

Paulo Vinícius Miyuki Yamabe

**Topology Optimization applied to supersonic inviscid  
compressible fluids**

São Paulo  
December 2022

Paulo Vinícius Miyuki Yamabe

# **Topology Optimization applied to supersonic inviscid compressible fluids**

**Corrected Version**

Thesis submitted to Polytechnic School of the University of São Paulo for the acquisition of the degree of Doctor of Science

Field of Study:  
Control and Mechanical Automation  
Engineering (3152)

Supervised by:  
Prof. Dr. Emilio Carlos Nelli Silva

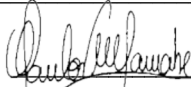
São Paulo  
December 2022

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

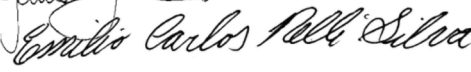
Este exemplar foi revisado e corrigido em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 19 de Dezembro de 2022

Assinatura do autor:



Assinatura do orientador:



Scanned with CamScanner

#### Catálogo-na-publicação

Yamabe, Paulo Vinicius Miyuki

Topology Optimization applied to supersonic inviscid compressible fluids  
/ P. V. M. Yamabe -- versão corr. -- São Paulo, 2022.

75 p.

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

1.Métodos Topológicos (Otimização) 2.Dinâmica dos Fluidos (Simulação)  
3.Método dos Elementos Finitos I.Universidade de São Paulo. Escola  
Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas  
Mecânicos II.t.

Yamabe, Paulo Vinicius Miyuki. **Topology Optimization applied to supersonic inviscid compressible fluids**, 2022, Ph.D. Thesis (Doctor of Science). Escola Politécnica – Universidade de São Paulo – São Paulo

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

Gostaria de agradecer todas e todos que fizeram parte da elaboração e da conclusão do presente trabalho, em especial minha esposa Gabriela e toda minha família.

Um destaque a todos os integrantes do Laboratório de Sistemas Multifísicos (MSOL), pelas conversas, reuniões, apresentações, cafés e confraternizações.

Obrigado Professor Emílio, pelas orientações que me guiaram durante todos esses anos, e principalmente por acreditar no meu trabalho.

Por último mas não menos importante, agradeço o apoio do RCGI — Research Centre for Greenhouse Gas Innovation, localizado na Universidade de São Paulo (USP) e financiado pela FAPESP — Fundação de Amparo à Pesquisa do Estado de São Paulo (2014/50279-4 and 2020/15230-5) e Shell Brasil, e a importância estratégica do apoio dado pela ANP (Agência Nacional do Petróleo, Gás Natural e Biocombustíveis) através do incentivo regulatório de P&D.

*"If you want to go fast, go alone  
If you want to go far, go together"*  
*African proverb*

# RESUMO

A área da engenharia possui grande interesse no estudo de equipamentos que lidam com fluidos compressíveis, podemos citar como exemplo o escoamento ao redor de perfis de asa, de pontes, e edifícios, ou ainda o escoamento dentro de bocais, turbinas, compressores, entre outros. O uso de simulações computacionais durante a fase inicial de projeto desses equipamentos tornou-se essencial, por causa da flexibilidade na análise de diferentes condições de operação antes da construção efetiva de um protótipo, o que diminui o custo e evita a realização de experimentos desnecessários. Uma ferramenta computacional que vem sendo bastante utilizada durante a fase de projeto é a chamada Otimização Topológica (OT), que é capaz de gerar geometrias não intuitivas que otimizam o dispositivo para uma dada função objetivo ou de mérito. A principal vantagem da OT é obter uma geometria otimizada, mesmo nos estágios iniciais do projeto. A OT foi aplicada com sucesso em outras áreas, como na análise estrutural, dispositivos térmicos, entre outras. Todavia, em problemas de escoamento de fluidos esta metodologia tem sido majoritariamente aplicada a problemas considerando fluidos incompressíveis. Portanto, o principal objetivo desta tese é desenvolver uma metodologia para aplicar a OT em escoamentos compressíveis invíscidos supersônicos. Neste trabalho o método escolhido para a resolução das equações governantes é o Método de Elementos Finitos por Mínimos Quadrados, ou em inglês *Least-Squares Finite Element Method (LSFEM)*, as sensibilidades são calculadas utilizando o Método Adjunto, a atualização das variáveis de projeto é feita usando a Otimização Topológica de Estruturas Binárias, ou em inglês *Topology Optimization of Binary Structures (TOBS)*, utiliza-se também um algoritmo para a identificação e suavização do contorno para cada iteração da otimização. O escopo deste trabalho é utilizar a OT para o caso de escoamentos supersônicos em bocais bidimensionais considerando as hipóteses de fluido compressível invíscido e gás perfeito. O programa computacional foi desenvolvido em linguagem de programação *Python*, com o auxílio das bibliotecas livres *FEniCS* e *pyadjoint*, o otimizador binário utilizado foi o CPLEX. A metodologia proposta demonstrou-se robusta e gerou resultados otimizados de um bocal convergente-divergente com a presença de uma onda de choque normal.

**Palavras-chave:** Otimização Topológica. Fluidos Compressíveis. Método dos Elementos Finitos por Mínimos Quadrados. Simulações Computacionais.

# ABSTRACT

The Engineering field has great interest in the study of devices that deals with compressible fluids, such as the flow around wing profiles, bridges, and buildings, or in the flow inside nozzles, turbines, and compressors. The use of computer simulations in the design of such devices has become essential, due to the flexibility in the analysis of different operation conditions, before the actual construction of the prototype, which lowers the cost and avoids unnecessary experiments. One computational tool that has been extensively used during the design project is the Topology Optimization (TO), that can generate non intuitive geometries that optimize the devices for a given objective function or metric of interest. The main advantage of TO is to obtain an optimized geometry, even in the early stages of the project. TO has been successfully applied to different fields of study, such as structural analysis, thermal devices, among others. However, in fluid flow problems this methodology has been mostly applied to incompressible fluids. Thus, the main objective of this thesis is to develop a methodology to apply TO in supersonic inviscid compressible fluid flow problems. In this work the method chosen to solve the governing equations is the Least-Squares Finite Element Method (LSFEM), the sensibilities are calculated by using the Adjoint Method, the update of the design variables is performed with the Topology Optimization of Binary Structures (TOBS), an algorithm for the identification and smoothing of the boundary for every optimization step is also used. The scope of this thesis is to use the TO for the case of supersonic flows in two-dimensional nozzles considering the hypothesis of inviscid compressible fluid and the perfect gas assumption. The computer program is developed in programming language *Python*, with the aid of the open-source libraries FEniCS and pyadjoint, the binary optimizer used is the CPLEX. The proposed methodology presented to be robust and generated optimized designs of a convergent-divergent nozzle with the presence of a normal shock wave.

**Keywords:** Topology Optimization. Compressible Fluids. Least-Squares Finite Element Method. Computer Simulations.



# LIST OF FIGURES

Figure 1.1 – Schematic representation of a nozzle . . . . .	16
Figure 1.2 – Graphic representation of a single variation of a generic control variable	16
Figure 1.3 – Control parameters of a parametric optimization. . . . .	17
Figure 1.4 – Control parameters (blue dots) of a parametric optimization. . . . .	17
Figure 1.5 – Topology optimization. . . . .	18
Figure 1.6 – Schematic representation of the double pipe problem. . . . .	19
Figure 1.7 – Optimized solution of the double pipe problem using TO. . . . .	19
Figure 2.1 – (a) Generic domain $\Omega$ and boundary $\Gamma$ . (b) Mesh discretization. (c) Mesh element . . . . .	26
Figure 3.1 – Schematic representation of the optimization loop . . . . .	35
Figure 3.2 – Adjoint Method applied at different stages. . . . .	37
Figure 3.3 – Material model evaluation of the variation of $q$ . . . . .	38
Figure 3.4 – Material model testing, spurious vortex inside nozzle at the divergent region. . . . .	39
Figure 3.5 – Optimization loop with a new mesh generation at every iteration. . . .	40
Figure 4.1 – Optimization loop with the main programs used in red. . . . .	43
Figure 4.2 – FEniCS components overview . . . . .	44
Figure 4.3 – Boundary identification and the associated mesh with holes . . . . .	48
Figure 4.4 – Boundary smoothing example . . . . .	49
Figure 4.5 – Generic nozzle mesh generated by FEniCS . . . . .	50
Figure 4.6 – Optimized generic nozzle mesh . . . . .	50
Figure 5.1 – Schematic representation of the shock reflection problem. . . . .	52
Figure 5.2 – Mesh with 60x20 quadrilateral elements used in the shock reflection problem. . . . .	53
Figure 5.3 – Solution obtained for Density, Pressure and Mach number in the shock reflection problem . . . . .	53
Figure 5.4 – Comparison of the Density solutions considering different time steps. . .	54
Figure 5.5 – Comparison of the Density solutions considering different mesh refinements.	54
Figure 5.6 – Comparison of the Density solutions considering the stagnation temperature conservation, without the additional term, and the analytical solution. . . . .	55
Figure 5.7 – Convergent-divergent nozzle and boundary conditions . . . . .	55
Figure 5.8 – Mesh of 700 quadrilateral elements. . . . .	56
Figure 5.9 – Density results for the Arina’s nozzle. . . . .	56
Figure 5.10–Pressure results for the Arina’s nozzle. . . . .	57
Figure 5.11–Mach results for the Arina’s nozzle. . . . .	57
Figure 5.12–Stagnation temperature results for the Arina’s nozzle. . . . .	58

Figure 5.13–Stagnation temperature error results for the Arina’s nozzle. . . . .	58
Figure 5.14–Stagnation pressure results for the Arina’s nozzle. . . . .	59
Figure 5.15–Mach number results for four mesh with different refinement levels. . .	60
Figure 5.16–Detail of shock position for different refinement levels . . . . .	60
Figure 5.17–Comparison of the influence of the stagnation temperature in the LSFEM formulation. . . . .	61
Figure 5.18–Comparison of Mach number for different software . . . . .	61
Figure 5.19–Optimization loop. . . . .	62
Figure 5.20–Different meshes used in the optimization iteration . . . . .	63
Figure 5.21–Sensitivity interpolation between meshes . . . . .	63
Figure 5.22–Optimization interpolation between meshes . . . . .	64
Figure 5.23–Detail of the inlet region after one optimization step . . . . .	64
Figure 5.24–Effect of the mesh optimization . . . . .	64
Figure 5.25–Optimization result obtained from the minimization of the pressure inside the nozzle, the green region is the initial guess . . . . .	65
Figure 5.26–Geometry Optimization result obtained from the minimization of the pressure inside the nozzle . . . . .	65
Figure 5.27–Variables plot result obtained from the minimization of the pressure inside the nozzle . . . . .	67
Figure 5.28–Solution became subsonic and lost the shock wave . . . . .	68
Figure 5.29–Intermediate optimized geometry . . . . .	68
Figure 5.30–Intermediate optimized Mach number solution . . . . .	68
Figure 5.31–Using half of optimization mesh . . . . .	68
Figure 5.32–Mach distribution Using half of optimization mesh . . . . .	68
Figure 5.33–Topology Optimization result for the minimization of temperature . . .	68
Figure 5.34–Intermediate optimization result with the presence of holes . . . . .	69
Figure 5.35–Intermediate optimization result with the presence of islands . . . . .	69
Figure 5.36–Temperature distribution of the optimization result . . . . .	69
Figure 5.37–Mach distribution of the optimization result . . . . .	69

# LIST OF TABLES

Table 2.1 – Number of admissible boundary conditions . . . . .	25
Table 2.2 – Relation between the choice of weight function and the numerical method	28
Table 3.1 – Material model behavior . . . . .	38
Table 5.1 – Error analysis of the shock reflection problem . . . . .	53
Table 5.2 – Number of elements of each mesh used in the mesh convergence test . .	59

# ACRONYMS

**CFD** Computational Fluid Dynamics

**FC** Form Compiler

**FDM** Finite Difference Method

**FEM** Finite Element Method

**FFC** FEniCS Form Compiler

**FIAT** FInite element Automatic Tabulator

**LSFEM** Least-Squares Finite Element Method

**PDEs** Partial Differential Equations

**TO** Topology Optimization

**UFC** Unified Form-Assembly Code

**UFL** Unified Form Language

# NOTATIONS

$\mathbf{u}$	Velocity
$p$	Pressure
$\rho$	Density
$t$	time
$Ma$	Mach Number
$a$	Speed of sound
$\gamma$	Heat capacity ratio
$T$	Temperature
$T_0$	Stagnation Temperature
$P_0$	Stagnation Pressure
$\phi$	control variable

# CONTENTS

<b>1</b>	<b>INTRODUCTION</b>	<b>15</b>
<b>1.1</b>	<b>Fluid Flow Topology Optimization</b>	<b>18</b>
<b>1.2</b>	<b>Motivation</b>	<b>21</b>
<b>1.3</b>	<b>Objectives</b>	<b>21</b>
<b>1.4</b>	<b>Scientific Contribution</b>	<b>21</b>
<b>1.5</b>	<b>Outline</b>	<b>22</b>
<b>2</b>	<b>MODELLING</b>	<b>23</b>
<b>2.1</b>	<b>Governing Equations</b>	<b>23</b>
2.1.1	Euler Equations	24
<b>2.2</b>	<b>Finite Element Method</b>	<b>26</b>
2.2.1	Least-Squares Finite Element Method	29
2.2.1.1	LSFEM applied to the Euler Equations	30
<b>3</b>	<b>OPTIMIZATION</b>	<b>34</b>
<b>3.1</b>	<b>Sensitivity analysis with Adjoint Method</b>	<b>36</b>
<b>3.2</b>	<b>Fluid Flow Topology Optimization</b>	<b>37</b>
3.2.1	Material model for inviscid compressible fluids	38
<b>3.3</b>	<b>Proposed optimization procedure</b>	<b>39</b>
<b>3.4</b>	<b>Topology Optimization of Binary Structures (TOBS)</b>	<b>41</b>
<b>4</b>	<b>NUMERIC IMPLEMENTATION</b>	<b>43</b>
<b>4.1</b>	<b>FEniCS</b>	<b>43</b>
4.1.1	LSFEM Implementation	46
<b>4.2</b>	<b><i>pyadjoint</i></b>	<b>47</b>
<b>4.3</b>	<b>TOBS and CPLEX</b>	<b>48</b>
<b>4.4</b>	<b>Boundary identification</b>	<b>48</b>
<b>4.5</b>	<b>Boundary smoothing</b>	<b>49</b>
<b>4.6</b>	<b>Mesh optimization</b>	<b>50</b>
<b>5</b>	<b>RESULTS</b>	<b>51</b>
<b>5.1</b>	<b>Forward solver verification</b>	<b>51</b>
5.1.1	Shock Reflection	51
5.1.2	Normal Shock inside a nozzle	55
5.1.3	Comparison with consolidated CFD softwares	60
<b>5.2</b>	<b>Topology Optimization</b>	<b>62</b>

5.2.1	Optimization Loop . . . . .	62
5.2.2	Minimization of the Pressure . . . . .	64
5.2.3	Maximization of the Mach number . . . . .	65
5.2.4	Minimization of the Temperature . . . . .	66
<b>6</b>	<b>CONCLUSIONS</b> . . . . .	<b>70</b>
<b>6.1</b>	<b>Future Work</b> . . . . .	<b>70</b>
	<b>REFERENCES</b> . . . . .	<b>71</b>

# 1 INTRODUCTION

Computational Fluid Dynamics (CFD) is the area of study of computer simulations applied to fluid flow problems. A computer simulation can be viewed as the study of a system considering models, assumptions, hypotheses, and numerical methods, to describe and predict its behavior, in a computer-based program.

All computer simulations are approximations of the real world system, as a consequence of the hypotheses adopted and the numerical methods implemented. Due to these intrinsic errors, these *in-silico* experiments should always be validated and compared with real laboratory experiments, literature references, analytical results (when they are available) or with other validated software (benchmark), in order to verify the accuracy and efficiency of the program (WINSBERG, 2018).

CFD has been extensively used in the last decades in order to improve the design of engineering devices that deal with compressible flow problems.

As stated in the introductory chapter of the book Perspectives in Flow Control and Optimization from Gunzburger (2003), any flow control or optimization problem is:

". . . an attempt to control the mechanical state, e.g., the rate and direction of motion, and/or the thermodynamic state, e.g., the temperature, of a fluid in order to achieve a desired purpose"

In other words, an optimization problem can be understood as the search for the optimal solution that minimizes or maximizes a given functional of interest or objective function, a metric that is related to the desired purpose or the efficiency of the equipment. The search of a wing profile that maximizes the lift (HUTCHISON et al., 1994), the minimization of a car drag (DUMAS, 2008), to minimize the energy loss inside a pump (SÁ et al., 2018), are examples of optimization problems, where the objective function is the lift, the drag, and the energy dissipation respectively.

It is also possible to perform optimization problems considering restrictions and/or constraints, in order to confine the optimization search of the solution, for example in the optimization of wind turbine design with structural constraint (manufacture) and specific operational constraints (natural frequency) (MUSKULUS; SCHAFHIRT, 2014).

Another strategy in optimization problems is to consider and evaluate several objective functions (multi-objective optimization), for example simultaneously maximize lift and minimize the drag of an airfoil profile (NEMEC; ZINGG; PULLIAM, 2004).



Different optimization methods can be found in the specialized literature, and one manner to categorize them is to subdivide in Parametric, Shape, and Topology, based on the type and location of the control variables.

A brief explanation of each one is presented in the following, where, for the sake of clarification, a bi-dimensional nozzle problem (Fig. 1.1) is chosen to explain all methods.

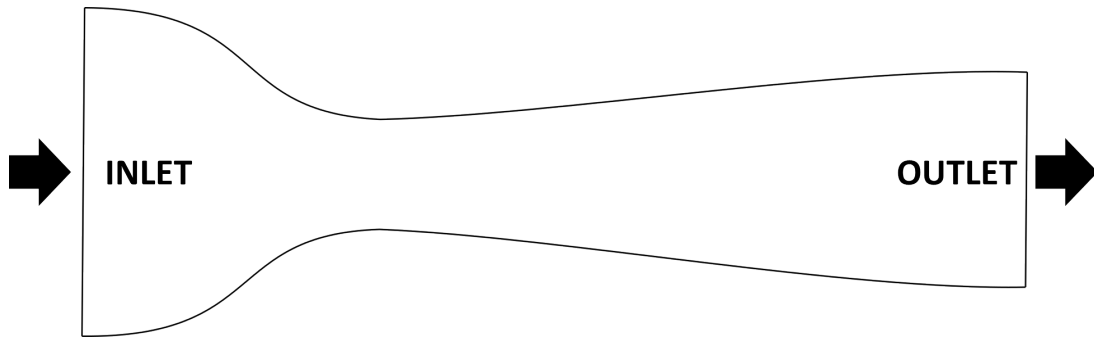


Figure 1.1 – Schematic representation of a nozzle

### Parametric Optimization

The main idea of the Parametric Optimization, is to choose appropriated control or design parameters and run several computer simulations for each parameter in an incremental manner, in order to produce a graphical representation (Fig. 1.2) of how a change in the control variable affects the objective function . This method can be viewed as a brute force search of the optimal solution.

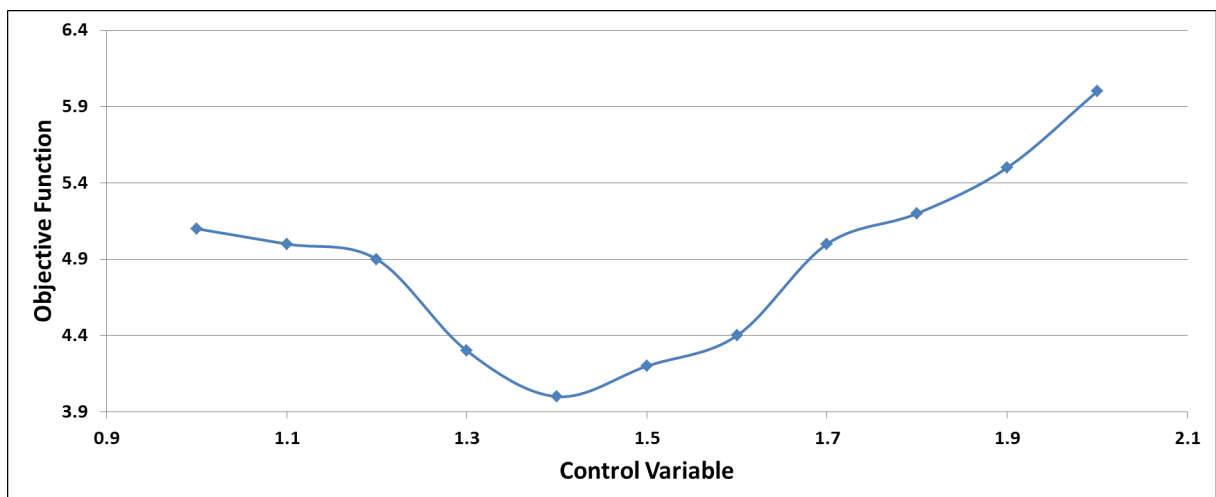


Figure 1.2 – Graphic representation of a single variation of a generic control variable

The control parameters can be related to the geometrical design of the nozzle such as: inlet and outlet diameter, the throat position and diameter, the length, as can be visualized in Fig 1.3. Or can be related to the operational conditions of the nozzle,

such as the inlet and outlet pressure, temperature, etc. The main disadvantage of the parametric optimization is the computational cost associated with the performance of several computer simulations.

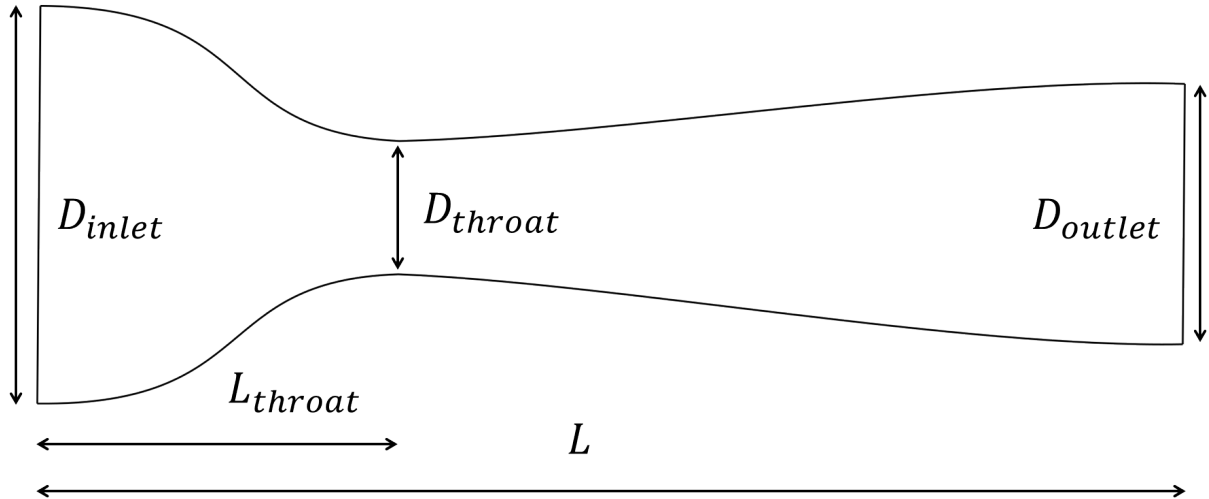


Figure 1.3 – Control parameters of a parametric optimization.

### Shape Optimization

In Shape Optimization ([HASLINGER; MAKINEN, 2003](#)) the primary interest is the geometrical shape of the device, normally the shape is parameterized using functions or splines to obtain fewer control parameters (Fig. 1.4). In this method all the control variables represent the shape of the device studied, and therefore the initial topology is preserved during the optimization, which means that the optimization does not allow the creation and the merging of holes or yet the separation of the design being optimized into two or more parts.

