Carlos Massaiti Okubo Júnior

Topology optimization of compressible flows using a discrete adjoint approach

São Paulo
2022

Carlos Massaiti Okubo Júnior

Topology optimization of compressible flows using a discrete adjoint approach

Revised Version

Ph. D. Thesis submitted to the Escola Politécnica, University of São Paulo, in partial fulfilment of the requirements for the degree of Doctor of Science.

Concentration area:
Control and Mechanical Automation Engineering (3152)

Advisor:
Prof. Dr. Emílio Carlos Nelli Silva

São Paulo
2022

Este exemplar foi revisado e corrigido em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, _____ de _____ de _____

Assinatura do autor: _____

Assinatura do orientador: _____

Catalogação-na-publicação

Okubo Júnior, Carlos Massaiti
Topology optimization of compressible flows using a discrete adjoint approach / C. M. Okubo Júnior -- versão corr. -- São Paulo, 2022.
132 p.

Tese (Doutorado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.

1.Métodos topológicos (otimização) 2.Método adjunto discreto 3.Escoamento compressível 4.Método dos volumes finitos 5.Escoamento rotativo I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos II.t.

Okubo Júnior, Carlos Massaiti. **Topology optimization of compressible flows using a discrete adjoint approach**. 2022. 132 p. Ph. D. Thesis (Doctor of Science) - Escola Politécnica, University of São Paulo, São Paulo, 2022.

Approved in:

Examination Committee

Prof. Dr. _____

Institution: _____

Judgement: _____

Prof. Dr. _____

Institution: _____

Judgement: _____

Prof. Dr. _____

Institution: _____

Judgement: _____

Prof. Dr. _____

Institution: _____

Judgement: _____

Prof. Dr. _____

Institution: _____

Judgement: _____

ACKNOWLEDGEMENTS

I am grateful to my supervisor, Professor Dr. Emílio Carlos Nelli Silva, for the guidance, attention, discussions, and suggestions during the execution of this thesis. His support was fundamental for the development of this work.

I want to thank my parents, Carlos Massaiti Okubo and Vera Lúcia Luvizutto Okubo, and my sisters, Jéssica Okubo and Júlia Okubo for all the love, patience and moments together. A strong and united family is the basis for everything.

A special mention goes to my laboratory friends, Hélio Emmendoerfer Junior, Luís Fernando Nogueira de Sá, César Kiyono, André Ferreira, Francisco Oliveira, Bruno Caldas, Ruben Sallas, Paulo Yamabe, Mohammad Mehdi Ghorbani and João Baptista Dias Moreira. All the discussions (academic or not), companionship, and supportive moments during the coffee were priceless.

I also would like to thank the RCGI – Research Centre for Greenhouse Gas Innovation, hosted by the University of São Paulo (USP) and sponsored by FAPESP – São Paulo Research Foundation (2014/50279-4 and 2020/15230-5) and Shell Brasil for the financial support.

RESUMO

Neste trabalho, o Método de Otimização Topológica é empregado para gerar projetos capazes de trabalhar com escoamentos compressíveis e rotativos. As equações de Navier Stokes e da energia são resolvidas para casos a regime permanente. O modelo de gás perfeito é utilizado. A penalização de Brinkman é aplicada para representar sólidos dentro do domínio de projeto. Os modelos são representados em um sistema de coordenadas rotativo e, quando a turbulência é considerada, a média de Favre é utilizada em conjunto com o modelo de turbulência de Wray Agarwal de 2018. O objetivo principal do trabalho é otimizar projetos para trabalhar com escoamentos compressíveis e rotativos, no entanto, escoamentos incompressíveis e sem rotação também foram abordados. As funções objetivo consideradas para os escoamentos incompressíveis são a dissipação de energia e a eficiência de bomba e, para escoamentos compressíveis, a variação de entropia e a eficiência isentrópica. O cálculo de sensibilidades para o problema de otimização é feito através do método adjunto nas suas abordagens contínua e discreta. A abordagem discreta desenvolvida é uma nova metodologia proposta e que se apoia em um esquema de diferenças finitas. A implementação numérica é feita com a biblioteca de volumes finitos *OpenFOAM*, a biblioteca de *C++ Eigen* e a biblioteca científica *PETSc*. Exemplos numéricos são apresentados considerando escoamentos incompressíveis laminares com e sem rotação, escoamentos compressíveis laminares com e sem rotação e escoamentos compressíveis turbulentos com e sem rotação. Além disso, uma avaliação do modelo de turbulência em um contexto de otimização é apresentada. Os exemplos numéricos mostram que o cálculo de sensibilidades está implementado corretamente e que a metodologia desenvolvida é capaz de gerar projetos para trabalhar com escoamentos compressíveis e rotativos.

Palavras-chave: otimização topológica, método adjunto discreto, escoamento compressível, método dos volumes finitos, escoamento rotativo

ABSTRACT

In this work the Topology Optimization Method is employed to generate designs with rotating compressible flows. The Navier Stokes and energy equations are solved for steady state cases. The perfect gas model is used. The Brinkman penalization is applied to represent the solid regions inside the domain. The physical model is represented in a rotating reference frame and, to account for turbulent flows, the Favre average is used with the Wray Agarwal turbulence model from 2018. The main objective of the work is to optimize designs with compressible rotating flows, however incompressible and non-rotating cases have also been accounted. The objective functions considered for incompressible flows are the energy dissipation and the pump efficiency and, for compressible flow problems, the entropy variation and the impeller isentropic efficiency. The calculation of the sensitivities for the optimization problem is executed with the adjoint method in the continuous and the discrete approaches. The discrete approach developed is a novel methodology and is based on a finite differences scheme. The implementation is made with the use of the finite volume library *OpenFOAM*, the *C++* library *Eigen* and the scientific library *PETSc*. Numerical examples are presented considering incompressible laminar flows with and without rotation, compressible laminar flows with and without rotation and compressible turbulent flows with and without rotation. Also, an assessment of the behavior of the turbulence model in an optimization context is performed. The numerical examples show that the sensitivity calculation is correctly implemented and the methodology developed is capable of generating designs to work with compressible rotating flows.

Key words: topology optimization, discrete adjoint method, compressible flow, finite volume method, rotating flow

LIST OF FIGURES

LIST OF TABLES

NOTATION

$\boldsymbol{a_{cent}}$	Centripetal acceleration
$\boldsymbol{a_{cor}}$	Coriolis acceleration
$\boldsymbol{a_{rot}}$	Sum of centripetal and Coriolis acceleration
AMR_{inf}	Inferior limit for adaptive mesh refinement
AMR_{sup}	Superior limit for adaptive mesh refinement
c	Constraint for the optimization problem
c_p	Specific heat
C_{scale}	Scaling constant
c_{sound}	Sound speed
c_{vol}	Constant for augmented lagrangian
h	Enthalpy
h_{in}	Enthalpy at inlet
h_{out}	Enthalpy at outlet
$h_{out,s}$	Enthalpy at outlet for an ideal compression
$\tilde{h}$	Mass averaged enthalpy
h''	Fluctuating enthalpy component
J	Objective function
J_Ω	Objective function calculated by integration in the domain
J_Γ	Objective function calculated by boundary integration
$J_{rot,comp}$	Objective function for rotating compressible problem
$J_{rot,incomp}$	Objective function for rotating incompressible problem
$J_{stat,comp}$	Objective function for non-rotating compressible problem
$J_{stat,incomp}$	Objective function for non-rotating incompressible problem
k	Gas thermal conductivity
L	Lagrangian
$\mathcal{L}$	Augmented lagrangian
Ma	Mach number
$\hat{\boldsymbol{n}}$	Normal vector
p	Pressure
$\overline{p}$	Averaged pressure
p'	Fluctuating pressure component
P_f	Power added to the flow
p_{out}	Pressure at outlet
Pr	Prandtl number
Pr_t	Turbulent Prandtl number

continue...

p_{ref}	Pressure at a reference state
q	Interpolation function parameter
R	Gas constant
$\boldsymbol{r}$	Position vector
$\boldsymbol{R}$	Residual for state equations
R_c	Residual for continuity equation
$\boldsymbol{R_u}$	Residual for momentum equation
t_η	Kolmogorov time scale
T	Temperature
T_{avg}	Averaging time interval
T_{ref}	Temperature at a reference state
$\boldsymbol{u}$	Velocity field
$\overline{\boldsymbol{u}}$	Averaged velocity field
$\boldsymbol{u'}$	Fluctuating component of the velocity field
$\boldsymbol{u''}$	Fluctuating component of the velocity field for mass averaging
$\tilde{\boldsymbol{u}}$	Mass averaged velocity field
$\boldsymbol{u_{in}}$	Velocity at inlet
$\boldsymbol{v}$	Absolute velocity
V_{frac}	Volume fraction
$\boldsymbol{w}$	Rotation vector
α	Penalization term
$\overline{\alpha}$	Maximum allowable penalization
$\underline{\alpha}$	Minimum allowable penalization
γ	Specific heat ratio
Γ	Boundary of domain
δ_{adj}	Perturbation for adjoint system assembly
δ_{FD}	Perturbation for finite difference verification
δ_{SD}	Size for steepest descent
ϵ	Average rate of dissipation of turbulent kinetic energy
ϵ_J	Tolerance for objective function
ϵ_{vol}	Tolerance for volume constraint
ζ	Arbitrary flow variable
$\overline{\zeta}$	Averaged arbitrary flow variable
ζ'	Fluctuating component of arbitrary flow variable
η	Kolmogorov length scale
κ	Turbulent kinetic energy
λ	Bulk viscosity
μ	Dynamic viscosity

continue...

μ_t	Turbulent dynamic viscosity
ν	Kinematic viscosity
ν_t	Turbulent kinematic viscosity
ρ	Density
$\overline{\rho}$	Averaged density
ρ_{des}	Design variable
$\boldsymbol{\tau}$	Stress tensor
$\overline{\boldsymbol{\tau}}$	Averaged stress tensor
ϕ	Flow variable
ψ	Arbitrary flow variable
$\overline{\psi}$	Averaged arbitrary flow variable
ψ'	Fluctuating component of arbitrary flow variable
$\boldsymbol{\psi_{adj}}$	Lagrange multiplier vector for discrete adjoint
ψ_c	Lagrange multiplier for continuity equation
$\boldsymbol{\psi_u}$	Lagrange multiplier for momentum equation
ψ_{vol}	Constant for augmented lagrangian
Ω	Interior part of domain

CONTENTS

1 INTRODUCTION

In the latest years, weather patterns have been changing and becoming more extreme all over the world. Oceans are warmer now and the amounts of snow and ice have diminished, leading to higher sea levels. From 1901 to 2010, the global average sea level increased by 19 cm as oceans expanded due to warming and ice melting. These changes affect every country in the world and are intimately linked to the emissions of the so-called greenhouse gases, where carbon dioxide (CO_2) plays a major role. With the current concentrations and on-going emissions, it is estimated that, by the end of the century, the increase in the global temperature will exceed 1.5^oC when compared to the period between 1850 and 1900 (UN, 2022). Even though the temperature changes may seem mild, several intense consequences are expected to happen, such as dirtier air, more acidic oceans and higher sea levels, leading to higher death and wildlife extinction rates (NRDC, 2022).

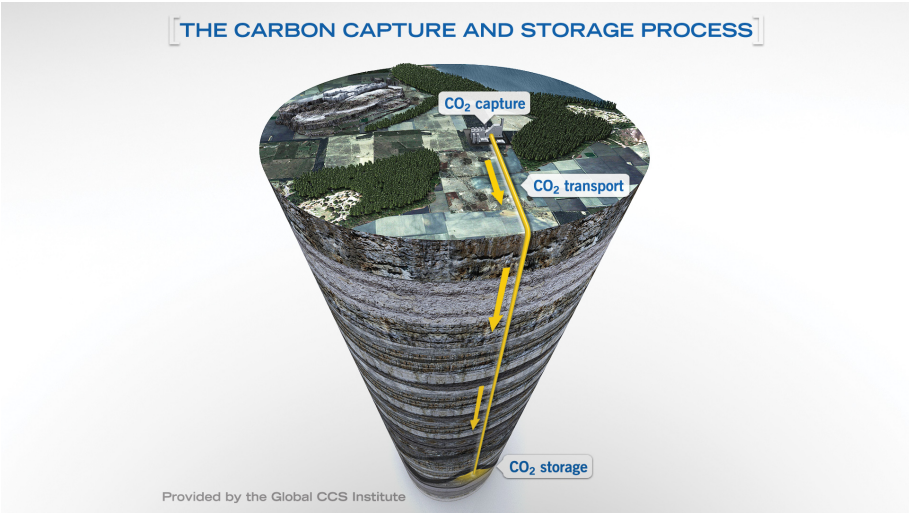
The harmful consequences and effects of climate changes are recognized worldwide and several nations have been working to contain and reverse the global emissions. Different strategies and initiatives have been developed. Some projects focus on Nature Based Solutions (NBS), where the goals involve the protection and restoration of native vegetation, the re-establishment of native species and land management systems as ways of avoiding new carbon dioxide emissions and improving the sequestration of the CO_2 currently released to the atmosphere (GIRARDIN et al., 2021). Other projects approach strategies to capture CO_2 emissions and convert them into valuable products by chemical processes. These are known as Carbon Capture and Utilization (CCU) (GHIAT; AL-ANSARI, 2021). Beyond that, there are projects focusing on techniques to capture the CO_2 in its main sources, transport and keep it in safe storing places. These are known as Carbon Capture and Storage (CCS). They are composed of three main steps (see Fig. 1.1) (INSTITUTE, 2022b):

a) The separation of CO_2 from other gases at industrial process facilities. The main CO_2 sources are: coal and natural gas based power plants, steel plants, cement plants and refineries;

b) Compression and transportation of CO_2 via pipelines, trucks, ships or any other methods to a safe site for storage;

c) Injection of the CO_2 into deep underground rock formations (usually at depths of 1 km or more).

Figure 1.1 – Overview of Carbon Capture and Storage (source: Institute (2022a)).



The adoption of CCS in industrial processes comes with uncertainties in energy consumption and costs implied by the extra infrastructure required. Therefore, the optimization of CCS processes aiming to reduce energy and cost impacts is an important field of study in the battle against climate changes (JACKSON; BRODAL, 2019). In order to be able to transport and inject the CO_2, the ability to compress carbon dioxide to pressures exceeding 20 MPa is fundamental. At this pressure, the fluid easily reaches a condition known as supercritical, temperature and pressure are above the critical point (31.05^{o}C and 7.39 MPa).

The compression process can be performed by centrifugal compressors and can dramatically increase operating costs due to the high density of the fluid in this condition. To allow economically viable large-scale CCS operations, advanced designs compatible with supercritical fluids are necessary (LETTIERI et al., 2014). Supercritical CO_2 ($scCO_2$) is not an usual condition, so there is space and need for the development and improvement of design procedures that aim to generate high efficiency projects. In this condition, the fluid is treated as compressible due to the high velocities and property variations as can be seen in the works of Rinaldi, Pecnik and Colonna (2015), Pham et al. (2016) and Ameli, Turunen-Saaresti and Backman (2018). This work finds itself within the CCS strategies and is part of a bigger project where the main goal is to develop design procedures to generate compressor rotors optimized for supercritical CO_2. The Topology Optimization Method (TOM) is the approach used here. During the development of the work, incompressible and compressible flows have been approached, with and without the presence of rotation. In the next chapters, these flow regimes are treated. Topology optimization for compressible flows is a very recent topic, making this one of the main contributions of this work. Even though compressors for supercritical CO_2 motivates and justifies the development of this work, $scCO_2$ is not considered here. The main objective is the establishment of a procedure capable of working with rotating compressible flows in topology optimization.

1.1 Compressible Rotating Flow

When high flow rates are present, high velocity components are inevitably involved and compressible flow becomes part of the situation (DIXON; HALL, 2010). A common definition to separate incompressible and compressible flow is based on flow velocity with respect to sound speed. The ratio between these two variables defines the flow Mach number Ma. For a given flow, if $Ma < 0.3$ flow can be considered incompressible and density is treated as a constant. If $Ma > 0.3$, density changes become important in flow modeling and must be taken in consideration. Compressible flow regimes can be divided in four different groups (ANDERSON, 2003):

- Subsonic: the local Mach number is everywhere less than unity. Hence, the flow velocity is less than the speed of sound for the whole domain considered. This flow is characterized by smooth behaviour and continuously varying properties;
- Transonic: if flow velocity at regions away from the geometry of interest is near sound speed, locally, Mach number can overcome the unity and reach a supersonic condition. In this case, the domain of interest has a mixed situation, with subsonic and supersonic regions. Shock waves can be present and severe property discontinuities can happen, requiring specific strategies to capture the variations appropriately;
- Supersonic: in this case, the whole flow field has velocities above the sound speed ($Ma > 1$ everywhere);
- Hypersonic: condition usually assumed for flows where $Ma > 5$. Temperatures can be high enough to dissociate or even ionize the gas. In these cases, chemical reactions between gas components become an important part of the problem.

Inside rotating devices, the flow can be very complex and with three dimensional characteristics. The addition of compressibility to the problem only increases the complexity of the system. In this scenario, Computational Fluid Dynamic (CFD) techniques play a major role since flow behavior can be accurately predicted and studied (WHITE, 1999). Several works using CFD to study the flow inside rotating devices can be found in literature. Next, some of the works focused on compressors are briefly described.

Mangani, Casartelli and Mauri (2012) present a study of a high pressure ratio centrifugal compressor for turbocharger applications. Compressible flow is considered and the main objective is to compare the use of different turbulence models, focusing on wall modeling. A modified version of the open-source code *OpenFOAM* is used. Efficiency and pressure ratio calculated by the numerical models are compared with experimental data. A good agreement for all the turbulence models is found. Thus, the authors conclude that the

simplest turbulence modeling among the options studied can be used when assessing pressure ratio and efficiency, saving computational efforts.

Heinrich and Schwarze (2013) use a segregated pressure-based solver for compressible flows to investigate the flow field inside the compressor of a turbocharger. The solver is based on the open source the library *OpenFOAM* and is validated with two test cases for centrifugal compressors. Performance maps are calculated for different rotational speeds and numerical results are compared experimental data, showing a good agreement. In some cases, the authors found that the performance is over predicted by up to 5% and attribute this differences to geometry and numerical model simplifications.

Kim et al. (2014) use the commercial code *Ansys CFX* to study an application involving the use of supercritical CO_2. The geometry used is derived from a water pump. Two operating points are studied: one next to the critical point and one far from it. An experimental setup is built to perform tests and compare with the numerical results. Comparisons of pressure ratio and efficiency are between the experimental measurements and numerical results are performed. For the operating condition far from the critical point, the errors found are up to 5% in efficiency. However, for the operating condition close to the critical region, errors of the order of 20% are observed. In the critical region, the large variation of the fluid properties and the possibility of phase change occurring inside the impeller are cited as the main source of errors, since they are very challenging to model.

In the work of Kim et al. (2016), unshrouded impellers are studied with the objective of assessing the performance effects of tip clearance variations due to mechanical deformations. The clearance between the impeller blade and the stationary casing surface is small when compared to the impeller dimensions, but the leakage that occurs through it is important and affects its performance. In this study, *Ansys CFX* is used for the flow analysis and *Ansys Structural* for the structural analysis. Two different blades are considered and the impeller deformed shape is calculated considering centrifugal, thermal and pressure loads. Considerable reductions in the clearance are observed (up to 47%), resulting in increases of the pressure ratio and efficiency of the order of 1% for both.

Pinto et al. (2017) presents a comprehensive review of CFD usage in turbomachinery. Studies for compressors, pumps and turbines are listed, analysed and several issues related to CFD applications in turbomachinery are identified. The main conclusions presented affirm that, nowadays, turbomachinery design is unimaginable without CFD. The high amount of details one can obtain in a short time justifies its use and the applications are expected to increase and become capable of approaching more complex phenomena. However, there is still space for development in the numerical modelings to reduce error, improve the geometry and flow simplifications and, also, improve the parallelization strategies.

Ameli, Turunen-Saaresti and Backman (2018) study the flow inside a centrifugal compressor used in a Brayton cycle. The working fluid is supercritical CO_2 and the focus of the work is to achieve an accurate modeling when working near the critical point. The fluid properties are inserted in *Ansys CFX* with the use of real gas properties (RGP) tables. When the fluid is compressed near the critical point, the sharp variations in the fluid properties at the vicinity of the critical point cause instabilities and difficulties to achieve converged results. By increasing the RGP tables resolution, more accurate fluid properties can be computed, however computational costs and simulation instabilities increase rapidly. Low-resolution tables alleviate the computational difficulties, but deteriorate considerably the accuracy of the solution. In this work, the authors propose a strategy of gradually increasing the resolution of the RGP table to achieve accuracy without experience divergence right at the beginning of the simulations. The results are encouraging and very low differences in isentropic efficiency and pressure ratio (when compared to experimental data) are observed. In a similar study, Ameli et al. (2018) used the same strategy of increasing the resolution of RGP tables, to study the effect of real gas properties in different operating conditions to identify trends in efficiency and torque and, also, calculate the compressor map.

The work of Galindo et al. (2019) focuses on the impact of placing different geometries at the compressor inlet for automotive applications. The commercial code *STAR-CCM+* is used and different parameters such as stability, efficiency and noise emission are studied for the geometries built. The results are compared with experimental tests and show that the performance is sensible to the geometry used at the inlet, indicating the possibility of benefits by choosing the correct inlet configuration.

Dewar et al. (2019) present a work comparing a CFD simulation with results from experiments for different operating conditions. Static pressure measurements at the diffuser are used to validate the simulations. The simulations are also used to investigate the effect of the position of the impeller-diffuser interface, a characteristic of the frozen rotor modeling approach. For studied cases, the optimal position found is approximately 2% of the blade radius, resulting in the best agreement between the experimental and numerical results.

In the work of Romei, Gaetani and Persico (2022), a centrifugal compressor for large-scale applications working with supercritical CO_2 is studied. The operating point is close to the critical point and the main objective of the work is to analyse the influence of inlet guide vanes (IGV) with different angles. Three inlet vane configurations are studied. The pre-rotation induced by the IGVs does not present a considerable effect on efficiency, but affects the extent of the two-phase region at the impeller inlet, having a negative impact on the compressor flexibility since it causes an early choking.

As can be seen by the works described, the possibilities of study are many. Each work

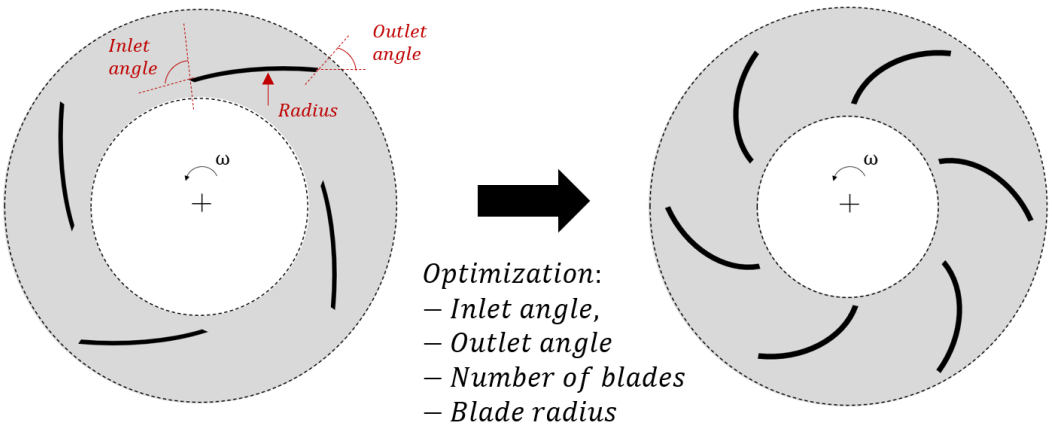
focused on a different aspect of the flow inside the compressor. Two common points among all the studies are: the use of the Finite Volume Method (FVM) and the modeling of the impellers by using only sectors (and not the full impellers). In this work, the FVM is also used to model sectors of the rotating flows.

The high level of detail obtained by CFD simulations allows engineers to carefully study changes in the designs without the need of conducting experiments for every design iteration. However, even with this resource, the design of rotating components can be a complicated and time consuming task given the amount of parameters that need to be defined. Rotors radius, widths, heights, number of blades, blades curvature, blades thicknesses are only part of the variables of the problem. To help accelerate the design process and reach solutions specifically suited for given applications, optimization techniques arise as a very interesting option.

1.2 Optimization

A common way of classifying optimization techniques is by dividing them in categories with respect to the definition of the design variables. Three main groups can be identified: parametric, shape and topology optimization. In parametric optimization (see scheme in Fig. 1.2), only a few parameters of the problem are used as design variables. Usually geometric features, process parameters or material properties are considered. In the case of rotating components, they can be number of blades, inlet angle, outlet angle, rotor width, rotation, among others. The work presented by Gölcü, Pancar and Sekmen (2006) exemplifies this category. Through experiments, the authors compare different deep well centrifugal pumps aiming to find the lowest energy consumption. The parameters changed are the number of blades and the presence or absence of blade splitters. A baseline case is defined and energy savings and improvements in efficiency are achieved by the optimization.

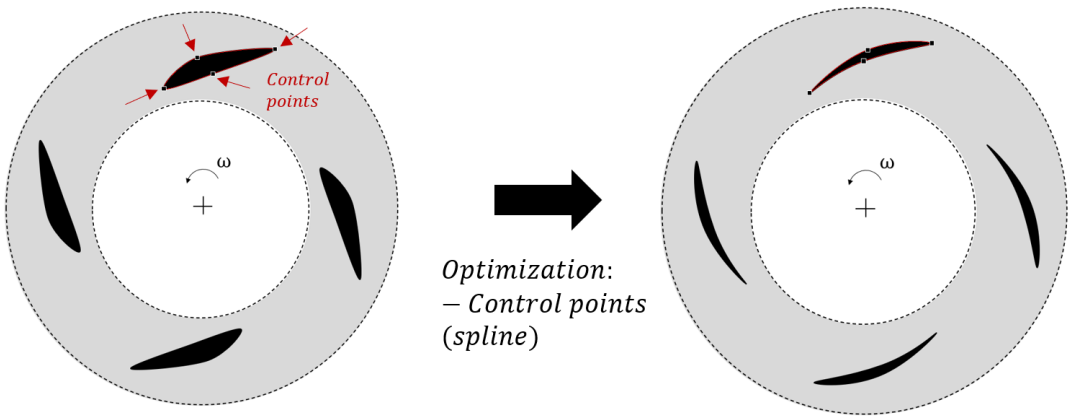
Figure 1.2 – Parametric optimization.



In compressor design, parametric optimization is seen in the work by Shaaban (2015), where the author optimizes the performance of a compressor by redesigning the vaneless diffuser. Two geometries are investigated and optimized. The objective is to minimize the loss coefficient and maximize the pressure coefficient. Simulations are performed by solving the flow in a 2D axisymmetric model. A genetic optimization algorithm is used and geometric parameters are chosen as design variables. The optimized geometry achieves a reduction in the loss coefficient by up to 10% and an increase in the pressure coefficient by up to 3.8%. The design is verified by 3D simulations with an impeller before the diffuser. In the 3D model, the optimized diffuser geometry reduces the loss coefficient by up to 4.7% and increases the pressure coefficient by up to 6.6%. Another example can be seen in Nejadali (2021) with the optimization of a regenerative flow compressor. In this case, the inlet and outlet blade angles are the design variables. A polynomial based response surface method is used. The isothermal efficiency and pressure coefficient are defined as objective functions. For the optimized geometry, the efficiency increase obtained is up to 3.2% and the pressure coefficient increase is up to 8%.

Shape optimization consists in changing the initial geometry by applying small displacements at the boundaries of the simulation domain. An example can be seen in scheme of Fig. 1.3. The geometry is parameterized using splines, B-splines or any other method that can be used to describe a contour and sensitivities of an objective function of interest are calculated with respect to small variations in these geometry descriptions.

Figure 1.3 – Shape optimization.



An example of design of rotating devices by shape optimization is presented by Economon, Palacios and Alonso (2013). The authors use the continuous approach of the adjoint method to calculate the sensitivities and optimize a rotating airfoil and a wind turbine blade, considering viscous compressible flow in both cases. The drag and the torque coefficients of baseline cases are improved by using this strategy.

In compressor design, Demeulenaere et al. (2015) present a case of a turbocharger. An initial design with a high performance is used, consisting of a challenging application

for the optimization. The strategy presented consists in a combination of a genetic algorithm, a neural network and a database. The basic idea of the methodology is to build an approximate model of the original analysis model. This model can be used inside an optimization loop replacing the original model, allowing fast evaluations, since the approximate model is less costly. To construct the approximate model, the authors use artificial neural networks (ANN). The geometry is parameterized by Bezier curves. The objectives are to improve the design performance, improve stall margin, keep the capacity (choke mass flow) and decrease the peak mechanical stresses by up to 20%. For the creation of the database, the flow problem is solved by the commercial package *NUMECA* (CADENCE, 2022) and the structural problem by the package *Oofelie* (OOFELIE, 2022). At the end, an optimized impeller with stress levels are below a fixed limit and with an increased capacity and stall performance is obtained.

In the work of Oka et al. (2017), a compressor impeller is optimized by using a genetic algorithm and a two dimensional inverse design method. The aerodynamic design is based on two steps: a meridional viscous flow calculation and a two dimensional inverse design. The calculation and the inverse design are performed repeatedly until the blade geometry and the flow field are converged. The meridional viscous flow calculation is performed on a two dimensional meridional grid with a blade force modeling in order to obtain the flow distribution around the impeller. By using the flow distribution and the predetermined blade loading distribution, the 3D impeller geometry is obtained by the inverse method. In the inverse blade design procedure, blade loading distribution is given as the design variable. The total pressure rise and the isentropic efficiency obtained from the meridional viscous flow analysis are considered objective functions. The optimized result shows performance improvements and suppression of the flow separations on the suction surfaces.

The work of Hehn et al. (2018) shows the optimization of a transonic centrifugal compressor, where the objectives are to increase the isentropic efficiency and reduce the acoustic signature by decreasing the amplitude of the shock pressure waves at the inlet of the compressor. The optimization is performed at three operating points to maintain choke mass flow and surge margin. Bezier curves and B-splines are used to describe the compressor geometry and a combination of artifical neural networks and genetic algorithms is used. In the optimized design, an increase of 1.4% in isentropic efficiency is observed together with a reduction of 23% in the shock strength. In a similar strategy, Ekradi and Madadi (2020) also use a combination of genetic algorithms with artificial neural networks to optimize a centrifugal compressor achieving an increase in the isentropic efficiency of 0.97% and increases in the total pressure ratio and mass flow rate are of 0.74% and 0.65%, respectively.

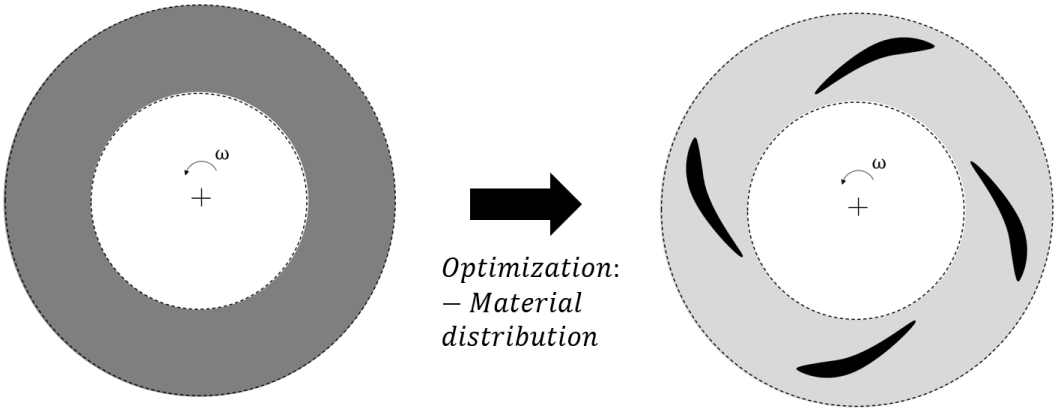
In the work presented by Xia et al. (2021), a centrifugal compressor working with supercritical CO_2 is designed by optimization. Initially, a 1D design is obtained by finding

optimized parameters with the use of Particle Swarm Optimization and genetic algorithms. Then, the optimized geometry is assessed in a 3D model and optimized with the use of artificial neural networks and genetic algorithms to achieve a high efficiency design.

McLaughlin et al. (2022) present the study of a vaned diffuser aiming to reduce the radial dimensions of the diffuser keeping the compressor performance at the design point. A metamodel using artificial neural networks is used in the optimization. The vanes are parameterized by using Bezier curves and a sequence of different objective functions based on the authors experience is defined. Genetic algorithms are applied to optimize a baseline geometry. An experimental setup is assembled to test the optimized geometries and the measured performance shows that the new designs are able to keep the original efficiency while having a reduction of 15% in the radial dimensions of the diffuser.

In Topology Optimization, the objective function of interest is minimized or maximized by distributing material inside a design domain. Governing equations of the problem are modified to allow the addition and removal of material at any part of the domain. Unlike the other techniques, in Topology Optimization, it is possible to open and close holes, resulting in a greater freedom of design with a broader range of possibilities. There is no need to have an initial design that makes sense for the problem (as in scheme of Fig. 1.4), the final geometry usually has no resemblance with this initial attempt and non-intuitive solutions can be generated. For rotating devices, the final geometry can have different blades, flow paths and intermediate bodies (blade splitters) can be created. In the work of Romero and Silva (2014), rotors for pumps and turbines operating with laminar incompressible flow are designed to minimize energy dissipation, vorticity and torque.

Figure 1.4 – Topology optimization.

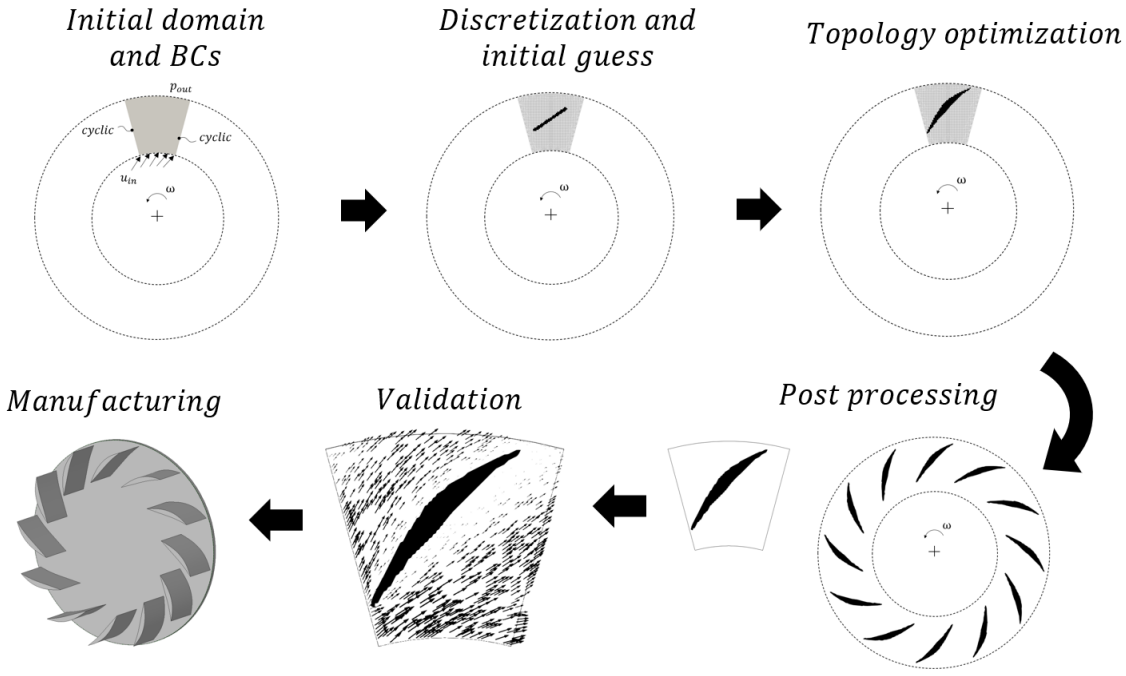


1.3 Topology Optimization Method

Topology Optimization is the most versatile strategy among the optimization approaches. It allows the placement and removal of material in any part of the domain. A complete topology optimization procedure consists of a combination of analysis, sensitivity calculations

and the use of optimization algorithms. The analysis step can be executed with the Finite Differences Method (FDM), the Finite Volume Method (FVM), the Finite Element Method (FEM) or any other numerical method that can be used to solve continuum mechanics problems. In this step, the physical problem is solved and the current state of the design can be analysed. Then, the sensitivities with respect to a desired objective function are calculated to guide the optimization algorithm with respect to where material must be placed and where it must be removed. The optimization algorithm updates the design variables and a new analysis step is performed. After doing this cycle of steps a couple times, an optimized design can be obtained. In the case of rotating flows, a scheme for the procedure can be seen in Figure 1.5. Initially, a domain where the design can exist is defined with respective boundary conditions. Then, the domain is discretized so the numerical method is applied. The problem is solved in the domain, the sensitivities of the objective function are calculated and the optimization algorithms are applied to update the design variables and generate an optimized design. After some stopping criteria are satisfied for the optimization algorithm, the resulting design can be post processed, validated through simulation and manufactured.

Figure 1.5 – Topology optimization for rotating flows.



Topology Optimization has its origins in structural analysis and the first use in fluid flow problems is presented at the work of Borrvall and Petersson (2003). The authors generate designs that minimize energy dissipation in several Stokes flow cases by using a gradient driven optimization scheme. Solid representation inside the domain is performed with the porous media modeling and an interpolation function is used to control how the inverse permeability changes with the design variables. Cases of internal and external flows are presented, showing how the flexibility of topology optimization can be applied to

fluid flow problems. In a different work, Moos, Klimetzek and Rossmann (2004) optimize air-guiding systems by using an heuristic scheme focused on recirculation areas. The inspiration of the algorithm comes from what happens in a river. In areas of high flow velocity the river becomes deeper since it removes sediments and, in areas of low velocities or intense recirculation, the river deposits sediments. The authors treat the material model as a "virtual sand" by depositing it in high recirculation areas with the objective of finding a design that reduces pressure drop. A traditional design is considered as a baseline case and compared with results from the optimization proposed. Reductions in the pressure drop up to 58% are observed. Following the work of Borrvall and Petersson (2003), Gersborg-Hansen, Sigmund and Haber (2005) minimize energy dissipation in fluid flow problems, this time using the complete Navier-Stokes equations for incompressible laminar flow. The sensitivities are calculated by the adjoint method and the problem is modeled by using the Finite Element Method. Numerical examples designing flow channels and flow mechanisms are presented.

Othmer (2008) presents the sensitivity calculation for flow topology optimization using the continuous approach of the adjoint method. The author develops the formulation considering internal flows and calculates the sensitivities to minimize energy dissipation and maximize flow uniformity at the outlet plan. Turbulent flow is considered by using the so-called frozen turbulence approach, where the variation of the Lagrangian does not consider the variation of eddy viscosity. In this work, the Finite Volume Method is used.

The first work to fully consider turbulence in fluid flow topology optimization is presented by Papoutsis-Kiachagias et al. (2011). Using the continuous approach of the adjoint method, the authors derive an adjoint model for the Spalart-Allmaras turbulence model and present numerical cases minimizing viscous losses. Heat transfer is also considered in the optimization problems and a constraint to allow the specification of volume flow at different outlets is implemented.

In the work of Papoutsis-Kiachagias and Giannakoglou (2016), the authors use an adjoint model for Spalart-Allmaras and also propose a method of applying the law of the wall to the interface created by topology optimization. Examples for several different applications are presented, considering 2D and 3D cases.

The only work found dealing with topology optimization of compressible flows is presented by Lapointe et al. (2017). The authors use *OpenFOAM* to simulate the flow and the discrete adjoint approach to calculate sensitivities with a modified version of the package developed by Towara and Naumann (2013). Supersonic backward facing step cases are optimized to remove recirculation areas and create desired angles at the flow outlet. Although interesting, the work lacks discussions about fundamental matters of topology optimization, such as material model, interpretation of final result and sensitivity verification.

Dilgen et al. (2018a) optimize several cases of confined turbulent flows aiming to minimize energy dissipation. Algorithmic differentiation is used to assemble the adjoint problem and calculate the sensitivities for the optimization problem. Cases considering 2D and 3D problems are presented. A similar framework is applied in the work of Dilgen et al. (2018b) where heat transfer is also considered and the objective function is the minimization of the average temperature in the domain. In order to avoid designs with high pressure loss, the authors apply a constraint to control the energy dissipation.

In all the works previously cited, the design variable is allowed to vary in a continuous range, meaning that the final result can have intermediate porosity values. A different approach conceived to avoid this possibility is seem in the works of Souza et al. (2021), Picelli et al. (2022) and Moscatelli et al. (2022). In all of them, the strategy called Topology Optimization of Binary Structures (TOBS) is used. The main advantage is that only 0 and 1 are allowed to the design variables, so well defined geometries are obtained. To do so, the authors rely on integer linear programming. In the work of Souza et al. (2021), classical examples (from the work of Borrvall and Petersson (2003)) are presented to show the methodology working. In the work of Picelli et al. (2022), beyond the use of the TOBS, the authors implement a strategy to extract the geometry at each optimization iteration and calculate the physical problem in a body fitted mesh. This way, some of the turbulence modeling resources, such as wall functions, can be used. The authors present 2D and 3D cases of minimization of energy dissipation. In the work of Moscatelli et al. (2022), a strategy to consider rotating and static parts of the domain is developed, allowing the model to consider the interaction between rotor and stator.

In the design of rotating devices, first use of topology optimization is presented by Romero and Silva (2014). Rotors for pumps and turbines are designed considering energy dissipation, vorticity and torque as objective functions. Flow is considered laminar and incompressible. This work is followed by Sá et al. (2017), where the concept of topological derivative is used to calculate the sensitivities and by Sá et al. (2018), where, beyond designing small pump rotors, the authors build a prototype using additive manufacturing and compare experimental results with numerical predictions. In Sá et al. (2021), turbulence is considered by using the Spalart-Allmaras model with rotation and curvature correction to design 2D rotors. Another interesting research line developed for rotor design is presented in the work of Alonso et al. (2018). The authors use a 2D swirl formulation to design axisymmetric devices with laminar flow laminar by minimizing the relative energy dissipation considering the viscous and porous effects. Other works extended this approach to Tesla pumps (ALONSO et al., 2019), Tesla turbines (ALONSO; SILVA, 2022), models with non-newtonian flow (ALONSO; SAENZ; SILVA, 2020), blood pump devices considering blood damage (ALONSO; SILVA, 2021) and designs with turbulence modeling (ALONSO et al., 2022).

