UNIVERSITAT POLITÈCNICA DE CATALUNYA

MASTER'S THESIS - REPORT

Study of lightweighting structural design considering 3D printing constraints

Author:
Jose Antonio Torres Lerma

Director:
Dr. Àlex Ferrer Ferré

A project developed as a compendium and fulfillment of the Master's degree in Aerospace Engineering requirements - MUEA

within

Application of finite element methods for optimization problems
Physics Department

Escola Superior d'Enginyeries Industrial, Aeroespacial i Audiovisual de Terrassa - ESEIAAT

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola Superior d'Enginyeries Industrial, Aeroespacial i Audiovisual de Terrassa

June 22, 2022

"A wise man will make more opportunities than he finds."

Francis Bacon

Declaration of Honor

I declare that,

- The work and effort dedicated to this Master's Thesis is completely my work.
- All references from other people's work have been clearly cited.

Finally, I understand that a violation of this statement leaves me subject to disciplinary action, provided by the *Universitat Politècnica de Catalunya - BarcelonaTECH*.

Jose Antonio Torres Lerma

Abstract

One of the current challenges of the aerospace industry is the exploration of new lightweighting structures to reduce fuel consumption and limiting the environmental impact. The use of numerical methods concerning topology optimization techniques allows the obtaining of such weight reduction, also minimizing both design time and costs, and hence accelerating the design process. Nevertheless, current structural optimization leads to the apparition of complex shapes and volumes with unintuitive holes, thus needing the use of additive manufacturing constraints - minimum length scales and overhanging - to ensure manufacturability.

Considering the background exposed above, the aim of this project is to study the feasibility of heuristic designs concerning lightweighting structures, materialized with additive manufacturing and considering 3D printing constraints. The design stage will be developed by means of topology optimization techniques, applied to anisotropic filtering.

The methodology employed has considered all details concerning Computational Solid Mechanics (CSM) techniques used in structures optimization, as well as additive manufacturing techniques, different case studies definition and their feasibility study. More specifically, in the context of CSM, the use of Finite Element Methods (FEM) in the classical elastic problem is reviewed, as well as current topology optimization techniques, so as to implement FEM in optimization algorithms. Thus, theoretical basis in additive manufacturing techniques are reviewed, along with the mathematical formulation of length scale and overhang constraints. Lastly, the programming stage is performed by previously defining the working environment, consisting in the use of Object-Oriented Programming within the git Version Control System, and hence establishing the computational domain definition for all cases, the meshing process and the simulation setup.

In the end, the present project has accomplished the main objectives, giving a positive answer to the creation of lightweighting structures and fulfillment of 3D printing constraints. Indeed, FEM combined with topology optimization techniques has led to the obtaining of optimized designs, fulfilling an objective function and a set of constraints, considering both design variables approaches, density and level set. Besides, an additional shape functional has been defined as a penalty contribution to the main cost function in order to fulfill 3D printing constraints - the anisotropic perimeter - being the evolution of the standard isotropic one, both applied to total and relative perimeters. This shape functional self-penalizes length scale constraints and keeps control in overhanging phenomena by orienting the topologies with the definition of a virtual anisotropic stiffness matrix. Results obtained show that the apparition of local features with small length scales has been avoided when including either isotropic or anisotropic perimeter as a penalty term. Furthermore, vertical tendency orientation of topologies has been generally obtained with the anisotropic cases, along with penalization of horizontal features.

Overall, this project has become clearly relevant for the exploration of new lightweighting structures, achieving weight reduction with topology optimization techniques. Further exploration remains in the course of PhD professionalization, specially when considering phase-field models, high-performance computing and large-scale optimization inside the non-linear regime.

Resumen

Entre los retos actuales de la industria aeroespacial se halla la exploración de nuevas estructuras ultraligeras con la finalidad de reducir el consumo de combustible y limitar el impacto ambiental. El uso de métodos numéricos mediante técnicas de optimización topológica permite obtener dicha reducción de peso, minimizando el tiempo de desarrolo del diseño y sus costes. Sin embargo, los últimos avances en optimización estructural llevan a la aparición de formas complejas y volúmenes con agujeros poco intuitivos. Por lo tanto se requieren limitaciones por fabricación aditiva (mínimas longitudes de escala y *overhaning*) para asegurar fabricación.

Con los antecedentes previamente expuestos, el propósito del presente proyecto es el estudio de la fiabilidad de diseños heurísiticos para estructuras ultraligeras, materializados mediante fabricación aditiva y considerando restricciones por impresión 3D. La fase de diseño se desarrollará mediante técnicas de optimización topológica, aplicados a filtración anisotrópica.

La metodología empleada considera todos los detalles del uso de las técnicas de *Computational Solid Mechanics* (CSM) para optimización estructural, así como técnicas de fabricación aditiva, la definición de diferentes casos de estudio y su análisis de fiabilidad. Concretamente, en el contexto de CSM se revisa el uso de los Métodos de Elementos Finitos (FEM) en el problema elástico, junto a las técnicas de optimización topológica actuales, para lograr la implementación de FEM en algoritmos de optimización. Posteriormente se revisa la base teórica de técnicas de fabricación aditiva, junto a la formulación matemática de las restricciones por mínima escala y *overhanging*. Finalmente, se procede a la fase de programación, donde se define el dominio computacional para cada caso de estudio, el proceso de mallado y la configuración de la simulación.

Finalmente, el presente proyecto ha cumplido con los objetivos principales, dando respuesta positiva a la creación de estructuras ultraligeras y cumplimiento de restricciones por impresión 3D. En efecto, la unión de FEM con las técnicas de optimización topológica ha llevado a la obtención de diseños optimizados, verificando una función objetivo y un conjunto de restricciones, y considerando dos variables de diseño: *density* y *level set*. Además, se ha definido una función adicional que contribuye como penalización a la función objetiva principal - el perímetro anisotrópico -, con la finalidad de cumplir las restricciones por impresión 3D, siendo una evolución del perímetro estándar y ambos aplicados para perímetros totales y relativos. Dicho funcional auto penaliza longitudes de escala y mantiene control en fenómenos de *overhanging* orientando las toplogías con la definición de una matriz de rigidez anisotrópica virtual. Los resultados obtenidos muestran cómo se evade la aparición de topologías de baja escala, empleando tanto el perímetro isotrópico como el anisotrópico. Además, las topologías adquieren una tendencia vertical en su orientación para los casos anisotrópicos, junto a la penalización de topologías horizontales.

En conjunto, el proyecto ha resultado ser relevante para la exploración de nuevas estructuras ultraligeras, obteniendo reducción de peso mediante técnicas de optimización topológica. Se requiere más investigación en el curso de una tesis doctoral, especialmente considerando modelos de *phase-field*, *high-performance computing* y optimizaciones de larga escala dentro del régimen no lineal.

Acknowledgments

I would like to express my special thanks of gratitude to my director Àlex Ferrer, who gave me the golden opportunity to perform this project on the topic *Study of lightweighting structural design considering 3D printing constraints*, where a lot of Research has been performed as the basis of my ulterior doctoral research degree.

I would also like to thank my parents and friends for their moral support throughout the finalization of this project.

Contents

List of Figures

List of Tables

Nomenclature

$(\vec{u}, \vec{v}, \vec{w})$	Displacement field
α	Perimeter penalty coefficient
α_h	Left orthotropic direction contribution
$\bar{t}^i$	Traction vectors
β	Heavside steepness factor
β_h	Right orthotropic direction contribution
χ	Solid/void vector
δ	Stretching displacement
ϵ	Perimeter target parameter
ϵ_{kl}	Strain tensor
η	Projection threshold
$\frac{d^2v}{dx^2}$	Curvature
Γ^i_σ	Neumann boundary
Γ^i_u	Dirichlet boundary
λ, μ	Lame parameters
λ	Lagrange multipliers
μ, κ	Hashin-Shtrikman bound functions
ν	Poisson ratio
Ω	Reference domain
$\psi(x)$	Level set
$\rho(x)$	Density
$\rho_\epsilon(x)$	Regularized density
ρ_P	Projected density
σ_{ij}	Stress tensor
θ	Shape or topological derivative change magnitude
$\underline{\underline{B}}$	Strain-displacement matrix

$\underline{\underline{C}}$	Constitutive or elasticity matrix
$\underline{\underline{K_{el}}}$	Element stiffness matrix
$\underline{\underline{K}}$	Global stiffness matrix
$\underline{\underline{N}}$	Matrix of shape functions
$\underline{c}$	Virtual displacements vector
$\underline{d}$	Global vector of displacements
$\underline{F}$	Global external force vector
$\underline{R}$	Reaction forces
$a(u,v)$	Virtual energy bilinear form
A	Prismatic bar cross section area
A_{2D}	Global conductivity matrix
c	Compliance
$D_S J$	Shape derivative
$D_T J$	Topological derivative
d_{min}	Minimum length scale
E	Young Modulus
$F(x)$	Filter function
f	Subset of free nodes
f_i	External volume forces
F_{ij}	Internal force at node i due to displacement at node j
f_{VM}	Von Mises stress function
g^{solid}, g^{void}	Minimum length constraints
g_i	Problem constraints
G_{KK}	Park and Kikuchi global stress function
G_{KS}	Kreisselmeier-Steinhauser global stress function
H	Heavside function
h	Orthotropic effective property
I	Centroidal moment of inertia
I^{solid}	Minimum length requirement structural function for solid phases
I^{vector}	Minimum length requirement structural function for void phases
J	Objective or cost function

$J^e(\xi_g)$	Jacobian matrix
k_a	Scale factor
$l(u)$	Load linear form
$L(x, \lambda)$	Lagrangian
L	Prismatic bar length
L_τ^{ell}	PDE filter
M	Global mass matrix
N_i	Interpolating shape function of DoF i
n_j	Normal vector to surface at node j
P	Perimeter
p	SIMP exponent
P_0h	Kernel Operator P0
P_1h	Kernel Operator P1
R	Rotation matrix
r	Subset of restricted nodes
R_f	Filter radius
t	Steepest descent parameter
u_i	Trial functions
$v(x)$	Plane beam element lateral displacement
V	Volume
v_i	Test functions
W	Double well potential
w_g	g-Gauss point weight

Chapter 1

Objectives and scope of the project

In this project all the knowledge acquired throughout the Master's Degree in Aerospace Engineering is gathered in order to complement it in the study of lightweighting structural design considering 3D printing constraints, also including the programming stage and its feasibility study. Therefore, the level of detail in which the project will be defined shall be initially presented in this section.

On one hand, the aim and scope definition will give an overall perspective regarding all work packages that will be developed in the project, along with the corresponding level of detail studied. On the other hand, the project justification will define the reason to carry out all the work by taking into account where the innovative factor is headed, ending with a brief definition of basic requirements that will define the most important specifications to fulfill during the development of the project.

1.1 Aim

The aim of this project is to study the feasibility of heuristic designs concerning lightweighting structures, materialized with additive manufacturing and considering 3D printing constraints. The design stage will be developed by means of a finite element method based code, specifically applied to topology optimization techniques. The existing base algorithm will be initially reviewed, so that new code will be added in the field of topology optimization applied to anisotropic filtering.

1.2 Scope

With the purpose of defining properly the scope all main elements in which the project is divided shall be identified. From this design and feasibility study, the following aspects are found:

- The details concerning which Computational Solid Mechanics (CSM) techniques will be employed throughout structures optimization.
- Additive manufacturing techniques.
- Case studies definition.
- Programming stage.

- Feasibility study.
- Time and costs management.

1.2.1 Computational Solid Mechanics

The techniques employed in the context of CSM will gather the following topics:

- Use of Finite Element Method (FEM) algorithms in the field of the classical elastic problem.
- Theoretical basis review of Topology Optimization techniques: density and level set, gradient formulation, filtering, optimizers, methods, shape functionals and interpolation schemes.
- Isotropic and anisotropic perimeter constraints within topology optimization simulations.
- Finite Element implementation in Topology Optimization techniques: unfitted mesh, incremental scheme and the Diffusion-Reaction equation.
- Optimization algorithms.

1.2.2 Additive manufacturing

Since the project is dealing with 3D printing constraints, a review of the state of art concerning additive manufacturing must be carried out in order to include its basis inside the code's mathematical formulation, specifically in the aspects depicted below.

- Theoretical basis review in additive manufacturing techniques.
- Length scale constraints.
- Overhang constraints.

1.2.3 Case studies definition

Although topology optimization can be employed for any generic structure, subjected to a set of forces and displacements constraints, some case studies are defined such that the feasibility of the implemented code can be assessed through weight minimization criteria and fulfillment of additive manufacturing constraints, from all possible candidates. The case studies considered are the following:

- Cantilever beam in 2D (benchmark case).
- Bridge in 2D.
- Arch in 2D.
- Microstructure case in 2D.

1.2.4 Programming stage

The programming stage will represent the most durable part of the project, since it defines the code that will be developed from scratch regarding structural optimization within anisotropic filtering. The following aspects will be assessed, which in part includes some theoretical background (programming environment) and thus the programming stage itself.

- Programming environment. Definition of Object-Oriented Programming, Unified Modeling Language utility, test-driven development philosophy, code language and git environment. A detailed description of the whole environment is exposed in Appendix A.
- Computational domain definition for all cases.
- Meshing process.
- Simulation setup and code programming.

1.2.5 Feasibility study

Once the code is finished, with the corresponding results of all simulations developed, the following points are assessed in order to decide whether the implemented code would have been able to virtually produce a lightweighting structure fulfilling 3D printing constraints.

- Analysis of results:
  - Code verification with pure perimeter.
  - Objective functions minimization.
  - Fulfillment of additive manufacturing constraints.
- Numerical study: a comparison between different numerical methodologies.

1.2.6 Time and costs management

Finally, inside the field of the project management part, a detailed budget of all the work developed will be prepared, taking into consideration both human and additional (related with software license costs) resources. Besides, the project impact will be assessed.

1.3 Justification

The exploration of new lightweighting structures is the current challenge in the aerospace industry, mainly due to the minimization of the aircraft's total weight in order to reduce fuel consumption and, therefore, achieving a limitation of the environmental impact, when it comes to green aviation. Nowadays, the main goal consists in reducing fuel emissions up to 50% by 2050, which is directly related with weight reduction on the structure, leading to an increase of the energy efficiency, as long as it did not imply the initial mechanical behaviour and performance of the structure detriment. Thus, lightweighting designs are currently achieved by means of structures with low-density materials or structures with less material, hence justifying the importance of topology optimization techniques. [1]

Moreover, the use of numerical methods concerning topology optimization techniques also allows the minimization of both design time and costs. In recent years, it has been an increase of industrial companies from a wide range of engineering areas that employs numerical methods to accelerate the preliminary design stage of a determined project. In fact, the use of commercial software inside the field of Computational Fluid Dynamics and Computational Solid Mechanics is useful to take decisions quickly regarding the evolution of the design rather than study such decisions experimentally, prior to the final prototype manufacturing. The addition of topology optimization techniques in these powerful numerical methods just upgrade this feature completely by allowing companies to obtain a final product that, apart from presenting the capabilities initially defined, these are obtained with a complete minimization of the material in the structure, hence implying a noticeable reduction of the manufacturing cost.

Nonetheless, current structure optimization leads to the apparition of complex shapes and volumes with unintuitive holes. Here is when it comes the use of additive manufacturing techniques to test these new designs, since functional components are currently fabricated with these techniques in a quick way and efficiently, from a CAD model that is sliced into a set of layers. During the manufacturing process both the geometry creation and part functionality will be crucial, mainly due to the lack of flexibility during overhanging regions extrusion, thus introducing the use of 3D printing constraints for anisotropic materials. This shall complement the optimization algorithm to propose compatible solutions for 3D printing processes. Additive manufacturing also would allow the customer to cut down the manufacturing time cost by process planning reduction. [2]

Topology optimization can also be complemented with the micro-structured materials philosophy. Microstructures present periodical material disposition such that high performance is achieved with minimum mass. Nevertheless, such properties are mainly defined through material cells topologies, and therefore current research is guided towards finding an effective method for microstructures optimization, using optimization methods such as homogenization, solid isotropic material with penalization (SIMP), evolutionary structural optimization or the level set method. [3]

At this point, apart from the topology optimization techniques justification itself, the finite element method base algorithms employed within these techniques also have a clear motivation in current research developments. Indeed, the Finite Element Method is a powerful numerical tool to solve a mathematical formulation of a physical problem, subjected to a set of hypotheses, now applicable to the virtual creation

of heuristic structural designs, depending on manufacturing conditions and constraints. However, FEM algorithms is commonly found in physical problems where the unknowns and external excitations are linked through a coefficient matrix that only depends on independent properties as it occurs in structural cases, but unlike it happens it fluid flow problems, where the coefficient matrix of the fluid shall depend on the own unknowns of the problem, relatively hard to implement and hence showing the reason of using Finite Volume Methods instead, with a lower level of complexity.