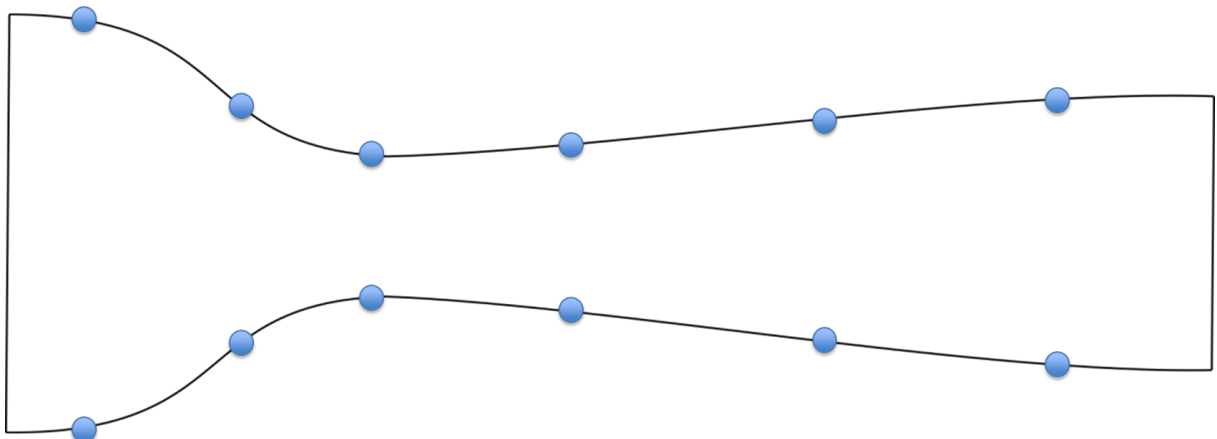


Figure 1.4 – Control parameters (blue dots) of a parametric optimization.

### Topology Optimization

Topology Optimization (TO) (BENDSØE; SIGMUND, 2003) is a different methodology, where the control parameters are localized in the entire project domain. The optimum design is achieved by choosing the locations of the domain that are fluid or solid (1.5), in other words, the optimized design is obtained by the different material distribution inside the allowed domain. This methodology can often generate non-intuitive designs and is most suitable in the initial design process. A negative feature in this methodology is the low fidelity solution in the regions at the interface between solid and fluid.

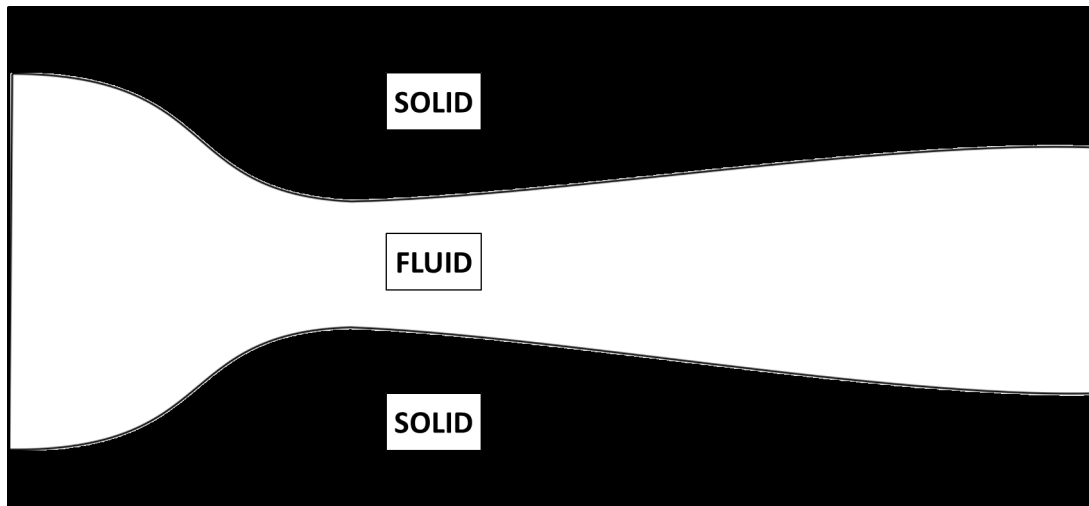


Figure 1.5 – Topology optimization.

It is important to point that all optimization methods can be used together and in an integrated manner during the design process: use the topology optimization for an initial design, then perform a shape optimization to improve the design near the boundaries, and finally execute a parametric optimization around the operational conditions to establish an acceptable range of applications.

All things considered, this thesis focus on the TO applied to fluid flow problems, because of its robustness in generating non-intuitive initial designs that can present holes or different structures inside it.

## 1.1 Fluid Flow Topology Optimization

Borrvall and Petersson (2003) were the first study to use TO in Fluid Flow problems, they analysed creeping flow problems using an incompressible Stokes problem. The main idea was to add a porous media flow term to the momentum equation, associated with the Darcy Law, and emulate the solid as a low permeability region with zero velocity region and the fluid as a region of high permeability. By performing this, the no-slip condition at the interface is readily enforced and the system of equations is well-posed. One of the

examples tested in their work was the double pipe problem described in Fig. 1.6, where the optimization problem is to find the fluid path that minimizes the energy loss of the flow.

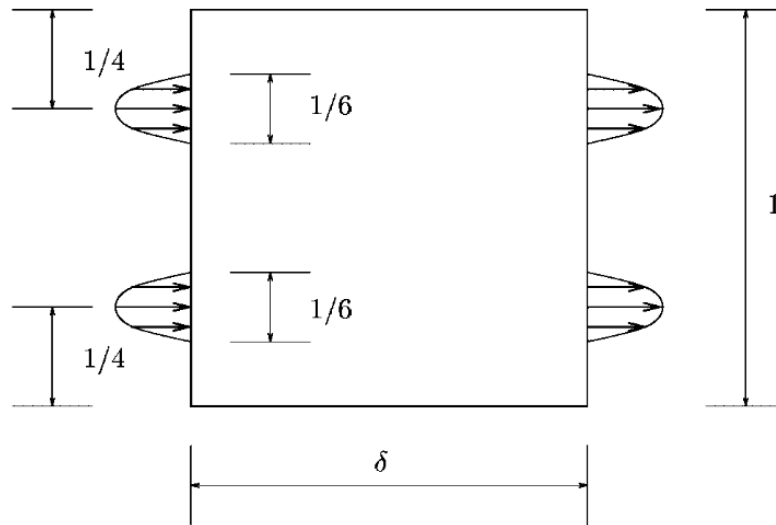


Figure 1.6 – Schematic representation of the double pipe problem.

The optimized solution given by the TO is non-intuitive as can be visualized in Fig. 1.7, in order to minimize the energy loss, the channels joined at the central part. This optimized solution could hardly be found using parametric or shape optimization methods, unless the topology of the pipes were known *a priori*.



Figure 1.7 – Optimized solution of the double pipe problem using TO.

Later this methodology was extended to Navier-Stokes equations in the work of Gersborg-Hansen, Sigmund and Haber (2005), where the non-linear effects of the convection were considered. They used a dimensionless approach of the primitive variables and tested different channels configurations for Reynolds numbers between  $500 \leq Re \leq 1000$  under the hypothesis of laminar flow.

The first three-dimensional result is presented in the work of [Pingen, Evgrafov and Maute \(2007\)](#), where they used the Lattice Boltzmann method (LBM) approximation of the Navier-Stokes equations with a coarse mesh discretization in a nozzle diffuser problem.

TO has been successfully being applied to a large variety of fluid flow problems such as rotating machines ([ROMERO; SILVA, 2014](#)), tesla pumps([ALONSO; SILVA, 2022](#)), fluid diodes ([LIN et al., 2015](#)), non-newtonian fluids ([PINGEN; MAUTE, 2010](#)), fluid solid interaction (FSI) ([YOON, 2010](#)) ([PICELLI et al., 2020](#)), fluid heat transfer ([HØGHØJ et al., 2020](#)), just to cite a few.

The inclusion of the turbulence in the TO modelling can be found in the work of [Yoon \(2016\)](#) where the author explored the Reynolds Average Navier-Stokes (RANS) with a one equation turbulence model, Spalart–Allmaras. Two equation turbulence models such as the Launder–Sharma  $k-\epsilon$  and the  $k-\omega$  were studied respectively by [Papoutsis-Kiachagias and Giannakoglou \(2016\)](#) and [Dilgen et al. \(2018\)](#).

By taking into account the turbulence, these works showed significant differences in the optimized geometries obtained, when compared with laminar non-turbulent geometries and frozen turbulence derivations (when the turbulent viscosity is considered constant and not taken into account in the sensitivities calculations).

The majority of papers found in the literature about TO and fluid flow problems target incompressible fluids, as a matter of fact, a thorough search of the relevant literature yielded few related articles or publications exploring the TO in the context of compressible fluid flow problems ([ALEXANDERSEN; ANDREASEN, 2020](#)).

Among the works dealing with TO in compressible fluid flow problems, the ones that stands out are the work of [Sa, Okubo and Silva \(2021\)](#) and [Okubo et al. \(2022\)](#) where the authors accounted for the compressibility effects of the fluid in the governing equations and solved the fully compressible Navier-Stokes. Both of them used objective functions based on the entropy variation, the main differences between the methods are the solvers used and the manner the sensitivities are calculated, the first used a Finite Element Method and an automatic differentiation algorithm and the later used a Finite Volume Method and a finite difference approach to estimate the derivatives.

However none of them dealt with supersonic problems (Mach number  $> 1$ ), possibly due to the peculiarity needed when solving flows with sonic and supersonic velocities. Therefore, there is an important knowledge gap that needs to be explored, the use of TO in problems where the flow surpasses the sonic velocity and becomes supersonic with, possibly, a shock formation.

## 1.2 Motivation

As explained in the previous section, few studies and works have addressed TO considering the hypothesis of compressible fluid and, to the best of this author's knowledge, the method has not yet been applied to supersonic flow problems that involves shock formation. Hence, there is an implicit importance to expand the Topology Optimization to a class of problems that has not been explored and researched by using this optimization method.

## 1.3 Objectives

The primary objective of this thesis is to study and develop a methodology to apply the Topology Optimization to a supersonic inviscid compressible fluid problem. In order to achieve this, the following major steps were necessary:

- solve the Euler equations with the Least-Square Finite Element Method (LSFEM), with the addition of the stagnation temperature conservation equation to correct the shock location.
- use the Topology Optimization of Binary Structures (TOBS) applied to fluid flow problems, to avoid gray scale problems and have a clearly defined boundary interface between the solid and the fluid.
- perform the boundary identification and smoothing at each optimization step, to decouple the CFD problem and always have the correct boundary condition applied.

## 1.4 Scientific Contribution

The main scientific contribution of this work is the development of a methodology to implement the Topology Optimization to inviscid fluid flow problems, that can deal with both subsonic and supersonic velocities, and with shock wave formation.

Three papers have been published as coauthor during the elaboration of this thesis:

- Topology optimization of turbulent rotating flows using Spalart–Allmaras model (SA et al., 2021)
- Topology optimization of fluid flow by using integer linear programming (SOUZA et al., 2021)
- Topology optimization of turbulent fluid flow via the TOBS method and a geometry trimming procedure (PICELLI et al., 2022)

## 1.5 Outline

This thesis is organized as follows, in Chapter 2 the problem modelling is presented, which comprehends the governing equations and all the theoretical methods used. Chapter 3 explains the Optimization methodology with more emphasis to Topology Optimization applied to inviscid compressible fluid flow problems. The numeric implementation and the computer programs used are described in Chapter 4. The results obtained are displayed in Chapter 5. Finally, in Chapter 6, the conclusions and the future works are discussed.

## 2 MODELLING

### 2.1 Governing Equations

All fluid flow problems are governed by Partial Differential Equations (PDEs) called Navier-Stokes equations (ANDERSON, 2011):

Continuity

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0$$

Momentum

$$\frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p - \rho \mathbf{f} - \mathbf{F}_{viscous} = \mathbf{0} \quad (2.1)$$

Energy

$$\frac{\partial (\rho E)}{\partial t} + \nabla \cdot (\rho E \mathbf{u}) - \rho \dot{q} + \nabla \cdot (p \mathbf{u}) - \rho (\mathbf{f} \cdot \mathbf{u}) - \dot{Q}_{viscous} - \dot{W}_{viscous} = 0$$

where  $\rho$  is the density,  $\mathbf{u}$  is the velocity,  $p$  is the pressure,  $\mathbf{F}_{viscous}$  are the viscous forces,  $E$  is the total energy per unit mass,  $\dot{q}$  is the rate of heat transfer per unit mass,  $\dot{Q}_{viscous}$  is the rate of heat transfer due to viscous effects, and  $\dot{W}_{viscous}$  is the work performed by the viscous forces.

Even though all fluids (liquids and gases) are compressible to some extent, they can be modeled as compressible or incompressible depending on the flow regime, more specifically depending on the Mach number ( $Ma$ ), the dimensionless relation between the absolute velocity and the speed of sound.

$$Ma = \frac{|\mathbf{u}|}{a}$$

where  $|\mathbf{u}| = \sqrt{u_1^2 + u_2^2 + u_3^2}$  with  $u_i$  being the velocity in the  $i$  direction, and  $a$  is the speed of sound.

As stated in Anderson (2011) a fluid can be modeled as compressible when the  $Ma$  is greater than 0.3. This rule of thumb comes from the fact that a  $Ma = 0.3$  represents a change of roughly 5% in the density, and, therefore, a fluid can be modeled as incompressible with a constant density, without any significant loss of accuracy, for flow regimes of  $Ma \leq 0.3$



One problem that frequently arises in high speed compressible fluid flow problems is the shock formation, a drastic change in the fluid properties that can be seen as a discontinuity. Formally the shock thickness is about a few times the size of the mean free path of the gas molecules (PRITCHARD, 2011). A necessary but not sufficient condition for the development of a shock wave is a flow with  $Ma \geq 1.0$ .

The interaction of shock waves with the viscous boundary layer is a highly complex problem (DOLLING, 2001), especially when turbulence is present, and is not going to be addressed in this thesis. Even though the importance of the Shock-wave boundary layer interactions, in this work, the compressible fluids are modeled using the inviscid hypothesis where all viscous effects are neglected.

### 2.1.1 Euler Equations

Considering the working fluid inviscid ( $\dot{Q}_{viscous} = \dot{W}_{viscous} = 0$ ), using the perfect gas assumption  $p = \rho RT$ , in the absence of body forces ( $\mathbf{f} = 0$ ) and heat generation ( $\dot{q} = 0$ ), the Navier-Stokes equations (Eq. 2.1) can be written as (PONTAZA et al., 2004):

Continuity

$$\frac{\partial \rho}{\partial t} + (\mathbf{u} \cdot \nabla) \rho + \rho (\nabla \cdot \mathbf{u}) = 0 \quad (2.2)$$

Momentum

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \frac{1}{\rho} \nabla p = \mathbf{0} \quad (2.3)$$

Energy

$$\frac{\partial p}{\partial t} + (\mathbf{u} \cdot \nabla) p + \gamma p (\nabla \cdot \mathbf{u}) = 0 \quad (2.4)$$

where  $\gamma$  is the heat capacity ratio or adiabatic index. Eqs 2.2, 2.3, and 2.4 are also known as the Euler Equations written in a non-conservative formulation<sup>1</sup>

The solution of the Euler Equations can be understood as a solution of the Navier-Stokes equations in the limit of vanishing viscosity, which is a valid solution for convection dominated problems such as High Reynolds number flows<sup>2</sup> (HIRSCH, 1990).

The hyperbolic nature of the Euler Equations implies that the information propagates like waves and the physical system is dominated by transport phenomena, therefore, it is related to the mathematical concept of "*characteristic surfaces*". The definition of the "*characteristics variables*" and its calculations are out of the scope of this thesis, the interested reader is referred to the books Hirsch (1988) and Hirsch (1990) for more in-depth information.

<sup>1</sup> using the primitive variables  $(\rho, \mathbf{u}, p)$ .

<sup>2</sup> Reynolds number is the dimensionless relation between the inertial and the viscous effects

The method of characteristics reveals the number of admissible boundary conditions that can be applied to a given hyperbolic system and is related to the flow regime, more precisely, the number of boundary conditions depends on the Mach number of the flow. Considering the local Mach number at an arbitrary point of the domain the flow can be defined as (ANDERSON, 2011):

**Subsonic:** Velocity lower than the speed of sound ( $Ma < 1$ )

**Sonic:** Velocity equal to the speed of sound ( $Ma = 1$ )

**Supersonic:** Velocity greater than the speed of sound ( $Ma > 1$ )

The number of admissible boundary conditions is summarized in Table 2.1.

Table 2.1 – Number of admissible boundary conditions

	subsonic	supersonic
inlet	3	4
outlet	1	none

It should be noted that in a supersonic flow, all flow variables are defined at the inlet region and none at the outlet.

From the method of characteristics is also possible to derive which physical variables should be enforced at the boundaries, to guarantee a well-posed system of equations. These boundaries conditions cannot be chosen at random, for instance, the subsonic inlet is always ill-posed for the pair velocity and pressure (HIRSCH, 1990). This makes the choice of boundary conditions problem dependent and not trivial.

For a subsonic inlet, usual boundary conditions are the direction of the flow, Stagnation Temperature, and Stagnation Pressure. Stagnation properties are reference values that could be hypothetically obtained if the fluid could be brought to rest adiabatically (ANDERSON, 2011).

The Stagnation Temperature ( $T_0$ ) can be calculated by:

$$T_0 = T \left( 1 + \frac{\gamma - 1}{2} Ma^2 \right) \quad (2.5)$$

And Stagnation Pressure ( $P_0$ ) by:

$$P_0 = P \left( 1 + \frac{\gamma - 1}{2} Ma^2 \right)^{\frac{\gamma}{\gamma - 1}} \quad (2.6)$$

The adequate boundary condition for a solid wall does not depend on the flow regime and can be modeled as a no penetration or slip wall defined by:

$$\mathbf{u} \cdot \mathbf{n} = 0 \quad (2.7)$$

where  $\mathbf{n}$  is the outward normal direction of the wall surface.

For most real-world problems the Euler Equations do not have an analytical solution, as a result of its non-linearity, hyperbolic behavior, and the presence of shock discontinuities. Therefore, the PDEs need to be solved using a numerical method, such as Finite Difference, Finite Volume Method, or Finite Element Method. This thesis is going to focus on the Finite Element Method.

## 2.2 Finite Element Method

The Finite Element Method (FEM) (ERN; GUERMOND, 2004) is a numerical method and a mathematical framework that can be used to find an approximate solution of a given set of PDEs. It basically consists in subdividing the domain of interest in a finite number of small parts or elements (discretization), and approximating the unknown variables by a linear combination of polynomial equations that can be solved numerically.

The mathematical definition of a domain  $\Omega$  is an open, bounded, connected set in  $\mathbb{R}^d$  ( $d$  being the cartesian dimension) with a boundary defined by  $\Gamma$  (Fig. 2.1 a). The subdivision of the domain is called a mesh (Fig. 2.1 b), that consists in the union of a finite number of compact, connected, set of mesh cells or mesh elements (Fig. 2.1 c). In this thesis, for the sake of simplicity, the word element is chosen to represent the mesh elements.

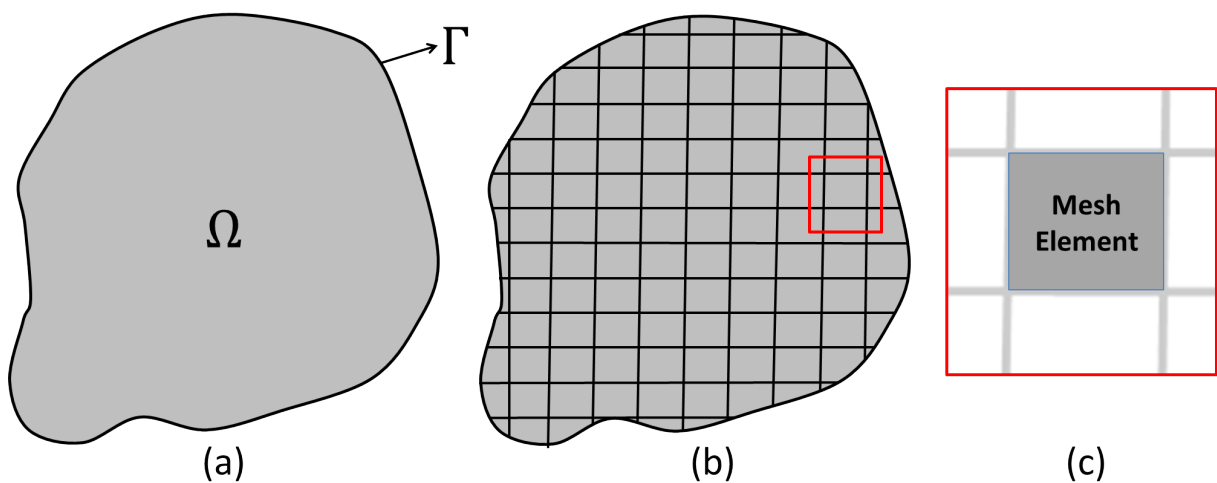


Figure 2.1 – (a) Generic domain  $\Omega$  and boundary  $\Gamma$ . (b) Mesh discretization. (c) Mesh element

For a general variational problem, the FEM can be defined in a canonical form as: find  $\mathbf{w} \in V$  such that:

$$a(\mathbf{w}, \mathbf{v}) = b(\mathbf{v}), \quad \forall \mathbf{v} \in \hat{V}$$

where  $a := V \times \hat{V} \rightarrow \mathbb{R}^d$  is the bilinear form,  $b := \hat{V} \rightarrow \mathbb{R}^d$  is the linear form,  $V$  is the trial function space and  $\hat{V}$  is the test function space.

For a given PDE defined by  $\mathcal{R}(\mathbf{w})$ , where  $\mathcal{R}$  is a differential operator and  $\mathbf{w}$  is the vector of unknown variables, the bilinear and linear forms are:

$$a(\mathbf{w}, \mathbf{v}) = \int_{\Omega} \mathcal{R}(\mathbf{w}) \cdot \mathbf{v} \, d\mathbf{x}$$

$$b(\mathbf{v}) = \int_{\Omega} f \cdot \mathbf{v} \, d\mathbf{x}$$

where  $f$  are the terms that do not depend on the variables (i.e. known boundary conditions, or source terms).

Now we discretize the variational problem and rewrite as: find  $\mathbf{w}_h \in V_h \subset V$  such that

$$a(\mathbf{w}_h, \mathbf{v}) = b(\mathbf{v}_h), \quad \forall \mathbf{v}_h \in \hat{V}_h \subset V$$

The unknown variables  $\mathbf{w}$  are approximated by  $\mathbf{w}_h$ , at each element, as a linear combination of polynomial equations:

$$\mathbf{w}(\mathbf{x}) \approx \mathbf{w}_h(\mathbf{x}) = \sum_{i=1}^{N_{dof}} U_i \Psi_i(\mathbf{x})$$

where  $U_i$  are the unknown values,  $\Psi_i(x)$  are the basis functions of the trial space and  $N_{dof}$  is the number of degrees of freedom of the element.

Using the approximated solution  $\mathbf{w}_h(\mathbf{x})$  in the governing equations, calculating the nonzero residual  $\mathcal{R}(\mathbf{w}_h)$  and making the residual zero in a weight-integral sense we can write:

$$\int_{\Omega} \mathcal{R} \cdot W_i(\mathbf{x}) \, d\Omega = 0 \quad \text{for } i = 1, \dots, N_{dof}$$

where  $W_i(\mathbf{x})$  are weighted functions.

This procedure generates a system of algebraic equations of the kind  $\mathbf{Ax} = \mathbf{b}$  that can be solved numerically. The problem now is to seek a solution of  $\mathbf{w}_h$  which ensures that the integration above is satisfied only at the unknown values of the elements. The choice of the weighted function dictates the numerical method used, Table 2.2 summarize the most common choices in the literature (KARNIADAKIS; SHERWIN, 2005).

Table 2.2 – Relation between the choice of weight function and the numerical method

Weight function	Numerical Method
$\Psi_i(\mathbf{x})$	Galerkin
$\Xi_i(\mathbf{x}) \neq \Psi_i(\mathbf{x})$	Petrov-Galerkin
$\delta\mathcal{R}$	Least-Squares

Use the same basis functions  $\Psi_i(x)$  (trial functions) as the weight functions  $W_i(x)$  (test functions) is known as the Galerkin's Method and is one of the most common FEM<sup>3</sup>.

However, the Galerkin's Method when applied to convection-dominated transport problems, which is the case of the Euler Equations, are often corrupted by spurious oscillations (DONEA; HUERTA, 2003).