1.4 Adjoint Problem Solution

In optimization using gradient driven approaches, the calculation of the sensitivities with respect to the design variables is one of the most important and challenging steps. In this work, the adjoint method is employed. It has been largely used in several different works. Only two solver calls are necessary for each optimization iteration, one for the primal system and one for the adjoint system (OTHMER, 2008; PAPOUTSIS-KIACHAGIAS; GIANNAKOGLOU, 2016). Two major approaches are used: the continuous and the discrete approach. The starting point on the continuous approach consists in the primal equations in their analytical form. Through the Lagrange multipliers method, the adjoint system is derived analytically and then implemented in any code to be discretized and solved. The discrete approach starts from the primal system already discretized and, by also using the Lagrange multipliers method, the adjoint system is defined. This time, no analytical form of the adjoint system is observed (GILES; PIERCE, 2000; NADARAJAH; JAMESON, 2000; OTHMER, 2008). Both approaches can be observed in the literature and there is no definite answer on which should be chosen. Advantages and disadvantages in both cases can be cited. When using the continuous approach, the physical significance of adjoint variables and the role of adjoint boundary conditions are clearer (GILES; PIERCE, 2000); the adjoint program requires less memory (GILES; PIERCE, 2000; NADARAJAH; JAMESON, 2000) and the adjoint equation terms can be discretized with different schemes (GILES; PIERCE, 2000). All this comes at the cost of the hand differentiation and implementation being challenging tasks, depending on the problem (KENWAY et al., 2019). When the discrete approach is used, the exact gradient of the discrete objective function is obtained (GILES; PIERCE, 2000), even for complex calculation routines, such as the ones involved in turbulence modeling (KENWAY et al., 2019); it can be a guide for the continuous approach (GILES; PIERCE, 2000; NADARAJAH; JAMESON, 2000); no hand differentiation is required (MARTA et al., 2007). The main disadvantage of the discrete approach is the high cost. Computing and storing the exact Jacobian matrices can be very demanding in terms of computation and memory (KENWAY et al., 2019). It is also worth to mention that, even though the approaches follow different paths, the results obtained with the continuous and the discrete approaches approximate to each other as the density of the discretization increases (NADARAJAH; JAMESON, 2000). In flow optimization in general (shape and topology), a large amount of interesting works can be found for both, the continuous and the discrete version of the adjoint method.

In the continuous approach, the work of Othmer (2008) is an important mark for fluid flow topology optimization. The author derived the adjoint problem for incompressible flows, focusing on internal flow problems. The objective functions considered are energy dissipation and flow uniformity at the outlet. This work is implemented in *OpenFOAM* and the code is available as a solver for all versions since its development. The frozen

turbulence assumption is used, which means that the turbulent viscosity is not considered as a state variable and the turbulence model is not considered during the development of the adjoint system. The work of Papoutsis-Kiachagias et al. (2011) is the first to fully consider turbulence when developing topology optimization for flow problems using the continuous approach. The Spalart-Allmaras turbulence model is used. Heat transfer is also considered in the numerical cases presented and a mass flow constraint is used to create designs where each outlet has a previously specified flow rate. A more practical application can be seen in the work of Villiers and Othmer (2012). The authors design the intake of combustion chambers by minimizing an objective function resulting of a combination of pressure drop and flow swirl in a desired region. The flow is modeled using a compressible approach due to the high velocities, but the adjoint system used is the one related to an incompressible modeling, constituting what can be seen as a "frozen density" approach, also in the continuous version of the adjoint method. A baseline case is defined and different geometries are generated, showing improvements in both the pressure drop and the flow swirl. The work of Papoutsis-Kiachagias and Giannakoglou (2016) is very extensive and explores the adjoint formulation for incompressible flows considering shape and topology optimizations. Cases for internal and external flows are presented, representing problems from automotive, aerospace and turbomachinery areas. Also, a discussion about the frozen turbulence assumption is performed, showing that it may be inappropriate depending on the case. Another interesting work using the continuous approach is presented by Karpouzas (2019). The flow is considered incompressible and the frozen turbulence assumption is used. In this work, the focus is the new way of treating the solid-fluid interface created by topology optimization. The Immersed Boundary Method is used and the mesh locally adapts to the interface, resulting in accurate flow calculations close to the surfaces and easier interpretation of the results.

When it comes to the discrete version of the adjoint method, different strategies can be used. It is possible to work directly with the matrices from the discretized system, as in the works of Borrvall and Petersson (2003) and Romero and Silva (2014). The first is the pioneer use of topology optimization in flow problems while the later is the first topology optimization work dedicated to rotating flows and the design of pumps and turbine rotors. It is also possible to use *automatic differentiation* (AD) (NOCEDAL; WRIGHT, 1999) techniques. AD is based in the fact that any function can be broken down into a composition of elementary arithmetic operations, to which the chain rule can be applied, thus it is possible to calculate the gradients of any function that can be implemented in a computer code (NOCEDAL; WRIGHT, 1999). The works of Towara and Naumann (2013) and Towara, Schanen and Naumann (2015) are examples where the authors differentiate the full CFD code. By overloading some very fundamental functions in *OpenFOAM* and using an AD library, it is possible to obtain gradients of any objective function. In their works, the authors present examples for topology optimization of incompressible flows.

This approach is robust and capable of dealing with a great level of complexity, but comes with a high memory consumption, demanding strategies to alleviate or compensate this problems when working with large cases. To circumvent this kind of problem, AD can be applied selectively to only some parts of the code, aiming to maintain the accuracy, but avoiding a high demand on the computers. In these cases, an adjoint system is assembled and must be solved by a linear solver. Examples of this kind of approach can be seen in the works of Dilgen et al. (2018a), Dilgen et al. (2018b), Alonso et al. (2018) and Sá et al. (2021). In the first and the second works, the authors use an in-house code to solve incompressible turbulent flow cases and assemble the adjoint system with the help of the AD library *Adept* (HOGAN, 2014). In the other two works, the authors use the library *FEniCS* to solve the physical problem and its adjoint module to apply AD to assemble the adjoint system and calculate the sensitivities. An interesting work that also uses *FEniCS* is presented by Alonso, Rodriguez and Silva (2021). A framework designed to extract the strongest capabilities of *OpenFOAM* and *FEniCS* is elaborated. An interface between them is created in such a way that the physical problem is solved in *OpenFOAM* and the sensitivities are calculated by the adjoint module of *FEniCS*. The numerical examples presented consider 2D and 3D cases and minimize the energy dissipation.

In the calculation of sensitivities for flow optimization problems, another important question that arises is regarding the consideration of the turbulence models in the sensitivity calculation. To simplify the derivation of the sensitivities, some authors use what is known as the frozen turbulence or Constant Eddy Viscosity (CEV) assumption, which means that the turbulence models are not considered when the sensitivity is being calculated. Comparisons of calculated gradients with finite differences show that, in some cases, this approximation can lead to wrong gradients (KIM; KIM; RHO, 2003; DWIGHT; BREZILLON, 2006; MARTA; SHANKARAN, 2013; ECONOMON; PALACIOS; ALONSO, 2015; DILGEN et al., 2018a). The validity of this assumption is problem-dependent (ECONOMON; PALACIOS; ALONSO, 2015) and, in several situations, the sensitivities are accurate enough to be used for design (KIM; KIM; RHO, 2003; DWIGHT; BREZILLON, 2006; MARTA; SHANKARAN, 2013; ECONOMON; PALACIOS; ALONSO, 2015). This can be confirmed by the fact that, in all works investigated (considering optimizations for shape and topology), the results obtained with frozen turbulence (or CEV) hypothesis and with complete sensitivity calculation are very similar if not indistinguishable. This hypothesis is used here when dealing with turbulent flows.

In this work, both approaches of the adjoint method are explored. The continuous approach is used for incompressible rotating flow cases. For the discrete approach, a new strategy is used. The matrices and vectors necessary to assemble the adjoint system and calculate the sensitivities are estimated by finite differences. This strategy is seen in the works of He et al. (2018), He et al. (2019), He et al. (2020) in shape optimization

cases and is developed here to topology optimization problems. The new strategy is robust, easier to apply to complex problems and its development is motivated by the complexity of a continuous adjoint system for compressible flows. Examples of applications to incompressible and compressible flow problems are presented.

1.5 Motivation

The need to have a systematic approach to rapidly create optimized designs for different conditions comes from the non-conventional situations arising from the different applications that involve rotating machinery (e.g.: compressors for supercritical CO_2). The flow inside rotating devices can exhibit very complex patterns due to the path the flow must travel inside the device. At high velocities, as in the case of compressors, compressibility effects are present, density changes become important and the flow modeling must be able to capture their influence, so a highly nonlinear set of equations must be solved. The complicated flow conditions and the number of geometric parameters that must be defined when designing rotating machinery make this process a non trivial and time consuming task.

In this scenario, the Topology Optimization Method arises as an alternative to integrate a design procedure that can systematically approach different situations and generate optimized designs. The use of this approach is motivated by its flexibility to create complex and non-intuitive geometries (which can be built by additive manufacturing techniques), by the significant reductions in the development time and also by the amount of information needed to initiate the design procedure (which is small compared to other procedures). Since the number of design variables is very high (equals to the number of discretization cells), the adjoint method is the choice to calculate the sensitivities. For incompressible flows, the optimization problem is solved with the continuous version of the adjoint method. However, for the complicated and nonlinear set of equations of compressible flows, the discrete version of the adjoint method is used. An approach using finite differences, inspired from the work of He et al. (2018), is developed. Thus, this work aims to apply the Topology Optimization Method to design devices operating with rotating flows by using the adjoint method.

Topology optimization for compressible flows is a field nearly unexplored and works in this area are of undeniable relevance and contribution. Also, its implementation with the adjoint method based on a finite differences strategy have not been presented in the literature, making this one of the main contributions of this work.

1.6 Objectives

The main objective of this work is to develop a systematic procedure to apply the Topology Optimization Method to problems of rotating compressible flows in the subsonic regime ($0.3 \leq Ma < 1$). The path to achieve the main objective can be decomposed in steps that are also considered objectives of this work:

- Find a penalization for the system of equations that can represent solid regions in compressible flow calculations;
- Implement the sensitivity calculation for flow problems using the continuous adjoint method;
- Implement the sensitivity calculation for flow problems using the discrete adjoint method based on a finite differences approach;
- Apply the Topology Optimization Method to incompressible laminar flows (with and without rotation) using the sensitivity calculations developed. In this case, energy dissipation and pump efficiency are considered as objective functions;
- Apply the Topology Optimization Method to compressible flows with and without rotation.

1.7 Scientific Contribution

The scientific contributions that resulted from this work are the following:

a) Derivation and implementation of the continuous adjoint model for topology optimization of incompressible rotating flows;

b) Development and implementation of a penalization strategy that can represent solids in compressible flow calculations;

c) Development and implementation of a topology optimization strategy using an adjoint system based on a finite differences approach. The strategy is not physics related, so it can be extended to other types of problems.

d) Application of the developed strategy to incompressible and compressible flows, with and without rotation;

e) Definition of objective functions related to compressible flow calculations;

f) Development and implementation of a topology optimization code capable of optimizing viscous subsonic compressible flows. The part of the code responsible for simulating flows comes from an existing open source code. All the development in this work is the part related to the application of topology optimization.

Four journal papers related to this work have been published:

a) Topology optimization applied to 3D rotor flow path design based on the continuous adjoint approach (OKUBO et al., 2021);

b) Topology optimization of subsonic compressible flows (SÁ; OKUBO; SILVA, 2021);

c) A discrete adjoint approach based on finite differences applied to topology optimization of flow problems (OKUBO et al., 2022).

d) Continuous boundary condition propagation model for topology optimization (SÁ et al., 2022)

1.8 Document Outline

In this work, the methodology developed to apply the Topology Optimization Method to incompressible and compressible flows with rotation is detailed. The document is organized as follows: in Chapter 2, the equations for the physical problem are presented with the modeling used for finite volume approaches. In Chapter 3, the optimization problem is defined and descriptions of the objective functions and constraint are presented. In Chapter 4, the adjoint problem is exhibited together with the sensitivity calculation and some examples for its verification. In Chapter 5, details about the numerical implementation are described. In Chapter 6, numerical examples are presented, showing optimization cases for incompressible and compressible flows and also tests for the penalization of equations. Finally, in Chapter 7, the conclusions for the developed work are reported and ideas for future work are suggested.

2 FLUID FLOW MODELING

In this section, the governing equations for rotating incompressible and compressible flows are introduced. A brief description of turbulence modeling is given together with details of the turbulence model used in this work. Then, a short explanation on fluid flow modeling with the Finite Volume Method is presented.

2.1 Governing Equations

In flows where compressibility can be neglected, the system of equations used is composed by the continuity and the momentum equations, constituting the incompressible version of the Navier-Stokes equations. For a steady state case, this system is given by (MOUKALLED; MANGANI; DARWISH, 2016b):

$$\nabla \cdot \boldsymbol{u} = 0 \tag{2.1}$$

$$\nabla \cdot \left(\boldsymbol{u}\boldsymbol{u}\right) - \nabla \cdot \left(\nu \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T\right)\right) + \frac{\nabla p}{\rho} = 0 \tag{2.2}$$

where $\boldsymbol{u}$ is the instantaneous velocity, p is the instantaneous pressure, ν is the kinematic viscosity, ρ is the specific mass.

When the domain being modelled is subject to a rotation $\boldsymbol{\omega}$, it is common to simulate the flow in the rotating reference frame. This requires the addition of the centripetal and the Coriolis acceleration components (KUNDU; COHEN; DOWLING, 2012):

$$\boldsymbol{a_{cent}} = \rho \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{r}) \tag{2.3}$$

$$\boldsymbol{a_{cor}} = 2\rho \left(\boldsymbol{\omega} \times \boldsymbol{u}\right) \tag{2.4}$$

An additional term composed by a scalar field α multiplied by the velocity vector is added to the momentum equations to represent solid regions inside the domain. This approach, known as Brinkman penalization, is necessary to perform the topology optimization and is explained in more details in Section 3.2. Thus, for incompressible rotating flows, the complete system of equations is given by:

$$\nabla \cdot \boldsymbol{u} = 0 \tag{2.5}$$

$$\nabla \cdot \left(\boldsymbol{u}\boldsymbol{u}\right) - \nabla \cdot \left(\nu \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T\right)\right) + \frac{\nabla p}{\rho} + \frac{\boldsymbol{a_{rot}}}{\rho} + \alpha \boldsymbol{u} = 0 \tag{2.6}$$

where $\boldsymbol{a_{rot}} = \boldsymbol{a_{cent}} + \boldsymbol{a_{cor}}$.

For compressible flows, density changes become important and the energy equation must be part of the system. In these cases, the compressible version of the Navier-Stokes equations is used and can be written as (HIRSCH, 2007):

$$\nabla \cdot (\rho \boldsymbol{u}) = 0 \tag{2.7}$$

$$\nabla \cdot (\rho \boldsymbol{u}\boldsymbol{u}) - \nabla \cdot \left(\mu \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T\right)\right) + \nabla p - \lambda \nabla (\nabla \cdot \boldsymbol{u}) = 0 \tag{2.8}$$

$$\nabla \cdot (\rho \boldsymbol{u} h) + \nabla \cdot \left(\rho \frac{\boldsymbol{u} \cdot \boldsymbol{u}}{2} \boldsymbol{u}\right) + \nabla \cdot \left(-\frac{k}{c_p} \nabla h\right) - \nabla \cdot (\boldsymbol{\tau} \cdot \boldsymbol{u}) = 0 \tag{2.9}$$

where the already presented variables hold their meaning, μ is the dynamic viscosity, λ is the bulk viscosity (here, considered as $-\frac{2}{3}\mu$), h is the specific enthalpy, k is the gas thermal conductivity, c_p is the specific heat, and $\boldsymbol{\tau}$ is the stress tensor (with $\boldsymbol{I}$ as an identity matrix):

$$\boldsymbol{\tau} = \mu \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T\right) + \lambda \left(\nabla \cdot \boldsymbol{u}\right) \boldsymbol{I} \tag{2.10}$$

In this work, perfect gas is used, hence, the density ρ and the sound speed c_{sound} are, respectively, calculated by:

$$\rho = \frac{c_p \, p}{R \, h} \tag{2.11}$$

$$c_{sound} = \sqrt{\gamma R T} \tag{2.12}$$

where γ is the ratio of specific heats (used as 1.4), R is the gas constant (considered 287 $[J/(kg \cdot K)]$) and T is the temperature.

Again, considering the rotating accelerations in the momentum equations, accounting for the work of the centrifugal forces in the energy equation (HIRSCH, 2007) and adding the Brinkman penalization, the complete system of equations can be read as:

$$\nabla \cdot (\rho \boldsymbol{u}) = 0 \tag{2.13}$$

$$\nabla \cdot (\rho \boldsymbol{u}\boldsymbol{u}) + \nabla p - \nabla \cdot \boldsymbol{\tau} + \boldsymbol{a_{rot}} + \alpha \boldsymbol{u} = 0 \tag{2.14}$$

$$\nabla \cdot (\rho \boldsymbol{u} h) + \nabla \cdot \left(\rho \boldsymbol{u} \left[\frac{\boldsymbol{u} \cdot \boldsymbol{u}}{2} - \frac{(\boldsymbol{\omega} \times \boldsymbol{r})^2}{2}\right]\right) + \nabla \cdot \left(-\frac{k}{C_p} \nabla h\right) - \nabla \cdot (\boldsymbol{\tau} \cdot \boldsymbol{u}) = 0 \tag{2.15}$$

As can be seen, only the momentum equations are penalized in the case of compressible flows. The details about this strategy of penalization are presented in Sec. 3.2.

The systems of equations described can be used to simulate laminar flows or calculate instantaneous values of velocity, pressure and temperature of turbulent flows. In the design of engineering machinery, the majority of flows is turbulent and solving the flow equations in the form presented implies in enormous computational requirements. Some additional modeling strategies can be applied to keep the computational requirements within feasible

specifications and the ones used in this work are explained next.

2.2 Turbulence Modeling

Most of the flows encountered in engineering practices are turbulent. While laminar flows are smooth and stable, turbulent flows are chaotic, diffusive, rapid mixing, time dependent and have three dimensional features. A broad range of time and length scales can be observed in this kind of flow. Turbulence is a very active research field with important questions yet to be answered and the most accepted theory to explain how it happens is based on the energy cascade concept developed and proposed by Kolmogorov. According to this theory, a composition of eddies of different sizes is the main mechanism of energy transmission. Each eddy possesses a certain amount of energy that depends on its dimension. The larger eddies break up transferring energy to smaller size eddies in a chained process. The smaller eddies go through similar breakup processes and transfer energy to even smaller eddies. This process continues until the smallest possible eddy is reached. These smallest eddies are at a scale at which the molecular viscosity is effective at dissipating the turbulent kinetic energy as heat. The smallest turbulent eddies are characterized by the Kolmogorov micro scales, being usually named as η for the length scale and t_η for the time scale. The size of the largest eddies (known as integral length scale) is proportional to the size of the geometry involved and the size of the smallest eddies is given by (MOUKALLED; MANGANI; DARWISH, 2016b; FERZIGER; PERIC; STREET, 2020):

$$\eta = \left(\frac{\nu^3}{\epsilon}\right)^{0.25} \tag{2.16}$$

$$t_\eta = \left(\frac{\nu}{\epsilon}\right)^{0.5} \tag{2.17}$$

where ϵ is the average rate of dissipation of turbulent kinetic energy.

Numerical simulations capable of directly solving the Navier-Stokes equations must be able to resolve the entire spectrum of temporal and spatial turbulent scales by capturing from the largest to the smallest eddies. Very fine meshes are necessary since the micro scales are somewhat proportional to the fluid viscosity. This is a computationally demanding approach known as Direct Numerical Simulation (DNS) and has been used by few researchers, mainly with simple geometries and aiming to study turbulent flow features. In order to solve problems with complex geometries, as in the design of machines and devices, statistical approaches can be used to simplify the solution of turbulent flows and reduce the computational requirements. Averaging techniques are applied to approximate random fluctuations observed in turbulent flows. Currently, the most popular approach is based on solving the Reynolds Averaged Navier-Stokes equations. The central idea is to apply a time

averaging to decompose the flow variables into a mean component and a fluctuating one and substitute both in the original equations, resulting in a set of time averaged equations. The decomposition is applied to all governing equations. For incompressible flows, the standard Reynolds averaging results in the Reynolds Averaged Navier-Stokes (RANS) equations. For compressible flows, a mass-weighted averaging technique can be employed. It is popularly known by Favre averaging and leads to the Favre Averaged Navier-Stokes (FANS) equations. In both, the intention is to model all scales of turbulent flow, thus alleviating mesh requirements (MOUKALLED; MANGANI; DARWISH, 2016b).

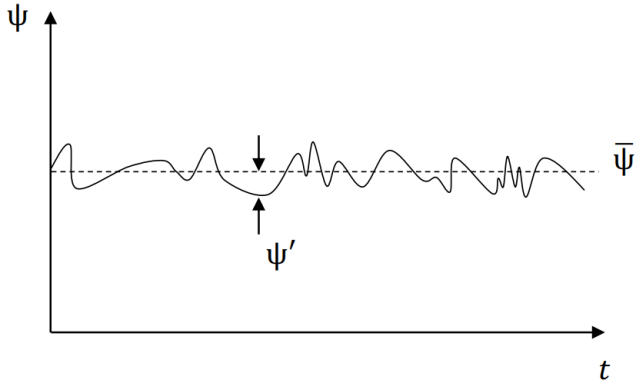
The averaging approach starts by recognizing that, in a steady flow, any variable ψ can be decomposed as the sum of a time averaged value $\overline{\psi}$ and fluctuation component ψ', as in the scheme of Fig. 2.1. So, ψ can be written as (MOUKALLED; MANGANI; DARWISH, 2016b; FERZIGER; PERIC; STREET, 2020):

$$\psi(\boldsymbol{x}, t) = \overline{\psi}(\boldsymbol{x}) + \psi'(\boldsymbol{x}, t) \tag{2.18}$$

where

$$\overline{\psi}(\boldsymbol{x}) = \lim_{T_{avg}\to\infty} \frac{1}{T_{avg}} \int_0^{T_{avg}} \psi(\boldsymbol{x}, t) dt \tag{2.19}$$

Figure 2.1 – Mean and fluctuation components.



In Eq. 2.19, t is the time and the T_{avg} is the averaging interval, which must be large compared to the typical time scale of the fluctuations.

By considering ζ as another scalar variable present in the flow, with fluctuations ζ', the following identities apply (MOUKALLED; MANGANI; DARWISH, 2016b; FERZIGER;

PERIC; STREET, 2020):

$$\overline{\psi'} = 0 \tag{2.20}$$

$$\overline{\psi} = \overline{\overline{\psi}} \tag{2.21}$$

$$\overline{\nabla\psi} = \nabla\overline{\psi} \tag{2.22}$$

$$\overline{\psi + \zeta} = \overline{\psi} + \overline{\zeta} \tag{2.23}$$

$$\overline{\overline{\psi}\zeta} = \overline{\psi}\,\overline{\zeta} \tag{2.24}$$

$$\overline{\overline{\psi}\zeta'} = 0 \tag{2.25}$$

$$\overline{\psi\zeta} = \overline{\psi}\overline{\zeta} + \overline{\psi'\zeta'} \tag{2.26}$$

The averaging of linear terms results in identical terms for the averaged quantity. For a quadratic nonlinear term, two terms are obtained, the product of the average and a covariance. And, for the fluctuations, the average is zero. By decomposing velocity and pressure in its mean and fluctuating components $\boldsymbol{u} = \overline{\boldsymbol{u}} + \boldsymbol{u}'$ and $p = \overline{p} + p'$ and applying the averaging identities (Eqs. 2.20 to 2.26) in the system formed by Eqs. 2.1 and 2.2, the resulting system of equations can be described in terms of the mean and fluctuating components as (MOUKALLED; MANGANI; DARWISH, 2016b; FERZIGER; PERIC; STREET, 2020):

$$\nabla \cdot \overline{\boldsymbol{u}} = 0 \tag{2.27}$$

$$\nabla \cdot \left(\overline{\boldsymbol{u}}\,\overline{\boldsymbol{u}}\right) - \nabla \cdot \left(\nu\left(\nabla\overline{\boldsymbol{u}} + \nabla\overline{\boldsymbol{u}}^T\right)\right) + \frac{\nabla\overline{p}}{\rho} + \nabla \cdot \left(\overline{\boldsymbol{u}'\boldsymbol{u}'}\right) = 0 \tag{2.28}$$

The tensor inside the divergent of last term in Eq. 2.28 is zero only if the two quantities are uncorrelated. This rarely happens in turbulent flows. It also cannot be represented in terms of the mean quantities and modeling this tensor constitutes a classical challenge of turbulence modeling. The tensor represented by $\overline{\boldsymbol{u}'\boldsymbol{u}'}$ is known as the Reynolds stress tensor and can be modelled based on the Boussinesq hypothesis, which assumes the Reynolds stress to be a linear function of the mean velocity gradients (WILCOX, 2006):

$$\overline{\boldsymbol{u}'\boldsymbol{u}'} = \nu_t\left(\nabla\overline{\boldsymbol{u}} + \nabla\overline{\boldsymbol{u}}^T\right) - \frac{2}{3}\left(\kappa + \nu_t(\nabla \cdot \overline{\boldsymbol{u}})\right)\boldsymbol{I} \tag{2.29}$$

where κ is the turbulent kinetic energy, defined as:

$$\kappa = \frac{1}{2}\overline{\boldsymbol{u}'\boldsymbol{u}'} \tag{2.30}$$

and ν_t is, now, a flow dependent property known as turbulent viscosity. The turbulent viscosity can be evaluated using a variety of turbulence models (MOUKALLED; MANGANI; DARWISH, 2016b; FERZIGER; PERIC; STREET, 2020).

Finally, the RANS equations considering rotating accelerations and the Brinkman penalization can be written as :

$$\nabla \cdot \overline{\boldsymbol{u}} = 0 \tag{2.31}$$

$$\nabla \cdot \left(\overline{\boldsymbol{u}}\,\overline{\boldsymbol{u}}\right) - \nabla \cdot \left((\nu + \nu_t)\left(\nabla \overline{\boldsymbol{u}} + \nabla \overline{\boldsymbol{u}}^T\right)\right) + \frac{\nabla \overline{p}}{\rho} + \frac{\boldsymbol{a_{rot}}}{\rho} + \alpha \overline{\boldsymbol{u}} = 0 \tag{2.32}$$

It resembles the original system of equations, differing only in the presence of ν_t.

When compressible flows are being calculated, beyond velocity and pressure fluctuations, density and temperature fluctuations must also be accounted. If the averaging procedure used is the same as in the incompressible flow equations, the resulting system contains additional terms that require additional correlations making the closure problem even more challenging. This situation can be simplified by using a density-weighted averaging. This is the mass-averaging or Favre-averaging and, when applied to the velocity, the mass averaged velocity $\tilde{\boldsymbol{u}}$ is defined as (WILCOX, 2006):

$$\tilde{\boldsymbol{u}} = \frac{1}{\overline{\rho}} \lim_{T_{avg} \to \infty} \frac{1}{T_{avg}} \int_0^{T_{avg}} \rho(\boldsymbol{x}, t) \boldsymbol{u}(\boldsymbol{x}, t) dt \tag{2.33}$$

The decomposition of the instantaneous velocity splits it in its mass-averaged and fluctuation parts (WILCOX, 2006):

$$\boldsymbol{u}(\boldsymbol{x}, t) = \tilde{\boldsymbol{u}}(\boldsymbol{x}) + \boldsymbol{u}''(\boldsymbol{x}, t) \tag{2.34}$$

The other state variables are decomposed as:

$$\rho = \overline{\rho} + \rho' \tag{2.35}$$

$$p = \overline{p} + p' \tag{2.36}$$

$$h = \tilde{h} + h'' \tag{2.37}$$

After inserting Eqs. 2.34 to 2.37 in the system composed by Eqs. 2.13 to 2.15 and performing some rearrangements, the Favre Averaged Navier-Stokes with the rotating and Brinkman terms can be obtained:

$$\nabla \cdot (\overline{\rho} \tilde{\boldsymbol{u}}) = 0 \tag{2.38}$$

$$\nabla \cdot (\overline{\rho} \tilde{\boldsymbol{u}} \tilde{\boldsymbol{u}}) + \nabla \overline{p} - \nabla \cdot \overline{\boldsymbol{\tau}} + \boldsymbol{a_{rot}} + \alpha \tilde{\boldsymbol{u}} + \nabla \cdot \left(\overline{\rho \boldsymbol{u}'' \boldsymbol{u}''}\right) = 0 \tag{2.39}$$

$$\nabla \cdot \left(\overline{\rho} \tilde{\boldsymbol{u}} \tilde{h}\right) + \nabla \cdot \left(\overline{\rho} \tilde{\boldsymbol{u}} \left[\frac{\tilde{\boldsymbol{u}} \cdot \tilde{\boldsymbol{u}}}{2} - \frac{(\boldsymbol{\omega} \times \boldsymbol{r})^2}{2}\right]\right) + \nabla \cdot \left(-\frac{k}{c_p} \nabla \tilde{h}\right) - \nabla \cdot (\overline{\boldsymbol{\tau}} \cdot \tilde{\boldsymbol{u}}) +$$
$$+ \nabla \cdot \left(-\overline{\boldsymbol{\tau} \cdot \boldsymbol{u}''} + \overline{\rho \frac{\boldsymbol{u}'' \cdot \boldsymbol{u}''}{2} \boldsymbol{u}''} + \overline{\frac{\rho \boldsymbol{u}'' \cdot \boldsymbol{u}''}{2}} \tilde{\boldsymbol{u}} + \overline{\rho \boldsymbol{u}'' h''} + \overline{\rho \boldsymbol{u}'' \boldsymbol{u}''} \tilde{\boldsymbol{u}}\right) = 0 \tag{2.40}$$

with the stress tensor and the perfect gas equations being:

$$\overline{\boldsymbol{\tau}} = \mu\left(\nabla\tilde{\boldsymbol{u}} + \nabla\tilde{\boldsymbol{u}}^T\right) - \frac{2}{3}\mu\left(\nabla\cdot\tilde{\boldsymbol{u}}\right)\boldsymbol{I} \tag{2.41}$$

$$\overline{\rho} = \frac{c_p\,\overline{p}}{R\,\tilde{h}} \tag{2.42}$$

In the last divergent of Eq. 2.40, complicated correlations written in terms of the fluctuations arise. According to Wilcox (2006), the first and second terms inside the divergent can be ignored for flows with Mach numbers up to the supersonic range. The third term is also neglected since it is usually $\ll \tilde{h}$. The fourth term is the turbulent heat flux vector and is assumed to be proportional to the mean temperature gradient:

$$\overline{\rho\boldsymbol{u}''h''} = -\frac{\mu_t}{Pr_t}\nabla\tilde{h} \tag{2.43}$$

where Pr_t is known as the turbulent Prandtl number (usually considered ≈ 0.9).

The last term can be approximated by a generalization of the Boussinesq hypothesis to compressible flows (WILCOX, 2006):

$$\overline{\rho\boldsymbol{u}''\boldsymbol{u}''} = \mu_t\left(\nabla\tilde{\boldsymbol{u}} + \nabla\tilde{\boldsymbol{u}}^T\right) - \frac{2}{3}\left(\overline{\rho}\kappa + \mu_t(\nabla\cdot\tilde{\boldsymbol{u}})\right)\boldsymbol{I} \tag{2.44}$$

where the turbulent viscosity is now given by μ_t.

Finally, remembering that the laminar Prandtl number is (WILCOX, 2006):

$$Pr = \frac{c_p\mu}{k} \tag{2.45}$$

The Favre Averaged Navier-Stokes with the rotating and Brinkman terms can be written as:

$$\nabla\cdot\left(\overline{\rho}\tilde{\boldsymbol{u}}\right) = 0 \tag{2.46}$$

$$\nabla\cdot\left(\overline{\rho}\tilde{\boldsymbol{u}}\tilde{\boldsymbol{u}}\right) + \nabla\overline{p} - \nabla\cdot\overline{\boldsymbol{\tau}}_{turb} + \boldsymbol{a_{rot}} + \alpha\tilde{\boldsymbol{u}} = 0 \tag{2.47}$$

$$\nabla\cdot\left(\overline{\rho}\tilde{\boldsymbol{u}}\left[\tilde{h} + \frac{\tilde{\boldsymbol{u}}\cdot\tilde{\boldsymbol{u}}}{2} - \frac{(\boldsymbol{\omega}\times\boldsymbol{r})^2}{2}\right]\right) - \nabla\cdot\left(\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\nabla\tilde{h}\right) - \nabla\cdot\left(\overline{\boldsymbol{\tau}}_{turb}\cdot\tilde{\boldsymbol{u}}\right) = 0 \tag{2.48}$$

where the stress tensor $\overline{\boldsymbol{\tau}}_{turb}$ has effects from the turbulent viscosity μ_t:

$$\overline{\boldsymbol{\tau}}_{turb} = (\mu + \mu_t)\left[\left(\nabla\tilde{\boldsymbol{u}} + \nabla\tilde{\boldsymbol{u}}^T\right) - \frac{2}{3}\left(\nabla\cdot\tilde{\boldsymbol{u}}\right)\boldsymbol{I}\right] \tag{2.49}$$

The sets of equations given by Eq. 2.31 to Eq. 2.32 and Eq. 2.46 to 2.48 can be used when solving topology optimization problems for incompressible and compressible rotating turbulent flows.

In order to complete the sets of equations, the turbulent viscosity (ν_t or μ_t) must be calculated by some mean. This comes from the turbulence models. Some models, such as Spalart Allmaras (SPALART; ALLMARAS, 1994), $\kappa - \epsilon$ (JONES; LAUNDER, 1972), $\kappa - \omega$ (WILCOX, 1988) and SST $\kappa - \omega$ (MENTER, 1994) became very popular due to the combination of satisfactory accuracy and feasible computational requirements. However, no model is universally applicable to all flow conditions. Each one has certain advantages and strengths and also limitations. In this work, a recent turbulence model named Wray Agarwal 2018 (WA2018) is used (HAN; RAHMAN; AGARWAL, 2018). Its development can be seen throughout the works of Wray and Agarwal (2014), Han et al. (2015) and Xu, Wray and Agarwal (2017). It is a one equation model and has been chosen by its simplicity and accuracy. The model is based on the SST $\kappa - \omega$, by defining a turbulent scalar variable $R = \kappa/\omega$. The turbulent viscosity is calculated by:

$$\nu_t = f_\mu R \tag{2.50}$$

$$\mu_t = \rho f_\mu R \tag{2.51}$$

where:

$$f_\mu = \frac{\chi^3}{\chi^3 + C_w^3} \tag{2.52}$$

$$\chi = \frac{R}{\nu} \tag{2.53}$$

$$C_w = 8.54 \tag{2.54}$$

The transport equations for incompressible and compressible flows are, respectively (HAN; RAHMAN; AGARWAL, 2018):

$$\begin{aligned}\nabla \cdot (\boldsymbol{u}R) - \nabla \cdot ((\sigma_R R + \nu)\nabla R) - C_1 RS - f_1 C_{2\kappa\omega}\frac{R}{S}(\nabla R \cdot \nabla S) + \\ +(1 - f_1) min\left[C_{2\kappa\epsilon}R^2\left(\frac{\nabla S \cdot \nabla S}{S^2}\right), C_m(\nabla R \cdot \nabla R)\right] + \alpha R = 0\end{aligned} \tag{2.55}$$

$$\begin{aligned}\nabla \cdot (\rho\boldsymbol{u}R) - \nabla \cdot (\rho(\sigma_R R + \nu)\nabla R) - \rho C_1 RS - \rho f_1 C_{2\kappa\omega}\frac{R}{S}(\nabla R \cdot \nabla S) + \\ +\rho(1 - f_1) min\left[C_{2\kappa\epsilon}R^2\left(\frac{\nabla S \cdot \nabla S}{S^2}\right), C_m(\nabla R \cdot \nabla R)\right] + \alpha R = 0\end{aligned} \tag{2.56}$$

where:

$$\sigma_R = f_1 \left(\sigma_{\kappa\omega} - \sigma_{\kappa\epsilon}\right) + \sigma_{\kappa\epsilon} \tag{2.57}$$

$$C_1 = f_1 \left(C_{1\kappa\omega} - C_{1\kappa\epsilon}\right) + C_{1\kappa\epsilon} \tag{2.58}$$

$$S = \sqrt{(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T) : (\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T)} \tag{2.59}$$

$$f_1 = tanh\left(arg_1^4\right) \tag{2.60}$$

$$C_{2\kappa\omega} = \frac{C_{1\kappa\omega}}{\kappa^2} + \sigma_{\kappa\omega} \tag{2.61}$$

$$C_{2\kappa\epsilon} = \frac{C_{1\kappa\epsilon}}{\kappa^2} + \sigma_{\kappa\epsilon} \tag{2.62}$$

$$C_m = 8.0 \tag{2.63}$$

$$arg_1 = \frac{\nu + R}{2} \frac{\eta^2}{C_\mu \kappa\omega} \tag{2.64}$$

$$\kappa = \frac{\nu_t S}{\sqrt{C_\mu}} \tag{2.65}$$

$$\omega = \frac{S}{\sqrt{C_\mu}} \tag{2.66}$$

$$\eta = S\ max\left(1, \left|\frac{W}{S}\right|\right) \tag{2.67}$$

$$W = \sqrt{(\nabla \boldsymbol{u} - \nabla \boldsymbol{u}^T) : (\nabla \boldsymbol{u} - \nabla \boldsymbol{u}^T)} \tag{2.68}$$

$$C_{1\kappa\epsilon} = 0.1284 \tag{2.69}$$

$$C_\mu = 0.09 \tag{2.70}$$

$$\sigma_{\kappa\omega} = 0.72 \tag{2.71}$$

$$\sigma_{\kappa\epsilon} = 1.0 \tag{2.72}$$

and the boundary conditions recommended are (NASA, 2022):

$$R_{wall} = 0 \tag{2.73}$$

$$R_{farfield} = 3\nu \text{ to } 5\nu \tag{2.74}$$

In Eqs. 2.55 and 2.56, the last term (αR) is not part of the original model, but added here in the spirit of the Brinkman penalization to force the R variable to go to zero in the solid regions (as explained in Section 3.2). Also, an assessment to verify the behavior of the penalization of the turbulence model is presented in Chapter 6.

2.3 Finite Volume Method

The equations previously described, be it instantaneous or averaged, can be solved analytically only for a limited number of simple cases. The solutions of these cases are

extremely useful in helping to understand the fluid flow behavior, but can rarely be used directly in engineering analysis or design. Thus, engineers have traditionally been forced to try different strategies instead of directly solving the equations. In the most common approaches, simplifications based on a combination of approximations, dimensional analysis and some empirical data have been employed. Although very successful in many cases, these strategies can only be used when the system is described by one or two parameters, so applications to complex geometries can become very challenging. Also, the use of experiments can be expensive and time consuming, if not impossible, depending on the case. With the development of electronic computers, a different path became available among the alternatives to study and solve partial differential equations. Numerical methods created a long time ago could be implemented and new ones have been developed. With the increasing of the computing capabilities, these methods have been becoming more sophisticated and have been solving harder problems in shorter times. Among them, the group of methods intended to solve fluid flow problems is known by *Computational Fluid Dynamics* (CFD). Several options of methods to solve the flow equations can be found and the most popular are the Finite Differences Method (FDM), the Finite Element Method (FEM) and the Finite Volume Method (FVM) (FERZIGER; PERIC; STREET, 2020). Among them, the FVM has been widely used to solve flow problems. Its flexibility and the fact that the discretization is carried out directly in the physical space with no need of any transformation between the physical and the computational coordinate systems made it very attractive as a choice to solve complex problems. Also, its numerics mirrors the physics and the conservation principles it models, the terms that need to be approximated have a physical meaning, the method is conservative by construction and is, perhaps, the simplest to understand and to program. All these factors contributed to its popularity when the objective is to calculate flows in several different conditions (MOUKALLED; MANGANI; DARWISH, 2016b; FERZIGER; PERIC; STREET, 2020). In this work, the FVM is used to solve the flow equations and a brief introduction is presented next.

The starting point is the integral form of the governing equations. The domain where the equations are going to be solved is divided in a set of control volumes (CV) and the equations are applied to each CV. At the centroid of each one, a computational node is considered, where the variable values are calculated. Interpolations are used to approximate the variable values at the CV surface. Each surface and volume integral is approximated using suitable quadrature formulas. As a result, an algebraic equation is obtained for each CV, with contributions from the centroid of the CV and the neighbour CVs. For a general scalar variable ϕ, the conservation equation can be written as (MOUKALLED; MANGANI; DARWISH, 2016b; FERZIGER; PERIC; STREET, 2020):

$$\frac{\partial \rho \phi}{\partial t} + \nabla \cdot (\rho \boldsymbol{u} \phi) = \nabla \cdot \left(\Gamma^{\phi} \nabla \phi \right) + Q^{\phi} \tag{2.75}$$

where $\boldsymbol{u}$ is the fluid velocity and Γ^{ϕ} is the diffusion coefficient of ϕ.

The steady-state form of Eq. 2.75 is:

$$\nabla \cdot (\rho \boldsymbol{u} \phi) = \nabla \cdot \left(\Gamma^{\phi} \nabla \phi \right) + Q^{\phi} \tag{2.76}$$

By integrating over a cell P, Eq. 2.76 becomes:

$$\int_{V_P} \nabla \cdot (\rho \boldsymbol{u} \phi) \, dV = \int_{V_P} \nabla \cdot \left(\Gamma^{\phi} \nabla \phi \right) dV + \int_{V_P} Q^{\phi} dV \tag{2.77}$$

and, by using the divergence theorem in the first two terms:

$$\int_{\Gamma_P} (\rho \boldsymbol{u} \phi) \cdot d\boldsymbol{S} = \int_{\Gamma_P} \left(\Gamma^{\phi} \nabla \phi \right) \cdot d\boldsymbol{S} + \int_{V_P} Q^{\phi} dV \tag{2.78}$$

where $\boldsymbol{S}$ is a surface area vector.

Replacing the integrals over cell P by a summation of the flux terms over the faces of the cell and considering one integration point (MOUKALLED; MANGANI; DARWISH, 2016b):

$$\sum_{f} \left(\rho \boldsymbol{u} \phi - \Gamma^{\phi} \nabla \phi \right)_f \cdot \boldsymbol{S}_f = Q^{\phi} V_P \tag{2.79}$$

where the subscript f denotes calculation at the CV faces.

The fluxes between the cell P and a neighbour N can be linearized and decomposed as:

$$\left(\rho \boldsymbol{u} \phi - \Gamma^{\phi} \nabla \phi \right)_f \cdot \boldsymbol{S}_f = FluxP_f \; \phi_P + FluxN_f \; \phi_N + FluxV_f \tag{2.80}$$

$$Q^{\phi} V_P = FluxP \; \phi_P + FluxV \tag{2.81}$$

where $FluxP_f$ and $FluxN_f$ are linearization coefficients used in the representation of the contributions of the cell P and N. The term $FluxV_f$ describes a nonlinear contribution that cannot be expressed in terms of ϕ_P and ϕ_N. The values of $FluxP_f$, $FluxN_f$ and $FluxV_f$ depend on the discretization schemes used. Detailed explanations on these schemes can be found in Ferziger, Peric and Street (2020) and Moukalled, Mangani and Darwish (2016b).

