Minimum Surface Size [-]	25 %
	Maximum Cell Size [-]	100 %
	Refinement Cell Size [-]	25 %
Prism Layer	Stretching factor [-]	1.5
	Number of layers [-]	5
	First Layer Tickness [m]	$3.5 \cdot 10^{-4}$
Total Cell Number		**≈ 250000**

To get an initial indication and idea of where to work to minimize system pressure loss, we analyzed the flow behavior, especially in the area near the junction of the two ducts. As reported in figure 4.29, it is possible to understand why there is such a pressure drop in the Starter: the connection between the ducts is indeed too sharp and too direct, which generates a huge flow separation and a high turbulent flow and leading to high pressure losses (fig. 4.30).

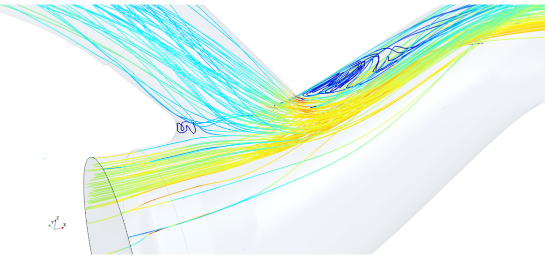


Figure 4.29: Streamlines in the connection zone. A huge circulation is generated after the junction. Blue indicates lower values

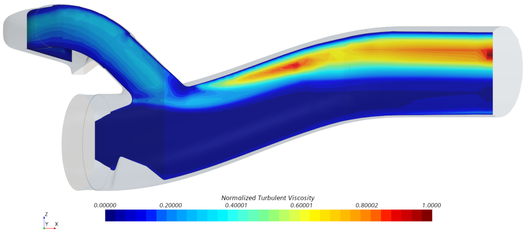


Figure 4.30: Section of the turbulent viscosity ratio. Values are normalized according to the maximum value

Based on these considerations, one might think that the simplest solution to the pressure problem would be to blunt the sharp edge connecting the two ducts (where flow acceleration is greatest and separation occurs) to help the flow not separate, or at least to reduce the separation generated. However, a sensitivity analysis performed on the geometry seems to indicate that there is another opportunity or, rather, another change in the geometry that may make a difference: according to the sensitivities of the starter pressure with respect to the position of the boundaries, in addition to the sharp edge smoothing identified earlier, it appears that reducing the cross section of the IPS duct prior to the connection would help in flow mixing. The final shape appears to be a more rounded starter duct in the connection and a narrower cross-section in the IPS, which would generate an acceleration of flow and, therefore, a lower pressure that would help in the intake of flow from the starter inlet (fig. 4.31).

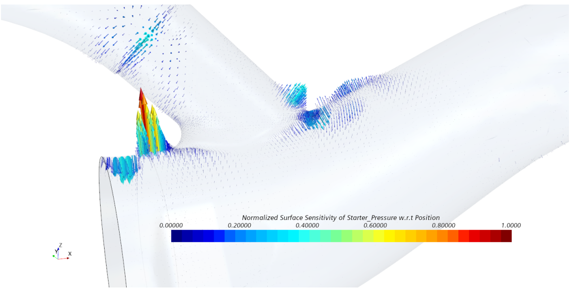
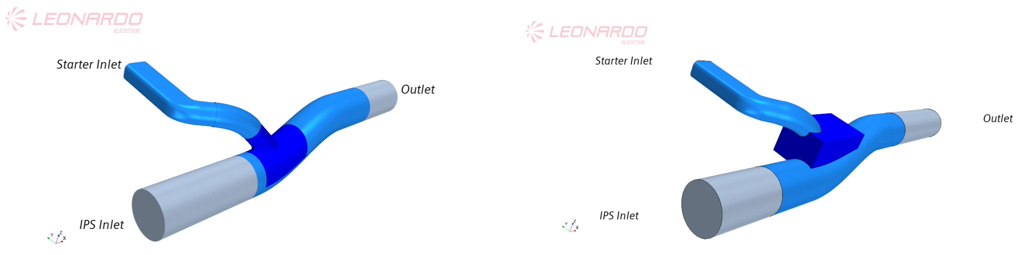

Normalized Surface Sensitivity of Starter_Pressure w.r.t Position
0.00000
0.20000
0.40000
0.60000
0.80000
1.0000

Figure 4.31: Surface sensitivity of the boundaries with respect to the starter pressure. Arrows point in the direction of increasing the objective function, so to reduce the pressure an opposite modification is required.

In this context the topology optimization for this test case is performed, with an idea of the boundaries on which the solver would work on.

To improve the convergence of the optimization, a slight modification was made in

the geometry, since the inlet section of the IPS is very close to the duct junction. Generating a porous medium so close to the IPS inlet boundary condition would unrealistically increase the pressure, as the influence of the boundary condition would corrupt the final results. The inlet section is therefore moved away from the duct junction with a simple straight duct of constant cross section, which would generate a negligible pressure drop. Similar considerations were made for the outlet section, which was also moved away from the junction region with a simple straight duct. Finally, another modification was made on the starter inlet: to reduce the number of cells, the beginning of the domain was cut off and the inlet section was imposed on the constant section straight duct before the junction. Obviously all these changes affect the final result of the analysis, but for our purpose (testing the optimization on a 3D geometry) these changes are acceptable. Regarding the topological region, some external constraints have to be considered in the definition of the modifiable region: these ducts have to remain inside the engine bay and are embedded in a larger piping system, so the possibilities of modifications are limited. In order to analyze different possibilities, two different optimizations were therefore proposed: the first one was done considering that the current geometry was the "largest possible" and therefore the solver could only "dig" within the domain. Based on the changes it seems to want to introduce in the domain, only the boundaries close to the junction are modifiable (*Case 1*); moreover, since the IPS section has external assembly constraints, the TO-Region does not reach its section. This case was also used to clarify the differences between the automatic sensitivity normalization result and the manual one.
The second optimization was performed in the opposite way, i.e., the goal was to maximize the mass flow of the starter given the total pressure at its inlet section. To give more freedom to the solver, a small box including the real connection was defined and a high solid volume ratio requirement was imposed in order to obtain a topology that is not too much bulkier than the real one (*Case 2*). The two final geometries that will undergo optimization are shown in figure 4.32a and 4.32b.



(a) Case 1

(b) Case 2

Figure 4.32: Geometry for the topology optimization of the IPS case. Light blue is used to highlight the TO-Region but only the dark blue boundaries are editable

In order to have a correct reference value for the new domain with respect to the base one, the initial geometry was modified with the above corrections and the pressures recomputed. While the IPS pressure did not change significantly (as expected) the Starter pressure has dropped by about 25 % .
The two cases have the same boundary condition for the inlet section of the IPS (the one given as specifications and depicted above) but differ for the starter boundary conditions and, obviously, for the definition and values of the constraints. The choice

to change the starter boundary condition derives from the different characterization of the two optimizations: while case 1 is done from the "industrial" point of view, with narrower component specifications and less possibility for the solver to modify the geometry, case B is done from the "optimization" point of view: instead of setting an inlet mass flow it has been characterized with a fixed total pressure obtained from the simulation of the base geometry, to make it more compliant with the optimization idea. This choice comes from the consideration that the mass flow out of the starter can change if the pressure drop is reduced and its value is not fixed. Since we don't know anything about the starter, from the industrial point of view that condition was a fixed constraint, but for the purpose of this analysis we can allow more possibilities of modifications to better explore its capabilities. The expected result should still be similar and the final geometry should be able to reduce the pressure even for the given mass flow. On the other hand, the temperatures are as given by the specifications, since no reason to change can be found. Regarding case 1, the amount of material to be introduced in the domain is not a fundamental constraint, since there are no problems from the assembly point of view. It is obvious that the more solid it is, the higher will be the pressure losses of the IPS inlet, but this is a parameter that the solver will manage: no solid will be added if this would generate too high a pressure increase. Considering that the TO-Region is quite large compared to the solid domain that we expect the solver to create, the SVR value is clearly very small (the best value was found at 3% of the volume, i.e. an SVR of 0.03) while for case 2, as mentioned, we need more volume to reduce the obstruction that the solution can generate in the engine compartment, so the best value for the SVR was found close to $0.15 - 0.16$. While the first choice is quite arbitrary and made on an expected result, the second choice comes from the consideration of external constraints and thus is a more severe constraint. On the other hand, this optimization is not done from an industrial point of view, so a value of 0.15 was found to be a good compromise between them. As for the goal of our optimization, as mentioned, it lies in minimizing the pressure at the starting section (or maximizing the mass flow), while the pressure on the IPS section was set as the maximum constraint. The problem here comes from the choice of the maximum value: as we already know, the physics of topology introduces a non-negligible error in the solution, which means that setting the maximum value equal to what we want may be unfair to the real problem and the optimization may not converge. Although we do not know the error introduced in the solution, it seems from previous results that the Topology Physics tends to overestimate pressure losses. Since we want to reduce the increment on that pressure as much as possible, its value has been increased by a 33 %. We must remember that, by definition, this value is the maximum acceptable by the solver, but the constraint function assumes negative values before this limit and, since the constraint is "$PressureCF_{MAX} = 0$" (chapter 3.4.2) the solver always tents to decrease that value.

4.2.2 Optimization Results

This section discusses the results of the optimization analysis. As expected, the solver tried to create an acceleration on the IPS duct to decrease its pressure and try to entrain more fluid from the starter. For what concerns *Case 1*, this is the only modification that the solver implemented: the final geometry is made for re-

ducing the section of the IPS duct, meanwhile bending the flow coming from the starter duct. The result is a more smooth intersection of the two flows and a reduction in the separation after the junction. Interestingly, even though multiple boundaries were characterized as a source for the solid domain (material indicator = 0), the solver only worked on the edges before the junction itself. Although we expected this result, it is useful to notice that no modifications in the sharp edge were made even thought the surface sensitivity analysis seemed to suggest it, perhaps because the required modifications were not possible with the actual geometry. This consideration is supported by the result of the second optimization: while the before-junction geometry is nearly the same as the one obtained in *Case 1*, the sharp edge is removed and replaced with a more smooth path for the starter flow. This is the result that we expected when analysing the flow behaviour at the junction and the solver seems to confirm it.
Let us focus on the results of *Case 1*: as stated, this optimization was performed with the goal of minimizing the pressure of the Starter section as much as possible, while keeping the pressure on the IPS section below a maximum value. In this case we also analyzed the differences between the manual and the automatic sensitivity normalization. Based on the results obtained in chapter 4.3, we already know that the automatic approach could lead to oscillation in the solution or un-satisfaction of some constraints. This behaviour is confirmed also in the 3-D case, since with this approach the solution hardly managed to converge but introduces some oscillations when one constraint starts being satisfied (4.34): this confirms that the solution seems to delete the constraint once satisfied, which obviously is like changing completely the problem in analysis. Such a behaviour is not present in the manual approach, which is capable of a smoother convergence; moreover, thanks to its particular definition, the optimization not only satisfied all the constraints but tried to remain aware from them, improving the solution from this point of view as well (this can be avoided acting on the β parameter in the report definition if necessary).

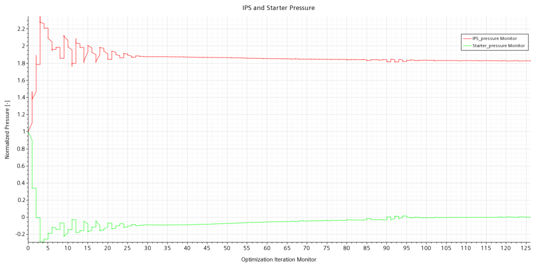


Figure 4.33: IPS and Starter pressure behaviour during the optimization iterations: Manual normalization case. Values are normalized with the one at 1^{st} iteration.

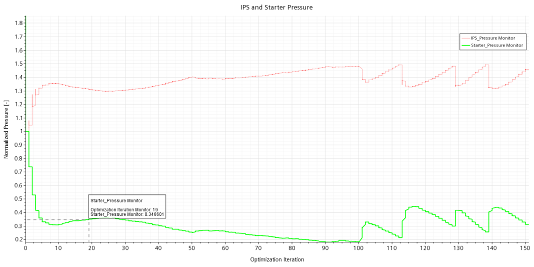


Figure 4.34: Behaviour of an automatic normalization optimization with a too high penalty value. Values are normalized with the one at 1^{st} iteration.

Comparing the topology obtained from the two approaches (fig. 4.35) it is possible to appreciate huge differences. The basic idea is the same: to reduce the section of the IPS duct in order to increase its velocity and help the starter flow to change its direction. However, while the geometry obtained with the automatic approach is "simpler", the one obtained with the manual one is more peculiar since two small support-like ducts are generated that connects the two pipes **before** the real junction. From figure 4.37, it is clear what effect on the flow field these structures have: the one on the right (lower Y coordinates) sucks in the IPS flow, generating a high vorticity flow that acts as a vent, removing a portion of fluid that is diverted from the new solid domain. The other (and larger) structure acts as a recirculation zone for the rotating flow coming from the Starter: as shown in figure 4.36, the bend of the starter pipe generates a vorticity that, near the junction, is almost perpendicular to the IPS flow direction. That portion of the domain acts as a recirculation pocket where the vorticity is very high but the flow velocity is very small: it acts as a kind of stilling chamber that slows down the flow but avoids it entering the main tube with a wrong direction. While this may seem like a source of vorticity and turbulence, the beneficial effect on the Starter flow behaviour is relevant, so that the final pressure on the Starer Inlet section drops at the 25.4% of the initial value, with an increment in the IPS of ≈ 30% only, which is absolutely consistent with the constraint that we imposed (all values are reported in tab. 4.7).

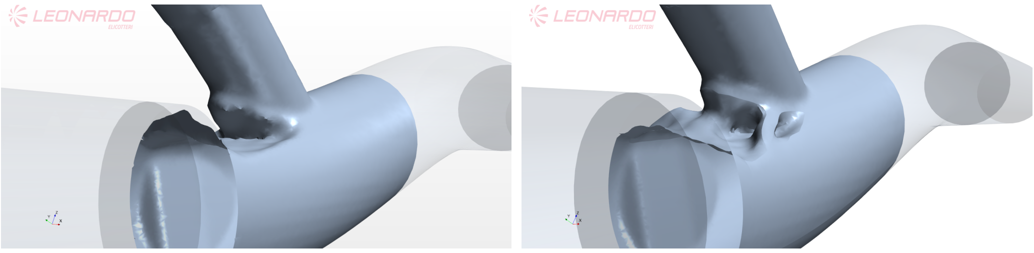


Figure 4.35: Comparison of the final topology between automatic (a) and manual (b) report defintion and normalization

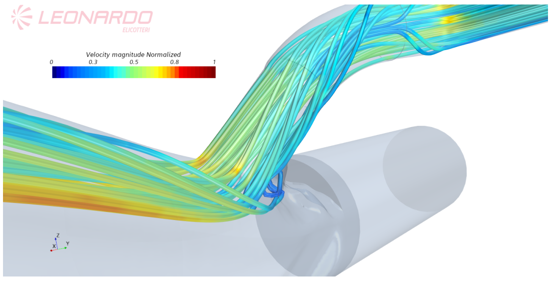


Figure 4.36: Streamline on the Starter duct before the junction. Note the birth of a rotational component in the flow

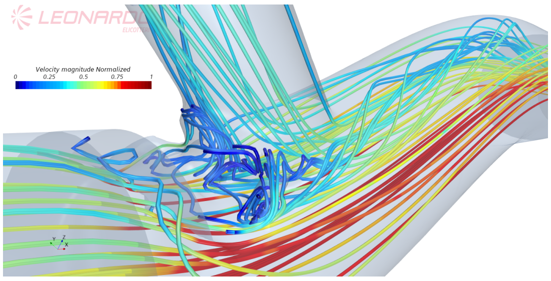


Figure 4.37: Effect of the two support-like structures on the flow behaviour

Table 4.7: Comparison between the improvement given from the automatic normalization or the manual one. Values are normalized with respect to the maximum IPS constraint value

	IPS pressure [-]	Starter pressure [-]
Base Geometry	0.545	0.69
Automatic Normalization	0.66	0.4
Manual Normalization	0.725	0.175

Let us now focus on *Case 2*; this optimization was performed with a different target: the maximization of the mass flow coming from the starter. In order to perform this optimization, the inlet boundary condition must be changed: the choice fell on fixing the total pressure at that section with the value found from the base configuration. Also the constraint on the pressure was slightly changed: to reproduce a more realistic behaviour, a maximum pressure drop of the whole system was considered. Provided that the total pressure of the Starter section is given as boundary condition, the constraint works on the IPS pressure but taking directly into account the increasing of the mass flow from the starter: to provide more freedom to the optimization, the constraint has been increased about 2.5 times the base value. Moreover, as previously said, the optimization domain was modified to give the solver more freedom and more possibilities of topology modifications. Since this optimization was not intended to compare manual normalization with automatic normalization, only the former was used, since it gave higher performance based on the results of *Case 1*. The final topology is shown in figure 4.38 and it is based on the same considerations that we made for the previous case; some (huge) difference can be found: first of all, the IPS input section was in this case modifiable and therefore the solid domain connection is smoother. Second, the mounting-like structures that were generated before are now reduced to a single, larger connection, perhaps due to the different approach the flow has in that section, due to the smoother connection mentioned above. The biggest changes are in the region after the junction: the final part of the starter pipe and the very first part of the junction are shaped by the solver to give a smoother connection between the pipes (as expected) and to follow the rotational structures that the Starter curve generates (unexpected). The flow is therefore almost "accompanied" in its path, with the new geometry trying to generate as little hindrance as possible to the natural development of the flow (fig. 4.39).