Another issue that frequently arises in the Galerkin's Method is the proper choice of the test functions space, for example, the use of same interpolation order for all variables can lead to numerical divergence in some problems (Mixed Finite Element problems), and is known as Ladyzenskaya-Babuška-Brezzi (LBB) condition or inf-sup condition (BABUSKA, 1971).

The spurious oscillations issue can be avoided with the use of Stabilization Methods such as Streamline Upwind Petrov-Galerkin (SUPG), or by the use of Discontinuous Galerkin elements, or yet by the use of the Least-Squares Finite Element Method (LSFEM) variational formulation of the problem, where the test functions are the functional derivative of the PDEs with respect to the variables.

Stabilization Methods are a modification of the Galerkin's Method where a stabilization term is added to the formulation, this stabilization parameter is problem dependent and require a fine tuning (CODINA; BLASCO, 2002).

Discontinuous Galerkin elements use the definition of numerical fluxes, derived from non-trivial Riemann problems, to perform the compatibility of the discontinuous solution at the inter-element level and higher order interpolations suffer from oscillations near the discontinuities, that can be alleviated by the use of numerical dissipation methods that acts as an artificial viscosity (BASSI; REBAY, 1997).

The variational formulation of LSFEM is explained in more details in the following and is the method chosen and implemented in the present work.

<sup>3</sup> Another popular method in CFD, the Finite Volume Method, can be obtained by choosing  $W_i(\mathbf{x}) = 1$  inside each volume and  $W_i(\mathbf{x}) = 0$  outside each volume.

### 2.2.1 Least-Squares Finite Element Method

The LSFEM is a variety of FEM that uses a variational formulation based on the minimization of a quadratic "energy norm"<sup>4</sup> or functional in a least-squares sense, which makes the LSFEM essentially an unconstrained minimization problem whose solution is also the solution of the PDEs. The main advantages of the LSFEM are summarized below (JIANG, 1998):

**General:** Has an unified formulation to a wide range of problems, there is no necessity of implementing a different scheme if the PDEs are elliptic, parabolic, hyperbolic, linear or non-linear.

**Efficient:** The discretized system of equations can be always symmetric positive definite, if the LSFEM is properly implemented, and can be efficiently solved numerically.

**Robust:** Unlike the Galerkin's Method, the LSFEM does not suffer from the LBB condition, and the same order of interpolation can be used for all variables.

The application of the LSFEM to solve PDEs consists in writing the residual as a quadratic energy norm in a suitable function space:

$$\|\mathcal{R}\|^2 = (\mathcal{R}, \mathcal{R}) = \int_{\Omega} \mathcal{R} \cdot \mathcal{R} \, d\mathbf{x} \quad (2.8)$$

where  $(\cdot, \cdot)$  is the inner product induced by the space of square-integrable functions  $L^2$ .

Then we take the variational of Eq. 2.8 and equal it to zero, this procedure is known as the stationary variational condition:

$$\delta \int_{\Omega} \mathcal{R} \cdot \mathcal{R} \, d\mathbf{x} = 0 \quad (2.9)$$

The variational can be understood as type of derivative, therefore the chain rule must be applied:

$$2 \int_{\Omega} \mathcal{R} \cdot \delta \mathcal{R} \, d\mathbf{x} = \int_{\Omega} \mathcal{R} \cdot \delta \mathcal{R} \, d\mathbf{x} = 0 \quad (2.10)$$

Eq. 2.10 represents a generic application of the LSFEM for any system of PDEs, for more comprehensive information about the methodology and the mathematical background of the LSFEM, see the works of Eason (1976) and Bochev and Gunzburger (1998) or the books from Jiang (1998) and Bochev and Gunzburger (2005), and the references therein.

<sup>4</sup> The quotes were used because the functional of interest does not need to be related to energy, in fact, can be any system of PDEs, even the ones that are not derived from energy properties. The use of the word energy is related to the Rayleigh-Ritz Method

### 2.2.1.1 LSFEM applied to the Euler Equations

Different authors applied the LSFEM to the Euler Equations and some of their results are described below:

The work of [Jiang and Carey \(1988\)](#) studied a one dimensional conservative formulation of the Euler Equations for a nozzle problem with the presence of a shock wave. They concluded that it is necessary to add an extra numerical dissipation because of the oscillations near the shock using an energy norm in  $H^1$  instead of  $L^2$ , where  $L^2$  is the space of square integrable functions and  $H^1$  is the Sobolev space functions whose derivatives are in  $L^2$ .

[Lefebvre, Peraire and Morgan \(1993\)](#) also explored the conservative formulation, using a backward difference in time for linear and quadratic elements. Better shock resolution was achieved using an adaptive refinement based on the pressure gradients, which refined the regions near the shock. Another conclusion of their work is that quadratic elements need extra artificial dissipation to achieve convergence.

The non-conservative formulation of the Euler Equations were studied by [Taghaddosi et al. \(1999\)](#) together with a grid adaptation method, where it was implemented a moving node strategy to obtain elements aligned with the shock (shock fitting approach). The slip boundary was implemented adding the boundary condition in the LSFEM functional.

[Pontaza et al. \(2004\)](#) also used a non-conservative formulation with a backward difference in time. They explored different problem such as a shock reflection and a transonic flow inside a channel.

Before applying the LSFEM to the Euler Equations, the time derivative is discretized first using a fully implicit backward Euler scheme:

$$\frac{\partial var}{\partial t} = \frac{var - var_0}{dt} \quad (2.11)$$

where  $var$  can be any primitive variable ( $\rho$ ,  $\mathbf{u}$ ,  $p$ ) at the next time step,  $var_0$  is the primitive variable at the previous time step, and  $dt$  is the time increment.

Which gives:

$$\mathcal{R}_{continuity} = \rho - \rho_0 + dt[(\mathbf{u} \cdot \nabla)\rho + \rho(\nabla \cdot \mathbf{u})] \quad (2.12)$$

$$\mathcal{R}_{momentum} = \mathbf{u} - \mathbf{u}_0 + dt \left[ (\mathbf{u} \cdot \nabla)\mathbf{u} + \frac{\nabla p}{\rho} \right] \quad (2.13)$$

$$\mathcal{R}_{energy} = p - p_0 + dt[(\mathbf{u} \cdot \nabla)p + \gamma p(\nabla \cdot \mathbf{u})] \quad (2.14)$$

By taking the variational of Eqs 2.12, 2.13, 2.14 we arrive at:

$$\delta\mathcal{R}_{continuity} = \delta_\rho + dt[(\delta_{\mathbf{u}} \cdot \nabla)\rho + (\mathbf{u} \cdot \nabla)\delta_\rho + \delta_\rho(\nabla \cdot \mathbf{u}) + \rho(\nabla \cdot \delta_{\mathbf{u}})] \quad (2.15)$$

$$\delta\mathcal{R}_{momentum} = \delta_{\mathbf{u}} + dt \left[ (\delta_{\mathbf{u}} \cdot \nabla)\mathbf{u} + (\mathbf{u} \cdot \nabla)\delta_{\mathbf{u}} + \frac{\rho\nabla\delta_p - \delta_\rho\nabla p}{\rho^2} \right] \quad (2.16)$$

$$\delta\mathcal{R}_{energy} = \delta_p + dt[(\delta_{\mathbf{u}} \cdot \nabla)p + (\mathbf{u} \cdot \nabla)\delta_p + \gamma\delta_p(\nabla \cdot \mathbf{u}) + \gamma p(\nabla \cdot \delta_{\mathbf{u}})] \quad (2.17)$$

where  $\delta_\rho$ ,  $\delta_{\mathbf{u}}$ , and  $\delta_p$  are the test functions of the density, velocity, and pressure respectively.

Using Eq. 2.10 we obtain:

$$\int_{\Omega} \mathcal{R}_{equation} \cdot \delta\mathcal{R}_{equation} d\mathbf{x} = 0 \quad (2.18)$$

where  $\mathcal{R}_{equation}$  and  $\delta\mathcal{R}_{equation}$  are all the equations from Eqs 2.12 to 2.17

It is possible to analyze the LSFEM terms as a combination of different methods, taking the continuity equation for example:

$$\mathcal{R}_{continuity} \cdot \{\delta_\rho + dt[(\delta_{\mathbf{u}} \cdot \nabla)\rho + (\mathbf{u} \cdot \nabla)\delta_\rho + \delta_\rho(\nabla \cdot \mathbf{u}) + \rho(\nabla \cdot \delta_{\mathbf{u}})]\}$$

The inner product between the  $\mathcal{R}_{continuity}$  and the test function  $\delta_\rho$  is the Galerkin's Method, where the PDE is multiplied by a test function.

The term  $\mathcal{R}_{continuity} \cdot dt(\mathbf{u} \cdot \nabla)\delta_\rho$  is exactly the SUPG stabilization term, with a stabilization parameter equal to  $dt$ . It is a residual based stabilization term with an upwinding in the direction of the velocity. The other terms can also be seen as a residual based stabilization, to ensure the symmetric positive definitive character of the linearized equations.

Carey and Jiang (1988) show that the addition of the derivatives of the test function terms can be seen as a Petrov-Galerkin formulation, where the interpolation function is different from the test function and the modified test function is  $\psi = \Phi + dt\nabla\Phi$ .

The time-step ( $dt$ ) acts as an artificial or numerical viscosity that aids the convergence of the solution, and should be large enough to damp numerical oscillations but small enough to not smear out the discontinuities.

Notice that even when solving steady-state problems, a time marching procedure is necessary in order to avoid non-physical solutions<sup>5</sup>, especially when dealing with shock wave problems. The physical or entropy solutions are achieved in the LSFEM because of the artificial viscosity explained above (JIANG; CAREY, 1990).

<sup>5</sup> physical solutions are known as entropy solutions



A pseudo-transient scheme is used in this work, where the time step is different for every element and based on the element size, this leads to an unrealistic intermediate transient solution but with the correct steady-state solution.

As claimed by [Gerritsma et al. \(2008\)](#), the non-conservative formulation produces a wrong shock position, in order to avoid this problem, a Stagnation Temperature conservation term in the entire domain is added to the LSFEM formulation.

$$\mathcal{R}_{T_0} = \frac{p}{\rho} + \frac{\gamma - 1}{2\gamma} \mathbf{u} \cdot \mathbf{u} - T_{0_{inlet}} \quad (2.19)$$

$$\delta \mathcal{R}_{T_0} = \frac{\delta_p \rho - p \delta_\rho}{\rho^2} + \frac{\gamma - 1}{\gamma} \mathbf{u} \cdot \delta \mathbf{u} \quad (2.20)$$

where  $T_{0_{inlet}}$  is the Stagnation Temperature at the inlet<sup>6</sup>.

In this thesis it is also performed the Weighted LSFEM as explained in [Bochev and Gunzburger \(2005\)](#), where each PDE is weighted with different values, to tell the solver which terms are more important or which should have a lower residual. More weight is applied to the continuity equation (Eqs. 2.12 and 2.15) because the unweighted LSFEM is known to suffer from mass conservation errors as discussed in the work of [Bolton and Thatcher \(2005\)](#).

The boundary conditions are derived using the LSFEM and are added to the functional in the same manner as the Stagnation Temperature described above, and the equations for the Stagnation Temperature at the inlet are the same as Eqs. 2.19 and 2.20.

The LSFEM derivation of the Stagnation Pressure is:

$$\mathcal{R}_{P_0} = \frac{\rho^k}{p^{k-1}} - \frac{P_{0_{inlet}}}{T_{0_{inlet}}^k} \quad (2.21)$$

$$\delta \mathcal{R}_{P_0} = \frac{k \rho^{k-1} p^{k-1} \delta_\rho - (k-1) \rho^k p^{k-2} \delta_p}{p^{2k-2}} \quad (2.22)$$

with  $k = \frac{\gamma}{\gamma-1}$

And the boundary condition of the no penetration or slip wall is modeled as:

$$\mathcal{R}_{wall} = \mathbf{u} \cdot \mathbf{n} \quad (2.23)$$

$$\delta \mathcal{R}_{wall} = \delta \mathbf{u} \cdot \mathbf{n} \quad (2.24)$$

where  $\mathbf{n}$  is the outward normal vector of the boundary.

<sup>6</sup> This value should be constant in the entire domain, except at the shock location ([ANDERSON, 2011](#))

The LSFEM also makes use of dimensionless variables in order to scale the residual functionals to the same magnitude order, therefore the variables used are:

$$x = \frac{x_{real}}{L_{ref}} \quad (2.25)$$

$$\mathbf{u} = \frac{\mathbf{u}_{real}}{\mathbf{u}_{ref}} \quad (2.26)$$

$$\rho = \frac{\rho_{real}}{\rho_{ref}} \quad (2.27)$$

$$p = \frac{p_{real}}{\rho_{ref} \mathbf{u}_{ref}^2} \quad (2.28)$$

This procedure does not change the formulation described in later sections, only scales the problem variables.

### 3 OPTIMIZATION

A general optimization problem can be stated as:

$$\begin{aligned}
 & \underset{w, \phi}{\text{minimize/maximize}} \quad \mathcal{J}(w, \phi) \\
 & \text{subject to} \quad \mathcal{F}(w, \phi) = 0 \\
 & \quad \quad \quad g_1(w, \phi) = C_1 \\
 & \quad \quad \quad g_2(w, \phi) \geq C_2 \\
 & \quad \quad \quad \phi_{min} \leq \phi \leq \phi_{max}
 \end{aligned} \tag{3.1}$$

where  $\phi$  are the control variables,  $w$  is the solution of the PDEs,  $\mathcal{J}(w, \phi)$  is the objective function,  $\mathcal{F}(w, \phi) = 0$  the restriction imposed by the PDEs (forward model),  $g_1$  is an equality constraint,  $g_2$  is an inequality constraint, and  $\phi_{min} \leq \phi \leq \phi_{max}$  is the minimum and maximum values allowed for the control variables (box constraint).

Eq. 3.1 can be read as find the best distribution of  $\phi$  that minimizes the objective function  $\mathcal{J}$  for a given set of constraints. In order to solve Eq. 3.1, a systematic procedure is implemented.

- First, the forward model  $\mathcal{F}(w, \phi) = 0$  is solved for a fixed set of controls and the objective function  $\mathcal{J}(w, \phi)$  is evaluated.
- Then a sensitivity analysis is performed, in order to predict which controls  $\phi$  affect most the objective function  $\mathcal{J}(w, \phi)$ . Mathematically, the derivative of the objective function with respect the controls  $d\mathcal{J}/d\phi$  is calculated.
- Finally, the controls are changed using the information of the sensitivity step in order to pursuit the desired extremum of the objective function.

This procedure is repeated in a loop until a given tolerance is achieved, for example when the objective function stabilizes and does not change significantly, or when it reaches a prescribed maximum number of iterations. This optimization loop can be visualized in Fig 3.1.

One important part of any gradient based optimization method is the sensitivity analysis step, by virtue of  $d\mathcal{J}/d\phi$  being the direction used by the optimization algorithm to minimize or maximize the objective function. Therefore, the wrong estimation of the sensitivities will lead to designs that are not optimum.

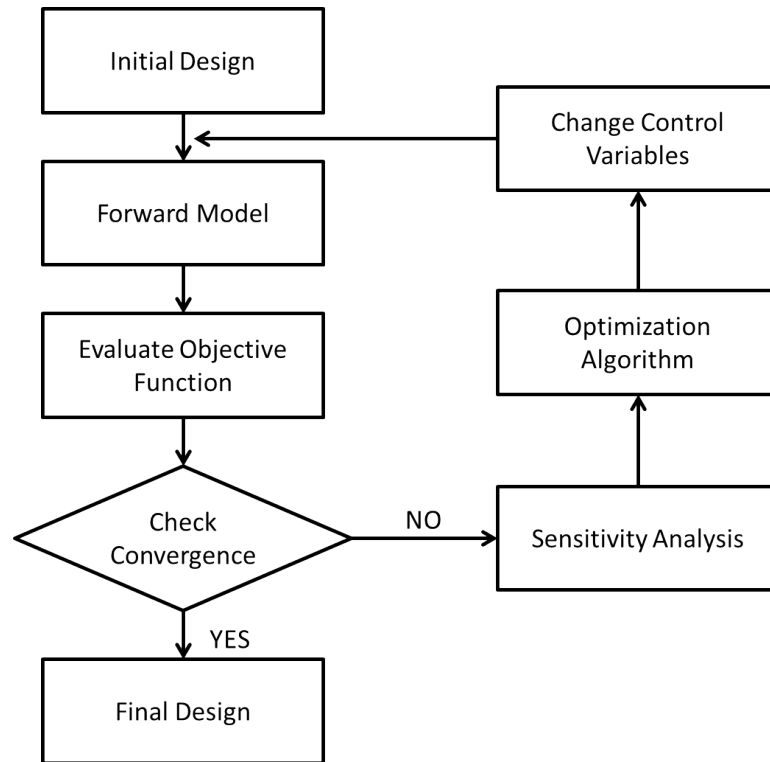


Figure 3.1 – Schematic representation of the optimization loop

There are different approaches to evaluate  $d\mathcal{J}/d\phi$ , such as the Finite Difference Method (FDM) or the Adjoint Method (FUNKE, 2012).

FDM is based on Taylor expansion series to approximate the derivatives, using a central difference scheme the  $d\mathcal{J}/d\phi$  can be calculated by:

$$\frac{d\mathcal{J}}{d\phi} = \frac{\mathcal{J}(\phi + h) - \mathcal{J}(\phi - h)}{h} + \mathcal{O}(h^2) \quad (3.2)$$

where  $h$  is the size of the variation of the control.

One known problem of the FDM is the numerical cancellation, that arises for small values of  $h$  and wrongly predicts the derivatives values. This can be circumvented by using the Complex Step scheme (LYNESS; MOLER, 1967).

Another problem is the computational cost required when the optimization problem has a large number of control variables, due to the large amount of simulations needed to estimate the derivatives for each control variable.

An alternative approach is the Adjoint Method, explained with more details in the next section.

### 3.1 Sensitivity analysis with Adjoint Method

In order to calculate the  $d\mathcal{J}/d\phi$ , we apply the chain rule first:

$$\frac{d\mathcal{J}}{d\phi} = \frac{\partial\mathcal{J}}{\partial w} \frac{dw}{d\phi} + \frac{\partial\mathcal{J}}{\partial\phi} \quad (3.3)$$

The derivative  $dw/d\phi$  can be obtained by taking the derivative of  $\mathcal{F} = 0$  with respect to  $\phi$ .

$$\frac{\partial\mathcal{F}}{\partial w} \frac{dw}{d\phi} + \frac{\partial\mathcal{F}}{\partial\phi} = 0 \quad (3.4)$$

Assuming  $\partial\mathcal{F}/\partial\phi$  invertible:

$$\frac{dw}{d\phi} = -\frac{\partial\mathcal{F}^{-1}}{\partial w} \frac{\partial\mathcal{F}}{\partial\phi} \quad (3.5)$$

Substituting into Eq. 3.3 we arrive at:

$$\frac{d\mathcal{J}}{d\phi} = \underbrace{-\frac{\partial\mathcal{J}}{\partial w} \frac{\partial\mathcal{F}^{-1}}{\partial w}}_{=\lambda^*} \frac{\partial\mathcal{F}}{\partial\phi} + \frac{\partial\mathcal{J}}{\partial\phi} \quad (3.6)$$

where  $\lambda^*$  are the adjoint variables.

The adjoint variables are calculated from:

$$\frac{\partial\mathcal{F}}{\partial w} \lambda^* = \frac{\partial\mathcal{J}}{\partial w} \quad (3.7)$$

The sensitivities  $d\mathcal{J}/d\phi$  are easily calculated by substituting the adjoint variables  $\lambda^*$  into Eq. 3.6.

The Adjoint Method calculates the sensitivities using one additional set of equations (the adjoint equations Eq. 3.7), and therefore, there is one additional system to solve. The main advantage of this method is the associated computational cost that is always related to the additional system to solve, it does not explicitly depends on the number of control variables. This is most suitable for optimization problems with few objective functions and several controls, which is exactly the case of the optimization problems explored in this thesis.

Another advantage is that the adjoint equations are always linear, even when derived from non-linear forward models (FUNKE, 2012).

The Adjoint Method can be applied following different approaches as can be visualized in Fig. 3.2, starting from the continuous form, from the discretized form, or yet from the source code using Automatic Differentiation tools (FUNKE, 2012).

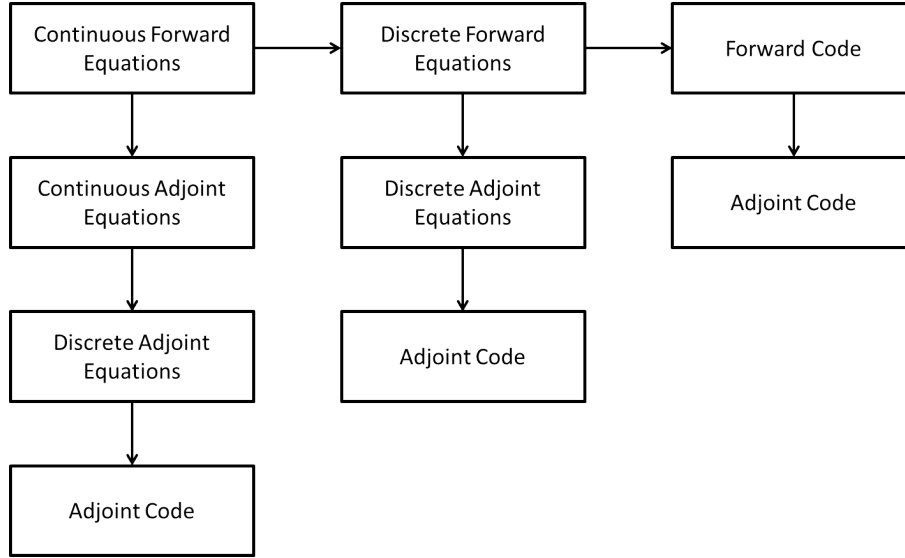


Figure 3.2 – Adjoint Method applied at different stages.

This thesis uses the discrete adjoint equations obtained from the discretized weak form of the PDEs (central block of Fig. 3.2) to calculate the sensitivities needed for the optimization step. The derivation of the discrete adjoint equations is performed in an automatic manner using the *pyadjoint* library and is explained in more details in Sec. 4.2.

## 3.2 Fluid Flow Topology Optimization

As described in the Sec. 1.1, Fluid Flow Topology Optimization started with the seminal work of [Borrvall and Petersson \(2003\)](#), where a porous Darcy term was added to the momentum equation of the Stokes formulation. Mathematically, this term can also be understood as a penalty method that enforces the restriction of no-slip (zero velocity) in the solid part. The altered momentum equation is:

$$\mathcal{R}_{momentum} + \alpha(\phi)\mathbf{u} \quad (3.8)$$

where  $\alpha$  is the reverse permeability and  $\phi$  is a phase field variable that selects the part of the domain that are solid or fluid<sup>1</sup>.

The equation for  $\alpha(\phi)$  used by [Borrvall and Petersson \(2003\)](#) is:

$$\alpha(\phi) = \bar{\alpha} + (\underline{\alpha} - \bar{\alpha})\phi \frac{1+q}{\phi+q} \quad (3.9)$$

where  $\bar{\alpha}$  is the maximum permeability (solid region),  $\underline{\alpha}$  is the minimum permeability (fluid region), and  $q$  is a penalization parameter.