Finally, the last motivation of this project consists in the use of the Object-Oriented Programming technique to implement all the required code, since it represents an excellent way to get familiar with collaborative projects where programming tools are used, such as the Git Environment. With all exposed, gathering these programming techniques and topology optimization methods, the work exposed in this project would allow the further exploration of future lines of research in the course of PhD professionalization.

1.4 Requirements

The main key requirements that the programmed code must satisfy are the following:

1. Able to obtain an optimized lightweighting structure, given a reference one, minimizing the corresponding cost function.
2. Able to satisfy all constraints imposed, such as additive manufacturing constraints.
3. Able to adapt to any type of initial structural shape.
4. Able to be used to 2D problems.
5. Able to compute the corresponding result with minimum computational cost.
6. Able to be combined with further improvements of the generic code repository.

Chapter 2

Lightweighting structures design with topology optimization techniques

In this chapter all the numerical basis required to design lightweighting structures considering additive manufacturing constraints is presented. Indeed, the possibility to obtain a structure with less volume and within an optimized shape, given a reference one, and by means of a numerical simulation is the main challenge of this project. Nevertheless, it is also required that the final design ensures manufacturability. Therefore, both minimum length scales and overhang constraints will be included inside the numerical optimization algorithm.

More specifically, the use of Finite Element Methods (FEM) in linear static elastic problems is reviewed, so as to couple this technique to the field of topology optimization. Hence, the mathematical formulation of these techniques is exposed, along with typical design variables employed and different shape functionals serving as an objective function or a constraint of the optimization problem. Later, the numerical methodology, as well as interpolation schemes, regularization of the design variable and the definition of shape and topological derivatives will be presented. Finally, all types of solvers available inside the field of topology optimization are exposed.

2.1 Introduction to topology optimization

Prior to define all the theoretical basis regarding topology optimization techniques, in Appendix B the revision of the physical and mathematical formulation of the Finite Element Method is provided.

2.1.1 Mathematical formulation

Topology optimization is a numerical approach based in Finite Element Method algorithms with the purpose to design new structures on any scale. Such techniques consider a reference or initial non-optimized structure from which new designs will be created, taking into consideration a set of constraints and the objective function governing the optimization case. Although these new designs would still keep similarity with the non-optimized structure, the weight will be reduced by the creation

of voids near to the less stressed areas of the structure, allowing the material to still be able to withstand the same distribution of external forces. Therefore, structures with less mass and accomplishing with the same capabilities would be computationally created. Besides that, the assessment of new materials, which would be useful candidates for the new design, may also be designed via Topology Optimization techniques.

Topology optimization deals with an optimization problem, thus an objective function and a set of constraints are required to formulate the mathematical model. Nevertheless, as the system is always in equilibrium for any possible optimized candidate, the structural matrix equation will be fulfilled. The general mathematical formulation is shown

$$P_{TO} : \left\{ \begin{array}{c} \min\ J(\chi, u) \\ s.t\ K(\chi) \cdot u = F(\chi) \\ g_i(\chi, u) \leq 0 \end{array} \right\} \quad \begin{array}{c} J \rightarrow \text{Objective (or cost) function} \\ \chi \rightarrow \text{Solid/void vector} \\ u \rightarrow \text{Displacement vector} \\ K \rightarrow \text{Global stiffness matrix} \\ F \rightarrow \text{Force vector} \\ g_i \rightarrow \text{Problem constraint.} \end{array} \tag{2.1}$$

Although there is a large variety of objective functions to discuss (shape functionals), as well as constraints, the first problem usually defined inside the field of topology optimization is known as the minimum compliance problem (maximum global stiffness) under simple resource constraints [4].

Consider a mechanical continuum element defined inside a domain Ω^{mat}, which is a subset of a larger reference domain Ω in $\mathbb{R}^2$ or $\mathbb{R}^3$. The reference domain is defined to allow the original definition of applied loads and boundary conditions, and it will be enclosed by a boundary. Hence, referring to Ω, we can define the optimal design problem as finding the optimal choice of the characteristic function χ that parametrizes $C_{ijkl}(\chi)$, variable over the domain. The bilinear form and the linear form of the equilibrium equation is introduced,

$$a(u, v) = \int_{\Omega} C_{ijkl}(x)\epsilon_{ij}(u)\epsilon_{kl}(v) \cdot d\Omega \tag{2.2}$$

$$l(u) = \int_{\Omega} du \cdot d\Omega + \int_{\Gamma_T} tu \cdot dS. \tag{2.3}$$

Besides, we need a limit of resource. For instance, this can be expressed as $\int_{\Omega_{mat}} 1 d\Omega \leq V$, thus indicating at which extent the volume shall be reduced during the heuristic optimization. Therefore, considering the finite element discretization of the domain, the minimum compliance problem (MC) takes the final form exposed in Equation (2.4).

$$P_{MC} : \left\{ \begin{array}{c} \min_x\ f^T \cdot u \\ s.t\ K(x) \cdot u = F(x) \\ x \in [0, 1] \\ \int_{\Omega_{mat}} 1 d\Omega \leq V \end{array} \right\} \tag{2.4}$$

2.1.2 Density vs Level Set approach

In the design of the optimized topology from a given structure, the placement of material in space must be determined, such that the classification of nodes into material points or void points could be carried out (black-white rendering of an image). Hence, considering the reference domain Ω, the optimal subset Ω^{mat} of material points must be determined. In mathematical form, the set of admissible elasticity matrices consists of those tensors satisfying

$$C_{ijkl} = I_{\Omega_{mat}} C^0_{ijkl}, \qquad I_{\Omega_{mat}} = \begin{Bmatrix} 1 & \text{if} & x \in \Omega^{mat} \\ 0 & \text{if} & x \notin \Omega^{mat} \end{Bmatrix}, \tag{2.5}$$

where C^0_{ijkl} is the elasticity matrix for the given material distribution, thus defining C_{ijkl} [4]. Nevertheless, a distributed, discrete valued design problem has been formulated, and this presents some incompatibilities when considering differential calculus operators near the boundary between black and white values. Therefore, the characteristic function χ is usually replaced by its continuous counterpart ρ, that may be understood in turn as a density of the domain.

Density approach. One popular possibility and efficient to penalize intermediate values of density is the so-called Solid Isotropic Material with Penalization (SIMP-model), which is defined in Equation (2.6). The density interpolates between the material properties 0 and the reference elasticity matrix as

$$\begin{matrix} C_{ijkl} = \rho(x)^p C^0_{ijkl}, \quad p > 1 \\ \int_\Omega \rho(x) \cdot d\Omega \leq V \end{matrix}. \tag{2.6}$$

In SIMP, a value of $p > 1$ is chosen so that intermediate densities (*grey* colors) are unfavourable in the optimal design. When the volume constraint is activated throughout the problem resolution, SIMP proposes a value of $p \geq 3$ concerning the auto-penalization phenomenon. Moreover, it exists a large variety of methods apart from SIMP, applied to density as design variable and considering both constrained and unconstrained problems.

Level set approach. Nevertheless, there are a second group of methodologies related to the concepts of shape and topological derivative, which are used to monitor the so-called level set design variable. For instance, and as it will be discussed later, the Hamilton-Jacobi and SLERP are some of the methods applied to level set [5].

The level set $\psi(x)$ is a continuous function that defines the classification of a nodal material into Ω^{mat} or void depending on its sign, thus avoiding the apparition of *grey* values which are present in density as design variable. The level set defines the black/white characteristic function χ, as depicted in Equation (2.7).

$$\chi = 1 - H(\psi) = \begin{Bmatrix} 1 & \text{if} & \psi \leq 0 \\ 0 & \text{if} & \psi > 0 \end{Bmatrix} \tag{2.7}$$

where $H(\psi)$ represents the Heaviside function. The minimum compliance problem within level set as design variable is exposed in Equation (2.8).

$$P_{MC} : \left\{ \begin{array}{c} \min_{\chi,u} \ f^T \cdot u \\ s.t \ K(\chi) \cdot u = F(\chi) \\ \chi \in \{0;1\} \end{array} \right\}. \tag{2.8}$$

Finally, a reduced formulation can also be proposed if a relationship between $u = f(\chi)$ might be established within the equilibrium equation depending on the resolution scheme, thus defining this formulation in Equation (2.9).

$$P_{MC} : \left\{ \begin{array}{c} \min_{\chi} \ f^T \cdot u(\chi) \\ s.t \ K(\chi) \cdot u(\chi) = F(\chi) \\ \chi \in \{0;1\} \end{array} \right\} \tag{2.9}$$

2.1.3 Shape functionals

Although the minimum compliance problem is a common topology optimization case applied to structures, there are a large variety of different objective functions and constraints that might be considered for different optimization processes. Among them, below the most common ones to consider are provided. [6]

Volume

This functional is generally used as constraint within the minimum compliance problem in order to reduce the new design weight and cost. In Equation (2.10) the definition of volume constraint is depicted, where Ω is the whole reference domain, $\rho(x)$ the distribution of continuous material density between 0 and 1 on the reference domain and V the desired geometric volume.

$$V \geq \int_{\Omega} \rho(x) \cdot d\Omega \tag{2.10}$$

Compliance

As it has been presented at the beginning of current section, the minimum compliance problem is typical among researchers due to its compatibility with current algorithms and accuracy of results. Given the external nodal force vector and the displacement vector of the finite element system, the compliance is obtained as

$$c = f^T \cdot u \tag{2.11}$$

where u is the solution of the equilibrium equation.

Perimeter

The perimeter is usually employed as an additional contribution to the compliance objective function, this is

$$J = c + \alpha \cdot P \tag{2.12}$$

where $\alpha \in [0, 1]$ is some parameter that defines the degree of perimeter contribution into the optimization problem (typically, $\alpha \simeq 0.1$). By including some dependence on the perimeter in the minimum compliance problem, the formation of small geometrical scales in beams is avoided, thus guaranteeing manufacturability. The perimeter is defined as

$$P = \frac{1}{2\epsilon} \int_{\Omega} (1 - \rho_{\epsilon})\rho \cdot d\Omega \tag{2.13}$$

where ϵ stands for the minimum mesh size. The perimeter definition also penalizes *gray* regions defined by the density design variable ρ and its regularized (filtered) version ρ_{ϵ}, since near these values the perimeter function is maximum.

Orthotropic effective properties

When dealing with microstructures, such as a set of fibers embedded by a matrix, the homogenized two-dimensional elasticity tensor $\mathbb{C}$ for orthotropic materials in matrix form must be considered, as [7]

$$\mathbb{C} = \begin{bmatrix} \mathbb{C}_{1111} & \mathbb{C}_{1122} & \mathbb{C}_{1112} \\ \mathbb{C}_{1122} & \mathbb{C}_{2222} & \mathbb{C}_{2212} \\ \mathbb{C}_{1112} & \mathbb{C}_{2212} & \mathbb{C}_{1212} \end{bmatrix}. \tag{2.14}$$

For the case of orthotropic symmetry, the effective properties - Young's, bulk and shear moduli, along with the Poisson's ratio - are related explicitly to the components of the compliance tensor expressed in matrix form $\mathbb{C}^{-1}$, as

$$\mathbb{C} = \begin{bmatrix} \mathbb{C}^{-1}_{1111} & \mathbb{C}^{-1}_{1122} & 0 \\ \mathbb{C}^{-1}_{1122} & \mathbb{C}^{-1}_{2222} & 0 \\ 0 & 0 & \mathbb{C}^{-1}_{1212} \end{bmatrix} = \begin{bmatrix} \frac{1}{E_1} & -\frac{\nu_{12}}{E_1} & 0 \\ -\frac{\nu_{21}}{E_2} & \frac{1}{E_2} & 0 \\ 0 & 0 & \frac{1}{G} \end{bmatrix} \tag{2.15}$$

where E_1, E_2 are the effective Young's moduli along orthotropic main directions and G is the effective in-plane shear modulus. Since the elasticity matrix is a symmetric matrix, the compliance matrix is also a symmetric one, thus ν_{12} and ν_{21} Poisson's ratio satisfies

$$\frac{\nu_{21}}{E_2} = \frac{\nu_{12}}{E_1}. \tag{2.16}$$

Therefore, some shape functional defined as $h(\mathbb{C})$ - participating in the cost function as $J(\Omega^1) = h + \lambda \frac{|\Omega^1|}{V}$ (where λ is a fixed Lagrange multiplier which imposes the volume ratio constraint) - shall be defined in order to obtain microstructures with optimized effective properties. This is

$$h(\mathbb{C}) = \alpha_h^T \mathbb{C}^{-1} \beta_h \tag{2.17}$$

where α_h and β_h are two vectors which define the contribution of effective properties to be minimized.

Stress norm

The stress is usually obtained from the displacement vector computed in the finite element system, and it is usually quantified with Von Mises criteria. Stress based optimization cases imply the treatment of local quantity and also highly nonlinear values regarding the design variables. One solution to assess these difficulties consists in transforming the stresses to a global measure [8]. Two global stress functions are found in the literature - the Kreisselmeier-Steinhauser (KS) and the Park and Kikuchi (KK) functions -, which are defined in Equations (2.18) and (2.19), respectively.

$$G_{KS} = \frac{1}{q} \ln \sum_{i=1}^{N} e^{q \frac{f_{VM,i}(\sigma)}{f_{VM,max}(\sigma)}} \tag{2.18}$$

$$G_{KK} = \left[\int_{\Omega} \left(\frac{f_{VM,i}(\sigma)}{f_{VM,max}(\sigma)} \right)^{q} \cdot d\Omega \right]^{\frac{1}{q}} \tag{2.19}$$

where f_{VM} is the Von Mises stress function and q a parameter that defines the difference between the original function and the global approximation.

Eigenvalues

The consideration of inherent free vibrations within the optimal structure is also a crucial topic to assess, for instance when considering aeroelastic phenomena. Now, it is wanted to maximize the objective function, defined as the first or fundamental eigenvalue of the structure, $\max(\lambda_{min})$. The eigenvalue problem typically reads to the expression as find (λ, u) such that [9]

$$(K(x) - \lambda M(x)) \cdot u = 0 \tag{2.20}$$

2.1.4 Filtering

The filtering problem consists in the replacement of the elastic properties dependence on the density based characteristic function ρ by the dependence of a filtered and smooth version, defined as $(F \circledast \rho)$, and related with the regularized density $\hat{\rho}$. Therefore, by defining a filter operator, the unfiltered field of the density design value would lead to a purely *black and white* design [10].

The filter operator is applied at any point of the domain and it is related with a characteristic radius of filter range, $R_f > 0$. The filter function F is defined such that:

- The filter acts inside a domain of infinite set of functions which first derivative is continuous in $\mathbb{R}^2$.

- $Supp(F) \subset B_R$. I.e, all space coordinates related to function F such that $F \neq 0$ are a subset of the open ball of centre 0 and radius R_f.
- $\int_{B_R} F \cdot d\Omega = 1$

The filtering operation is defined as the convolution product of F with the characteristic function, as seen in Equation (2.21).

$$(F \circledast \rho)(x) = \int_{\mathbb{R}^2} F(x-y) \cdot \rho(y) \cdot dy \tag{2.21}$$

Thus, the regularized density reads to a smooth and differentiable variable. Lastly, the filter function $F(x)$ can be defined with several options, employing mainly those previously studied by Amstutz, Dapogny and Ferrer. [11]

The Kernel operator P0

The first proposed operator P_0 is the orthogonal projection from the original characteristic function space $L^2(D)$ into the subspace of regularized density V_0, as seen in

$$P_0 h = \sum_k \frac{1}{|T_k|} (N_k^0, h)_{L^2(D)} N_k^0 \tag{2.22}$$

where T_k is each of the triangles conforming a triangular mesh and N_k^0 is each of the functions that $N_k^0 \equiv 1$ on T_k and $N_k \equiv 0$ on $T_{k'}$, $k \neq k'$. This filter is understood as a self-adjoint.

The Kernel operator P1

Another operator $P_1 : L^2(D) \to V_0$ is defined by Equation (2.23).

$$P_1 h = \sum_{k=1}^{K} (Q_k, h)_{L^2(D)} N_k^0 \tag{2.23}$$

where Q_k is defined as

$$Q_k = \frac{1}{\sum_{i=1}^{3} (N_j^1, 1)_{L^2(D)}} \sum_{i=1}^{3} N_j^1. \tag{2.24}$$

This filter is not self-adjoint from the original domain into itself, but allows to take into consideration the neighborhood of any finite element within the structural domain.