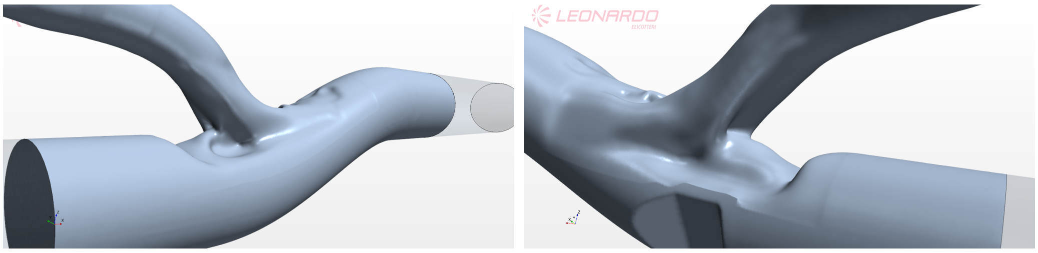


Figure 4.38: Final topology for the maximization of the mass flow case. Highly 3D structures are evident in the new topology that help the natural flow development

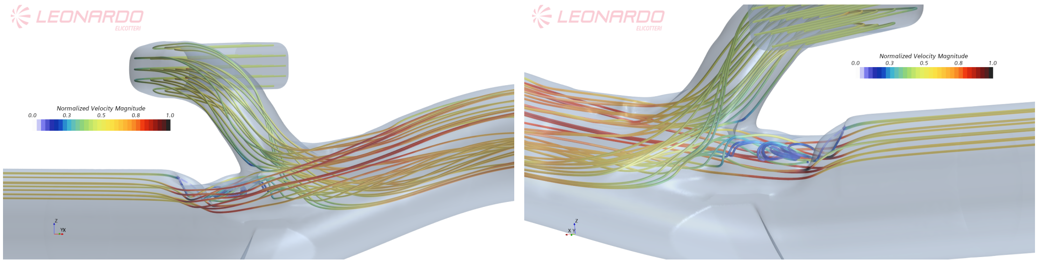


Figure 4.39: Flow behaviour in the optimized geometry. Note how the geometry follow the swirl of the starter before the mixing.

Although the two optimizations are performed with different objective function and domains, it may be interesting to analyze how the new geometries behave when the boundary conditions are changed. In fact, the effects of the optimizations were almost similar in the final topology and it is not immediate to understand if the solver felt the different setup and objective function. It is important to remember that these values refer to the geometry with the Starter inlet section cut off. The table 4.8 shows the characteristic quantities of the two analyses: the mass flow rate at the starter inlet, the pressure at the IPS section, the pressure drop of the system and the total pressure at the starter section. From the results it is quite evident that the *Case 1* performs better in terms of mass flow even with the switched boundary condition, but generates a higher pressure at the IPS section, while case 2 performs better in pressure drop. Although this was expected, since the two optimizations perform better in the physical quantities considered in the optimizations, it is a confirmation of the sensitivity that the optimization itself has on the objective function with respect to the imposed constraints. It is therefore difficult to say which geometry performs better in an absolute way, but it clear that the two optimizations managed to find the better result for **their own** constraints

Table 4.8: Comparison between the results of the IPS optimization cases. In bold are characterized the boundary conditions. Pressures are normalized w.r.t the maximum IPS constraint value, while mass flows w.r.t the starter boundary condition

		Mass Flow Starter [-]	IPS Pressure [-]	Pressure Drop [-]	Starter Total Pressure [-]
Case 1	Nominal	**1**	0.71	0.28	0.18
	Switched	1.55	0.8675	0.288	**0.764**
Case 2	Nominal	1.38	0.74	0.2335	**0.764**
	Switched	**1**	0.643	0.24	0.4755

4.2.3 Topology physic error analysis

As done for the Quasi-2D test case, a small section is dedicated to the analysis of the error introduced by the topology physics model with the Brinkman penalization, here performed for the 3D cases. In tables 4.9 and 4.10 the values obtained during the optimization are compared to those obtained after the geometry reconstruction (with the Starter inlet section cut off). As noticed in the Quasi 2D test case, the solver tents to overestimate the pressure losses of the system, but this time the error also reflects in the mass flow values (for obvious reasons: the quasi 2D test case had the mass flow imposed as boundary condition), which also are overestimated.
Another important physical quantity that may be of interest is the temperature at the outlet section, even though it was not considered as constraint during the optimization. The porous medium is also characterized by an effective thermal conductivity (see cap 2.3.3.2 and eq. 2.47) and it is capable of thermal conduction within the flow domain. Even thought it is not possible to combine the Adjoint topology optimization model with the Radiation model inside STAR-CCM+ (it is not possible to simulate conduction or radiation) the porous media tent to simulate

the conduction using the energy equation and an increased conductivity. In the case just analyzed, temperature was not in any way a design constraint and, therefore, was not taken into account as a parameter for optimization. However, the flow coming from the IPS and the one coming from the Starter are characterized by two different temperatures, even if slightly (the temperature of the IPS is 0.9 times that of the Starter). Therefore, even if temperatures between the flows are almost similar, analyzing the temperature at the outlet can give us a further indication of the quality of the model. From figure 4.40 it is possible to see the thermal diffusion inside the porous media, both in the upper part of the domain and in the new junction between the IPS and the Starter ducts. The test cases here presented are not suitable for a more precise and correct analysis of the effect of the porous media and the effective thermal conductivity on the solution, which can be found for example in [11].

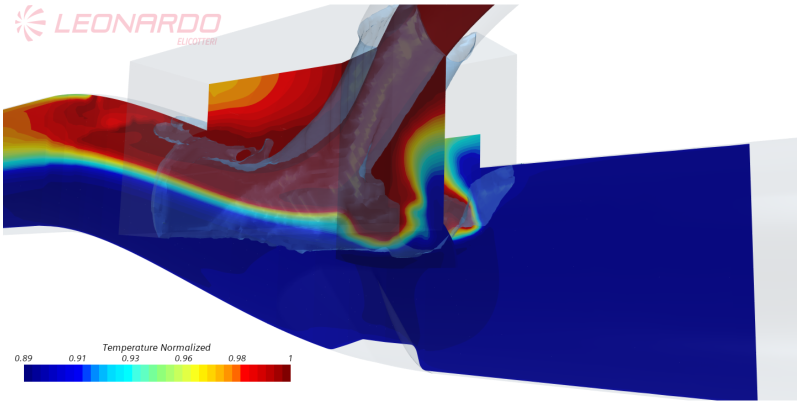


Figure 4.40: Temperature field inside the porous media. The convection is simulated with an effective thermal conductivity. Values are scaled w.r.t the maximum value

Table 4.9: Error introduced by the Topology physic model - Case 1. Pressure values are normalized w.r.t the maximum IPS constraint value, while temperatures w.r.t the Starter boundary condition

	IPS Pressure [-]	Starter Pressure [-]	Temperature at Outlet [-]
Topology physics	0.954	0.04	0.91
Reconstructed	0.71	-0.005	0.89

Table 4.10: Error introduced by the Topology physic model - Case 2. Pressure values are normalized w.r.t the maximum IPS constraint value, while mass flows and temperatures w.r.t the Starter boundary condition

	IPS Pressure [-]	Mass Flow Starter [-]	Temperature at Outlet [-]
Topology physics	1.165	1.788	0.915
Reconstructed	0.74	1.4	0.91

The results seem to confirm the first observation made on the Quasi-2D test case: all the physical quantities are overestimated during the optimization, but the error does not seem to be unique, constant or predictable. Besides this, we can again confirm what was previously noted: even with this (non negligible) error, the optimization performs well and manages to optimize the topology of the components, i.e. the topology does not seem to be influenced by the numerical error, but it is more dependent on the general behaviour of the flow field.

5

Topology Optimization of the Primal Exhaust of the NGCTR Engine

In this chapter a brand-new application of the topology optimization model for fluid dynamic problems is presented, based based on knowledge gained from test cases: the optimization of the lobed exhaust of a turboshaft engine. This application is completely different from the others previously addressed, since the TO-Region is constrained inside a fluid domain and not at its boundary, with the porous media being included in the flow field and not vice versa. This optimization was performed on the Next-Generation Civil TiltRotor (NGCTR) engine with two different purposes: the first optimization aims to maximize the mass flow driven by the engine bay, while the second one aims to minimize the backpressure on the turbine outlet section.

After a brief introduction of the problem and of the geometry, the two different optimizations are discussed separately and a brief comparison of the resulting topology is presented at the end of the chapter.

All the optimization here proposed were obtained with the new sensitivity normalization procedure discussed and developed previously.

5.1 Definition and Setup of the Problem

When considering an helicopter engine (or VTOL Tilt Rotorcfraft), it is critical to remember that unlike an airplane jet engine, which uses the acceleration of an air mass to produce thrust, the main task of the turbine stages is to transform the kinetic and internal energy of the fluid into the angular velocity of a shaft that is connected to the blades. The result is that the remaining kinetic energy of the exhaust flow is only needed to move the hot flow away from the engine itself, but no thrust can be recovered from it. However, since cooling the engine and its compartment is a critical task to ensure proper operating conditions, it is easy to see that the amount of energy remaining in the exhaust can be used for this purpose.

Starting from this idea, the jet pump effect (see Appendix A) is taken to the next level: the gas exiting the low pressure turbine is the *high pressure flow* while the air inside the *Engine bay* (see fig. 5.1) is the reservoir of the *driven flow*. The aim of the jet pump effect here is the **motion of the air** along the engine to generate a forced convection, so to ensure a proper speed of the fluid along the whole engine some intakes are needed in the Engine Bay, whose position is predominantly determined by the mechanical component inside. These air intakes are placed on the very outer side of the engine bay in order to sucks the air from the environment, thanks both to the propeller (especially during VTOL phases) and the external aerodynamics during the cruise phase. For this kind of applications, the final pressure or velocity of the combined flow is not a design parameter, but it is fundamental that the pressure at the exhaust of the turbine does not rise above a certain value that can reduce the performance of the engine.
The fundamental component to ensure the optimal performance of the cooling system is the exhaust system and its geometry, in particular the topology of the so-called *primary exhaust* (see fig. 5.2): this component has been design via experiments, knowledge and trial-and-error procedures leading to the actual lobed-geometry. Since these can lead to incorrect and sub-optimal designs, it is clear the need for a procedure for a topology optimization of this component.

5.1.1 Engine base geometry

The exhaust topology optimization process, along with the development of its procedure, was carried out on the **CT7-2EI** engine, a commercial version of the T700 designed and developed by GE Aviation [1], that will equip the **Next Generation Civil TiltRotor** (**NGCTR**) of Leonardo S.p.A. Helicopters [18]. As said before, the optimization concerns only the exhaust components, so to simplify the geometry the smallest engine features have been removed, keeping the same obstruction of the engine inside its bay but at the same time allowing the generation of a coarser mesh. The purpose of this analysis is not a proper fully analysis of the smallest features of the engine, which have only the functionality of representing the flow domain and an obstruction to the passage of air.

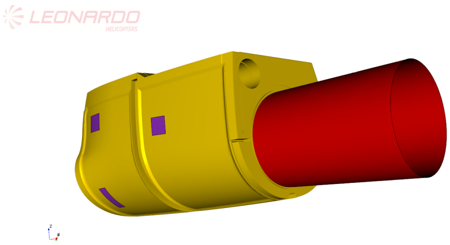


Figure 5.1: Engine bay (in yellow) with 3 of the 4 inlets (in purple) and the secondary exhaust (in red)

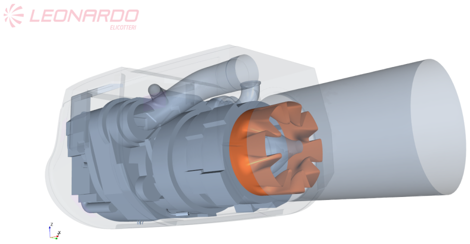


Figure 5.2: Internal components of the engine bay: In gray all the engine block with the actual geometry of the primal exhaust in orange

From figures 5.1 and 5.2 it is possible to distinguish some of the components of the engine system that are relevant for the analysis:

[1] https://www.geaviation.com/

- the main component (in gray) is the engine itself with some auxiliary components like the IPS and the Starter ducts (the ones analyzed in chapter 4.2) and a simplified support on the rear (it is interesting to know that this component was subjected to a structural topology optimization). These bodies are collected into a single one since they will never be modified during the optimization process.
- the primary exhaust (in orange): this is the principal component of the topology optimization analysis, since is the one that must be optimized and the one that can maximize the jet pump effect. It is connected to the engine along with the conical component at the exit section of the Low Pressure Turbine, which in our analysis represents the a mass flow inlet boundary condition.
- the secondary exhaust (in red): this is a conical section that represent the allowable domain of the exhaust. Inside this domain, the flow exiting the turbine sucks the flow inside the bay and mixes together to form a single exiting flow. This part of the exhaust was not part of the optimization process.
- the engine bay (in yellow) along with the intakes (in purple): the engine bay represents the whole domain of the flow inside where every component must stay. The walls of the bay are of two different types:
  - the back wall is the firewall: its shape and material distributions are governed by the fire resistance criterion for the airwothiness and therefore no air intakes can be introduced.
  - all the other walls represent the sides of the bay, which are part of the external nacelle and that's why the intakes are placed here. Their location is governed by the internal distribution of the components.

It is fundamental to remember that the external aerodynamic plays an important role in this problem, since the pressure and the mass flow inside these intakes depend on the velocity and pressure fields along the nacelle. However, the optimization presented here did not take these phenomena into account not only for computational reasons, but also because the primary exhaust should be optimized under all operating conditions with the addition of the propeller in front of the engine itself. In addition, the optimization process for this component aims to identify a topology that is as general as possible and can perform well under different conditions, not a specific one

5.1.2 Problem Setup and Mesh Analysis

The boundary condition setup deserves special attention especially for the TO-Region definition. The engine is simulated under a general operating condition, with flow exiting the turbine stage at a constant mass flow rate and temperature. Since no information about the swirl component of the flow is provided by the engine installation manual, the flow is considered as normal to the boundary: this is obviously an unrealistic behavior of the flow but we have to accept this error, always focusing on the objective of this analysis, which is not the numerical result but the behavior of the topology optimization in a complex aerodynamic scenario.

Moreover, as stated in the previous chapter, the topology optimization model does not support the *Radiation* model inside STAR-CCM+, therefore all the physics concerning the conduction and radiation inside the engine bay cannot be simulated: this is consistent with the idea of the topology optimization, since the radiation model needs to create patches on the solid boundaries to calculate the view factors for the simulation of the radiation. The solid generated via the porous medium cannot be patched, for two different reasons: the first one lies in the fact that the porous media changes from iteration to iteration, therefore at every iteration the new patches and the new view factor matrix should be computed, which will lead to an enormous increase in the computational time. The second reason is simply related to the aforementioned characterization of the *new solid*: the solver treats it as fluid, therefore it cannot generate patches on its boundaries. As a consequence of that, the temperatures of the engine block are not considered in the following analysis. However, for the mesh convergence analysis and validation of the new geometries the temperature and velocity profiles, together with their average value at the outlet section, are taken into account, since the flow exiting the turbine stage is characterized with a high temperature.
Here the boundary conditions for the physics are summarized:

- Turbine inlet section: although it is called "Turbine inlet" it is in reality the outlet section of the turbine which, for our analysis, is a mass flow inlet condition with also a fixed temperature.

- Air intakes: this should represent a connection with the environment, therefore are characterized with a fixed total pressure equals to the reference pressure and the temperature is set at $T_{intakes} = 293.15\ K$

- Outlet section: this is a simple pressure outlet that represents the discharge of the exhausts in the environment. To reduce its effect in the solution, the outlet section is moved away from the real one.

The definition of the To-Region and of the boundary and initial conditions for the Material indicator function deserves a separate discussion. As stated before, the region inside which the topology optimization will be performed is the one that is actually occupied by the primal exhaust, therefore it is completely immersed in the fluid region. This can represent a problem from the optimization point of view, since it is not easy to characterize a boundary as source for the solid domain. Moreover, some mounting constraints reduce the modifications that the solver can make at the very beginning of the exhaust (see fig. 5.3).

The solution that has been found to overcome the latter problem is quite obvious: the first part of the exhaust is fixed and not modifiable at all. For the first observation, on the other hand, the idea was to introduce a particular initial condition for the material indicator that would retrace the original geometry and characterising as source for the porous media only the boundary that connects the TO-Region to the first non-editable part of the exhaust. It is important to recall that the porous medium cannot develop solid from boundaries that are not sources, but it can be generated from the one that already exists. Therefore, the actual geometry performs

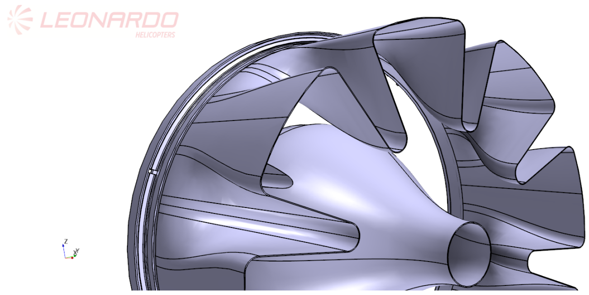


Figure 5.3: Part of the mounting system of the Primal Exhaust

as initial condition (fig. 5.4) and also as source for the solid domain. To obtain that particular initial condition, the following functions were created [7] (field functions inside STAR-CCM+ are based on Java programming language):

- *UnitNormalVector*: this function is a Vector function that simply normalizes the area of the cells considered creating a vector that is parallel to the normal. It is defined as:
$$\frac{\$\$Area}{mag(\$\$Area)} \tag{5.1}$$
- The scalar components of the result vectors are collected in a user-defined table (called *"UnitNormalTable"*) that applies this field function to the surface mesh of a selected part only, with the relative (X,Y,Z) coordinates in the general reference frame. In this case it is computed on the surface mesh of the primal exhaust base geometry;
- *InterpolateFunction*: this function interpolates the position of the volume cells with the one of the tables and performs a scalar product with the *UnitNormalVector* components:

$$\begin{aligned}&InterpolateFunction =\\&dot(\$\$Position - interpolatePositionTable(@Table(''UnitNormalTable''),\\&''X'',''Y'',''Z''), interpolatePositionTable(@Table(''UnitNormalTable''),\\&''UnitNormalVector[i]'',''UnitNormalVector[j]'',''UnitNormalVector[k]''))\\&< 0\end{aligned} \tag{5.2}$$

  The result of this function is therefore a Boolean field, that assumes the value 1 on the cells whose centroid is located inside the selected part and 0 elsewhere;
- *InitialConditionFunction*: this latter function has the only purpose of switching the values assumed by the *InterpolateFunction*, since we need a value of 0 (=solid) on the cells located inside the base geometry of the primal exhaust:
$$InitialConditionFunction = 1 - InterpolateFunction \tag{5.3}$$

Since the Coupled solver may suffer convergence issues, especially on the continuity, if the initial condition is very different from the converged solution, also for the

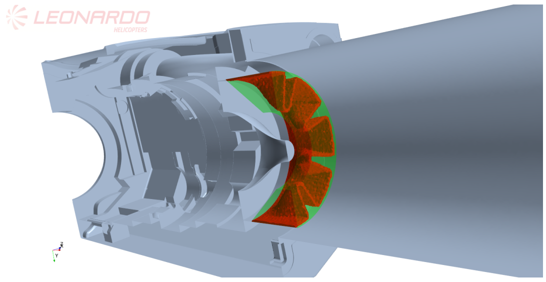


Figure 5.4: Initial Condition for the Material Indicator (in red). In green is highlighted the TO-Region

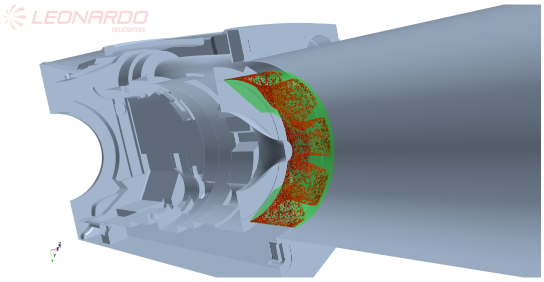


Figure 5.5: Example of the result of the *InitialConditionFunction* with a too coarse mesh

physics of the problem an indication of the final solution can be helpful. Generally, one possible solution is using the Segregated solver until a reasonable convergence and then switches to the coupled one for the final convergence. However, in this case this procedure is not so reasonable, since all the adjoint setups should be done after the switching and, therefore, for every optimization that we perform we must set up the adjoint solver from the beginning, since the segregated solver is incompatible with the adjoint one. A possible solution to this problem is the usage of an initialization with a *multigrid* approach: the analysis is by the very beginning set up with a coupled solver, while the grid-sequencing (GS) initialization performs the normal initialization of the problem with the given initial condition followed by the computation of an approximate inviscid solution to the flow problem [41]. The GS works with the following steps:

- A series of coarser mesh are generated (the number of the mesh are chosen by the user);
- The coarser mesh is initialized with the given standard initial conditions;
- Starting from the coarser mesh, a series of iterations are run until a chosen convergence or a given maximum number of iterations is reached and the obtained solution is then interpolated on the new mesh. The step is repeated until the finer mesh.