<sup>1</sup> This formulation is the Brinkman formulation for porous media

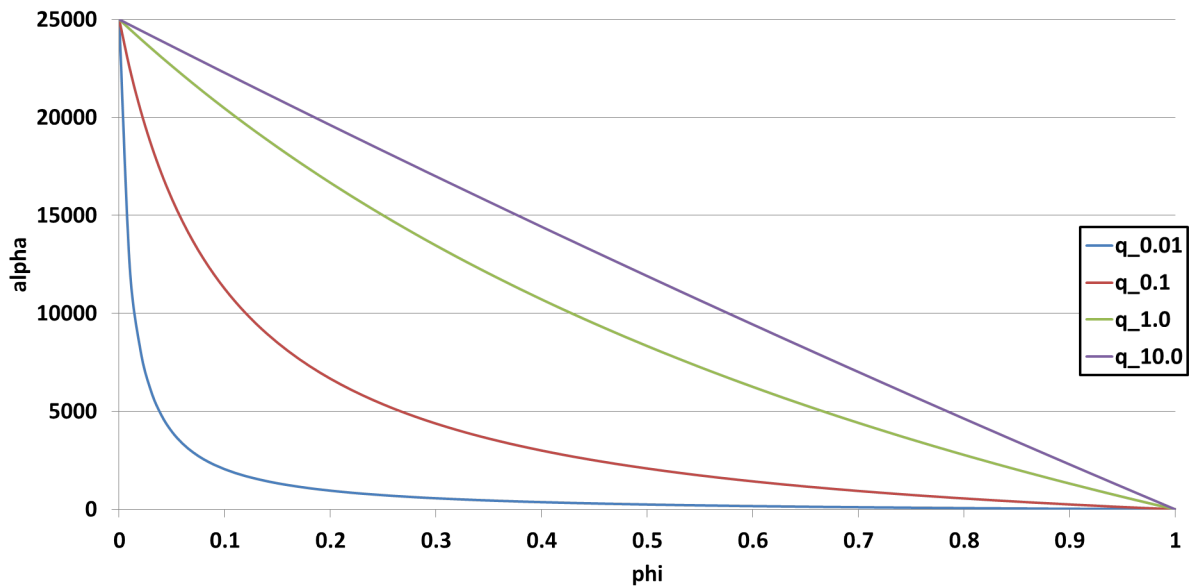
Eq. 3.9 is a material interpolation function, and can be seen as a Material Model that shifts the behavior from fluid to solid. Basically, when  $\phi \rightarrow 1$ ,  $\alpha \rightarrow \underline{\alpha}$  and the momentum equation does not suffer any alteration, which means that it is a fluid region. Conversely, when  $\phi \rightarrow 0$ ,  $\alpha \rightarrow \bar{\alpha}$  and the velocity solution of the momentum equation goes to zero, which means it is a solid region. In the limit the values should be  $\bar{\alpha} \rightarrow \infty$  and  $\underline{\alpha} \rightarrow 0$ . The different behaviors are summarized in Tab. 3.1.

Table 3.1 – Material model behavior

	$\phi$	$\alpha$
Fluid	1	0
Solid	0	$\infty$

The values of  $\bar{\alpha}$  should be high enough to enforce the zero velocity in the solid region, but not so high to make the system ill-conditioned, and the values of  $\underline{\alpha}$  should be low enough to not alter the momentum equations in the fluid region.

The penalization parameter  $q$  controls the level of intermediate values of  $\phi$ , also called "grey scale" in the TO literature. High values of  $q$  induce Eq. 3.9 to assume a linear behavior and consequently the optimization becomes more discrete, with  $\phi$  having only 0 or 1 values. This pattern can be observed in Fig. 3.3

Figure 3.3 – Material model evaluation of the variation of  $q$ .

### 3.2.1 Material model for inviscid compressible fluids

Taking into consideration the shortage of specialized literature about TO applied to compressible fluids, the first Material Model tested for the compressible case is the

same described by Eq. 3.9 for the incompressible case. However this preliminary approach resulted in a non-physical solution of the forward model with spurious vortex inside the tested convergent-divergent nozzle configuration, as can be visualized in Fig. 3.4.

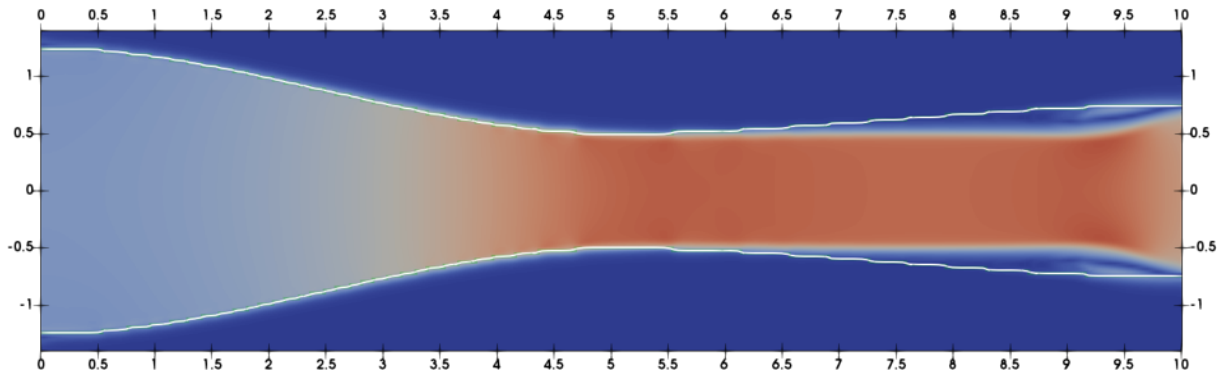


Figure 3.4 – Material model testing, spurious vortex inside nozzle at the divergent region.

The main cause of this behavior is due to the wrong boundary condition applied to the interface between the fluid and the solid. The proper boundary condition should be the slip condition for the Euler Equations and not the no-slip (zero velocity) enforced by the material model used.

The suitable Euler material model should apply the slip condition at the interface, however the development of a novel material model showed to be a challenging and cumbersome task. Therefore, a different procedure is proposed and implemented in this thesis in order to correctly perform the topology optimization of supersonic inviscid compressible fluids.

### 3.3 Proposed optimization procedure

Due to the difficulties encountered in the development of a suitable material model for the inviscid compressible case, this work proposes the following procedure: decouple the fluid region from the optimization and remove the influence of the material model in the forward problem. This can be accomplished by using a geometry trimming algorithm, making the forward problem a regular CFD problem with the proper boundary condition applied. This strategy is similar of what is performed in [Picelli et al. \(2022\)](#).

The proposed method can be described by the following steps:

- 1) Solve the forward problem in a conforming mesh with appropriate boundary conditions, which is a standard CFD solution of a fluid problem.
- 2) Calculate the sensitivity using the Adjoint Method in the conforming mesh.
- 3) Interpolate the sensitivities from the conforming mesh to a larger mesh (extended domain mesh), where the optimization process occurs.



- 4) Solve the optimization problem in the extended domain mesh, using the sensitivities (gradient based) to guide the optimizer.
- 5) Get the new design variables in the extended domain mesh (solution of step 4)
- 6) Identify the fluid/solid interface in the extended domain and extract the fluid part only (geometry trimming).
- 7) Smooth the boundary of the fluid part, in order to avoid an irregular contour at the boundaries.
- 8) Create a new mesh and repeat step 1.

This methodology is performed as an iterative process until a convergence criteria is reached and the loop ends with an optimized design. This optimization loop can be visualized in Fig. 3.5.

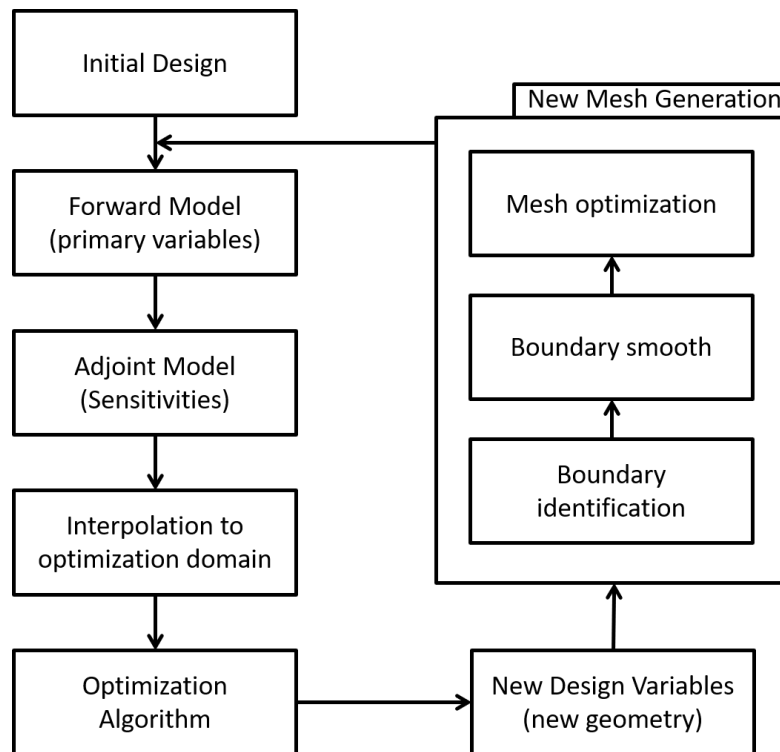


Figure 3.5 – Optimization loop with a new mesh generation at every iteration.

The proposed methodology can be seen as a post-processing step at every optimization iteration, where the intermediate optimized geometry is interpreted and solved without the material model influence (pure fluid standard CFD).

In order to accomplish the proposed methodology, a binary [0,1] pure fluid/solid optimization model called Topology Optimization of Binary Structures (TOBS) is used.

### 3.4 Topology Optimization of Binary Structures (TOBS)

The binary optimization method used in this work is the one described in the work of [Sivapuram and Picelli \(2018\)](#), that has as advantage the absence of intermediate values of the material model (gray scale) due to the use of binary variables, pure solid/fluid values of the material model. This feature facilitates identification of the solid/fluid interface and posterior geometry trimming explained in the later section.

Despite the fact that TOBS has been developed for structural topology optimization it has already been used in the context of fluid flow problems as in the work of [Souza et al. \(2021\)](#) and [Picelli et al. \(2022\)](#).

TOBS is a sequential approximation of integer linear optimization subproblems. The method is derived from the linear part of the first order Taylor's expansion. Considering the control variable  $\phi$ , the discrete binary control variable  $\Phi$ , the objective function  $\mathbf{F}$ , and the volume restriction  $\mathbf{V}_\Phi$  at the  $k$  optimization iteration.

$$\begin{aligned}\mathbf{F}(\Phi) &\approx \mathbf{F}(\Phi^k) + \frac{d\mathbf{F}(\Phi^k)}{d\Phi} \cdot \Delta\Phi^k + O(\|\Delta\Phi^k\|_2^2), \\ V_\Phi(\Phi) &\approx V_\Phi(\Phi^k) + \frac{dV_\Phi(\Phi^k)}{d\Phi} \cdot \Delta\Phi^k + O(\|\Delta\Phi^k\|_2^2),\end{aligned}\tag{3.10}$$

where  $O(\|\Delta\Phi^k\|_2^2)$  is the truncation error and  $\Delta\Phi^k$  represents the changes of the control variable.

In order to keep the control variable with binary values,  $\Delta\Phi^k$  needs to be restricted by:

$$\begin{cases} -1 \leq \Delta\phi_j^k \leq 0 & \text{if } \phi_j^k = 1, \\ 0 \leq \Delta\phi_j^k \leq 1 & \text{if } \phi_j^k = 0, \end{cases}\tag{3.11}$$

or, in a compact notation:

$$\Delta\phi_j^k \in \{-\phi_j^k, 1 - \phi_j^k\},\tag{3.12}$$

where  $\Delta\phi_j^k \in \{-1, 0, 1\}$ .

Eqs. 3.11 and 3.12 show that a fluid material ( $\phi_j = 1$ ) is restricted by  $\Delta\phi_j \in \{-1, 0\}$ , which means that it can turn into a solid ( $\Delta\phi_j = -1$  and  $\phi_j = 0$ ) or keep its value ( $\Delta\phi_j = 0$  and  $\phi_j = 1$ ) during the optimization iteration  $k$ . Similarly, a solid material ( $\phi_j = 0$ ) is restricted by  $\Delta\phi_j \in \{0, 1\}$ , which means that it can turn into a fluid ( $\Delta\phi_j = 1$  and  $\phi_j = 1$ ) or keep its value ( $\Delta\phi_j = 0$  and  $\phi_j = 0$ ) during the optimization iteration  $k$ .

The linear approximation from Eq. 3.10 is only valid when  $O(\|\Delta\Phi^k\|_2^2)$  is sufficiently small, therefore an additional constraint is needed. This is accomplished by restricting the number of elements that are allowed to change from fluid to solid and vice-versa.

$$\|\Delta\Phi^k\|_1 \leq \beta_{flip} N_\phi.\tag{3.13}$$

where  $\beta_{flip}$  is a percentage and  $N_\phi$  is the total number of elements.

Using small values of the  $\beta_{flip}$ , the number of changes or flips at each iteration  $k$  is small, which keeps the truncation error also small and ensure the validity of the linear approximation.

Therefore the approximate binary linear optimization subproblem can be written as:

$$\begin{aligned}
& \text{Minimize } \frac{d\mathbf{F}(\Phi^k)}{d\Phi} \cdot \Delta\Phi^k, \\
& \text{Subject to } \frac{dV_\Phi(\Phi^k)}{d\Phi} \cdot \Delta\Phi^k \leq \mathbf{V} - V_\Phi(\Phi^k) := \Delta V_\Phi^k, \\
& \quad \|\Delta\Phi^k\|_1 \leq \beta_{flip} N_\phi, \\
& \quad \Delta\phi_j^k \in \{-\phi_j^k, 1 - \phi_j^k\}, \quad j \in [1, N_\phi].
\end{aligned} \tag{3.14}$$

In order to avoid unfeasible solutions during the optimization, the bound constraint ( $\Delta V_\Phi^k$ ) is modified by:

$$\Delta V_\Phi^k = \begin{cases} -\epsilon V_\Phi(\Phi^k) & : \mathbf{V} < (1 - \epsilon)V_\Phi(\Phi^k), \\ \mathbf{V} - V_\Phi(\Phi^k) & : \mathbf{V} \in [(1 - \epsilon)V_\Phi(\Phi^k), (1 + \epsilon)V_\Phi(\Phi^k)], \\ \epsilon V_\Phi(\Phi^k) & : \mathbf{V} > (1 + \epsilon)V_\Phi(\Phi^k) \end{cases} \tag{3.15}$$

where  $\epsilon$  is a relaxation parameter related to the constraint  $V_\Phi$ .

The relaxation procedure in Eq. 3.15 aids in generating feasible solutions when the initial guess or the optimization iteration is distant from feasibility.

Eq. 3.14 can be solved by means of Integer Linear Programming (ILP) or other integer or binary optimization solver, in this thesis the linear optimization subproblem is solved with the optimization solver CPLEX (developed by IBM®).

## 4 NUMERIC IMPLEMENTATION

In this chapter, all programs and details from the methodology implemented to perform the Topology Optimization of supersonic inviscid compressible fluids are described.

The forward model is solved using the FEniCS libraries, the sensitivities are calculated with *pyadjoint*, the binary optimization solver used is CPLEX, the boundary identification is performed with a python package named *polylidar*, the boundary smooth is performed with signal filters, and a mesh optimization is accomplished using *optimesh*. The optimization loop with an emphasis on the programs used at each step can be visualized in Fig. 4.1.

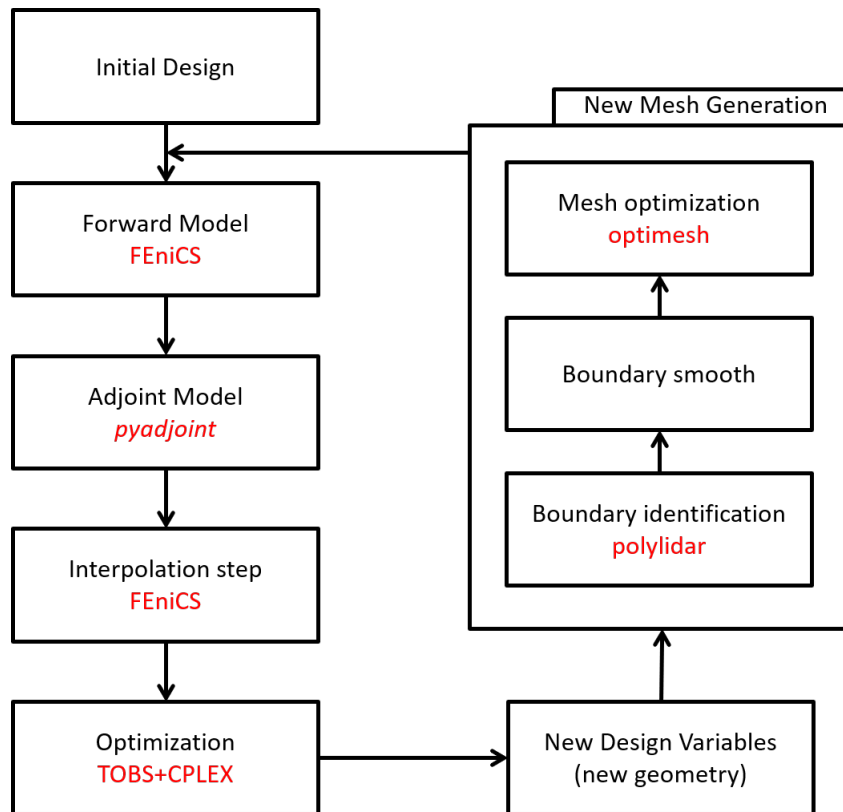


Figure 4.1 – Optimization loop with the main programs used in red.

### 4.1 FEniCS

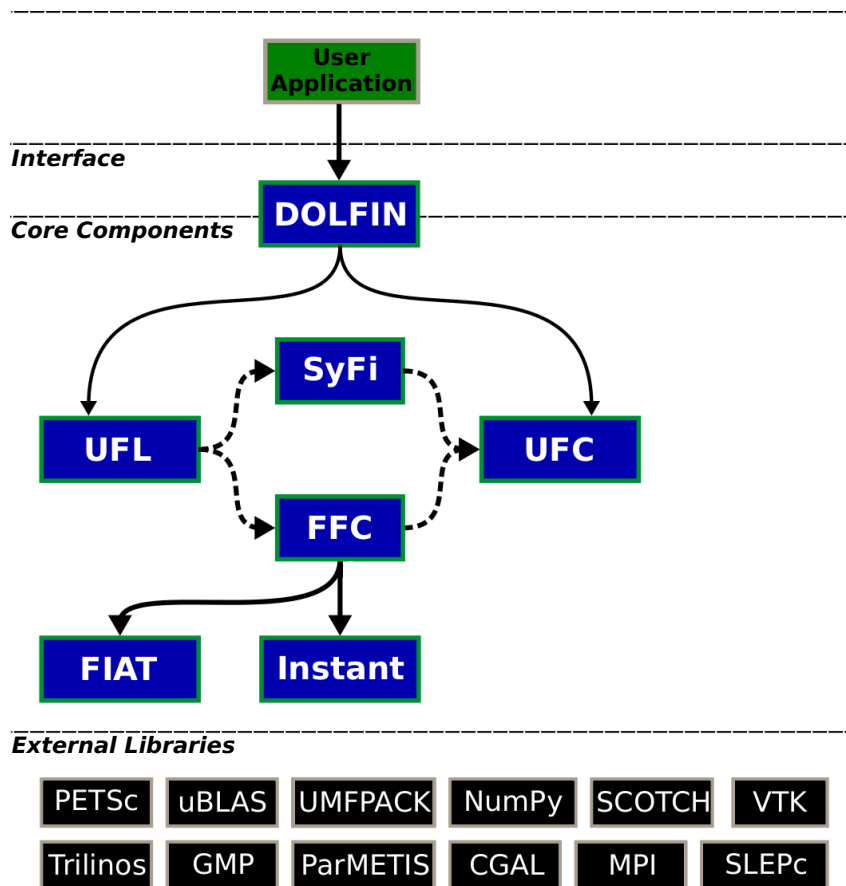
FEniCS<sup>1</sup> is an open-source software developed to automate the solution of mathematical models described by PDEs. It starts from the variational formulation to derive and solve the desired problem by using the FEM.

<sup>1</sup> Available at <<http://fenicsproject.org/>>

User applications in FEniCS can be programmed in *C++* and/or Python, where the latter is the language chosen to implement the computer simulation in the present work.

Essentially, the FEniCS software consists in core components named DOLFIN, FEniCS Form Compiler (FFC), Unified Form Language (UFL), Unified Form-assembly Code (UFC), FInite element Automatic Tabulator (FIAT) and Instant. The basic relationship of these components can be seen in Figure 4.2 and a brief explanation of each one is presented next.

Figure 4.2 – FEniCS components overview



Adapted from (LOGG; MARDAL; WELLS, 2012)

The user application is then implemented by using the DOLFIN package, it functions as a user interface and is a core component of FEniCS, because it is responsible for all the communications between the user program and the other FEniCS components.

DOLFIN is a collection of libraries with efficient *C++* classes developed for finite element computing. All its modules and features can be imported to a Python environment in a single command line (Listing 4.1).

```
1 from dolfin import *
```

Listing 4.1 – Importing the python DOLFIN package

The "dolfin" is a Python package providing access to the all DOLFIN libraries, features and functions. DOLFIN also uses external libraries for the linear algebra computations, such as PETSc, Trilinos and uBLAS, and for the mesh partitioning, such as ParMETIS and SCOTCH.

The mesh must be stored in a specific file format called DOLFIN XML. The reading of a mesh file is performed by the DOLFIN Mesh class and can be visualized in Listing 4.2.

```
1 mesh = Mesh("mesh.xml")
```

Listing 4.2 – Importing the mesh file

The variational forms of the governing equations must be expressed in a suitable form language called Unified Form Language (UFL), that has as high point a very close correspondence between the Python syntax and the mathematical expressions.

For example, the LSFEM formulation of the continuity equation (Eqs. 2.12 and 2.15 are rewritten below for the sake of completeness):

$$\begin{aligned}\mathcal{R}_{continuity} &= \rho - \rho_0 + dt[(\mathbf{u} \cdot \nabla)\rho + \rho(\nabla \cdot \mathbf{u})] \\ \delta\mathcal{R}_{continuity} &= \delta\rho + dt[(\delta\mathbf{u} \cdot \nabla)\rho + (\mathbf{u} \cdot \nabla)\delta\rho + \delta\rho(\nabla \cdot \mathbf{u}) + \rho(\nabla \cdot \delta\mathbf{u})]\end{aligned}$$

Can be written in the UFL Python code as shown in Listing 4.3

```
1 Rc = rho - rho_0 + dt*(dot(u, nabla_grad(rho)) + rho*nabla_div(u) )
2 var_Rc = del_r + dt*(dot(del_u, nabla_grad(rho)) + dot(u, nabla_grad(del_r))
3     + del_r*nabla_div(u) + rho*nabla_div(del_u) )
4 Rr = inner(Rc, var_Rc)*dx
```

Listing 4.3 – LSFEM Continuity equation written in UFL

The variational formulation in the UFL format is passed to the FEniCS Form Compiler (FFC) that generates a code in the Unified Form-assembly Code (UFC) format, in order to facilitate and automate the assembly of the sparse matrices system of equations from the FEM. The FFC depends on the FInite element Automatic Tabulator (FIAT), that contains a library of pre-defined elements, and on the just-in-time compiler Instant.

The FIAT is an extensive list of pre-defined elements, the elements used in this thesis are the nodal Lagrange elements with linear and quadratic interpolations, and the shapes used are triangles or quadrilaterals.

All elements are built and computed, only once, in a standard element  $\Omega_{st}$  with generic coordinates  $\xi_i \in [-1; 1]$  with  $i = 1, 2$  and  $3$ , where the index  $i$  represents the direction

of the variable  $\xi$ . The generic local coordinates  $\xi_i$  of standard element are then mapped to the global Cartesian coordinates  $x_i$  by a linear transformation for each element of the mesh.

Finally, the code, generated by the FFC in the UFC format, can be used by DOLFIN to assemble the systems using a linear algebra backend. In the default installation, FEniCS uses the linear algebra package denominated PETSc (BALAY et al., 1997), to handle the vectors, matrices and system solver methods, such as direct or iterative methods and also eigenvalue solvers.

The solver method used for the nonlinear LSFEM equations is the Newton Method with a quadratic line search procedure. For the linear solver it is used a Generalized Conjugate Residual method (SANAN; SCHNEPP; MAY, 2016) in conjunction of a Multigrid Preconditioner. The preconditioners can be described as a set of algebraic operators used for improving the convergence of the solution. The standard  $Ax = b$  linear system is transformed into  $M^{-1}Ax = M^{-1}b$ , where the fundamental trade off is to choose  $M$  to reduce the convergence rate without increasing the computational cost of solving systems with  $M$ . Further explanation of FEniCS and all its components can be found in the *FEniCS Book* (LOGG; MARDAL; WELLS, 2012).

### 4.1.1 LSFEM Implementation

This section shows the implementation of LSFEM method with Python code examples.