The PDE Filter

Finally, the PDE operator $L_\tau^{ell} : L^2(D) \to V_0$ is defined in Equation (2.25).

$$L_{\tau}^{ell} = P_0 q_{\tau,h} \tag{2.25}$$

where $q_{tau,h}$ is precisely defined as

$$q_{\tau,h} = \sum_{i=1}^{J} \left(\sum_{j=1}^{J} R_{ij} (N_j^1, h)_{L^2(D)} \right) N_i^1, \text{ where } R = (\epsilon^2 K + M)^{-1}, \tag{2.26}$$

solution of a structural Partial Differential Equations set, where K and M are the stiffness and mass matrices, respectively. This filter is not self-adjoint operator from the original domain into itself, and its smoothing effect is controlled at the same time by the mesh size. Besides, it is expected to obtain large grayscale areas by using the PDE filter.

2.1.5 Methods

Prior to define the algorithm of all different optimizers and solvers that will represent the *backbone* for the topology optimization techniques, below some general definitions to be considered are exposed.

Shape and topological derivatives definition

The solid shape embedded in a fixed computational domain, and defined within a black-and-white design, is optimized either by considering a steepest descent heuristic or based on the notions of shape and topological derivatives. This will represent also a tool complementary with the filtering process, useful when a finer mesh is employed and thus smaller scales shall appear during the optimization process and these must be avoided in order to ensure manufacturability. Under these assumptions, the relaxed and filtered version of an optimal design problem is consistent with the original black-and-white shape and topology optimization problem. Specifically, if the gradient of regularized cost function of the topology optimization problem is computed in the whole computational domain, hence it shall converge to the known shape derivative of J (when restricted to the boundary) or to its topological derivative (when restricted to inside the domain). [11]

Shape derivative. The shape derivative of the cost function with respect to the characteristic function $D_S J(\chi)$ gives the information of how the cost function increases by moving the internal boundaries in such a way that holes become greater. Let θ indicate the magnitude of change of the boundary. Therefore, a *helpful* Taylor expansion expression gives us an idea of how the shape derivative works,

$$J_1(\chi) = J_0(\chi) + \theta \cdot D_S J(\chi) \tag{2.27}$$

where J_1 and J_0 represents the cost function after and before the boundary variation, respectively. This expression reads to a linear approximation of the variation of the cost function for small changes in the boundary. Moreover, the precise definition of the shape derivative is shown in Equation (2.28), proposed by Amstutz, Dapogny and Ferrer as [11]

$$\begin{gathered} D_S J = \int_\Gamma g_\Omega^S \theta \cdot n \cdot ds \\ \text{where } g_\Omega^S = \gamma_\Omega \nabla_\Gamma u_\Omega \cdot \nabla_\Gamma u_\Omega - \left(\frac{1}{\gamma_\Omega}\right)\left(\gamma_\Omega \frac{\partial u_\Omega}{\partial n}\right)^2 + l. \end{gathered} \tag{2.28}$$

Topological derivative. The topological derivative of the cost function with respect to the characteristic function $D_T J(\chi)$ gives the information of how the cost function increases by inserting a new infinitesimal hole in some black region, without moving the boundaries (i.e changing the topology of the design). If one lets θ to indicate the magnitude of the hole insertion, the same Taylor expansion than in Equation (2.27) can be used to define analogously how the cost function changes in function of θ and $D_T J$.

More precisely, considering some function $f(\epsilon)$ such that $f(\epsilon) \to 0$ (for instance, $f(\epsilon) = \pi\epsilon^2$), one defines preliminarily the topological derivative as shown in Equation (2.29) [12].

$$D_T J = \lim_{\epsilon \to 0^+} \frac{J_1(\chi) - J_0(\chi)}{f(\epsilon)} \tag{2.29}$$

For a given Young's modulus and Poisson's ratio of matrix $\Omega_\epsilon \backslash \bar{B}_\epsilon$ and the inclusion B_ϵ, represented as E_m, ν_m and E_i, ν_i, respectively, hence a fourth-order polarization tensor is defined as

$$\mathbb{P} = p_1 \mathbb{I} + p_2 I \otimes I. \tag{2.30}$$

Therefore, the topological derivative at some point is defined as $D_T J = \sigma : \mathbb{P} : \nabla^s u$. Again, Amstutz, Dapogny and Ferrer proposed the precise definition of the topological derivative,

$$D_T J = -(\gamma_1 - \gamma_0) k(\chi) \nabla u_\Omega(x) \cdot \nabla u_\Omega(x) + l \quad \forall \quad x \in \Omega(\chi). \tag{2.31}$$

Finally, both derivatives have some properties at the optimal point of the design, useful for developers to propose strategies for solvers definition. These properties are shown in Equation (2.32).

$$\begin{Bmatrix} D_T J \leq 0 & \text{if} & \chi = 1 \\ D_S J = 0 & \text{if} & x \in \Gamma \\ D_T J \geq 0 & \text{if} & \chi = 0 \end{Bmatrix} \tag{2.32}$$

Density interpolation schemes

Regarding the use of density as design variable approach, it was mentioned previously that intermediate values (grey region) must be penalized, focusing on low density values rather than larger ones. These strategies are known as interpolation schemes and in the literature we found mainly two approaches: the SIMP and SIMP-ALL schemes. [6]

The so-called Solid Isotropic Material with Penalization (SIMP) method auto-penalizes the material properties as shown in Equation (2.33).

$$C_{ijkl} = \rho(x)^p C^0_{ijkl} \quad p > 1 \tag{2.33}$$

Typically, the most common heuristic sets $p \simeq 3$ to give accurate results. Nevertheless, it is also common to compute the penalization factor according the Hashin-Shtrikman (HS) bounds for two-phase materials accomplishment, as shown in

$$p(\nu) = \max\left\{\frac{2}{1-\nu}, \frac{4}{1+\nu}\right\}. \tag{2.34}$$

Moreover, SIMP-ALL material interpolation (proposed by Ferrer) is based on the interpolation between the upper bound (matrix) and lower bound (void inclusion) of Hashin-Shtrikman bounds, depicted as (μ^+, κ^+) and (μ^-, κ^-), respectively. This approach employs the topological derivative to compute a rational function (see Equation (2.35)) which interpolates μ and κ in function of the density and some coefficients that shall be determined assessing a set of constraints.

$$\begin{aligned} \kappa(\rho) &= \tfrac{a_2\rho^2 + a_1\rho + a_0}{b_1\rho + 1} \\ \mu(\rho) &= \tfrac{c_2\rho^2 + c_1\rho + c_0}{d_1\rho + 1} \end{aligned} \tag{2.35}$$

$$\begin{aligned} \kappa(1) &= \kappa^+ & \kappa(0) &= \kappa^- \\ \mu(1) &= \mu^+ & \mu(0) &= \mu^- \\ \kappa'|_{\rho=1} &= D_T J^+ & \kappa'|_{\rho=0} &= D_T J^- \\ \mu'|_{\rho=1} &= D_T J^+ & \mu'|_{\rho=0} &= D_T J^- \end{aligned} \tag{2.36}$$

2.1.6 Optimizers

Unconstrained solvers

Unconstrained solvers deal with a minimization problem with no additional constraints besides both box constraints and final volume fraction achievement. The most common algorithms employed to solve these problems is the steepest descent algorithm, so as to find a local optimum with exploitation of the neighborhood of solutions, rather than exploration.

The steepest descent solver will try to improve a starting solution iteratively, since we will try to find the element of the neighborhood of the solution with best value for the cost function at each step. Once we arrive to some step where the cost function is not reduced by exploring the whole neighborhood, thus the last value of the design variable is selected as the local optima at this point of the topology optimization process.

Consider the minimization problem

$$P_{US} : \left\{\min_x \ f(x)\right\} \qquad x \in \mathbb{R}^n \tag{2.37}$$

clearly, the optimality condition is achieved when $\nabla f(x^*) = 0$, where the gradient also is defined inside $\mathbb{R}^n$. Therefore, the steepest descent algorithm at step $k+1$ is

depicted in Equation (2.38), after defining the use of a line search parameter t enough small to ensure the descent process.

$$x_{k+1} = x_k - t \cdot \nabla f(x_k) \tag{2.38}$$

Inside the field of topology optimization, three unconstrained solvers are found. First of all, the known **Projected Gradient** is a steepest descent based heuristic employed during optimization processes with density as design variable. Specifically, this solver ensures compliance with box constraints by adding an infinite penalization when ρ is larger than 1 or less than 0,

$$min_\rho \; f(\rho) + \delta_{[0,1]}(\rho). \tag{2.39}$$

Furthermore, the steepest descent algorithm is slightly modified by imposing alternating direction heuristic, as depicted in Equation (2.40).

$$\begin{aligned} \rho_{k+1} &= \rho_k - t \cdot \nabla f(\rho_k) \\ \rho_{k+1} &= \max(0, \min(1, \rho_{k+1})) \end{aligned} \tag{2.40}$$

Moreover, the **SLERP** is another unconstrained optimizer usually employed in topology optimization processes, but using now the level set as design variable. Here, the term topological derivative $D_T J(x)$ is recovered, since this could be directly related with the level set due to the nature of its sign in function of the characteristic function values, at the optimality condition (i.e $D_T J(x) \parallel \psi$). For instance, by taking $||\psi|| = 1$, hence a fixed point algorithm is proposed,

$$\begin{aligned} \psi_{k+1} &= \alpha_k \psi_k + \beta_k D_T J(x(\psi_k)) \\ &\text{where } \alpha_k, \beta_k \text{ are taken s.t } ||\psi|| = 1. \end{aligned} \tag{2.41}$$

During the monitoring of convergence between the level set function and the topological derivative, a parameter θ is defined as the angle between ψ_k and $D_T J(x(\psi_k))$, which must tend to zero.

Finally, the **Hamilton-Jacobi** optimizer is the third unconstrained solver, which also uses the level set as design variable. Now the term shape derivative $D_S J(x)$ is recovered in order to be used as the gradient which is multiplied by the line search factor, as exposed in Equation (2.42).

$$\psi_{k+1} = \psi_k - t \cdot D_S J(x_k) \tag{2.42}$$

PDE constrained solver

The PDE constrained optimizer is an extension to constrained optimization problems. It deals with the resolution of the monolithic problem governed by the density and the displacement vector as design variables, in the particular case where the minimization statement is a PDE Constraint Optimization Problem, meaning that given some field the density or the displacement vector, the other one can be computed. Let $a(\rho, u, v)$ and $l(v)$ represent the variational formulation for the internal

and external forces of the equilibrium equation, respectively. Thus, the monolithic problem formulation reads as

$$P_{MC} : \left\{ \begin{array}{c} \min_{\rho,u} \ l(v) \\ s.t \ a(\rho, u, v) = l(v) \\ 0 \leq \rho \leq 1 \end{array} \right\}. \tag{2.43}$$

Nevertheless, a canonical formulation is obtained by transforming previous formulation into a reduced one, imposing a relationship $u = u(\rho)$. Under this assumption, the *adjoint problem* formulation is employed in order to compute the gradient of the cost function and therefore to propose a gradient-based algorithm.

Consider Equation (2.44), where the adjoint problem is defined as

$$\left\{ \begin{array}{c} \min_{x,y} \ J(x, y) \\ s.t \ A(x) \cdot y = b(x) \end{array} \right\} \rightarrow \text{ with A invertible } \rightarrow \left\{ \min_x \ J(x, A^{-1}(x) \cdot b(x)) \right\}. \tag{2.44}$$

In order to solve the minimization problem, the gradient computation of J is defined as

$$\nabla_x J = \frac{\partial J}{\partial x} + \frac{\partial J}{\partial y} \frac{\partial y}{\partial x}. \tag{2.45}$$

By manipulating with elementary calculus the derivative $\frac{\partial y}{\partial x}$, one gets the expression depicted in

$$\frac{\partial y}{\partial x} = A^{-1}(x) \cdot \left[-\frac{\partial A(x)}{\partial x} A^{-1}(x) b + \frac{\partial b(x)}{\partial x} \right]. \tag{2.46}$$

Combining all expressions above, the gradient of J is completely formulated in Equation (2.47) in function of a new parameter p.

$$\nabla_x J = \frac{\partial J}{\partial x} + p \cdot \left[-\frac{\partial A(x)}{\partial x} y + \frac{\partial b(x)}{\partial x} \right] \text{ with } A(x) p = \frac{\partial J}{\partial y} \tag{2.47}$$

Finally, below is exposed the whole algorithm of the PDE constrained optimizer.

1. Compute the *state*, *primal* variable y - Equation (2.48).

$$A(x) \cdot y = b \tag{2.48}$$

2. Solve the *adjoint* - Equation (2.49).

$$A(x) \cdot p = \frac{\partial J}{\partial y}(x, y) \tag{2.49}$$

3. Compute the gradient of J - Equation (2.50).

$$\nabla_x J = \frac{\partial J}{\partial x} + p \cdot \left[-\frac{\partial A(x)}{\partial x} y + \frac{\partial b(x)}{\partial x} \right] \tag{2.50}$$

4. Update x - Equation (2.51).

$$x_{k+1} = x_k - \alpha \cdot \nabla J \text{ with } \alpha \text{ s.t } J(x_{k+1}) < J(x_k) \tag{2.51}$$

General constrained solvers

Constrained solvers deal with a minimization problem with the inclusion of additional constraints that the design variable must fulfill, i.e $c(x) = 0$. This new set of equation justifies the use of Lagrange multipliers to solve the minimization problem. In fact, although the Lagrange factors are not the unkowns of interest for us, we will need to compute them in order to cover the whole solution of the system. This phenomenon is called the dual problem and it is presented in Equation (2.52). A parametrization of objective function f with the constraints and Lagrange multipliers is presented.

$$\left\{ \begin{matrix} \min_x \ f(x) \\ s.t \ c(x) = 0 \quad (\lambda) \end{matrix} \right\} \rightarrow D(\lambda) = \min f(x) + \lambda^T \cdot c(x) = \min L(x, \lambda) \tag{2.52}$$

where $L(x, \lambda)$ is usually known as the Lagrangian. We will want to minimize the Lagrangian in function of the design variable and later to maximize the resulting expression for λ, since this implicitly allows to fulfill the constraints,

$$\begin{matrix} \max(D(\lambda)) = \max_\lambda \ \min_x \ f(x) + \lambda^T \cdot c(x) \\ \frac{\partial L}{\partial \lambda} \equiv \frac{\partial D}{\partial \lambda} = c(x) = 0 \end{matrix}. \tag{2.53}$$

The optimum λ^* obtained will give the information of how important is the constraint. Finally, in Equation (2.54), it is depicted the optimality condition that must be fulfilled in the topology optimization process. Besides, the expressions for the evolution of design variables and Lagrange multipliers are exposed

$$\begin{matrix} \frac{\partial L}{\partial x} = \frac{\partial f}{\partial x} + \lambda \frac{\partial L}{\partial x} = 0 & \text{Primal Opt.Cond.} \\ \frac{\partial L}{\partial y} = c(x) = 0 & \text{Dual Opt. Cond.} \end{matrix} \tag{2.54}$$

$$\begin{matrix} x_{k+1} = x_k - t \cdot \frac{\partial L}{\partial x}(x_k, \lambda) & \text{s.t} & L(x_{k+1}) < L(x_k) \\ \lambda_{k+1} = \lambda_k + c(x_k) & \text{s.t} & |c(x_{k+1})| < |c(x_k)| \end{matrix}. \tag{2.55}$$

Inside the Swan's git repository, four types of constrained solvers are found as shown in Table 2.1. The first one, Augmented Lagrangian, is simultaneously divided into two variants: Alternating Primal-Dual and Dual Nested In Primal. Both cases need the definition of an unconstrained solver using density or level set, complementary to the constrained optimizer. Nevertheless, the remaining solver typologies do not need the additional use of an unconstrained solver.

Solver	Definition	Unconstr. solver
Augmented Lagrangian	Similar to the standard Lagrangian, with the inclusion of squared terms (quadratic penalty function). Two variants of this approach are considered: Alternating Primal-Dual and Dual Nested In Primal	ρ / ψ
MMA	The MMA Optimizer is an algorithm available in MATLAB, usually employed as a black box.	Not needed.
IPOPT	The Interior Point Optimizer is an algorithm available in MATLAB for non-linear large size problems.	Not needed.
Fmincon	This optimizer includes several algorithms from other solvers, with the IPOPT one between them. It is also a MATLAB implementation of constrained multi-variable solver.	Not needed.

TABLE 2.1: General constrained solvers used in topology optimization techniques.