The result of this process is a first-order inviscid solution that deserves as initial condition for the Coupled solver, improving the robustness and the convergence of the primal solution.
With the given physics condition the mesh convergence was performed, with the following parameters taken into account at this purpose:

- Average velocity at the outlet section;
- Mass flow at the air intakes;
- Average temperature of the flow at the outlet section;
- Velocity and temperature profile at outlet section;

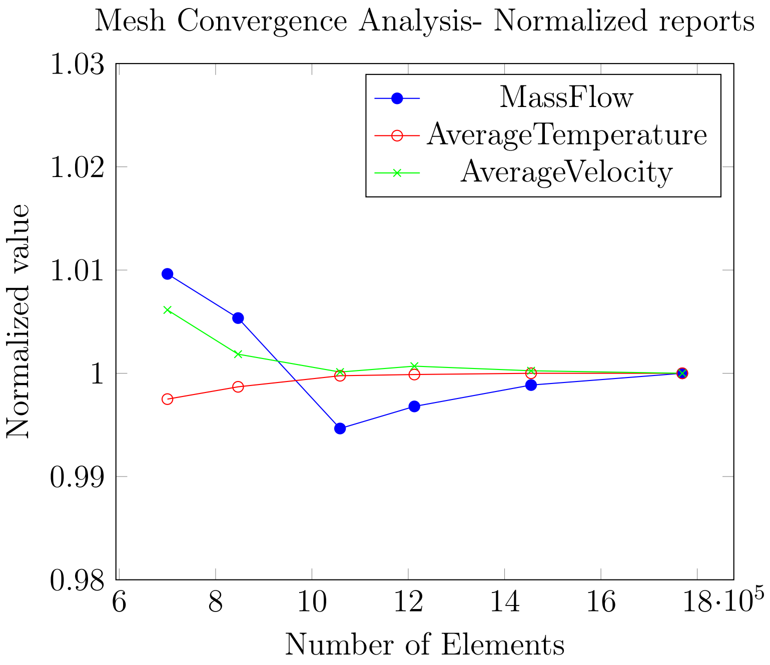


Figure 5.6: Mesh convergence analysis - The reports are normalized with their asymptotic value

The mesh model used for the analysis is the Poly one, since the high complexity of the geometry is not suitable for a trimmed one. The results of the convergence analysis are shown in figure 5.6 and parameter of the chosen mesh are reported in tab 5.1.

Since the engine, regardless its simplified model, has small features, and inside the bay there are other components (like piping systems or big structural elements), a stricter control on the prism layer is necessary to correctly keeping into account the different velocity scales of the problem. In the bay the velocity of the flow is times smaller than the one in the exhausts sections and the gap between the engine and some other components cannot accept the same boundary layer that is needed on the secondary exhaust walls. Therefore the prism layer was characterized following the Base size but was also given the possibility of reducing the total thickness up to the 1% of is generic total thickness, reducing also the number of the layers if needed (fig. 5.9).

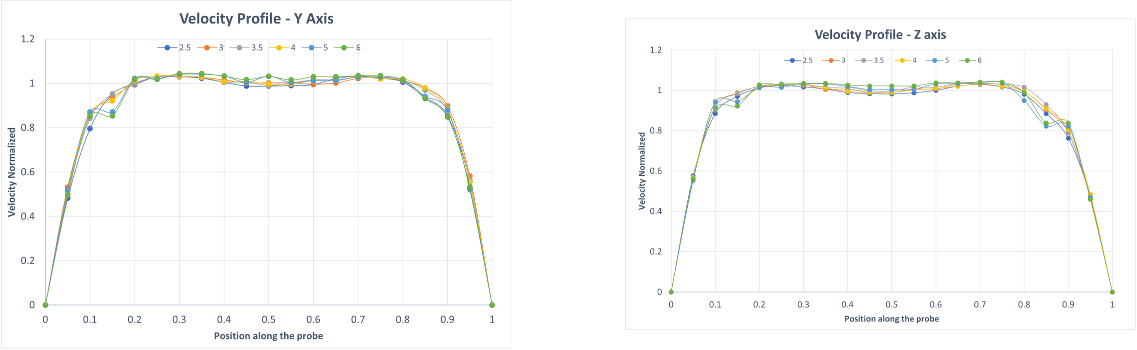


Figure 5.7: Normalized Velocity profile at outlet section - Mesh convergence analysis: Data refer to Base Size

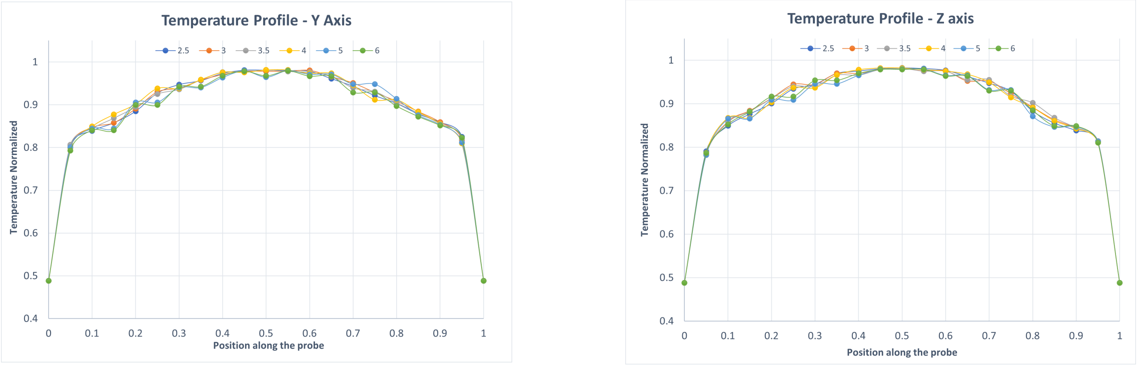


Figure 5.8: Normalized Temperature profile at outlet section - Mesh convergence analysis: Data refer to Base Size

Table 5.1: Mesh parameters: Engine primal exhaust optimization. Percentage refer to the Base Size

Mesh parameters	Base Size [m]	$2.5 \cdot 10^{-2}$
	Minimum Surface Size [-]	10 %
	Maximum Cell Size [-]	300 %
	Refinement To-Region Cell Size [-]	20 %
Prism Layer	Stretching factor [-]	1.5
	Number of layers [-]	5
	First Layer Thickness [m]	$2.5 \cdot 10^{-4}$
	Total Thickness reuction [-]	1%
Total Cell Number		$\approx \mathbf{1.8 \cdot 10^6}$

The chosen mesh is fine enough to capture the correct flow behaviour but the domain for the optimization is in the middle of the fluid one, where the mesh is bigger than the Base size and it is a bit too coarse for the definition of the initial condition of the material indicator function. This one, indeed, is given with the particular initial condition generated by the above described field functions: a greater definition of the cells guarantee a greater characterization of the initial condition. A too coarse mesh will generate sort of pockets of solid cells scattered through the domain and not a real porous domain (fig. 5.5). Therefore, the TO-Region was characterized by a local refinement (20% of the Base size) in order to improve its definition.

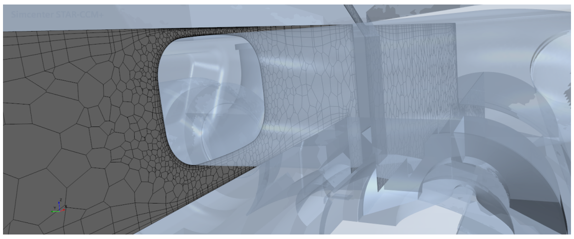


Figure 5.9: Detail of the Prism layer reduction in the mesh. Here, the Starter pipe reduces the fluid domain and the prism layer general settings cannot fit correctly

Once the problem is defined from the physical point of view, it is fundamental to analyze and choose the objective functions and the constraints for the optimization, together with their normalization. In the following, two different optimization were performed with two different goals: the first optimization concerns the maximization of the mass flow driven from the engine bay with a maximum constraint on the back pressure that it generates at the turbine inlet section. The second optimization is the opposite: the minimization of the pressure at the turbine inlet section with the less possible loss in the mass flow. Both optimizations start with the same initial conditions on the material indicator function and this already gives us the first constraint of our analysis: evaluating the SVR of the initial condition, we can obtain the amount of solid of the base geometry of the primal exhaust. The goal is therefore optimizing the geometry keeping the same (on nearly the same) amount of solid which is near the 11 % of the TO-Region (i.e. $SVR_{target} = 0.11$), in order not to increase its weight.

5.2 Mass Flow Secondary Exhaust optimization

The cooling of the Engine and of its Bay is a fundamental task to guarantee a proper working condition and reliability of the engine itself. Part of this process is entrusted to the air flowing through the bay, which cools down all the components thanks to the forced convection: the greater is the mass flow (and therefore the velocity) the greater is the cooling effect that it can achieve. In standard working conditions, the air can flow through the bay thanks to the Primal exhaust that mixes it with the flow exiting from the turbine stages, therefore it is clear that the fundamental component that controls the amount of air that can flow throughout the bay is the mixer itself. However, this is not the only parameter that it governs, since it must guarantee that the mixing effect does no generate too much pressure losses and too much backpressure on the turbine outlet stages, both for reducing the fuel consumption and guaranteeing a correct outflow of the gasses. These two parameters are therefore respectively our objective function (mass flow flowing in the bay, called also "mass flow secondary") and the second of the two constraints for the analysis (pressure at the turbine inlet section); the other, as stated before, is the SVR.
In order to have an idea of the maximum allowable value of the backpressure, we must consider the specifications of the engine itself: the fig. 5.10, taken from the engine installation manual, represents the minimum allowable backpressure coefficient with respect to the swirl of the flow. The represented coefficient is a standard

pressure coefficient defined as follow:

$$BP = \frac{P_{atm} - P_7}{VH_7} \tag{5.4}$$

where "7" is the number of the stage with respect to the whole engine, and denotes the turbine exhaust section (i.e. our mass flow inlet boundary): P_7 is therefore the static pressure at the turbine exhaust and VH_7 is the dynamic pressure at the same stage.

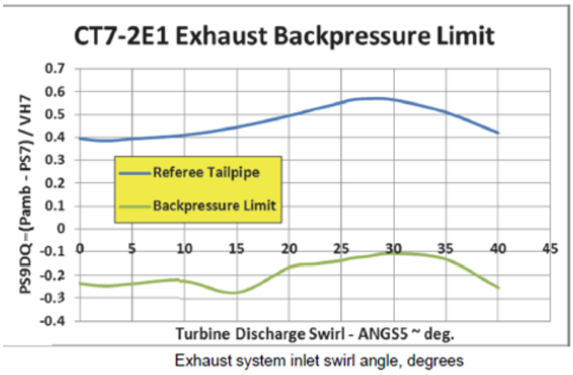


Figure 5.10: Minimum backpressure coefficient allowable for the turbine exhaust section

Since we do not have indication about the swirl, and in order to simplify the analysis to better understand the possibility given from the topology optimization in these complex aerodynamic cases, all the analysis were performed without the swirl of the flow. Even though this is a fundamental aspect of the flow behaviour, it is important to consider that the complexity of such a problem is almost times higher than the one of the standard topology optimization problems like the test cases discussed before, and the introduction of the swirl will lead to perform an optimization for different swirl conditions (at least every 5 deg) which translates in an unattainable computational time (later on the CPU times required for the performed analysis are reported).
With the actual exhaust geometry, the backpressure coefficient is equal to

$$BP_{base} = 0.42 \tag{5.5}$$

Of course the Backpressure limit is fully satisfied with a high margin for modifications. With a view to the optimization, this leaves a high design space where to move; moreover, keeping in mind that the Brinkman penalization model introduces a non-negligible error which affect the setting of the constraints, let us to define a maximum value for the pressure with a reasonable safety margin. The choice to define the pressure as constraint and not the backpressure coefficient itself comes from the knowledge acquired from the previous optimizations, where we always worked with the sensitivities of pressures and mass flows.
Since the computation time for the whole domain geometry would have been too high for the correct settings of the problem, the optimization was performed step by step starting from a reduced portion of the domain. This helped in the definition of the reports for the correct sensitivity normalization and, subsequently, the acquired knowledge was applied to the whole problem. Therefore, these steps are presented

below, starting from the optimization of the primal exhaust with the reduction of the engine bay of only a quarter, introducing the hypothesis that the bay is 2-times symmetric.

5.2.1 Optimization with a quarter of the domain

When analyzing the Engine bay (fig. 5.1) and what there is inside (fig. 5.2), it is clear that the domain is anything but symmetrical. However, facing a so complex problem for the first time is easier if some simplifications are introduced. The engine bay domain is therefore considered as made of 4 equals regions, therefore with a 2-times symmetry with respect to the Y and Z axis. This simplification obviously introduces some errors in the simulation and some 3-Dimensional effects might get lost. As we already know, 3-Dimensional effects seem to play an important role in the final topology that the solver achieves, so it is not possible to consider the result of this optimization as the final one, but it is a good starting point if we are interested in analyzing the possible topology and why it is so. The new fluid domain (fig. 5.11) is obtained with a slicing of the Engine bay and the two sliced boundary are characterized as Symmetry boundaries.

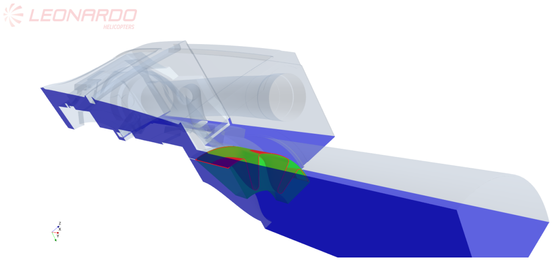


Figure 5.11: Quarter of the Engine bay: in blue the symmetry boundaries

As it can be seen, the chosen part of the bay was the one that could guarantee almost 2 air intakes in the same position of the one in the full domain geometry, in order to keep the suction both from the rear and the front of the engine. Since the base geometry of the primal exhaust is axial-symmetric, its configuration was taken as reference for the cutting plane, in order to consider 2 of the 8 lobed section in a perfectly axial-symmetric way. The mass flow exiting the turbine was reduced exactly by a quarter, even though we cannot expect that the resulting mass flow entering the bay is exactly the quarter of the one computed with the full domain (we may expect a greater value since we have 2 of the 4 intakes) with a possible reflection on the value of the pressure on the turbine. Another important difference, as stated, is on the different flow behaviour between the full and the sliced domain. As shown in figure 5.12, the slicing of the domain destructs some important recirculation zones (in the upper left part) while creating an important one in the upper right region, near the IPS outlet section. The latter moreover generates a lower pressure zone that drives the flow in a x-positive vorticity in all the IPS zone, deleting the reciruclation zone that is clearly visible in the full domain under the IPS. Moreover, the flow in the Quart Domain analysis seems less chaotic and more clean, with the recirulation zones that are more structured. This translates in an easier path for the flow in the bay, which means a greater mass flow in the secondary exhaust, as confirmed by the results shown in table 5.2: the obtained mass flow is grater then the quarter of the

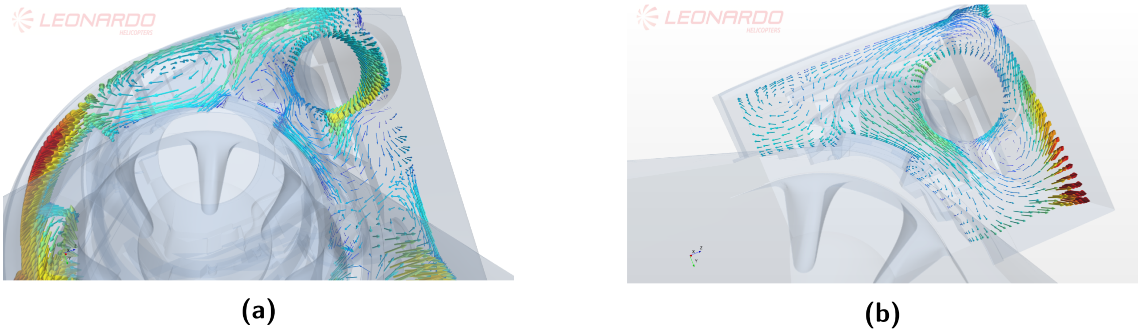


Figure 5.12: Comparison of the flow direction and magnitude between the Full Domain (a) and the Quarter Domain (b) at the secondary exhaust section. Blue indicates lower values.

original (+38 %), which reflects on a greater pressure at the turbine inlet stage and leading, as natural consequence, to a lower BackPressure coefficient.