Inserting Eqs. 2.80 and 2.81 in Eq. 2.79 results in:

$$a_P \phi_P + \sum_{F} (a_N \phi_N) - b_P = 0 \tag{2.82}$$

where the coefficients are:

$$a_P = \sum_f FluxP_f - FluxP \tag{2.83}$$

$$a_N = FluxN_f \tag{2.84}$$

$$b_P = -\sum_f FluxP_f + FluxP \tag{2.85}$$

Each CV has an equation in the form of 2.82 relating the value of ϕ in its center to ϕ at the center of the neighbour cells. Since there is an equation for each CV and one unknown per center of CV, the number of equations and unknowns are equal and a well-posed system of equations is formed (MOUKALLED; MANGANI; DARWISH, 2016b; FERZIGER; PERIC; STREET, 2020). The solution of this system to calculate velocity, pressure and temperature is explained in Section 5.1.

3 TOPOLOGY OPTIMIZATION OF FLOW PROBLEMS

In this work, Topology Optimization is applied to incompressible and compressible flow problems. When turbulence is considered, the frozen turbulence hypothesis is used. The continuous approach of the adjoint method is used for incompressible flows and the discrete approach is employed for incompressible and compressible flows. In the next sections, the objective functions and constraints considered are explained. Also, the material model used is detailed.

3.1 Objective Function and Constraints

The optimization problem is defined as:

$$\begin{aligned} \text{Minimize :} \quad & J(\boldsymbol{w}(\boldsymbol{\rho_{des}}), \boldsymbol{\rho_{des}}) \\ \boldsymbol{\rho_{des}} & \\ \text{subjected to :} \quad & \boldsymbol{R}(\boldsymbol{w}(\boldsymbol{\rho_{des}}), \boldsymbol{\rho_{des}}) = 0 \\ & c(\boldsymbol{\rho_{des}}) = 0 \\ & 0 \leq \boldsymbol{\rho_{des}} \leq 1 \end{aligned} \tag{3.1}$$

where J is the objective function to be minimized, $\boldsymbol{w}$ is a vector with the state variables velocity, pressure and enthalpy (for compressible flow), $\boldsymbol{\rho_{des}}$ is the vector of design variables, $\boldsymbol{R}$ represents the system of equations written in the residual form, and c is a constraint on the problem. In this work, a volume constraint is considered. It is applied to impose the percentage of the domain, V_{frac}, to be fluid at the end of the optimization and is defined as:

$$c = \frac{\int_\Omega \rho_{des} \, d\Omega}{\int_\Omega \, d\Omega} - V_{frac} \tag{3.2}$$

The objective functions considered aim to obtain a flow path design that minimizes energy losses. For the non-rotating incompressible flows, the objective is to minimize the energy dissipated by the flow in the control volume. The function is shown in Eq. 3.3 and can also be seen in the works by Othmer (2008) and Papoutsis-Kiachagias and Giannakoglou (2016):

$$J_{stat,incomp} = -\int_\Gamma \left(p + \rho \frac{\boldsymbol{u} \cdot \boldsymbol{u}}{2} \right) (\boldsymbol{u} \cdot \hat{\boldsymbol{n}}) \; d\Gamma \tag{3.3}$$

When the flow is incompressible and subject to rotation, the goal is to design a high

efficiency device that adds energy to the flow, therefore the objective function considered is given by the negative of the pump efficiency equation (WHITE, 1999), given by:

$$J_{rot,incomp} = -\frac{P_f}{\boldsymbol{T}\cdot\boldsymbol{\omega}} \tag{3.4}$$

where P_f is the power added to the flow and $\boldsymbol{T}$ is the total torque over the control volume. Considering $\boldsymbol{v}$ as the velocity in the system of reference that is not rotating ($\boldsymbol{v} = \boldsymbol{u}+\boldsymbol{\omega}\times\boldsymbol{r}$), the power and the torque are calculated by Eqs. 3.5 and 3.6, respectively (WHITE, 1999):

$$P_f = \int_\Gamma \left(p + \rho\frac{\boldsymbol{v}\cdot\boldsymbol{v}}{2}\right)(\boldsymbol{v}\cdot\hat{\boldsymbol{n}})\ d\Gamma \tag{3.5}$$

$$\boldsymbol{T} = \int_\Gamma \rho\,(\boldsymbol{r}\times\boldsymbol{v})\,(\boldsymbol{v}\cdot\hat{\boldsymbol{n}})\ d\Gamma \tag{3.6}$$

In the continuous adjoint approach, the objective function is adapted to keep the mathematical simplicity when deriving the adjoint problem. The minimized quantity is the power dissipated inside the domain, calculated by the difference between P_f and the power used to drive the component (product of torque and rotation):

$$P_{diss} = \boldsymbol{T}\cdot\boldsymbol{\omega} - P_f \tag{3.7}$$

The power dissipation defined in Eq. 3.7 can be simplified by using the absolute velocity expression:

$$\begin{aligned}
P_{diss} &= \int_\Gamma \rho\boldsymbol{\omega}\cdot(\boldsymbol{r}\times\boldsymbol{v})\,(\boldsymbol{v}\cdot\hat{\boldsymbol{n}})\ d\Gamma - \int_\Gamma \left(p+\rho\frac{\boldsymbol{v}\cdot\boldsymbol{v}}{2}\right)(\boldsymbol{v}\cdot\hat{\boldsymbol{n}})\ d\Gamma \\
&= \int_\Gamma \Big[\rho\boldsymbol{\omega}\cdot(\boldsymbol{r}\times(\boldsymbol{u}+\boldsymbol{\omega}\times\boldsymbol{r}))\,((\boldsymbol{u}+\boldsymbol{\omega}\times\boldsymbol{r})\cdot\hat{\boldsymbol{n}})\Big]d\Gamma + \\
&- \int_\Gamma \left[\left(p+\rho\frac{(\boldsymbol{u}+\boldsymbol{\omega}\times\boldsymbol{r})\cdot(\boldsymbol{u}+\boldsymbol{\omega}\times\boldsymbol{r})}{2}\right)((\boldsymbol{u}+\boldsymbol{\omega}\times\boldsymbol{r})\cdot\hat{\boldsymbol{n}})\right]d\Gamma \\
&= \int_\Gamma \Big[(\rho\boldsymbol{\omega}\cdot(\boldsymbol{r}\times\boldsymbol{u}) + \rho\boldsymbol{\omega}\cdot(\boldsymbol{r}\times(\boldsymbol{\omega}\times\boldsymbol{r})))\,(\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\Big]d\Gamma + \\
&- \int_\Gamma \left[\left(p+\rho\frac{(\boldsymbol{u}\cdot\boldsymbol{u}+2\boldsymbol{u}\cdot(\boldsymbol{\omega}\times\boldsymbol{r})+(\boldsymbol{\omega}\times\boldsymbol{r})\cdot(\boldsymbol{\omega}\times\boldsymbol{r}))}{2}\right)(\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\right]d\Gamma \\
&= \int_\Gamma \rho\frac{(\omega r)^2}{2}\,(\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\ d\Gamma - \int_\Gamma \left(p+\rho\frac{\boldsymbol{u}\cdot\boldsymbol{u}}{2}\right)(\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\ d\Gamma
\end{aligned} \tag{3.8}$$

In Eq. 3.8, the first term is constant given that it is a boundary integral term and the inner and outer radius are considered fixed, so the objective function considered is defined only by the last integral. Note that it only depends on the specific mass, pressure and relative velocity and the mathematical expression is equivalent to Eq. 3.3.

In the case of non-rotating compressible flows, the variation of entropy s in a control

volume is considered as the objective function. Entropy is directly associated with losses and heat transfer. For this work, no heat is exchanged with the outer media through the walls, so all entropy generation is related to losses inside the flow. By minimizing the difference of entropy between the inlet and outlet of the domain an efficient flow path is obtained at the end of the optimization process (SÁ; OKUBO; SILVA, 2021). The expression to account for the entropy change inside the control volume for a perfect gas is (BORGNAKKE; SONNTAG, 2008; PRITCHARD, 2010):

$$J_{stat,comp} = \int_{\Gamma} s\rho(\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\, d\Gamma = \int_{\Gamma} \left(C_p\, ln\left(\frac{T}{T_{ref}}\right) - R\, ln\left(\frac{p}{p_{ref}}\right)\right)\, \rho\,(\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\, d\Gamma \tag{3.9}$$

where the subscripts *ref* indicate an arbitrary state of reference.

Finally, for rotating compressible flows, the compressor isentropic efficiency is considered and the optimization is conducted by minimizing the negative of the isentropic efficiency, given by (DIXON; HALL, 2010):

$$J_{rot,comp} = -\frac{h_{out,s} - h_{in}}{h_{out} - h_{in}} \tag{3.10}$$

where the subscripts *in* and *out* refer to the inlet and the outlet of the domain and the subscript s indicates an ideal compression process. To calculate these values, the enthalpy is mass averaged at the respective surfaces.

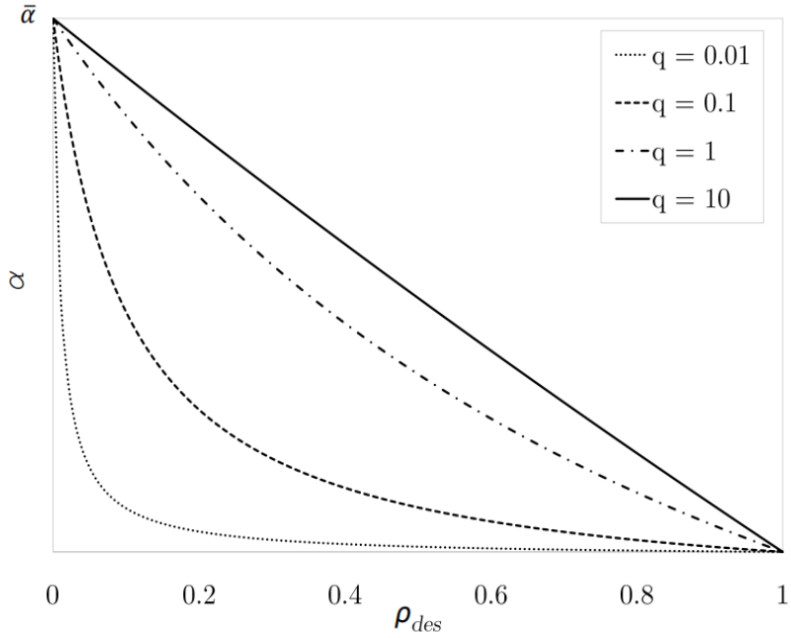
3.2 Material Model for Compressible Flow

In Topology Optimization, the most common way of representing boundaries inside the domain is by modifying the governing equations with some type of penalization to enforce Dirichlet boundary conditions. This usually avoids the need to change the geometry and rebuild the mesh during the optimization. In the pioneering work of Borrvall and Petersson (2003), the modification employed to represent solid regions inside the domain is based on the porous media representation and is performed by adding the term $\alpha\boldsymbol{u}$ (see Eq. 2.6) to the momentum equations. This constitutes the Brinkman penalization. The scalar field α represents an inverse permeability. Regions where α is zero have fluid behavior, since the momentum equations assume its original form. Regions where α is high, present low velocities and can be interpreted as solid portions. Still in the work of Borrvall and Petersson (2003), an interpolation function is presented, relating the design variables ρ_{des} to the penalization variable α. This function can be seen in Eq. 3.11:

$$\alpha = \overline{\alpha} + (\underline{\alpha} - \overline{\alpha})\,\rho_{des}\frac{(1+q)}{(\rho_{des}+q)} \tag{3.11}$$

The design variables ρ_{des} can range from 0 to 1, representing solid and fluid, respectively. Maximum and minimum values allowed for α are $\overline{\alpha}$ and $\underline{\alpha}$. The relation between α and ρ_{des} is controlled by the changing the parameter q and different behaviors for this function for several values of q can be seen in Fig. 3.1.

Figure 3.1 – Interpolation function for different values of q.



This sort of penalization with the interpolation function described have been widely explored in topology optimization applied to incompressible flow problems and is used in this work. However, as topology optimization for compressible flows is still a field to be explored, no material model is established yet and finding a suitable representation for the solid portions is part of the task. Even so, some authors worked with the objective of finding a way to represent solid regions inside compressible viscous flows. Next, a short description of the main works is made.

Liu and Vasilyev (2007) study wave propagation and propose an extension of the Brinkman penalization to compressible flows by modifying continuity, momentum and energy equations in a similar way as it is done to incompressible flows. By comparing numerical simulations with analytical solutions, the authors conclude that the penalization produces acceptable results for the cases studied. Boiron, Chiavassa and Donat (2009) study compressible flows by only modifying momentum and energy equations to represent the solid regions. The authors present some numerical tests by comparing with solutions from a commercial software. Results are considered very promising, showing a good agreement between the penalized and the body-fitted mesh cases. Piquet, Roussel and Hadjadj (2016) compare penalization methods with immersed boundary for supersonic flow cases. Using a penalization similar to the study of Boiron, Chiavassa and Donat (2009), the authors also

conclude that penalization can produce accurate results.

In the mentioned studies, the penalization is performed by adding a term to momentum and energy equations that can enforce a Dirichlet boundary condition inside the domain. The general idea can be seen from Eqs. 3.12 and 3.13:

$$Momentum + \chi \frac{1}{\Phi}(\boldsymbol{u} - \boldsymbol{u_p}) = 0 \tag{3.12}$$

$$Energy + \chi \frac{1}{\Phi}(T - T_p) = 0 \tag{3.13}$$

The scalar field χ can only assume values 0 or 1. For regions inside the solid, $\chi = 1$, meaning that penalization is active. For regions outside the solid, $\chi = 0$, meaning that penalization is not active and the equations assume their original form. The scalar field Φ is the inverse permeability and the small its value is, the more penalized is the model. All studies show the penalized model results are closer to the reference results with lower values of Φ. It is important to notice that this kind of penalization only enforces a Dirichlet boundary condition, so for velocity, $\boldsymbol{u_p}$ is enforced in the penalized region and, for energy, the same happens with T_p. In topology optimization, it is common to use $\boldsymbol{u_p} = \mathbf{0}$, so a solid region without movement is being represented. The only work found that proposes a way to enforce Neumann and Robin boundary conditions is from Brown-Dymkoski, Kasimov and Vasilyev (2014). They propose a method entitled the Characteristic-Based Volume Penalization (CBVP), where, for a constitutive equation:

$$\frac{\partial u}{\partial t} = RHS \tag{3.14}$$

Dirichlet, Neumann and Robin boundary conditions can be applied by the modified equations:

$$\frac{\partial u}{\partial t} = (1 - \chi) \times RHS - \frac{\chi}{\Phi}\left(u - u_p(x,t)\right) + \chi \nu_n \nabla^2 u \tag{3.15}$$

$$\frac{\partial u}{\partial t} = (1 - \chi) \times RHS - \frac{\chi}{\Phi}\left(\nabla u \cdot \hat{\boldsymbol{n}} - q(x,t)\right) \tag{3.16}$$

$$\frac{\partial u}{\partial t} = (1 - \chi) \times RHS - \frac{\chi}{\Phi}\left(a(x,t)u + b\nabla u \cdot \hat{\boldsymbol{n}} - g\right) \tag{3.17}$$

Variables χ and Φ have the same behavior as described before and do not necessarily need to have the same value for all the boundary conditions. In the Dirichlet boundary condition, numerical ν_n is used to guarantee no discontinuities between solid and fluid regions. In Neumann case, $\hat{\boldsymbol{n}}$ is an inward-oriented surface normal and $q(x,t)$ is the desired Neumann condition. For the Robin condition, a, b and g respects $a(x,t)u + b\partial_n u = g(x,t)$. As in the other works, the authors compare some tests with numerical or analytical cases, finding an acceptable agreement between them.

When simulating rotating compressible flows, as is usually the case for rotors of compressors and turbines, the walls are considered adiabatic. Examples can be seen in the works of Kim et al. (2010), Mangani, Casartelli and Mauri (2012), Ju, Zhang and Chi (2012) and Guo et al. (2015) among others. So, the enforcement of Dirichlet boundary conditions for the energy equation is not adequate. In this work, the strategy used to impose solid regions in compressible flow problems is different from the studies mentioned. In Eqs. 2.14 and 2.15, only the momentum equations are penalized. By looking at them, it is possible to see that, in regions where velocities are close to zero, the first and second terms of the energy equation go to zero. Under the assumption of small velocity gradients ($\nabla \boldsymbol{u} \approx 0$) inside the solid portions, the last term also vanishes and the energy equation is reduced to (SÁ; OKUBO; SILVA, 2021):

$$\nabla \cdot \left(-\frac{k}{C_p} \nabla h \right) = 0 \tag{3.18}$$

Thus, integrating a solid region Ω_{solid}, enclosed in a surface Γ_{solid}, it is possible to use the Gauss theorem to see that:

$$\int_{\Omega_{solid}} \left(\nabla \cdot \left(-\frac{k}{C_p} \nabla h \right) \right) \, d\Omega_{solid} = \int_{\Gamma_{solid}} \left(-\frac{k}{C_p} \nabla h \cdot \hat{\boldsymbol{n}} \right) \, d\Gamma_{solid} = 0 \tag{3.19}$$

Therefore, if the material model penalization is applied only in the momentum equation and the solid region has negligible velocities ($\boldsymbol{u} \approx 0$), most terms of the energy equation loose influence, and the remaining part of the energy equation describes a portion of the domain surrounded by an almost adiabatic perimeter. An example to assess this hypothesis is presented in the numerical example at Sec. 6.4.

4 SENSITIVITY CALCULATION

The continuous and the discrete approaches of the adjoint method are employed to calculate the sensitivities for the optimization problem. The continuous approach is only applied to incompressible flow cases. Compressible flow problems are highly nonlinear and the use of the continuous approach for them results in the need of hand derivation of the adjoint, which proved to be challenging. Also, the implementation of the resulting adjoint system is not a trivial task. Thus, for compressible flows, the discrete approach is used. The strategy developed for the discrete approach is flexible and optimization cases considering incompressible flow are also presented.

In the sensitivity calculations for compressible turbulent flows, the frozen turbulence hypothesis is used. This means that the turbulence model is not considered when deriving and solving the adjoint problem, so the turbulent viscosity is used as a flow property.

In the next sections, the derivation of the adjoint system is explained for both (continuous and discrete) approaches and a sensitivity verification is performed by comparing the values calculated by the adjoint method with values calculated directly by finite differences.

4.1 Continuous Adjoint

Considering a generic objective function J, where J_Ω represents the terms in the interior of the domain and J_Γ the terms at the boundary (Eq. 4.1), it is possible to use the Lagrange multipliers approach to define the Lagrangian as in Eq. 4.2:

$$J = \int_\Omega J_\Omega \, d\Omega + \int_\Gamma J_\Gamma \, d\Gamma \tag{4.1}$$

$$L = J + \int_\Omega \boldsymbol{R_u} \cdot \boldsymbol{\psi_u} \, d\Omega + \int_\Omega R_c \, \psi_c \, d\Omega \tag{4.2}$$

The continuity and momentum equations (Eqs. 2.5 and 2.6) are represented by R_c and $\boldsymbol{R_u}$, respectively, and the Lagrange multipliers for them are given by ψ_c and $\boldsymbol{\psi_u}$. The total variation of the Lagrangian can be computed as:

$$\delta L = \delta J + \int_\Omega \delta \boldsymbol{R_u} \cdot \boldsymbol{\psi_u} \, d\Omega + \int_\Omega \delta R_c \, \psi_c \, d\Omega, \tag{4.3}$$

and the terms are defined as:

$$\delta J = \int_\Omega \left(\frac{\partial J_\Omega}{\partial \boldsymbol{u}} \cdot \delta \boldsymbol{u} + \frac{\partial J_\Omega}{\partial p} \delta p + \frac{\partial J_\Omega}{\partial \rho_{des}} \delta \rho_{des} \right) d\Omega + \\ + \int_\Gamma \left(\frac{\partial J_\Gamma}{\partial \boldsymbol{u}} \cdot \delta \boldsymbol{u} + \frac{\partial J_\Gamma}{\partial p} \delta p + \frac{\partial J_\Gamma}{\partial \rho_{des}} \delta \rho_{des} \right) d\Gamma \tag{4.4}$$

$$\delta R_u = -\nu \nabla^2 \delta \boldsymbol{u} - \nu \nabla \cdot \left(\nabla \delta \boldsymbol{u}^T \right) + \nabla \boldsymbol{u} \cdot \delta \boldsymbol{u} + \nabla \delta \boldsymbol{u} \cdot \boldsymbol{u} + (\nabla \cdot \boldsymbol{u}) \, \delta \boldsymbol{u} + (\nabla \cdot \delta \boldsymbol{u}) \, \boldsymbol{u} + \\ + \frac{\nabla \delta p}{\rho} + 2 \left(\boldsymbol{\omega} \times \delta \boldsymbol{u} \right) + \alpha \delta \boldsymbol{u} + \frac{\partial \alpha}{\partial \rho_{des}} \boldsymbol{u} \delta \rho_{des} \tag{4.5}$$

$$\delta R_c = -\nabla \cdot \delta \boldsymbol{u} \tag{4.6}$$

By using integration by parts and the scalar triple product (see App. A), the components in Eq. 4.3 can be grouped in terms with respect to $\delta \boldsymbol{u}$, δp and $\delta \rho_{des}$, so the variation of the Lagrangian can be written as:

$$\delta L = \int_\Omega \Bigg(- \nu \nabla \cdot \left(\nabla \boldsymbol{\psi_u} + \nabla \boldsymbol{\psi_u}^T \right) - \left(\nabla \boldsymbol{\psi_u} + \nabla \boldsymbol{\psi_u}^T \right) \cdot \boldsymbol{u} + \nabla \psi_c + 2 \left(\boldsymbol{\psi_u} \times \boldsymbol{\omega} \right) + \\ + \alpha \boldsymbol{\psi_u} + \frac{\partial J_\Omega}{\partial \boldsymbol{u}} \Bigg) \cdot \delta \boldsymbol{u} \, d\Omega + \\ + \int_\Omega \left(- \frac{\nabla \cdot \boldsymbol{\psi_u}}{\rho} + \frac{\partial J_\Omega}{\partial p} \right) \cdot \delta p \, d\Omega + \int_\Omega \left(\frac{\partial J_\Omega}{\partial \rho_{des}} + \frac{\partial \alpha}{\partial \rho_{des}} \boldsymbol{u} \cdot \boldsymbol{\psi_u} \right) \cdot \delta \rho_{des} \, d\Omega + \\ + \int_\Gamma \Bigg(\nu \delta \boldsymbol{u} \cdot \left(\nabla \boldsymbol{\psi_u} \cdot \hat{\boldsymbol{n}} \right) + \nu \left(\nabla \cdot \boldsymbol{\psi_u} \right) \left(\delta \boldsymbol{u} \cdot \hat{\boldsymbol{n}} \right) - \nu \boldsymbol{\psi_u} \cdot \left(\nabla \delta \boldsymbol{u} \cdot \hat{\boldsymbol{n}} \right) - \nu \left(\nabla \cdot \delta \boldsymbol{u} \right) \left(\boldsymbol{\psi_u} \cdot \hat{\boldsymbol{n}} \right) + \\ + \left(\boldsymbol{\psi_u} \cdot \delta \boldsymbol{u} \right) \left(\boldsymbol{u} \cdot \hat{\boldsymbol{n}} \right) + \left(\boldsymbol{u} \cdot \boldsymbol{\psi_u} \right) \left(\delta \boldsymbol{u} \cdot \hat{\boldsymbol{n}} \right) - \psi_c \left(\delta \boldsymbol{u} \cdot \hat{\boldsymbol{n}} \right) + \frac{\partial J_\Gamma}{\partial \boldsymbol{u}} \cdot \delta \boldsymbol{u} \Bigg) \, d\Gamma + \\ + \int_\Gamma \left(\frac{\boldsymbol{\psi_u} \cdot \hat{\boldsymbol{n}}}{\rho} + \frac{\partial J_\Gamma}{\partial p} \right) \cdot \delta p \, d\Gamma + \int_\Gamma \left(\frac{\partial J_\Gamma}{\partial \rho_{des}} + \frac{\partial \alpha}{\partial \rho_{des}} \boldsymbol{u} \cdot \boldsymbol{\psi_u} \right) \cdot \delta \rho_{des} \, d\Gamma \tag{4.7}$$

The adjoint equations and boundary conditions arise from vanishing the terms multiplied by $\delta \boldsymbol{u}$ and δp (OTHMER, 2008). Hence, the adjoint system of equations considering a generic objective function is given by:

$$- \nu \nabla \cdot \left(\nabla \boldsymbol{\psi_u} + \nabla \boldsymbol{\psi_u}^T \right) - \left(\nabla \boldsymbol{\psi_u} + \nabla \boldsymbol{\psi_u}^T \right) \cdot \boldsymbol{u} + \nabla \psi_c + 2 \left(\boldsymbol{\psi_u} \times \boldsymbol{\omega} \right) + \alpha \boldsymbol{\psi_u} + \\ + \frac{\partial J_\Omega}{\partial \boldsymbol{u}} = 0 \tag{4.8}$$

$$- \nabla \cdot \boldsymbol{\psi_u} + \frac{\partial J_\Omega}{\partial p} = 0 \tag{4.9}$$

with the following boundary conditions:

$$\nu\delta\boldsymbol{u}\cdot(\nabla\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}})+\nu\left(\nabla\cdot\boldsymbol{\psi_u}\right)(\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}})-\nu\boldsymbol{\psi_u}\cdot(\nabla\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}})-\nu\left(\nabla\cdot\delta\boldsymbol{u}\right)(\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}})+$$
$$+(\boldsymbol{\psi_u}\cdot\delta\boldsymbol{u})(\boldsymbol{u}\cdot\hat{\boldsymbol{n}})+(\boldsymbol{u}\cdot\boldsymbol{\psi_u})(\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}})-\psi_c(\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}})+\frac{\partial J_\Gamma}{\partial\boldsymbol{u}}\cdot\delta\boldsymbol{u}=0 \tag{4.10}$$

$$\frac{\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}}{\rho}+\frac{\partial J_\Gamma}{\partial p}=0 \tag{4.11}$$

The remaining terms from Eq. 4.7 are:

$$\delta L=\int_\Omega\left(\frac{\partial J_\Omega}{\partial\rho_{des}}+\frac{\partial\alpha}{\partial\rho_{des}}\boldsymbol{u}\cdot\boldsymbol{\psi_u}\right)\cdot\delta\rho_{des}\,d\Omega+\int_\Gamma\left(\frac{\partial J_\Gamma}{\partial\rho_{des}}+\frac{\partial\alpha}{\partial\rho_{des}}\boldsymbol{u}\cdot\boldsymbol{\psi_u}\right)\cdot\delta\rho_{des}\,d\Gamma \tag{4.12}$$

At the boundaries, ρ_{des} is constant, so $\delta\rho_{des}=0$ at Γ and the last term from Eq. 4.12 vanishes. Then, the variation of the Langragian with respect to the variation of the design variables (sensitivities for the optimization problem) can be calculated by the remaining term of Eq. 4.12 and is written as:

$$\frac{dL}{d\rho_{des}}=\int_\Omega\left(\frac{\partial J_\Omega}{\partial\rho_{des}}+\frac{\partial\alpha}{\partial\rho_{des}}\boldsymbol{u}\cdot\boldsymbol{\psi_u}\right)\,d\Omega \tag{4.13}$$

Hence, for a discretized domain, since the design variable $\rho_{des,i}$ only enters the primal equations in cell i, the contributions from other cells are left out of the integration and the sensitivity for cell i can be calculated by:

$$\frac{dL}{d\rho_{des,i}}=\left(\frac{\partial J_\Omega}{\partial\rho_{des,i}}+\frac{\partial\alpha}{\partial\rho_{des,i}}\boldsymbol{u_i}\cdot\boldsymbol{\psi_{u,i}}\right)V_i \tag{4.14}$$

where V_i is the cell volume.

Considering that the velocity is usually specified at the inlet and walls, $\boldsymbol{\delta u}=0$. For these regions, the necessary derivative of the objective function presented in Eq. 3.3 and the adjoint boundary conditions become:

$$\frac{\partial J_\Gamma}{\partial p}=\frac{\partial}{\partial p}\left(-\frac{1}{\rho}\left(p+\frac{\boldsymbol{u}\cdot\boldsymbol{u}}{2}\right)(\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\right)=-\frac{\boldsymbol{u}\cdot\hat{\boldsymbol{n}}}{\rho} \tag{4.15}$$

$$\frac{\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}}{\rho}+\frac{\partial J_\Gamma}{\partial p}=0\;\rightarrow\;\frac{\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}}{\rho}=\frac{\boldsymbol{u}\cdot\hat{\boldsymbol{n}}}{\rho}\;\rightarrow\;\boldsymbol{\psi_u}=\boldsymbol{u} \tag{4.16}$$

This derivation does not impose a condition for the Lagrange multiplier of the continuity equation ψ_c. As this term appears in the adjoint Navier-Stokes equations similarly as pressure does for the primal Navier-Stokes equations, at the inlet and the walls, a zero gradient ($\nabla\psi_c\cdot\hat{\boldsymbol{n}}=0$) boundary condition is assumed for this variable (OTHMER, 2008).

At the outlet, the necessary derivative of the objective function (Eq. 3.3) together

with the definition of the outlet pressure results in:

$$\frac{\partial J_\Gamma}{\partial \boldsymbol{u}} = -\left(\frac{p}{\rho} + \frac{\boldsymbol{u}\cdot\boldsymbol{u}}{2}\right)\cdot\hat{\boldsymbol{n}} - (\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\cdot\boldsymbol{u} = -\left(\frac{\boldsymbol{u}\cdot\boldsymbol{u}}{2}\right)\cdot\hat{\boldsymbol{n}} - (\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\cdot\boldsymbol{u} \tag{4.17}$$

Assuming that, at the outlet, the pressure and the gradient of the velocity (in normal direction) are zero, $\delta p = 0$ and $\nabla\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}} = 0$. From Eq. 4.9, it can be seen that $\nabla\cdot\boldsymbol{\psi_u} = 0$ for objective functions that depend only on boundary terms. From the continuity equation, it can be stated that $\nabla\cdot\delta\boldsymbol{u} = 0$, so Eq. 4.10 can be written as:

$$\left(\nu\left(\nabla\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}\right) + \left(\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\right)\boldsymbol{\psi_u} + \left(\boldsymbol{u}\cdot\boldsymbol{\psi_u}\right)\hat{\boldsymbol{n}} - \psi_c\hat{\boldsymbol{n}} + \frac{\partial J_\Gamma}{\partial \boldsymbol{u}}\right)\cdot\delta\boldsymbol{u} = 0 \tag{4.18}$$

Finally, for the outlet, the adjoint boundary conditions can be written by substituting Eq. 4.17 in Eq. 4.18:

$$\nu\left(\nabla\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}\right) + \left(\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\right)\boldsymbol{\psi_u} + \left(\boldsymbol{u}\cdot\boldsymbol{\psi_u}\right)\hat{\boldsymbol{n}} - \psi_c\hat{\boldsymbol{n}} - \left(\frac{\boldsymbol{u}\cdot\boldsymbol{u}}{2}\right)\cdot\hat{\boldsymbol{n}} - (\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\cdot\boldsymbol{u} = 0 \tag{4.19}$$

Thus, the final adjoint Navier-Stokes system is given by:

$$-\nu\nabla\cdot\left(\nabla\boldsymbol{\psi_u} + \nabla\boldsymbol{\psi_u}^T\right) - \left(\nabla\boldsymbol{\psi_u} + \nabla\boldsymbol{\psi_u}^T\right)\cdot\boldsymbol{u} + \nabla\psi_c + 2\left(\boldsymbol{\psi_u}\times\boldsymbol{\omega}\right) + \alpha\boldsymbol{\psi_u} = 0 \tag{4.20}$$

$$-\nabla\cdot\boldsymbol{\psi_u} = 0 \tag{4.21}$$

with the inlet conditions as (Eq. 4.16 and assumption for ψ_c):

$$\boldsymbol{\psi_u} = \boldsymbol{u_{in}} \tag{4.22}$$

$$\nabla\psi_c\cdot\hat{\boldsymbol{n}} = 0 \tag{4.23}$$

For the walls, the boundary conditions used are (Eq. 4.16 and assumption for ψ_c):

$$\boldsymbol{\psi_u} = 0 \tag{4.24}$$

$$\nabla\psi_c\cdot\hat{\boldsymbol{n}} = 0 \tag{4.25}$$

and, for the outlet, the boundary conditions are:

$$\nu\left(\nabla\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}\right) + \left(\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\right)\boldsymbol{\psi_u} + \left(\boldsymbol{u}\cdot\boldsymbol{\psi_u}\right)\hat{\boldsymbol{n}} - \psi_c\hat{\boldsymbol{n}} - \left(\frac{\boldsymbol{u}\cdot\boldsymbol{u}}{2}\right)\cdot\hat{\boldsymbol{n}} - (\boldsymbol{u}\cdot\hat{\boldsymbol{n}})\cdot\boldsymbol{u} = 0 \tag{4.26}$$

4.2 Discrete Adjoint Method using Finite Differences

The discrete approach is developed in this work to overcome the difficulties found in the derivation and implementation of the continuous adjoint for compressible flow cases.

The nonlinearities found in a compressible flow modeling make the hand derivation of the adjoint a challenging task and favour the use of the discrete approach. The initial point of the discrete approach is also the Lagrange multipliers method. This time, the definition is made by describing the system in a discretized manner. So, considering the Lagrange multiplier vector $\boldsymbol{\psi_{adj}}$, the Lagrangian function L can be written as:

$$L = J(\boldsymbol{w}(\boldsymbol{\rho_{des}}), \boldsymbol{\rho_{des}}) - \boldsymbol{\psi_{adj}^T} \boldsymbol{R}(\boldsymbol{w}(\boldsymbol{\rho_{des}}), \boldsymbol{\rho_{des}}) \tag{4.27}$$

where $\boldsymbol{w}$ is a vector with the state variables velocity, pressure and enthalpy (for compressible flows), $\boldsymbol{\rho_{des}}$ is a vector with the design variables and $\boldsymbol{R}$ represents the residual equations (equations for the physical system).

The derivative with respect to $\boldsymbol{\rho_{des}}$ is calculated by:

$$\begin{aligned} \frac{dL}{d\boldsymbol{\rho_{des}}} &= \frac{\partial J}{\partial \boldsymbol{\rho_{des}}} + \frac{\partial J}{\partial \boldsymbol{w}} \frac{d\boldsymbol{w}}{d\boldsymbol{\rho_{des}}} - \boldsymbol{\psi_{adj}^T} \left(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{\rho_{des}}} + \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{w}} \frac{d\boldsymbol{w}}{d\boldsymbol{\rho_{des}}} \right) = \\ &= \frac{\partial J}{\partial \boldsymbol{\rho_{des}}} - \boldsymbol{\psi_{adj}^T} \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{\rho_{des}}} + \left(\frac{\partial J}{\partial \boldsymbol{w}} - \boldsymbol{\psi_{adj}^T} \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{w}} \right) \frac{d\boldsymbol{w}}{d\boldsymbol{\rho_{des}}} \end{aligned} \tag{4.28}$$

The term $d\boldsymbol{w}/d\boldsymbol{\rho_{des}}$ is computationally very expensive to calculate, so the expression multiplied by this term in Eq. 4.28 can be set to zero by finding the correct Lagrange multipliers $\boldsymbol{\psi_{adj}}$. This constitutes the adjoint problem:

$$\frac{\partial \boldsymbol{R}^T}{\partial \boldsymbol{w}} \boldsymbol{\psi_{adj}} = \frac{\partial J}{\partial \boldsymbol{w}} \tag{4.29}$$

After solving the system defined by Eq. 4.29, the sensitivities for the optimization problem can be calculated by:

$$\frac{dL}{d\boldsymbol{\rho_{des}}} = \frac{\partial J}{\partial \boldsymbol{\rho_{des}}} - \boldsymbol{\psi_{adj}^T} \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{\rho_{des}}} \tag{4.30}$$

The matrices and vectors necessary to the sensitivity calculation ($\partial \boldsymbol{R}/\partial \boldsymbol{w}$, $\partial J/\partial \boldsymbol{w}$, $\partial \boldsymbol{R}/\partial \boldsymbol{\rho_{des}}$, and $\partial \boldsymbol{J}/\partial \boldsymbol{\rho_{des}}$ in Eqs. 4.29 and 4.30) are constructed by estimating its components by finite differences.

In a finite volume implementation, the discretization of the governing equations for a cell P can be written in a general residual form as (MOUKALLED; MANGANI; DARWISH, 2016b):

$$R_{\phi,P} = a_P \phi_P + \sum_N (a_N \phi_N) - b_P \approx 0 \tag{4.31}$$

where ϕ can be any scalar variable, a represents the coefficients of the discretized equation,

N indicates contributions from neighbour cells and b_P represents any source terms. Hence, the partial derivatives of the residual equations with respect to each state variable can be calculated by making a small perturbation with size δ_{adj} in the state variable, calculating the difference with the original residual and dividing it by δ_{adj}. For the residual $R_{\phi,P}$ of cell P, the partial derivative with respect to ϕ_P calculated by forward finite differences is:

$$\frac{\partial R_{\phi,P}}{\partial \phi_P} \approx \frac{R_{\phi,P}(\phi_P + \delta_{adj}) - R_{\phi,P}(\phi_P)}{\delta_{adj}} \tag{4.32}$$

The state variables are the velocity components in the three directions u_1, u_2 and u_3, the pressure p and the enthalpy h (if the flow is compressible). The vector $\boldsymbol{w}$ can be $\boldsymbol{w} = [u_1,\ u_2,\ u_3,\ p]$ or $\boldsymbol{w} = [u_1,\ u_2,\ u_3,\ p,\ h]$. The solution procedure used performs a linearization of the convective term (in Eqs. 2.6 and 2.14) by using the fluxes ϕ^f calculated at the faces f. For each cell P, this term is written (for incompressible and compressible flows) as:

$$\int_P \nabla \cdot (\boldsymbol{u} \otimes \boldsymbol{u})\, dV = \int_S (\boldsymbol{u} \otimes \boldsymbol{u}) \cdot d\boldsymbol{S} = \sum_f (\boldsymbol{u}_f \otimes \boldsymbol{u}_f) \cdot \boldsymbol{S}_f = \sum_f \phi^f \boldsymbol{u}_f \tag{4.33}$$

$$\int_P \nabla \cdot (\rho\boldsymbol{u} \otimes \boldsymbol{u})\, dV = \int_S (\rho\boldsymbol{u} \otimes \boldsymbol{u}) \cdot d\boldsymbol{S} = \sum_f (\rho\boldsymbol{u}_f \otimes \boldsymbol{u}_f) \cdot \boldsymbol{S}_f = \sum_f \rho\phi^f \boldsymbol{u}_f \tag{4.34}$$

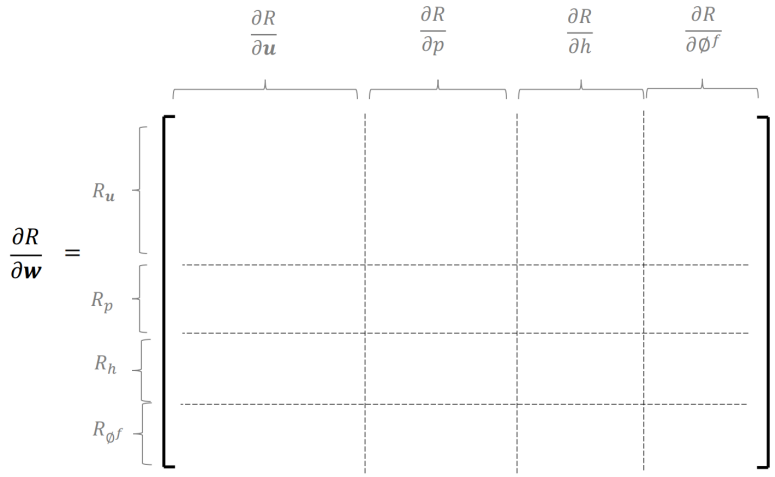
where $\boldsymbol{S}$ represents a surface vector.

During the flow calculation, the values used for ϕ^f are obtained from the previous iteration. This strategy complicates the dependency among ϕ^f and the other state variables, making it is more straightforward to also consider ϕ^f as an independent variable (ROTH; ULBRICH, 2013; HE et al., 2018). Therefore, for an incompressible flow, the state variable vector becomes $\boldsymbol{w} = [u_1,\ u_2,\ u_3,\ p,\ \phi^f]$ and, for a compressible flow, it is $\boldsymbol{w} = [u_1,\ u_2,\ u_3,\ p,\ h,\ \phi^f]$. The residuals assessed for each cell are $R = [R_{u_1},\ R_{u_2},\ R_{u_3},\ R_p,\ R_{\phi^f}]$ and $R = [R_{u_1},\ R_{u_2},\ R_{u_3},\ R_p,\ R_h,\ R_{\phi^f}]$, respectively. As an example, in Fig. 4.1, the structure of the matrix $\partial\boldsymbol{R}/\partial\boldsymbol{w}$ for a compressible flow case is shown . The columns show the perturbed state variables while the rows present the perturbed residuals. When only incompressible flows are considered, the matrix is assembled without the set of rows and columns referring to the enthalpy and to the energy equation residual.

The state variables have different orders of magnitude (pressure may be in the order of 10^6 while face flux can be around 10^{-4}), hence a scaling in the perturbation is applied. In the numerical cases, the state variables are perturbed by $C_{scale}\delta_{adj}$, where C_{scale} is a power of ten chosen based on the state variable being perturbed. Also, choosing a value for δ_{adj} is part of the task. In Sec. 4.3, sensitivity checks for different magnitudes of δ_{adj} are presented, showing that, even though this is not an obvious choice, there is a wide

range of values where the calculations work properly.

Figure 4.1 – Jacobian matrix assembly for compressible flows.



4.2.1 Grouping Algorithm

The perturbation process can be very slow and become prohibitive if state variables are perturbed one at a time. Thus, a strategy to accelerate the process is used. It consists in assembling groups of cells and faces at the beginning of the optimization or every time the mesh is changed. The cells and faces contained in one group are perturbed at the same time. In Fig. 4.2, an example of a mesh can be seen. The highlighted region in Fig. 4.2a shows a cell with a set of neighbours that are not allowed to be in the same group. By doing this, there is no residual being perturbed by two different state variables, and the partial derivatives are calculated correctly. This same observation holds for the faces. In the same figure, the faces belonging to the highlighted region are not allowed to be on the same group, guaranteeing that their residuals are not perturbed by more than one state variable. At the beginning of the optimization (or when the mesh is changed), a list is created relating each cell to the IDs of the "forbidden neighbours". Then, the grouping algorithm runs over the whole mesh by iterating over the cell IDs and does not place cells from the same forbidden region in the same group. This way, the groups only contain cells that are far enough from each other.