Finally, below it is provided a summary of the numerical optimization of the only dependent optimizer Augmented Lagrangian. First of all, the Lagrangian function is now defined as indicated in Equation (2.56). [13]

$$L(x, \lambda, \rho) = f(x) + \lambda c(x) + \frac{\rho}{2} c(x)^2 \tag{2.56}$$

Therefore, the quadratic penalty term is included in the topology optimization problem definition, as shown in Equation (2.57).

$$\left\{ \begin{matrix} \max_\lambda \min_x \left(f(x) + \lambda c(x) + \frac{\rho}{2} c(x)^2 \right) \\ s.t\ c(x) = 0 \end{matrix} \right\} \tag{2.57}$$

In this case, the optimality conditions read $\nabla f(x) + [\lambda + \rho c(x)] \nabla c(x) = 0$. Line search methods are used to find an optimizer iteratively, obtaining a better solution from an initial guess, and terminating the algorithm once the variable satisfies the optimality conditions and the constraints. The continuity of improvement on the current iterate is guaranteed if following expressions are satisfied.

$$\nabla c(x_k)^T p_k \simeq 0 \tag{2.58}$$

$$\nabla f(x_k)^T p_k < 0 \tag{2.59}$$

2.2 Numerical methodology

2.2.1 Unfitted mesh method

Topology optimization of structures deals with a combination of several domains, which are enclosed by a larger reference domain Ω in $\mathbb{R}^2$ or $\mathbb{R}^3$. Initially, the original structure may be defined inside all Ω, but as the algorithm process evolves, the characteristic function will assign void values to some control volumes. Besides, it might appear some boundary control volumes with a combination of *black* and *white* values. Here is where the term cut mesh appears.

The known unfitted finite element method variant will be employed when dealing with cut meshes, so as to facilitate computations on complex geometries of the structure for boundaries and interfaces treatment, as well as the discretization of PDEs on surfaces. [14]

In the unfitted mesh method, the boundary of the material domain will be represented on a background grid, related with the reference domain Ω, by means of the density or a level set function. The starting point consists in discretizing the different geometries.

First of all, an arbitrary geometric description is immersed in the background grid, typically a structured mesh in order to facilitate data handling and communication. Hence, to describe stationary boundaries on this mesh, we will now focus on level set, for instance. The location of the boundary is defined by the zero level set of a function $\psi : \mathbb{R}^n \to \mathbb{R}$, as depicted in Equation (2.60).

$$\left\{\begin{array}{lll} \psi(x) < 0 & \text{if} & x \in \Omega^{mat} \\ \psi(x) = 0 & \text{if} & x \in \Gamma \\ \psi(x) > 0 & \text{if} & x \in \Omega^{void} \end{array}\right\} \tag{2.60}$$

Now, the interface is approximated by sub-triangulation of cut elements. The values of the level-set function in element nodes are employed to classify elements into one of the following three categories: fully contained in Ω^{mat} (all nodes with negative level set value), fully contained in Ω^{void} (all nodes with positive level set value) and intersected by Γ (some nodes with positive or negative level set values). For the last category, a sub-triangulation of the element is performed in order to apply standard quadrature rules. In Figures 2.1 and 2.2 this procedure is exposed.

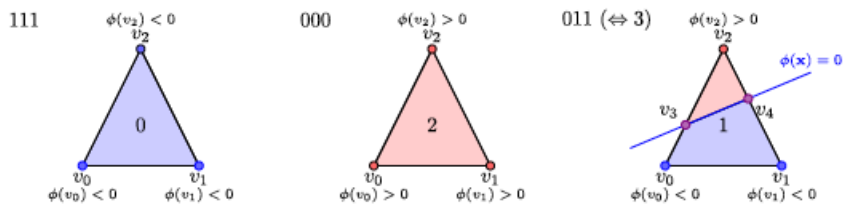


FIGURE 2.1: Sub-triangulation of triangular elements. Cells fully contained in Ω^{mat} are marked with 0, cells fully contained in Ω^{void} are marked with 2, and cells that are intersected are marked with 1. [14]

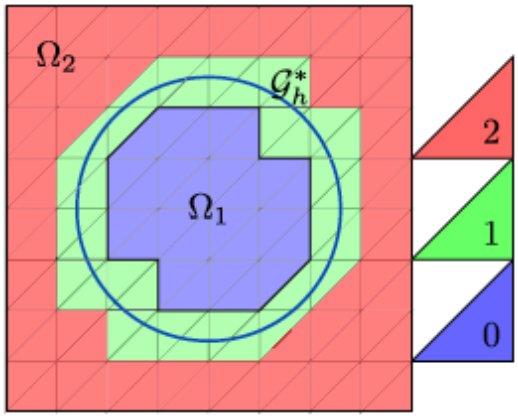


FIGURE 2.2: Example of resulting unfitted mesh for $\psi(x) = x^2 + y^2 - 1$ of circular Ω^{mat}. [14]

Only linear intersections of the zero level-set with elements is considered, obtaining one straight line segment per intersected elements by linear interpolation between the level set values in the nodes connected to the facet. Hence, a sub-triangulation is built for each cut cell part, as shown in Figure 2.3. Later, new sub-triangle elements are classified in inner or outer elements depending on the level set value, see Figure 2.4.

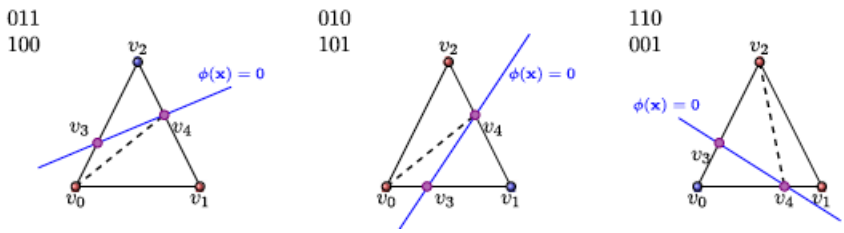


FIGURE 2.3: Straight intersections in 2D, along with the corresponding cell flags and sub-triangulation. [14]

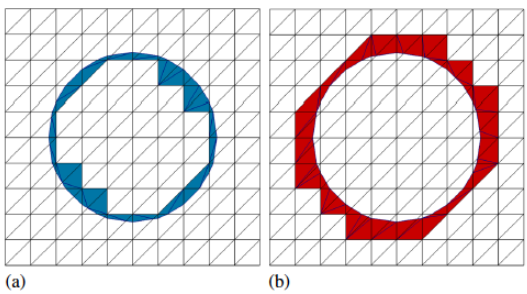


FIGURE 2.4: Cell sub-triangulation for (a) Ω^{mat} and (b) Ω^{void} example. [14]

Finally, integrals over sub-triangulation will be evaluated by using two mappings: the linear affine mapping (χ_p) and the mapping between reference element and the *parent* cell (χ_w). χ_p transforms a quadrature rule defined on the reference element in function of quadrature points and weights into a quadrature rule on the sub-triangle element, as seen in Figure 2.5. Besides, χ_w is employed to map the quadrature points

defined on the physical sub-triangle element to its location in the reference domain of the whole *parent*.

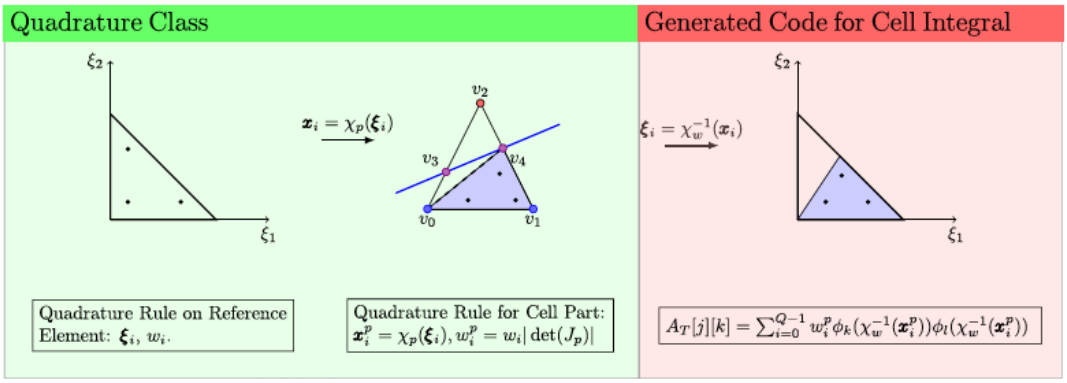


FIGURE 2.5: Schematics of integration over the sub-triangulation. [14]

In Figure 2.6 typical nomenclature referring to each typologies inside the overall mesh is exposed. Note that a classification into inner and cut elements is analogous between the 2D mesh and the 1D boundary cut mesh.

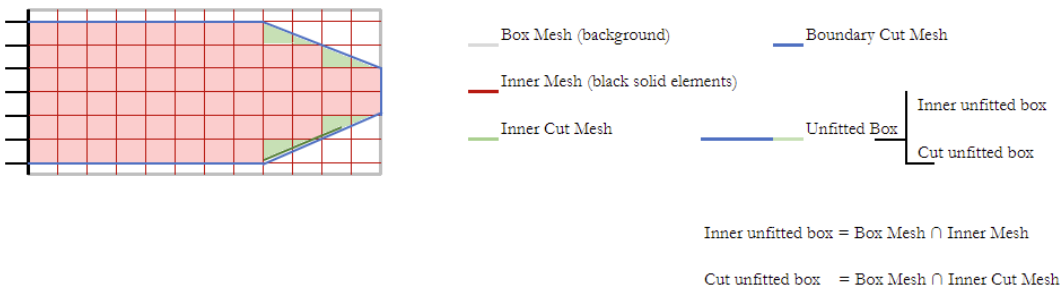


FIGURE 2.6: Summary of the nomenclature used for different mesh typologies.

2.2.2 The Diffusion-Reaction equation

When both the compliance and the perimeter appear in the objective function, a mathematical formulation for the methodology must be defined. In this case, the general diffusion-reaction equation will fit with the topology optimization model. [15]

Consider the optimization problem

$$\min_{u_i \in \Omega; \sum u_i = 1} \left\{ \left| f - \sum_{i=1}^{l} c_i u_i \right| + \frac{\alpha}{2c} \sum_{i=1}^{l} F(u_i) \right\}. \tag{2.61}$$

In previous expression, for numerical difficulties purposes it is desirable to replace F with an alternative model F_ϵ, dependent on the perimeter minimum mesh size ϵ parameter, so as to allow the problem to be solved by approximating its minimizers.

One way for choosing F_ϵ is the well-known Modica-Mortola method, as shown in Equation (2.62).

$$F_\epsilon = \int_\Omega \frac{1}{\epsilon} W(u(x)) + \epsilon |\nabla u(x)|^2 dx \tag{2.62}$$

where the double well potential $W : \mathbb{R} \to [0, +\infty)$ satisfies continuity conditions, $W(t) = 0$ if and only if $t \in \{0, 1\}$ and there exist $L > 0$ and $T > 0$ such that $W(t) \geq L|t| \; \forall \, t \in \mathbb{R}$ with $|t| \geq T$.

Another way for choosing this functionals is exposed in Equation (2.63).

$$\bar{F}_\epsilon(u) = inf_{v \in H^1(\Omega)} \int_\Omega \epsilon |\nabla v|^2 + \frac{1}{\epsilon}(v^2 + u(1 - 2v)) d\Omega \tag{2.63}$$

The infimum in the definition of F_ϵ is attained in some virtual function $v \in H^1(\Omega)$. This justifies the uniqueness of this function, as F_ϵ solves the Euler-Lagrange PDE depicted in Equation (2.64), in the weak sense.

$$\left\{ \begin{matrix} -\epsilon^2 \Delta v + v = u & \text{in } \Omega \\ \partial_n v = 0 & \text{on } \partial\Omega \end{matrix} \right\} \tag{2.64}$$

Thus, the known definition of the perimeter is obtained, in function of the minimum mesh size. At this point, there are two possibilities to define at which boundaries the perimeter will contribute as a penalty term to the cost function: total and relative perimeter. In the case of total perimeter, both the boundary of background mesh with black values and the transition regions between black and white values inside the domain will be taken into consideration, so as defining Robin boundary conditions at $\partial\Omega$. Regarding relative perimeter, only the transition region between black and white values inside the domain is considered, thus defining Neumann boundary conditions at $\partial\Omega$.

2.2.3 Incremental scheme

During topology optimization some numerical issues will appear if the final optimization constraints are imposed directly in one step, thus defining the need to use several steps, where the objective function will be updated at the beginning of each one with sudden changes, and hence it will be optimized through the evolution of each step. Typically, each step can represent a determined reduction in the volume fraction, among others applications. For instance, considering an optimization problem where the volume is reduced an amount 50% in 10 steps, thus at each step the objective function will be minimized through sequential reductions of 5% related to the volume.

Therefore, an incremental optimization scheme is implemented, which calculates the target values for the constraints at each step by using Equation (2.65). [6]

$$x_k = x_0 + t(x^* - x_0) \tag{2.65}$$

Where x_k stands for the target value at step k, $t \in [0, 1]$ is the interpolation factor and x_0, x^* are the initial and target value for the constraint, respectively.

In Figure 2.7 the cost function related to a minimum compliance problem example is represented in terms of the number of iterations, in order to expose how the incremental scheme approach works. Clearly, 3 steps are employed since 3 unexpected jumps are produced at iterations 0, 35 and 140, approximately. At each of these jumps the volume is reduced a certain targeted value. Besides, for each step the solver is trying to find the best material configuration such that the cost function becomes as low as possible. Thus, inside the minimization process of each jump, the volume remains constant.

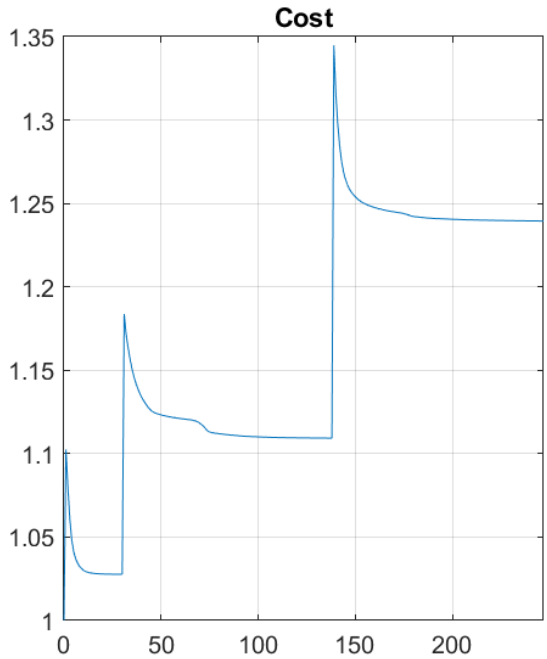


FIGURE 2.7: Incremental scheme example applied to the minimum compliance problem of a 2D cantilever beam with 3 steps.

In swan's code the following parameters can be controlled at each step: target volume, minimum mesh size parameter in the perimeter function, target homogenized elasticity matrix and tolerances values. For each parameter a mathematical law for its evolution can be defined separately, although the volume typically evolves as a linear function.

Lastly, there are different available interpolation laws for the interpolation factor, mainly the following ones: linear, exponential, potential and free. Nevertheless, the linear and potential laws are the ones more extended in its use. Let N to denote the total number of steps, k the current and t_0 the interpolation value at the first step. Hence, in Equations (2.66), (2.67), (2.68) and (2.69), the mathematical expressions for the linear, exponential, potential and free interpolation laws are depicted, respectively.

$$t_k = \begin{Bmatrix} 1 & N = 1 \\ t_0 + \frac{1-t_0}{N-1}(k-1) & N > 1 \end{Bmatrix} \tag{2.66}$$

$$t_k = \begin{Bmatrix} 1 & N = 1 \\ t_0^{\frac{N-k}{N-1}} & N > 1 \end{Bmatrix} \tag{2.67}$$

$$t_k = \begin{Bmatrix} 1 & N = 1 \\ 1 - (1 - t_0)\left(\frac{N-k}{N-1}\right)^p & N > 1 \end{Bmatrix} \qquad \forall\, p \geq 1 \tag{2.68}$$

$$t_k = \begin{Bmatrix} 1 & k = N \\ 0 & \text{otherwise} \end{Bmatrix} \tag{2.69}$$

2.3 Additive manufacturing in topology optimization

2.3.1 Additive manufacturing

The Additive Manufacturing (AM) technology is referred as 3D CAD data printing technology, which creates new industrial designs layer-by-layer by means of liquid, wire, sheet and powdered materials, and without a need of subsequent processing tasks. Therefore, different prototypes can be obtained with a reduced manufacturing time, near 100% of material utilization and a relative geometric freedom of design. [16]

The most widely used materials in AM are polymers, although in recent years metals and ceramics have also been introduced due to its capability to withstand higher loads and temperatures. Nevertheless, ceramics models are difficult to manufacture when complex geometries are involved. Regarding AM applicability, its industrial development has located it in a large variety of sectors, such as consumer products, electronics, motor vehicles, medical devices, aerospace and architecture.