The creation of the function spaces over the mesh for the linear Lagrange elements and the Test Functions are shown in Listing 4.4.

```

1 R1 = FiniteElement("Lagrange", mesh.ufl_cell(), 1) # rho element
2 V2 = VectorElement("Lagrange", mesh.ufl_cell(), 1) # velocity element
3 P1 = FiniteElement("Lagrange", mesh.ufl_cell(), 1) # pressure element
4 element = MixedElement([R1, V2, P1])
5 W = FunctionSpace(mesh, element)
6
7 #Test functions
8 wt = TestFunction(W)
9 (wr, wu, wp) = split(wt) # Get test function for each variable

```

Listing 4.4 – Definition of the function spaces and the Test Functions

The code implementation of the LSFEM can be visualized in Listing 4.5

```

1 # Continuity Equation
2 Rr1 = r-r0 + dt*(dot(u,nabla_grad(r)) + r*nabla_div(u) )
3 Rr2 = wr + dt*(dot(wu,nabla_grad(r)) + dot(u,nabla_grad(wr))
4       + wr*nabla_div(u) + r*nabla_div(wu) )
5 Rr = (C_r)*inner(Rr1,Rr2)*dx # C_r is the weight used to enforce the mass conservation
6
7

```

```

8 #Momentum Equation
9 Ru1 = u-u0 + dt*(dot(u,nabla_grad(u)) + 1.0/r*nabla_grad(p) )
10 Ru2 = wu + dt*(dot(wu,nabla_grad(u)) + dot(u,nabla_grad(wu))
11     + (nabla_grad(wp)*r - nabla_grad(p)*wr)/(r*r) )
12 Ru = inner(Ru1,Ru2)*dx
13
14 #Pressure/Energy Equation
15 Rp1 = p-p0 + dt*(dot(u,nabla_grad(p)) + gama*p*nabla_div(u) )
16 Rp2 = wp + dt*(dot(wu,nabla_grad(p)) + dot(u,nabla_grad(wp))
17     + gama*wp*nabla_div(u) + gama*p*nabla_div(wu) )
18 Rp = inner(Rp1,Rp2)*dx

```

Listing 4.5 – LSFEM implementation

The solver parameters are implemented as shown in Listing 4.6.

```

1 PETScOptions.set('snes_type', 'newtonls') # Newton line search
2 PETScOptions.set('snes_linesearch_type','bt') # Backtracking line search over the L2 norm
3 PETScOptions.set('snes_linesearch_order', 2)
4 PETScOptions.set("snes_max_it", 1000) # Maximum number of iterations for each linear solve
5 PETScOptions.set("snes_rtol", 1.0e-12) # Residual tolerance
6 PETScOptions.set("snes_atol", 1.0e-10) # Absolute tolerance
7 PETScOptions.set('ksp_type','pipegr') # linear solve
8 PETScOptions.set('pc_type','mg') # preconditioner
9 PETScOptions.set("pc_factor_mat_solver_package", "mumps")

```

Listing 4.6 – Solver and preconditioner parameters

The solution of the transient problem is performed in a time-loop, where for each time-step the LSFEM formulation is implemented and solved, and the Python code can be visualized in Listing 4.7.

```

1 while steady_state is not True:
2     F = FEM(w, w0, wt, dt, mesh, C_to, C_r)
3     solve(F == 0, w, bcs, solver_parameters={"nonlinear_solver": "snes"})
4     w0.assign(w) # assign the old solution to the next
5     if abs(w - w0)/dt < steady_state_criteria:
6         steady_state = True

```

Listing 4.7 – Time loop for the transient solution

## 4.2 *pyadjoint*

The adjoints equations described in Sec. 3.1 are calculated with the aid of the *pyadjoint*<sup>2</sup>, an open-source module created and developed to calculate the solution of discrete adjoint equations.

Basically, *pyadjoint* annotates in a tape all operations during the solve of the forward model written in UFL, and automatically derives the adjoint equations. That is possible because after the discretization of the forward model, the associated adjoint model

<sup>2</sup> <<http://www.dolfin-adjoint.org/en/latest/>>



implementation can be obtained in a prescriptive process (FUNKE, 2012), and because the forward model in FEniCS is written as a sequence of linear (or non-linear) equations that are solved consecutively.

### 4.3 TOBS and CPLEX

The TOBS method described in Sec. 3.4 is implemented in Matlab<sup>®</sup> and is interfaced to the main Python program by means of the Octave<sup>3</sup>. The source code used as a guide to implement the TOBS is available at <<https://github.com/renatopicelli/tobs>>.

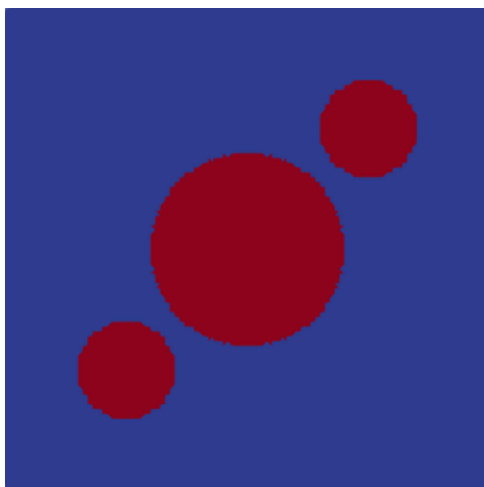
CPLEX<sup>®</sup> is a commercial optimization solver developed by the IBM<sup>®</sup> with a free academic license, its ILP solver uses the branch-and-cut search method. Its algorithm performs a branch-and-bound search with a cutting plane strategy to get rid of the unfeasible solutions.

Sivapuram and Picelli (2020) showed that the main bottleneck of the TOBS is the Finite Element Analysis and not the optimization step, if the  $\beta_{flip}$  and the  $\epsilon$  are kept small due to the decrease size of the tree search needed during the optimization solve.

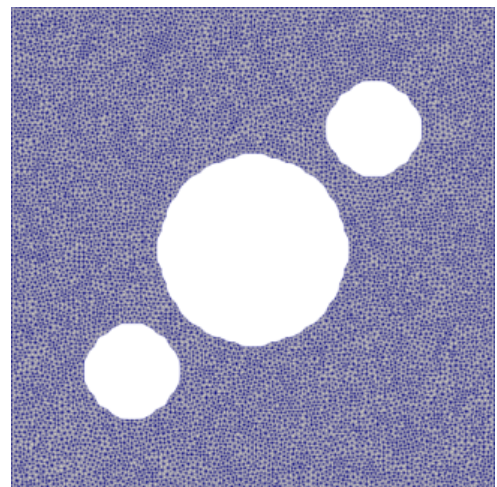
### 4.4 Boundary identification

The boundary identification is performed by using the library *polylidar*<sup>4</sup> (Castagno; Atkins, 2020). It is an efficient algorithm that identifies polygons with holes from point sets. A three circle example is shown next, for the sake of clarification of the method.

Figure 4.3 – Boundary identification and the associated mesh with holes



(a) Three circle test case



(b) Resulting mesh

<sup>3</sup> <<https://octave.org/>>

<sup>4</sup> <<https://github.com/JeremyBYU/polylidar>>

As can be seen in Fig. 4.3b, the boundaries of the holes presented an irregular staircase contour, due to the regular mesh nodes distribution used. This can be problematic in CFD simulations, especially for high speed compressible flows because these irregularities in the boundaries can generate shock waves.

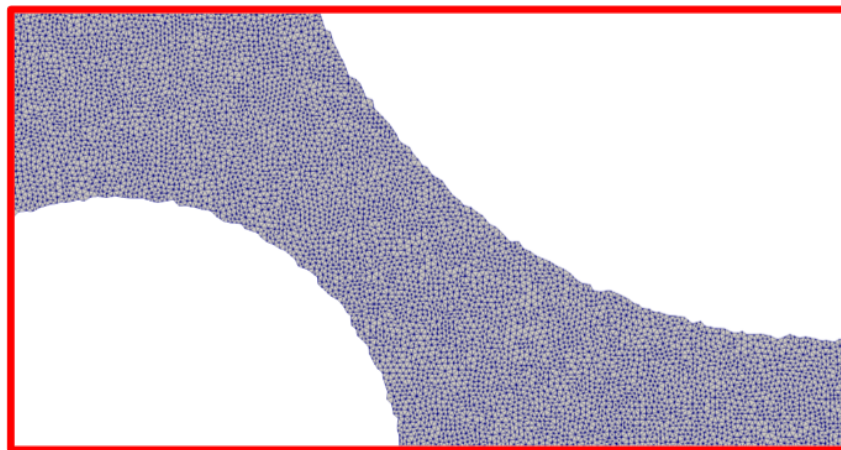
Thus, it makes clear the necessity to perform a boundary smoothing algorithm before using the generated mesh in the computer simulation.

## 4.5 Boundary smoothing

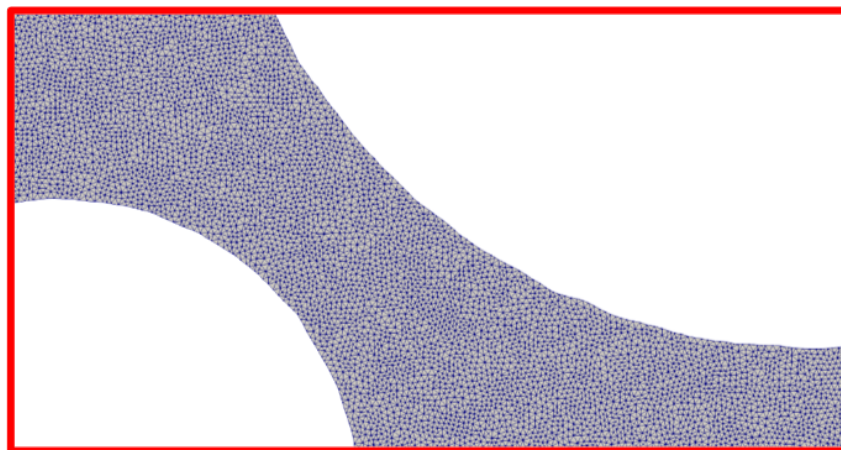
In order to smooth out the boundary irregularities, the Savitzky-Golay (SAVITZKY; GOLAY, 1964) signal filter is used. It is a digital filter based on moving average polynomial curve fitting using a least-square approximation from the adjacent points of the data set.

The smooth algorithm is tested in the same three circle example and using a more refined mesh, the result is shown in Fig.4.4

Figure 4.4 – Boundary smoothing example



(a) Irregular boundary mesh



(b) Smoothed boundary mesh

## 4.6 Mesh optimization

A mesh optimization step is needed to improve the convergence of the forward model, the mesh generated automatically from FEniCS creates irregular triangular elements with varying aspect ratio as can be seen in Fig. 4.5

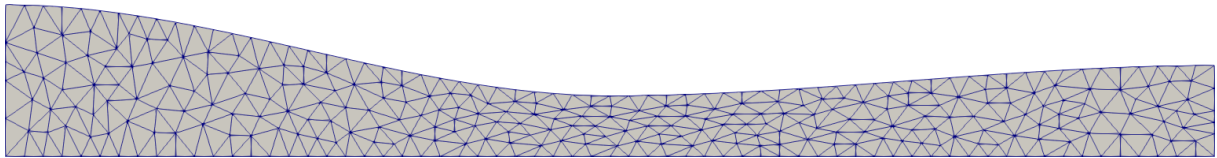


Figure 4.5 – Generic nozzle mesh generated by FEniCS

The nodes of the mesh are reorganized using *optimesh*<sup>5</sup>, a python package that has several mesh optimization methods. The optimized mesh is shown in Fig. 4.6

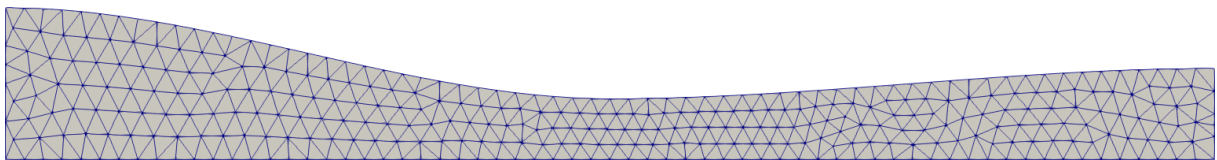


Figure 4.6 – Optimized generic nozzle mesh

*optimesh* successfully rearranged the nodes and created regular elements, improving the mesh quality. It has to be pointed that this mesh optimization test is performed using a coarse mesh, just for visualization purposes.

---

<sup>5</sup> <<https://github.com/meshpro/optimesh>>

## 5 RESULTS

All computer simulations used the FEniCS version 2018.1 running in a Intel Core i7-7700K CPU with 8 cores of 4.2GHz and 32 GB RAM.

### 5.1 Forward solver verification

The implementation of LSFEM applied to the Euler Equations is tested against two cases: a supersonic shock reflection problem, and a normal shock formation inside a nozzle.

The chosen test problems have a steady-state solution, nevertheless a time marching scheme is used to avoid non-physical solutions as explained in Sec. 2.2.1.1. The solutions are considered in the steady-state when all variables achieve a time stabilization using the following:

$$\frac{|var - var_0|}{dt} \leq \varepsilon_{ss} \quad (5.1)$$

where  $var$  are the variables  $\rho$ ,  $\mathbf{u}$  and  $p$ ,  $var_0$  are the variables at the previous time step, and  $\varepsilon_{ss}$  is the steady-state criteria, where the value of  $5 \times 10^{-4}$  is used unless stated otherwise.

In all computer simulations the time-step was chosen as a fraction of the element size, in order to respect the CFL condition for sonic velocities.

#### 5.1.1 Shock Reflection

The first problem tested is a simple shock reflection problem that can be visualized in Fig. 5.1. [Taghaddosi et al. \(1999\)](#) and [Pontaza et al. \(2004\)](#) also implemented the LSFEM applied to inviscid Euler Equations and simulated this problem.

The left boundary condition is:

$$\begin{aligned} \rho_1 &= 1.0 \\ u_1 &= 2.9 \\ v_1 &= 0.0 \\ p_1 &= 0.7143 \\ Ma_1 &= 2.9 \end{aligned}$$

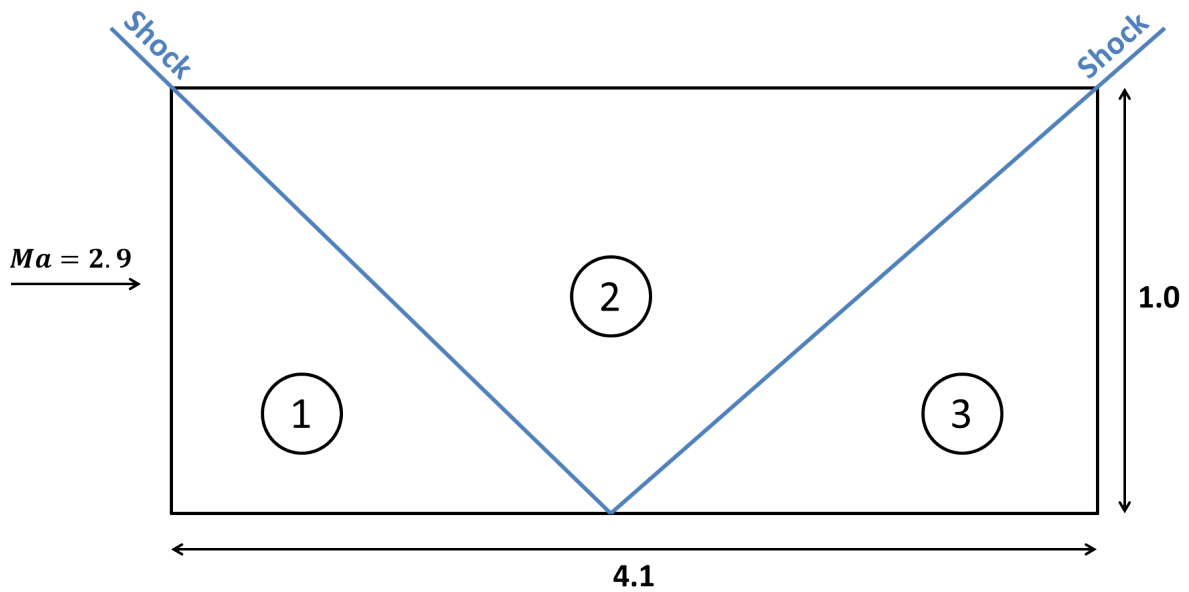


Figure 5.1 – Schematic representation of the shock reflection problem.

And the top boundary condition:

$$\begin{aligned}\rho_2 &= 1.0 \\ u_2 &= 2.9 \\ v_2 &= 0.0 \\ p_2 &= 0.7143 \\ Ma_2 &= 2.3781\end{aligned}$$

These boundary conditions are applied using a Dirichlet condition. The solid wall is implemented using the LSFEM explained in Sec 2.2.1.1.

This problem divides the domain in three distinct regions with known analytical solutions that can be derived from shock relations, the region of interest here is the right most one (number 3 in Fig. 5.1), because the other two are defined by the boundary conditions applied. The analytical solution of region 3 is:<sup>1</sup>

$$\begin{aligned}\rho_3 &= 2.6872 \\ u_3 &= 2.4015 \\ v_3 &= 0.0 \\ p_3 &= 2.934 \\ Ma_3 &= 1.9424\end{aligned}$$

<sup>1</sup> The formulas for the shock relations can be found in Anderson (2011) or Pritchard (2011)

For this simulation a mesh of 60x20 quadrilateral elements is used (Fig. 5.2).

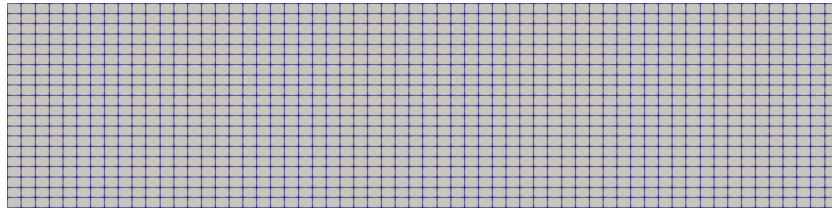


Figure 5.2 – Mesh with 60x20 quadrilateral elements used in the shock reflection problem.

The results for the density, pressure and Mach number can be visualized in Fig. 5.3.



Figure 5.3 – Solution obtained for Density, Pressure and Mach number in the shock reflection problem

The results of the region 3 are analyzed in Tab. 5.1

Table 5.1 – Error analysis of the shock reflection problem

	LSFEM	Analytical	Error(%)
Density ( $\rho$ )	2.7119	2.6872	0.9186
Pressure ( $p$ )	2.9749	2.934	1.3937
Mach number ( $Ma$ )	1.9314	1.9424	0.5661

The next solutions are analyzed in a horizontal line in middle of the domain.

As explained in Sec. 2.2.1.1 the time-step  $dt$  acts as a numerical viscosity, this can be seen in Fig. 5.4, where a higher time-step produces more shock smearing and higher overshoot near the shock.

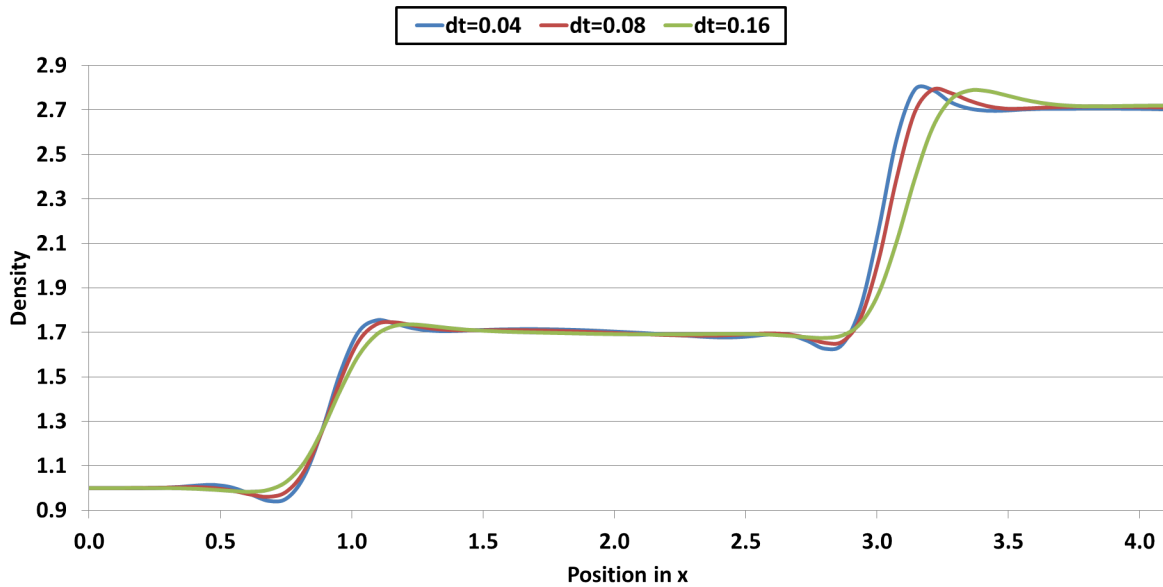


Figure 5.4 – Comparison of the Density solutions considering different time steps.

A mesh convergence study is performed with two more refined meshes with  $90 \times 30$  and  $120 \times 40$  elements. The results are displayed in Fig. 5.5.

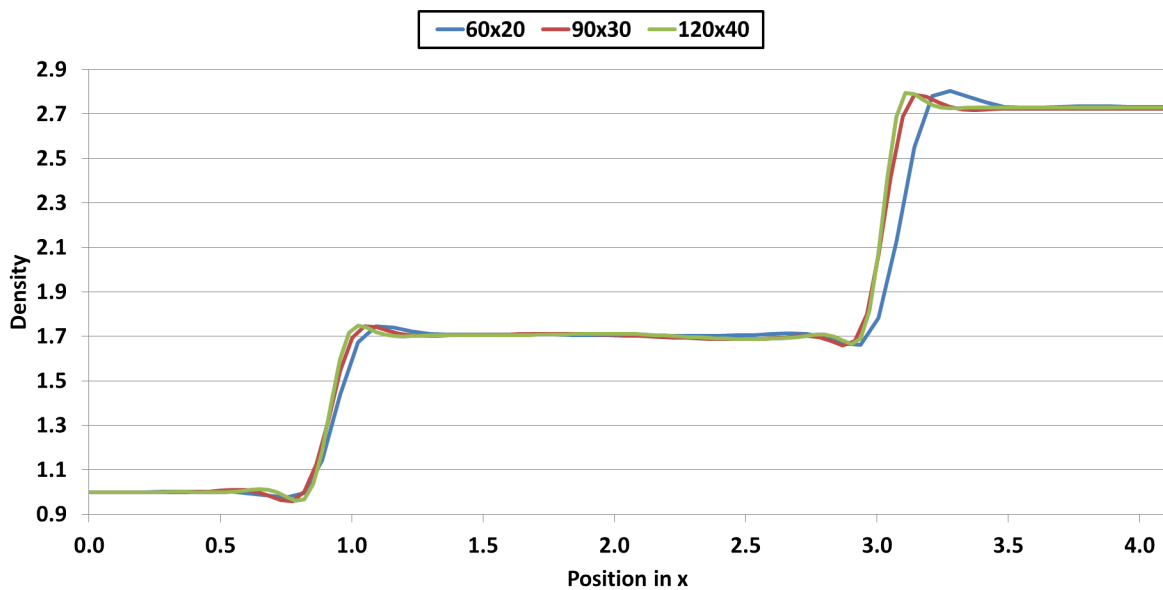


Figure 5.5 – Comparison of the Density solutions considering different mesh refinements.

With the mesh refinement, the shock region gets more defined and sharp. The solution after the oscillations are not affect significantly.

Next, the inclusion of the term enforcing the stagnation temperature conservation is analyzed and can be visualized in Fig. 5.6, where a simulation with and without the stagnation temperature conservation are performed.