In Fig. 4.2b, the group that contains the highlighted cell from Fig. 4.2a can be seen. The cells sharing one face with a cell of interest (for example, the cell marked by a red dot) are considered level 1 neighbours. Cells sharing faces with level 1 neighbours are considered level 2 neighbours, and so on. Thus, the assembly of groups is performed by avoiding cells and faces up to a specified level to be on the same group. After the perturbation of a group is executed, the residuals are recalculated and collected only in a few cells and faces around the cell/face being perturbed. The partial derivatives are calculated and placed in the matrices by mapping their position with respect to cells/faces IDs. The grouping algorithm works by the following steps:

a) Create list of cells and faces with respective neighbours IDs (up to a specified level);

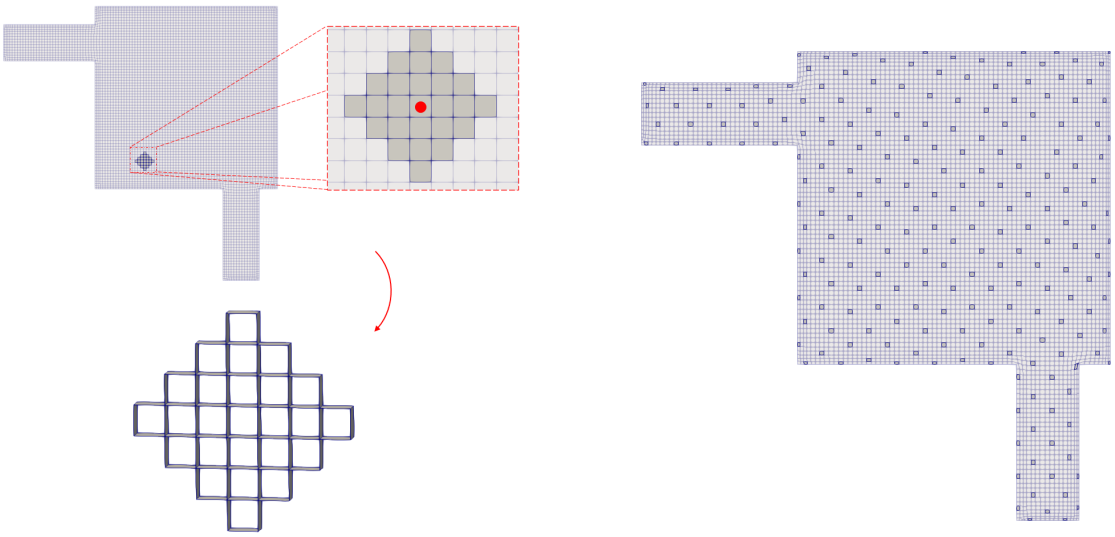
b) Iterate over all cells IDs. If the ID is on the neighbour list, go to the next ID. If the ID is not on the neighbour list, add it to the current group and start checking the new cell's neighbours also;

c) After the iteration over all cells is finished, go back to the lower ID of the remaining cells and repeat the process from step b);

d) Do the same for faces.

Figure 4.2 – Cell grouping.

(a) Cell with forbidden neighbours. **(b)** Example of cells from the same group.

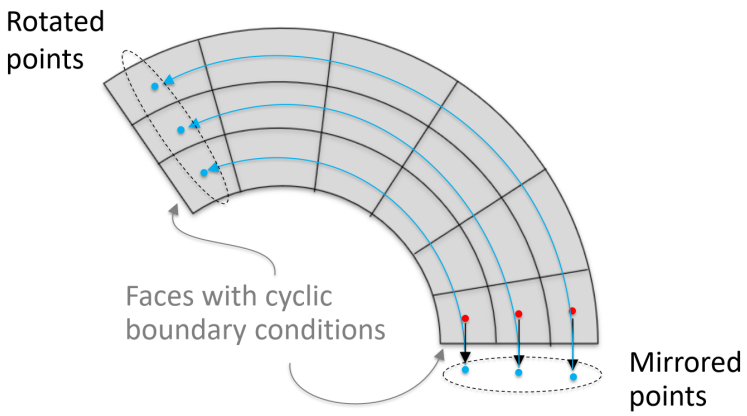


4.2.2 Cyclic Boundaries

Rotating domains usually have some sort of cylindrical shape and are commonly assumed to have rotational periodicity. In these cases, only one sector is modelled and cyclic boundary conditions can be used. The perturbation of a cell close to the cyclic boundary affects the residuals of the cells close to the opposite boundary. To avoid the perturbation of residuals by more than one state variable, the cells adjacent to the cyclic faces are listed as connected at the beginning of the creation of the groups. The centers of the cells adjacent to the cyclic boundary are mirrored with respect to the cyclic face and rotated by an angle equal to the sector being represented. They "fall" inside the cells adjacent to the opposite cyclic face. The cells where they fell into are identified and listed as neighbours of the original cells (in item "a" of the previously mentioned steps). Thus,

the cyclic faces are listed as connected and the grouping algorithm considers that the mesh is united in these faces. This procedure is illustrated in Fig. 4.3. To avoid duplicate neighbours and correctly execute the procedure, the discretization used for the cyclic faces is identical.

Figure 4.3 – Connection of cyclic faces.



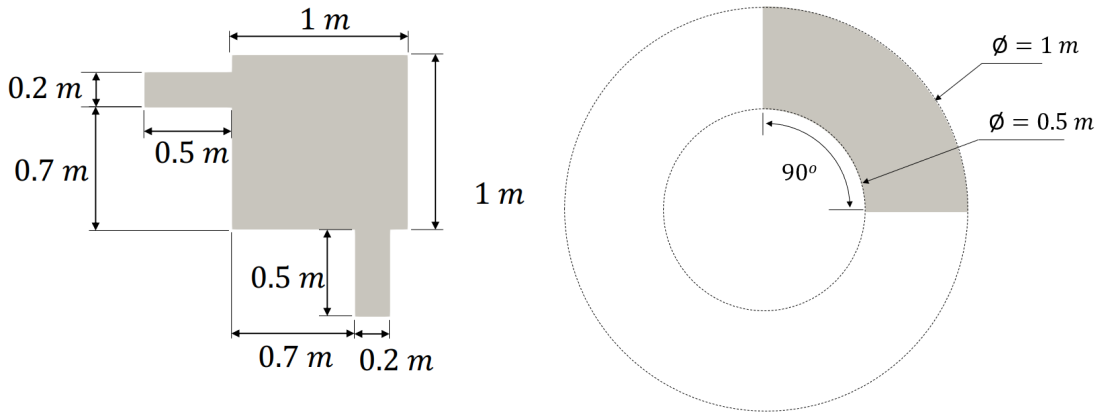
4.3 Sensitivity Verification

In order to verify the sensitivities calculated by the methodology described, numerical examples are defined for each of the adjoint approaches with several flow situations. In all cases, the same idea is used: a random geometry is created with regions where the design variables define solid (black), fluid (white) and intermediate material (grey) and ten points are randomly chosen to perform a comparison for the sensitivities. Two points are chosen in the solid region, two points in the intermediate region and the other six in the fluid region. The flows are calculated until the residuals for all state variables are below 10^{-5} and the objective functions exhibit a steady behavior. The adjoint system is solved until the convergence error is below 10^{-5}. The verification is performed by comparing the sensitivities calculated (for cell i) by the adjoint method $(dJ/d\rho_{des,i})_{adj}$ with the sensitivities calculated by backward finite differences:

$$\left(\frac{dJ}{d\rho_{des,i}}\right)_{FD} \approx \frac{J(\boldsymbol{w}(\rho_{des,i}), \rho_{des,i}) - J(\boldsymbol{w}(\rho_{des,i} - \delta_{FD}), \rho_{des,i} - \delta_{FD})}{\delta_{FD}} \tag{4.35}$$

For all the examples, the geometries used are the same and can be seen in Fig. 4.4, for the non-rotating and rotating cases.

Figure 4.4 – Geometries and dimensions for the sensitivity verification cases.
(a) Dimensions of the non-rotating domain. **(b)** Dimensions of the rotating domain.



For the continuous adjoint approach, the following cases are defined:

a) Incompressible laminar non-rotating flow;

b) Incompressible laminar rotating flow.

and, for the discrete adjoint approach, the cases are:

a) Incompressible laminar non-rotating flow;

b) Incompressible laminar rotating flow;

c) Compressible laminar non-rotating flow;

d) Compressible laminar rotating flow;

e) Compressible turbulent non-rotating flow (frozen turbulence);

f) Compressible turbulent rotating flow (frozen turbulence).

In all laminar cases, the value used for δ_{FD} is 10^{-5}. In the turbulent cases, $\delta_{FD} = 10^{-3}$. They are obtained by testing different values for δ_{FD} and choosing the smallest values that are far from roundoff errors but big enough to make a small change in the flow.

4.3.1 Continuous Adjoint - Incompressible Laminar Flows

In Fig. 4.5, the boundary definitions for the incompressible laminar cases (non-rotating and rotating) can be seen. The inlet velocity for the rotating case is defined in terms of a radial and a tangential components (see Fig. 4.5c). The viscosity and values for boundary conditions are presented in Tab. 4.1.

Figure 4.5 – Boundary condition definitions for the incompressible laminar cases.

(a) BCs for the incompressible non-rotating case. **(b)** BCs for the incompressible rotating case. **(c)** Radial and tangential directions.

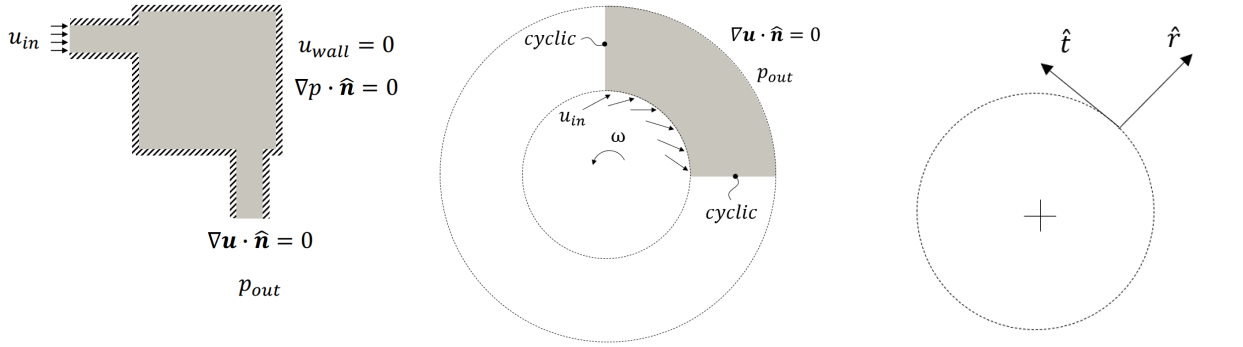


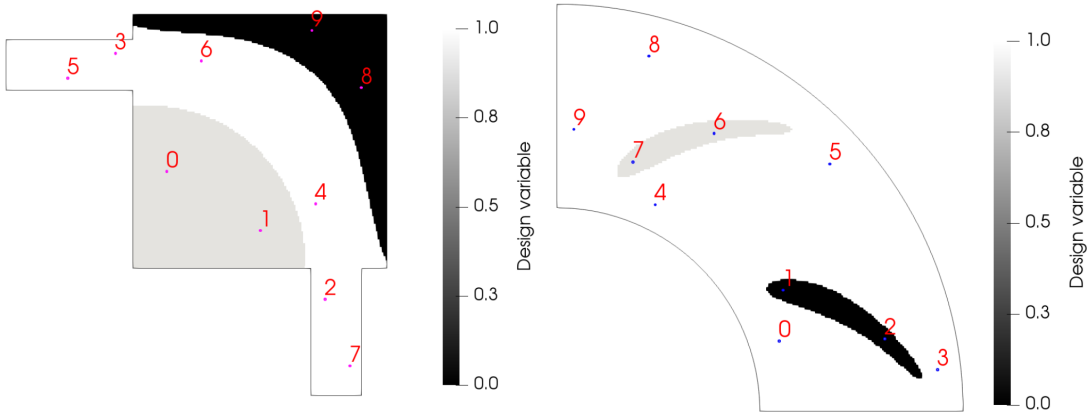
Table 4.1 – BCs for the incompressible laminar cases.

	Non-rotating	Rotating
$\nu\ (m^2/s)$	1.0	0.1
$u_{in}\ (m/s)$	1	$10\hat{\boldsymbol{r}} - 52.4\hat{\boldsymbol{t}}$
$p_{out}\ (Pa)$	0	0
Rotation (rpm)	0	1000

The solution of the direct and adjoint problems are conducted as explained in Sections 5.1 and 5.2, respectively. For both cases (non-rotating and rotating), the objective function used is presented in Eq. 3.3. The cells chosen to perform the sensitivity verification are presented in Fig. 4.6.

Figure 4.6 – Cells for the sensitivity verification.

(a) Non-rotating. **(b)** Rotating.

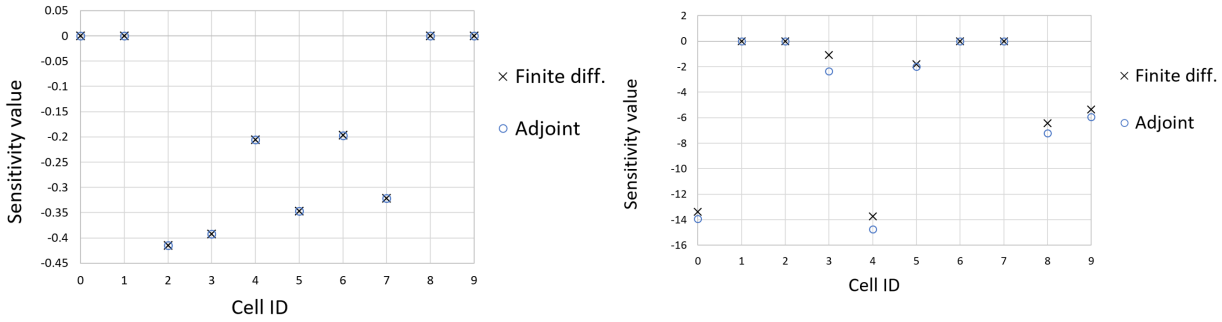


The values of the sensitivities calculated for the set of chosen cells can be seen in Fig. 4.7. The 'X' marks the sensitivities directly calculated by finite differences and the 'O'

marks the sensitivities calculated by the continuous adjoint approach.

Figure 4.7 – Sensitivity verification for incompressible non-rotating and rotating flows (continuous adjoint).

(a) Incompressible non-rotating. **(b)** Incompressible rotating.



The differences observed for the non-rotating case are smaller than 0.5% (with the finite difference value as a reference) and, for the rotating case, up to 7.5% for the highest value. As can be seen, the good agreement indicates that the sensitivity calculation is correctly implemented for the continuous adjoint approach.

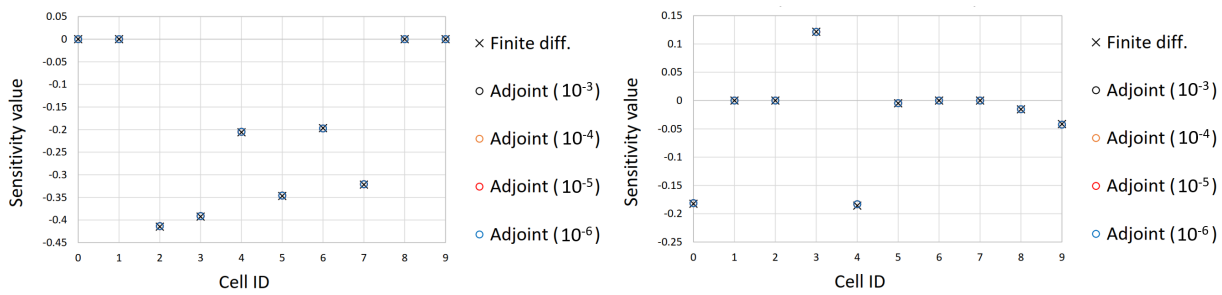
4.3.2 Discrete Adjoint - Incompressible Laminar Flows

In this case, the same geometries, boundary conditions and points are used (see Figs. 4.4 to 4.6 and Tab. 4.1). The solution of the direct problem is again as explained in Section 5.1, however the solution of the adjoint problem is as explained in Sec. 5.3 (it is also used for the other examples of the discrete adjoint). For the non-rotating case, the objective function used is the energy dissipation presented in Eq. 3.3 and, for the rotating case, the pump efficiency presented in Eq. 3.4.

In this adjoint approach, finite differences are also used to assemble the adjoint system, hence the value of the step size δ_{adj} must also be chosen. In Fig. 4.8, the results from the finite differences estimates can be seen together with the results from the discrete adjoint with several step sizes δ_{adj} (between parenthesis).

Figure 4.8 – Sensitivity verification for incompressible laminar non-rotating and rotating flows (discrete adjoint).

(a) Incompressible non-rotating. **(b)** Incompressible rotating.



In both graphs, the sensitivities calculated are in good agreement for a wide range of steps used to perturb the primal system. The differences observed for highest values of all cases are smaller than 1.3% (with the finite difference value as a reference). Thus, the good agreement indicates that the sensitivity calculation is correctly implemented for the discrete adjoint for incompressible laminar flows.

4.3.3 Discrete Adjoint - Compressible Laminar Flows

For the compressible flows without turbulence, the geometry and points from Figs. 4.4 and 4.6 are still used. The boundary conditions now have definitions for the energy equation, as presented in Fig. 4.9. Again, for the rotating case, the inlet velocity is defined in terms of a radial and a tangential components (see Fig. 4.5c) The flow properties and boundary condition values used are shown in Tab. 4.2

Figure 4.9 – Boundary condition definitions for the compressible cases.
(a) BCs for the compressible non-rotating case. **(b)** BCs for the compressible rotating case.

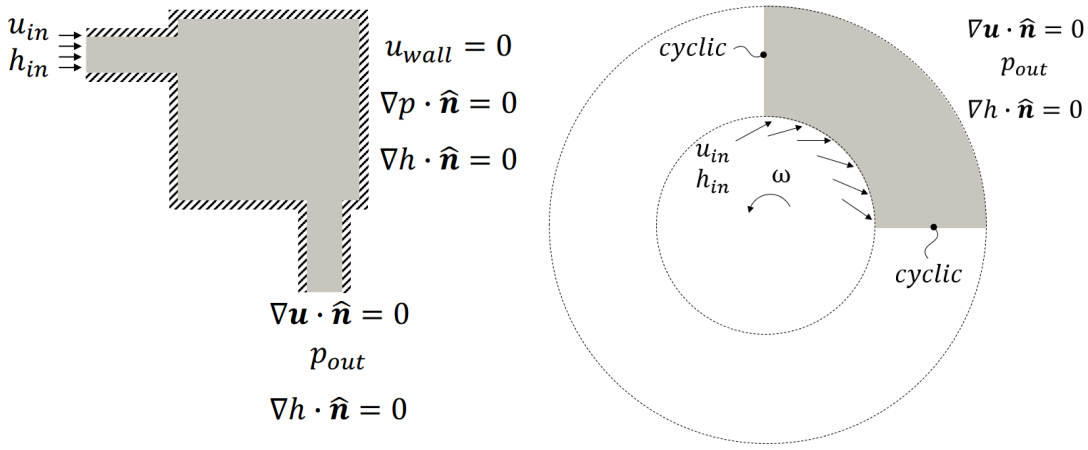


Table 4.2 – BCs for the compressible laminar cases.

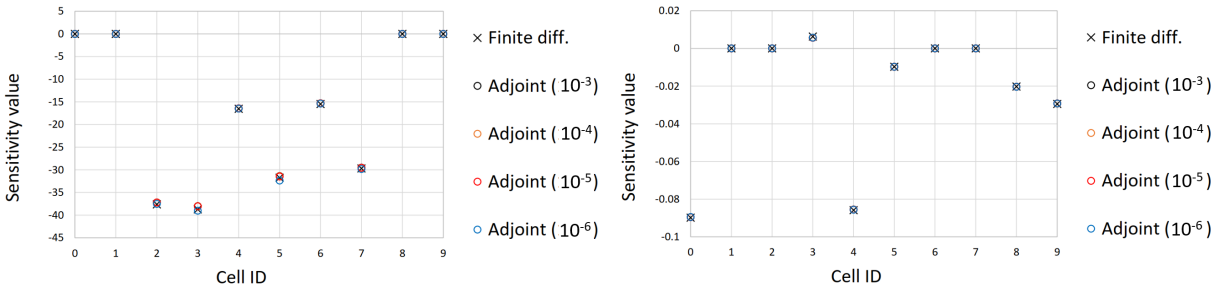
	Non-rotating	Rotating
$\mu\ (Pa \cdot s)$	1.0	0.5
$u_{in}\ (m/s)$	102	$100\hat{\boldsymbol{r}} - 261.8\hat{\boldsymbol{t}}$
$p_{out}\ (Pa)$	101325	101325
$h_{in}\ (J/kg)$	$2.95 \cdot 10^5$	$2.95 \cdot 10^5$
$c_p\ (J/(kg \cdot K))$	1005	1005
$Pr\ (Pa)$	0.705	0.705
Rotation (rpm)	0	5000

Now, the objective functions used are the entropy variation from Eq. 3.9 in the non-rotating case and the isentropic efficiency from Eq. 3.10 in the rotating case. As in the last example, the value of the step size δ_{adj} must also be chosen. In Fig. 4.10, the results

for the compressible flows without turbulence models are presented together with several the step sizes δ_{adj} between parenthesis.

Figure 4.10 – Sensitivity verification for compressible laminar non-rotating and rotating flows (discrete adjoint).

(a) Compressible non-rotating. **(b)** Compressible rotating.



The sensitivities calculated are in good agreement for the range of steps used to perturb the primal system. The differences observed for highest sensitivity values of all cases are smaller than 2.0% (with the finite difference value as a reference). Thus, the good agreement indicates that the sensitivity calculation is correctly implemented for the discrete adjoint for compressible laminar flows.

4.3.4 Discrete Adjoint - Compressible Turbulent Flows

In the compressible turbulent flow cases, the turbulence model from Sec. 2.2 is used. The geometry for the non-rotating case is changed to have a well behaved flow. The geometry for the rotating case is kept, since a good convergence is obtained by only changing the boundary condition values. The new geometry for the non-rotating case with the points used is presented in Fig. 4.11. The values for boundary conditions for both cases can be seen in Tab. 4.3.

Figure 4.11 – Cells for sensitivity verification in new non-rotating model.

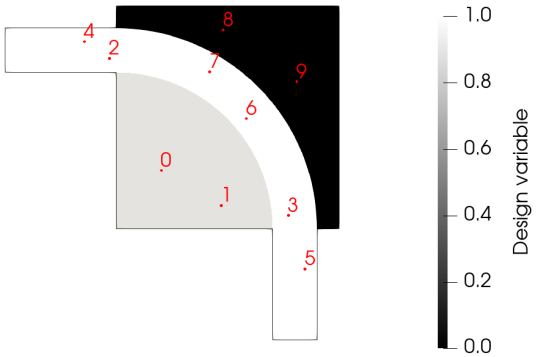


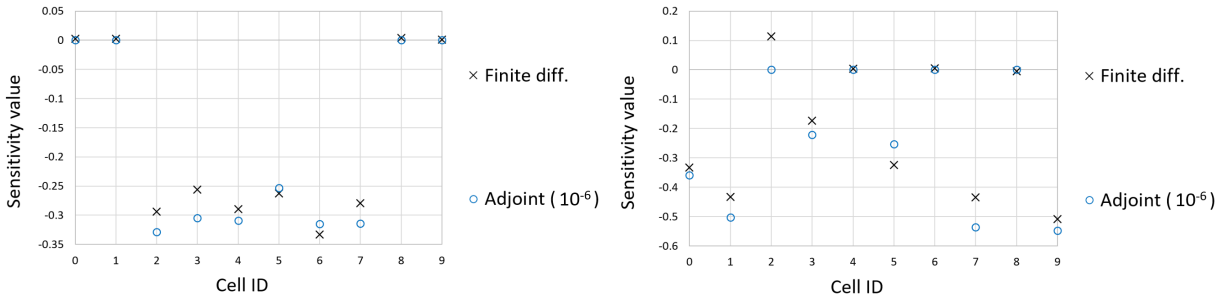
Table 4.3 – BCs for the compressible laminar cases.

	Non-rotating	Rotating
$\mu\ (Pa \cdot s)$	1.8×10^{-5}	1.8×10^{-5}
$u_{in}\ (m/s)$	102	$100\hat{\boldsymbol{r}} - 130.9\hat{\boldsymbol{t}}$
$p_{out}\ (Pa)$	101325	101325
$h_{in}\ (J/kg)$	$2.95 \cdot 10^5$	$2.95 \cdot 10^5$
$c_p\ (J/(kg \cdot K))$	1005	1005
$Pr\ (Pa)$	0.705	0.705
Rotation (rpm)	0	2500

The objective function used is the entropy variation (Eq. 3.9) for the non-rotating case and the isentropic efficiency (Eq. 3.10) for the rotating case. The step size used is $\delta_{adj} = 10^{-6}$. In Fig. 4.12, the results from the finite differences estimates can be seen together with the results from the discrete adjoint.

Figure 4.12 – Sensitivity verification for compressible turbulent non-rotating and rotating flows (discrete adjoint).

(a) Compressible non-rotating. **(b)** Compressible rotating.



In both graphs, the sensitivities calculated show the same tendency. The differences observed for highest values of all cases can reach 28.0% (with the finite difference value as a reference). This difference comes mainly from the frozen turbulence hypothesis. Even not being as accurate as the sensitivities for the laminar cases, the values indicate the correct direction of the sensitivities and can be used in the optimization.

5 NUMERICAL IMPLEMENTATION

The work is developed in the finite volume library *OpenFOAM* version 2006 (ESI, 2020a). It is an open-source, *C++* and object-oriented code, designed to be applied to continuum mechanics problems. The library comes with several applications coded to handle different problems (incompressible, compressible, multi-phase, non-Newtonian, structural, heat transfer, etc.) and a broad range of boundary conditions ready to be used. Over the years, it has been mainly used on fluid mechanics applications. It allows programmers and users to build their own classes and applications by simply calling specific objects and customizing existing solvers. When implementing or modifying a solver, the main idea is to see the partial differential equations as a sum of single differential operators that can be discretized separately and with different discretization schemes (MANGANI; CASARTELLI; MAURI, 2012). Therefore, the implementation of different types of equations is performed by the coding of a set of differential operators acting on field variables. A language similar to the mathematical definition is used to facilitate code comprehension. As an example, the momentum equation for an incompressible flow (with p being pressure divided by specific mass ρ):

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot (\boldsymbol{u} \otimes \boldsymbol{u}) - \nabla \cdot (\nu \nabla \boldsymbol{u}) = -\nabla p \tag{5.1}$$

can be solved with the following piece of code:

```
 solve
(
   fvm::ddt(U)
   + fvm::div(phi,U)
   - fvm::laplacian(nu,U)
   ==
   -fvc::grad(p)
)
```

All differential operators are treated like finite volume calculus (`fvc`) or finite volume method (`fvm`) operators. Explicit derivatives are calculated by the former, while, for the later, an implicit derivation converts the expression into matrix coefficients. Three dimensional polyhedral meshes can be handled by *OpenFOAM*, making it versatile and applicable to problems going from simple and well behaved cases to complex and large-scale problems.

For the continuous adjoint approach, all the infrastructure of *OpenFOAM* is used. The

existing solver called *adjointShapeOptimizationFoam* (ESI, 2022a) is modified to account for rotation and to be able to apply the material model and the volume constraint.

For the discrete adjoint, the flow calculations are performed by the solvers *simpleFoam* (ESI, 2022d) and *rhoSimpleFoam* (ESI, 2022c) for the incompressible and compressible flows, respectively. The existing codes for these solvers are modified to include the parts related to topology optimization developed in this work. All the algebraic manipulations, except for the solution of the adjoint system, are executed by the library *Eigen*, version 3.3.9 (EIGEN, 2020). It is a fast and lightweight library for linear algebra, written in *C++*. It is open-source and can be easily compiled with *OpenFOAM* codes.

The solution of the discrete adjoint system is calculated with the library *PETSc*, version 3.15 (PETSC, 2021). *PETSc* stands for "Portable, Extensible Toolkit for Scientific Computation". It is developed for use in large-scale application projects. A large suite of parallel linear, nonlinear equation solvers and preconditioners are available. The libraries can be found in *C, C++, Fortran* and *Python*.

In short, the structure developed in this work is composed by two main codes: *PETSc* in its default implementation and *OpenFOAM*, modified with the perturbations described in Chap. 4 and compiled with *Eigen* to perform the algebraic manipulations. The communication between *PETSc* and the modified solvers from *OpenFOAM* is performed via writing and reading of files. The solver starts, calculates the flow until some stopping criteria, assembles the adjoint system, writes it in a binary file, call *PETSc* and waits. *PETSc* reads the file, solves the system, writes the solution and close, letting the modified solver from *OpenFOAM* read the adjoint solution and continue with the optimization. With several sequences of these steps, the topology optimization is applied.

5.1 Solution of the Direct Problem

The solution of the physical problem is performed with the implementations of the SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) algorithm (PATANKAR, 1980) available in *OpenFOAM*. The algorithm is pressure based and works in a segregated way, by solving the pressure and velocity separately and then making corrections. While the velocity is calculated by solving the momentum equations, the pressure is calculated by an equation resulting from the combination of momentum and continuity equations. If the flow is compressible, the temperature is calculated from the energy equation. A detailed description of the algorithms for incompressible and compressible flows can be seen in Jasak (1996) and Jasak (2006). A short explanation is presented next.

The momentum equation for a cell with center P can be written in semi-discretized

form as:

$$a_P^{\boldsymbol{u}} \boldsymbol{u}_P + \sum_N a_N^{\boldsymbol{u}} \boldsymbol{u}_N = \boldsymbol{r} - \nabla p \tag{5.2}$$

where $\boldsymbol{u}_P$ is the velocity in a cell of interest, $\boldsymbol{u}_N$ are the velocities in the neighbor cells and $a_i^{\boldsymbol{u}}$ are the coefficients from the discretization.

Defining an $\boldsymbol{H(u)}$ operator, containing the off-diagonal part of the momentum matrix and any associated r.h.s. contributions:

$$\boldsymbol{H(u)} = \boldsymbol{r} - \sum_N a_N^{\boldsymbol{u}} \boldsymbol{u}_N \tag{5.3}$$

Velocity can be calculated by:

$$\boldsymbol{u}_P = (a_P^{\boldsymbol{u}})^{-1} \boldsymbol{H(u)} - (a_P^{\boldsymbol{u}})^{-1} \nabla p \tag{5.4}$$

By substituting Eq. 5.4 in the incompressible continuity equation, an equation to calculate the pressure in incompressible flows can be derived:

$$\nabla \cdot \boldsymbol{u} = \nabla \cdot \Big((a_P^{\boldsymbol{u}})^{-1} \boldsymbol{H(u)} \Big) - \nabla \cdot \Big((a_P^{\boldsymbol{u}})^{-1} \nabla p \Big) = 0 \tag{5.5}$$

During the calculation a linearization in the convective term is performed as described by Eqs. 4.33 and 4.34, so the face flux ϕ^f must be updated with the new pressure field. By opening the continuity equation and using Eq. 5.4, ϕ^f can be written as:

$$\nabla \cdot \boldsymbol{u} = \sum_f \boldsymbol{S}_f \cdot \boldsymbol{u}_f = \sum \phi^f \tag{5.6}$$

$$\phi^f = \boldsymbol{S}_f \cdot \Big[(a_P^{\boldsymbol{u}})^{-1} \boldsymbol{H(u)} - (a_P^{\boldsymbol{u}})^{-1} \nabla p \Big]_f \tag{5.7}$$

In order to avoid divergence during the calculation, under relaxation is applied to all equations of the system. Also, for turbulent flows, the turbulence equations are solved inside the same loop, but after the velocity and pressure are calculated.

In *OpenFOAM* v2006, the pressure-based solver for incompressible flows works by the following steps:

a) Initialize $\boldsymbol{u}$ and p using latest available values;

b) Assemble the momentum equation (Eq. 5.4);

c) Under-relax the momentum equations;

d) Solve the momentum equation to obtain a prediction for $\boldsymbol{u}$ (momentum predictor step);

e) Assemble the pressure equation (Eq. 5.5) with the calculated velocity;

f) Calculate the new pressure (pressure correction step);

g) Based on pressure solution, assemble face flux (Eq. 5.7);

h) Under-relax the calculated pressure field;

i) Correct the velocity field using the new pressure field (use Eq. 5.4);

j) Solve turbulence equation(s);

k) Repeat until reach convergence.

In the case of compressible flows, the same idea is used, however the loop involves more steps due to the presence of the energy equation and an equation of state. A different equation for the pressure calculation arises from the compressible continuity equation. Defining the compressibility ψ from the ideal gas law as:

$$\rho = \frac{p}{RT} = \psi p \tag{5.8}$$

The compressible continuity can be written as:

$$\nabla \cdot (\rho \boldsymbol{u}) = \nabla \cdot \left(\rho (a_P^{\boldsymbol{u}})^{-1} \boldsymbol{H}(\boldsymbol{u})\right) - \nabla \cdot \left(\rho (a_P^{\boldsymbol{u}})^{-1} \nabla p\right) = 0 \tag{5.9}$$

So, with Eq. 5.8, the pressure equation is:

$$\nabla \cdot \left(\psi (a_P^{\boldsymbol{u}})^{-1} \boldsymbol{H}(\boldsymbol{u}) p\right) - \nabla \cdot \left(\rho (a_P^{\boldsymbol{u}})^{-1} \nabla p\right) = 0 \tag{5.10}$$

And the flux at the faces can be calculated by:

$$\nabla \cdot (\rho \boldsymbol{u}) = \sum_f \boldsymbol{S}_f \cdot \rho \boldsymbol{u}_f = \sum \phi^f \tag{5.11}$$

$$\phi^f = \boldsymbol{S}_f \cdot \left[\psi (a_P^{\boldsymbol{u}})^{-1} \boldsymbol{H}(\boldsymbol{u}) p - \rho (a_P^{\boldsymbol{u}})^{-1} \nabla p\right]_f \tag{5.12}$$

For the velocity calculation, Eq. 5.4 is used.

In *OpenFOAM* v2006, the pressure-based solver for compressible flows works by the steps described as follows:

a) Initialize $\boldsymbol{u}$, p and T using latest available values;

b) Assemble the momentum equation (Eq. 5.4);

c) Under-relax the momentum equations;

d) Solve the momentum equation to obtain a prediction for $\boldsymbol{u}$ (momentum predictor step);

e) Assemble, under-relax and solve the energy equation;

f) Calculate density field from state equation;

g) Assemble the pressure equation (Eq. 5.10) with the calculated velocity and density;

h) Calculate the new pressure (pressure correction step);

i) Based on pressure solution, assemble face flux (Eq. 5.12);

j) Under-relax the calculated pressure field;

k) Correct the velocity field using the new pressure field (use Equation 5.4);

l) Solve turbulence equation(s);

m) Repeat until reach convergence.

During the calculation of the steps described for incompressible and compressible flows, it is necessary to solve linear systems several times. In *OpenFOAM*, different types of linear solvers are available (ESI, 2020b):

a) Diagonal: diagonal solver for explicit systems.

b) PCG and PBiCG: preconditioned (bi-)conjugate gradient solver;

c) PBiCGStab: stabilized Preconditioned bi-conjugate gradient solver;

d) GAMG: generalised geometric algebraic multigrid. The principle is to generate a quick solution on a mesh with a small number of cells, map this solution onto a finer mesh, use it as an initial guess to obtain a more accurate solution on the fine mesh.

For the conjugate gradient solvers, a number of preconditioners can be found (ESI, 2020b):

a) Diagonal preconditioner;

b) DIC: diagonal incomplete-Cholesky (symmetric);

c) FDIC: a faster version of the DIC preconditioner for symmetric matrices, in which the reciprocal of the preconditioned diagonal and the upper coefficients divided by the diagonal are calculated and stored (MOUKALLED; MANGANI; DARWISH, 2016a);

d) DILU: diagonal incomplete-LU (asymmetric);

e) GAMG: a geometric algebraic multigrid preconditioner. It uses a mutigrid cycle as preconditioner (MOUKALLED; MANGANI; DARWISH, 2016a).

Also, smoothers can be used (ESI, 2020b):

a) Gauss-Seidel: Gauss-Seidel method for both symmetric and asymmetric matrices;

b) DIC: a diagonal-based incomplete Cholesky smoother;

c) DICGaussSeidel: a combined DIC/Gauss-Seidel smoother in which DIC smoothing is followed by Gauss-Seidel to smooth out "spikes" that may be created by the DIC sweeps (MOUKALLED; MANGANI; DARWISH, 2016a).

In this work, GAMG solver is used for the pressure equation and smoothers with the Gauss-Seidel method for the momentum equations.

5.2 Solution of the Continuous Adjoint Problem

The solution of the adjoint problem in the continuous approach is performed with an adaptation of the SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) algorithm (PATANKAR, 1980) available in *OpenFOAM* versions. As in the version used in the physical problem, the algorithm is pressure based and works in a segregated way, by solving the adjoint pressure and velocity separately and then making corrections. While the adjoint velocity is calculated by solving the adjoint momentum equations (Eq. 4.20), the adjoint pressure is calculated by an equation resulting from the combination of adjoint momentum and adjoint continuity (Eq. 4.21) equations. A short explanation is presented next.

The adjoint momentum equation for a cell with center P can be written in semi-discretized form as:

$$a_P^{\boldsymbol{\psi_u}} \boldsymbol{\psi}_{\boldsymbol{u}P} + \sum_N a_N^{\boldsymbol{\psi_u}} \boldsymbol{\psi}_{\boldsymbol{u}N} = \boldsymbol{r} - \nabla \psi_c \tag{5.13}$$

where $\boldsymbol{\psi}_{\boldsymbol{u}P}$ is the adjoint velocity in a cell of interest, $\boldsymbol{\psi}_{\boldsymbol{u}N}$ are the adjoint velocities in the neighbor cells and $a_i^{\boldsymbol{u}}$ are the coefficients from the discretization.

The adjoint velocity can be calculated by:

$$\boldsymbol{\psi_u}_P = (a_P^{\boldsymbol{\psi_u}})^{-1}\boldsymbol{H}(\boldsymbol{\psi_u}) - (a_P^{\boldsymbol{\psi_u}})^{-1}\nabla\psi_c \tag{5.14}$$

By substituting Eq. 5.14 in the incompressible adjoint continuity equation, an equation to calculate the adjoint pressure can be derived:

$$\nabla\cdot\boldsymbol{\psi_u} = \nabla\cdot\left((a_P^{\boldsymbol{\psi_u}})^{-1}\boldsymbol{H}(\boldsymbol{\psi_u})\right) - \nabla\cdot\left((a_P^{\boldsymbol{\psi_u}})^{-1}\nabla\psi_c\right) = 0 \tag{5.15}$$

During the calculation a linearization in the adjoint convective term is performed as described by Eqs. 4.33 and 4.34 for the physical problem, so the face flux $\phi^f_{\psi_u}$ must be updated with the new adjoint pressure field. By opening the adjoint continuity equation and using Eq. 5.14, $\phi^f_{\psi_u}$ can be written as:

$$\nabla\cdot\boldsymbol{\psi_u} = \sum_f \boldsymbol{S}_f\cdot\boldsymbol{\psi_u}_f = \sum\phi^f_{\psi_u} \tag{5.16}$$

$$\phi^f_{\psi_u} = \boldsymbol{S}_f\cdot\left[(a_P^{\boldsymbol{\psi_u}})^{-1}\boldsymbol{H}(\boldsymbol{\psi_u}) - (a_P^{\boldsymbol{\psi_u}})^{-1}\nabla\psi_c\right]_f \tag{5.17}$$

In order to avoid divergence during the calculation, under relaxation is applied to all equations of the system.

In *OpenFOAM* v2006, the adjoint pressure-based solver for incompressible flows works by the following steps:

a) Initialize $\boldsymbol{\psi_u}$ and ψ_c using latest available values;

b) Assemble the adjoint momentum equation (Eq. 5.14);

c) Under-relax the adjoint momentum equations;

d) Solve the adjoint momentum equation to obtain a prediction for $\boldsymbol{\psi_u}$;

e) Assemble the adjoint pressure equation (Eq. 5.15) with the calculated adjoint velocity;

f) Calculate the new adjoint pressure;

g) Based on the adjoint pressure solution, assemble the adjoint face flux (Eq. 5.17);

h) Under-relax the calculated adjoint pressure field;

i) Correct the adjoint velocity field using the new adjoint pressure field (use Eq. 5.14);

j) Repeat until reach convergence.

5.3 Solution of the Discrete Adjoint Problem

The solution of the discrete adjoint system corresponds to solving the problem stated by Eq. 4.29. This time, the system is solved in a coupled manner, so all the adjoint variables are calculated by solving the same linear system and no correction steps are involved. As the system is being perturbed, the components of the matrix $\partial \boldsymbol{R}/\partial \boldsymbol{w}$ and the vector $\partial J/\partial \boldsymbol{w}$ are stored. After perturbation of all groups are done, the matrix and the vector are written in a binary file and a separate executable is called. The main code stops and waits for the adjoint system to be solved. When the desired convergence error is achieved, the executable save the adjoint solution $\boldsymbol{\psi_{adj}}$ in a binary file and the main code starts to work again. It reads the new solution and proceeds with the optimization. Among all the options for solvers inside *PETSc*, the flexible version of the Generalized Minimal Residual (*FGMRES*) with restart or the Conjugate Gradient Method on the Normal Equations (*CGNE*) are used, depending on the problem. For the preconditioning, the block Jacobi is used as a global preconditioner and the incomplete lower and upper factorization (*ILU*) as a local preconditioner for each subproblem. Also, the Jacobian matrix is scaled to improve the diagonal dominance and condition number. This combinations proved to be effective and the option using FGMRES can also be seen in other works, such as He et al. (2018), He et al. (2019), Kenway et al. (2019) and He et al. (2020).

5.4 Adaptive Mesh Refinement

In order to improve the definition of the boundaries of the designs without excessively increasing the computational cost of the problem, an adaptive mesh refinement (AMR) scheme is implemented. To refine only the borders of the design being created, AMR is applied only in the cells chosen by the magnitude of the gradient of the design variables. All cells where:

$$\mathrm{AMR}_{inf} \leq |\nabla \left(\rho_{des,i}\right)| \leq \mathrm{AMR}_{sup} \tag{5.18}$$

are cut in all three directions. The predefined constants AMR_{inf} and AMR_{sup} establish lower and upper values for the choice of cells to be refined.