From the CAD model to the actual part, AM technology is divided into the following steps:

- CAD model design. The output model must be a 3D or surface representation of the real design.
- Conversion for AM accepted file type. STL is the standard file type, which translates the CAD model to a controlled-size mesh of triangles.
- Transfer STL files to AM machine, after verifying size and build orientation.
- AM machine setup of parameters to achieve the tolerances of the design (layer thickness, orientation, energy, timing and roller speeds).
- Build phase.
- Removal of the part, following the corresponding safety protocols.
- Post-processing of AM parts (curing, sintering and cleaning).

A large variety of AM systems have been introduced to the commercial market, which classification is proposed by the ASTM F42 into the seven areas exposed below.

Powder bed fusion

Powder bed fusion (PBF) systems employ an electron beam or a laser source to melt and fuse powdered materials together in a selective manner, see Figure 2.8. These processes involve repetition of deposition layer after layer, by means of different mechanisms, such as a roller or a blade, and a hopper for fresh material supply.

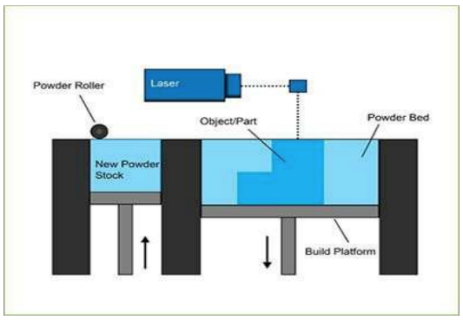


FIGURE 2.8: Powder bed fusion (PBF) methods. [16]

Directed energy deposition

Directed energy deposition (DED) is a harder system used to add additional material to current parts. The DED process consists of a multi axis arm nozzle, which provides melted material on the desired surface, and from any angle. The process is typically employed with metals in wire or powder form. Material cooling rates are relatively fast, between 1000-5000 Celsius/second.

Material extrusion

Fused deposition modelling (FDM) is an extrusion process where material is drawn through a nozzle and hence deposited layer by layer, after the material is heated, see Figure 2.9. The base platform can move vertically after each new layer is provided, whereas the nozzle moves horizontally. In FDM the material must be deposited through the nozzle under constant pressure and continuous stream in order to assure the quality of the final product. Moreover, layers are bonded either by the use of chemical agents or temperature control.

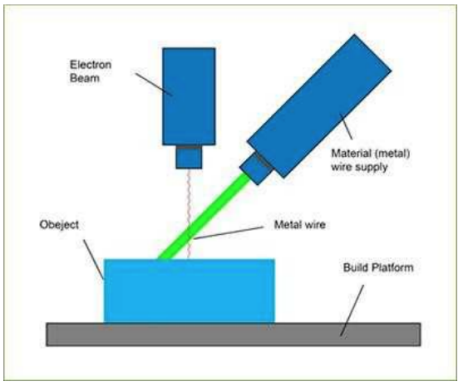


FIGURE 2.9: Metal extrusion process. [16]

Vat polymerization

Vat polymerization employs liquid photopolymer resin, out of which the part is manufactured layer by layer. Ultraviolet light is used to cure or harden the resin, whereas the base platform moves the part downwards after the cure process of each new layer is completed, see Figure 2.10. In this process, support structures will often be added, since the system employs liquid to form objects during the build phase.

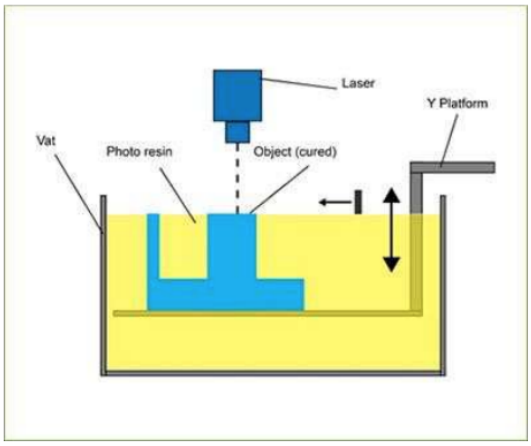


FIGURE 2.10: Vat polymerization process. [16]

Binder jetting

The binder jetting system uses two materials: a powder one (build material) and a binder (liquid), which acts as an adhesive between powder layers. A print head moves along the horizontal plane of the machine and provides alternating layers of the build material and the binding one, see Figure 2.11. After each layer is completed, the platform is lowered. Although this process is faster than others, additional post-processing may add significant time to the overall process due to the binding material characteristics.

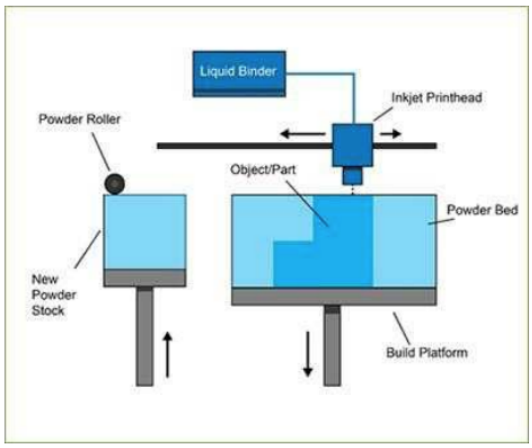


FIGURE 2.11: Binder jetting process. [16]

Material jetting

Material jetting manufactures parts similarly to a 2D ink jet printer. Material is jetted onto the build platform in a continuous or Drop on Demand (DOD) manner, see Figure 2.12. Therefore, the material solidifies in the platform and the model is built layer by layer. Material is provided from a nozzle moving horizontally across the build platform, and hence the layers are cured using ultraviolet light. The number of materials available to use is limited, since this shall be deposited in drops.

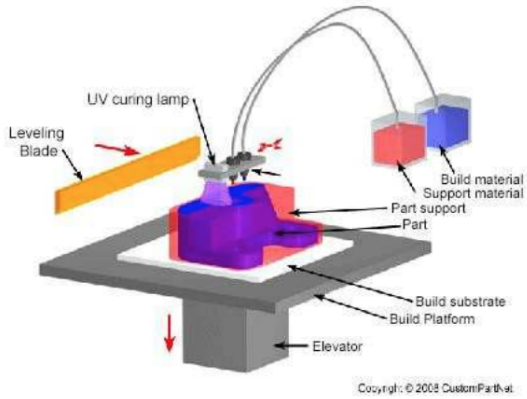


FIGURE 2.12: Material jetting. [16]

Sheet lamination process

Sheet lamination processes include ultrasonic additive manufacturing (UAM) and laminated object manufacturing (LOM), which produce aesthetic and visual models following the means depicted below.

- The UAM system employs sheets or ribbons of metal, bounded together by means of ultrasonic welding. CNC machining will be required during the process.
- The LOM employs a similar layer by layer approach but uses paper as material and adhesive instead of welding. Cross hatching method is used during the process to allow for easy removal after the part is manufactured.

2.3.2 AM Length Scale constraints

Topology Optimization techniques can be applied to create new designs and therefore manufacture them by means of AM technology. Nevertheless, some additional constraints must be taken into consideration in order to assure actual manufacturability. One of these restrictions is known as the minimum length scale constraint.

Consider an initial design which is being optimized through the minimum compliance problem. Since topology optimization are heuristic-based algorithms, a large variety of solutions may exist if the numerical problem is not enough constrained. Therefore, regarding the actual compliance minimization problem, the tendency of the algorithm near the final volume fraction achievement would consist in the creation of several internal bars. Besides, the compliance function becomes lower as

the number of internal bars increases. Thus, the optimality condition is virtually achieved at an infinite number of bars. Numerically, this is translated to the obtaining of a large number of bars inside the initial domain. Indeed, in order to keep several bars inside a fixed domain it will be necessary to decrease their length scales (i.e to reduce the thickness of bars). Here is where the minimum length scale constraint will become relevant, since AM technology would not be able to create a new design of any value for the minimum length scale, mainly due to its sensitivity during material deposition, among others (see Figure 2.13).

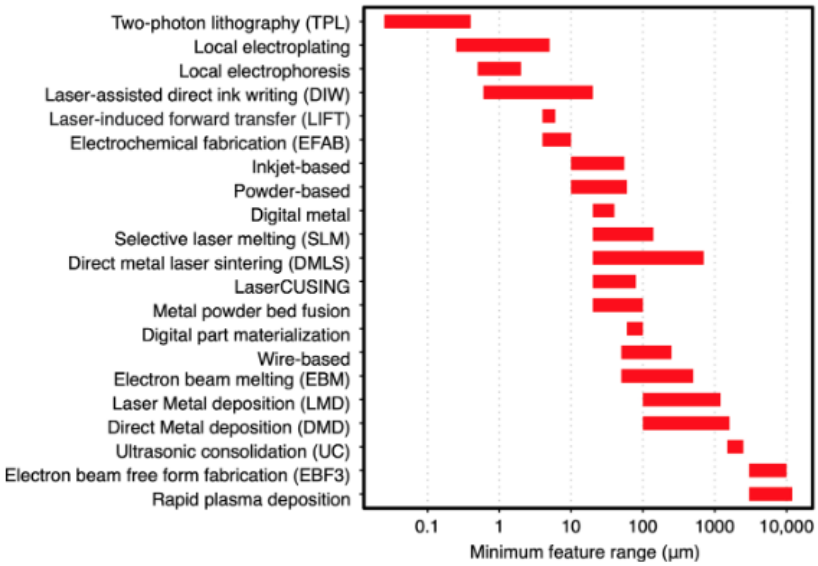


FIGURE 2.13: Comparison of the minimum feature sizes in metal printing via various AM technologies. [17]

Hence, a remedy to decide at which number of bars stop for the final volume fraction is needed. There are several ways to proceed depending in the type of design variable or whether the length scale is controlled as a shape functional or a constraint.

Perimeter as objective function penalty term

Now, we recall from Section 2.1.3 the use of Perimeter as a contribution to the compliance shape functional in order to avoid the formation of small geometrical scales in structures. The overall cost function is exposed

$$J = f^T \cdot u + \frac{\alpha}{2\epsilon} \int_\Omega (1 - \rho_\epsilon)\rho \cdot d\Omega \tag{2.70}$$

where $\alpha \in [0, +\infty)$ is some parameter that defines the degree of perimeter contribution into the optimization problem and ϵ stands for the minimum mesh size. Basically, the perimeter is a penalization of grey regions defined by the density, or the regularized density. Indeed, a design with a large variety of internal bars would consist of a larger region dominated by grey values. Therefore, the minimum length scale of the model would become greater by introducing the perimeter contribution to the cost function, thus minimizing the grey region inside the model and consequently increasing the geometric scales.

A relationship between the length scale and parameter α is not possible to be obtained by means of an explicit expression, neither with a numerical algorithm. This is due to the strongly dependence of the perimeter regarding the mesh employed, the accuracy of the numerical simulation, the final fraction volume and the ϵ parameter incremental scheme. Thus, given a simulation set-up, different designs are obtained by changing the value of α, iteratively.

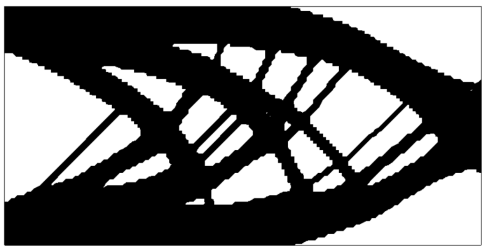


FIGURE 2.14: Minimum compliance problem example applied to a cantilever beam.

FIGURE 2.15: Isotropic perimeter penalization applied to the cantilever beam example (left).

Penalty functional with level set

The previous procedure to control the length scale of the optimized structure was monitored by including a penalty term in the compliance function, without controlling directly the desired value of the length scale. Nevertheless, enforcing the minimum thickness as a single penalty functional approach is exposed below, using the level set as design variable and introducing a parameter to set the minimum length, d_{min}.

Following the idea proposed in [18] by Feppon, the minimum thickness requirement is imposed with the minimization of the penalty functional $P_{MinT}(\Omega)$, assuring minimum thickness under constraints on the volume and compliance of shapes, as depicted in Equation (2.71).

$$P_{ML} : \left\{ \begin{array}{c} \min \ P_{MinT}(\Omega) \\ s.t \ C(\Omega) \leq g_{max} \\ Vol(\Omega) \leq V_{max} \end{array} \right\} \tag{2.71}$$

Where the shape functional, compliance of shape constraints and volume constraints are defined in Equations (2.72), (2.73) and (2.74), respectively.

$$P_{MinT}(\Omega) = -\int_{\Omega} d_{\Omega}^{2} \max(d_{\Omega} + d_{min}/2, 0)^{2} dx \tag{2.72}$$

$$C(\Omega) = \int_{\Omega} (2\mu\epsilon + \lambda Tr(\epsilon) \cdot I) : \epsilon dx \tag{2.73}$$

$$Vol(\Omega) = \int_{\Omega} dx \tag{2.74}$$

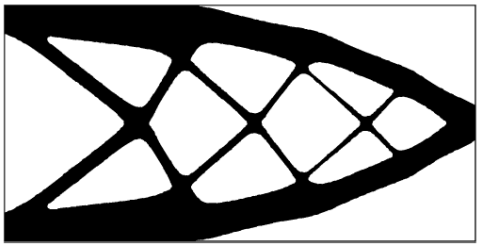


FIGURE 2.16: Optimized shape without including minimum thickness constraint. [18]

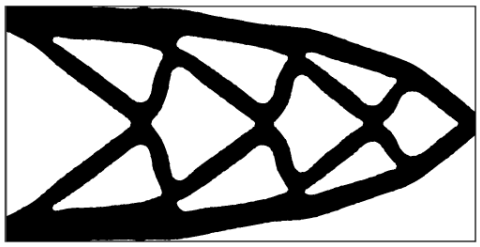


FIGURE 2.17: Optimized shape with minimum thickness constraint ($d_{min} = 0.1$). [18].

Constraint functional with density

Here it is proposed a procedure to guarantee minimum length requirements by imposing an additional constraint, rather than a contribution to the objective function. This new approach limits the design's perimeter with the computation of the total variation of density field, $|\partial\Omega_S|$, defining the perimeter as a measure of the boundary of the solid region. [19]

Indeed, an upper bound constraint on the perimeter excludes the apparition of microscopic perforations in the solution. The total variation of density field deals with a SIMP continuous interpolation model for density as design variable, approaching the perimeter as the amount of transitional material is forced to zero. The total variation of the density function is depicted

$$|\partial\Omega_S| = \int_{\Omega\setminus\Gamma_J} g_h(\nabla\rho, \xi)d\Omega + \int_{\Gamma_J} j(<\rho>, \xi)d\Gamma \leq \bar{P} \tag{2.75}$$

where ξ and $<\rho>$ accounts for a smoothing parameter circumventing numerical problems and the jump in ρ across Γ_J, respectively. Besides, g_h and j are functions defined in Equations (2.76) and (2.77), respectively, using parameter h as the size of a finite element.

$$g_h(w, \xi) = \left[(1 + 2\xi)w^T w + \frac{\xi^2}{h^2}\right]^{\frac{1}{2}} - \frac{\xi}{h} \tag{2.76}$$

$$j(r, \xi) = \left[(1 + 2\xi)r^2 + \xi^2\right]^{\frac{1}{2}} - \xi \tag{2.77}$$

Constraint functional with a density based projection method

Finally, here it is depicted another procedure to guarantee minimum length scales by means of an additional geometric constraint, using a robust formulation based in the projection method, guaranteeing well-convergence of results and without large gray scale areas. Therefore, this will represent the most recommended approach to impose a length scale requirement in a design, since this procedure shall guarantee

such scales not only inside the black region, but also length scales inside the void domain. Besides, this approach would also solve local features convergence issues due to mesh refinement, so as to ensure manufacturing tolerant designs.

The following projection method is based in the density as design variable and it follows a filtering-threshold topology optimization scheme, defining a 3-field scheme with the design variable ρ, a filtered version ρ_ϵ and a projected version ρ_P. Note that the design variable shall come from an adjoint sensitivity analysis, thus defining an optimization problem strongly dependent on the mesh size.