Before proceeding with the optimization analysis, it is required to understand and to choose the maximum pressure that we may accept. In the previous cases, we cannot have any indication about the differences between the topology physics and the reconstruct geometry, but in this case we can: with the particular initial condition, indeed, we can compare the base geometry described both with the real component and the porous one, and analyze how the pressure and the mass flow are modified. With the Brinkman penalization model, the mass flow decreases of about 3 % while the pressure on the turbine drastically increases (tab 5.2) reducing the BackPressure coefficient at -0.21. Since we need to keep the Backpressure coefficient greater than $-0.22 \div -0.23$, it is obvious that the error introduced must take into account the overestimation generated by the model. Based on the aforementioned increment, a coherent value for the BackPressure coefficient may be around -0.8 but, since the error on the pressure depends on the mass flow and on the topology that will be developed, giving a bit of margin for the solver can help the convergence, therefore the minimum allowable BackPressure value was set at -1.0. For an easier characterization of the optimization problem, the constraint was imposed on the turbine static pressure, which value has been set accordingly to 2.32 times the value obtained with the Base geometry. It is important however to remember that the optimization should always aim to decrease its value (remember: $\beta_p < 0$). Once known the constraints value, their functions can be correctly defined (see figure 5.13 and 5.14).

Table 5.2: Comparison of the main parameters of the base primal exhaust between the real component and the porous-made one - Quart-Domain case. Mass flows and Pressures are normalized with the Base Geometry values

	Turbine inlet pressure (relative to Outlet) [-]	Mass flow secondary [-]
Real Geometry	-1	1
Quart Domain	-0.922	0.344
Quart Domain Porous Media	0.477	0.333

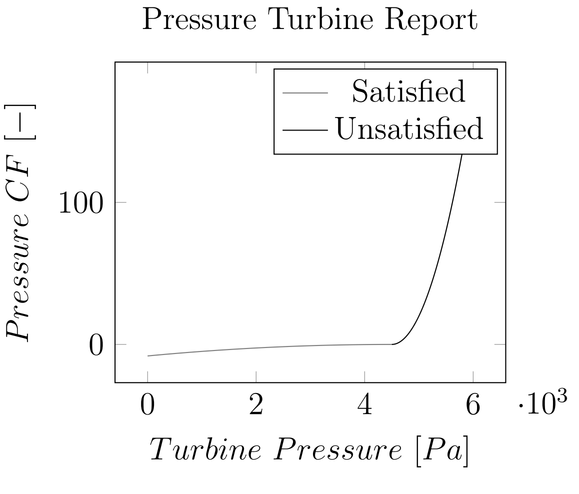


Figure 5.13: Pressure turbine report definition: $\alpha_p = -0.1$, $\beta_p = 20$, $P_{norm} = 0.257$, $P_{max} = 2.32$ (w.r.t the normalized value).

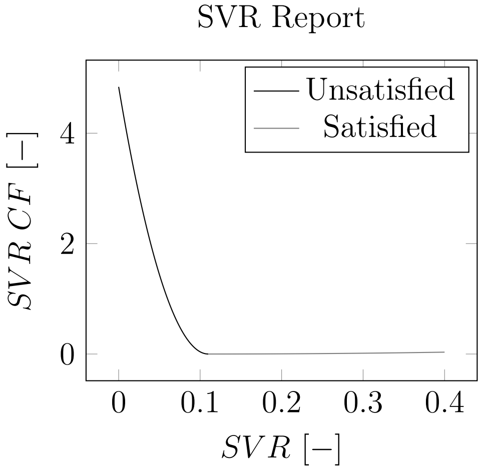
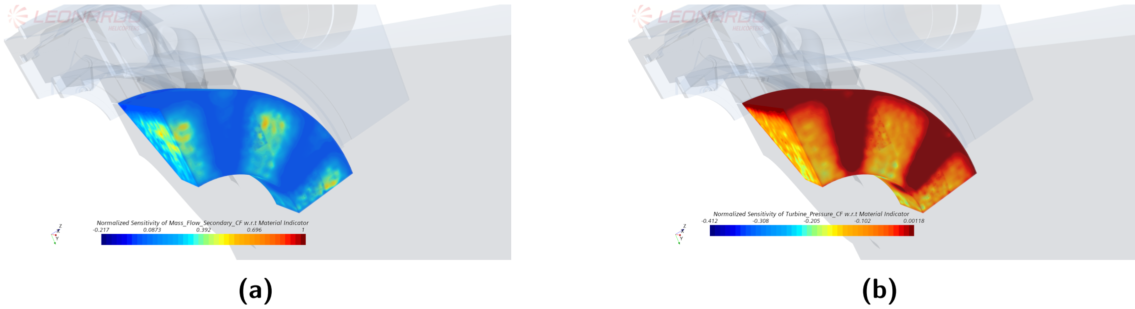


Figure 5.14: SVR report definition: $\alpha_{svr} = 1$, $\beta_{svr} = 0.1$, $SVR_{norm} = 0.01$, $SVR_{min} = 0.11$

As seen with the test cases, more than the absolute values of the reports is fundamental to compare the sensitivities of the constraint with the one of the objective function. In figure 5.15 this comparison is presented: the values of the sensitivity of the pressure are quite low compared to the one of the material indicator, which is absolutely coherent if we consider the initial condition from which the optimization starts (the pressure constraint is widely satisfied), while the one of the mass flow is higher but only of one order of magnitude. Based on the results of the previous test cases and considering that the values of the constraint function will rapidly increase in the optimization process, these gaps will be fulfilled when the pressure reaches (and overcomes) the maximum allowable value and the resulting sensitivity will therefore be considered as equal as (or even greater than) the one of the objective function. Beside the numerical values, it is also interesting to compare qualitatively the fields of the sensitivities: apart from particular cases, the two fields appear to be one the opposite of the other, since the cells that would lead to an optimization of the objective function would decrease the satisfaction of the constraint. Although this can be quite obvious, it is the formal representation of the idea behind the constrained-optimization and clearly depicts the importance of the correct computation of the sensitivity to avoid a over-constrained (or under-constrained) optimization.

The analysis satisfied the OCC stopping criteria (even if with some oscillations with the SVR constraint, fig 5.17) after just 57 optimization iterations (which on a 6-core Intel CPU i7-9^{th} gen @ $3.80GHz$ required $\approx 22hrs$) thanks to the particular initial condition given to the Material Indicator function, with a final topology that incorporates the basic characteristics of the base geometry (a lobed-like shape) but favouring a more vertical development of the lobed part with respect to the lateral one, adding a huge flow deflector-like component at the very beginning section of the exhaust that accelerates the primal flow and adding a new feature to the geometry: a two horn-like shape at the top end part of the lobed geometry (fig. 5.16). To better understand the improvement and the effect of the new topology, a model reconstruction was performed both whit a smoothing of the surface and removal of the smallest features and with a hand-made CAD model (fig 5.18).


Normalized Sensitivity of Mass_Flow_Secondary_CF w.r.t Material Indicator
-0.217
0.0873
0.392
0.696
1
Normalized Sensitivity of Turbine_Pressure_CF w.r.t Material Indicator
-0.412
-0.308
-0.205
-0.102
0.00118

(a)

(b)

Figure 5.15: Sensitivity values for the objective (a) and pressure constraint (b) report. Values are normalized w.r.t the greatest value.

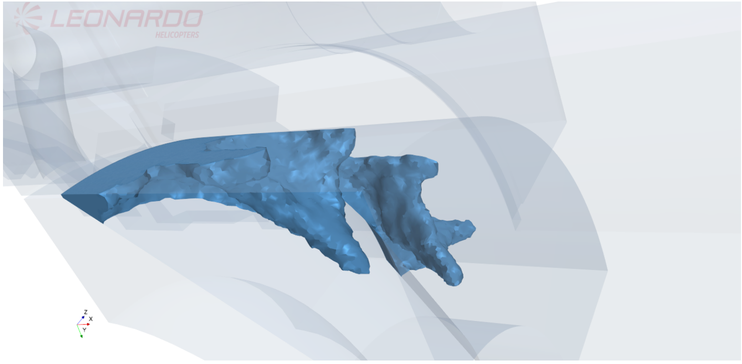


Figure 5.16: Optimized topology - Quarter of domain

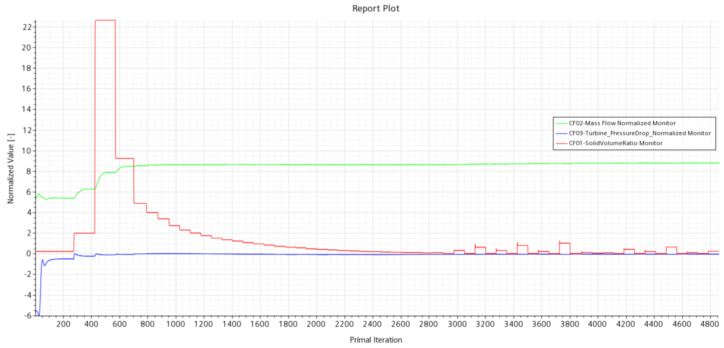


Figure 5.17: Convergence plot - objective function and constraint reports

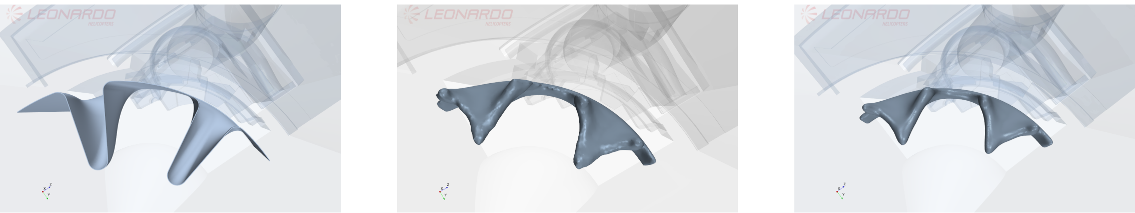
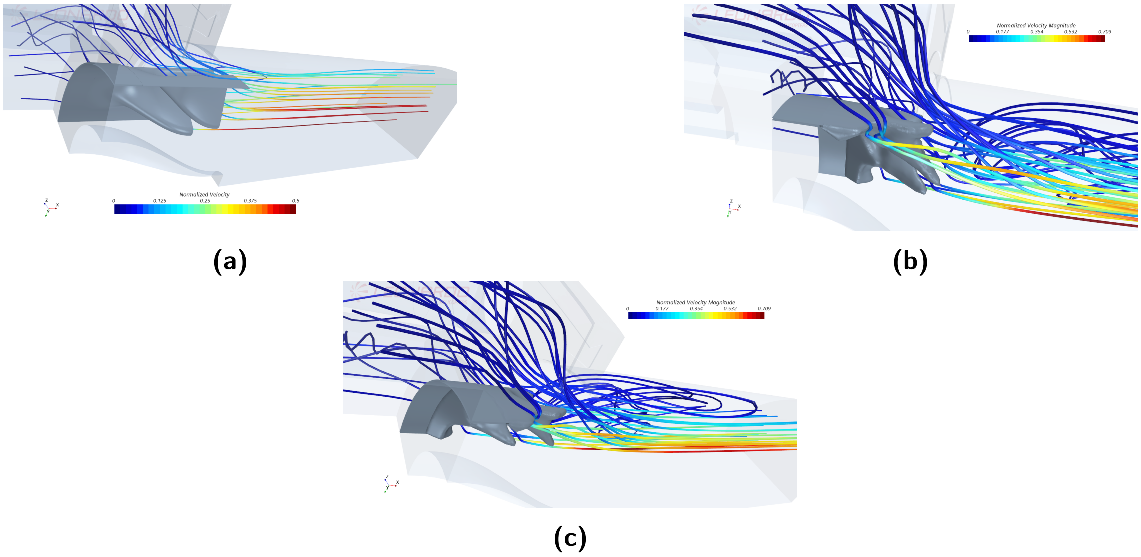


Figure 5.18: Comparison between Base (a), topology reconstructed (b) and CAD (c) geometry of the primal exhaust

Table 5.3: Results comparison: Quart Domain - Mass flow maximization case. Values are normalized with the Base Geometry values.

	Mass flow secondary [-]	Turbine Pressure [-]	BackPressure Coefficient [-]
Reference geometry	0.344	-0.922	0.37
Opt. Topology Physics	0.428	2.28	-0.93
Reconstructed geometry	0.4	-0.165	0.07
CAD geometry	0.395	-0.155	0.06

The improvement in the objective function is clearly evident (+ 15 %) while the increase in the pressure, and therefore the reduction of the back pressure coefficient, is clearly lower than the one predicted by the topology physics. An in-depth analysis of the flow field depicts the enhancement made by the new topology: underneath the horn-like shape, a low-pressure zone in generated thanks to the acceleration of the primal flow, increasing the suction effect on the secondary flow and creating the required space for its acceleration (fig.5.19c), while shrinking the lobed-region entrance section for the remaining flow, channeling it into the main one, which in turn is accelerated by the obstruction generated by the flow deflector at the beginning of the exhaust. The so-defined lobed feature, on the other hand, helps in the mixing of the flows but the lack of an horizontal development of the geometry introduces a huge recirculation zone just after the exhaust itself (fig. 5.19b) which leads to a less uniform flow at the outlet section.



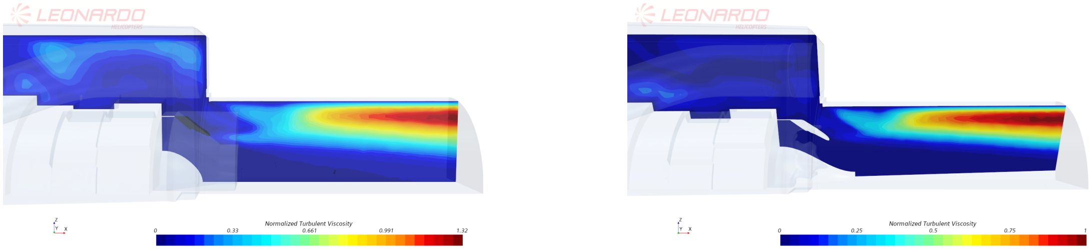
(a)

(b)

(c)

Figure 5.19: Comparison of the flow behaviour between the Base exhaust (a) and the optimized one (b) (c). Values are normalized with the greatest one.

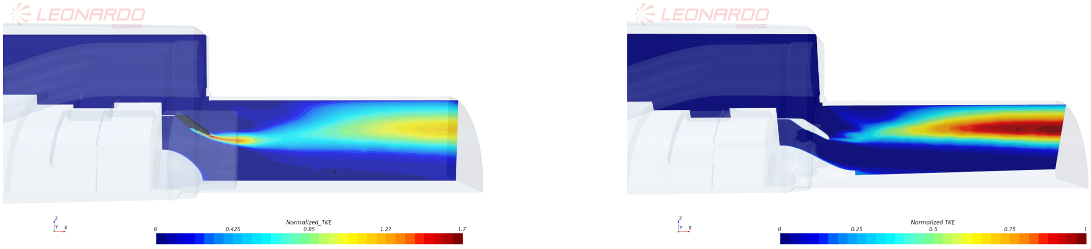
Even though the final topology is capable of such improvement in the objective function, the error introduced by the Brinkman penalization model was higher than the one expected by the previous analysis. The reason of such discrepancy may rely on the turbulence field: as we already know, the porous media is not capable of correctly capturing the turbulence development, but by now all the test cases analyzed had a lower Reynolds number. Considering the flow exiting the turbine stage as reference velocity and the diameter of the secondary exhaust as length scale we obtain $Re = 2.5 \cdot 10^6$, which is an order of magnitude greater than the one studied in the IPS test case and two orders grater than the Mass flow splitting test case. Moreover, the porous media is inserted exactly in the middle of the high-velocity field, leading to an enormous turbulence production, as clearly shown in fig. 5.21: the porous media, both for the roughness of the surface and for the incorrect wall-treatments, is the main source of turbulence in the domain which is nearly absent with the rebuild geometry. The final turbulence viscosity field correctly captured the secondary exhaust flow separation but the huge production in the lower part of the porous media generates a high-turbulent flow that, mixed with the already turbulent one, leads to an overestimation of the turbulence viscosity ratio (fig. 5.20) in the development inside the secondary exhaust (up to + 25%). This results in a rise of the pressure losses and of the back pressure at the turbine section, which is overestimated by more than an order of magnitude.



(a)

(b)

Figure 5.20: Comparison between the turbulence viscosity field with the exhaust simulated as porous media (in black) (a) and the one of the real geometry (b). Values are normalized with the greatest one.



(a)

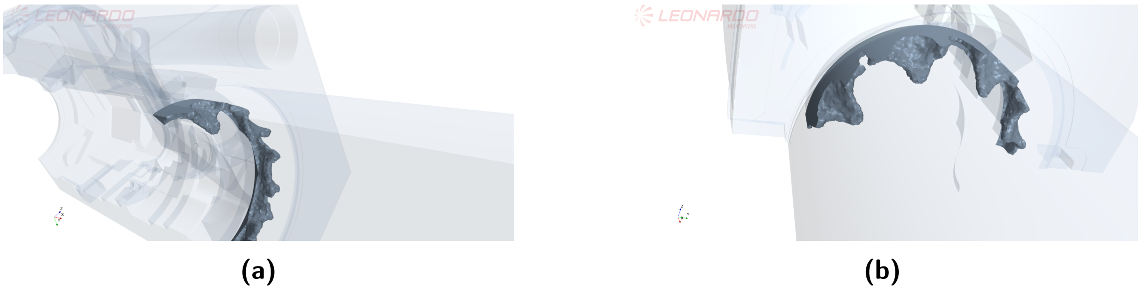
(b)

Figure 5.21: Comparison between the turbulence kinetic energy with the exhaust simulated as porous media (in black) (a) and the one of the real geometry (b). Values are normalized with the greatest one.

Beside those considerations and the error introduced, the optimization managed to converge to a realistic and reasonable geometry with a correctly satisfaction of all the constraints, setting the stage for the Full-Domain optimization.

5.2.2 Mid-Domain optimization

With the acquired knowledge and the correct optimization settings obtained in the previous analysis, the domain is now enlarged to a half of the engine bay. This analysis aims to verify the scalability and the robustness of the problem and of the settings with respect to a similar but not equal geometry, and to understand the effect that the 3-D structures before destroyed can have in the final geometry. Since this optimization is exactly the same of the previous one, apart from the geometry, and has the purpose of confirming the correct settings, we will not go through all the details previously analyzed. but we will focus only on the geometry obtained and on the numerical results.
The first and most important result is the final topology, shown in figure 5.22: it is interesting to notice that the optimization seems to converge to a topology very close to the one obtained with the Quart-domain, showing that the simplification introduced in the domain did not affect the general behaviour of the flow. However, some small features that are based on a particular flow development can be found, especially in the horn-like shapes between the two middle lobes and the external ones, where are slightly defeatured and a smaller space between the lobe is left for the flow (fig. 5.22b).