The refinement can be applied only one time at a specific optimization iteration, several times at each m iterations (defined before the optimization starts) or when the difference between the value of the objective function at iteration n and its value at the previous iteration is smaller than or equal to a specified tolerance ϵ_{AMR}. For the volume constraint, the comparison is performed between the volume fraction at iteration n and the specified volume fraction. The criteria for objective function and volume constraint

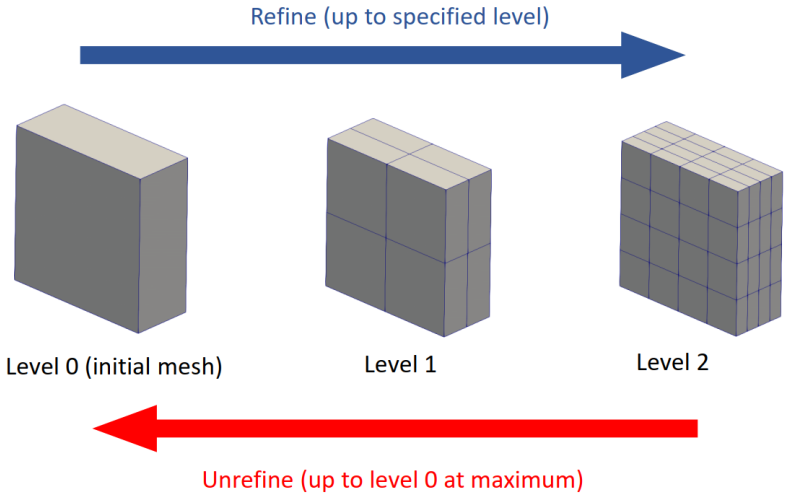
are presented in Eqs. 5.19 and 5.20:

$$\left| \frac{J^n - J^{n-1}}{J^{n-1}} \right| \leq \epsilon_{\text{AMR}} \tag{5.19}$$

$$\left| \frac{\int_\Omega \rho_{des}^n \, d\Omega}{\int_\Omega \, d\Omega} - V_{frac} \right| \leq \epsilon_{\text{AMR}} \tag{5.20}$$

This feature can be applied only to hexahedral cells. Cell coarsening is also allowed, but only up to the original mesh size. In Fig. 5.1, an example of the refinement for one cell can be seen.

Figure 5.1 – Adaptive mesh refinement for 1 cell.



5.5 Optimization Scheme

In the continuous adjoint approach, the Globally Convergent version of the Method of Moving Asymptotes (GCMMA) is employed. In this version, a set of inner and outer iterations is used. An outer iteration starts from the current values of the design variables and ends up with an updated set of values. In each inner iteration (inside an outer iteration), a convex subproblem is generated and solved. Within this subproblem, the original objective and constraints are replaced by convex separable functions which approximate the original functions around the current state. The solution found for the subproblem can be accepted or rejected. If it is accepted, the solution becomes the new state for the design variables and the outer iteration is finished. If the solution is rejected, a new inner iteration is made with a modified subproblem based on modified approximating functions. These inner iterations are repeated until the approximating objective and constraint functions become greater than or equal to the original functions at the optimal solution of the subproblem. For specific details of the algorithm, the reader is referred to Svanberg (2002) and Svanberg (2007).

In the discrete adjoint approach, the optimization is performed by using the steepest descent. Hence, the design variables are updated by stepping in the opposite direction of the gradient as described by Eq. 5.21:

$$\rho_{des}^{n+1} = min\left(max\left(\rho_{des}^{n} - \delta_{SD}\ \frac{dL}{d\rho_{des}}, 0\right), 1\right) \tag{5.21}$$

where the superscript n indicates the current optimization iteration and δ_{SD} is the step used in the steepest descent. The min and max functions are used to keep the design variables in the range of $[0, 1]$. The volume constraint c is applied by using the Augmented Lagrangian Method (WRIGHT; NOCEDAL, 1999). The Augmented Lagrangian $\mathcal{L}$ is defined as:

$$\mathcal{L} = J(\boldsymbol{w}(\boldsymbol{\rho_{des}}), \boldsymbol{\rho_{des}}) - \boldsymbol{\psi_{adj}^{T}}\boldsymbol{R}(\boldsymbol{w}, \boldsymbol{\rho_{des}}) - \psi_{vol}c + w_{vol}c^2 \tag{5.22}$$

where ψ_{vol} is the Lagrange multiplier for the constraint function c and w_{vol} is the weight for this same function. Hence, when the constraint is enabled, the design variables are updated by considering additional terms as follows:

$$\rho_{des}^{n+1} = min\left(max\left(\rho_{des}^{n} - \delta_{SD}\ \left(\frac{d\mathcal{L}}{d\rho_{des}} - (\psi_{vol}^{n} - 2w_{vol}^{n}c_{vol}^{n})\ \frac{\partial c}{\partial \rho_{des}}\right), 0\right), 1\right) \tag{5.23}$$

At the beginning of the optimization, $\psi_{vol} = 0$ and w_{vol} assumes a positive value. Both variables are updated during the optimization by:

$$\psi_{vol}^{n+1} = \psi_{vol}^{n} - 2w_{vol}^{n}c^{n} \tag{5.24}$$

$$w_{vol}^{n+1} = min\left(\gamma w_{vol}^{n}, w_{max}\right) \tag{5.25}$$

where γ is a multiplier (positive and > 1) to increase the weight of the constraint and w_{max} is the maximum value that the weight can assume. Except for ψ_{vol}, all the variables defined to implement the constraint by the Augmented Lagrangian Method must have values assigned before the optimization starts and depend on each case. In the numerical examples presented, all values used are found by trial and error.

Whether it is the continuous or the discrete approach, the sequence of steps used in the optimization is performed as follows (see Fig. 5.2):

a) An initial design is defined.

b) The groups of cells and faces to perturb the system and assemble the adjoint problem is defined for the initial mesh.

c) The primal problem (flow problem) is solved.

d) The adjoint problem is assembled and solved.

e) The sensitivities are calculated with the results of the previous steps.

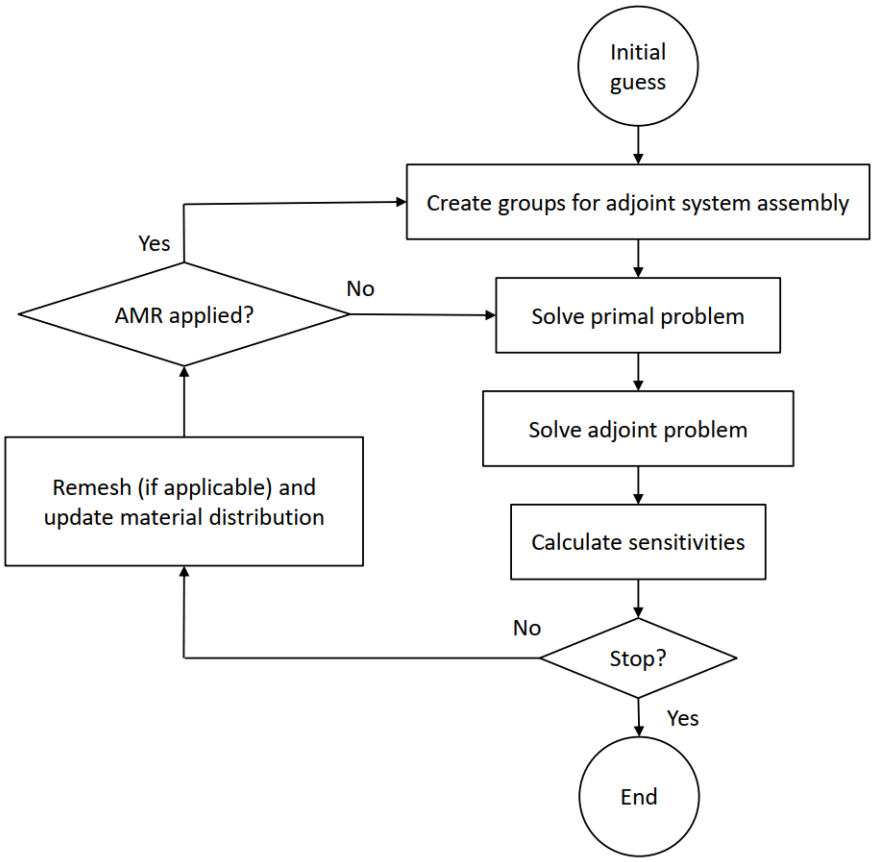
f) The stopping criteria are assessed.

g) If the stopping criteria are not satisfied, design variables are updated and AMR can be applied.

h) If AMR is applied, new groups to perform the perturbation of the primal problem are created and the code proceeds to solve the primal problem. If AMR is not applied, the code proceeds directly to solve the primal problem.

i) The steps described are repeated until the stopping criteria are satisfied.

Figure 5.2 – Optimization flowchart.



The optimization stops if the maximum number of iterations is reached or if the difference between two consecutive objective function values is less than a specified tolerance ϵ_J. Also, if the volume constraint is active, a tolerance ϵ_{vol} is checked. The stopping criteria for objective function and volume constraint can be seen in Eqs. 5.26

and 5.27:

$$\left| \frac{J^n - J^{n-1}}{J^{n-1}} \right| \leq \epsilon_J \tag{5.26}$$

$$\left| \frac{\int_\Omega \rho_{des}^n \, d\Omega}{\int_\Omega \, d\Omega} - V_{frac} \right| \leq \epsilon_{vol} \tag{5.27}$$

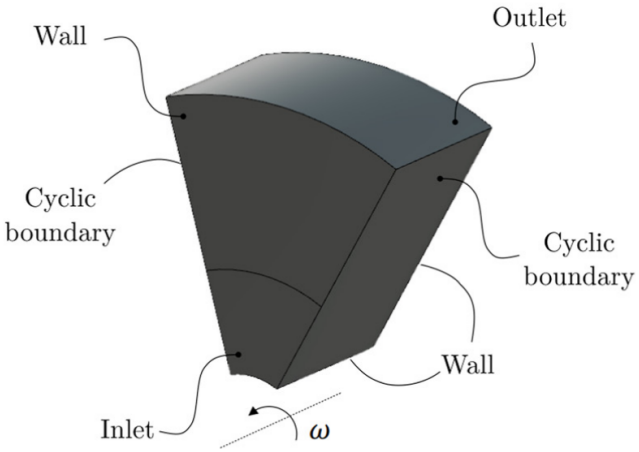
6 NUMERICAL EXAMPLES

In this section, numerical examples are presented to show the frameworks described in the previous chapters working. In the first example, the continuous adjoint is used in an optimization case to design a 3D rotor working with incompressible laminar flow. In the second example, the discrete adjoint developed is applied to the traditional double-channel case. The third example consists in a comparison of times between the continuous and the discrete approaches for the same optimization problem. The fourth example is an assessment of the hypothesis explained for the adiabatic boundary representation in Sec. 3.2. The fifth case is an application of the discrete adjoint to a compressible flow problem without turbulence, considering different cell topologies. The sixth example shows the application of the discrete adjoint to a 3D case of compressible flow. The seventh case is an optimization of a rotor working with compressible flow and without turbulence. The eightieth case is an assessment of the penalization needed when the turbulence model is considered. In the ninth case, a channel with compressible turbulent flow is optimized with the discrete adjoint approach. Finally, in the tenth case, a high efficiency design of a rotor is obtained by optimization to work with compressible turbulent flow. In all cases, the divergence scheme linear upwind (ESI, 2022b) is used. For the turbulent cases, the properties of air at 20^oC and 1 atm are considered.

6.1 Incompressible Laminar Rotor

In the first numerical example, the continuous adjoint approach is used to design a 3D rotor. The objective is to minimize the dissipated power inside the rotor (Eq. 3.3) and compare the value of the optimized case with a traditional design. The domain used for the optimization is a 60^o sector of a circular crown. Cyclic boundary conditions are defined to represent a full rotor. The boundary condition definitions are presented in Fig. 6.1. The flow enters in the domain axially and leaves at the outer radius. The inlet area goes from a diameter of 0.10 m to 0.25 m. The outlet diameter is 0.50 m and the domain width is 0.125 m. The inlet velocity is composed by a normal component of 5 m/s and a tangential component according to a rotation of 1000rpm. The kinematic viscosity ν is 0.01 m^2/s. The outlet pressure is 0 Pa. The value used for $\overline{\alpha}$ is 2.5×10^4.

Figure 6.1 – Boundary condition definitions for the 3D rotor case.



In this case, the traditional design considered is composed of six straight radial blades. In Fig. 6.2, the full rotor and the 60^o sector used in the simulations are presented. To assess the performance of the straight bladed design, the physical problem is calculated until the residuals reach 10^{-4} and the objective function exhibits a steady behavior. Velocity and pressure contours and streamlines are presented in Fig. 6.3 for the full geometry (the results for the sector are rotated). As can be seen, the misalignment of the blades with the incoming flow results in sharp variations for the velocities and pressures and causes big recirculation areas. The final value for the objective function of the sector is $-6.09\,\mathrm{W}$.

Figure 6.2 – Straight blade design.

(a) Full rotor. **(b)** Sector used in the simulation.

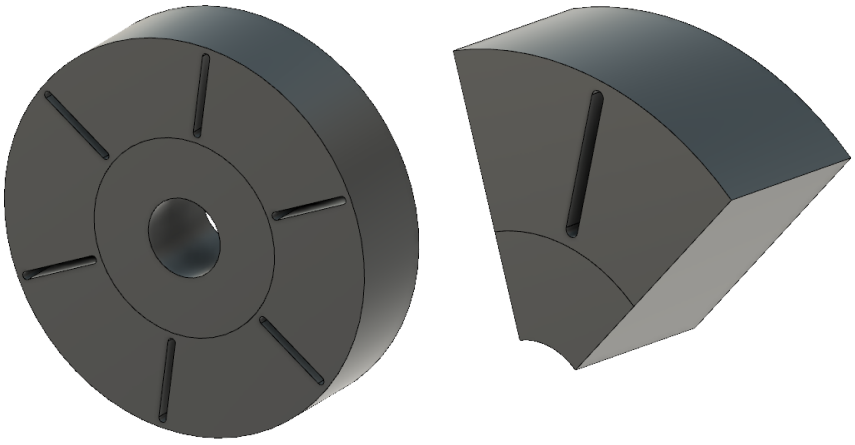
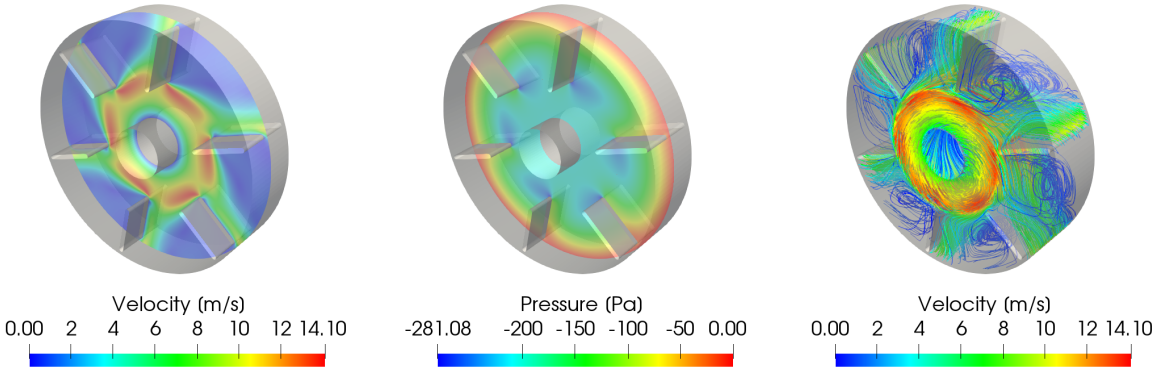


Figure 6.3 – Velocity, pressure and streamlines for the straight bladed rotor.

(a) Velocity contour. **(b)** Pressure contour. **(c)** Streamlines.



In the optimization, a volume constraint of 50% is used. The initial state is a uniform field where the design variables satisfy the volume constraint. The parameter q (from Eq. 3.11) starts with a value of 0.01 and, after 25 iterations, increases with steps from 0.1 up to the value of 1.0. The physical and adjoint problems are calculated until the residuals reach 10^{-4}. The optimization is conducted until the relative difference between objective function values is lower than 10^{-4}. The optimized topology for the sector and the full rotor can be seen in Fig. 6.4. Velocity and pressure contours at a plane at the mid width are presented in Fig. 6.5 together with streamlines.

Figure 6.4 – Optimized rotor.

(a) Full optimized rotor. **(b)** Sector used in the optimization.

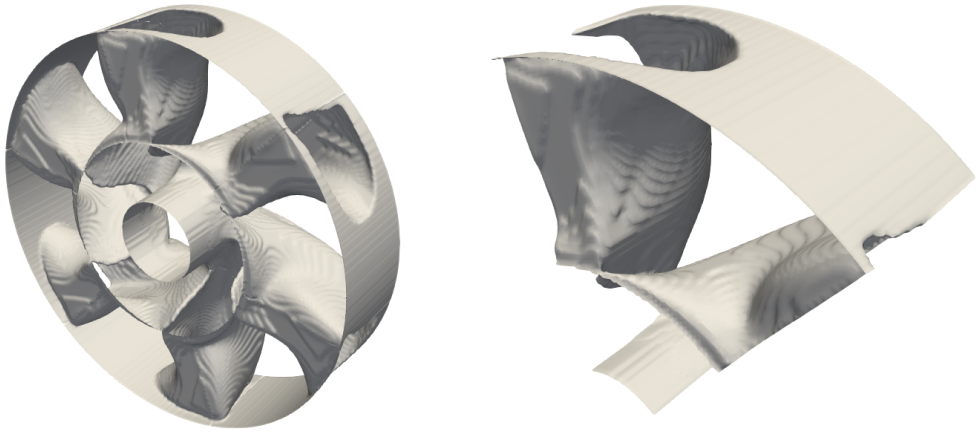
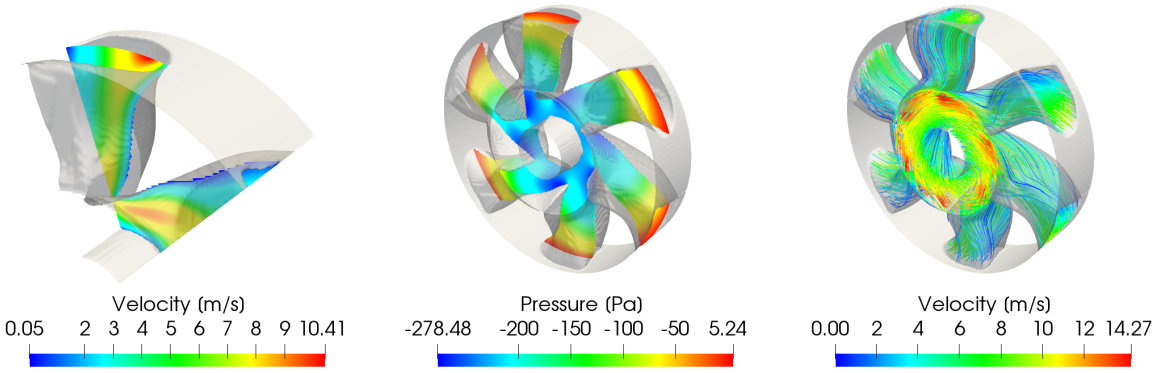


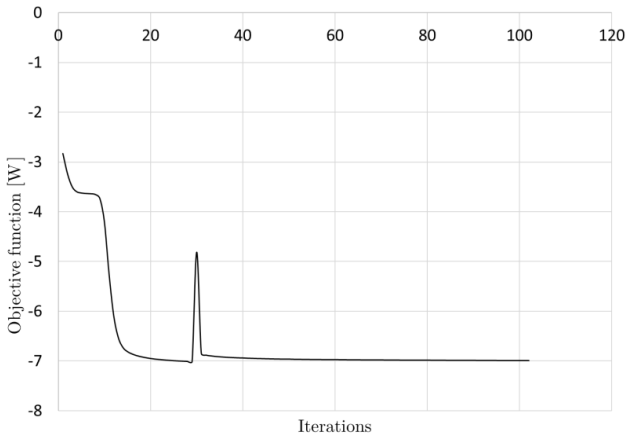
Figure 6.5 – Velocity, pressure and streamlines for the optimized rotor.

(a) Velocity contour. **(b)** Pressure contour. **(c)** Streamlines.



In the complete rotor, it is possible to see six channels. The flow path created connects the inlet to only a region of the outlet, creating an oval outlet area. The path is twisted close to the inlet. This twist corrects the initial flow rotation and no big recirculation region is seen in the streamlines. At the outlet, the channel has a smooth surface inclined in a direction opposed to the rotation, similar to traditional backward-facing blades. In the pressure plots, it is possible to see that the path division near the inlet is smooth and does not create pressure spikes at the corner. The resulting design is well defined and with no regions of intermediary values for the design variables. The objective function is approximately $-7.00\,\mathrm{W}$ for the sector. The objective function evolution is presented in Fig. 6.6. At approximately 30 iterations, a spike can be seen. This point marks where parameter q starts to change.

Figure 6.6 – Objective function for the 3D rotor case.

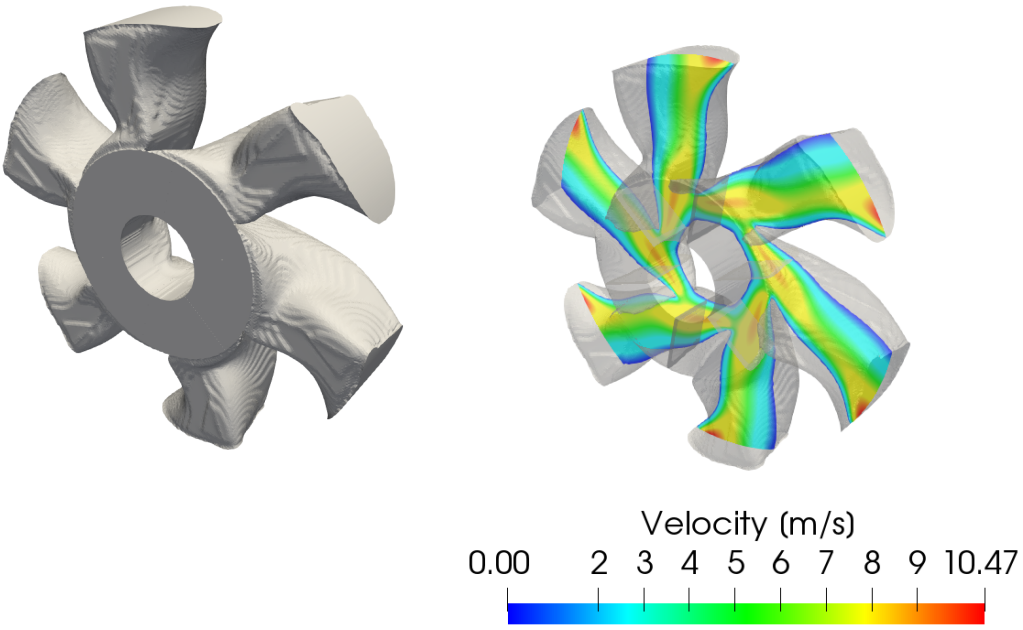


The optimized design is post processed to remove the material model effects and compare its performance with the straight bladed design. Again, a sector is used for the simulation, but the results in Fig. 6.7 are presented for the full rotor. As can be seen, the velocity and pressure contours indicate a smooth variation inside the flowpath. Also, the

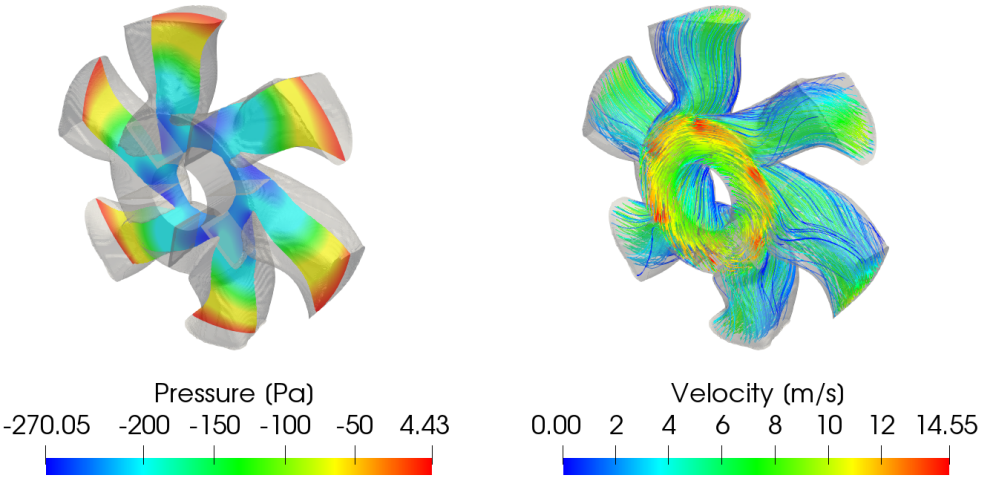
streamlines show a well organized flow without big recirculation areas. The final objective function for post processed case is approximately $-6.67\,\mathrm{W}$, being 9.52% better than the straight bladed rotor.

Figure 6.7 – Velocity, pressure and streamlines for the post processed geometry.

(a) Post processed geometry. **(b)** Velocity contour.



(c) Pressure contour. **(d)** Streamlines.

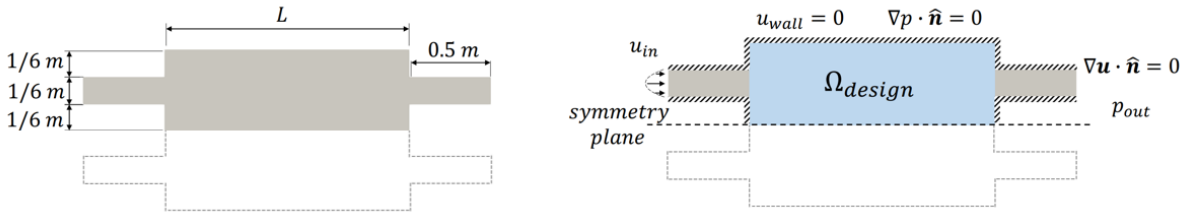


6.2 Incompressible Laminar Double Channel

In this first example using the discrete adjoint developed, the traditional double-channel optimization is presented with the objective of verifying the method working with well established results from the literature. Two domains are defined. One is composed by a short path ($L = 1\,\mathrm{m}$ in Fig. 6.8) and the other is a longer path ($L = 1.5\,\mathrm{m}$). The case considers a volume constraint $V_{frac} = 1/3$. The adaptive mesh refinement is used with

the option of refining after the objective function changes are smaller than a specified tolerance. The values used are $\epsilon_{\text{AMR}} = 10^{-2}$ for the AMR and $\epsilon_J = \epsilon_{vol} = 5 \times 10^{-4}$ for the optimization. The dimensions, boundary definitions and optimization domain can be seen in Fig. 6.8.

Figure 6.8 – Dimensions, design domain and boundary conditions.



The kinematic viscosity ν considered is $1\,\text{m}^2/\text{s}$, the outlet pressure p_{out} is $0\,\text{Pa}$, the inlet velocity u_{in} has a parabolic profile with maximum velocity of $11\,\text{m/s}$ and the walls are defined as zero velocity. The region considered for the optimization (Ω_{design}) is highlighted in blue and the objective function is the energy dissipation (Eq. 3.3).

In the short channel case, the optimization runs for 56 iterations. The mesh is refined at iterations $32, 34, 35, 42$ and 43. The final topology can be seen in Fig. 6.10 at the left and the objective function is $0.57\,\text{W}$. The starting mesh has 3108 cells and the final mesh is composed of 35226 cells (see Fig. 6.9a). In the case of the long channel, the optimization stops after 198 iterations. The AMR is applied at iterations $77, 148, 174, 175$ and 176. The final topology is presented in Fig. 6.10 at the right and presents an objective function value of $0.58\,\text{W}$. The starting mesh has 4672 cells and the final mesh is composed of 48495 cells (see Fig. 6.9b). The evolution of the objective function and the fluid fraction for this case are exhibited in Fig. 6.11. As can be noticed, the application of the AMR causes peaks of the objective function during the optimization, caused by small deviations in the interpolation.

Figure 6.9 – Initial and final meshes for the double channel optimization cases.

(a) Short case. **(b)** Long case.

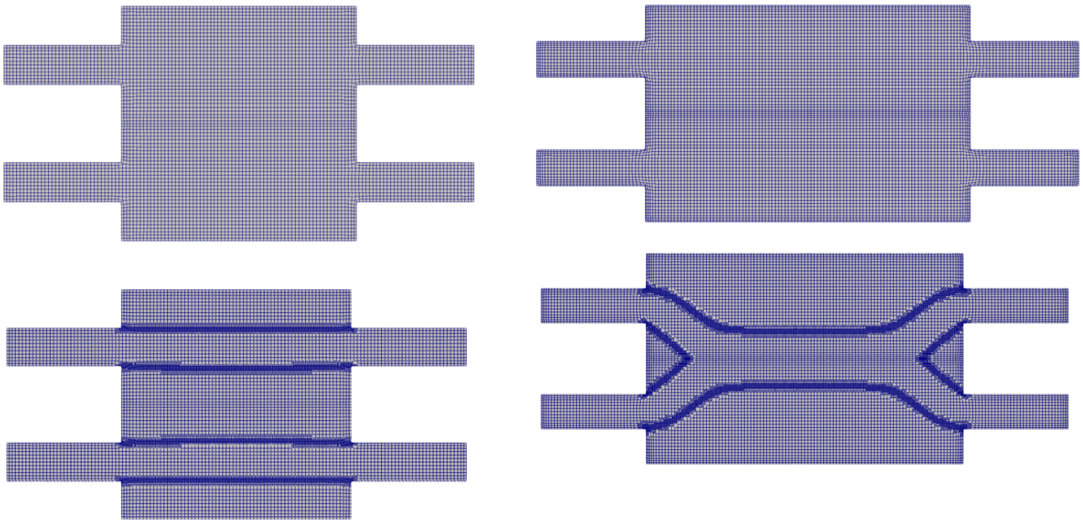


Figure 6.10 – Optimized designs.

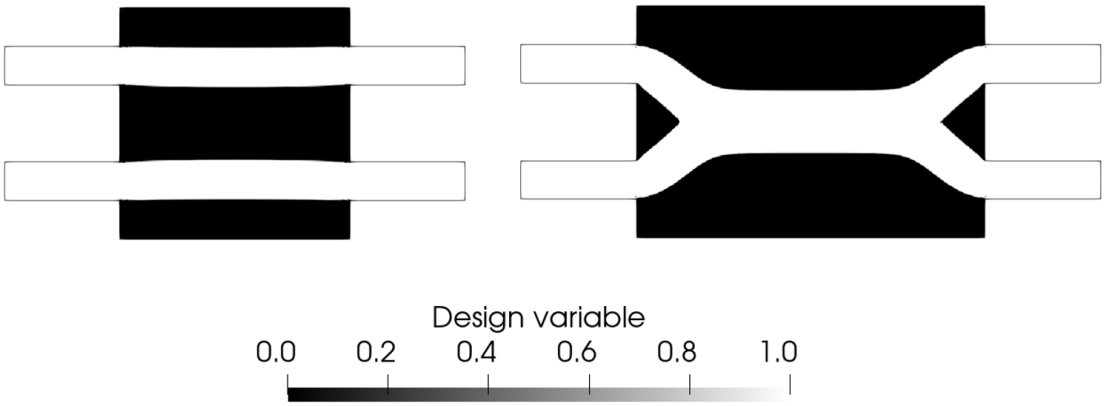
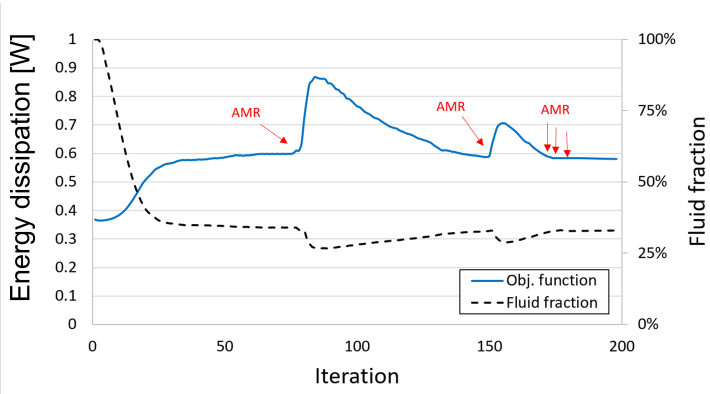


Figure 6.11 – Objective function and fluid fraction for the long channel.



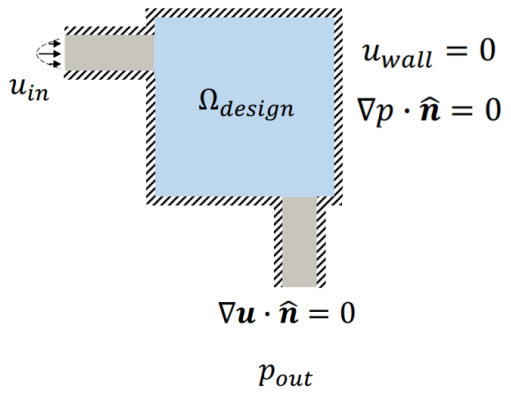
In both cases, the increase in the number of cells is slightly higher than 10 times the

initial values. As the AMR strategy only refines the interfaces, the final geometries have smooth boundary definitions and the computational cost is lower than it would be if a uniform and very fine mesh was used. For both cases, the results are coherent with the results seen in Borrvall and Petersson (2003), Challis and Guest (2009), Sá et al. (2016), where the short channel minimum is composed of two straight channels and the long channel case is optimized by having a design with a central wide channel. Hence, the proposed methodology is capable of reaching the well known results in the literature and is a viable option to perform the optimization for flow channels.

6.3 Incompressible Laminar Pipe Bend

This case presents a comparison between the two adjoint approaches described in this work, the continuous and the discrete, with the objective of comparing the computational cost of them by assessing the time taken to perform optimization cases. This time, a pipe bend example considering incompressible laminar flow is used. In Fig. 6.12, the geometry, the design domain and the boundary definitions are presented. The dimensions for this case are the same used for non-rotating cases of Sec. 4.3. The kinematic viscosity ν is $1\,\mathrm{m}^2/\mathrm{s}$, the outlet pressure p_{out} is $0\,\mathrm{Pa}$, the inlet velocity u_{in} is parabolic with a maximum velocity of $1\,\mathrm{m/s}$ and the walls are defined as zero velocity. The region considered for the optimization (Ω_{design}) is highlighted in blue and the objective function is again the energy dissipation (Eq. 3.3).

Figure 6.12 – Design domain and boundary conditions.



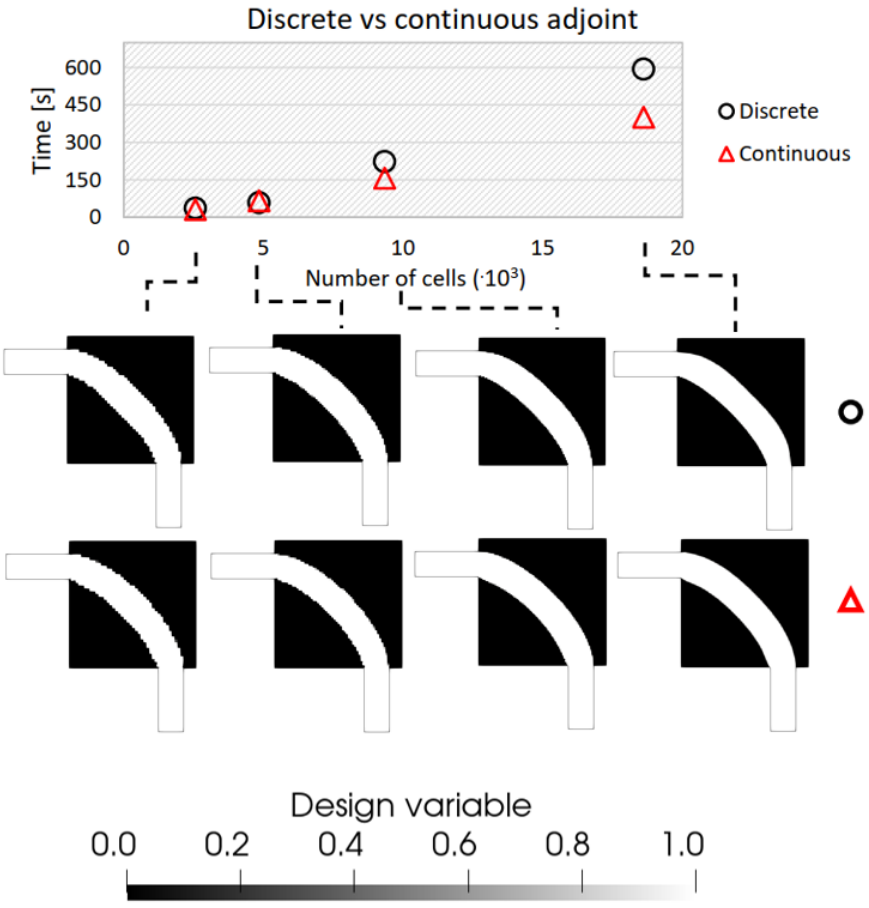
Different meshes are also used to assess the variation of time when the number of cells is increased. Four mesh refinements are defined. In the continuous adjoint , the strategy used is the one-shot approach (PAPOUTSIS-KIACHAGIAS; GIANNAKOGLOU, 2016), which means that the design variables are updated before the primal and adjoint systems are fully converged, *i.e.*, few solver iterations are run between the updates. In the discrete approach, the update of the design variables is performed after the primal and the adjoint

systems are solved and the tolerance is met (10^{-4} for this case). The volume constraint defined is defined $V_{frac} = 0.08\pi$, which represents 1/4 of a circular crown between the inlet and the outlet. The stopping criteria are $\epsilon_J = \epsilon_{vol} = 1 \cdot 10^{-2}$. The parameters necessary for the solutions of the primal and adjoint problems are set as close as possible for both approaches in order to have a fair comparison. Only one thread is used for the calculations, so no parallelization is considered in the comparison. In Tab. 6.1, the number of cells used for each mesh and the final objective function values are presented. In Fig. 6.13, the computational times for each approach versus the number of cells, as well as the resulting topologies are exhibited.

Table 6.1 – Objective function values of continuous and discrete adjoint approaches..

Number of cells	J_{incomp} (cont.)	J_{incomp} (disc.)
2560	0.46 W	0.45 W
4841	0.48 W	0.50 W
9332	0.50 W	0.47 W
18604	0.52 W	0.46 W

Figure 6.13 – Time and optimized topologies (continuous vs discrete approaches).



The two approaches are capable of reaching the same local minimum, considering that optimized topologies are very similar. The computational time for both approaches is very similar for the less refined meshes. However, the difference becomes more significant as the number of cells is increased, which is coherent with the results seen in (NADARAJAH; JAMESON, 2000). The continuous adjoint approach presents a better computational cost for the more refined meshes, given that it performs fewer operations to calculate the sensitivities.

The energy dissipation values in Tab. 6.1 show that the values are very close for the less refined meshes. However, the discrete approach achieved lower energy dissipation values (0.46 W) than the continuous adjoint approach for the more refined meshes (0.52 W). This occurs due to the more curved design from the cases solved with the discrete approach, which is caused by the fact that the optimizer updates the design variables by using the sensitivities calculated over fully converged fields (contrary to the *one-shot* scheme). The computational cost (here, measured by time) indicate a tendency of the discrete approach to present a higher cost increase than the continuous approach. For applications in large-scale cases, the discrete approach may require improvements focusing on the speed of the matrix assembling processes. It is worth to highlight that, even though it takes more time than the continuous approach, the discrete approach developed in this work is especially useful in problems where the complexity of the governing equations is such that the derivation of the adjoint problem by hand becomes too complicated (e.g.: compressible flows, non-newtonian flows). Also, sophisticated objective functions and constraints may difficult the use of the continuous adjoint, highlighting the importance of being able to use the discrete approach.

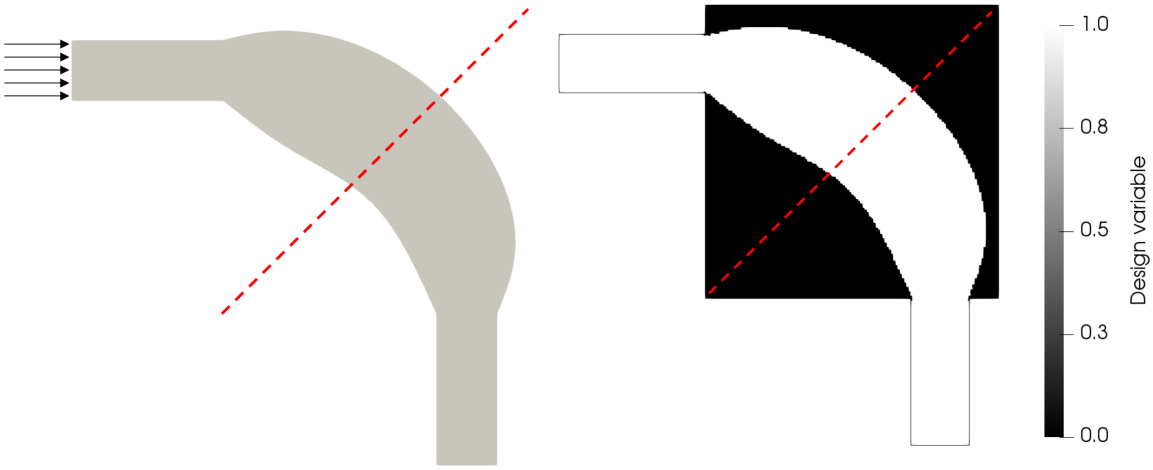
6.4 Material Model Assessment for Compressible Flows

In this example, the idea is to compare a body fitted mesh case with a case where the penalization is used to represent the geometry. The dimensions of the numerical domain used for this case are the same used for non-rotating cases of Sec. 4.3. The geometries used can be seen in Fig. 6.14. Flow enters the domain by the left edge (indicated by the black arrows) and leaves at the bottom edge. All other edges are considered walls. The dynamic viscosity μ is $0.1\,\mathrm{Pa}\cdot\mathrm{s}$, the specific heat C_p is $1005\,\mathrm{J/(kg}\cdot\mathrm{K)}$, the Prandtl number Pr is 0.705, the outlet pressure p_{out} is 101 325 Pa, the inlet velocity u_{in} is a uniform profile with 102 m/s, the inlet enthalpy h_{in} is $2.95\cdot10^5$ J/kg and the walls are defined as no-slip and adiabatic. The penalized domain is composed by a $1\times1\,\mathrm{m}$ square with extensions of length equals to 0.5 m. The material model is distributed so the body fitted geometry is draw inside the domain. The colors black and white indicates solid and fluid regions, respectively. Velocity, pressure and temperature are assessed in the section cut by the red line in Fig. 6.14 for α values going from 10^2 to 10^7. The calculations are performed until

the residuals are below 10^{-5} or a total of 5000 iterations is reached.

Figure 6.14 – Domains for material model assessment.