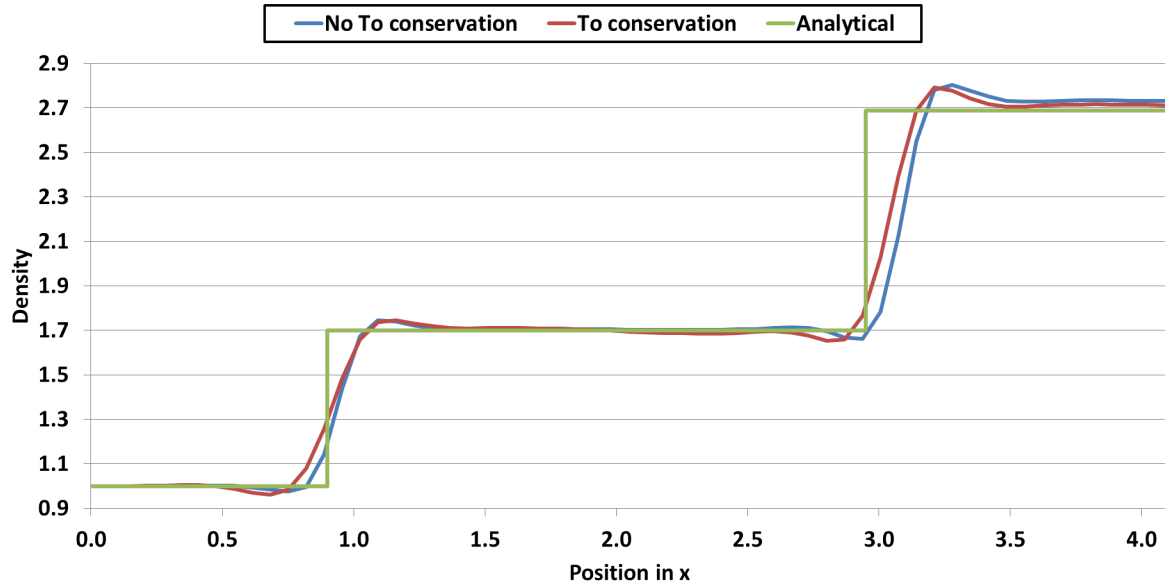


Figure 5.6 – Comparison of the Density solutions considering the stagnation temperature conservation, without the additional term, and the analytical solution.

The formulation with the stagnation temperature term improved the results, and predicts the shock closer to the analytical solution.

### 5.1.2 Normal Shock inside a nozzle

For the normal shock inside a nozzle, the reference used is the work of Arina (2004). The schematic representation of the problem with the boundary conditions used is displayed in Fig. 5.7.

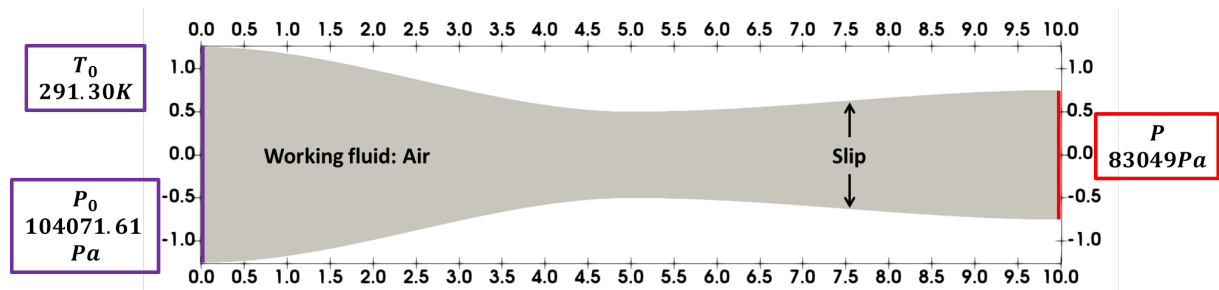


Figure 5.7 – Convergent-divergent nozzle and boundary conditions

This problem consists in a convergent-divergent nozzle with the formation of a normal shock in the divergent part. The minimal area of the nozzle is called throat and for this problem is located at the middle of the geometry ( $x = 5$ ).



This geometry is generated using the formula given in Arina's work:

$$\begin{aligned} A(x) &= 2.5 + 3 \left( \frac{x}{5} - 1.5 \right) \left( \frac{x}{5} \right)^2 & \text{for } x \leq 5 \\ A(x) &= 3.5 - \frac{x}{5} \left[ 6 - 4.5 \left( \frac{x}{5} \right) + \left( \frac{x}{5} \right)^2 \right] & \text{for } x > 5 \end{aligned} \quad (5.2)$$

where  $A(x)$  is the area distribution along the length of the nozzle.

Using an one-dimensional analysis, it is possible to obtain the shock position at 7.0, for the given boundary conditions.

The mesh used has 700 quadrilateral elements and is shown in Fig. 5.8.

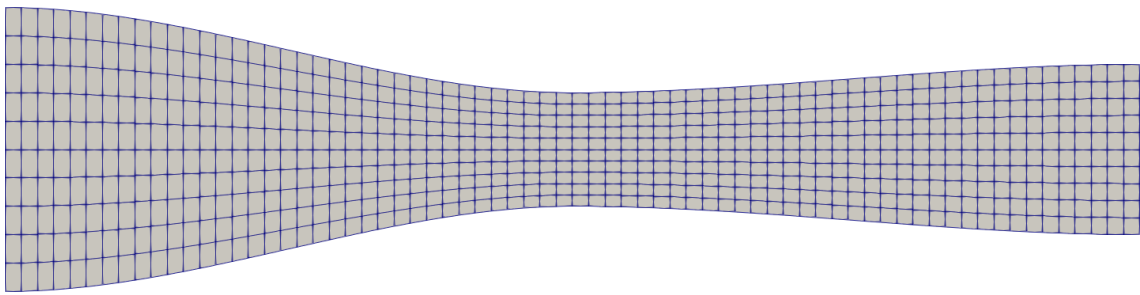


Figure 5.8 – Mesh of 700 quadrilateral elements.

The results for the density, pressure, and Mach number can be visualized respectively in Figs. 5.9, 5.10, 5.11.

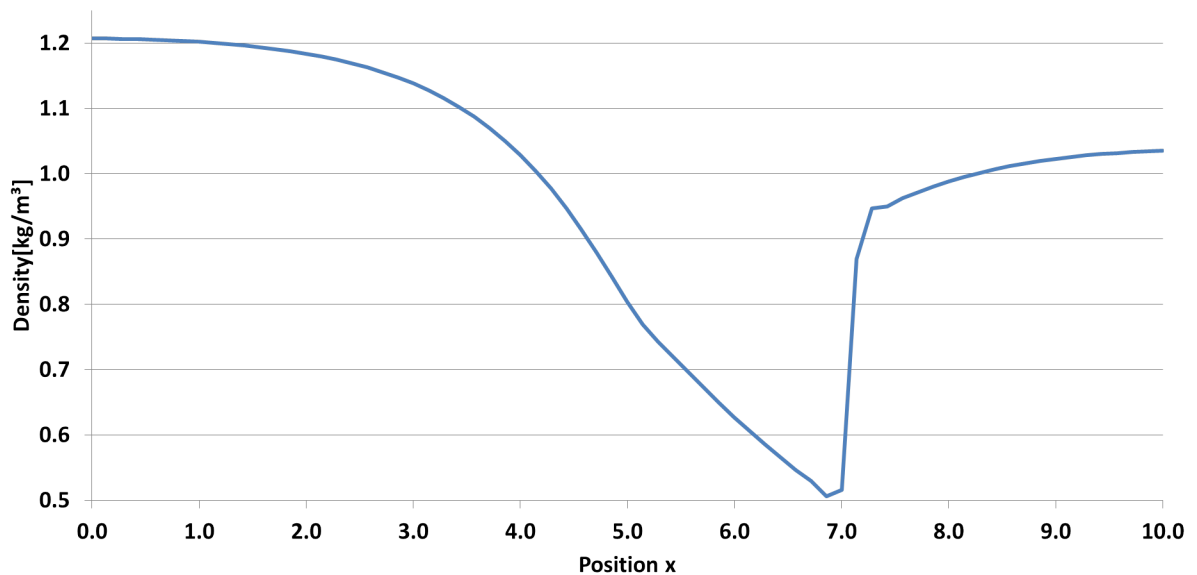


Figure 5.9 – Density results for the Arina's nozzle.

Even for the coarse mesh used, the LSFEM could predict the normal shock near the  $x = 7.0$ , as obtained in Arina (2004).

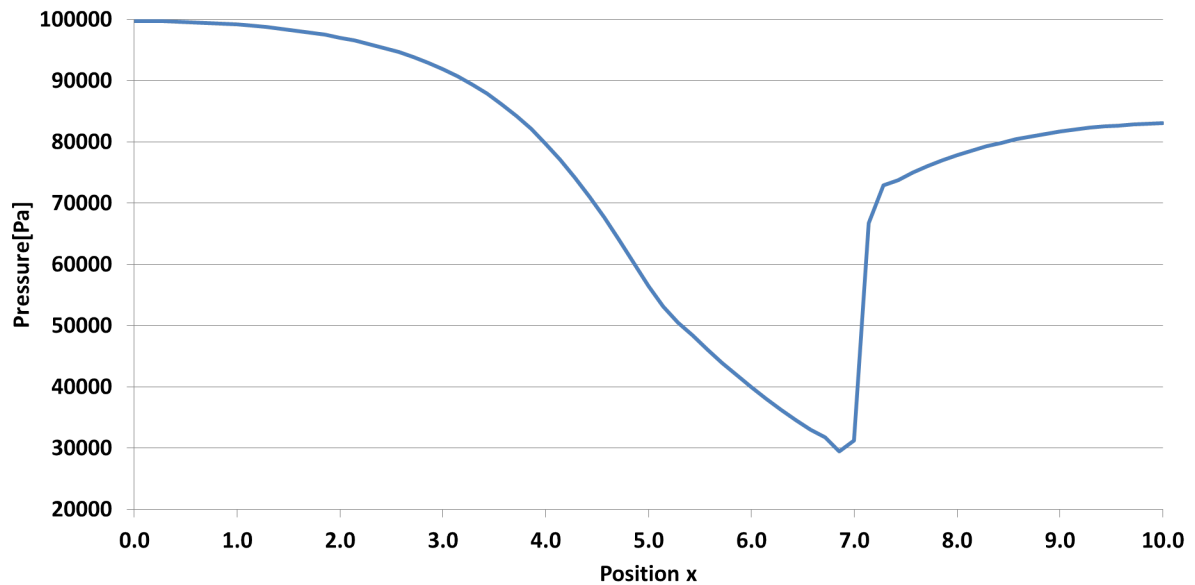


Figure 5.10 – Pressure results for the Arina's nozzle.

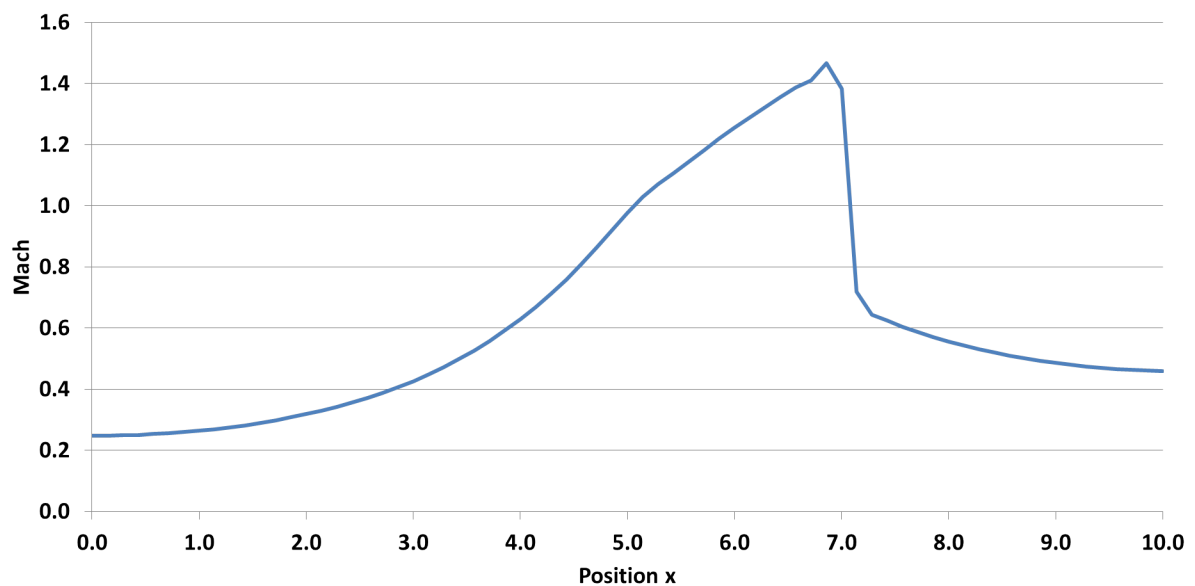


Figure 5.11 – Mach results for the Arina's nozzle.

There are some theoretical fundamentals of transonic compressible flows, that can be observed analyzing the Mach number result inside the nozzle. First, the subsonic flow  $Ma < 1$  accelerates in the convergence part. Then the flow achieves the sonic velocity ( $Ma = 1$ ) at the throat and continues to accelerate in the divergent part, due to the area expansion. Finally, after the shock, the flow returns to the subsonic velocity. This is an indicative that the compressible solver implemented using the LSFEM is able to grasp the fundamental physics of the problem, for the hypothesis considered, such as a perfect, inviscid fluid.

Another important aspect that needs attention is the stagnation temperature, for this problem it should be constant in the entire domain. The result for the Stagnation Temperature is displayed in Fig. 5.12.

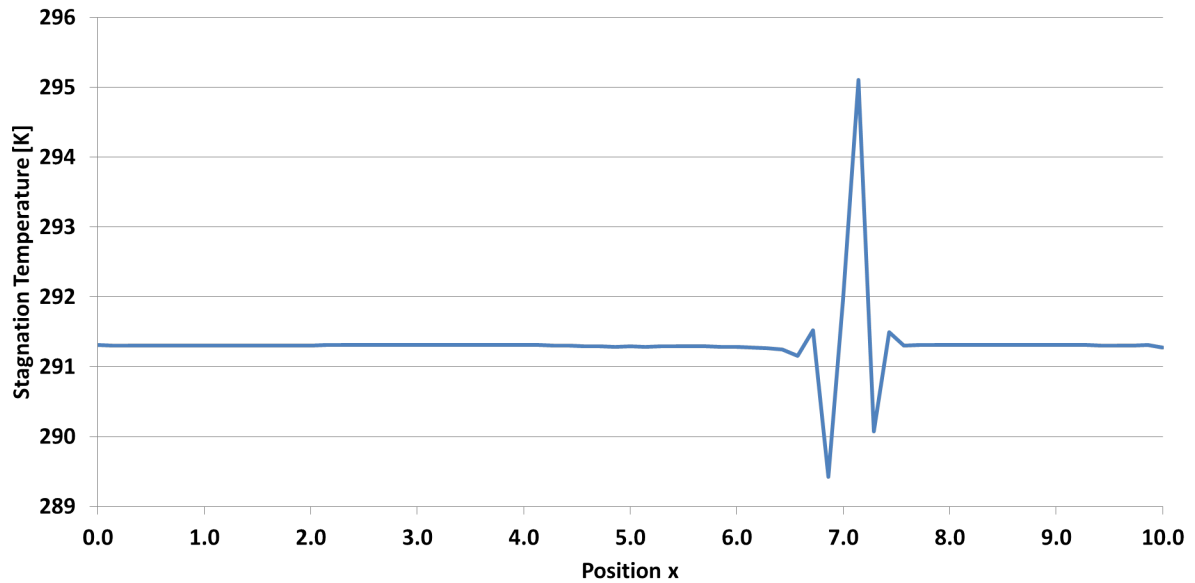


Figure 5.12 – Stagnation temperature results for the Arina's nozzle.

An oscillation of the stagnation temperature in the vicinity of the shock is observed. The relative error can be visualized in Fig. 5.13, where the maximum error of 1.3% occurred only in the shock region and is not significant.

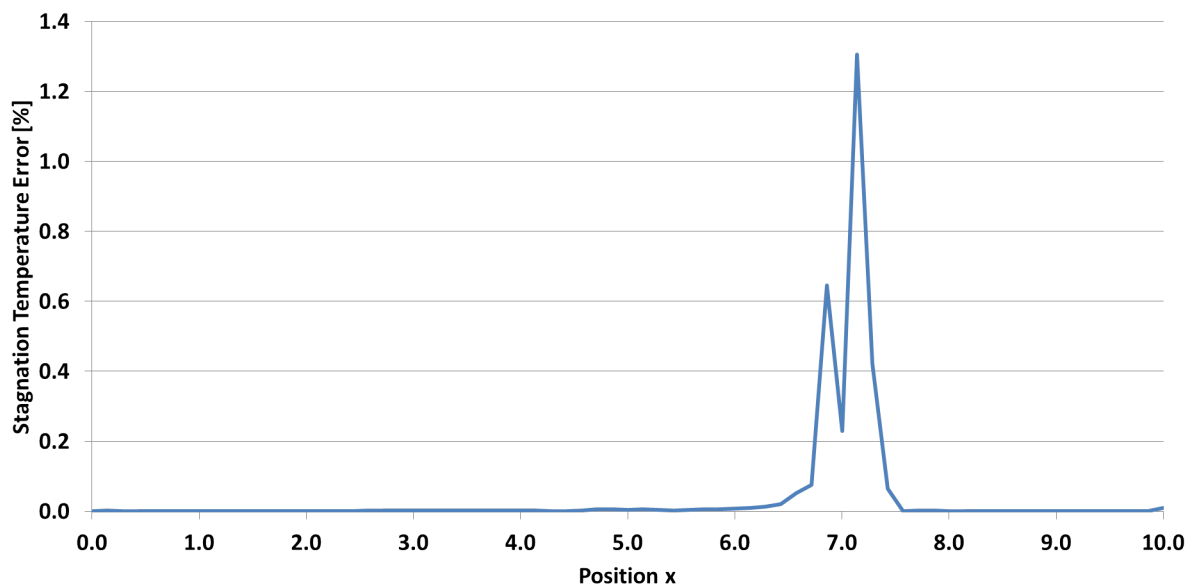


Figure 5.13 – Stagnation temperature error results for the Arina's nozzle.

For this problem the Stagnation Pressure should present two flat levels or plateaus with different values before and after the shock. This behavior can be observed in Fig 5.14.

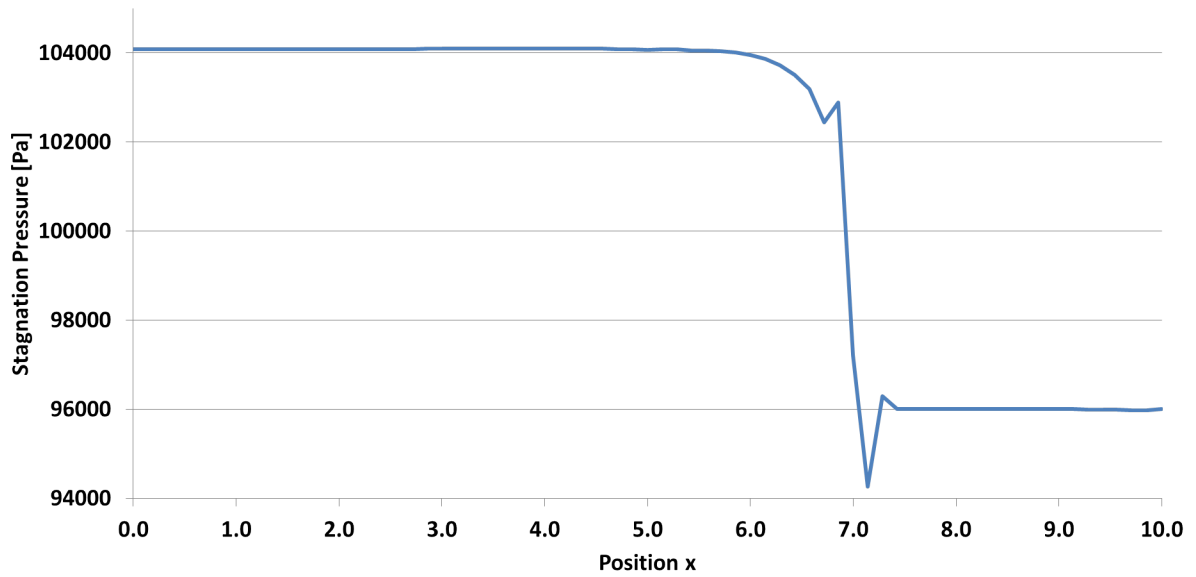


Figure 5.14 – Stagnation pressure results for the Arina’s nozzle.

Next, a mesh convergence test with different refinement levels is performed. The number of elements in each mesh is summarized in Tab. 5.2, and the results of the Mach number for all meshes are presented in Fig. 5.15.

Table 5.2 – Number of elements of each mesh used in the mesh convergence test

n	elements
1	700
2	2800
3	6300
4	11200

Fig. 5.16 displays the Mach results closer the shock region. The refinement of the meshes increased the sharpness of the shock, and approximated the shock to the position predicted by Arina ( $x = 7.0$ ).

The influence of the stagnation temperature term added to LSFEM is observed in Fig. 5.17. These results were obtained using the most refined mesh n4.

It is clear that the LSFEM with the inclusion of the stagnation temperature term achieved better results. Without this term the shock position is misplaced upstream and the stagnation temperature is not conserved downstream the shock.

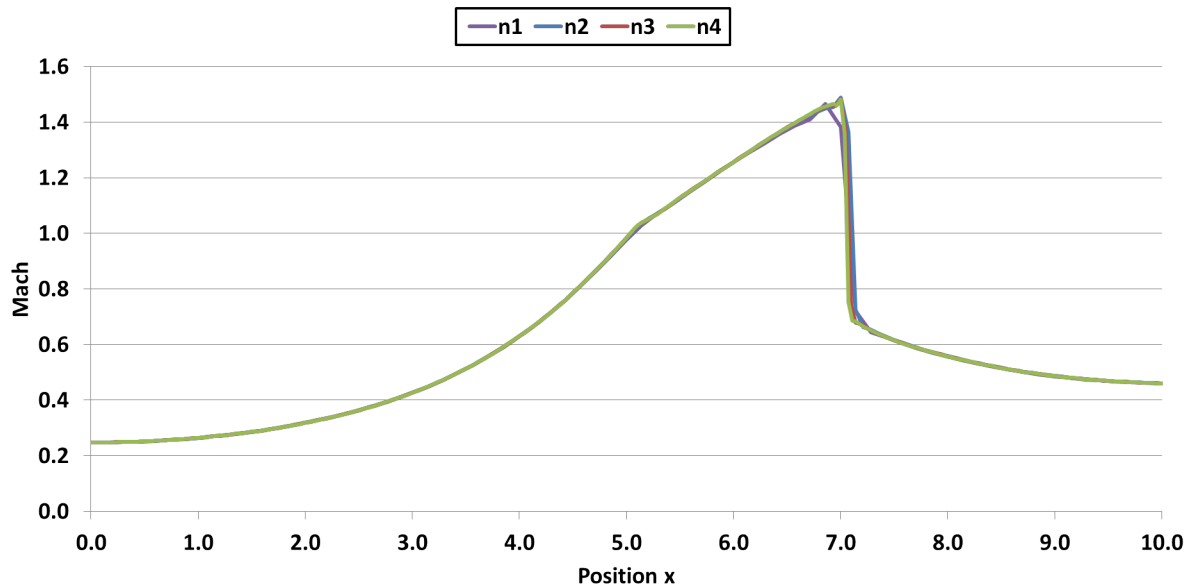


Figure 5.15 – Mach number results for four mesh with different refinement levels.