(a) (b)

Figure 5.22: Final topology - Mid Domain optimization

The effect of these small features is shown in fig 5.23 and 5.24, where the reconstructed geometry was analyzed: the smallest gap between the outer lobe help the primal flow in its development, with a reduced influence of the secondary one and a less increasing in the pressure. Moreover, the secondary flow entering the exhaust suffers from some recirculation regions that reduced the x-component of the velocity, which is recovered with the flat surface between the lobe. This behaviour is confirmed analyzing the streamlines in the lobed region (fig. 5.24a), where it is clear that the secondary flow do not enter in the primal flow region but is simply driven into the secondary exhaust. For what concerns the lobes in the middle, the behaviour is nearly the same of the one obtained with the Quart domain topology, with the horn-like shapes that allow for the secondary flow to enter in the primal one (fig. 5.24b).

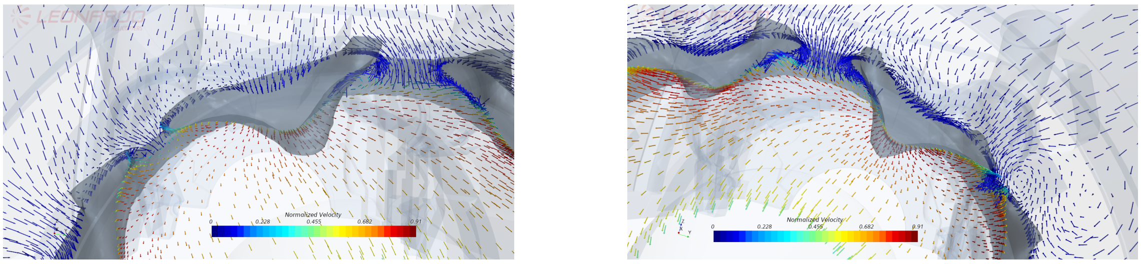
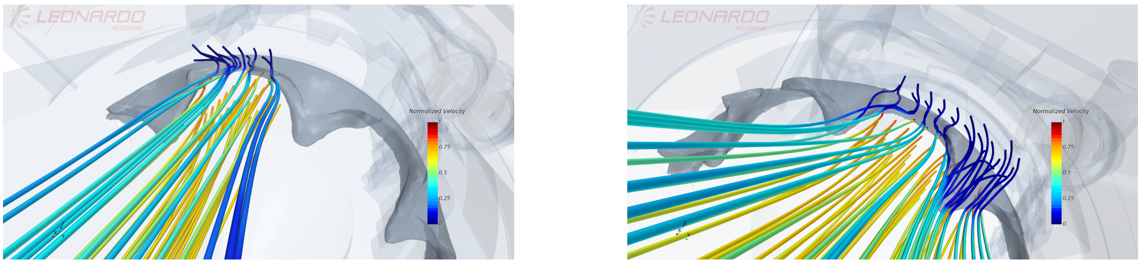


Figure 5.23: Velocity field in the lobed region. Comparison between the outer and the middle lobes - Mid Domain case. Values are normalized w.r.t the greatest one.



(a)

(b)

Figure 5.24: Streamlines comparison between the outer (a) and the middle (b) lobes - Mid Domain case. Note how the secondary flow enters the primal exhaust in case (b). Values are normalized w.r.t the greatest one.

As done for all the previous simulations, the optimization is validated with the reconstructed geometry. From the results reported in tab 5.4 the improvement in the objective function seems to be confirmed with the Mid-Domain optimization (+16%) while there is a small discrepancy in the pressure: here, the turbine pressure is greater than the one obtained in the Quart-Domain, which maybe is the reason why the topology has the above depicted differences in the lobed features. However, the predicted value from the topology physics seems more coherent with respect to the one computed in the Quart-domain analysis, in confirmation of the un-predictable error introduced. However, since the constraints are fully satisfied and the objective function increased, the optimization fully succeeded.

Table 5.4: Results comparison: Mid Domain - Mass flow maximization case. Values are normalized according to the Quart Domain case.

	Mass flow secondary [-]	Turbine Pressure [-]	BackPressure Coefficient [-]
Reference geometry	0.562	-0.953	0.39
Topology Physics	0.76	1.71	-0.7
Reconstructed geometry	0.65	-0.031	0.013

5.2.3 Full-Domain optimization

The reduction of the domain to just a quarter, with the introduction of two symmetry boundary to simulate a two-axis symmetry behaviour of the flow field, reduced

the correct 3-D development of the flow inside the engine bay influencing the final topology obtained by the optimization process. The switching to a Mid-Domain geometry shown that some small features can be generated due to the different 3-D flow behaviour, but the general topology is the same. In this section the full geometry is analyzed, but with a new hypothesis introduced: in order to reduce obtain a geometry that was not strictly dependent on the smallest features in the engine bay, so that the new topology could be suitable for a more general working condition, and to reduce the cells number, the engine bay was defeatured and the new geometry is shown in figure 5.25. With the proper report scaling obtained by the sensitivity analysis in the reduced domain, the solver is therefore tested, to verify whether the simplified domain has an influence on the geometry or not. Beside the mere purpose of obtaining a complete 360° geometry, this analysis aims to verify the effective sensitivity of the solver to the different 3-D flow behaviour: we already stated (chapter 4.2) that 3-D structures strongly influences the topology that the solver aims at, but it is fundamental to investigate if and how small changes in these structures play an important role.
The optimization is based on the previously depicted definitions and characterizations, therefore only the results will be presented, focusing on the analysis of the obtained topology and on its influences on the flow behaviour. In order to have a coherent comparison with the base geometry, the obtained topology is reconstructed and simulated in the real and complete engine bay.

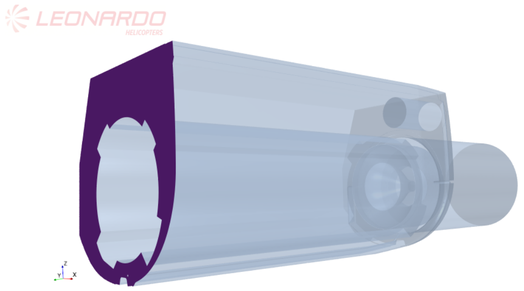


Figure 5.25: Defeatured engine bay for the Full-Domain analysis - In purple the new engine bay inlet boundary replace the air intakes

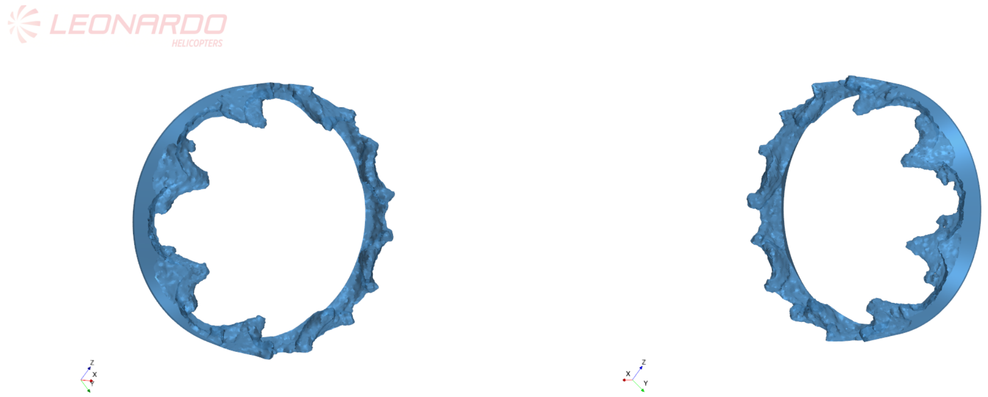


Figure 5.26: Final topology - Full Domain Mass-Flow-secondary maximization case

Starting from a qualitatively analysis of the final geometry obtained (fig. 5.26), the switch to a full 360° geometry seems not to strongly modify the general topology the solver aims at, since the lobed geometry with the two horn-like shape is still reproduced in all the 8 lobed features obtained. A closer analysis (fig. 5.27) however reveals that the latter, together with the horn-like shapes, are slightly shorter and, therefore, with a less invasion on the primal flow development. This behaviour may be due to the satisfaction of the pressure constraint, since a closer analysis at the convergence of the reports (fig. 5.28) reveals that it nearly reaches the maximum allowable value after just few iterations. On the opposite, the SVR constraint at the very first iteration leads to a removal of almost all the porous media generated by the particular initial condition (the same behaviour is visible in the Quart-domain simulation (fig. 5.17) but with a much lower strength), sign that the distribution of the solid domain is strongly penalizing the objective function. By the way this is not a surprise: if we consider the value of the sensitivity of the mass flow with respect to the material indicator function, all the cells above the exhaust are highly favorable for the increasing of the objective function, while there is nothing in the pressure sensitivity that is strong enough to counteract (fig. 5.29). Moreover, since we start with such a particular initial condition, at the first iterations the Level-set interface velocity has no momentum that can help in the stabilization of the solution (see eq. 2.52), which therefore drastically increases, leading to a huge topology change. Even though the initial condition is almost lost, a small feature is retained: the lobed-liked part of the primal exhaust is not completely deleted in the first iteration (fig. 5.30), which is clearly an indication of the positive effect that it has on the objective function, and which is the new "initial condition" of the solver from which the porous media grows and develops. Except from the smaller dimension of the lobed shape, from the topology point of view the final geometry does not show other important differences with the simplified case .

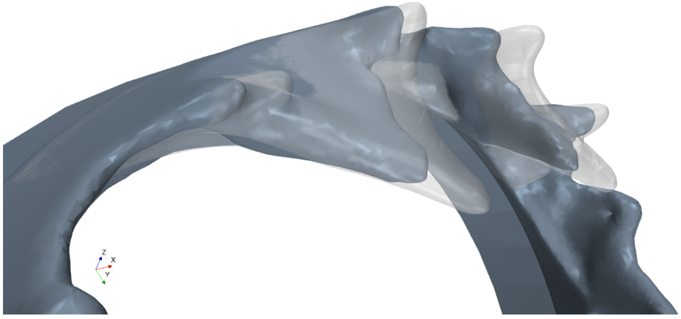


Figure 5.27: Comparison between the Quart-Domain (in transparency) and the Full-Domain topology result

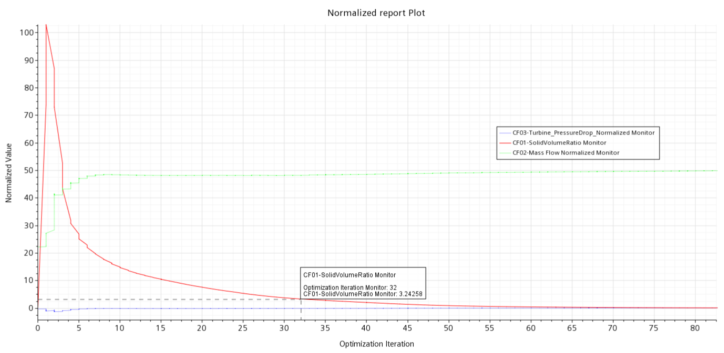
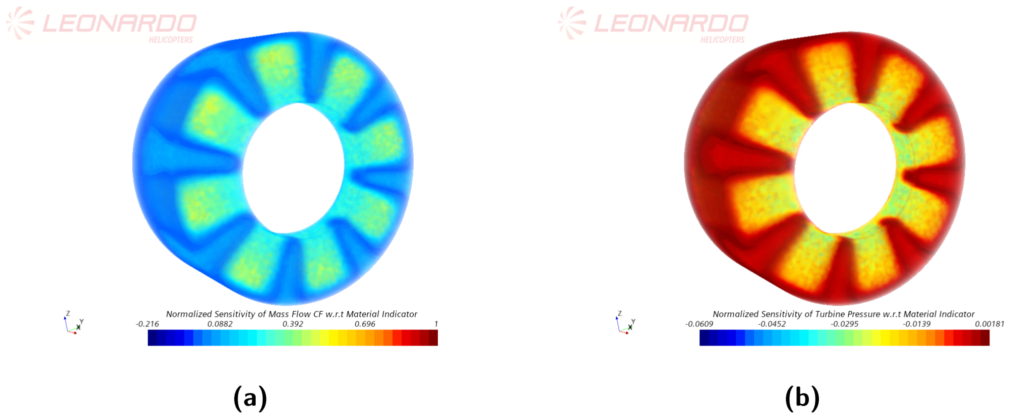


Figure 5.28: Convergence plot - objective function and constraint reports. The solution converged after 82 iterations ($\approx 4hrs$ on 48 cores)


LEONARDO
HELICOPTERS
Normalized Sensitivity of Mass Flow CF w.r.t Material Indicator
-0.216
0.0882
0.392
0.696
1
LEONARDO
HELICOPTERS
Normalized Sensitivity of Turbine Pressure w.r.t Material Indicator
-0.0609
-0.0452
-0.0295
-0.0139
0.00181

(a)

(b)

Figure 5.29: Sensitivity values for the objective (a) and pressure constraint (b) report at the first iteration - Full Domain case. Values are normalized with the greatest one.

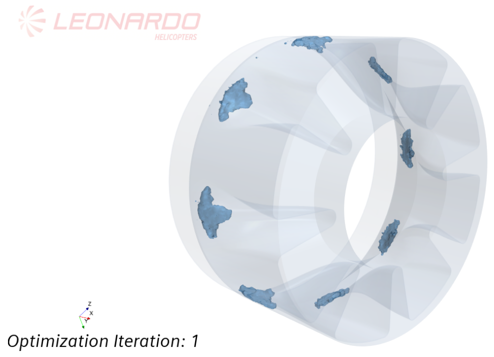


Figure 5.30: Isosurface of the porous media after 1 optimization iteration compared with the base geometry of the exhaust

In order to verify the constraint satisfaction, and to quantify the improvement in the objective function, the topology depicted by the porous media was reconstructed and re-analyzed with the creation of a CAD model (fig. 5.31). As reported in tab 5.5, the optimization succeeded in an improvement of the objective function of about 15.5%, which is in line with the Quart-Domain and Mid-Domain tests. However, the turbine pressure suffers from a huge increment that reflects on a BackPressure coefficient value, which drops down to -0.11, which is still greater than the minimum allowable and it is fully coherent with the constraint impose. A possible cause of the pressure increasing can be the CAD model itself: in the hand-made reconstruction of the geometry, such particular features like the horn-like shape or the correct lobed development are difficult to reproduce as they are. Even though the geometry is very close to the predicted one, especially in the horn-like shape the discrepancy is too high that the flow development in that region is highly affected. The comparison with the values predicted by the porous media in this case in not coherent, since the geometry on which the topology was performed was different.

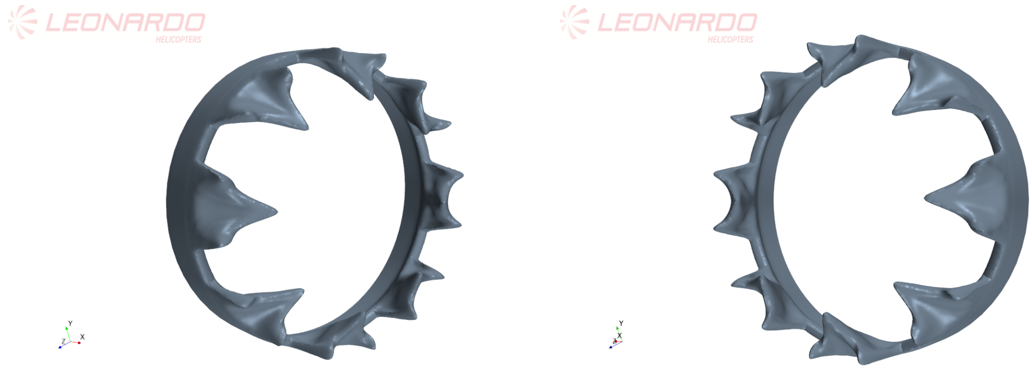


Figure 5.31: Final topology - hand-made CAD model based on the topology result

Table 5.5: Results of the optimization - Mass Flow Secondary maximization. Values are normalized according to the Quart Domain case. In red the results of the topology physics, since referred to the defeatured geometry

	Mass flow secondary [-]	Turbine Pressure [-]	BackPressure Coefficient [-]
Reference geometry	1	-1	0.41
Base Geometry with Topology	1.54	0.232	-0.1
Topology Physics optimized	2.577	2.36	-0.95
Reconstructed geometry	1.155	0.277	-0.11

Comparing the flow behaviour between the correct geometry and the CAD model in figure 5.32, is possible to understand why the turbine felt such an increasing on the pressure with the latter: in the topology predicted by the optimization, the horn-like are made for generating a low-pressure region at their bottom that deviates the primal flow and recall the mass flow of the secondary exhaust, in a very smooth way. On the contrary, with the CAD model the primal flow goes straightforward to the horn-like shape, which in this way behave like an obstruction in the flow development: the result is a rapidly increment on the velocity on the primal flow which increases the pressure on the turbine section. It is possible to fix such a problem with a more precise geometry reconstruction or with an optimization of the obtained one using, for example, a Shape Optimization. An ultimate consideration can be made for the general flow development in the secondary exhaust; as we saw in the Quart domain case, the so-shaped primal exhaust is capable of an high suction of the secondary mass flow, but the obtained flow shown a huge separation and recirculation zone at the beginning of the secondary exhaust. From fig. 5.33 we can see that this behaviour is also obtained in the Full-Domain case: the secondary flow suffers of a huge separation which is recovered only near the outlet section, reducing therefore the mixing of the two flows, even with different horn-like features that generates a lover axial component of the primal flow. The ultimate effect is a high-velocity region in the middle of the exhaust bounded by a near-null velocity field in the outer part of the exhaust (fig. 5.42 in chapter 5.4).

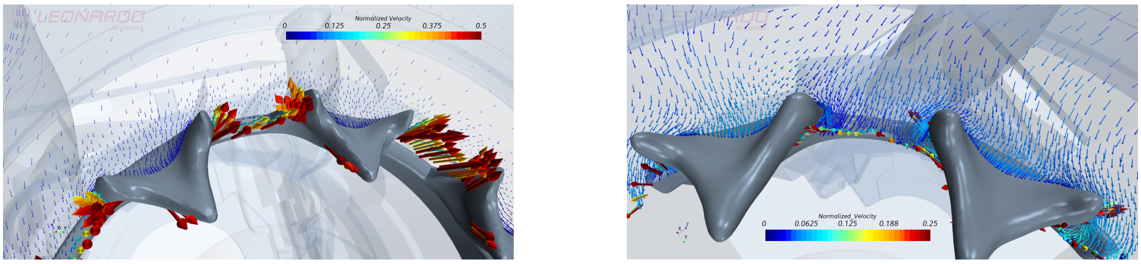


Figure 5.32: Differences in the velocity field near the horn-like shapes with the final CAD (a) and with the reconstructed topology of the Quart-Domain (b).