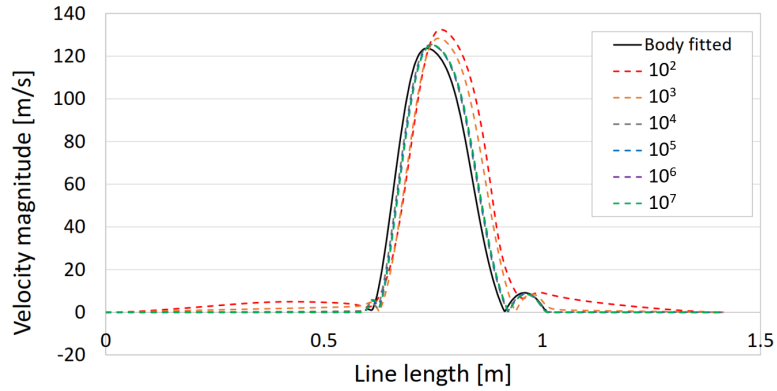
(a) Domain for body fitted mesh. **(b)** Domain for cases with penalization.



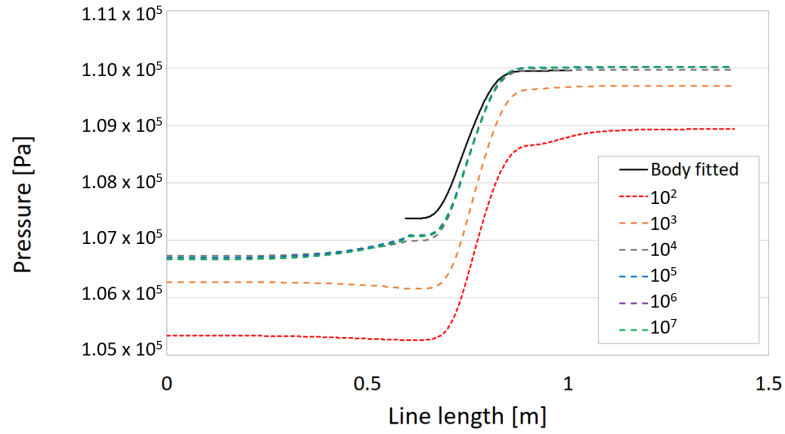
The results can be seen in Fig. 6.15. The black solid line is the reference case (body fitted mesh) and the dashed colorful lines are used to show the distribution of the variables for the penalized cases. As can be seen, the values calculated in the penalized cases exhibit a similar behavior when compared to the reference case and this behavior is improved for higher values of α. In the cases where $\alpha = 10^4$ or higher, almost no difference is visible, indicating that, for this case, 10^4 can already be used to represent the solid regions. Overall, the material model works as expected, showing that it can be used to represent solid regions inside the domain in topology optimization of compressible flows. It is important to note that the representation does not fit perfectly the reference values because the contour is not perfect. The representation using Brinkman penalization does not result in a smooth contour, so small differences are expected.

Figure 6.15 – State variables for different penalizations.

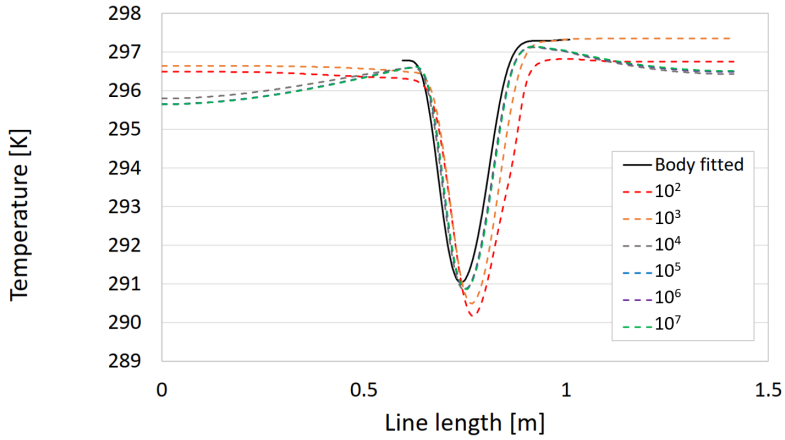
(a) Velocity for different penalizations.



(b) Pressure for different penalizations.



(c) Temperature for different penalizations.

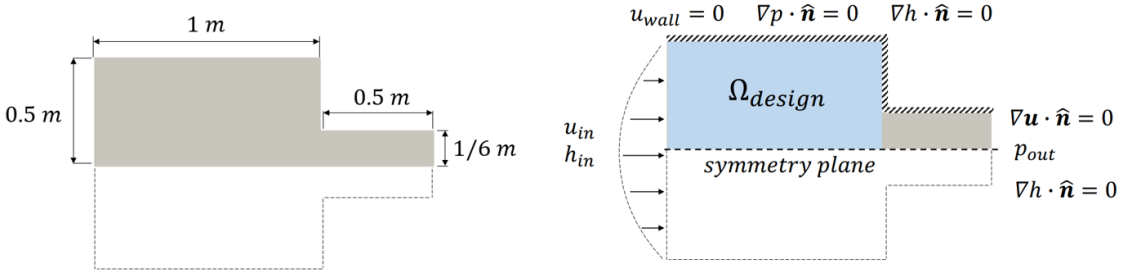


6.5 Compressible Laminar Convergent Channel

In this example, half of a convergent channel is optimized in three meshes with different types of cells, showing the adaptability of the method for different cell topologies. The dimensions, boundary conditions and design domain can be seen in Fig. 6.16. In this case,

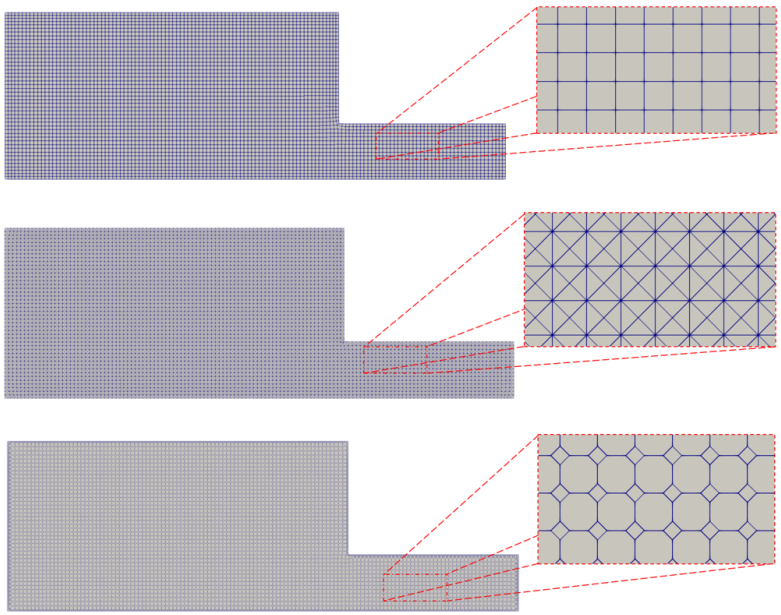
the inlet velocity u_{in} has a parabolic profile with maximum velocity of 65 m/s, resulting in a maximum inlet Mach number of 0.2. The dynamic viscosity μ is $1\,\text{Pa}\cdot\text{s}$, the specific heat C_p is $1005\,\text{J/(kg}\cdot\text{K)}$, the Prandtl number Pr is 0.705, the outlet pressure p_{out} is 101 325 Pa, the inlet enthalpy h_{in} is $2.95\cdot10^5$ J/kg and the walls are defined as no-slip. The objective function is the entropy variation, given by Eq. 3.9.

Figure 6.16 – Dimensions and boundary conditions.



Three meshes composed of different cell topologies are evaluated. Quadrilateral, triangular, and polygonal cells are used, as can be seen in Fig. 6.17. The volume constraint used for this case is $V_{frac} = 0.65$. The stopping criteria used for the optimization are $\epsilon_J = \epsilon_{vol} = 1\cdot10^{-3}$.

Figure 6.17 – Meshes used (top: quadrilateral; middle: triangular; bottom: polygonal).



The optimized designs for the three meshes are almost identical as can be seen in Fig. 6.18. The quadrilateral mesh case achieves convergence in 32 optimization iterations, the triangular mesh case takes only 27 and the polygonal mesh case uses 37 iterations. For the three meshes, the optimized topology connects the inlet to the path with a smaller section

area by a slightly curved design. The Mach numbers and density contours are exhibited in Fig. 6.19, showing that the flow inside the domains of the three cases are very similar.

Figure 6.18 – Optimized designs.

(a) Quadrilateral. **(b)** Triangular. **(c)** Polygonal.

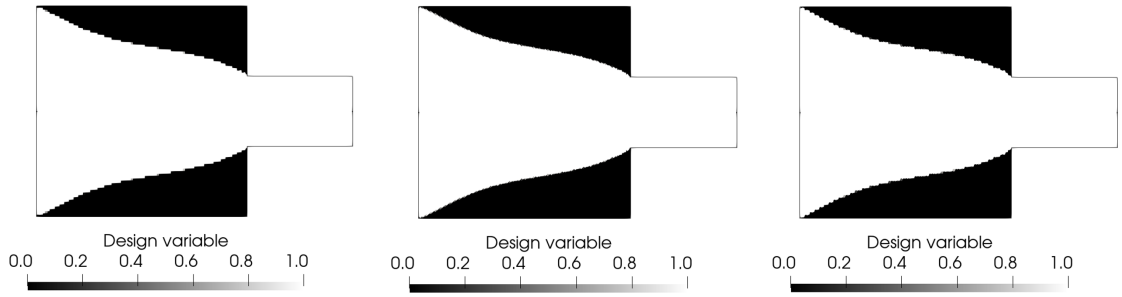
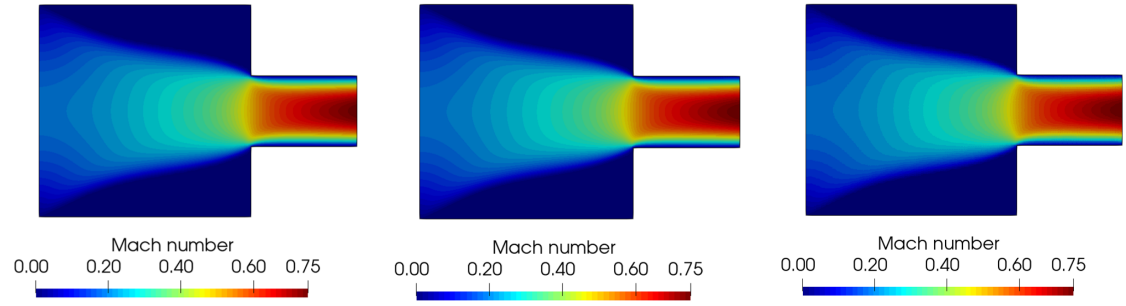
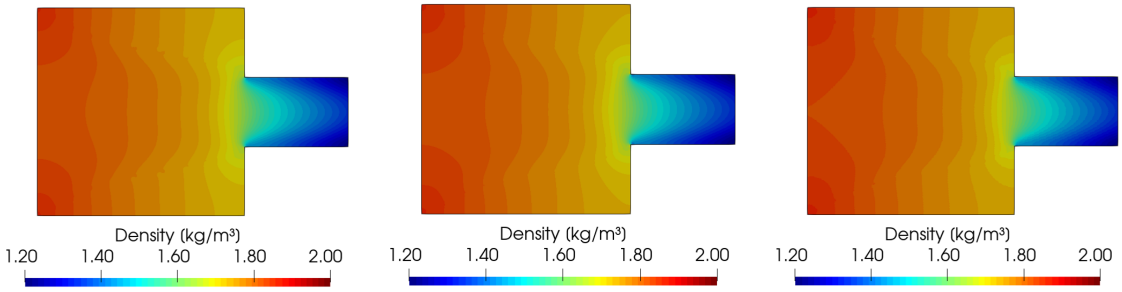


Figure 6.19 – Mach and density contours.

(a) Mach number (quadrilateral). **(b)** Mach number (triangular). **(c)** Mach number (polygonal).



(d) Density (quadrilateral). **(e)** Density (triangular). **(f)** Density (polygonal).

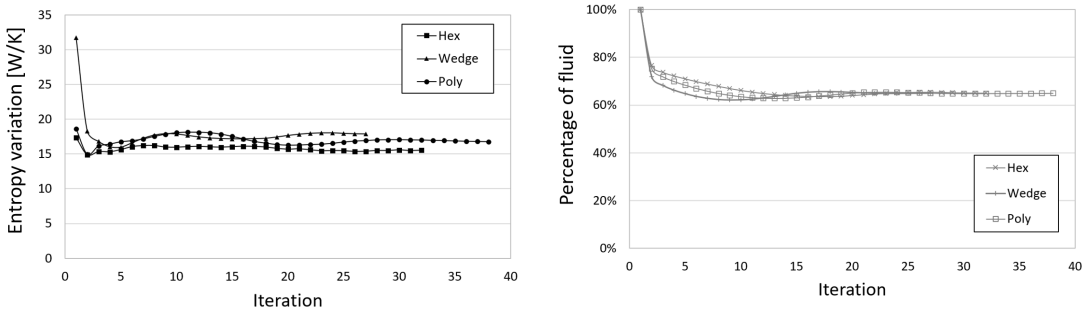


In Fig. 6.20, the objective function and the volume of fluid evolution for the three cases are presented. The mesh with triangular cells presents the highest value at the first iteration, however, all values are close at the end of the optimization. The final objective function values are 15.5 W/K, 17.9 W/K and 16.8 W/K for the quadrilateral, triangular and polygonal meshes, respectively. The differences observed are related to small differences at the solid-fluid interface created by the optimization. The quadrilateral mesh case resulted in almost no intermediate values for the design variable (no grey regions). The mesh with triangular cells shows a thin layer with intermediate values at the solid-fluid

interface and the polygonal mesh case is in between these two. Also, it is well known that different cell topologies have different accuracy in the calculation of the flow problem. This contributes to the differences observed in the objective function values. Even so, it is demonstrated that the methodology is capable of working with meshes using different cell topologies and reach very similar designs for the same problem.

Figure 6.20 – Objective function and fluid fraction for the convergent channel.

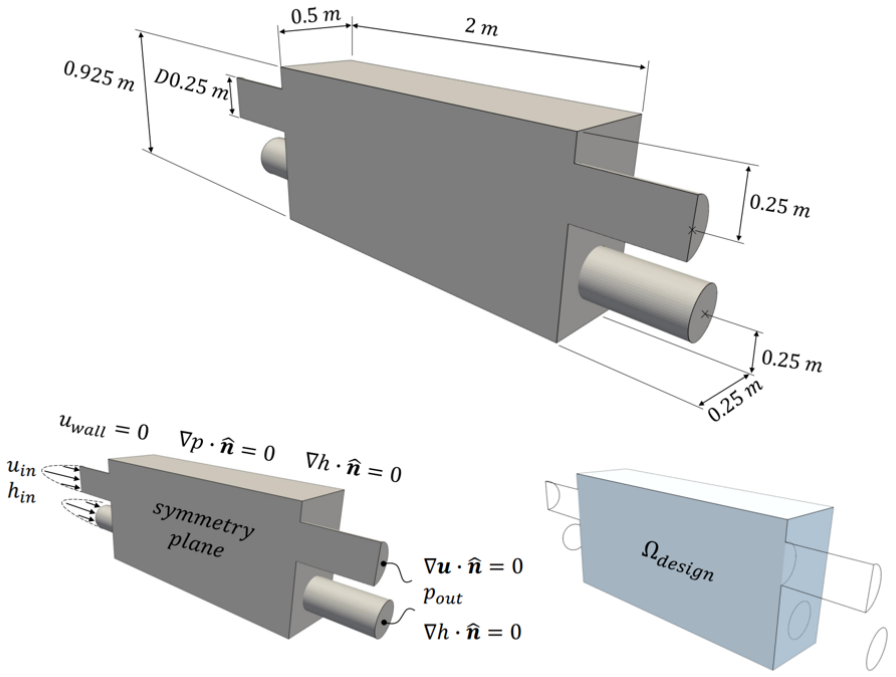
(a) Objective function. **(b)** Fluid fraction.



6.6 Compressible Laminar Triple Channel

This example is intended to show that the methodology can be directly expanded to 3D problems. In this case, a comparison between a case defined as a baseline and an optimized design generated by the discrete adjoint is presented. The model has a domain with three inlets and three outlets. The numerical domain calculates only half of the case by using a symmetry boundary condition. The dimensions, boundary condition definitions and design domain can be seen in Fig. 6.21. The inlet velocity u_{in} is a paraboloid with maximum velocity of 102 m/s (maximum inlet Mach number of 0.3), the dynamic viscosity μ is 1 Pa $\cdot$ s, the specific heat C_p is 1005 J/(kg $\cdot$ K), the Prandtl number Pr is 0.705, the outlet pressure p_{out} is 101325 Pa, the inlet enthalpy h_{in} is $2.95 \cdot 10^5$ J/kg and the walls are defined as no-slip. The objective function is the entropy variation (Eq. 3.9).

Figure 6.21 – Dimensions, boundary conditions and design domain.



The baseline case defined in this example is composed by an intuitive design for connecting the inlets to the outlets, which consists of three straight ducts. The fluid volume constraint V_{frac} is defined as $15\pi/296$ (based in the baseline design). The AMR is used in this case with a tolerance of $\epsilon_{\text{AMR}} = 10^{-2}$. The stopping criteria for the optimization is $\epsilon_J = \epsilon_{vol} = 10^{-3}$. The mesh used for the optimization is composed of hexahedral cells, starting with 60 216 cells and, after refinements at iterations $79, 123, 124$ and 143, finishing with 124 007 cells. The optimization stops after 144 iterations. In Fig. 6.22, the baseline and the optimized designs can be seen with the solid-fluid interface reflected w.r.t. to the symmetry plane. The Mach number and density contours are presented in Fig. 6.23.

The optimized design is composed of a wide central channel connecting the inlets and outlets. The connections created are smoothly curved to conduct the flow with minimal losses. This central channel has a bigger transverse section than the baseline straight ducts. The average *Ma* number is higher in the optimized case (Fig. 6.23) than in the baseline case (Fig. 6.23), indicating that the flow has lower resistance in this design. Also, it is possible to see that the pressure changes in the optimized case are smaller than in the baseline. This directly reflects in the entropy variation, being lower in the optimized topology. Hence, while the objective function for the baseline is around 1.41×10^3 W/K, for the optimized case this value is 1.09×10^3 W/K, representing a reduction of 22.7%.

Figure 6.22 – Baseline and optimized designs.

(a) Baseline. **(b)** Optimized design.

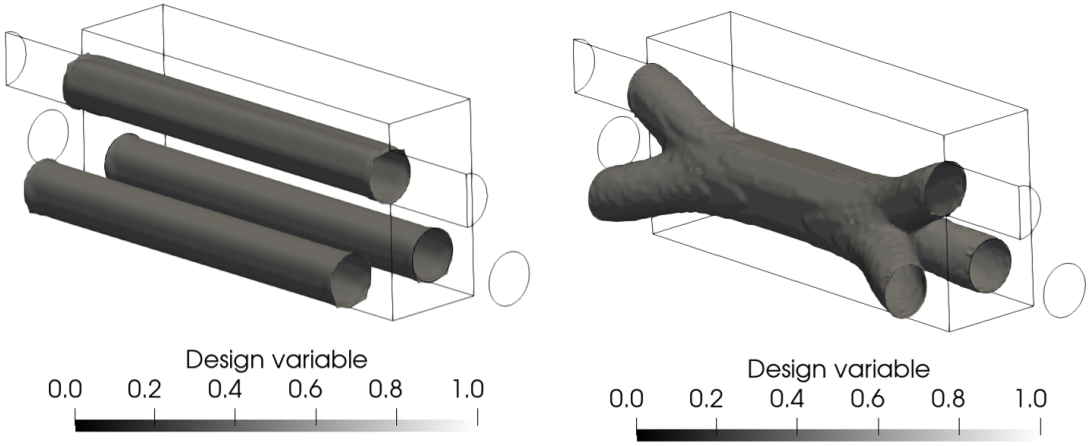
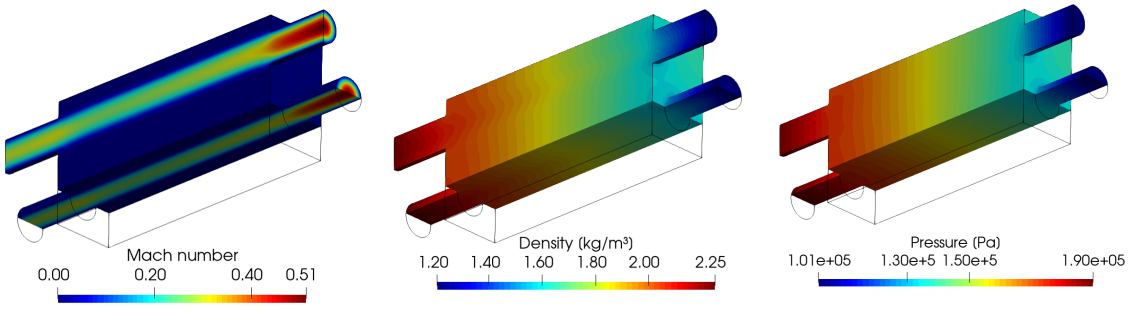
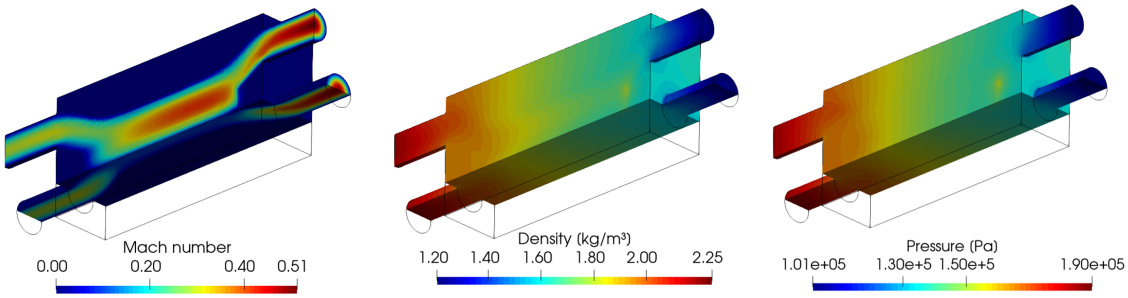


Figure 6.23 – Mach and density contours.

(a) Baseline Mach number. **(b)** Baseline density field. **(c)** Baseline pressure field.



(d) Optimized Mach number. **(e)** Optimized density field. **(f)** Optimized pressure field.



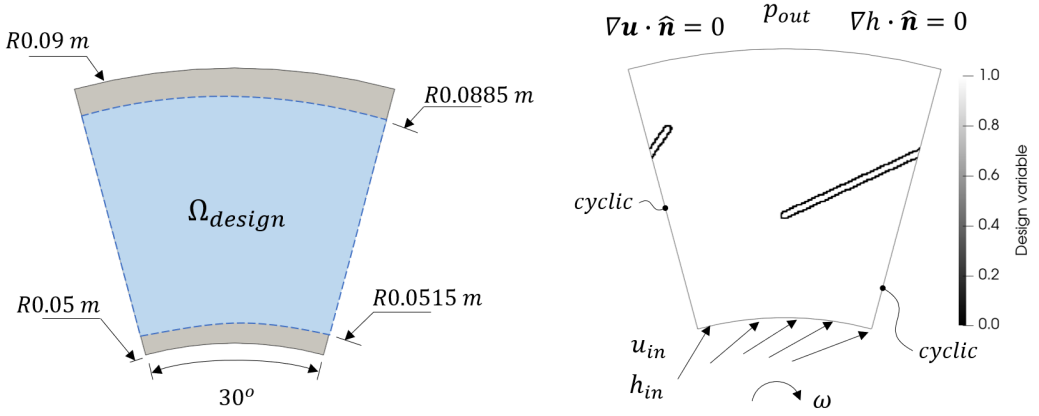
6.7 Compressible Laminar Rotor Optimization

In this example, topology optimization is used in a rotating compressible flow example, with the objective of designing a rotor with high isentropic efficiency (Eq. 3.10). The domain and design area used are presented in Fig. 6.24a. The boundary conditions can be seen with the initial guess in Fig. 6.24b. The dynamic viscosity μ considered is 10^{-2} Pa $\cdot$ s. The specific heat c_p is 1005 J/(kg $\cdot$ K), the Prandtl number Pr is 0.705, the outlet pressure p_{out} is 101325 Pa and the inlet velocity u_{in} is composed by a radial component of 100 m/s

and a tangential component according to the rotation of 22500 rpm.

Figure 6.24 – Rotor domain, boundary conditions and initial guess.

(a) Dimensions and design domain. **(b)** Boundary conditions and initial guess.



The optimization runs with a tolerance ϵ_J of 10^{-4} or a maximum of 50 iterations. Between each optimization iteration, the flow solution is calculated for 2 500 iterations. The optimization stops after 29 iterations. While the initial design (Fig. 6.24b) has an isentropic efficiency of 39.7%, the final design results in an isentropic efficiency of 44.7%. As can be seen in Fig. 6.25, it is composed by a thin slightly curved blade with a small solid portion after the trailing edge and. The objective function value for each optimization iteration is presented in Fig. 6.26. A noticeable improvement is seen in the first two iterations. After that, the efficiency improvements slowly decreases until the final design is obtained.

Figure 6.25 – Optimized rotor for compressible laminar flow.

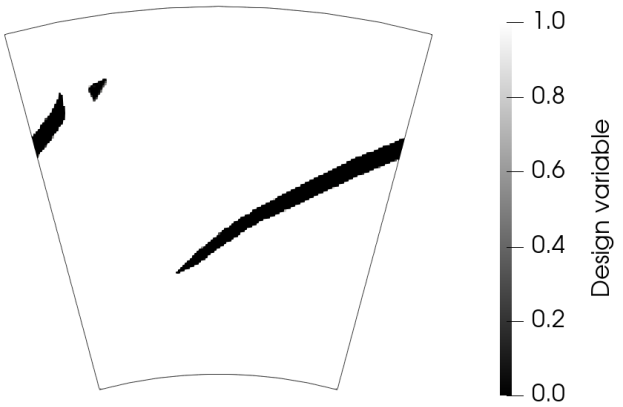
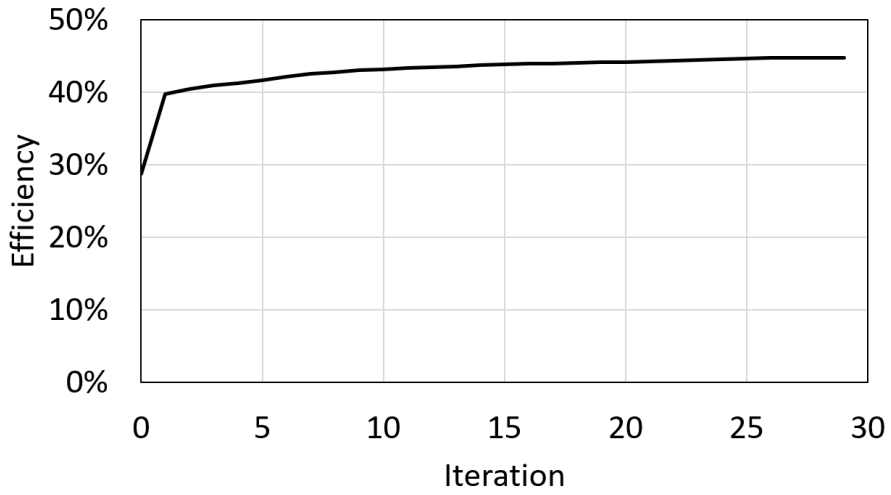


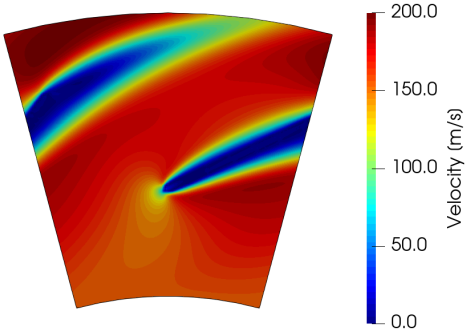
Figure 6.26 – Impeller isentropic efficiency during the optimization.



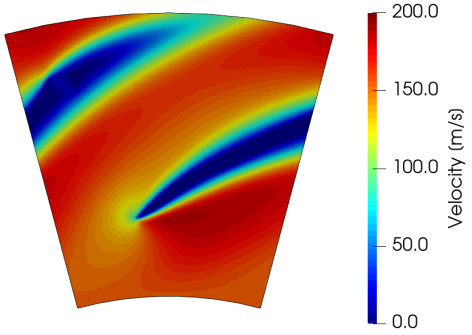
The velocity, pressure and temperature contours of the initial guess and of the final stage of the optimization are presented in Fig. 6.27. The Mach number and density contours can be seen in Fig. 6.28. As can be seen, the increase in the pressure gain in the optimized design (Figs. 6.27c and 6.27d) is expressive when compared to the initial guess. With a well behaved flow inside the rotor, this results in a more efficient design.

Figure 6.27 – Velocity, pressure and temperature contours (compressible laminar rotating case).

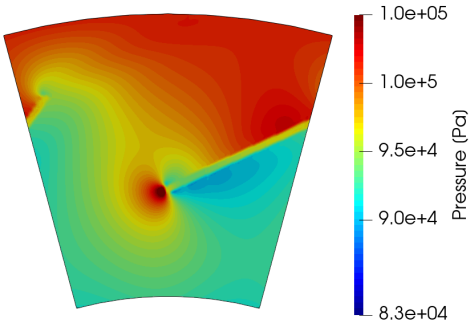
(a) Velocity (initial guess).



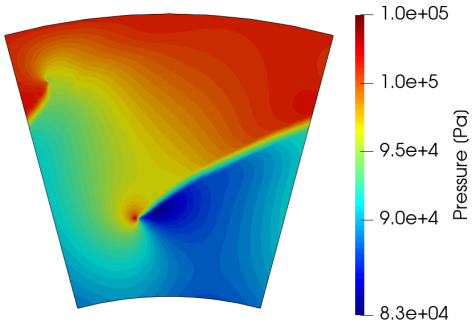
(b) Velocity (optimized design).



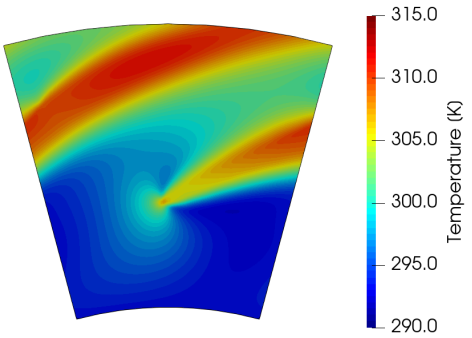
(c) Pressure (initial guess).



(d) Pressure (optimized design).



(e) Temperature (initial guess).



(f) Temperature (optimized design).

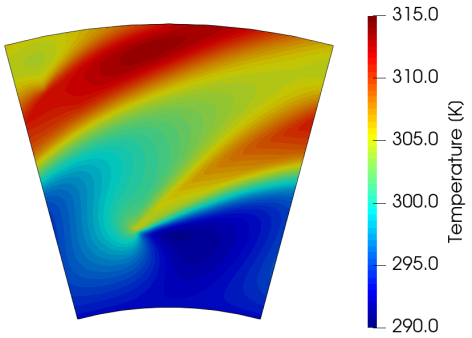
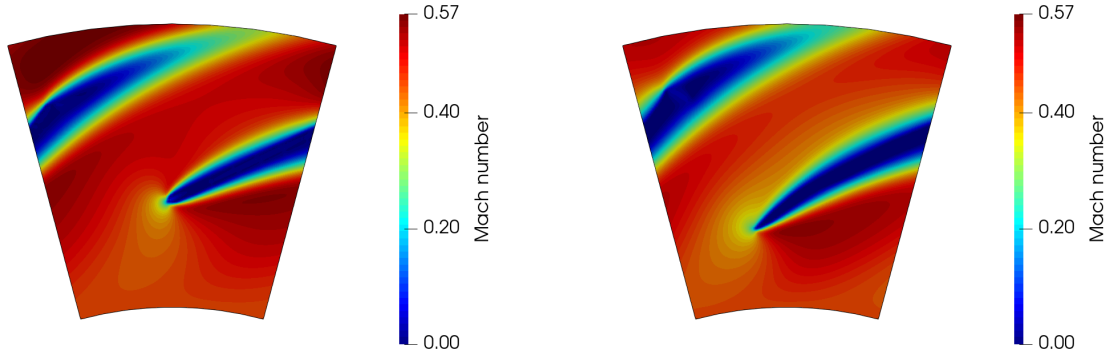
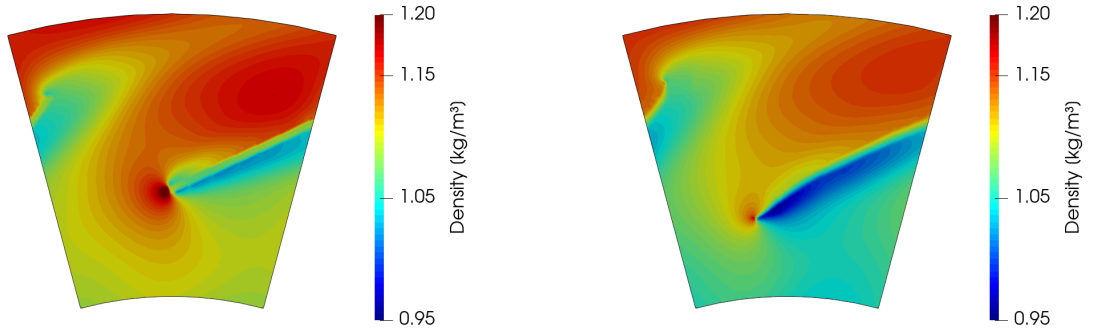


Figure 6.28 – Mach number and density (compressible rotating case).

(a) Mach number (initial guess). **(b)** Mach number (optimized design).



(c) Density (initial guess). **(d)** Density (optimized design).

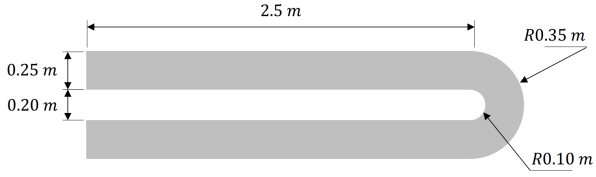


6.8 Penalization Assessment for Turbulent Cases

In this example, the objective is to assess the influence of different penalization values aiming to find the maximum value of $\overline{\alpha}$ needed when the turbulence model presented in Sec. 2.2 is used. The main concern is to have the material model performing as close as possible to impenetrable solid regions, resulting in realistic flow behaviors inside the domain.

A flow simulation performed in a body fitted mesh is established as a reference case. Simulations using the material model to represent the geometry are used to perform the comparisons by checking objective function values and profiles of the state variables at flow sections. The penalization assessment is performed only for compressible flow. The geometry of a U-shaped channel is used. The dimensions can be seen in Fig. 6.29.

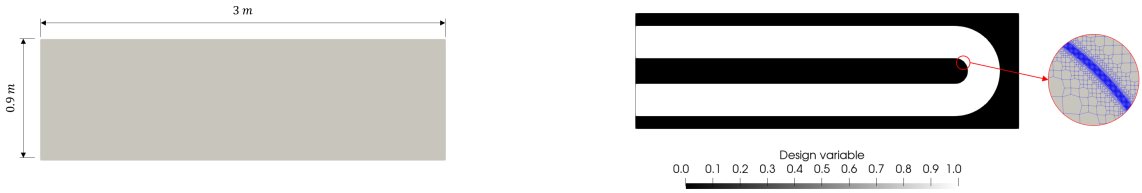
Figure 6.29 – U-channel dimensions.



The tests of the different penalization values are performed in a mesh larger than the U-shaped channel. Inside this domain, the channel is represented by the material distribution. The cell sizes used are similar to the body fitted case and, at the solid-fluid interfaces, mesh refinements similar to the prism layer of the body fitted case are applied. The mesh has 4 785 326 cells. The domain and an example of the material distribution can be seen in Fig. 6.30 with a detail of the mesh refinement. The black regions ($\rho_{des} = 0$) represent solid and the white regions ($\rho_{des} = 1$) represent fluid. This color code is used for the other numerical examples also.

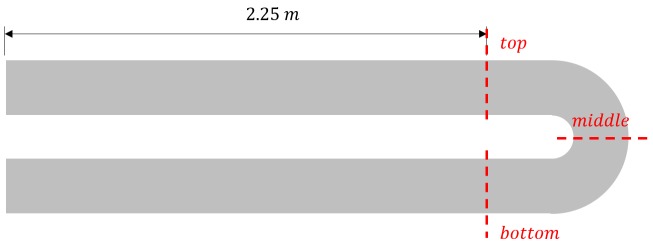
Figure 6.30 – Domain and geometry representation using material model.

(a) Domain dimensions. **(b)** Representation and mesh detail.



The state variables are compared in the three sections (top, middle and bottom) exhibited in Fig. 6.31. A mesh convergence verification is performed for the values of the objective function in the body fitted cases by using the Grid Convergence Index (GCI) described in Celik et al. (2008) and explained in App. B.

Figure 6.31 – Sections to compare state variables.



The boundary condition definitions, the flow properties and the boundary condition values are presented in Fig. 6.32 and Tab. 6.2.

Figure 6.32 – BCs for the compressible case.

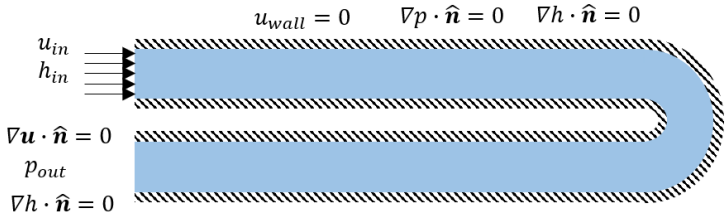


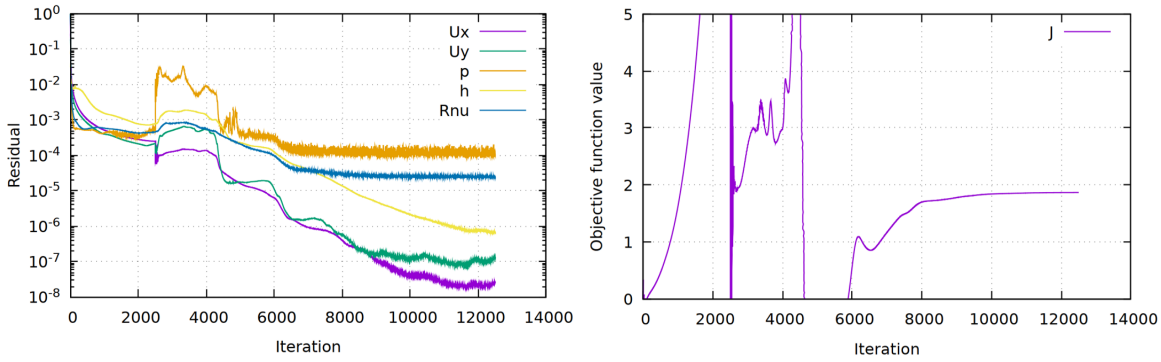
Table 6.2 – BCs and objective function for the compressible case.

Property	Value
$\mu\ (Pa \cdot s)$	1.8×10^{-5}
$u_{in}\ (m/s)$	125
$p_{out}\ (Pa)$	101325
$h_{in}\ (J/kg)$	$2.95 \cdot 10^5$
$c_p\ (J/kg)$	1005
Pr	0.705
Objective function	Entropy variation (Eq. 3.9)

Three meshes for the body fitted case are built, with 57 360, 136 266 and 322 958 cells. Close to the walls, a prismatic layer with 20 layers and a growth ratio of 1.2 are used. The flow problem is solved with more than 10 000 iterations of the algorithm described in 5.1 to achieve low residuals and a constant objective function (see Figs. 6.33a and 6.33). The entropy variation calculated is 1.87 W/K $\pm$ 0.9%.

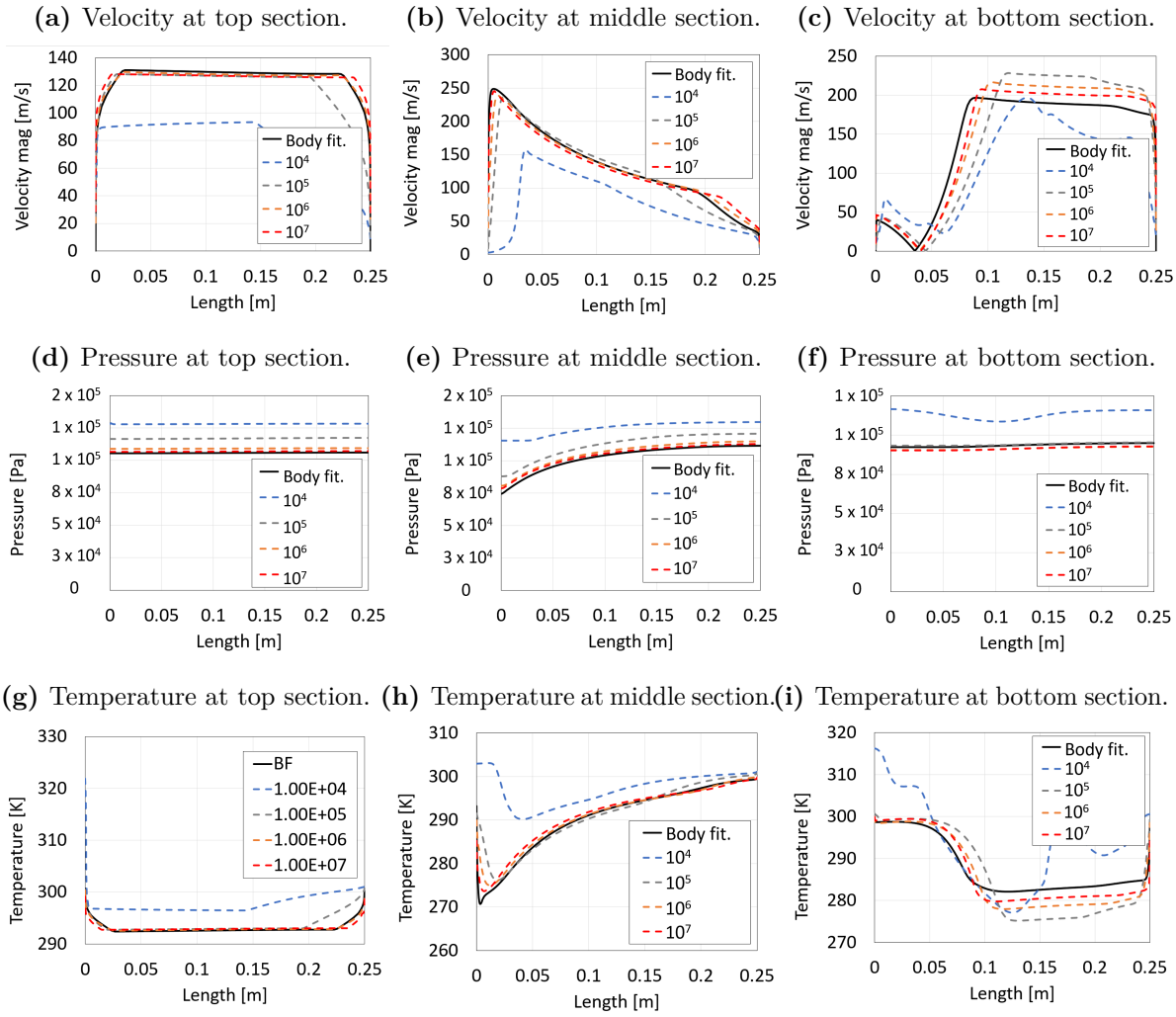
Figure 6.33 – Residuals and objective function for the compressible body fitted case.