Considering the neighborhood set $\mathbb{N}_i$ within a filter domain of an element i, thus Zhou et. al. propose the expression to filter the design variable exposed in Equation (2.78). [20]

$$\rho_{\epsilon,i} = \frac{\sum_{j\in\mathbb{N}_i} w(x_j)V_j\rho_j}{\sum_{j\in\mathbb{N}_i} w(x_j)V_j} \tag{2.78}$$

Where $w(x_j) = R_f - |x_i - x_j|$ is a weighting function of a filter with radius R_f. Besides, V_j stands for the volume of neighbour element j and x_i, x_j for the cell coordinates of considered element i and neighbour element j. Thus, a threshold projection into the $0/1$ space is defined as a function of previous filtered density $\rho_P = f(\rho_\epsilon)$, as shown in Equation (2.79), proposed by Wang et. al. [21]

$$\rho_{P,i} = \frac{\tanh(\beta\eta) + \tanh(\beta(\rho_{\epsilon,i} - \eta))}{\tanh(\beta\eta) + \tanh(\beta(1-\eta))} \tag{2.79}$$

In previous expression, β represents a Heaviside steepness factor and η the projection threshold. The key idea is as follows: for a given threshold η and sufficiently large value of β hence,

- All $\rho_{\epsilon,i} > \eta$ verifies that $\rho_{P,i} = 1$.
- All $\rho_{\epsilon,i} < \eta$ verifies that $\rho_{P,i} = 0$.

Some special cases are deduced from these conditions. Indeed, for $\eta = 0$ a dilation of the domain is produced, guaranteeing minimum length scales at the solid region, whereas for $\eta = 1$ an erosion of the domain appears, guaranteeing minimum length at the void region. Therefore, an initial problem may be produced here, as it is not possible to control both solid and void length scales at a first instance. Hence, minimum length scales will be imposed separately for an inflection region at solid and void phase, defining a threshold interval $(\eta_{min}, \eta_{int}, \eta_{max})$. Specifically,

- Given a value $\eta = \eta_{max}$, hence all $\rho_{\epsilon,i} > \eta_{max}$ will define an inflection region where $\rho_{P,i} = 1$ and $\nabla\rho_{\epsilon,i} = 0$.
- Given a value $\eta = \eta_{min}$, hence all $\rho_{\epsilon,i} < \eta_{min}$ will define an inflection region where $\rho_{P,i} = 0$ and $\nabla\rho_{\epsilon,i} = 0$.

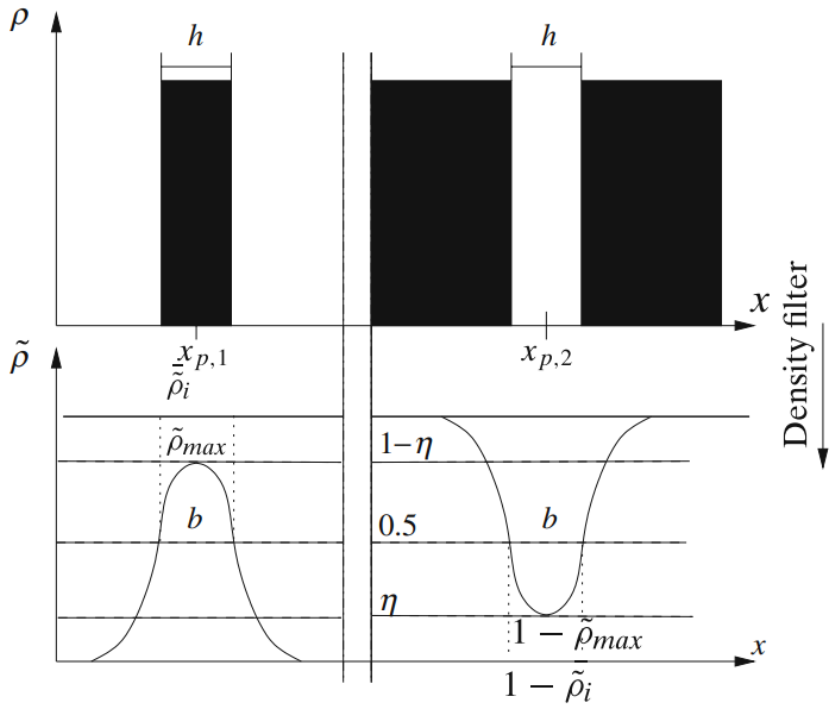


FIGURE 2.18: Density filtering of solid and void phases and their Heaviside projections. [21]

Typically, for a given value of $\eta^* \in (0, 0.5)$, the threshold interval is defined as $\eta_{min} = \eta^*$, $\eta_{int} = 0.5$, $\eta_{max} = 1 - \eta^*$ (see Figure 2.18). In order to identify previous inflection regions, the structural indicator functions exposed in Equations (2.80) and (2.81) are employed, so as to impose the minimum length requirements in solid and void phases, respectively.

$$I^{solid} = \rho_P \cdot \exp\{-c \cdot |\nabla \rho_\epsilon|^2\} \tag{2.80}$$

$$I^{void} = (1 - \rho_P) \cdot \exp\{-c \cdot |\nabla \rho_\epsilon|^2\} \tag{2.81}$$

The workable range of steepest parameter c has been previously studied, and this shall be defined as $c \in [0.4, 1.4] \cdot \left(\frac{R}{h}\right)^4$, where h is the mesh element size. [20] For instance, $c = r^4$ would provide effective results to the optimization problem.

Thus, the geometric constraints defined in Equations (2.82) and (2.83) will be included through the definition of the optimization problem, for solid and void phases respectively. In these expressions, n stands for the total number of elements.

$$g^{solid} = \frac{1}{n} \sum_{i \in \mathbb{N}} I_i^{solid} \cdot [\min((\rho_{\epsilon,i} - \eta_{max}), 0)]^2 \leq \epsilon \tag{2.82}$$

$$g^{void} = \frac{1}{n} \sum_{i \in \mathbb{N}} I_i^{void} \cdot [\min((\eta_{min} - \rho_{\epsilon,i}), 0)]^2 \leq \epsilon \tag{2.83}$$

Where $\epsilon > 0$ is a residual parameter. Lastly, the minimum length scales problem is defined in Equation (2.84).

$$P_{MLS} : \left\{ \begin{array}{c} \min_{\rho} \ f^T \cdot u(\rho) \\ s.t \ K(\rho) \cdot u(\rho) = F(\rho) \\ V_f \leq V^* \\ g^{solid} \leq \epsilon \\ g^{void} \leq \epsilon \\ \rho \in [0,1] \end{array} \right\} \tag{2.84}$$

2.3.3 AM Overhang constraints

The use of perimeters in 3D printing is also useful concerning one of the significant problems in AM technology, *overhanging* - see Figure 2.19.

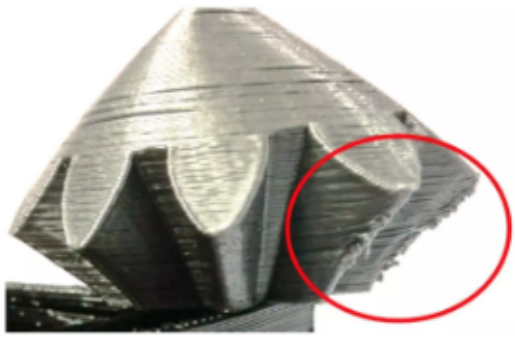


FIGURE 2.19: Overhanging phenomena. [15]

If in some region of the part the angle between the vertical axis and tangent vectors to the object is larger than 45 degree, hence it may happen that new layers do not stick firmly together, leading to the apparition of small bulges. In this case, the tool known as *anisotropic perimeter* will play an important role to overcome this issue.

The known *anisotropic total variation* of the characteristic function with respect to the elastic conductivity of the material is the main tool to define the functionals computing the *anisotropic perimeter* [15]. These functions are unique and solve the Euler-Lagrange PDE in the weak sense.

$$\left\{ \begin{array}{cc} -\epsilon^2 \nabla (A \cdot \nabla u) + u = g & \text{in } \Omega \\ \partial_{A_n} u = 0 & \text{on } \partial\Omega \end{array} \right\} \tag{2.85}$$

where A corresponds to the elastic conductivity matrix. In general, A is a symmetrical matrix defined in 2D or 3D. The main purpose of this matrix will be to define virtually in which direction the material has the maximum stiffness intensity, so as to prioritize the direction where the bars emerging from the minimum compliance problem will be placed.

Nevertheless, in practice A may be defined as a diagonal matrix, where each term

defines the stiffness intensity penalty in each main direction of the coordinate system. Therefore, the component with the maximum stiffness would define the direction tendency of the bars obtained from the optimization process. Thus, one may rotate such matrix later with the corresponding Euler angles in order to orientate these bars in a determined direction. In practice, when avoiding overhanging phenomena one wants to prioritize the orientation along the vertical axis, hence using a conductivity matrix similar to the one depicted in Equation (2.86) for a 2D problem.

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 100 \end{bmatrix} \tag{2.86}$$

Regarding the topology optimization algorithm, the elastic conductivity matrix will be employed prior to compute the left hand side stiffness matrix, after selecting the type of filter to use during the resolution of the diffusion-reaction PDE. In this step, the known elementary *anisotropic stiffness matrix* will be defined as shown in Equation (2.87).

$$\underline{\underline{K_{el}}} = \iiint_{\Omega} \underline{\underline{B^T}} \cdot \underline{\underline{A}} \cdot \underline{\underline{B}} \cdot d\Omega \tag{2.87}$$

In order to extend the anisotropic perimeter to any direction of the domain, below is exposed the required mathematical formulation. Let Ω to represent a 2-dimensional domain referenced with a global axis system XY. Let also α to represent the rotation of a local axis system $X'Y'$ with respect to the global one, as depicted in Figure 2.20. Therefore, a local matrix A' is defined such that the topologic bars resulting from the optimization process are oriented along the local axis X' (see an instance in Equation (2.88)).

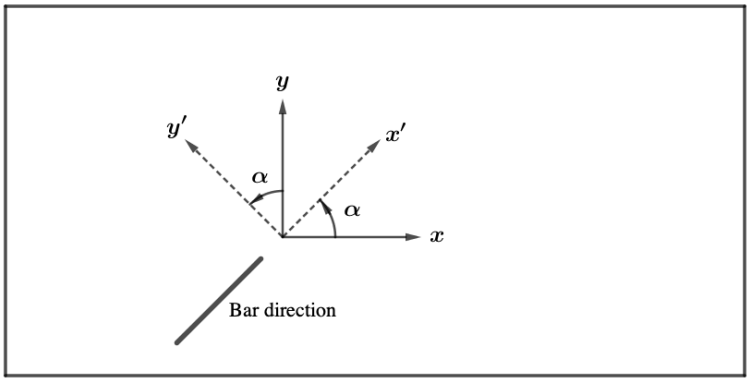


FIGURE 2.20: Global and local axis system representation to orientate the resulting topology.

$$A' = \begin{bmatrix} 100 & 0 \\ 0 & 1 \end{bmatrix} \tag{2.88}$$

Nevertheless, as the algorithm work with the global axis system, we will need the matrix $A_{2D}(\alpha)$ defined in the global system. As it is not trivial to rotate directly a matrix, let v' and n' to represent a local vector and a local normal vector, respectively, combined with A' as shown in Equation (2.89).

$$v' = A' \cdot n' = \begin{bmatrix} v'_x \\ v'_y \end{bmatrix} \tag{2.89}$$

Previous expression applies analogously to their corresponding global measures. Now, in Equation (2.90) the relationship between v and v' is represented in function of a rotation matrix. Note that, as the rotation matrix is orthonormal, hence the inverse of it is directly the same matrix transposed.

$$\begin{gathered} v = \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} v'_x \cdot \cos\alpha - v'_y \cdot \sin\alpha \\ v'_x \cdot \sin\alpha + v'_y \cdot \cos\alpha \end{bmatrix} \\ v = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \cdot \begin{bmatrix} v'_x \\ v'_y \end{bmatrix} = R \cdot v' \\ v' = R^T \cdot v \end{gathered} \tag{2.90}$$

Analogously, one finds the relationship between n and n', see Equation (2.91).

$$n' = R^T \cdot n \tag{2.91}$$

Combining expressions (2.90) and (2.91) with (2.89), we have

$$v = R \cdot A' \cdot R^T \cdot n. \tag{2.92}$$

Finally, the global elastic conductivity matrix is defined in Equation (2.93).

$$A_{2D}(\alpha) = R(\alpha) \cdot A' \cdot R^T(\alpha) \tag{2.93}$$

Chapter 3

Feasibility study

In this chapter the feasibility of Topology Optimization techniques regarding the creation of new structures with less volume and within an optimized shape is verified, along with the fulfillment of additive manufacturing constraints.

Specifically, the considered case studies where the numerical approach is assessed are firstly exposed, as well as an independent anisotropic perimeter validation. Besides, for each case study all variants regarding isotropy, perimeter and design variable are simulated, thus obtaining eight simulations for each part, in general. Lastly, a comparison between the numerical methodology implemented with SIMP and SIMP-ALL is introduced, among other target parameters.

3.1 Case studies definition

3.1.1 Cantilever beam

The classic cantilever beam in 2D is a benchmark case, consisting of a rectangular box initially filled with material (i.e $\chi = 1$ at all the reference domain), clamped at the left side and with a vertical point load applied at the middle of right side, as depicted in Figure 3.1.

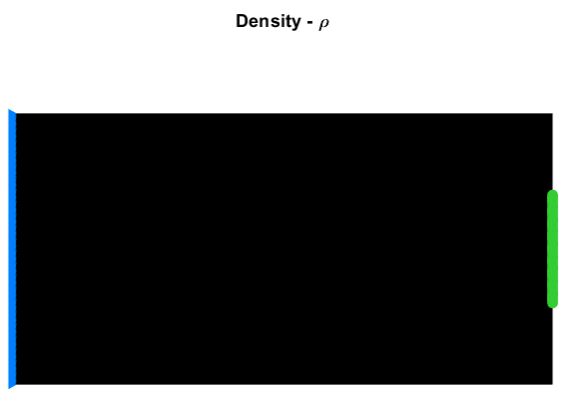


FIGURE 3.1: Cantilever beam computational domain. Initial case representation with density.

The clamped nodes are constrained in all degrees of freedom. Furthermore, the horizontal and vertical length of the box have been chosen as 2 and 1 length units, respectively. Besides, 150 and 75 divisions have been implemented in the horizontal and vertical sides, respectively, in order to obtain quadrilateral elements and later applying triangulation to each one, thus obtaining a mesh with 22052 elements, shown in Figure 3.2. The aim of this case study is to find the optimal topology by minimizing the compliance with an isotropic/anisotropic perimeter penalty, using level set and density as design variables.

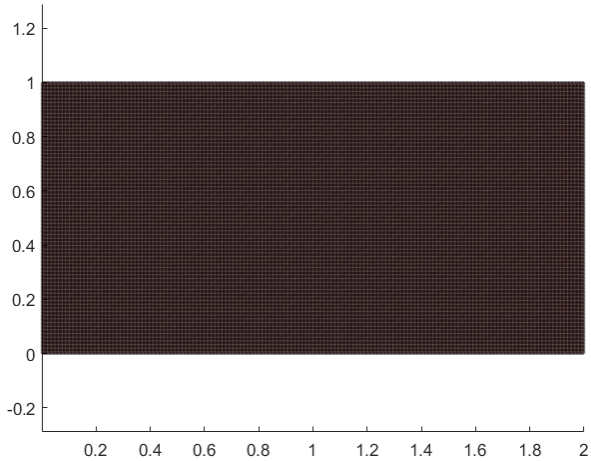


FIGURE 3.2: Cantilever beam mesh (reference domain).

3.1.2 Bridge

The bridge case in 2D consists of a rectangular box initially filled with material, clamped at two set of nodes: one located in the left end of the lower edge and another in the right end of the lower edge. Besides, a vertical point load is applied at the middle of upper edge, as seen in Figure 3.3.

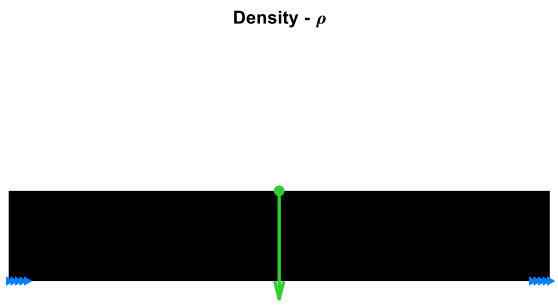


FIGURE 3.3: Bridge computational domain. Initial case representation with density.