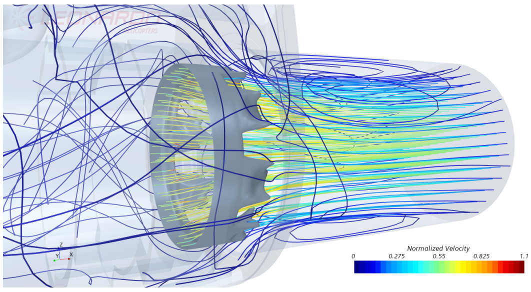


Figure 5.33: Flow behaviour with the optimized primal exhaust - Mass Flow optimization case. Note the huge recirculation generated in the secondary exhaust-

5.3 Turbine Pressure optimization

The optimization proposed before had the primal goal of maximizing the mass flow of the secondary exhaust without generating a too high pressure on the turbine exhaust stage. Despite the optimization successfully succeeded in maximizing the objective function while complying with the imposed constraints, the resulting pressure on the turbine section turned out to be much higher than the base configuration. A possible solution could be to repeat the optimization with a more stringent constraint on the pressure itself and evaluate the result; the error introduced by the physical model, however, as repeated several times, makes it difficult to determine which values would be more appropriate and could therefore require many tests before obtaining a more acceptable value. A second solution is to carry out a new optimization inverting objective function and constraint: in this chapter we follow this approach, i.e. we will research for a geometry capable of minimizing the pressure at the turbine section while keeping the mass flow of the secondary exhaust as close as possible to the base value. Thanks to the acquired knowledge of the sensitivities from the previous optimization, this case was studied directly in its Full-Domain characterization.
Contrary to the expectations, this optimization did not manage to converge as well as the previous, and did not reach a ready-to-use geometry. According to fig. 5.34, even with the new objective function, in the first iteration the solver removes a huge part of the initial condition, which drastically reduces the pressure but also reduces the mass flow of the secondary exhaust. In the following iterations the mass flow constraint (which, accounting for the error introduced by the porous media, was increased of about 30 %) is fully recovered together with the SVR which slowly decreases down to the desired value. The optimization however required more iterations for the convergence, which is never fully reached (the *OCC* criterion was never satisfied); in fact the SVR reports start oscillating in the very last iterations, sign that the solver did not manage to find the optimum solution. The optimization stopped when the maximum allowed optimization iterations (130) are reached.

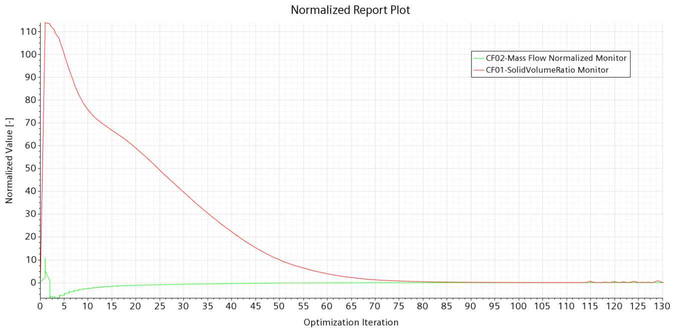


Figure 5.34: Convergence plot - Pressure turbine minimization case - objective function and constraint reports. The solution stopped after 130 iterations ($\approx 7hrs$ on 48 cores)

The difficulty in reaching an optimal solution has an impact on the topology obtained, which is not so well-defined as the one obtained with the Mass flow optimization (fig. 5.35). Due to the particular given initial condition, the solver converged to a geometry that is not a single portion of solid but is made of more pockets of solid closed to each other, which is not straightforward to reconstruct. The solver itself cannot create isolated solid pockets inside the domain, but the growth from cells already characterized as solid is allowed: setting such particular initial condition implicitly allows the possibility of isolated porous cells inside the fluid domain. Despite this, an idea of geometry is given, therefore a CAD model was created starting from the suggested geometry (fig. 5.37), in order to analyze the possible improvement in the objective function.

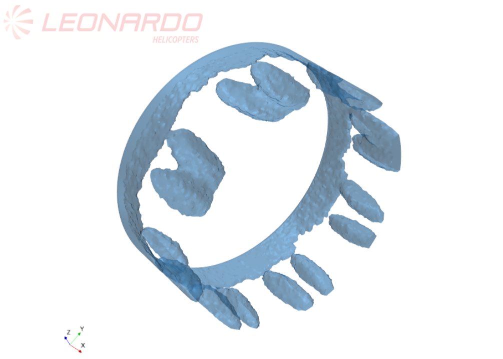


Figure 5.35: Final topology obtained - Pressure minimization case

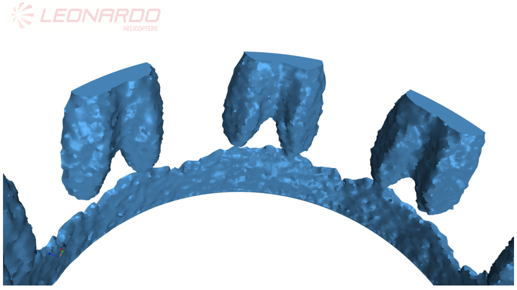


Figure 5.36: Detail of the final topology - Pressure minimization case.

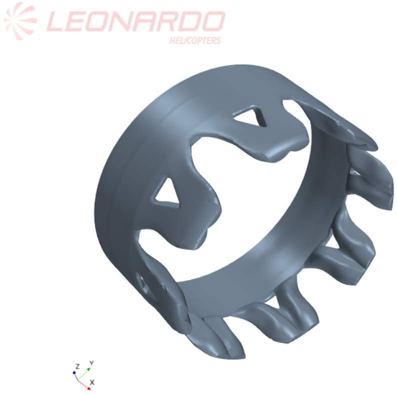


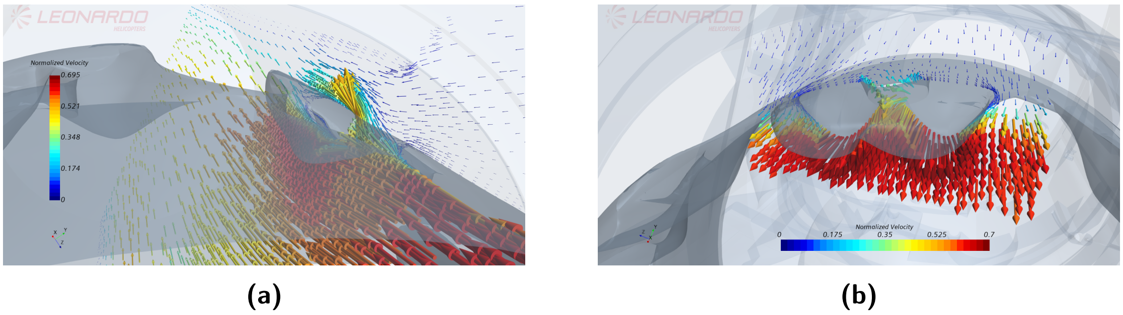
Figure 5.37: CAD model of the optimal solution - Pressure minimization case.

Table 5.6: Results of the Pressure turbine minimization optimization. Values are normalized according to the previous cases.

	Turbine pressure [-]	Mass flow secondary [-]	BackPressure Coefficient [-]
Base Geometry	-1	1	0.41
Base Geometry with Topology	0.232	1.54	-0.1
Topology physics optimized	-0.283	1.37	0.12
Optimized Geometry	-1.12	0.93	0.47

The optimize topology depicts an improvement in the objective function of about 12 % while the reduction in the mass flow of the secondary exhaust is contained at a -7.5 %. It is interesting to notice that the solver aims to improve the mass flow of the secondary exhaust far beyond the minimum allowable value, which is the objective of the so-defined constraint reports: the negative part of the report is an indication to always search for an improvement (if possible) and not to settle for the minimum satisfaction possible.

The obtained geometry deserves an in depth analysis of its new features: first of all the vertical development of the lobed-shape is completely deleted and replaced with a purely longitudinal development with the generation of a sort of A-shape. Focusing on the bottom side of these new features (fig. 5.36), from the edge of the triangular hole starts a groove that develops until the end of the lobed feature and digs into the domed shape. To better understand the functionality of such features, a closer analysis in the flow field in their proximity is presented in fig. 5.38: the triangular hole channels the primal flow into the secondary one, reducing the mass flow that undergoes in the lobed region and, therefore, reducing the pressure on the turbine section. The same flow however, filling part of the secondary exhaust, obstructs the suction of the mass flow from the engine bay. The groove, on the other hand, channels part of the primal flow, driving it into the main exhaust section and avoiding (or at least reducing) the separation of the same (fig. 5.38a and 5.39). Both these two features help reducing the pressure at the turbine section, while the domed shape seems to reproduce the effect of the horn-like shape obtained in the mass flow optimization, recalling the secondary flow and improving its mixing with the primal one.



(a)

(b)

Figure 5.38: Velocity field near the A-Shaped feature of the optimized exhaust

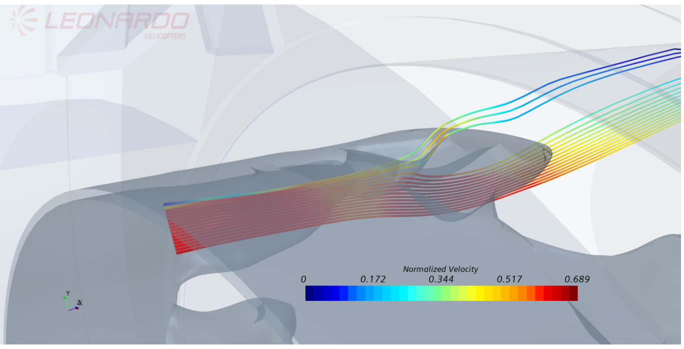


Figure 5.39: Effect of the triangular hole on the primal flow

5.4 Velocity field comparison

An important function of the primal exhaust is mixing the gasses outgoing the turbine with the one coming from the engine bay in the best possible way, to ensure not only their proper expulsion, but also to reduce the noise generated by the engine itself. Even if this parameter was not considered during the optimization, analysing this aspect can better highlight the differences between the obtained configurations and the base one; this analysis was based on the comparison of the velocity and temperature profiles at the outlet section and on the velocity field at different stages of the secondary exhaust.
Let us focus on the velocity and temperature profiles reported in figure 5.40 and 5.41: The Mass-Flow-optimized configuration generates a high-velocity region in the middle of the exhaust while in the external part the velocity drops down drastically, with values that are much lower than the ones of the base configuration. This translates in a very non-uniform flow at the outlet section and a bad mixing performances, confirmed also by the temperature profile: the low-temperature region in the outer zone of the exhaust is the sign that the mass flow of the secondary exhaust does not mix well with the high-temperature one exiting the turbine. On the other hand, the Pressure-optimized configurations seem to correctly manage in the mixing of the flows, with a slightly higher velocity compared to the base configuration, due to the less pressure on the turbine section. However, from the temperature point of view, not such uniformity is reached. This result also confirms what we have seen in the previous chapters, as it was immediately clear that the configuration optimized for the mass flow secondary generated a huge recirculation zone that degrades the mixing performance, while the blowing of primary flow in the secondary one obtained with the second configuration energizes the flow and keeps it more attached and uniform in the outlet region.
The correct mixing performed by the latter configuration is also confirmed by the analysis of the sections of the velocity field (fig. 5.42): the triangular-shaped hole generates the above mentioned blowing in the secondary flow, which helps in its acceleration and mixing too. Such improvement cannot be found in the velocity sections of the Mass-Flow-optimized configuration: taking into account the behaviour of the flow in the primal exhaust section previously analyzed (fig 5.33), it is again clear that the two flows join in a single reduced section, with the remaining zone of the secondary exhaust that is characterized by a high recirculating region that includes the main flow itself. This behaviour may be the main cause of such an increasing in the turbine pressure and a further optimization can help in its reduction (a Shape optimization in this case would probably be the better choice).
On the basis of this comparison, we can affirm that both configurations have brought a notable increase of performance in the respective objective function, but the pressure-optimized configuration (although it has not arrived to convergence as much as mass-flow-optimized one) results in a more performing geometry also on other aspects and points of view, resulting therefore more interesting for a real implementation.

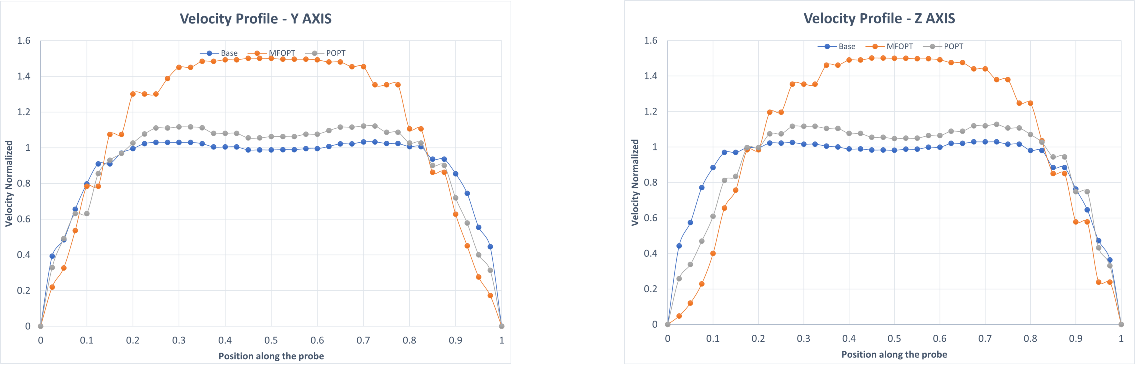


Figure 5.40: Velocity profile at outlet section - Comparison between the obtained configurations

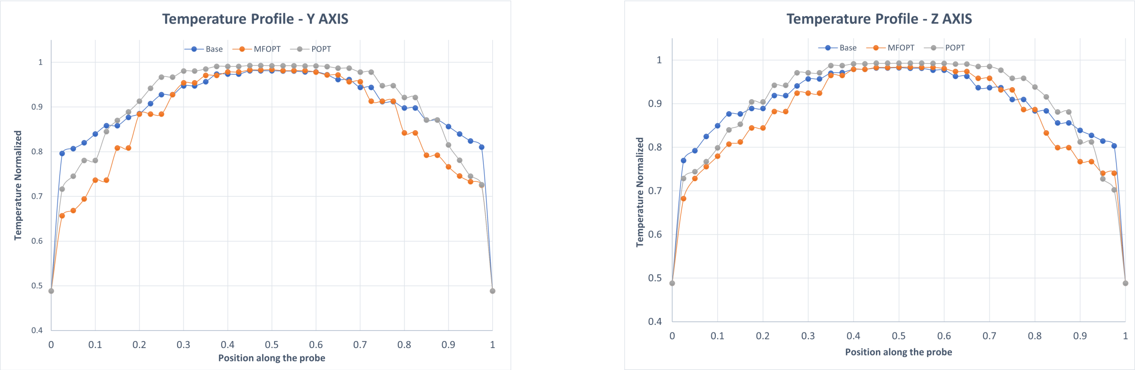


Figure 5.41: Temperature profile at outlet section - Comparison between the obtained configuration

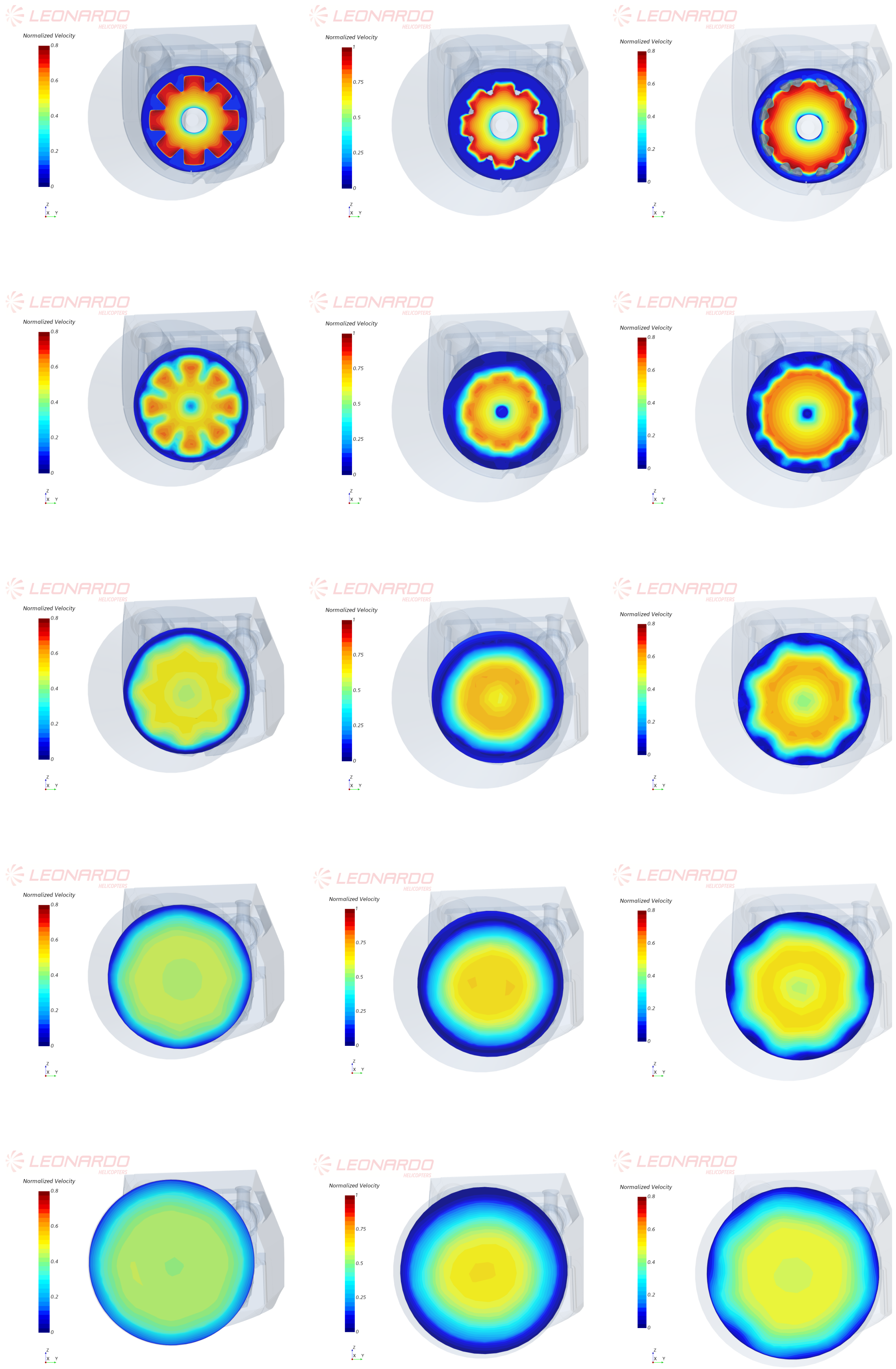


Figure 5.42: Comparison of the flow field in different sections of the exhaust - Base geometry, Mass-flow optimized and Pressure optimized. Values are normalized with respect to the biggest one.