(a) Residuals. **(b)** Entropy variation.



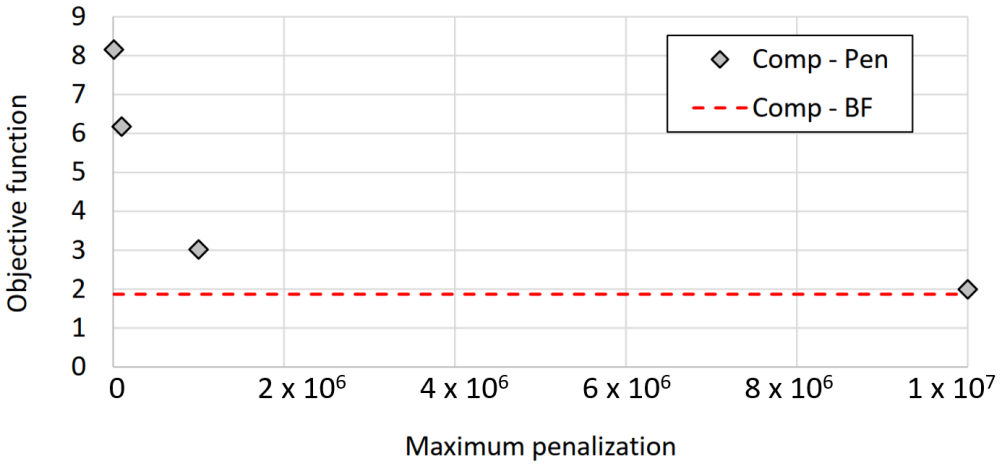
The state variables are compared in the three sections from Fig. 6.31. Four values for the maximum penalization are tested: 10^4, 10^5, 10^6 and 10^7. For each penalization, the flow problem is solved with more than 30 000 iterations to achieve low residuals and a constant objective function. The velocity, pressure and temperature values in each section can be seen in Fig. 6.34. The black solid line is the profile extracted from the reference case and the colored dashed lines are the values from the penalized cases.

Figure 6.34 – Velocity, pressure and temperature profiles for the compressible flow.



In all cases, the velocity, pressure and temperature profiles exhibit a shape similar to the reference case. For the lowest penalization, the differences in the magnitudes can be expressive. As an example, the temperature profile for the middle section (Fig. 6.34h) starts around 280 K for the reference case while, for a penalization of 10^4, it is around 300 K. As the penalization increases, the profiles approximate to the reference, meaning that the flow exhibits a behavior closer to the one it must represent. The objective function values also follow this behavior (see Fig. 6.35). For $\overline{\alpha} = 10^4$, the entropy variation is 8.16 W/K (336% higher than the reference) and, for $\overline{\alpha} = 10^7$, it is 2.00 W/K, only 7% higher than the reference value. Thus, for this case, the Brinkman penalization values necessary to have an accurate flow representation are around 10^7.

Figure 6.35 – Entropy variation for different penalizations.



To better understand this behavior, the turbulent viscosity fields for the body fitted case, $\overline{\alpha} = 10^7$ and $\overline{\alpha} = 10^5$ are presented in Fig. 6.36. The white line identifies the interface established by the material model. As can be seen, the turbulent viscosity field for the lowest value of penalization has higher values and these cover a bigger area. The two white arrows indicate regions where the turbulent viscosity is almost not noticeable for the body fitted and the $\overline{\alpha} = 10^7$ cases. However, for the $\overline{\alpha} = 10^5$ case, these regions already have a growing layer where turbulent viscosity is noticeable. These higher amount of turbulent viscosity present in the less penalized case is the main cause of the higher entropy variation observed. To understand why they happen, it is interesting to take a closer look at the regions highlighted by the white dashed rectangles. In Fig. 6.37, these regions are represented with the mesh cells and a scale of maximum velocity of 3 m/s. It is possible to see that, in the less penalized case, a layer of two cells has a velocity magnitude higher than zero. Even being a small portion, these layers where a solid region should be represented (but the velocity is not reduced to negligible values) act as a porous layer and increase the turbulent viscosity. Since the flow travels for a long path of solid-fluid interface with this effect, it becomes cumulative and the distribution of turbulent viscosity affects the whole flow. In the most penalized case, this effect is not observed, so the results are closer to the reference case.

Figure 6.36 – Turbulent viscosity for body fitted and penalized cases (compressible).

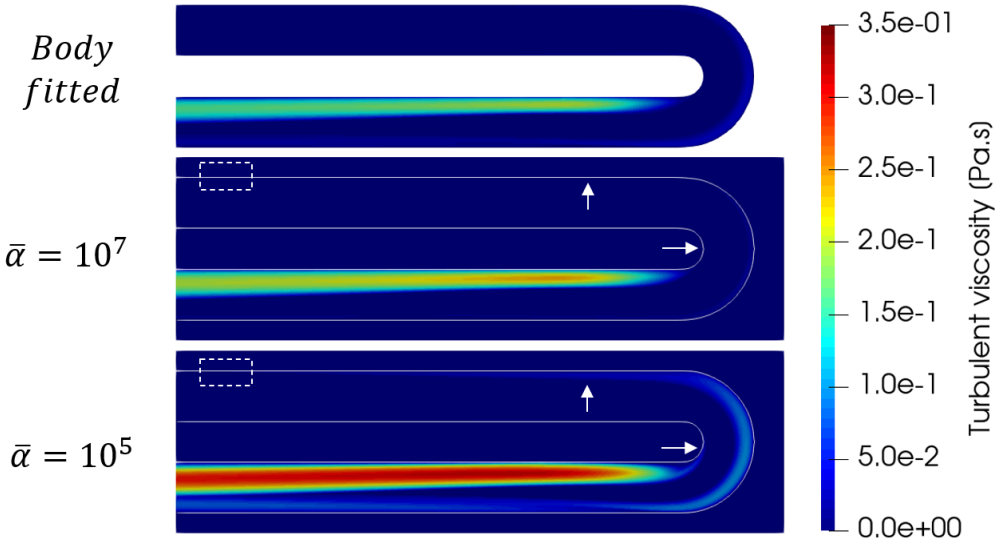
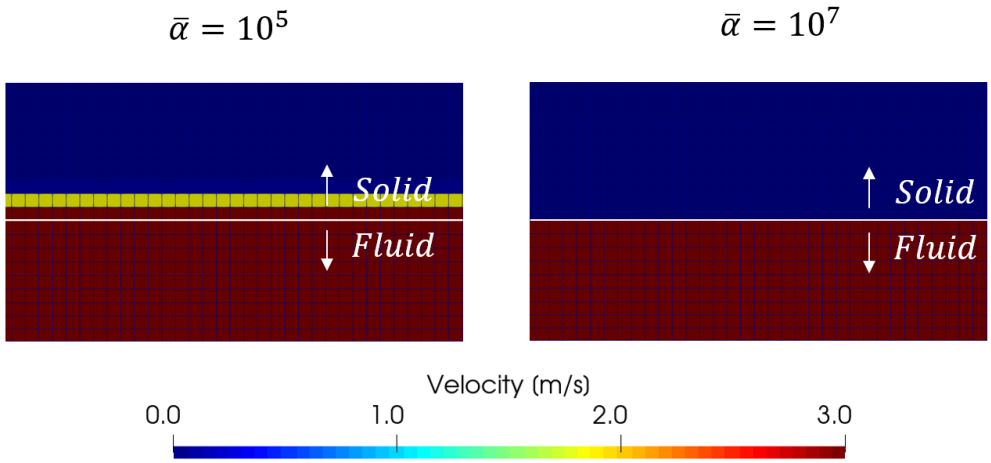


Figure 6.37 – Velocity at solid-fluid interface (compressible).

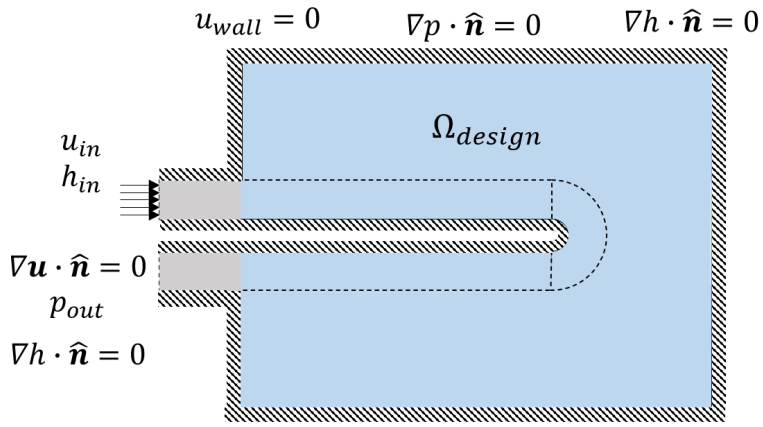


6.9 Compressible Turbulent U-Channel Optimization

The case explored previously is now considered as a baseline for an optimization example. The baseline represents an intuitive design, which is also the shortest path connecting the inlet to the outlet in the domain defined for the optimization. It is expected to have a low entropy variation when compared to other possible designs. In the optimization, a volume constraint is imposed, so the optimized design and the baseline have the same fluid volume and a fair comparison is made. For the same volume of fluid, longer channels must be narrower, so higher losses and, consequently, higher entropy variations can be expected. With all these conditions, this problem establishes an interesting challenge where a balanced solution must be found by the optimizer. The objective is to verify the

possibility of finding a design with the same fluid volume connecting the inlet to the outlet with lower entropy variation than the example showed in the previous numerical case. The domain and boundary conditions can be seen in Fig. 6.38 and the BC values, viscosity and objective function are presented in Tab. 6.2.

Figure 6.38 – Boundary conditions and design domain for compressible U-channel.

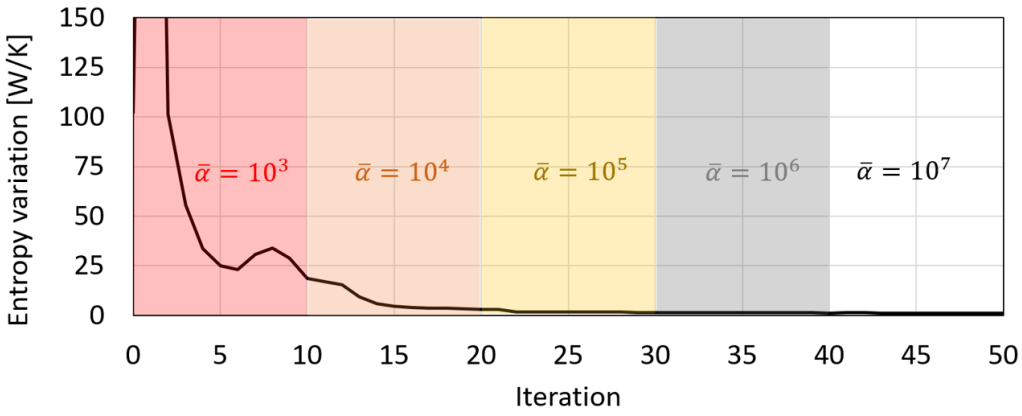


A volume fraction V_{frac} equals to 19% of the design domain is used. To be able to reach a steady-state condition in the first optimization iterations, an initial guess with a uniform value of 0.19 for the design variables is considered (see Fig. 6.39). A continuation strategy is used to start with a low porosity value and achieve a nice representation of solid regions at the end. The variable $\overline{\alpha}$ starts with $\overline{\alpha} = 10^3$, the optimization runs for 10 iterations or until ϵ_J is smaller than 10^{-4}. After, $\overline{\alpha}$ is changed to 10^4, 10^5, 10^6 and 10^7 with the same definitions for the maximum number of optimization iterations and tolerance. Between each optimization iteration, a total of 2 500 iterations of flow calculation are used. The final design results in an entropy variation of 1.15 W/K with a volume fraction of approximately 19.3% and can be seen in Fig. 6.40. The objective function evolution during the optimization can be seen in Fig. 6.41 with all the changes in the maximum value used for the penalization.

Figure 6.39 – Initial guess.

Figure 6.40 – Optimized channel.

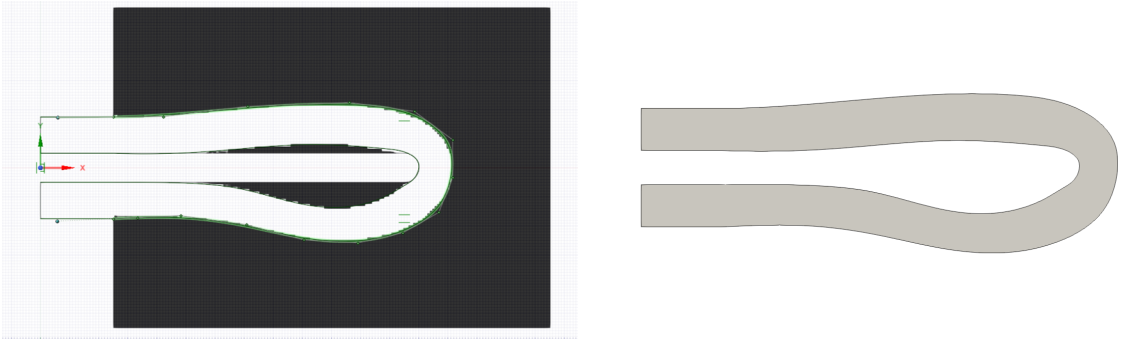
Figure 6.41 – Entropy variation during optimization.



As can be seen, the final solution has no intermediary values, so a well defined solid-fluid interface can be identified. The design is composed by a channel slightly narrower than the baseline and travels through a longer path. The optimized channel has a smooth curve next to the inlet, then a turn with a smaller radius conducts the flow to the outlet. The flow direction is changed in a smoother manner than in the baseline, so it can represent a better option. During the optimization, the highest value observed for the entropy variation happens in the very first iterations due to the porous region acting as an obstacle for the flow. After approximately 5 iterations, a path is opened and the objective function drops considerably. With a maximum penalization of 10^4, the objective function continues to drop, as the channel is being established. Then, in the next iterations, only small changes can be seen. To assess the optimized design and compare it to the baseline, the final geometry is post processed with a CAD software and extracted by manually adjusting splines (see Fig. 6.42).

Figure 6.42 – CAD post processing and resulting channel.

(a) CAD post processing with splines. **(b)** Optimized channel.



The same boundary conditions are used and the solid-fluid interfaces obtained in the optimization are now defined as no slip walls. Three meshes are built, with 11 084, 45 830 and 108 276 cells. The grid convergence assessment explained in Appendix B is used. For the finer mesh, the residuals for velocity, pressure, temperature and the turbulence model can be seen in Fig. 6.43a. For the same mesh, the objective function evolution during the flow calculation is presented in Fig. 6.43b. As can be seen, after 10 000 iterations, the residuals are at a low level (smaller than 10^{-6}) and the objective function presents a steady behavior, so the solution can be considered converged. The entropy variation obtained is $J_{comp} = 0.69$ W/K $\pm$ 4.6%, representing approximately 63% of reduction when compared to the baseline. The velocity, pressure and temperature contours for the optimization result (with material model) and for the post processed geometry can be seen in Fig. 6.44, while the Mach number and density fields are presented in Fig. 6.45.

Figure 6.43 – Residuals and objective function for optimized channel.

(a) Residuals for the final geometry. **(b)** Objective function for the final geometry.

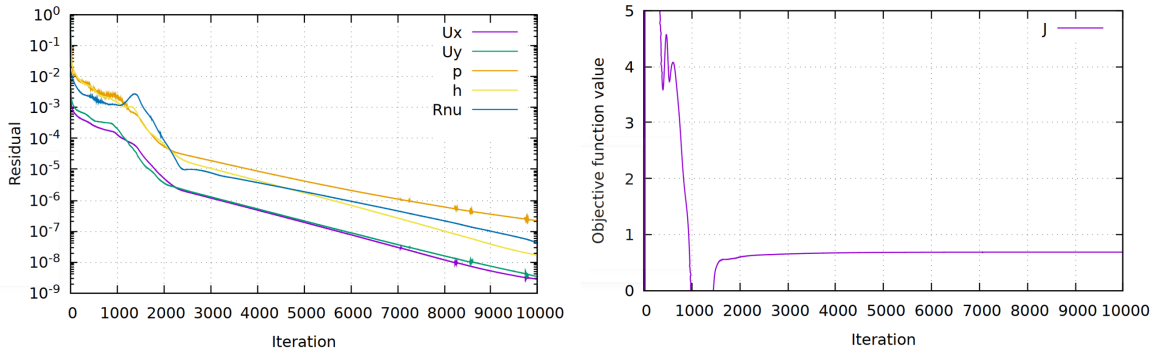
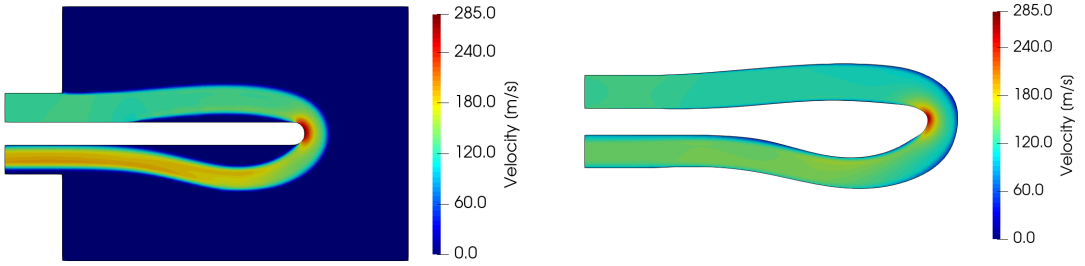
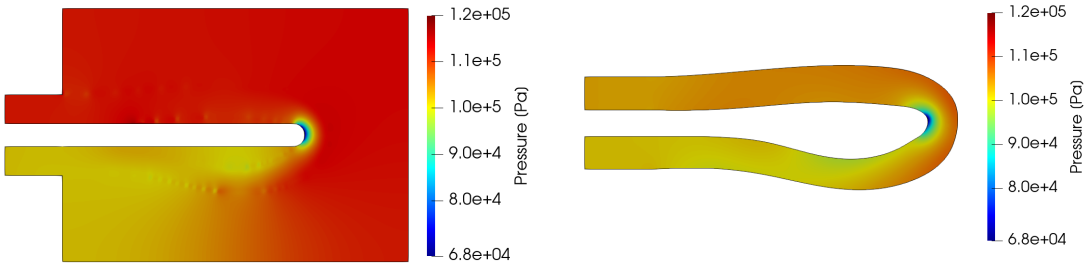


Figure 6.44 – Velocity, pressure and temperature contours for the compressible U-channel.

(a) Velocity (optimization). **(b)** Velocity (final geometry).



(c) Pressure (optimization). **(d)** Pressure (final geometry).



(e) Temperature (optimization). **(f)** Temperature (final geometry).

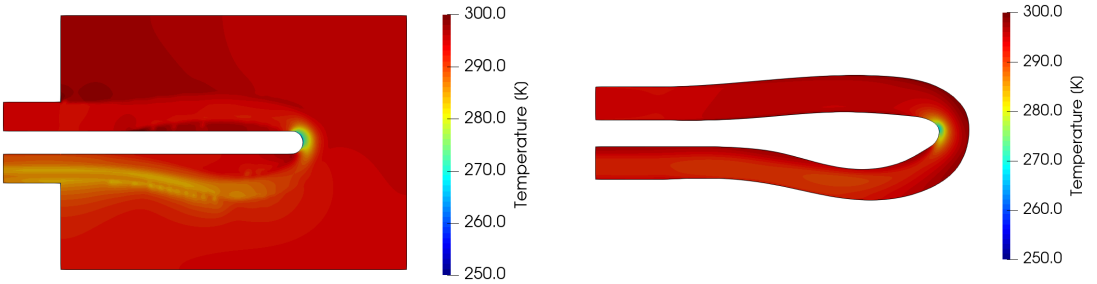
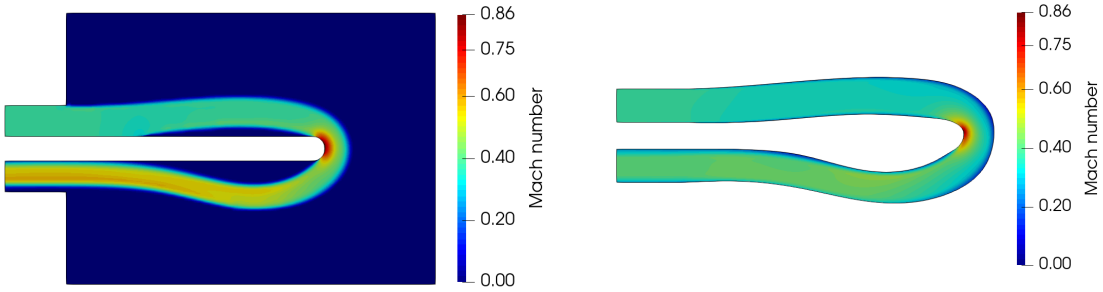
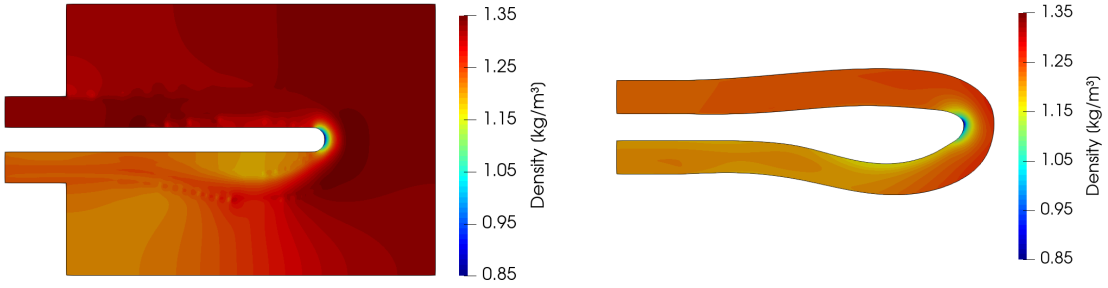


Figure 6.45 – Mach number and density contours for the compressible U-channel.

(a) Mach number (optimization). **(b)** Mach number (final geometry).



(c) Density (optimization). **(d)** Density (final geometry).



The flows in the optimization result and in the post processed geometry are not exactly the same and the differences arise mainly from the fact that the mesh definitions used in both cases are different. The mesh from the optimization case is too coarse to accurately represent the flow inside this geometry. Since the mesh requirements are not known a priori in the optimization cases, an accurate representation is not always possible. However, the model used in the optimization is able to capture the main flow features and results in an optimized design as is proved by the results from the interpreted geometry. Thus, the methodology is capable of designing non-rotating components working with compressible flow by topology optimization.

6.10 Compressible Turbulent Rotor Optimization

In this final example, topology optimization is used in a rotating compressible flow example considering a turbulent flow. The objective is to achieve a rotor with high isentropic efficiency (Eq. 3.10). The domain and design area used are the same from the example in Sec. 6.7 (see Fig. 6.24a). The boundary conditions, flow properties and initial guess (Fig. 6.24b) are also from Sec. 6.7, except for the the dynamic viscosity μ which, in this case, is 1.8×10^{-5} Pa $\cdot$ s.

A continuation strategy is used for the viscosity since the initial guess does not result

in a steady-state flow. The initial value used for the viscosity is 1.0×10^{-2} Pa $\cdot$ s. The optimization runs with a tolerance ϵ_J of 10^{-4} or a maximum of 25 iterations. After, the viscosity is changed to 5.0×10^{-3} Pa $\cdot$ s with the same tolerance and maximum number of iterations. Then, reduced again to 1.0×10^{-3} Pa $\cdot$ s and this process is repeated until the final value is achieved. Continuation is also used for $\overline{\alpha}$. From Sec. 6.8, a value of 10^7 is observed to generate results in good agreement with the body fitted mesh for compressible flows. However, a value of 10^6 is used initially, since it results in a steady-state behavior. The final value for $\overline{\alpha}$ is used at the same time the final value for the viscosity is used. After the optimization runs with the final values for the viscosity and the maximum penalization, the adaptive mesh refinement is enabled to act at each 10 optimization iterations and the optimization runs for a maximum of 150 iterations or until ϵ_J is smaller than 10^{-5}. Before the AMR, 5 000 iterations for the flow calculation are used between each optimization iteration. When the AMR is enabled, the flow is calculated by using 10 000 iterations between each optimization iteration. The final design results in an isentropic efficiency of 64.4% and can be seen in Fig. 6.46 with the respective full rotor. The initial mesh and the mesh after the refinement are presented in Fig. 6.47. The objective function value for each optimization iteration can be seen in Fig. 6.48 with respective values for the viscosity and maximum penalization.

Figure 6.46 – Optimized rotor for compressible flow.

(a) Optimized design. **(b)** Full rotor from the optimized design.

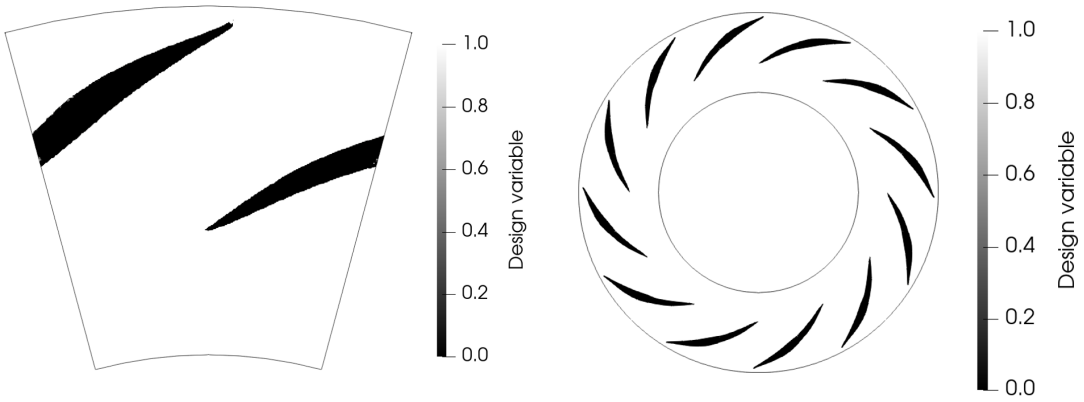


Figure 6.47 – Initial and final meshes.

(a) Initial mesh (23 926 cells). **(b)** Final mesh (38 465 cells).

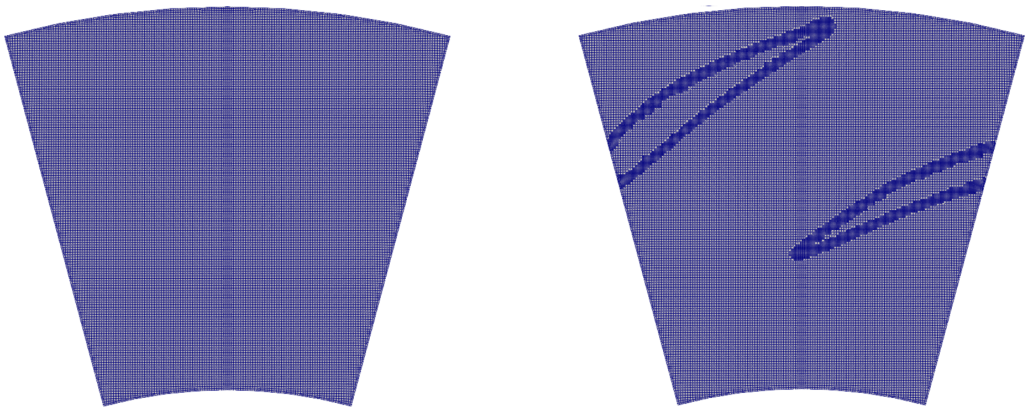
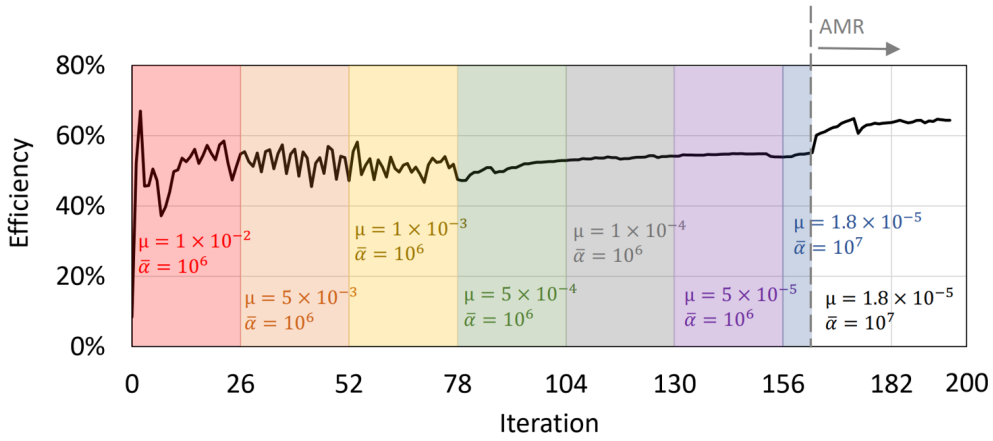


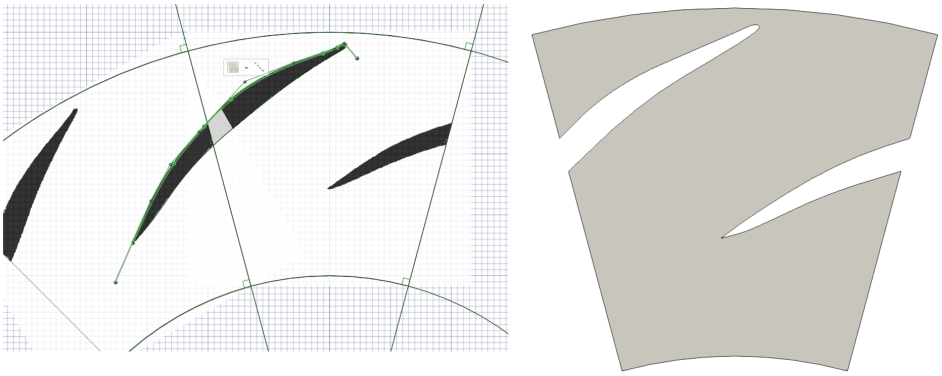
Figure 6.48 – Impeller isentropic efficiency during the optimization.



During the first steps of the optimization, it is possible to see that the objective function oscillates, but does not present any real improvement. This indicates a changing behavior of the geometry that still does not result in a steady condition. After μ reaches the value of 5×10^{-4} Pa $\cdot$ s and $\overline{\alpha}$ is 10^6, the objective function stabilizes and starts to show some slow improvements. When the AMR is applied and the final values for μ and $\overline{\alpha}$ are used, a more evident improvement can be noticed until it reaches its final value. The final geometry is extracted by using a CAD software (Fig. 6.49) and an assessment in a body fitted mesh is performed.

Figure 6.49 – CAD post processing and geometry.

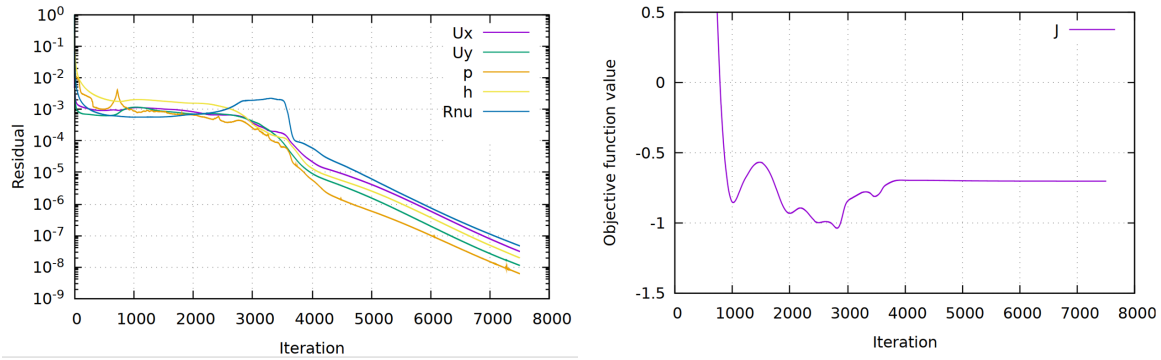
(a) CAD post processing with splines. **(b)** Optimized geometry.



Three meshes are built (22 778, 78 427 and 181 278 cells) and assessed with the GCI (Appendix B. The residuals and objective functions for 7 500 iterations of flow calculations can be seen in Figs. 6.50a and 6.50b, indicating a converged state.

Figure 6.50 – Residuals and objective function.

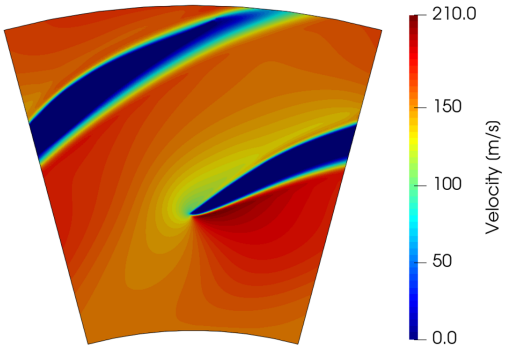
(a) Residuals for the final geometry. **(b)** Objective function for the final geometry.



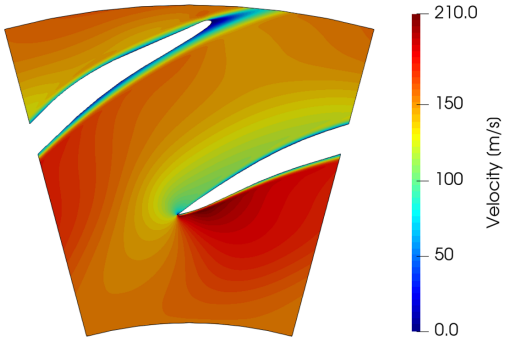
The final value for the isentropic efficiency obtained is 70.3% $\pm$ 0.001%, being approximately 5.9% higher than the value obtained in the optimization. The velocity, pressure and temperature contours of the final stage of the optimization and the body fitted geometry are presented in Fig. 6.51. The Mach number and density contours can be seen in Fig. 6.52. The flow fields and, consequently, the objective functions are very similar and show that, in this example, the strategy using AMR results in an accurate flow and an optimized design.

Figure 6.51 – Velocity, pressure and temperature contours (compressible rotating case).

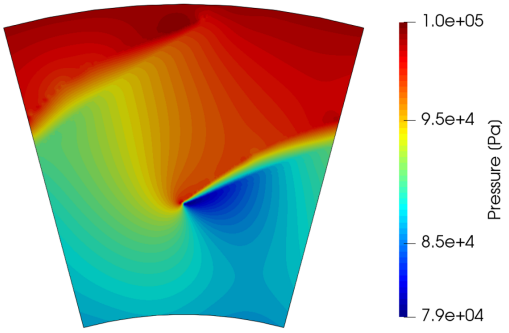
(a) Velocity (optimization).



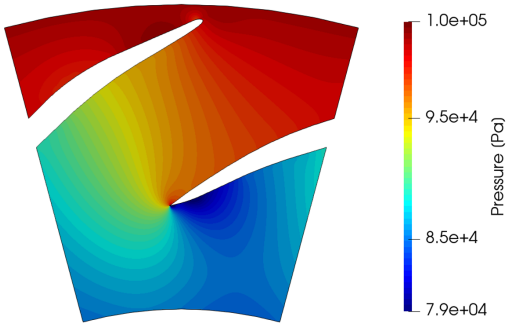
(b) Velocity (final geometry).



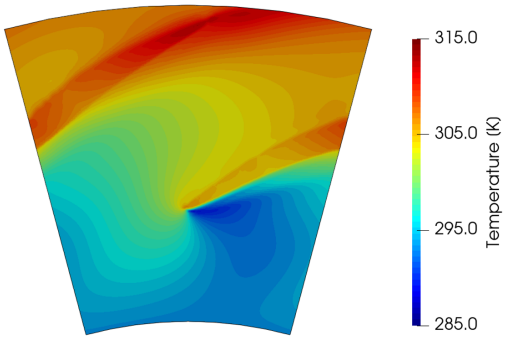
(c) Pressure (optimization).



(d) Pressure (final geometry).



(e) Temperature (optimization).



(f) Temperature (final geometry).

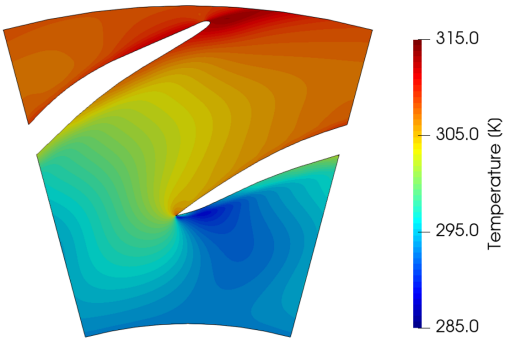
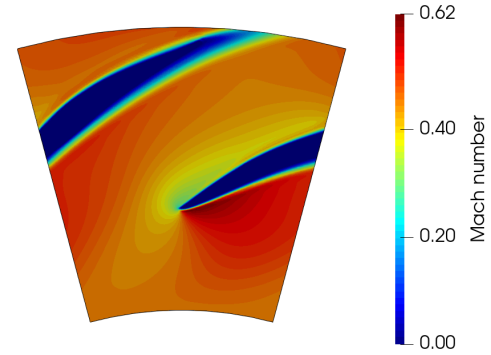
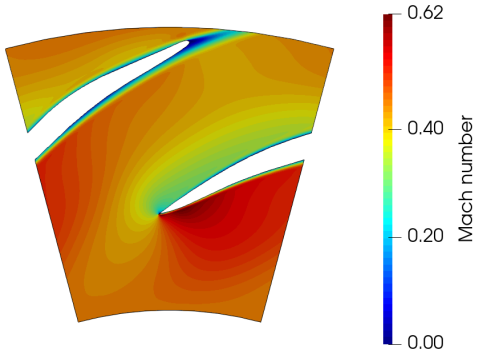


Figure 6.52 – Mach number and density (compressible rotating case).

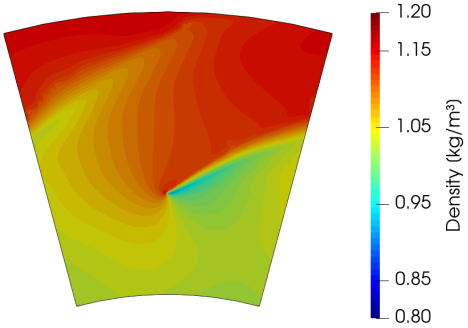
(a) Mach number (optimization).



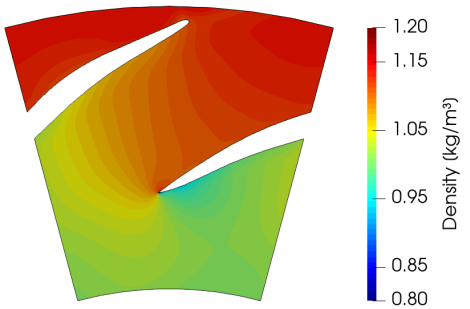
(b) Mach number (final geometry).



(c) Density (optimization).



(d) Density (final geometry).



7 CONCLUSIONS

In this work, topology optimization is applied to compressible flows with and without rotation. The Brinkman penalization is used to represent the geometry inside the domain. A material model focused on representing adiabatic walls is studied and applied. The sensitivities for the optimization problem are calculated by using the adjoint method. The continuous version was the initial strategy and its development for incompressible laminar flows has been done and worked well. However, using the continuous approach in compressible flow problems proved to be very challenging since the system is highly non-linear and a new path had to be explored. The discrete version of the adjoint method has been applied in a novel strategy based on finite differences. It is flexible and has been motivated by the difficulties found when deriving the adjoint system for compressible flows in the continuous approach. The flexibility of the developed approach allowed it to be applied to incompressible and compressible flow cases. A sensitivity verification by finite differences shows that the implementations are working correctly for the laminar cases and that the correct tendencies are obtained for the compressible turbulent cases. Ten numerical examples are presented.

In the 3D rotor working with incompressible laminar flow, an optimized design is obtained by topology optimization. The result is post processed and the final geometry is compared to a traditional straight bladed design. The optimized design proved to be approximately 9.52% better than the traditional one. The optimization designed a geometry with no recirculation areas, resulting in a well behaved flow with better objective function than the straight bladed design. The resulting paths have a twisted non intuitive geometry, inclined to the opposite direction of the rotation. This shows that the flow inside this kind of device can have complicated trajectories, so finding an optimized geometry is not trivial and topology optimization can be very beneficial to the design process. In this case, the continuous adjoint has been used and worked well. Since the application of the continuous approach for compressible flows was not possible, in all other examples, the discrete approach developed in this work is used.

To assess the capabilities of the discrete adjoint developed, it is applied to the traditional double-channel case. The flow is laminar and incompressible and the objective is to verify if the methodology is able to reach well known results. The adaptive mesh refinement is used to improve the boundary definitions. The results obtained are in a good agreement with results published in the topology optimization literature, showing that the methodology can reach well established designs. The use of the AMR allowed to have a well refined mesh only in important regions of the design, changing during the optimization and correctly

following the boundaries. Beyond that, a comparison of the continuous and the discrete adjoint approaches for an incompressible and laminar pipe bend case is presented. Four different meshes are used. The designs obtained by both approaches are very similar. The continuous approach proved to be faster, as already expected. Also, as the mesh density increases, the differences in the computational cost between the discrete and the continuous approach increases. However, the value of the discrete approach is not in its computational demand, but in its flexibility. This example draws attention to the fact that focused work can be necessary to use the discrete approach to large-scale cases.

Before applying topology optimization to compressible flows, a representation of adiabatic walls has been studied. The example considering a pipe bend with compressible laminar flow demonstrates a comparison between a body fitted mesh and the same geometry represented by the material model. The values of velocity, pressure and temperature for both models (body fitted and material model) are compared at a diagonal line crossing the domain. By using high values of penalization only at the momentum equations, it is possible to have a representation that mimics adiabatic walls. This strategy is used in all the other examples.

In the convergent channel example, compressible laminar flow is considered and the objective is to compare different cell topologies. Three different cells are used: quadrilateral, triangular and polyhedral. In finite volume models, it is common to have different cell topologies in the same mesh. This example proves that the methodology is able to work with different cells, reaching the same design at the end of the optimization.

In the 3D compressible flow case, laminar regime is considered and the idea is to verify the discrete adjoint developed working in a 3D problem. A baseline case is defined with a design that can be considered intuitive and a volume constraint is applied with the objective of finding a better design with the same volume of fluid. The optimized design is approximately 22.7% better than the baseline. This case shows that the methodology can be extended to 3D problems.