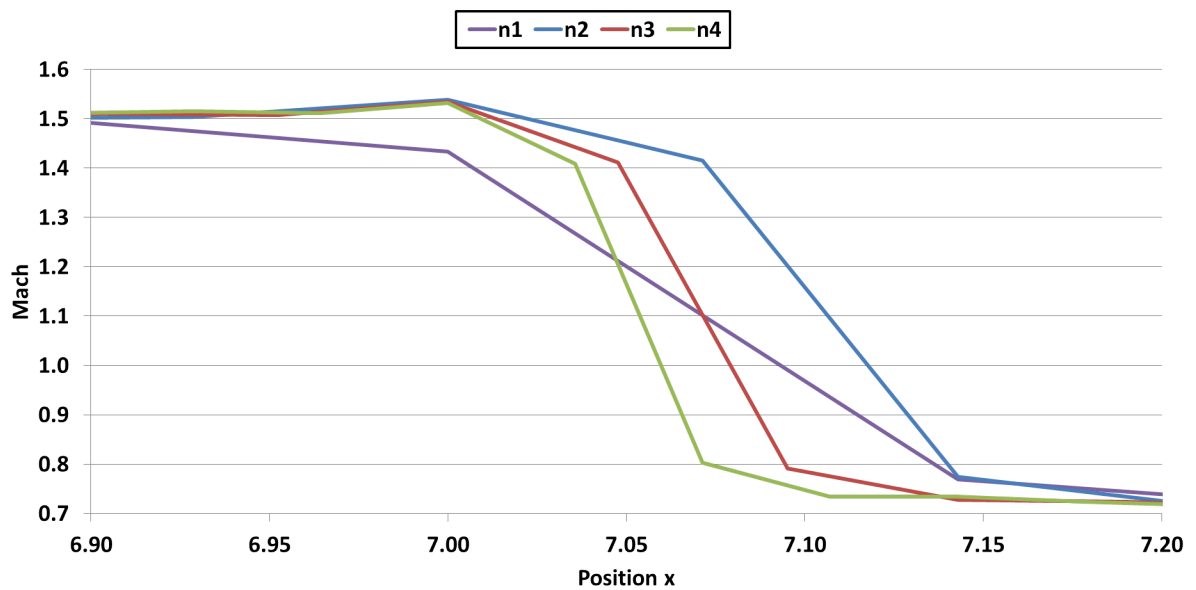


Figure 5.16 – Detail of shock position for different refinement levels

### 5.1.3 Comparison with consolidated CFD softwares

The Euler LSFEM implemented in FEniCS is compared with three CFD programs that uses different numerical schemes to solve compressible fluid problems. Next a concise explanation about each program tested is presented:

**FLUENT**<sup>2</sup>: A commercial package that uses the Finite Volume Method and is widely used for industrial applications.

<sup>2</sup> <[www.ansys.com/products/fluids/ansys-fluent](http://www.ansys.com/products/fluids/ansys-fluent)>

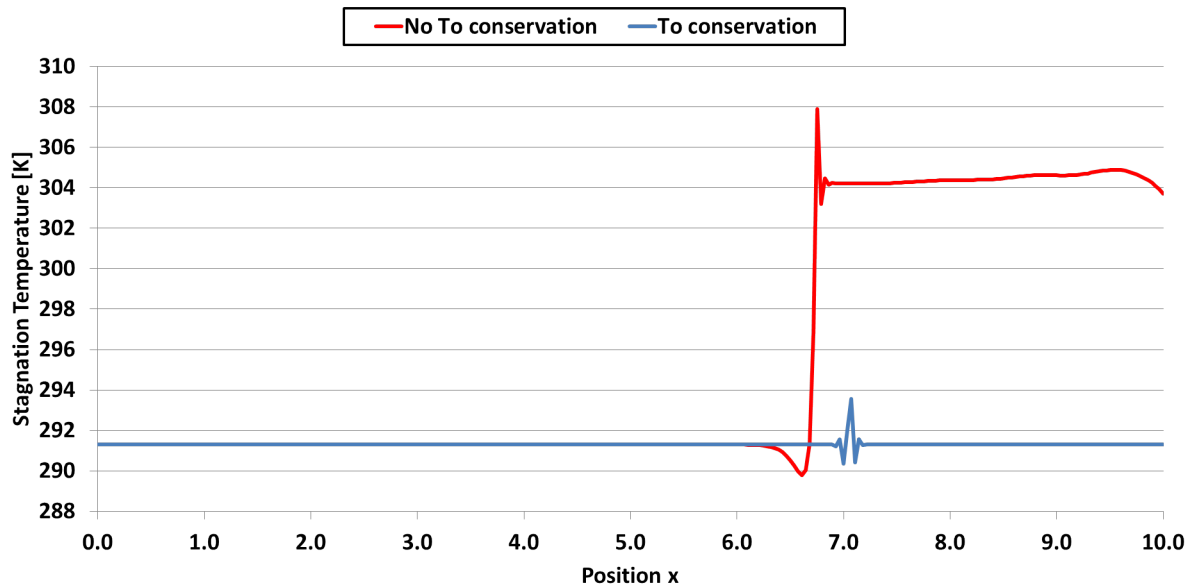


Figure 5.17 – Comparison of the influence of the stagnation temperature in the LSFEM formulation.

**SU2**<sup>3</sup>: Stanford open-source FVM designed specifically for compressible fluid flow problems and shape optimization problems.

**Nektar++**<sup>4</sup>: High order FEM framework with Discontinuous Galerkin elements.

The same Arina problem is used for this comparison, all computer simulations are performed using the same mesh with the same nodes locations, and the results show that all software reach the same solution for all variables analyzed. Only the Mach number results are displayed in Fig. 5.18, for the sake of simplicity.

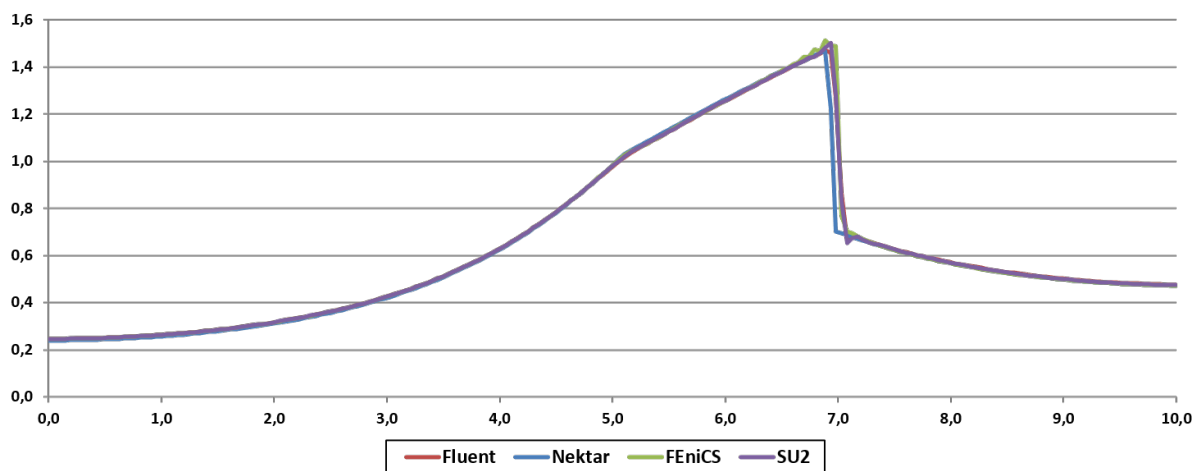


Figure 5.18 – Comparison of Mach number for different software

<sup>3</sup> <<https://su2code.github.io/>>

<sup>4</sup> <[www.nektar.info](http://www.nektar.info)>

## 5.2 Topology Optimization

This section shows the results of the proposed methodology applied to a supersonic inviscid fluid considering as initial guess the bidimensional convergent–divergent nozzle of the Arina’s work in order to have the presence of a normal shock in the first optimization iteration. A volume restriction is used to keep the volume constant.

### 5.2.1 Optimization Loop

The first optimization iteration is shown in details in a coarse mesh resolution, just to clarify the method developed. The optimization loop used is shown again and can be visualized in Fig. 5.19

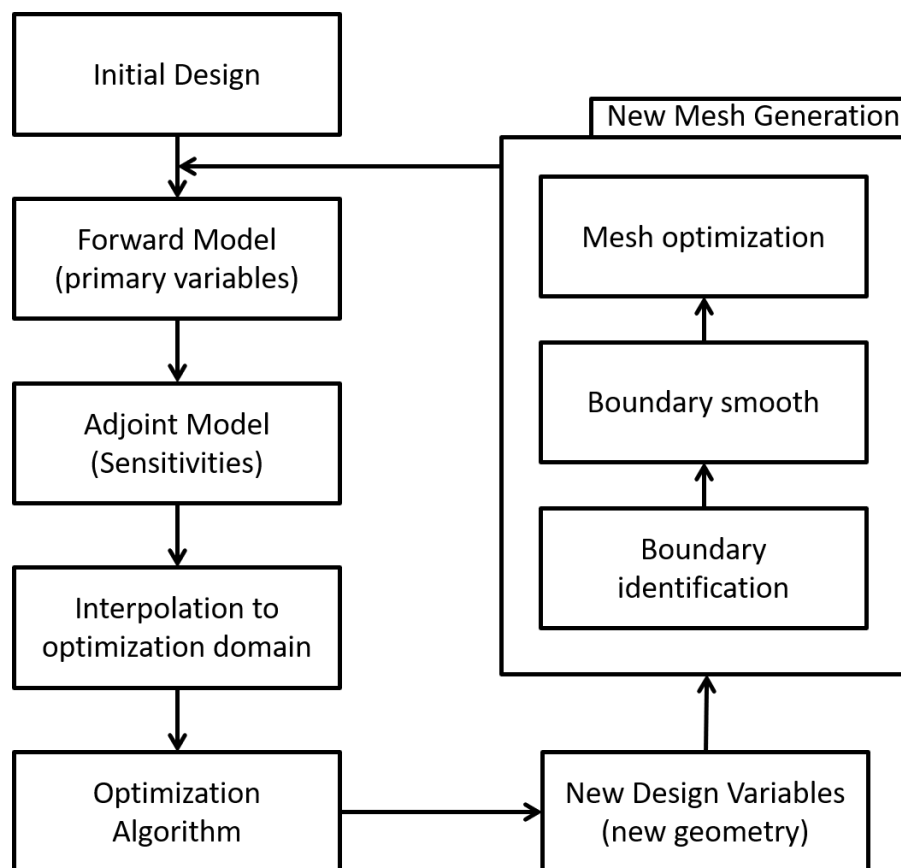


Figure 5.19 – Optimization loop.

The main meshes used are displayed in Fig. 5.20. The top mesh (a) is where the forward and adjoint models are solved, the middle mesh (b) is where the optimization is performed, and the bottom mesh (c) is used for the boundary identification.

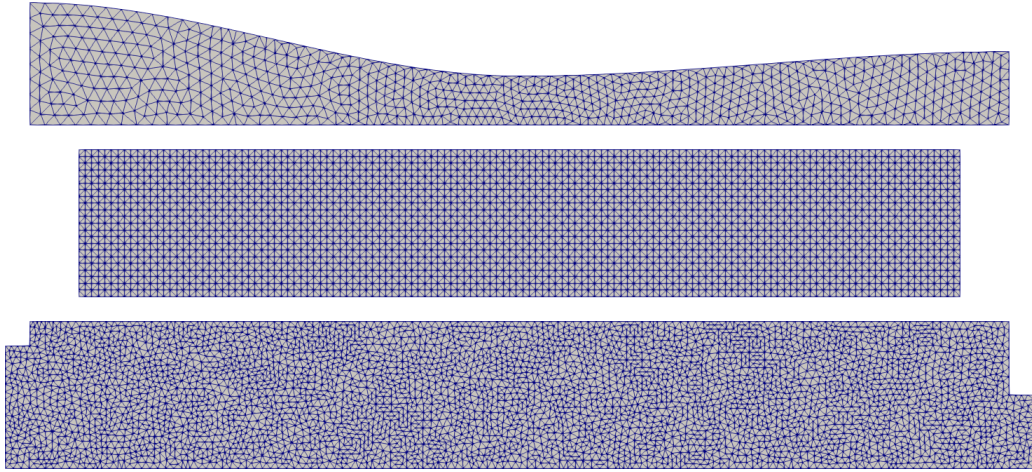


Figure 5.20 – Different meshes used in the optimization iteration

The inlet and outlet regions are not considered in the optimization step to ensure that these regions does not change their sizes, reason why the optimization mesh (b) in Fig. 5.20 is shorter.

The sensitivities are calculated in the first mesh (a) and interpolated to the optimization mesh (b) (Fig. 5.21).



Figure 5.21 – Sensitivity interpolation between meshes

After the optimization step, the new design variables are calculated in the middle mesh (b) and interpolated to the bottom mesh (c).

The left part of Fig 5.23 shows the region where the optimization altered the design in the optimization mesh, and the right part shows the effect of the boundary identification and boundary smoothing.

The final step of the optimization loop is the mesh optimization and is displayed in Fig. 5.24. The optimized mesh has more regular elements that improves the convergence of the forward model.





Figure 5.22 – Optimization interpolation between meshes

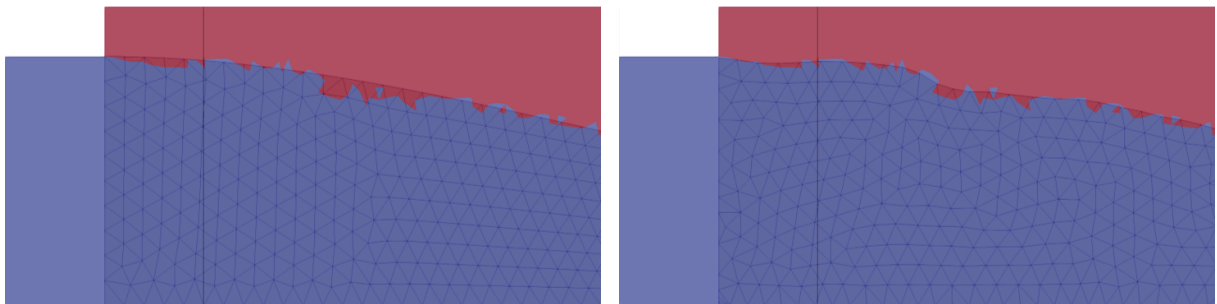


Figure 5.23 – Detail of the inlet region after one optimization step

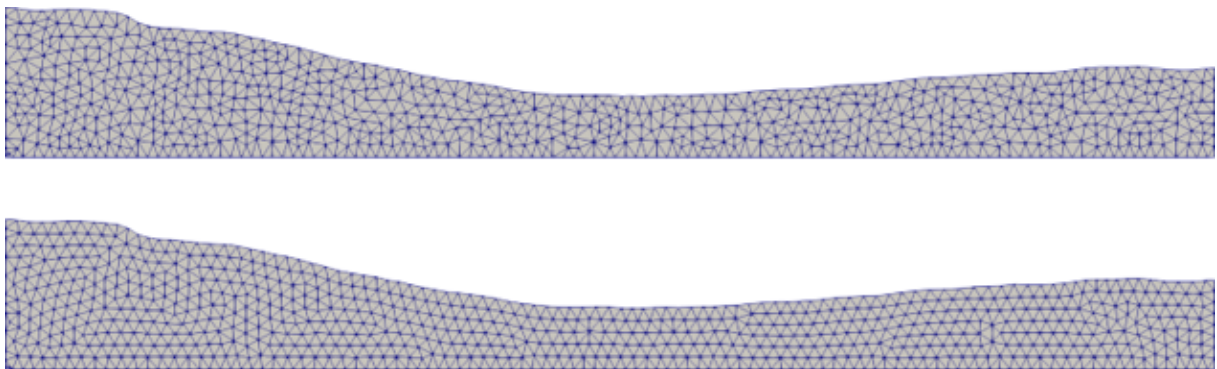


Figure 5.24 – Effect of the mesh optimization

The proposed optimization loop is implemented with three different objective functions: the minimization of Pressure, the maximization of the Mach number, and the minimization of the Temperature.

## 5.2.2 Minimization of the Pressure

The first optimization performed is the minimization of the pressure inside the nozzle with:

$$F_{obj} = \int_{\Omega} p^2 dx \quad (5.3)$$

Which generated the following optimized geometry (Fig. 5.25):

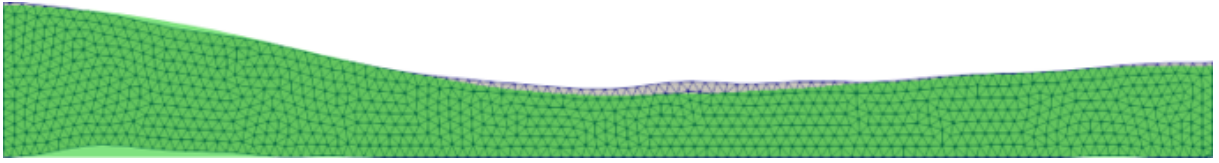


Figure 5.25 – Optimization result obtained from the minimization of the pressure inside the nozzle, the green region is the initial guess

A symmetry boundary condition is used at the bottom plane, and therefore the optimized geometry can be interpreted as shown in Fig. 5.26. This shows an inner body inside the nozzle near the inlet.

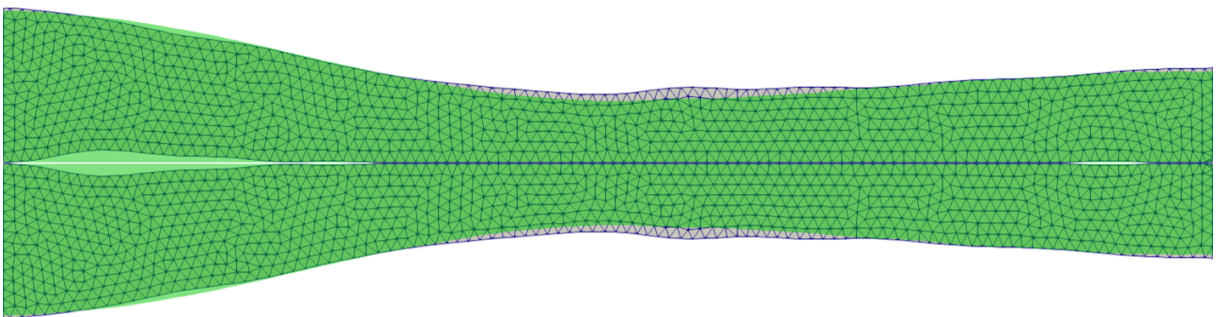


Figure 5.26 – Geometry Optimization result obtained from the minimization of the pressure inside the nozzle

The variables plots are shown in Fig. 5.27. For this case, successive shock reflections happened inside the nozzle due to the oscillations in the boundary, probably caused by the boundary smoothing operator associated with the coarse mesh used in this simulation.

The objective function altered from  $F_i = 2.66$  to  $F_f = 2.44$ , which represents a minimization of 8.08%.

### 5.2.3 Maximization of the Mach number

The next objective function implemented is the maximization of the Mach number, in order to improve the velocity inside the nozzle.

$$F_{obj} = - \int_{\Omega} Ma^2 d\mathbf{x} \quad (5.4)$$

Some of the results for this case fail to reach a stable solution, the optimization seems to skip the local minimum and the flow inside the nozzle became subsonic as can be seen in Fig. 5.28

However, one intermediate optimization step solution resulted in a nozzle with an optimized geometry as can be seen in Fig. 5.29 with an improved Mach distribution inside

the nozzle shown in Fig. 5.30.

The objective function altered from  $F_i = -3.31$  to  $F_f = -4.13$ , which represents an improvement of 24.92%.

The optimization changed the throat and violated the volume restriction for this case, probably due to the successive interpolations between the meshes.

One manner to circumvent this problem is to restrict the throat variation considering only the convergent part of the optimization mesh as can be seen in Fig. 5.31. However the results did not showed any improvement (Fig. 5.32).

## 5.2.4 Minimization of the Temperature

The last objective function implemented is the minimization of the temperature inside the nozzle.

$$F_{obj} = \int_{\Omega} T^2 d\mathbf{x} \quad (5.5)$$

The topology optimization result can be seen in Fig. 5.33.

During the optimization of this case the intermediate solutions presented holes and disconnected islands as can be seen in Figs. 5.34 and 5.35. Nevertheless the method implemented is robust enough to deal with these geometric features.

The temperature and Mach distributions are presented respectively in Figs. 5.36 and 5.37.

The objective function of this case altered from  $F_i = 3.40$  to  $F_f = 3.02$ , which represents an improvement of 11.14%.

For this case, the optimization moved the throat and the normal shock upstream, with a shorter convergent part and a long and almost flat divergent part.

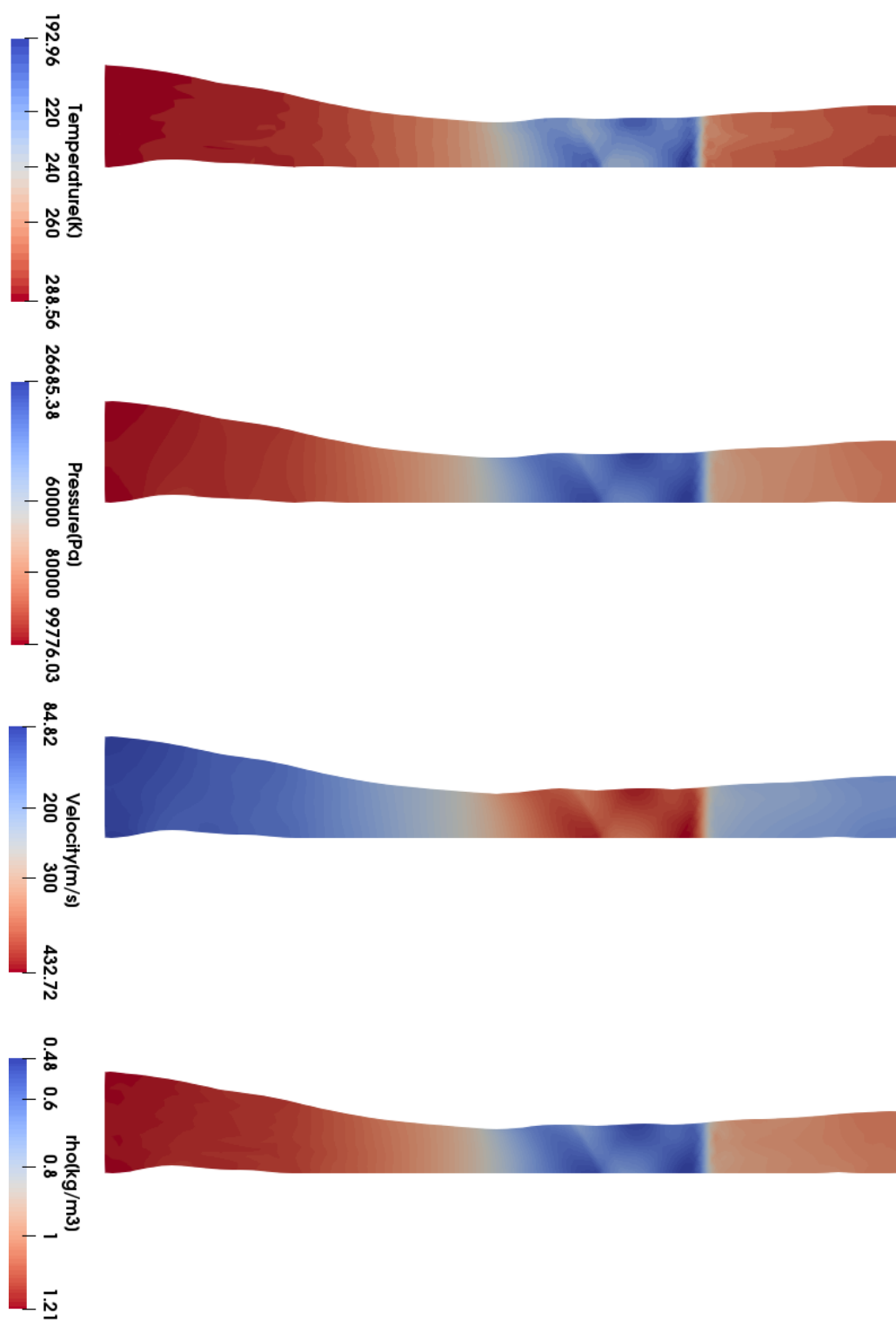


Figure 5.27 – Variables plot result obtained from the minimization of the pressure inside the nozzle