6
Conclusions

The goal of this thesis was to analyze and explore discrete adjoint of the compressible Navier-Stokes equations coupled with the level-set method for the topology optimization. Starting from the correct definition of the topology region, we moved to the analysis of the mesh characteristics, both from the model and the size point of view, to quantify the influence that it have on the algorithm and on the result. Throughout this analysis, we have seen how using a fully-converged mesh is absolutely not necessary, since the optimization managed to converge also with a coarser definition of the domain, as long as it is fine enough to correctly capture the main fluid-dynamic characteristics and the general flow behaviour. Trough simple test cases, we moved to the evaluation of the optimization loop itself, with a particular focus on the sensitivity analysis: those are the real driving forces of the algorithm, so it is necessary to perform it correctly. In order to let the optimization to converge at the optimal solution, with the satisfaction of all the imposed constraints, it appears necessary that the different sensitivities are correctly scaled one with the others. At this purpose, different possibilities of normalization were tested, starting from the simplest possible: the scaling of each sensitivity with its maximum value (also called "automatic normalization"). This approach was compered to the "manual normalization" one, which aims to normalize the sensitivities with a scaling and a proper definition of the constraints' reports. From this analysis, it was clear that the first approach is extremely sensible to the weight assigned (via the *Penalty* parameter) at their satisfaction (which, by the way, is unique for all the constraints): a too high weight can lead the solver to not feel the objective function, while the opposite can make it blind to the constraints themselves. In both cases, the penalty value is not so easy to guess a priori and in order to find the most suitable one a series of optimization may be required; moreover, with this approach the optimization seems to take into account a constraint only when it is not satisfied, without having a real indication of the distance between its value and the limit one. The consequence is the introduction of oscillations, even huge ones, in the solution and the consequent divergence of the optimization. The manual normalization is therefore introduced to avoid (or reduce) this behaviour, with a particular constraint definition based on the

characterization of the same with a piecewise-defined function which is continuous and differentiable. The latter is the fundamental property of this function, since it allows us to give to the solver a gradient which is capable of representing the distance between the value of the report and its limit value, also when the constraint is satisfied. The true challenge with this definition is in the correct normalization of such functions, since the sensitivity values strictly depends on their values: with this approach, indeed, the sensitivities are not modified after their computation. An analysis of the parameters that make up such functions was indeed necessary to understand how they reflect on the optimization itself; the results of such analysis highlight how the objective function often requires higher sensitivity values with respect to the one of the constraints, which must increase and reaches the same order of magnitude when the constraint approaches the limit value.
A separate discussion is deserved for the constraint on the amount of solid required for the optimization (the SVR constraint). Even it can appear counter-intuitive, the solver need an indication on the ratio of solid that it should achieve, since the objective function alone seems not enough for the porous media to develop inside the domain. Being such a value unknown most of the time, it is clear that more optimization with different values of the constraint may be required, leading to the analysis of the different geometries obtained.
After the sensitivity analysis the focus was posed on the comparison between the results predicted by the topology physics and the one really achieved. This clearly shown how the characterization of the solid domain trough the porous media introduces a not negligible error in the numerical solution, which makes it mandatory to reconstruct the geometry to validate and quantify the real improvements. In particular, it shows a lack of a correct turbulence modeling, both inside and at the boundaries of the porous media. In the first case, even tough the porous is characterized with a momentum penalization (forcing the velocity field near 0), no turbulence corrections are implemented, leaving the same to propagate and diffuse inside the "solid" domain. Apart from the completely un-realistic turbulence field obtained inside the porous media, which seems not to influence the final topology, this diffusion can affect the outer flow behaviour, leading to a high numerical error: the final consequence is that it gets hard to predict the physical quantities obtained during the optimization and so the limit value of the imposed constraints. A first correction proposed was to force to zero the turbulent viscosity in all the cells with a Material Indicator < 0.5, which actually reduced the above depicted error, but did not manage to delete it. To completely remove the error,is required the introduction of a particular turbulence model that can (more) correctly predict its development in the first fluid cell near the porous media: since its boundaries are not characterized as "walls", and without the possibility of generating a correct mesh around them, the actual turbulence model is not suitable and, in general, its production is overestimated; the consequence is an overestimation of the physical quantities, like pressure losses, temperatures or mass flows.
Based on the acquired knowledge, in the last chapter the optimization is applied to a complex aerodynamic scenario: the primal exhaust of an aeronautical engine. The main difference between this optimization and the previous made is in the definition of the topology region, which in this case is completely surrounded by the fluid domain and not viceversa. This depicts the necessity for the optimization to start from a particular initial condition for the porous media generation, since no

physical walls are inserted. This, in according with the correct report definition, leads to a positive convergence and to the generation of new improved geometries, validated and confirmed with the analysis made on the reconstructed CAD models. In conclusion, it is clear that the optimization algorithm, if set correctly, is capable of converge to a optimal solution which satisfies all the imposed constraints in a smooth and confidence way. However, due to the introduced error, a geometry reconstruction appears to be absolutely necessary to verify the real goodness of the new topology.

6.1 Future Developments

According to the analysis here presented, it is clear that the topology optimization applied to aerodynamic elements and to a more complex fields compared to the structural one, such as the fluid-dynamic field, is still an immature and under development technology. The methodology here proposed can provides non-trivial optimizations from the topology point of view, but it completely not able in solve the proper turbulence development of the new component. The need for such an improvement reflects not only in the possibility of directly evaluate the performances of the new geometry, but also allows for a better and conformal constraint definition, which by now must be over-estimated without a clear and unique indication of such overestimation.

Moreover, the procedure for the report definition still appears non-linear, without the possibility of predict their effect on the sensitivities; by now, their definition is based on an initial guess which is performed with a series of analysis. The automatic normalization can be a starting point, but needs to be improved and modified, so that it can better quantify the constraint violation and also take into account their satisfaction, which seems to be almost neglected.

Based on the above depicted error, to correctly quantify the improvements it is obliged a geometry reconstruction, which however is anything but trivial: the topology given from the optimization is described via a threshold o the mesh based on the fictitious porosity introduced; this do not provide a solid model but generates a tessellation, whit all the issues related, and the difficulty in the interactions and modifications. These problem, however, is not exclusively related to the topology optimization, but generally covers all those analysis which works with the characterization of the cells in the domain for a geometry description.

The implementation of such correction can brake new ground and new application fields for the fluid-dynamic topology optimization, overcoming the actual (limited) applications. Exhausts, fins, compressors or turbine blades, turning vanes or aerodynamic appendices are all possible applications for these technology: their design would experience an huge improvement, since it can be highly sped-up and improved, bypassing the trial-and-error procedures or the geometry based on the knowledge of the designer or the engineer.

Appendix

A
Jet Pump Effect

A jet pump is a particular type of pump in which the energy of a fluid (liquid or gas) is transferred to another fluid via the well-known Venturi effect [47]. A high-pressure fluid is forced into a nozzle which converts its energy into kinetic energy, decreasing its pressure. The resulting low-pressure field creates a suction that drives another fluid from a reservoir or from an external domain within the primary fluid. The combined flow is then slowed down, to reconvert the energy into pressure through an exhaust or a pressure discharge (fig. A.1).

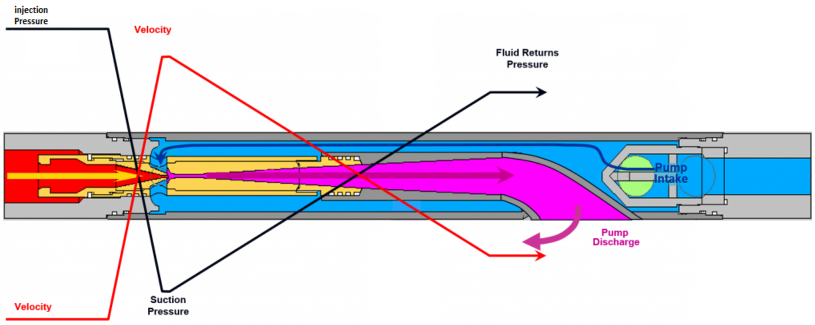


Figure A.1: Basic scheme of the jet pump principle [47]

There are two different approaches for the computational methods of the performance of a jet pump. the first and most common one uses pressures and flow ratios [26], but requires parameters that not always are known. The second approach uses experimental information, such as adimensional coefficients, to compute efficiency [22]. The efficiency of a hydraulic pressure pump (HPJ) is defined as the ratio of the energy added to the secondary fluid in relation to the energy lost by the primary one.
Let us briefly introduce the most important parameters considering the scheme of a HJP as in figure A.2.

Considering as $(.)_1$ the flow characteristics of the high-pressure flow, $(.)_2$ the flow

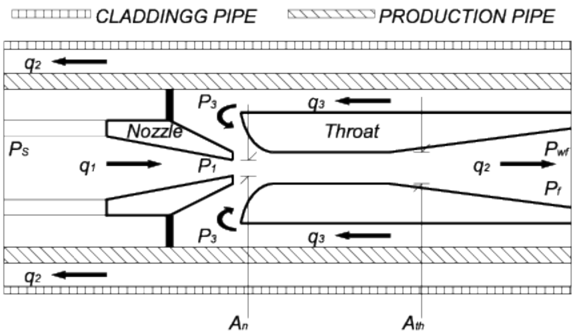


Figure A.2: Jet Pump scheme

characteristics of the mixed flow and $(.)_3$ the ones of the driven flow, we can define the input to net flow ratio M:

$$M = \frac{q_3}{q_1} \tag{A.1}$$

Moreover, we can define the dimensionless head recovery ratio H:

$$H = \frac{P_2 - P_3}{P_1 - P_2} = \frac{1 - N}{N + M} \tag{A.2}$$

Where N is a coefficient that represent the experimental information concerning the loss coefficients along the pipes and junctions and the geometry of the HJP itself. The equation for N and its derivation can be found in [22]. The pressure P_1 can be expressed as the sum of the hydrostatic pressure and the surface pressure P_s given by the hydraulic pump:

$$P_1 = h_1 G_w + P_s \tag{A.3}$$

Also the intake pressure P_3 need to be computed. Knowing the nozzle area A_n, the pressure is than computed as follows:

$$P_3 = P_1 - g\left(\frac{q_1}{1515.5A_n}\right) \tag{A.4}$$

Finally, is possible to compute the flow rate q_3 knowing an average pressure of its reservoir P_r and using a *Vogel's performance relation*:

$$q_3 = q_max\left[1 - 0.2\left(\frac{P_3}{P_r}\right) - 0.8\left(\frac{P_3}{P_r}\right)^2\right] \tag{A.5}$$

Therefore, we can compute the hydraulic efficiency of the pump as in equation A.6:

$$\eta = M \cdot H \tag{A.6}$$

If the Jet pump deals with multi-phase fluid, the cavitation problem must be take in account due to the high speed of the flow combined with the presence of gas that generates bubbles in the fluid. Since the problem of interest for this thesis deals only with gasses, cavitation is not considered but more information can be found in [8].

There are several advantages in the use of this pump, some of them are here summarized:

- first of all, this pump has no moving parts, which means that this system is reliable, requires very-low maintenance and is cheap. The pump uses only the Venturi effect so is also quite easy to design and once produced can last for a very long time before a substitution is required;
- the amount of fluid driven by the Venturi effect is controlled by the inlet pressure given to the primary fluid, so it is quite easy to control;
- since the basic principle of this pump is a simple and well-known effect, it is also easy to design and optimize for the required purpose;

These features led the jet pump to be used in a variety of applications, which the most famous and big one is on Oil Wells. The little need for maintenance make this type of pumps perfect for an underground applications where the access is usually difficult and requires not only a not negligible time but also qualified operators, leading to high maintenance costs. These applications use the energy of the high-pressurized flow to drive the gasses or the oils from their reservoir to the surface generating an *hydraulic artificial lift* [38].

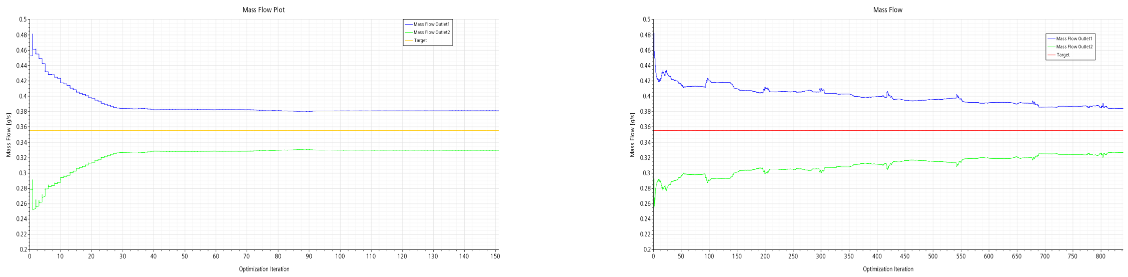
For our purpose the jet pump effect is used to drive the flow from the engine bay through the secondary exhaust, in order to improve the cooling effect of the engine and of its component. The concepts above discussed are referred to the generic and basic design of a Jet Pump; however, this is a side effect obtained with the primal exhaust, whose first task is to ensure a proper expulsion of the gasses from the turbine stages. All the concepts here reported are just for completeness and are intended as a side knowledge for the reader.

B

Mass Flow Splitting Test Cases - Images

In this appendix are reported for completeness all the plots and images referred to the analysis performed in cap 4.1.

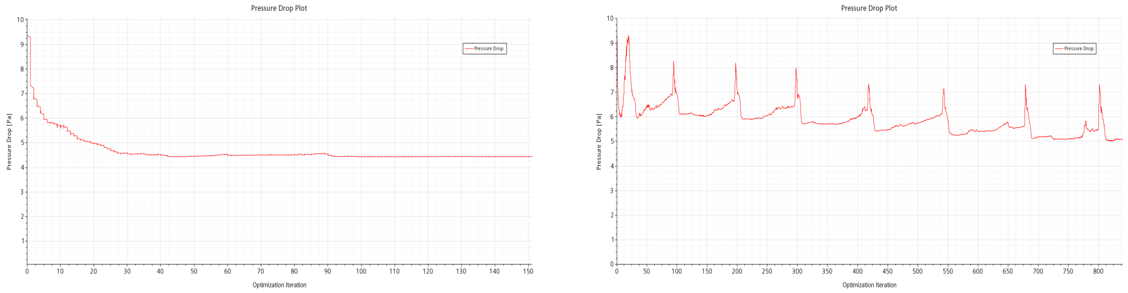
B.1 Penalty analysis



(a) Penalty = 100

(b) Penalty = 5000

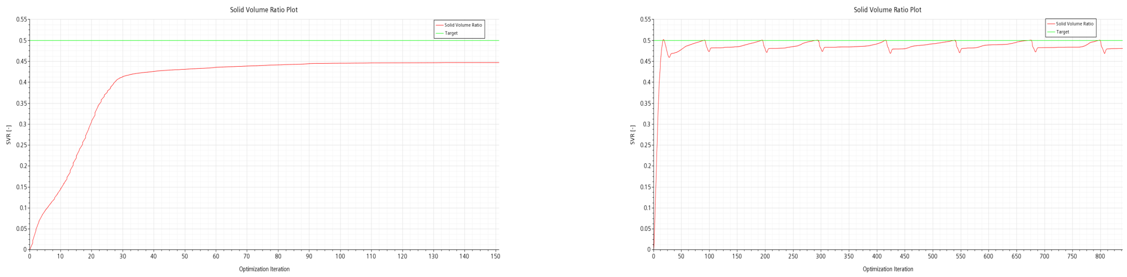
Figure B.1: Mass Flow behaviour - Penalty Analysis



(a) Penalty = 100

(b) Penalty = 5000

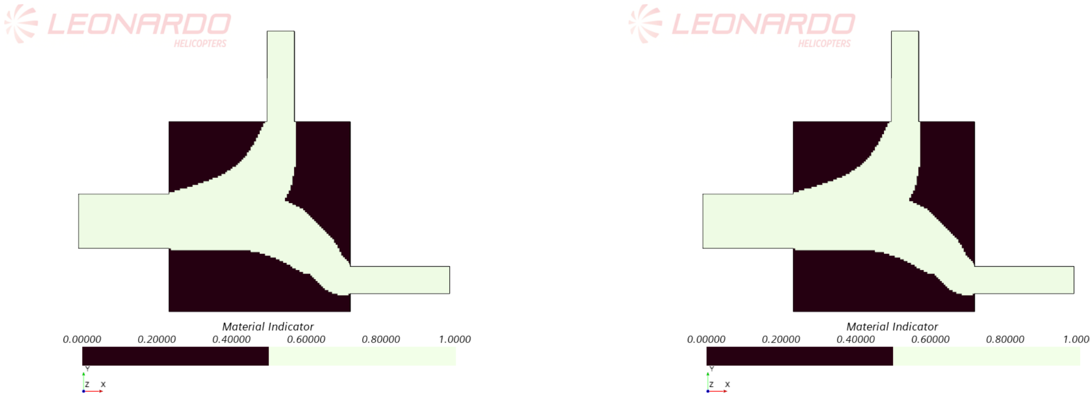
Figure B.2: Pressure Drop behaviour - Penalty Analysis