Before applying the methodology to compressible turbulent and rotating flows, a laminar rotating case is presented. An initial guess of a straight and backward blade is used. The final design is composed of a thin blade with a small body after the trailing edge. The isentropic efficiency is increased in approximately 5%. From this example, it is possible to conclude that the methodology is able to optimize compressible laminar rotating flows.

To better understand the behavior of the penalization when a turbulence model is present, a U shaped channel case is analyzed considering compressible turbulent flow. The objective is to find penalization values that result in an accurate representation of the flow inside the domain. A case with a body fitted mesh is established as a reference and

its geometry is reproduced by porosity fields inside the test cases. The results show that high values of penalization are necessary (in the order of 10^7) to have an accurate flow representation, resulting in similar flow fields between the reference and penalized cases. For low penalization values, the turbulent viscosity can be very high, changing the flow inside the domain (with respect to a flow in a body fitted case) causing big deviations in the objective function. This is a point of attention that is not present in the laminar flow cases, but one must be cautious when designing devices with turbulent flows, since the results can be very different if the penalization used is not high enough.

The U-channel studied is also used in an optimization case. This time, the flow is considered compressible and turbulent, the channel is defined as a baseline and the discrete adjoint strategy developed is applied to design a channel with less entropy variation. An initial guess considering a uniform porosity field is used and a continuation strategy is used for the maximum penalization value. The optimized channel is composed by a sequence of curves, starting very close to the inlet and leading the flow smoothly to the outlet. The interpreted geometry shows a reduction of approximately 63% of entropy variation when compared to the baseline case. This case shows that it is possible to find designs better than the most obvious one for compressible flows. Additionally, even though the fluid inside the optimization model is not accurate when compared to a body fitted case, it is capable of representing the main flow features and reaching an optimized design.

Finally, a rotating domain with compressible turbulent flow is considered. The objective is to generate a design with a high isentropic efficiency. An initial guess representing an inclined straight blade is used to have a well behaved flow in the first optimization iterations. In this case, continuation is used in the viscosity and maximum penalization values. The adaptive mesh refinement is used to improve the boundary definitions. The final design achieves an efficiency of approximately 70.3% and the flow inside the optimization model closely represents the post processed geometry. This example shows that the methodology is able to optimize compressible turbulent flows with rotation.

The numerical examples clearly show that the methodologies using the continuous and the discrete adjoint work and can be used to design optimized geometries for incompressible and compressible flows with and without rotation. The objective functions presented are energy dissipation, entropy variation, pump efficiency and isentropic efficiency, but the methodology can work other functions as well, requiring small changes in the adjoint derivations.

In some numerical examples, continuation strategies proved to be necessary to have steady-state flows in the first optimization iterations. If the flow is not well represented by a steady-state modeling in the first iterations, the adjoint problem becomes very difficult to solve, leading to useless results or even diverging and stopping the optimization procedure.

The objectives established initially, in Sec. 1.6, have been achieved. All the theoretical explanations are described in Chapters 3, 4 and 5 and exemplified by the content of Chapter 6. The contributions of this work can be summarized in two main items: the application of topology optimization to compressible flow problems and the development of topology optimization using a finite differences strategy for the adjoint system. The first item enlarged the area of application of topology optimization. Thus, new problems can be approached and new applications can arise. The second item represents an innovation that can helps in problems with complex physics. Since the finite differences strategy is independent of how hard it is to solve the physical problem, it represents a new option to be considered when developing topology optimization where it has not been applied yet.

7.1 Future Work

The work developed so far considers steady-state flows and the turbulence modeling are based on RANS and FANS approaches. The frozen turbulence is used when calculating the sensitivities. Also, perfect gas is considered when working with compressible flows. So, possible paths to explore and augment the capabilities of the methodology developed are:

a) Consider the turbulence models in the adjoint derivation. This will increase the accuracy of sensitivity calculations and better optimized designs can be found;

b) Develop capabilities to apply the methodology to problems focused on heat transfer (e.g.: heat sink device design). Problems with flow exchange between fluids and solids can be challenging and this approach may help to improve the existing strategies;

c) Develop strategies to account for solid region design during the optimization (fluid-structure interaction). This area is still poorly explored in topology optimization. The physical problems can be complicated, so the strategy using finite differences may simplify the development of this application;

d) Expand the methodology to transient flows. Transient problems are common in fluid dynamics, so expanding the capabilities of the approach developed in this work can enlarge the area of application of topology optimization and help finding new optimized designs for different problems;

e) Use other turbulence approaches, such as LES (large eddy simulation) and DES (detached eddy simulation);

f) Use other gas models. Problems where real gas properties are required are not unusual. Hence, improving the methodology developed here to account for other gas models can allow the use of topology optimization in different problems of practical applications;

g) Improve the optimizer. The steepest descent is used in this work. Even though the results show that the optimization is working well, improvements in the optimizer can help finding better minima and also reduce the time necessary in the optimization cases.

Additionally, the assemble of the adjoint system for the discrete approach is still a time-consuming step, prohibitive for 3D cases. Thus, to improve the methodology and allow it to work with large-scale cases, developments in the assemble and solution of the adjoint system are necessary and can be interesting options for further research.

REFERENCES

ALONSO, D. H.; RODRIGUEZ, L. F. G.; SILVA, E. C. N. Flexible framework for fluid topology optimization with openfoam and finite element-based high-level discrete adjoint method (fenics/dolfin-adjoint). **Structural and Multidisciplinary Optimization**, v. 64, 2021. Cited in page 30.

ALONSO, D. H.; SÁ, L. F. N.; SAENZ, J. S. R.; SILVA, E. C. N. Topology optimization applied to the design of 2d swirl flow devices. **Structural and Multidisciplinary Optimization**, v. 58, 2018. Cited 2 times in pages 27 and 30.

ALONSO, D. H.; SÁ, L. F. N. de; SAENZ, J. S. R.; SILVA, E. C. N. Topology optimization based on a two-dimensional swirl flow model of tesla-type pump devices. **Computers & Mathematics with Applications**, Elsevier, v. 77, n. 9, p. 2499–2533, 2019. Cited in page 27.

ALONSO, D. H.; SAENZ, J. S. R.; PICELLI, R.; SILVA, E. C. N. Topology optimization method based on the wray-agarwal turbulence model. **Structural and Multidisciplinary Optimization**, v. 65, 2022. Cited in page 27.

ALONSO, D. H.; SAENZ, J. S. R.; SILVA, E. C. N. Non-newtonian laminar 2d swirl flow design by the topology optimization method. **Structural and Multidisciplinary Optimization**, v. 62, 2020. Cited in page 27.

ALONSO, D. H.; SILVA, E. C. N. Topology optimization for blood flow considering a hemolysis model. **Structural and Multidisciplinary Optimization**, v. 63, 2021. Cited in page 27.

ALONSO, D. H.; SILVA, E. C. N. Topology optimization applied to the design of tesla-type turbine devices. **Applied Mathematical Modelling**, v. 103, 2022. Cited in page 27.

AMELI, A.; AFZALIFAR, A.; TURUNEN-SAARESTI, T.; BACKMAN, J. Effects of real gas model accuracy and operating conditions on supercritical co2 compressor performance and flow field. **Journal of Engineering for Gas Turbines and Power**, v. 140, 2018. Cited in page 20.

AMELI, A.; TURUNEN-SAARESTI, T.; BACKMAN, J. Numerical investigation of the flow behavior inside a supercritical co2 centrifugal compressor. **Journal of Engineering for Gas Turbines and Power**, v. 140, 2018. Cited 2 times in pages 17 and 20.

ANDERSON, J. Modern compressible flow with historical perspective, mc graw hill. series in aeronautical and aerospace engineering. 2003. Cited in page 18.

BOIRON, O.; CHIAVASSA, G.; DONAT, R. A high-resolution penalization method for large mach number flows in the presence of obstacles. **Computers & fluids**, Elsevier, v. 38, n. 3, p. 703–714, 2009. Cited in page 49.

BORGNAKKE, C.; SONNTAG, R. E. **Fundamentals of Thermodynamics**. Wiley, 2008. ISBN 9780470041925. Available from Internet: <https://books.google.com.br/books?id=7GRDPgAACAAJ>. Visited on: 02 Jan. 2018. Cited in page 48.

BORRVALL, T.; PETERSSON, J. Topology optimization of fluids in stokes flow. **International journal for numerical methods in fluids**, Wiley Online Library, v. 41, n. 1, p. 77–107, 2003. Cited 6 times in pages 25, 26, 27, 29, 48, and 86.

BROWN-DYMKOSKI, E.; KASIMOV, N.; VASILYEV, O. V. A characteristic based volume penalization method for general evolution problems applied to compressible viscous flows. **Journal of Computational Physics**, Elsevier, v. 262, p. 344–357, 2014. Cited in page 50.

CADENCE. Fidelity cfd. 2022. Available from Internet: <https://www.numeca.com/omnis>. Visited on: 21 Dez. 2021. Cited in page 23.

CELIK, I. B.; GHIA, U.; ROACHE, P. J.; FREITAS, C. J.; COLEMAN, H.; RAAD, P. E. Procedure for estimation and reporting of uncertainty due to discretization in cfd applications. **Journal of Fluids Engineering, Transactions of the ASME**, v. 130, 2008. Cited 2 times in pages 100 and 131.

CHALLIS, V. J.; GUEST, J. K. Level set topology optimization of fluids in Stokes flow. **International Journal for Numerical Methods in Engineering**, v. 79, n. 10, p. 1284–1308, sep 2009. Available from Internet: <http://onlinelibrary.wiley.com/doi/10.1002/nme.3279/fullhttp://doi.wiley.com/10.1002/nme.2616>. Visited on: 20 Jan. 2018. Cited in page 86.

DEMEULENAERE, A.; BONACCORSI, J. C.; GUTZWILLER, D.; HU, L.; SUN, H. Multi-disciplinary multi-point optimization of a turbocharger compressor wheel. In: . [S.l.: s.n.], 2015. v. 2C. Cited in page 22.

DEWAR, B.; TIAINEN, J.; JAATINEN-VÃ¤RRI, A.; CREAMER, M.; DOTCHEVA, M.; RADULOVIC, J.; BUICK, J. M. Cfd modelling of a centrifugal compressor with experimental validation through radial diffuser static pressure measurement. **International Journal of Rotating Machinery**, v. 2019, 2019. Cited in page 20.

DILGEN, C. B.; DILGEN, S. B.; FUHRMAN, D. R.; SIGMUND, O.; LAZAROV, B. S. Topology optimization of turbulent flows. **Computer Methods in Applied Mechanics and Engineering**, Elsevier, v. 331, p. 363–393, 2018. Cited 2 times in pages 27 and 30.

DILGEN, S. B.; DILGEN, C. B.; FUHRMAN, D. R.; SIGMUND, O.; LAZAROV, B. S. Density based topology optimization of turbulent flow heat transfer systems. **Structural and Multidisciplinary Optimization**, Springer, v. 57, n. 5, p. 1905–1918, 2018. Cited 2 times in pages 27 and 30.

DIXON, S.; HALL, C. **Fluid Mechanics and Thermodynamics of Turbomachinery**. Sixth edition. Boston: Butterworth-Heinemann, 2010. ISBN 978-1-85617-793-1. Cited 2 times in pages 18 and 48.

DWIGHT, R. P.; BREZILLON, J. Effect of approximations of the discrete adjoint on gradient-based optimization. **AIAA journal**, v. 44, n. 12, p. 3022–3031, 2006. Cited in page 30.

ECONOMON, T. D.; PALACIOS, F.; ALONSO, J. J. A viscous continuous adjoint approach for the design of rotating engineering applications. In: **21st AIAA computational fluid dynamics conference**. [S.l.: s.n.], 2013. p. 2580. Cited in page 22.

ECONOMON, T. D.; PALACIOS, F.; ALONSO, J. J. Unsteady continuous adjoint approach for aerodynamic design on dynamic meshes. **AIAA Journal**, American Institute of Aeronautics and Astronautics, v. 53, n. 9, p. 2437–2453, 2015. Cited in page 30.

EIGEN. Eigen 3.3.9. 2020. Available from Internet: <https://gitlab.com/libeigen/eigen/-/releases/3.3.9>. Visited on: 08 Nov. 2020. Cited in page 68.

EKRADI, K.; MADADI, A. Performance improvement of a transonic centrifugal compressor impeller with splitter blade by three-dimensional optimization. **Energy**, v. 201, 2020. Cited in page 23.

ESI. Opencfd release openfoam v2006 (20 06). 2020. Available from Internet: <https://www.openfoam.com/news/main-news/openfoam-v20-06>. Visited on: 04 Jan. 2021. Cited in page 67.

ESI. Solution and algorithm control. 2020. Available from Internet: <https://www.openfoam.com/documentation/user-guide/fvSolution.php>. Visited on: 22 Oct. 2020. Cited 2 times in pages 71 and 72.

ESI. adjointshapeoptimizationfoam.c file reference. 2022. Available from Internet: <https://www.openfoam.com/documentation/guides/latest/api/adjointShapeOptimizationFoam_8C.html>. Visited on: 04 Jan. 2022. Cited in page 68.

ESI. Linear upwind divergence scheme. 2022. Available from Internet: <https://www.openfoam.com/documentation/guides/latest/doc/guide-schemes-divergence-linear-upwind.html>. Visited on: 04 Jan. 2022. Cited in page 79.

ESI. rhosimplefoam. 2022. Available from Internet: <https://www.openfoam.com/documentation/guides/latest/doc/guide-applications-solvers-compressible-rhoSimpleFoam.html>. Visited on: 04 Jan. 2022. Cited in page 68.

ESI. simplefoam. 2022. Available from Internet: <https://www.openfoam.com/documentation/guides/latest/doc/guide-applications-solvers-incompressible-simpleFoam.html>. Visited on: 04 Jan. 2022. Cited in page 68.

FERZIGER, J. H.; PERIC, M.; STREET, R. L. **Computational Methods for Fluid Dynamics**. [S.l.: s.n.], 2020. Cited 6 times in pages 36, 37, 38, 43, 44, and 45.

GALINDO, J.; GIL, A.; NAVARRO, R.; TARI, D. Analysis of the impact of the geometry on the performance of an automotive centrifugal compressor using cfd simulations. **Applied Thermal Engineering**, v. 148, 2019. Cited in page 20.

GERSBORG-HANSEN, A.; SIGMUND, O.; HABER, R. B. Topology optimization of channel flow problems. **Structural and Multidisciplinary Optimization**, Springer, v. 30, n. 3, p. 181–192, 2005. Cited in page 26.

GHIAT, I.; AL-ANSARI, T. A review of carbon capture and utilisation as a co2abatement opportunity within the ewf nexus. **Journal of CO2 Utilization**, v. 45, 2021. Cited in page 16.

GILES, M. B.; PIERCE, N. A. An introduction to the adjoint approach to design. **Flow, turbulence and combustion**, Springer, v. 65, n. 3-4, p. 393–415, 2000. Cited in page 28.

GIRARDIN, C. A.; JENKINS, S.; SEDDON, N.; ALLEN, M.; LEWIS, S. L.; WHEELER, C. E.; GRISCOM, B. W.; MALHI, Y. Nature-based solutions can help cool the planet - if we act now. **Nature**, v. 593, 2021. Cited in page 16.

GÖLCÜ, M.; PANCAR, Y.; SEKMEN, Y. Energy saving in a deep well pump with splitter blade. **Energy Conversion and Management**, Elsevier, v. 47, n. 5, p. 638–651, 2006. Cited in page 21.

GUO, Z.; SONG, L.; ZHOU, Z.; LI, J.; FENG, Z. Multi-objective aerodynamic optimization design and data mining of a high pressure ratio centrifugal impeller. **Journal of Engineering for Gas Turbines and Power**, v. 137, 2015. Cited in page 51.

HAN, X.; RAHMAN, M. M.; AGARWAL, R. K. Development and application of a wall distance free wray-agarwal turbulence model (wa2018). In: . [S.l.: s.n.], 2018. Cited in page 41.

HAN, X.; WRAY, T. J.; FIOLA, C.; AGARWAL, R. K. Computation of flow in s ducts with wray-agarwal one-equation turbulence model. **Journal of Guidance, Control, and Dynamics**, v. 38, 2015. Cited in page 41.

HE, P.; MADER, C. A.; MARTINS, J. R. R. A.; MAKI, K. J. An aerodynamic design optimization framework using a discrete adjoint approach with openfoam. **Computers & Fluids**, v. 168, p. 285–303, 2018. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S0045793018302020>. Visited on: 12 Feb. 2020. Cited 4 times in pages 30, 31, 57, and 74.

HE, P.; MADER, C. A.; MARTINS, J. R. R. A.; MAKI, K. An object-oriented framework for rapid discrete adjoint development using openfoam. In: _____. **AIAA Scitech 2019 Forum**. [s.n.], 2019. Available from Internet: <https://arc.aiaa.org/doi/abs/10.2514/6.2019-1210>. Visited on: 15 Mar. 2020. Cited 2 times in pages 30 and 74.

HE, P.; MADER, C. A.; MARTINS, J. R. R. A.; MAKI, K. J. Dafoam: An open-source adjoint framework for multidisciplinary design optimization with openfoam. **AIAA Journal**, v. 58, n. 3, p. 1304–1319, 2020. Available from Internet: <https://doi.org/10.2514/1.J058853>. Visited on: 19 Dec. 2020. Cited 2 times in pages 30 and 74.

HEHN, A.; MOSDZIEN, M.; GRATES, D.; JESCHKE, P. Aerodynamic optimization of a transonic centrifugal compressor by using arbitrary blade surfaces. **Journal of Turbomachinery**, v. 140, 2018. Cited in page 23.

HEINRICH, M.; SCHWARZE, R. Simulation of the compressor stage of a turbocharger - validation of the open source library openfoam. In: . [S.l.: s.n.], 2013. v. 6 B. Cited in page 19.

HIRSCH, C. **Numerical computation of internal and external flows: The fundamentals of computational fluid dynamics**. [S.l.]: Elsevier, 2007. Cited in page 35.

HOGAN, R. J. Fast reverse-mode automatic differentiation using expression templates in c++. **ACM Transactions on Mathematical Software (TOMS)**, ACM New York, NY, USA, v. 40, n. 4, p. 1–16, 2014. Cited in page 30.

INSTITUTE, G. C. Ccs image library. 2022. Available from Internet: <https://www.globalccsinstitute.com/resources/ccs-image-library/>. Visited on: 08 Jun. 2022. Cited 2 times in pages 7 and 17.

INSTITUTE, G. C. What is ccs? 2022. Available from Internet: <https://www.globalccsinstitute.com/about/what-is-ccs/>. Visited on: 20 Jan. 2022. Cited in page 16.

JACKSON, S.; BRODAL, E. Optimization of the energy consumption of a carbon capture and sequestration related carbon dioxide compression processes. **Energies**, Multidisciplinary Digital Publishing Institute, v. 12, n. 9, p. 1603, 2019. Cited in page 17.

JASAK, H. Error analysis and estimation for the finite volume method with applications to fluid flows. [doctoral thesis, university of london]. **Imperial College of Science, Technology and Medicine**, M, 1996. Cited in page 68.

JASAK, H. Numerical solution algorithms for compressible flows: Lecture notes. **Faculty of Mechanical Engineering and Naval Architecture**, v. 2007, 2006. Cited in page 68.

JONES, W. P.; LAUNDER, B. E. The prediction of laminarization with a two-equation model of turbulence. **International Journal of Heat and Mass Transfer**, v. 15, 1972. Cited in page 41.

JU, Y. P.; ZHANG, C. H.; CHI, X. L. Optimization of centrifugal impellers for uniform discharge flow and wide operating range. **Journal of Propulsion and Power**, v. 28, 2012. Cited in page 51.

KARPOUZAS, G. K. A hybrid method for shape and topology optimization in fluid mechanics. 2019. Cited in page 29.

KENWAY, G. K. W.; MADER, C. A.; HE, P.; MARTINS, J. R. R. A. Effective adjoint approaches for computational fluid dynamics. **Progress in Aerospace Sciences**, v. 110, 2019. Cited 2 times in pages 28 and 74.

KIM, C.; LEE, H.; YANG, J.; SON, C.; HWANG, Y. Study on the performance of a centrifugal compressor considering running tip clearance. **International Journal of Refrigeration**, v. 65, 2016. Cited in page 19.

KIM, C. S.; KIM, C.; RHO, O. H. Feasibility study of constant eddy-viscosity assumption in gradient-based design optimization. **Journal of Aircraft**, v. 40, n. 6, p. 1168–1176, 2003. Cited in page 30.

KIM, J. H.; CHOI, J. H.; HUSAIN, A.; KIM, K. Y. Multi-objective optimization of a centrifugal compressor impeller through evolutionary algorithms. **Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy**, v. 224, 2010. Cited in page 51.

KIM, S. G.; LEE, J.; AHN, Y.; LEE, J. I.; ADDAD, Y.; KO, B. Cfd investigation of a centrifugal compressor derived from pump technology for supercritical carbon dioxide as a working fluid. **Journal of Supercritical Fluids**, v. 86, 2014. Cited in page 19.

KUNDU, P. K.; COHEN, I. M.; DOWLING, D. R. **Fluid Mechanics**. Elsevier Science, 2012. ISBN 9780123821003. Available from Internet: <https://books.google.com.br/books?id=iUo_4tsHQYUC>. Visited on: 22 Jan. 2019. Cited in page 34.

LAPOINTE, C.; CHRISTOPHER, J. D.; WIMER, N. T.; HAYDEN, T. R.; RIEKER, G. B.; HAMLINGTON, P. E. Optimization for internal turbulent compressible flows using adjoints. In: **23rd AIAA Computational Fluid Dynamics Conference**. [S.l.: s.n.], 2017. p. 4115. Cited in page 26.

LETTIERI, C.; BALTADJIEV, N.; CASEY, M.; SPAKOVSZKY, Z. Low-flow-coefficient centrifugal compressor design for supercritical co2. **Journal of Turbomachinery**, American Society of Mechanical Engineers Digital Collection, v. 136, n. 8, 2014. Cited in page 17.

LIU, Q.; VASILYEV, O. V. A brinkman penalization method for compressible flows in complex geometries. **Journal of Computational Physics**, Elsevier, v. 227, n. 2, p. 946–966, 2007. Cited in page 49.

MANGANI, L.; CASARTELLI, E.; MAURI, S. Assessment of various turbulence models in a high pressure ratio centrifugal compressor with an object oriented cfd code. **Journal of Turbomachinery**, v. 134, 2012. Cited 3 times in pages 18, 51, and 67.

MARTA, A. C.; MADER, C.; MARTINS, J.; WEIDE, E. Van der; ALONSO, J. A methodology for the development of discrete adjoint solvers using automatic differentiation tools. **International Journal of Computational Fluid Dynamics**, Taylor & Francis Group, v. 21, n. 9-10, p. 307–327, 2007. Cited in page 28.

MARTA, A. C.; SHANKARAN, S. On the handling of turbulence equations in rans adjoint solvers. **Computers & Fluids**, Elsevier, v. 74, p. 102–113, 2013. Cited in page 30.

MCLAUGHLIN, L.; SPENCE, S.; RUSCH, D.; GALLOWAY, L.; GERON, M.; SO, K. K.; FISCHER, M. Numerical and experimental investigation of a radially reduced diffuser design concept for a centrifugal compressor performance at design point. **Aerospace Science and Technology**, v. 126, p. 107590, 2022. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S1270963822002644>. Visited on: 02 Jul. 2022. Cited in page 24.

MENTER, F. R. Two-equation eddy-viscosity turbulence models for engineering applications. **AIAA Journal**, v. 32, 1994. Cited in page 41.

MOOS, O.; KLIMETZEK, F. R.; ROSSMANN, R. **Bionic optimization of air-guiding systems**. [S.l.], 2004. Cited in page 26.

MOSCATELLI, E.; ALONSO, D. H.; Sá, L. F.; PICELLI, R.; SILVA, E. C. N. Topology optimisation for rotor-stator fluid flow devices. **Structural and Multidisciplinary Optimization**, 2022. Cited in page 27.

MOUKALLED, F.; MANGANI, L.; DARWISH, M. **The finite volume method in computational fluid dynamics**. [S.l.]: Springer, 2016. v. 113. Cited in page 72.

MOUKALLED, F.; MANGANI, L.; DARWISH, M. **The Finite Volume Method in Computational Fluid Dynamics An Advanced Introduction with OpenFOAM and Matlab**. [S.l.: s.n.], 2016. v. 113. Cited 8 times in pages 34, 36, 37, 38, 43, 44, 45, and 56.

NADARAJAH, S.; JAMESON, A. A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization. In: **38th Aerospace Sciences Meeting and Exhibit**. [S.l.: s.n.], 2000. p. 667. Cited 2 times in pages 28 and 88.

NASA. The wray-agarwal turbulence model. 2022. Available from Internet: <https://turbmodels.larc.nasa.gov/wray_agarwal.html>. Visited on: 20 Jul. 2022. Cited in page 42.

NEJADALI, J. Shape optimization of regenerative flow compressor with aero-foil type blades using response surface methodology coupled with cfd. **Structural and Multidisciplinary Optimization**, v. 64, 2021. Cited in page 22.

NOCEDAL, J.; WRIGHT, S. **Numerical optimization**. [S.l.]: Springer Science & Business Media, 1999. Cited in page 29.

NRDC. Are the effects of global warming really that bad? 2022. Available from Internet: <https://www.nrdc.org/stories/are-effects-global-warming-really-bad>. Visited on: 09 Aug. 2022. Cited in page 16.

OKA, N.; FURUKAWA, M.; YAMADA, K.; ITOU, S.; IBARAKI, S.; IWAKIRI, K.; HAYASHI, Y. Optimum aerodynamic design of centrifugal compressor impeller using an inverse method based on meridional viscous flow analysis. In: . [S.l.: s.n.], 2017. v. 2C-2017. Cited in page 23.

OKUBO, C. M.; KIYONO, C. Y.; SÁ, L. F.; SILVA, E. C. N. Topology optimization applied to 3d rotor flow path design based on the continuous adjoint approach. **Computers & Mathematics with Applications**, v. 96, p. 16–30, 2021. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S0898122121001863>. Visited on: 20 Jan. 2022. Cited in page 33.

OKUBO, C. M.; SÁ, L. F.; KIYONO, C. Y.; SILVA, E. C. N. A discrete adjoint approach based on finite differences applied to topology optimization of flow problems. **Computer Methods in Applied Mechanics and Engineering**, v. 389, p. 114406, 2022. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S0045782521006551>. Visited on: 17 Jul. 2022. Cited in page 33.

OOFELIE. The oofelie multiphysics suite. 2022. Available from Internet: <https://www.open-engineering.com/index.php/products/the-oofeliemultiphysics-suite/>. Visited on: 21 Dez. 2021. Cited in page 23.

OTHMER, C. A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows. **International Journal for Numerical Methods in Fluids**, Wiley Online Library, v. 58, n. 8, p. 861–877, 2008. Cited 5 times in pages 26, 28, 46, 53, and 54.

PAPOUTSIS-KIACHAGIAS, E. M.; GIANNAKOGLOU, K. C. Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: industrial

applications. **Archives of Computational Methods in Engineering**, Springer, v. 23, n. 2, p. 255–299, 2016. Cited 5 times in pages 26, 28, 29, 46, and 86.

PAPOUTSIS-KIACHAGIAS, E. M.; KONTOLEONTOS, E. a.; ZYMARIS a. S.; PAPADIMITRIOU, D. I.; GIANNAKOGLOU, K. C. Constrained topology optimization for laminar and turbulent flows, including heat transfer. **EUROGEN, Evolutionary and Deterministic Methods for Design, Optimization and Control**, 2011. Cited 2 times in pages 26 and 29.

PATANKAR, S. **Numerical heat transfer and fluid flow**. [S.l.]: Taylor and Francis, 1980. Cited 2 times in pages 68 and 72.

PETSC. Changes: 3.15. 2021. Available from Internet: <https://petsc.org/release/docs/changes/315/>. Visited on: 29 Oct. 2021. Cited in page 68.

PHAM, H. S.; ALPY, N.; FERRASSE, J. H.; BOUTIN, O.; TOTHILL, M.; QUENAUT, J.; GASTALDI, O.; CADIOU, T.; SAEZ, M. An approach for establishing the performance maps of the sc-co2 compressor: Development and qualification by means of cfd simulations. **International Journal of Heat and Fluid Flow**, v. 61, 2016. Cited in page 17.

PICELLI, R.; MOSCATELLI, E.; YAMABE, P. V. M.; ALONSO, D. H.; RANJBARZADEH, S.; GIORIA, R. dos S.; MENEGHINI, J. R.; SILVA, E. C. N. Topology optimization of turbulent fluid flow via the tobs method and a geometry trimming procedure. **Structural and Multidisciplinary Optimization**, v. 65, 2022. Cited in page 27.

PINTO, R. N.; AFZAL, A.; D'SOUZA, L. V.; ANSARI, Z.; SAMEE, A. D. M. Computational fluid dynamics in turbomachinery: A review of state of the art. **Archives of Computational Methods in Engineering**, v. 24, 2017. Cited in page 19.

PIQUET, A.; ROUSSEL, O.; HADJADJ, A. A comparative study of brinkman penalization and direct-forcing immersed boundary methods for compressible viscous flows. **Computers & Fluids**, Elsevier, v. 136, p. 272–284, 2016. Cited in page 49.

PRITCHARD, P. J. **Fox and McDonald's Introduction to Fluid Mechanics, 8th Edition**. John Wiley & Sons, 2010. ISBN 9781118139455. Available from Internet: <https://books.google.com.br/books?id=RdIbAAAAQBAJ>. Visited on: 02 Jun. 2018. Cited in page 48.

RINALDI, E.; PECNIK, R.; COLONNA, P. Computational fluid dynamic simulation of a supercritical co2 compressor performance map. **Journal of Engineering for Gas Turbines and Power**, v. 137, 2015. Cited in page 17.

ROMEI, A.; GAETANI, P.; PERSICO, G. Computational fluid-dynamic investigation of a centrifugal compressor with inlet guide vanes for supercritical carbon dioxide power systems. **Energy**, v. 255, p. 124469, 2022. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S036054422201372X>. Visited on: 10 Aug. 2022. Cited in page 20.

ROMERO, J. S.; SILVA, E. C. N. A topology optimization approach applied to laminar flow machine rotor design. **Computer Methods in Applied Mechanics and Engineering**, Elsevier, v. 279, p. 268–300, 2014. Cited 3 times in pages 24, 27, and 29.

ROTH, R.; ULBRICH, S. A discrete adjoint approach for the optimization of unsteady turbulent flows. **Flow, Turbulence and Combustion**, v. 90, p. 763–783, 2013. Cited in page 57.

SÁ, L. F. N.; AMIGO, R. C.; NOVOTNY, A. A.; SILVA, E. C. N. Topological derivatives applied to fluid flow channel design optimization problems. **Structural and Multidisciplinary Optimization**, v. 54, n. 2, p. 249–264, aug 2016. Available from Internet: <http://link.springer.com/10.1007/s00158-016-1399-0>. Visited on: 22 Feb. 2018. Cited in page 86.

SÁ, L. F. N.; NOVOTNY, A. A.; ROMERO, J. S.; SILVA, E. C. N. Design optimization of laminar flow machine rotors based on the topological derivative concept. **Structural and Multidisciplinary Optimization**, Springer, v. 56, n. 5, p. 1013–1026, 2017. Cited in page 27.

SÁ, L. F. N.; OKUBO, C. M.; SÁ, A. N.; SILVA, E. C. N. Continuous boundary condition propagation model for topology optimization. **Structural and Multidisciplinary Optimization**, v. 65, 2022. Cited in page 33.

SÁ, L. F. N.; OKUBO, C. M.; SILVA, E. C. N. Topology optimization of subsonic compressible flows. **Structural and Multidisciplinary Optimization**, v. 64, n. 1, p. 1–22, jul 2021. Available from Internet: <https://link.springer.com/10.1007/s00158-021-02903-5>. Visited on: 20 Jan. 2022. Cited 3 times in pages 33, 48, and 51.

SÁ, L. F. N.; ROMERO, J. S.; HORIKAWA, O.; SILVA, E. C. N. Topology optimization applied to the development of small scale pump. **Structural and Multidisciplinary Optimization**, Springer, v. 57, n. 5, p. 2045–2059, 2018. Cited in page 27.

SÁ, L. F. N.; YAMABE, P. V.; SOUZA, B. C.; SILVA, E. C. N. Topology optimization of turbulent rotating flows using spalart-allmaras model. **Computer Methods in Applied Mechanics and Engineering**, v. 373, 2021. Cited 2 times in pages 27 and 30.

SHAABAN, S. Design optimization of a centrifugal compressor vaneless diffuser. **International Journal of Refrigeration**, v. 60, 2015. Cited in page 22.

SOUZA, B. C.; YAMABE, P. V.; Sá, L. F.; RANJBARZADEH, S.; PICELLI, R.; SILVA, E. C. N. Topology optimization of fluid flow by using integer linear programming. **Structural and Multidisciplinary Optimization**, 2021. Cited in page 27.

SPALART, P. R.; ALLMARAS, S. R. One-equation turbulence model for aerodynamic flows. **Recherche aerospatiale**, 1994. Cited in page 41.

SVANBERG, K. A class of globally convergent optimization methods based on conservative convex separable approximations. **SIAM Journal on Optimization**, v. 12, 2002. Cited in page 75.

SVANBERG, K. Mma and gcmma - two methods for nonlinear optimization, versions september 2007. **Technical report, Optimization and Systems Theory**, v. 1, 2007. Cited in page 75.

TOWARA, M.; NAUMANN, U. A discrete adjoint model for openfoam. **Procedia Computer Science**, Elsevier, v. 18, p. 429–438, 2013. Cited 2 times in pages 26 and 29.

TOWARA, M.; SCHANEN, M.; NAUMANN, U. Mpi-parallel discrete adjoint openfoam. In: . [S.l.: s.n.], 2015. v. 51. Cited in page 29.

UN. Goal 13: Take urgent action to combat climate change and its impacts. 2022. Available from Internet: <https://www.un.org/sustainabledevelopment/climate-change/>. Visited on: 20 Jul. 2022. Cited in page 16.

VILLIERS, E. D.; OTHMER, C. Multi-objective adjoint optimization of intake port geometry. In: . [S.l.: s.n.], 2012. Cited in page 29.

WHITE, F. M. **Fluid mechanics**. [S.l.]: McGraw-hill, 1999. Cited 2 times in pages 18 and 47.

WILCOX, D. C. Reassessment of the scale-determining equation for advanced turbulence models. **AIAA Journal**, v. 26, 1988. Cited in page 41.

WILCOX, D. C. Turbulence modeling for cfd (third edition). **DCW Industries**, 2006. Cited 3 times in pages 38, 39, and 40.

WRAY, T. J.; AGARWAL, R. K. A new low reynolds number one-equation turbulence model based on a k-Ï‰ closure. In: . [S.l.: s.n.], 2014. Cited in page 41.

WRIGHT, S.; NOCEDAL, J. **Numerical optimization**. [S.l.: s.n.], 1999. Cited in page 76.

XIA, W.; ZHANG, Y.; YU, H.; HAN, Z.; DAI, Y. Aerodynamic design and multi-dimensional performance optimization of supercritical co2 centrifugal compressor. **Energy Conversion and Management**, v. 248, 2021. Cited in page 23.

XU, H.; WRAY, T. J.; AGARWAL, R. K. Application of a new des model based on wray-agarwal turbulence model for simulation of wall-bounded flows with separation. In: . [S.l.: s.n.], 2017. Cited in page 41.

APPENDIX A – NOTES ON INTEGRATION

The integration by parts and scalar triple product for terms of Equation 4.5 can be seen below:

$$\int_\Omega \nabla^2\delta\boldsymbol{u}\cdot\boldsymbol{\psi_u}\,d\Omega = -\int_\Omega \nabla\delta\boldsymbol{u}:\nabla\boldsymbol{\psi_u}\,d\Omega + \int_\Gamma \boldsymbol{\psi_u}\left(\nabla\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\right)\,d\Gamma = \int_\Omega \nabla^2\boldsymbol{\psi_u}\cdot\delta\boldsymbol{u}\,d\Omega + \\ + \int_\Gamma \left(\boldsymbol{\psi_u}\left(\nabla\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\right) - \delta\boldsymbol{u}\left(\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}\right)\right)\,d\Gamma \tag{A.1}$$

Knowing that $\nabla\cdot\left(\nabla\delta\boldsymbol{u}^T\right) = \nabla\left(\nabla\cdot\delta\boldsymbol{u}\right)$:

$$\int_\Omega \nabla\left(\nabla\cdot\delta\boldsymbol{u}\right)\cdot\boldsymbol{\psi_u}\,d\Omega = -\int_\Omega \left(\nabla\cdot\delta\boldsymbol{u}\right)\left(\nabla\cdot\boldsymbol{\psi_u}\right)\,d\Omega + \int_\Gamma \left(\nabla\cdot\delta\boldsymbol{u}\right)\left(\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}\right)\,d\Gamma = \\ = \int_\Omega \nabla\left(\nabla\cdot\boldsymbol{\psi_u}\right)\cdot\delta\boldsymbol{u}\,d\Omega + \int_\Gamma \left(\left(\nabla\cdot\delta\boldsymbol{u}\right)\left(\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}\right) - \left(\nabla\cdot\boldsymbol{\psi_u}\right)\left(\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\right)\right)\,d\Gamma \tag{A.2}$$

$$\int_\Omega \boldsymbol{\psi_u}\cdot\left(\nabla\delta\boldsymbol{u}\cdot\boldsymbol{u}\right)\,d\Omega = -\int_\Omega \delta\boldsymbol{u}\cdot\left(\nabla\boldsymbol{\psi_u}\cdot\boldsymbol{u}\right)\,d\Omega + \int_\Gamma \left(\boldsymbol{\psi_u}\cdot\delta\boldsymbol{u}\right)\left(\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\right)\,d\Gamma \tag{A.3}$$

$$\int_\Omega \left(\nabla\cdot\delta\boldsymbol{u}\right)\left(\boldsymbol{u}\cdot\boldsymbol{\psi_u}\right)\,d\Omega = -\int_\Omega \nabla\left(\boldsymbol{u}\cdot\boldsymbol{\psi_u}\right)\cdot\delta\boldsymbol{u}\,d\Omega + \int_\Gamma \left(\boldsymbol{u}\cdot\boldsymbol{\psi_u}\right)\cdot\left(\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\right)\,d\Gamma \tag{A.4}$$

$$\int_\Omega \nabla\delta p\cdot\boldsymbol{\psi_u}\,d\Omega = -\int_\Omega \delta p\nabla\cdot\boldsymbol{\psi_u}\,d\Omega + \int_\Gamma \delta p\left(\boldsymbol{\psi_u}\cdot\hat{\boldsymbol{n}}\right)\,d\Gamma \tag{A.5}$$

$$\int_\Omega 2\left(\boldsymbol{\omega}\times\delta\boldsymbol{u}\right)\cdot\boldsymbol{\psi_u}\,d\Omega = \int_\Omega 2\left(\boldsymbol{\psi_u}\times\boldsymbol{\omega}\right)\cdot\delta\boldsymbol{u}\,d\Omega \tag{A.6}$$

$$\int_\Omega \left(\nabla\cdot\delta\boldsymbol{u}\right)\psi_c\,d\Omega = -\int_\Omega \nabla\psi_c\cdot\delta\boldsymbol{u}\,d\Omega + \int_\Gamma \psi_c\left(\delta\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\right)\,d\Gamma \tag{A.7}$$

APPENDIX B – ESTIMATION OF DISCRETIZATION UNCERTAINTIES IN CFD

The uncertainties due to the discretization reported in this work are calculated based on the Grid Convergence Index (GCI) method. The procedure to calculate the GCI is described by Celik et al. (2008) and is explained in this Section.

The procedure begins by calculating a representative mesh size h. For 3D and 2D meshes, it is calculated, respectively, by:

$$h = \left[\frac{1}{N}\sum_{i=1}^{N}(\Delta V_i)\right]^{1/3} \tag{B.1}$$

$$h = \left[\frac{1}{N}\sum_{i=1}^{N}(\Delta A_i)\right]^{1/2} \tag{B.2}$$

where V_i is the volume, A_i is the area of the ith cell and N is the total number of cells.

Then, three different sets of grids are chosen and simulations are performed to determine the value of the variable of interest ϕ. It is recommended that the refinement factor used for the meshes $r = h_{coarse}/h_{fine}$ be greater than 1.3.

With three meshes of representative sizes h_1, h_2 and h_3 and defining $r_{21} = h_2/h_1$ and $r_{32} = h_3/h_2$, the apparent order p is calculated using the expressions:

$$p = \frac{1}{ln(r_{21})}\left|ln\left|\frac{\epsilon_{32}}{\epsilon_{21}}\right| + q(p)\right| \tag{B.3}$$

$$q(p) = ln\left(\frac{r_{21}^{p} - s}{r_{32}^{p} - s}\right) \tag{B.4}$$

$$s = 1\ sgn\left(\frac{\epsilon_{32}}{\epsilon_{21}}\right) \tag{B.5}$$

where $\epsilon_{32} = \phi_3 - \phi_2$, $\epsilon_{21} = \phi_2 - \phi_1$ and ϕ_k is the variable of interest calculated from the kth mesh. The set of equations B.3, B.4 and B.5 can be solved using fixed-point iteration with the initial guess equal to the first term.

After, the extrapolated values are calculated with:

$$\phi_{ext}^{21} = \frac{r_{21}^{p}\phi_1 - \phi_2}{r_{21}^{p} - 1} \tag{B.6}$$

$$\phi_{ext}^{32} = \frac{r_{32}^{p}\phi_1 - \phi_2}{r_{32}^{p} - 1} \tag{B.7}$$

Finally, the estimated error can be calculated in its approximated relative form, extrapolated relative form and fine-grid convergence index:

$$e_a^{21} = \left| \frac{\phi_1 - \phi_2}{\phi_1} \right| \tag{B.8}$$

$$e_{ext}^{21} = \left| \frac{\phi_{ext}^{21} - \phi_1}{\phi_{ext}^{21}} \right| \tag{B.9}$$

$$GCI_{fine}^{21} = \frac{1.25 e_a^{21}}{r_{21}^{p} - 1} \tag{B.10}$$

In Tabs. B.1 and B.2, the procedure is exemplified for the compressible body fitted case in Sec. 6.8 and considering ϕ as the objective function.

Table B.1 – Mesh parameters and ϕ values.

	Mesh 1	Mesh 2	Mesh 3
N	322 958	136 266	57 360
$V\ (m^3)$	7.13×10^{-3}	7.13×10^{-3}	7.13×10^{-3}
$h\ (m)$	2.81×10^{-3}	3.74×10^{-3}	4.99×10^{-3}
ϕ	1.87	1.82	1.59

Table B.2 – GCI calculation example.

Parameter	Value
r_{21}	1.33
r_{32}	1.33
p	5.31
ϕ_{ext}^{21}	4.62
e_a^{21}	2.67%
e_{ext}^{21}	1.88%
GCI_{fine}^{21}	0.92%