The clamped nodes are constrained in all degrees of freedom. Besides, the horizontal and vertical length of the box have been chosen as 6 and 1 length units, respectively. The mesh has been computed with GiD software, with a total number of elements of 137138 (see Figure 3.4). The aim of this case study is to find the optimal topology by minimizing the compliance with an isotropic/anisotropic perimeter penalty, using level set and density as design variables.

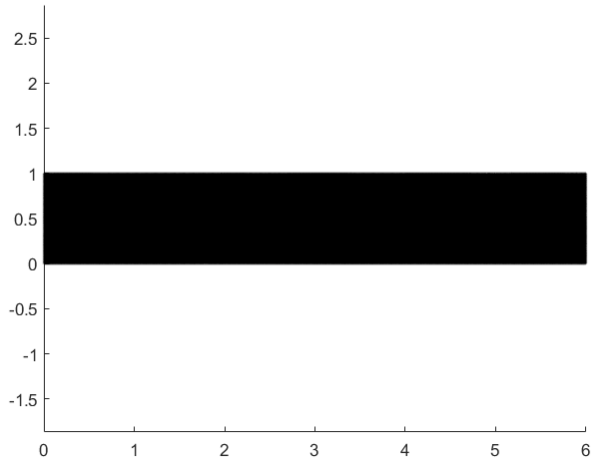


FIGURE 3.4: Bridge mesh (reference domain).

3.1.3 Arch

The arch case in 2D consists of a rectangular box initially filled with material, clamped at two set of nodes: one located near the left end of the lower edge and another near the right end of the lower edge. Besides, a vertical distributed load is applied at the middle of lower edge, as seen in Figure 3.5.

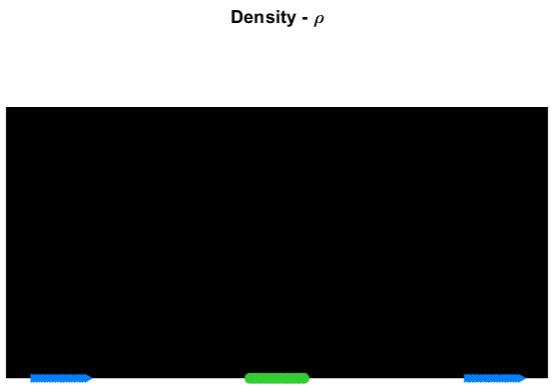


FIGURE 3.5: Arch computational domain. Initial case representation with density.

The clamped nodes are constrained in all degrees of freedom. Besides, the horizontal and vertical length of the box have been chosen as 2 and 1 length units, respectively. The mesh has been computed with GiD software, with a total number of elements of 183570 (see Figure 3.6). The aim of this case study is to find the optimal topology by minimizing the compliance with an isotropic/anisotropic perimeter penalty, using level set and density as design variables.

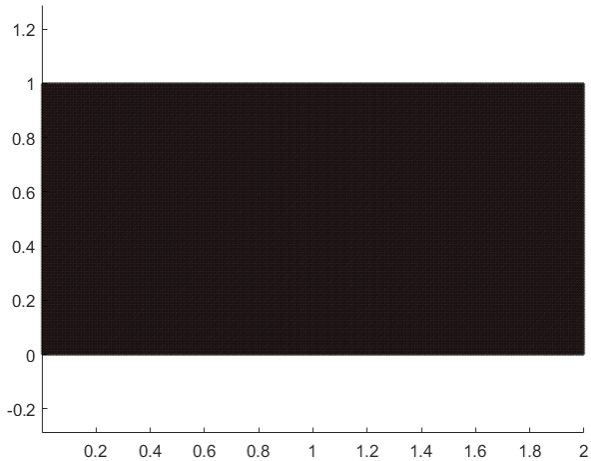


FIGURE 3.6: Arch mesh (reference domain).

3.1.4 Microstructures

In general, microstructures in 2D deal with a square domain with a geometric inclusion inside it as an initial case (for instance, a circle inclusion), representing the shape of a fiber inside a matrix, among similar examples. Conditions at the edges are set to periodic boundary conditions, thus implying that the square domain is indeed an unit cell part from a larger structure, as it occurs in composite materials. In Figure 3.7 the computational domain is shown.

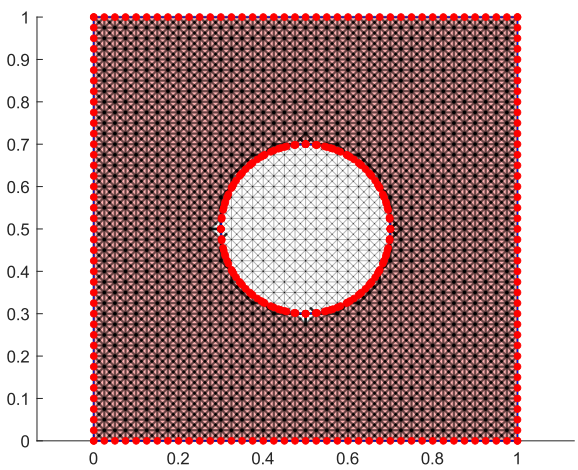


FIGURE 3.7: Microstructure computational domain. Initial case representation with level set.

Furthermore, the vertices of the square are constrained in all degrees of freedom, and no external load is applied on the domain. The mesh employed is composed of 20000 triangular elements, depicted in Figure 3.8.

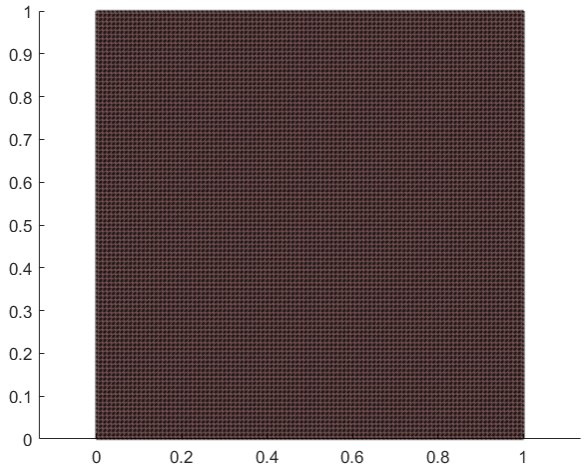


FIGURE 3.8: Micro mesh (reference domain).

The aim of this case study is to find the optimal topology by minimizing some orthotropic effective properties with an isotropic/anisotropic perimeter penalty, using level set and density as design variables. Specifically, in Table 3.1 some benchmark cases regarding different effective properties to optimize are depicted.

Case	Parameters
Bulk (I)	$\alpha_h = [1, 1, 0]$ $\beta_h = [1, 1, 0]$
Bulk (II)	$\alpha_h = [1, 0.5, 0]$ $\beta_h = [1, 0.5, 0]$
Shear	$\alpha_h = [0, 0, 1]$ $\beta_h = [0, 0, 1]$
Shear - Bulk	$\alpha_h = [1, 1, 0.5]$ $\beta_h = [1, 1, 0.5]$

TABLE 3.1: Orthotropic effective properties optimization cases.

Moreover, a last case consisting in the minimum Poisson's ratio problem can be also defined using $\alpha_h = [1, 0, 0]$ and $\beta_h = [0, -1, 0]$. Nevertheless, the corresponding shape functional is not the same as the expression defined in Equation (2.17). Instead, a rational expression is employed for this exceptional case,

$$h(\mathbb{C}) = \frac{\alpha_h^T \mathbb{C}^{-1} \beta_h}{\alpha_h^T \mathbb{C}^{-1} \alpha_h} + \frac{\beta_h^T \mathbb{C}^{-1} \alpha_h}{\beta_h^T \mathbb{C}^{-1} \beta_h} \tag{3.1}$$

3.2 Results

3.2.1 Anisotropic perimeter validation

Prior to solve each optimization case study, an individual validation of the anisotropic perimeter tool is performed, so as to check if the expected behaviour is achieved given a set of inputs regarding both orientation and scaling factor, applied to a pure minimum perimeter problem.

First of all, consider a generic conductivity matrix with a scale factor k_a in Equation (3.2), expressed in its local coordinates system and setting the priority to obtain topologies aligned at local X-axis, with respect to the second main inertia direction.

$$A' = \begin{bmatrix} k_a & 0 \\ 0 & 1 \end{bmatrix} \tag{3.2}$$

Thus, the global conductivity matrix can be obtained by rotating the local matrix with the desired fiber orientation α, as seen in Equation (3.3).

$$A_{2D}(k_a, \alpha) = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \cdot \begin{bmatrix} k_a & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} \tag{3.3}$$

For the pure minimum perimeter problem, by including previous conductivity matrix in the definition of the anisotropic stiffness matrix when solving the diffusion-reaction PDE with the anisotropic filter, hence the anisotropic perimeter tool would be finally defined. Consider then the optimization problem expressed in Equation (3.4).

$$P_{MP} : \begin{Bmatrix} \min_{\chi} \; Per_{\epsilon}^{a}(\chi) \\ s.t \; \int_{\Omega} \chi = 0.75 \\ \chi \in \{0; 1\} \end{Bmatrix}. \tag{3.4}$$

Hence, the same mesh defined as for the microstructure case will be employed (see Figure 3.8), along with a void square inclusion of 25% of the total volume as the initial case, as depicted in Figure 3.9.

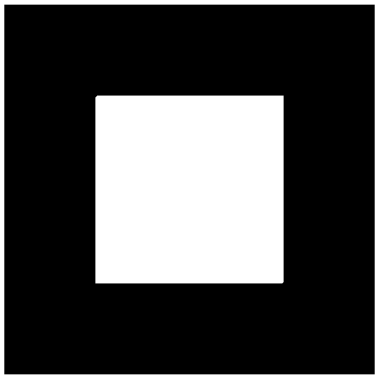


FIGURE 3.9: Initial case of the minimum perimeter problem.

The corresponding results for different orientation angles and scale factors are exposed in Figure 3.10.

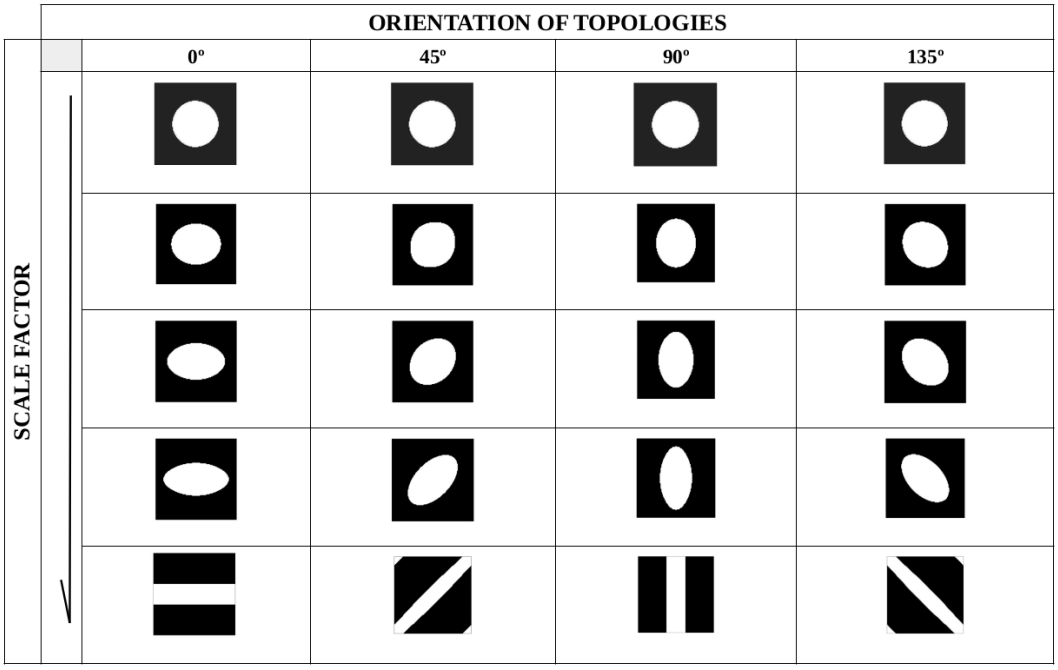


FIGURE 3.10: Anisotropic perimeter optimization for different topologies orientation and scale factor values.

The converged topologies obtained for each case clearly validates the accurate functioning of the anisotropic perimeter. First of all, in first row it is depicted the result for different orientation angles at the minimum scale factor, $k_a = 1$. Equivalently, the standard isotropic perimeter is recovered, and this can be observed as the initial square has been transformed into a perfect circle with the same area as the square, regardless the imposed orientation angle. Indeed, the circle is the shape that minimizes the perimeter of a certain volume of material, at isotropic conditions.

Moreover, for any orientation angle, the circle turns into an ellipse with higher eccentricity as the scale factor increases, until converging into a perfect aligned fiber when $k_a \to \infty$. These results also validate the anisotropic perimeter performance, since now the square is transformed into an ellipse, which optimizes the perimeter of the given volume of material, but also taking advantage of the material virtual stiffness matrix.

Lastly, as the orientation angle of topologies changes, the major axis orientation of the obtained ellipse also does and it is completely aligned with the local X-axis of the conductivity matrix A', which forms the same angle with the global X-axis as the angle defined by the user regarding topologies orientation. Therefore, a control to impose overhang constraints has been clearly established.

All case studies presented in next sections will employ $k_a = 100$ and $\alpha = 90°$, so as to fulfill overhang control in the vertical direction.

3.2.2 Cantilever beam

FIGURE 3.11: Isotropic total perimeter with density.

FIGURE 3.12: Anisotropic total perimeter with density.

FIGURE 3.13: Isotropic relative perimeter with density.

FIGURE 3.14: Anisotropic relative perimeter with density.

FIGURE 3.15: Isotropic total perimeter with level set.

FIGURE 3.16: Anisotropic total perimeter with level set.

FIGURE 3.17: Isotropic relative perimeter with level set.

FIGURE 3.18: Anisotropic relative perimeter with level set.

Case:	**Stage 1**						
Cantilever	**optUnconstr**	**optConstr**	**filter**	**nSteps**	**maxIter**	**Vi**	**Vf**
Pure Compliance (density)	Projected Gradient	Dual Nested In Primal	P1	1	2000	1	0.5
Pure Compliance (level set)	SLERP	Dual Nested In Primal	P1	1	2000	1	0.5

TABLE 3.2: Settings of different cantilever beam cases (stage 1).

Case:	**Stage 2**							
Cantilever	**Cost**	**optUnconstr**	**optConstr**	**filter**	**nSteps**	**maxIter**	**Vi**	**Vf**
Isotropic total perimeter (density)	c + P	-	MMA	P1	10	200	0.5	0.5
Anisotropic total perimeter (density)	c + 0.02P	-	MMA	P1	10	200	0.5	0.5
Isotropic relative perimeter (density)	c + P	-	MMA	P1	10	200	0.5	0.5
Anisotropic relative perimeter (density)	c + 0.02P	-	MMA	P1	10	200	0.5	0.5
Isotropic total perimeter (level set)	c + P	SLERP	Dual Nested In Primal	P1	10	200	0.5	0.5
Anisotropic total perimeter (level set)	c + 0.01P	SLERP	Dual Nested In Primal	P1	10	100	0.5	0.5
Isotropic relative perimeter (level set)	c + P	SLERP	Dual Nested In Primal	P1	7	140	0.5	0.5
Anisotropic relative perimeter (level set)	c + 0.01P	SLERP	Dual Nested In Primal	P1	10	100	0.5	0.5

TABLE 3.3: Settings of different cantilever beam cases (stage 2).

The results of all variants regarding the cantilever beam case study are shown above, along with the settings configuration defined for each case.

As the cantilever beam is mainly submitted to a bending load, it is relatively complex to obtain a converged solution directly with a single stage simulation using both compliance and a vertical anisotropic perimeter penalty inside the cost function. Therefore, a 2-stage simulation has been performed for each of the 8 variants (also including the isotropic cases, for convenience), as their settings are depicted in Tables 3.2 and 3.3. In stage 1, the cantilever beam is optimized with a minimum compliance problem algorithm, without keeping control in overhang and length scales. As the code deals with two design variables, hence just two different simulations for the pure compliance shall be carried out: density and level set. Thus, the final converged result from the minimum compliance problem is numerically stored and this serves as the initial case of stage 2, where the cantilever beam is postprocessed using now both the compliance and a perimeter penalty inside the cost function, so as to filter small length scales and horizontal topologies, keeping the volume constant throughout this process. Notice that the factor multiplying the perimeter penalty term might be different for each of the 8 variants in order to obtain a well-converged result. Lastly, all variants are solved with SIMP-ALL as the interpolation method.