(a) Penalty = 100

(b) Penalty = 5000

Figure B.3: SVR behaviour - Penalty Analysis

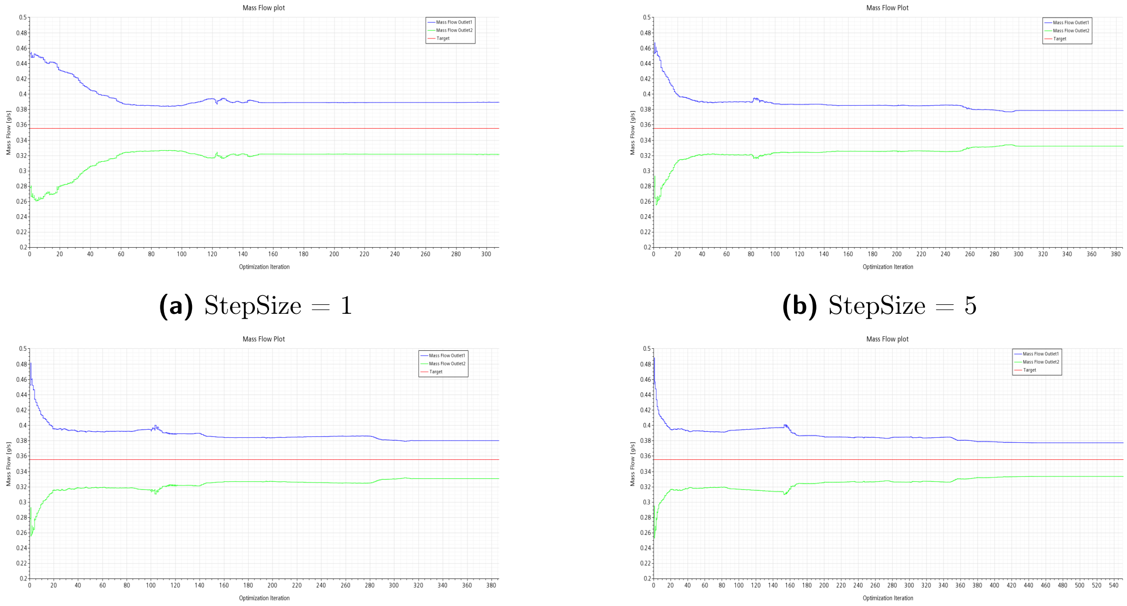


(a) Penalty = 100

(b) Penalty = 5000

Figure B.4: final topology - Penalty Analysis

B.2 StepSize Analysis



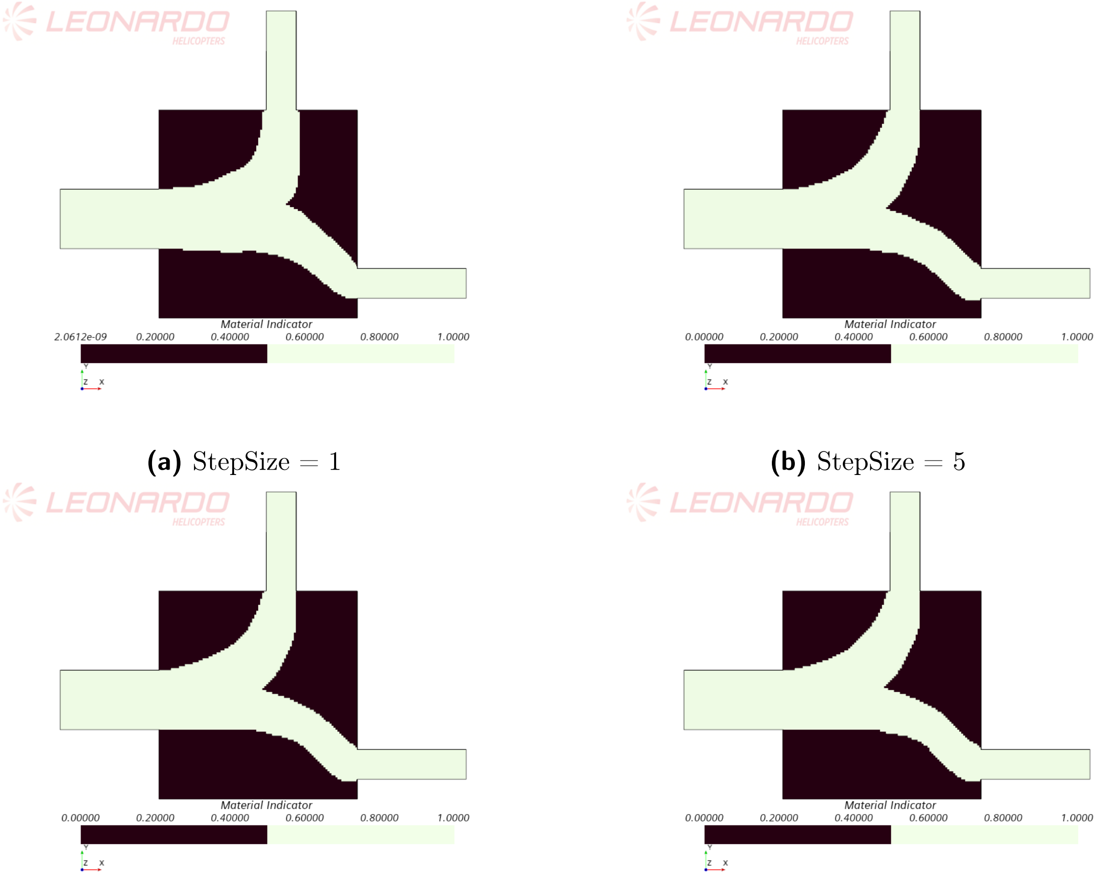
(a) StepSize = 1

(b) StepSize = 5

(c) StepSize = 10

(d) StepSize = 20

Figure B.5: Mass Flow comparison - StepSize Analysis - Penalty = 1000



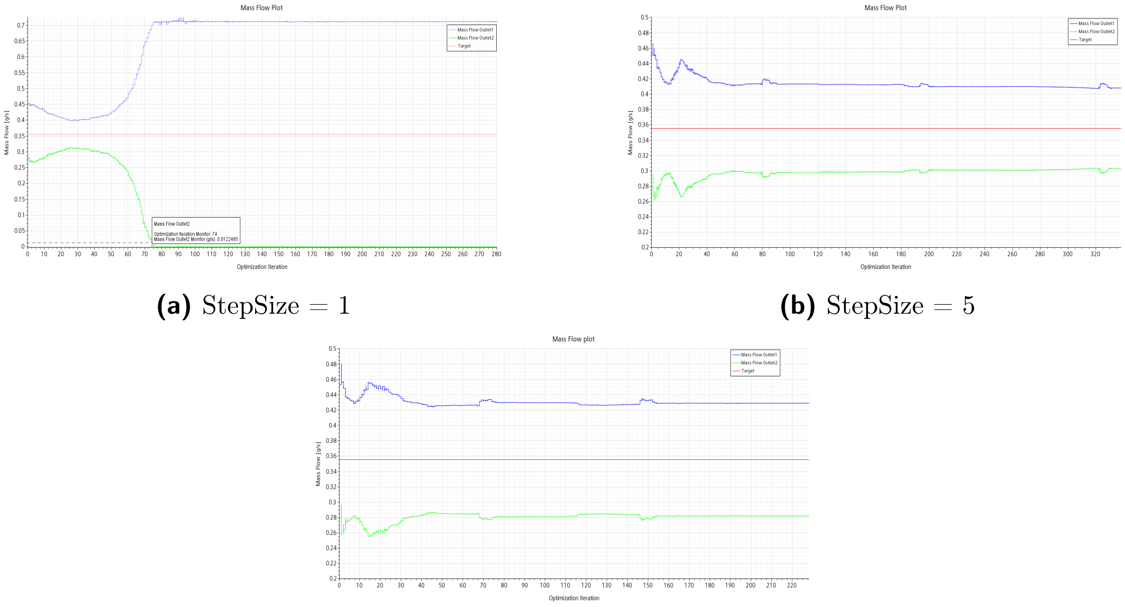
(a) StepSize = 1

(b) StepSize = 5

(c) StepSize = 10

(d) StepSize = 20

Figure B.6: Final Topology - StepSize Analysis - Penalty = 1000

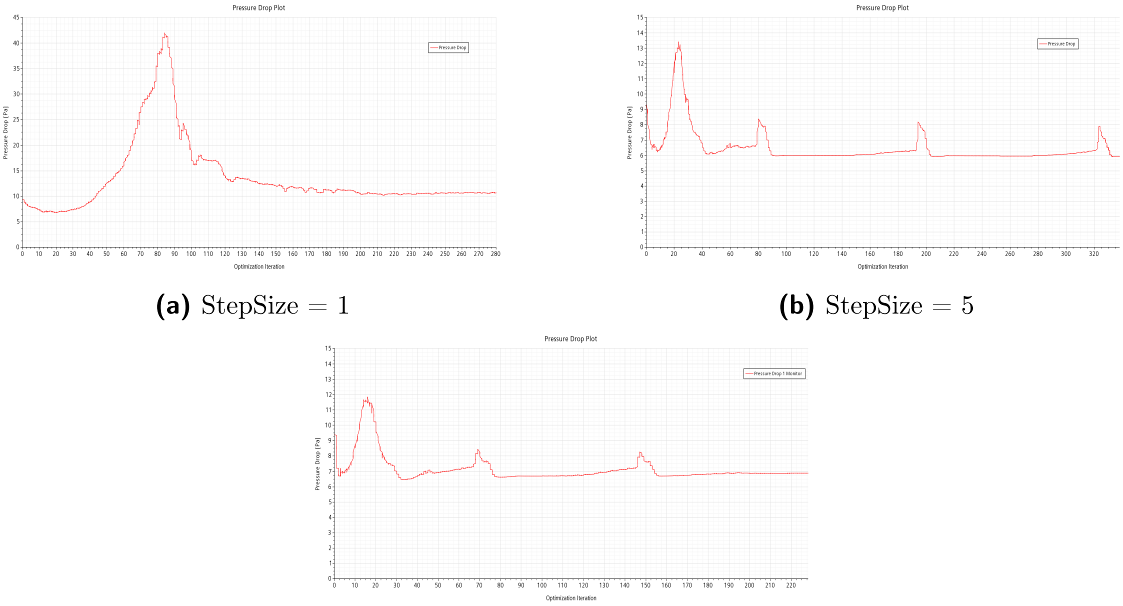


(a) StepSize = 1

(b) StepSize = 5

(c) StepSize = 10

Figure B.7: Mass Flow behaviour comparison - Step Size Analysis - Penalty = 10000



(a) StepSize $= 1$

(b) StepSize $= 5$

(c) StepSize $= 10$

Figure B.8: Pressure Drop behaviour comparison - Step Size Analysis - Penalty $= 10000$

Bibliography

[1] M. J. D. Powell A. C. Faul G. Goodsell. “A Krylov subspace algorithm for multiquadric interpolation in many dimensions”. In: (2005). DOI: 10.1093/imanum/drh021.

[2] Wang et al. “A level set method for structural topology optimization”. In: (2003). DOI: 10.1016/S0045-7825(02)00559-5.

[3] Krishnan Suresh Amir M. Mirzendehdel. *A Hands-On Introduction to Topology Optimization*. 2017.

[4] Philippe Angot. “A penalization method to take into account obstacles in incompressible viscous flows”. In: *Numerische Mathematik* (1999). DOI: 10.1007/s002110050401.

[5] M.P. Bendsøe and N Kikuchi. “Generating optimal topologies in structural design using a homogenization method”. In: (1988). DOI: https://doi.org/10.1016/0045-7825(88)90086-2.

[6] Marco Evangelos Biancolini. “RBF Mesh Morphing”. In: *Fast Radial Basis Functions for Engineering Applications*. Springer International Publishing, 2017. DOI: 10.1007/978-3-319-75011-8_6.

[7] Siemens Support Center. URL: %5Curl%7Bhttps://support.sw.siemens.com/en-US/product/226870983/knowledge-base/KB000038001_EN_US?pid=sc%5C%3Apc-typeahead&index=content-external&audience=external%7D. (Accessed: 7/01/2021).

[8] Na T.Y. Cunningham R.G. Hansen A.G. “Jet pump cavitation”. In: (1970). DOI: 10.1115/1.3425040.

[9] C. Dapongy. “An Introduction to shape and topology optimization: hands-on session”. In: (2018). DOI: http://www-ljk.imag.fr/membres/Charles.Dapogny/cours/tpoptim.pdf.

[10] Subhayan De et al. “Topology Optimization under Uncertainty using a Stochastic Gradient-based Approach”. In: *Structural and Multidisciplinary Optimization* (2020). DOI: 10.1007/s00158-020-02599-z.

[11] Ercan M. Dede. “Multiphysics Topology Optimization of Heat Transfer and Fluid Flow Systems”. In: (2009). URL: https://www.comsol.com/paper/download/44388/Dede.pdf.

[12] Jimmy Ba Diederik P. Kingma. “Adam: A Method for Stochastic Optimization”. In: (2014).

[13] Ramana V Grandhi. “A survey of structural and multidisciplinary continuum topology optimization: post 2000”. In: (2014). DOI: 10.1007/s00158-013-0956-z.

[14] Clyde Gumbert, Gene Hou, and Perry Newman. "Simultaneous Aerodynamic and Structural Design Optimization (SASDO) for a 3-D Wing". In: (2001). DOI: 10.2514/6.2001-2527.

[15] Raphael T. Haftka and Ramana V. Grandhi. "Structural shape optimization—A survey". In: *Computer Methods in Applied Mechanics and Engineering* (1986). DOI: https://doi.org/10.1016/0045-7825(86)90072-1.

[16] Rolland L. Hardy. "Multiquadric equations of topography and other irregular surfaces". In: *Journal of Geophysical Research (1896-1977)* (1971). DOI: https://doi.org/10.1029/JB076i008p01905.

[17] Airbus Helicopter. URL: https://www.airbus.com/. (Accessed: 10/12/2020).

[18] Leonardo Helicopters. URL: https://www.leonardocompany.com/it/innovation/future-programmes/next-generation-civil-tiltrotor-ngctr-. (Accessed: 9/12/2020).

[19] Wei-hong Zhang Ji-hong Zhu Liang Xia. "Topology Optimization in Aircraft and Aerospace Structures Design". In: (2015). DOI: https://link.springer.com/article/10.1007/s11831-015-9151-2.

[20] Gene Hou and Arunkumar Satyanarayana. "Analytical sensitivity analysis of a static aeroelastic wing". In: *8th Symposium on Multidisciplinary Analysis and Optimization* (2000). DOI: 10.2514/6.2000-4824.

[21] Chang-Wei Huang and K. Chou. "Volume Adjustable Topology Optimization with Multiple Displacement Constraints". In: *Journal of Mechanics* (2017). DOI: 10.1017/jmech.2017.68.

[22] et al. Igor J. Karassik Joseph P. Messina. *Pump Handbook, fourth edition.* The McGraw-Hill Companies, 2008.

[23] Andreas Wiegmann J.A.Sethian. "Structural Boundary Design via Level Set and Immersed Interface Methods". In: (2000). DOI: https://www.sciencedirect.com/science/article/pii/S0021999100965811.

[24] Bert Pohl Jocelyne Erhel Kevin Burrage. "Restarted GMRES preconditioned by deflation". In: (1995).

[25] Casper Schousboe Andreasen Joe Alexandersen. "A Review of Topology Optimisation for Fluid-Based Problems". In: (2020). DOI: 10.3390/fluids5010029.

[26] Brown K.E. *The technology of artificial lift methods.* Petroleum Publishing Co, 1993.

[27] Aykut Kentli. "Topology Optimization - Applications on Engineering Structures". In: (2019). DOI: https://www.researchgate.net/publication/338438926_Topology_Optimization_Applications_on_Engineering_Structures.

[28] Jibum Kim, Brian J. Miller, and Suzanne M. Shontz. "A hybrid mesh deformation algorithm using anisotropic PDEs and multiobjective mesh optimization". In: *Computers Mathematics with Applications* (2015). DOI: https://doi.org/10.1016/j.camwa.2015.08.008.

[29] Sebastian Kreissl, Georg Pingen, and Kurt Maute. "An explicit level set approach for generalized shape optimization of fluids with the lattice Boltzmann method". In: *International Journal for Numerical Methods in Fluids* (2011). DOI: `https://doi.org/10.1002/fld.2193`.

[30] Sebastian Kreissl, Georg Pingen, and Kurt Maute. "Topology optimization for unsteady flow". In: *International Journal for Numerical Methods in Engineering* (2011). DOI: `https://doi.org/10.1002/nme.3151`.

[31] Weichen Li and Xiaojia Shelly Zhang. *Momentum-based Accelerated Mirror Descent Stochastic Approximation for Robust Topology Optimization under Stochastic Loads.* 2020.

[32] K. Maute and M. Allen. "Conceptual design of aeroelastic structures by topology optimization". In: (2004). DOI: `10.1007/s00158-003-0362-z`.

[33] Sebastian Kreissl · Kurt Maute. "Levelset based fluid topology optimization using the extended finite element method". In: (2012). DOI: `10.1007/s00158-012-0782-8`.

[34] L. Dean Oliver and George Christakos. "Boundary condition sensitivity analysis of the stochastic flow equation". In: *Advances in Water Resources* (1996). DOI: `https://doi.org/10.1016/0309-1708(95)00027-5`.

[35] Anders Klarbring Peter W. Christensen. *An Introduction to Structural optimization.* Springer, 2009.

[36] Qihua Ma et al. Po Wu. "Topology Optimization Design of Automotive Engine Bracketn". In: (2016). DOI: `http://www.scirp.org/journal/epehttp://dx.doi.org/10.4236/epe.2016.84021`.

[37] Wikipedia - Clean Sky Programme. URL: `https://en.wikipedia.org/wiki/Clean_Sky`. (Accessed: 9/12/2020).

[38] et al. Rogelio de Jesús Portillo-Vélez J. A. Vásquez-Santacruz. "Efficiency Maximization of a Jet Pump for an Hydraulic Artificial Lift System". In: (2019). DOI: `10.23967/j.rimni.2018.11.002`.

[39] Mohamed Selim and Roy Koomullil. "Mesh Deformation Approaches – A Survey". In: *Journal of Physical Mathematics* (2016). DOI: `10.4172/2090-0902.1000181`.

[40] Lin Shi et al. "A fast multigrid algorithm for mesh deformation". In: *ACM Transactions on Graphics* (2006). DOI: `10.1145/1179352.1142001`.

[41] Siemens. *Siemens STAR CCM+.* URL: `https://www.plm.automation.siemens.com/global/it/products/simcenter/STAR-CCM.html`.

[42] *Stochastic Gradient Descent.* `https://en.wikipedia.org/wiki/Stochastic_gradient_descent`. Accessed: 2020-10-10.

[43] J. Petersson T. Borrvall. "Topology optimization of fluids in Stokes flow". In: (2003). DOI: `10.1002/fld.426.`.

[44] Clean Sky Joint Undertaking. URL: `https://www.cleansky.eu/`. (Accessed: 9/12/2020).

[45] Holger Wendland. "Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree". In: *Advances in Computational Mathematics* (1995). DOI: `10.1007/BF02123482`.

[46] WSelim and J Phys Math Koomullil. “Mesh Deformation Approaches – A Survey”. In: *Journal of Physical Mathematics* (2016). DOI: `110.4172/2090-0902.1000181`.

[47] et al. Xue-Guan Song Joon-Hong Park. “Performance comparison and erosion prediction of jet pumps by using a numerical method”. In: (2011). URL: `%5Curl%7Bhttps://doi.org/10.1016/j.mcm.2011.06.040%7D`.

[48] et al. Zhen Luo Nong Zhang. “A meshfree level-set method for topological shape optimization of compliantmultiphysics actuators”. In: (2012). DOI: `\url{https://doi.org/10.1016/j.cma.2012.02.011}`.