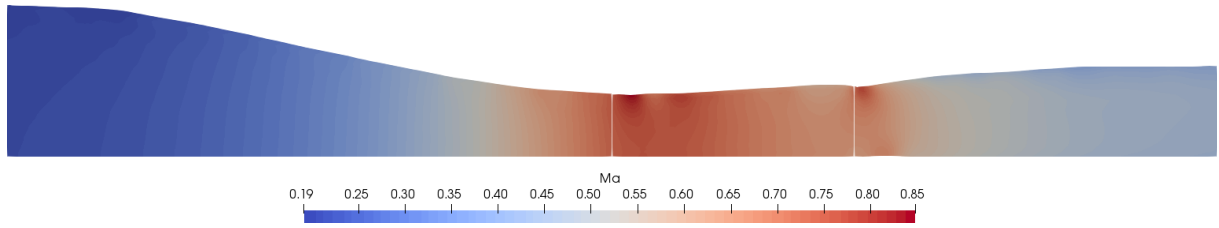


Figure 5.28 – Solution became subsonic and lost the shock wave

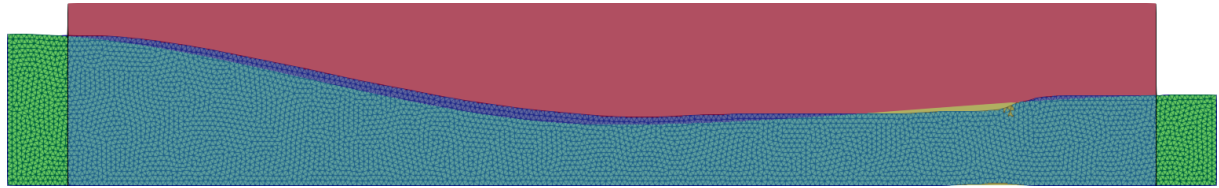


Figure 5.29 – Intermediate optimized geometry

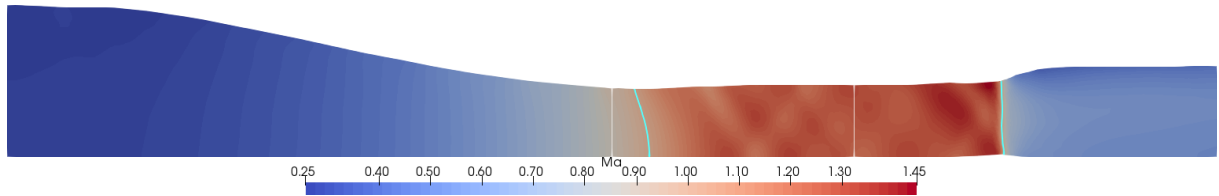


Figure 5.30 – Intermediate optimized Mach number solution

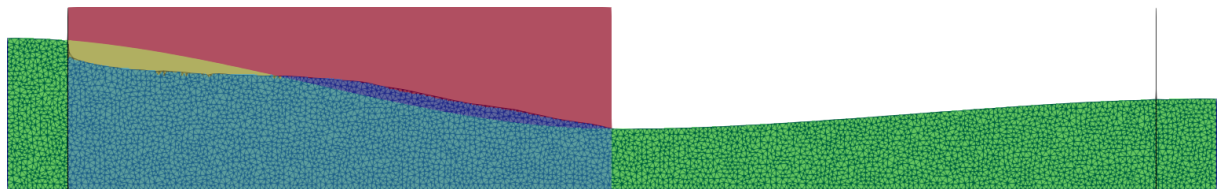


Figure 5.31 – Using half of optimization mesh

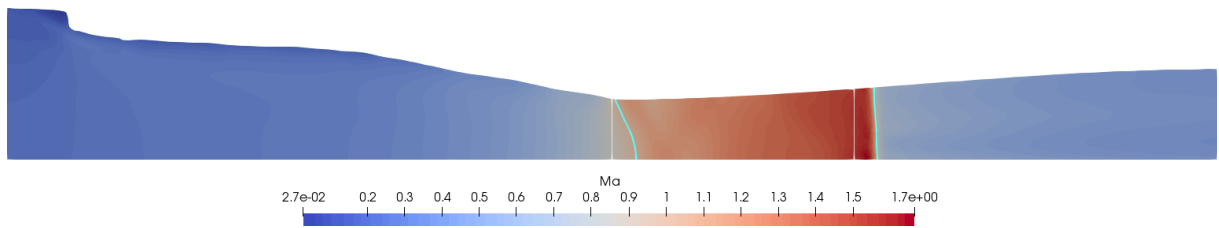


Figure 5.32 – Mach distribution Using half of optimization mesh

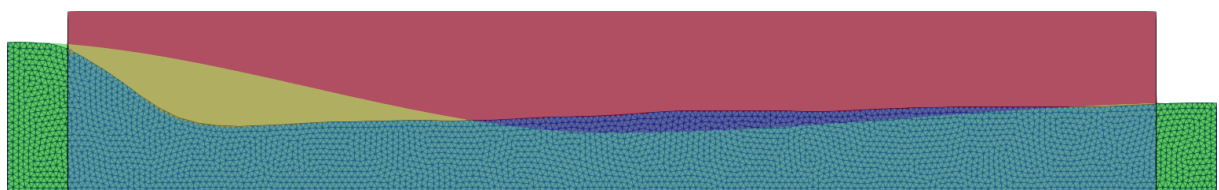


Figure 5.33 – Topology Optimization result for the minimization of temperature

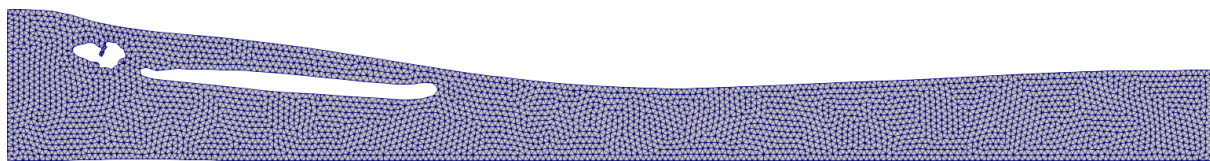


Figure 5.34 – Intermediate optimization result with the presence of holes

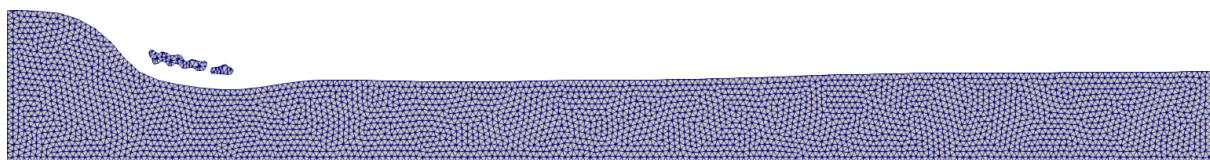


Figure 5.35 – Intermediate optimization result with the presence of islands

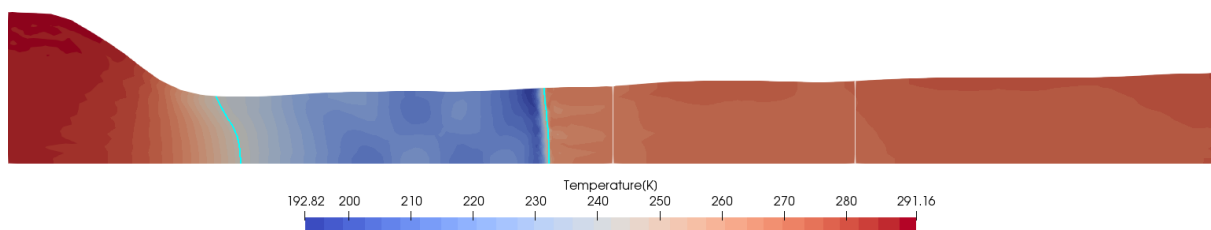


Figure 5.36 – Temperature distribution of the optimization result

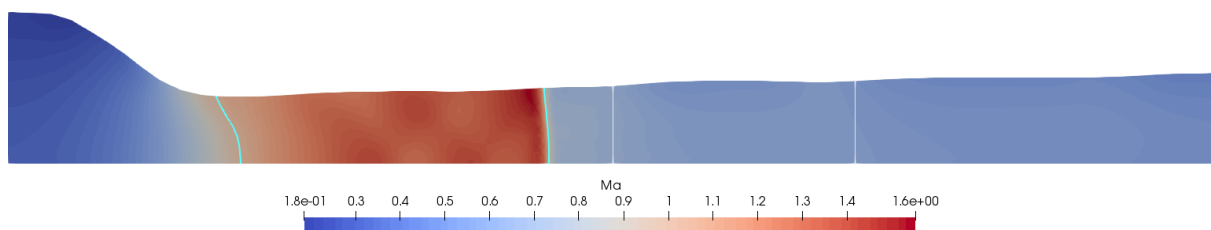


Figure 5.37 – Mach distribution of the optimization result

## 6 CONCLUSIONS

The topology optimization of supersonic inviscid compressible fluid is performed in a bidimensional convergent-divergent nozzle with different objective functions. The results in Chapter 5 shows that the methodology is robust enough to deal with the presence of shock waves, and generated optimized geometries for the objective functions implemented.

Regarding the forward model, the LSFEM applied to the non-conservative formulation of the Euler Equations, with the addition of the stagnation temperature conservation term, showed to predict the right shock position for the cases tested.

Decouple the optimization from the forward model showed to be a workaround strategy to correct implement the suitable boundary conditions without any modifications to the material model.

The successive interpolations between the different meshes and the boundary smoothing operation can aid the optimization, at the cost of more parameters to tune.

### 6.1 Future Work

This thesis opened a range of questions that still require investigation and research:

- In regard to the material model, enforce the zero velocity at the solid region with the addition of the Darcy term in the momentum equation is an inadequate method for the compressible case, and therefore, the development of an inviscid compressible material model is still needed.
- The full effects of the decoupling procedure implemented are not clear yet. The influence of the successive interpolations between the meshes and the influence of the boundary smoothing operation on the optimization needs to be evaluated.
- In theory, the methodology can be extended to investigate the viscosity effects and consider the full compressible Navier-Stokes equations as well as to include turbulence models in the formulation.
- The implementation the three dimensional case is not straight forward, due to the boundary identification, the boundary smoothing and the mesh generation algorithms used in this thesis.

# REFERENCES

- ALEXANDERSEN, J.; ANDREASEN, C. S. A review of topology optimisation for fluid-based problems. **Fluids**, MDPI, v. 5, n. 1, p. 29, 2020. Cited in page [20](#).
- ALONSO, D. H.; SILVA, E. C. N. Topology optimization applied to the design of tesla-type turbine devices. **Applied Mathematical Modelling**, Elsevier, v. 103, p. 764–791, 2022. Cited in page [20](#).
- ANDERSON, J. D. **Fundamentals of Aerodynamics**. 5th. ed. [S.l.]: McGraw-Hill, 2011. 1128 p. ISBN 978-007-128908-5. Cited 4 times in pages [23](#), [25](#), [32](#), and [52](#).
- ARINA, R. Numerical simulation of near-critical fluids. **Applied Numerical Mathematics**, v. 51, n. 4 Special Issue, p. 409–426, 2004. ISSN 01689274. Cited 2 times in pages [55](#) and [56](#).
- BABUSKA, I. Error-bounds for finite element method. **Journal of Numerical Mathematics**, Springer-Verlag New York, Inc., Secaucus, NJ, USA, v. 16, n. 4, p. 322–333, jan. 1971. ISSN 0029-599X. Cited in page [28](#).
- BALAY, S.; GROPP, W. D.; MCINNES, L. C.; SMITH, B. F. Efficient Management of Parallelism in Object Oriented Numerical Software Libraries. In: BRUASET, A.; ARGE, E.; LANGTANGEN, H. P. (Ed.). **Modern software tools in scientific computing**. 1. ed. [S.l.]: Birkhauser Press, 1997. cap. 8, p. 163–202. ISBN 978-1-4612-7368-4. Cited in page [46](#).
- BASSI, F.; REBAY, S. High-order accurate discontinuous finite element solution of the 2d euler equations. **Journal of Computational Physics**, v. 138, n. 2, p. 251 – 285, 1997. ISSN 0021-9991. Cited in page [28](#).
- BENDSØE, M. P.; SIGMUND, O. **Topology optimization: theory, methods, and applications**. [S.l.: s.n.], 2003. v. 2nd Edition. 370 p. ISSN 00189464. ISBN 3540429921. Cited in page [18](#).
- BOCHEV, P. B.; GUNZBURGER, M. D. Finite Element Methods of Least-Squares Type. **SIAM Review**, v. 40, n. 4, p. 789–837, 1998. ISSN 0036-1445. Cited in page [29](#).
- BOCHEV, P. B.; GUNZBURGER, M. D. **Least Squares Finite Element Methods**. New York: Springer, 2005. v. 166. 1049–1053 p. ISSN 1058-5893. ISBN 978-0-387-30888-3. Cited 2 times in pages [29](#) and [32](#).
- BOLTON, P.; THATCHER, R. W. On mass conservation in least-squares methods. **Journal of Computational Physics**, v. 203, n. 1, p. 287–304, 2005. ISSN 00219991. Cited in page [32](#).
- 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 2 times in pages [18](#) and [37](#).



- CAREY, G. F.; JIANG, B.-N. Least-squares finite elements for first-order hyperbolic systems. **International Journal for Numerical Methods in Engineering**, v. 26, n. 1, p. 81–93, jan 1988. ISSN 0029-5981. Cited in page [31](#).
- Castagno, J.; Atkins, E. Polylicdar - polygons from triangular meshes. **IEEE Robotics and Automation Letters**, v. 5, n. 3, p. 4634–4641, 2020. Cited in page [48](#).
- CODINA, R.; BLASCO, J. Analysis of a stabilized finite element approximation of the transient convection-diffusion-reaction equation using orthogonal subscales. **Computing and Visualization in Science**, v. 4, n. 3, p. 167–174, Feb 2002. ISSN 1433-0369. Cited in page [28](#).
- 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 in page [20](#).
- DOLLING, D. S. Fifty Years of Shock-Wave/Boundary-Layer Interaction Research: What Next? **AIAA Journal**, v. 39, n. 8, p. 1517–1531, aug 2001. ISSN 0001-1452. Cited in page [24](#).
- DONEA, J.; HUERTA, A. **Finite Element Methods for Flow Problems**. [S.l.]: John Wiley & Sons, 2003. 350 p. ISBN 0-471-49666-9. Cited in page [28](#).
- DUMAS, L. Cfd-based optimization for automotive aerodynamics. In: **Optimization and computational fluid dynamics**. [S.l.]: Springer, 2008. p. 191–215. Cited in page [15](#).
- EASON, E. D. A review of least-squares methods for solving partial differential equations. **International Journal for Numerical Methods in Engineering**, v. 10, n. 5, p. 1021–1046, 1976. ISSN 0029-5981. Cited in page [29](#).
- ERN, A.; GUERMOND, J.-L. **Theory and Practice of Finite Elements**. [S.l.: s.n.], 2004. v. 159. 524 p. ISSN 00255572. ISBN 978-1-4419-1918-2. Cited in page [26](#).
- FUNKE, S. W. **The automation of PDE-constrained optimisation and its applications**. 207 p. Thesis (PhD Thesis) — Imperial College London, 2012. Cited 3 times in pages [35](#), [36](#), and [48](#).
- GERRITSMA, M.; BAS, R. van der; De Maerschalk, B.; KOREN, B.; DECONINCK, H. Least-squares spectral element method applied to the Euler equations. **International Journal for Numerical Methods in Fluids**, v. 57, n. 9, p. 1371–1395, jul 2008. ISSN 02712091. Cited in page [32](#).
- GERSBORG-HANSEN, A.; SIGMUND, O.; HABER, R. B. Topology optimization of channel flow problems. **Structural and Multidisciplinary Optimization**, v. 30, n. 3, p. 181–192, 2005. ISSN 1615147X. Cited in page [19](#).
- GUNZBURGER, M. **Perspectives in Flow Control and Optimization**. [S.l.]: Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2003. ISBN 9780898718720. Cited in page [15](#).
- HASLINGER, J.; MAKINEN, R. A. E. **Introduction to Shape Optimization: Theory, Approximation, and Computation**. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2003. ISBN 0898715369. Cited in page [17](#).

- HIRSCH, C. **Numerical computation of internal and external flows, Volume 1, Fundamentals of Numerical Discretization**. [S.l.]: John Wiley & Sons, 1988. 515 p. ISBN 047192385. Cited in page [24](#).
- HIRSCH, C. **Numerical computation of internal and external flows, Volume 2, Computational Methods for Inviscid and Viscous flows**. Chichester: [s.n.], 1990. 691 p. ISBN 0471923516. Cited 2 times in pages [24](#) and [25](#).
- HØGHØJ, L. C.; NØRHAVE, D. R.; ALEXANDERSEN, J.; SIGMUND, O.; ANDREASEN, C. S. Topology optimization of two fluid heat exchangers. **International Journal of Heat and Mass Transfer**, Elsevier, v. 163, p. 120543, 2020. Cited in page [20](#).
- HUTCHISON, M.; UNGER, E.; MASON, W.; GROSSMAN, B.; HAFTKA, R. Variable-complexity aerodynamic optimization of a high-speed civil transport wing. **Journal of Aircraft**, v. 31, n. 1, p. 110–116, 1994. Cited in page [15](#).
- JIANG, B.-N. **The Least-Squares Finite Element Method**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998. 418 p. ISBN 978-3-642-08367-9. Cited in page [29](#).
- JIANG, B.-N.; CAREY, G. F. A stable least-squares finite element method for non-linear hyperbolic problems. **International Journal for Numerical Methods in Fluids**, v. 8, n. 8, p. 933–942, 1988. ISSN 10970363. Cited in page [30](#).
- JIANG, B.-N.; CAREY, G. F. Least-squares finite element methods for compressible Euler equations. **International Journal for Numerical Methods in Fluids**, v. 10, n. 5, p. 557–568, 1990. ISSN 0271-2091. Cited in page [31](#).
- KARNIADAKIS, G.; SHERWIN, S. **Spectral/hp Element Methods for Computational Fluid Dynamics**. Oxford: Oxford University Press, 2005. 657 p. ISSN 1098-6596. ISBN 9780198528692. Cited in page [27](#).
- LEFEBVRE, D.; PERAIRE, J.; MORGAN, K. Finite Element Least Squares Solution of the Euler Equations Using Linear and Quadratic Approximations. **International Journal of Computational Fluid Dynamics**, v. 1, n. 1, p. 1–23, 1993. ISSN 1061-8562. Cited in page [30](#).
- LIN, S.; ZHAO, L.; GUEST, J. K.; WEIHS, T. P.; LIU, Z. Topology optimization of fixed-geometry fluid diodes. **Journal of Mechanical Design**, v. 137, p. 081402, 2015. Cited in page [20](#).
- LOGG, A.; MARDAL, K. A.; WELLS, G. N. **Automated Solution of Differential Equations by the Finite Element Method**. [S.l.: s.n.], 2012. v. 84. 724 p. ISBN 978-3-642-23098-1. Cited 2 times in pages [44](#) and [46](#).
- LYNESS, J. N.; MOLER, C. B. Numerical differentiation of analytic functions. **SIAM Journal on Numerical Analysis**, Society for Industrial and Applied Mathematics, v. 4, n. 2, p. 202–210, 1967. ISSN 00361429. Cited in page [35](#).
- MUSKULUS, M.; SCHAFHIRT, S. Design optimization of wind turbine support structures—a review. **Journal of Ocean and Wind Energy**, v. 1, n. 1, p. 12–22, 2014. Cited in page [15](#).

- NEMEC, M.; ZINGG, D. W.; PULLIAM, T. H. Multipoint and multi-objective aerodynamic shape optimization. **AIAA journal**, v. 42, n. 6, p. 1057–1065, 2004. Cited in page 15.
- OKUBO, C. M.; SÁ, L. F.; KIYONO, C. Y.; SILVA, E. C. A discrete adjoint approach based on finite differences applied to topology optimization of flow problems. **Computer Methods in Applied Mechanics and Engineering**, Elsevier, v. 389, p. 114406, 2022. Cited in page 20.
- PAPOUTSIS-KIACHAGIAS, E. M.; GIANNAKOGLU, 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 in page 20.
- 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**, Springer, v. 65, n. 1, p. 1–25, 2022. Cited 3 times in pages 21, 39, and 41.
- PICELLI, R.; RANJBARZADEH, S.; SIVAPURAM, R.; GIORIA, R. d. S.; SILVA, E. C. N. Topology optimization of binary structures under design-dependent fluid-structure interaction loads. **Structural and Multidisciplinary Optimization**, Springer, v. 62, n. 4, p. 2101–2116, 2020. Cited in page 20.
- PINGEN, G.; EVGRAFOV, A.; MAUTE, K. Topology optimization of flow domains using the lattice boltzmann method. **Structural and Multidisciplinary Optimization**, Springer, v. 34, n. 6, p. 507–524, 2007. Cited in page 20.
- PINGEN, G.; MAUTE, K. Optimal design for non-newtonian flows using a topology optimization approach. **Computers & Mathematics with Applications**, Elsevier, v. 59, n. 7, p. 2340–2350, 2010. Cited in page 20.
- PONTAZA, J.; DIAO, X.; REDDY, J.; SURANA, K. Least-squares finite element models of two-dimensional compressible flows. **Finite elements in analysis and design**, Elsevier, v. 40, n. 5–6, p. 629–644, 2004. Cited 3 times in pages 24, 30, and 51.
- PRITCHARD, P. **Fox and McDonald’s Introduction to Fluid Mechanics**. 8th. ed. [S.l.]: John Wiley & Sons, 2011. 875 p. ISBN 9780470547557. Cited 2 times in pages 24 and 52.
- ROMERO, J.; SILVA, E. 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 in page 20.
- SÁ, L.; ROMERO, J.; 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 15.
- SA, L. F.; OKUBO, C. M.; SILVA, E. C. Topology optimization of subsonic compressible flows. **Structural and Multidisciplinary Optimization**, Springer, v. 64, n. 1, p. 1–22, 2021. Cited in page 20.

- SA, L. F.; YAMABE, P. V.; SOUZA, B. C.; SILVA, E. C. Topology optimization of turbulent rotating flows using spalart–allmaras model. **Computer Methods in Applied Mechanics and Engineering**, Elsevier, v. 373, p. 113551, 2021. Cited in page 21.
- SANAN, P.; SCHNEPP, S.; MAY, D. Pipelined, Flexible Krylov Subspace Methods. **SIAM Journal on Scientific Computing**, v. 38, n. 5, p. C441–C470, jan 2016. ISSN 1064-8275. Cited in page 46.
- SAVITZKY, A.; GOLAY, M. J. Smoothing and differentiation of data by simplified least squares procedures. **Analytical chemistry**, ACS Publications, v. 36, n. 8, p. 1627–1639, 1964. Cited in page 49.
- SIVAPURAM, R.; PICELLI, R. Topology optimization of binary structures using integer linear programming. **Finite Elements in Analysis and Design**, Elsevier, v. 139, p. 49–61, 2018. Cited in page 41.
- SIVAPURAM, R.; PICELLI, R. Topology design of binary structures subjected to design-dependent thermal expansion and fluid pressure loads. **Structural and Multidisciplinary Optimization**, Springer, v. 61, n. 5, p. 1877–1895, 2020. Cited in page 48.
- SOUZA, B.; YAMABE, P.; SÁ, L.; RANJBARZADEH, S.; PICELLI, R.; SILVA, E. Topology optimization of fluid flow by using integer linear programming. **Structural and Multidisciplinary Optimization**, Springer, v. 64, n. 3, p. 1221–1240, 2021. Cited 2 times in pages 21 and 41.
- TAGHADDOSI, F.; HABASHI, W.; GUEVREMONT, G.; AIT-ALI-YAHIA, D. An adaptive least-squares method for the compressible Euler equations. **International Journal for Numerical Methods in Fluids**, v. 31, n. 7, p. 1121–1139, dec 1999. ISSN 0271-2091. Cited 2 times in pages 30 and 51.
- WINSBERG, E. Computer simulations in science. In: ZALTA, E. N. (Ed.). **The Stanford Encyclopedia of Philosophy**. Summer 2018. Metaphysics Research Lab, Stanford University, 2018. Available from Internet: <<https://plato.stanford.edu/entries/simulations-science/>>. Visited on: 30/11/2018. Cited in page 15.
- YOON, G. H. Topology optimization for stationary fluid–structure interaction problems using a new monolithic formulation. **International Journal for Numerical Methods in Engineering**, v. 82, n. 5, p. 591–616, 2010. Available from Internet: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2777>>. Cited in page 20.
- YOON, G. H. Topology optimization for turbulent flow with Spalart–Allmaras model. **Computer Methods in Applied Mechanics and Engineering**, Elsevier B.V., v. 303, p. 288–311, may 2016. ISSN 00457825. Cited in page 20.