In Figures 3.11 and 3.12 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the density as design variable, respectively. Regarding the isotropic case, no local features with small length scales are appearing in the solution. In fact, just one reinforcing bar is acting in the bottom side of the box mesh. Besides, the relationship between the degree of contribution of compliance and perimeter shape functionals is 1:1, which is relatively high. Therefore, regions near nodes with imposed displacement or force converge with curved boundaries, as it is the best way to optimize locally the perimeter. In relation with the anisotropic case, two bars with a vertical tendency orientation have appeared at the top side of the box mesh, reinforcing the beam with optimal resources since a virtual anisotropic stiffness matrix has been imposed with a priority in the global Y-axis. Furthermore, previous curved boundaries have converged into vertical ones, fitting again with the box mesh boundary. Finally, it is surprising how for the density case the converged results are not symmetric with respect the beam's axis, probably due to the non-symmetry of the mesh.

In Figures 3.13 and 3.14 the results related with isotropic relative perimeter and anisotropic relative perimeter are shown for the density as design variable, respectively. Results are very similar to their total perimeter complementary cases, with two main differences. Firstly, as now only the interior perimeter accounts to the penalization term of the cost function, the boundaries of the beam adjacent to the boundary of box mesh obtained from the pure compliance case always remain in such position, hence obtaining now abrupt variations at interfaces between the boundaries of the inner mesh and the inner cut mesh. Secondly, and as a consequence of the first difference, local reinforcing bars appearing in both variants have been penalized more severely, obtaining minor length scales.

In Figures 3.15 and 3.16 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the level set as design variable, respectively. These cases converge into a solution symmetric with respect to the beam's axis. Regarding the isotropic case, again no local features with small length scales appear. Nevertheless, in relation with the anisotropic case, previous reinforcing bars have converged into ones slightly aligned vertically, and also with a higher thickness, thus satisfying the anisotropic perimeter penalty term.

Finally, in Figures 3.17 and 3.18 the results related with isotropic relative perimeter

and anisotropic relative perimeter are shown for the level set as design variable, respectively. Again, results are very similar to their total perimeter complementary case, but with abrupt variations at the inner mesh and inner cut mesh interfaces. Nevertheless, now the penalization of interior perimeter is so notorious that even an internal reinforcing bar has been removed for the isotropic case.

3.2.3 Bridge

FIGURE 3.19: Isotropic total perimeter with density.

FIGURE 3.20: Anisotropic total perimeter with density.

FIGURE 3.21: Isotropic relative perimeter with density.

FIGURE 3.22: Anisotropic relative perimeter with density.

FIGURE 3.23: Isotropic total perimeter with level set.

FIGURE 3.24: Anisotropic total perimeter with level set.

FIGURE 3.25: Isotropic relative perimeter with level set.

FIGURE 3.26: Anisotropic relative perimeter with level set.

Case:	Stage 1						
Bridge	**optUnconstr**	**optConstr**	**filter**	**nSteps**	**maxIter**	**Vi**	**Vf**
Pure Compliance (density)	Projected Gradient	Dual Nested In Primal	P1	1	2000	1	0.5
Pure Compliance (level set)	SLERP	Dual Nested In Primal	P1	10	2000	1	0.5

TABLE 3.4: Settings of different bridge cases (stage 1).

Case:	Stage 2							
Bridge	**Cost**	**optUnconstr**	**optConstr**	**filter**	**nSteps**	**maxIter**	**Vi**	**Vf**
Isotropic total perimeter (density)	c + 0.1P	-	MMA	P1	10	100	0.5	0.5
Anisotropic total perimeter (density)	c + 0.1P	-	MMA	P1	10	100	0.5	0.5
Isotropic relative perimeter (density)	c + 0.1P	-	MMA	P1	10	100	0.5	0.5
Anisotropic relative perimeter (density)	c + 0.1P	-	MMA	P1	10	100	0.5	0.5
Isotropic total perimeter (level set)	c + 0.3P	SLERP	Dual Nested In Primal	P1	10	100	0.5	0.5
Anisotropic total perimeter (level set)	c + 0.3P	SLERP	Dual Nested In Primal	P1	10	100	0.5	0.5
Isotropic relative perimeter (level set)	c + 0.3P	SLERP	Dual Nested In Primal	P1	10	100	0.5	0.5
Anisotropic relative perimeter (level set)	c + 0.3P	SLERP	Dual Nested In Primal	P1	10	100	0.5	0.5

TABLE 3.5: Settings of different bridge cases (stage 2).

The results of all variants regarding the bridge case study are shown above, along with the settings configuration defined for each case.

Again, the same 2-stage simulation strategy as the cantilever beam case has been performed for each of the 8 variants, as their settings are depicted in Tables 3.4 and 3.5. In stage 1, the bridge is optimized with a minimum compliance problem algorithm. Similarly to the cantilever beam, just two different simulations for the pure compliance shall be carried out depending on the design variable: density and level set. Thus, the final converged result from the minimum compliance problem is numerically stored and this serves as the initial case of stage 2, where the bridge is postprocessed using now both the compliance and a perimeter penalty inside the cost function. Notice that the factor multiplying the perimeter penalty term might be different for each of the 8 variants in order to obtain a well-converged result. Lastly, all variants are solved with SIMP-ALL as the interpolation method.

In Figures 3.19 and 3.20 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the density as design variable, respectively. Regarding the isotropic case, just two reinforcing bars with high thickness are acting in the middle of the structure, avoiding the formation of small features. In relation with the anisotropic case, these cross bars have disappeared, inducing the formation of two vertical pillars, as it is the best material distribution due to the imposed virtual anisotropic stiffness matrix, with a priority in the global Y-axis. Furthermore, Dirichlet nodes have converged also into vertical topologies that are linked with the main pillars. Finally, note that both cases are symmetric with respect the boundary conditions set-up.

In Figures 3.21 and 3.22 the results related with isotropic relative perimeter and anisotropic relative perimeter are shown for the density as design variable, respectively. Results are very similar to their total perimeter complementary cases. However, note again that as now only the interior perimeter accounts to the penalization term of the cost function, abrupt variations are obtained at interfaces between the boundaries of the inner mesh and the inner cut mesh. Also, penalization is acting more severely in the anisotropic case, where the bridge is even split in two structures symmetrically.

In Figures 3.23 and 3.24 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the level set as design variable, respectively. These cases converge into a solution symmetric with respect to the bridge's vertical axis. Regarding the isotropic case, again no local features with small length scales appear, obtaining a very similar solution as with density, but including reinforcing bars with less thickness. Nevertheless, in relation with the anisotropic case, previous reinforcing bars have converged into ones slightly aligned vertically with higher thickness, instead of the formation of two pillars, thus satisfying the anisotropic perimeter penalty term. Note also that in the middle part of the bottom edge of the box mesh a penalization of a horizontal bar has been produced, splitting this region of the bridge's base in two parts.

Finally, in Figures 3.25 and 3.26 the results related with isotropic relative perimeter and anisotropic relative perimeter are shown for the level set as design variable, respectively. As usually, results are very similar to their total perimeter complementary case, but with abrupt variations at the inner mesh and inner cut mesh interfaces.

3.2.4 Arch

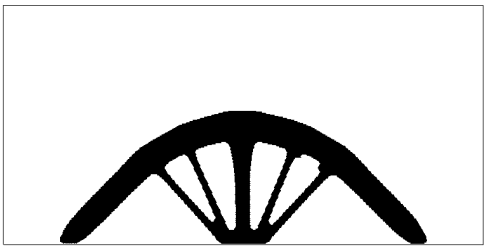


FIGURE 3.27: Isotropic total perimeter with density.

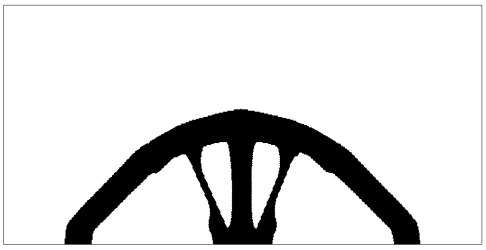


FIGURE 3.28: Anisotropic total perimeter with density.

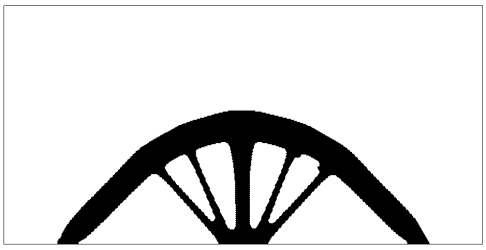


FIGURE 3.29: Isotropic relative perimeter with density.

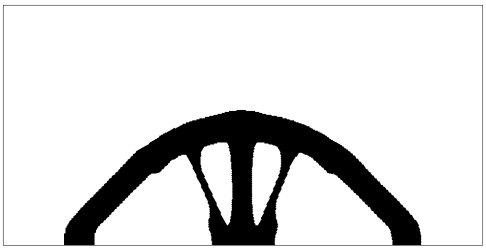


FIGURE 3.30: Anisotropic relative perimeter with density.

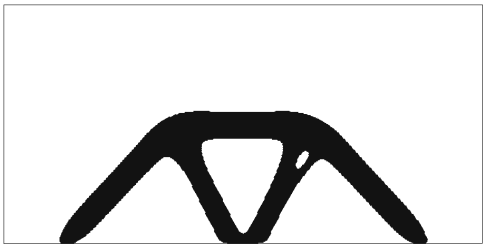


FIGURE 3.31: Isotropic total perimeter with level set.

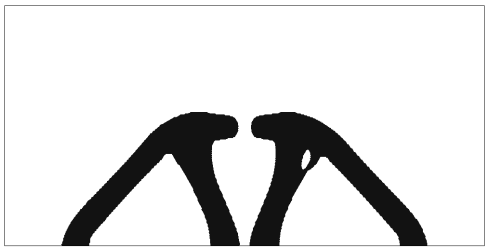


FIGURE 3.32: Anisotropic total perimeter with level set.

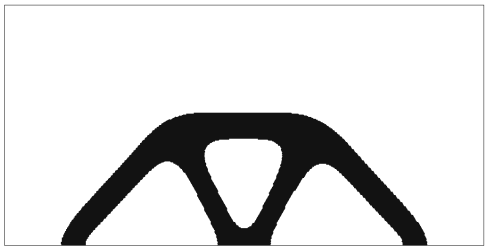


FIGURE 3.33: Isotropic relative perimeter with level set.

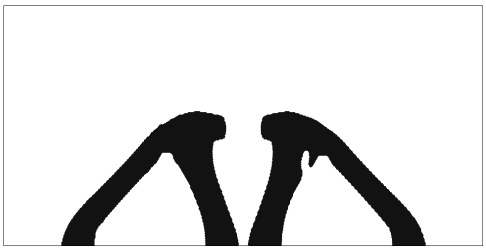


FIGURE 3.34: Anisotropic relative perimeter with level set.

Case:	**Stage 1**						
Arch	**optUnconstr**	**optConstr**	**filter**	**nSteps**	**maxIter**	**Vi**	**Vf**
Pure Compliance (density)	Projected Gradient	Dual Nested In Primal	P1	1	2000	1	0.15
Pure Compliance (level set)	SLERP	Dual Nested In Primal	P1	4	2000	1	0.15

TABLE 3.6: Settings of different arch cases (stage 1).

Case:	**Stage 2**							
Arch	**Cost**	**optUnconstr**	**optConstr**	**filter**	**nSteps**	**maxIter**	**Vi**	**Vf**
Isotropic total perimeter (density)	c + 0.01P	-	MMA	P1	10	200	0.15	0.15
Anisotropic total perimeter (density)	c + 0.01P	-	MMA	P1	10	200	0.15	0.15
Isotropic relative perimeter (density)	c + 0.01P	-	MMA	P1	10	200	0.15	0.15
Anisotropic relative perimeter (density)	c + 0.01P	-	MMA	P1	10	200	0.15	0.15
Isotropic total perimeter (level set)	c + P	SLERP	Dual Nested In Primal	P1	10	100	0.15	0.15
Anisotropic total perimeter (level set)	c + P	SLERP	Dual Nested In Primal	P1	10	100	0.15	0.15
Isotropic relative perimeter (level set)	c + P	SLERP	Dual Nested In Primal	P1	10	100	0.15	0.15
Anisotropic relative perimeter (level set)	c + P	SLERP	Dual Nested In Primal	P1	10	100	0.15	0.15

TABLE 3.7: Settings of different arch cases (stage 2).

The results of all variants regarding the arch case study are shown above, along with the settings configuration defined for each case.

The same 2-stage simulation strategy as previous cases has been performed for each of the 8 variants, as their settings are depicted in Tables 3.6 and 3.7. In stage 1, the arch is optimized with a minimum compliance problem algorithm. Just two different simulations for the pure compliance shall be carried out depending on the design variable: density and level set. Thus, the final converged result from the minimum compliance problem is numerically stored and this serves as the initial case of stage 2, using now both the compliance and a perimeter penalty inside the cost function. Notice that the factor multiplying the perimeter penalty term might be different for each of the 8 variants in order to obtain a well-converged result. Lastly, all variants are solved with SIMP-ALL as the interpolation method.

In Figures 3.27 and 3.28 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the density as design variable, respectively. Regarding the isotropic case, one vertical reinforcing bar has appeared at the middle, along with two additional crossed bars at each side, symmetrically. In relation with the anisotropic case, the cross bars with highest horizontal orientation have disappeared, inducing the increase in thickness of the vertical bar, as it is the best material distribution due to the imposed virtual anisotropic stiffness matrix. Lastly, the anisotropic case is also symmetric with respect the vertical axis.

In Figures 3.29 and 3.30 the results related with isotropic relative perimeter and anisotropic relative perimeter are shown for the density as design variable, respectively. Results are very similar to their total perimeter complementary cases. Nevertheless, abrupt variations are obtained at interfaces between the boundaries of the inner mesh and the inner cut mesh because only the interior perimeter accounts to the penalization term of the cost function, hence avoiding curve boundaries near Neumann nodes.

In Figures 3.31 and 3.32 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the level set as design variable, respectively. These cases also converge into a solution symmetric with respect to the arch's vertical axis, except to the formation of a very small feature, possibly meaning that a higher penalization coefficient is needed. Regarding the isotropic case, no local features with small length scales are appearing, obtaining two crossed reinforcing bars with high thickness. However, in relation with the anisotropic case, previous reinforcing bars have converged into ones slightly aligned vertically in the lower side, but splitting the whole arch in two parts, thus satisfying the virtual stiffness in Y-axis. Indeed, the arch is a case study highly sensitive to a vertical anisotropic perimeter constraint.

Finally, in Figures 3.33 and 3.34 the results related with isotropic relative perimeter and anisotropic relative perimeter are shown for the level set as design variable, respectively. Again, results are very similar to their total perimeter complementary case, but with abrupt variations at the inner mesh and inner cut mesh interfaces. Also, previous small hole now is fulfilled in the isotropic case, probably due to now only the interior perimeter accounts to the penalty term.

3.2.5 Microstructures

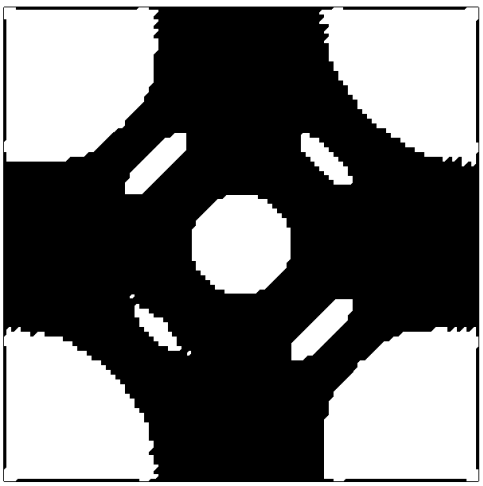


FIGURE 3.35: Isotropic total perimeter with density (bulk I).

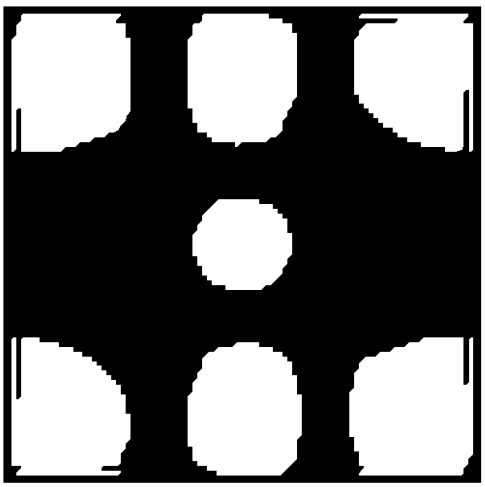


FIGURE 3.36: Anisotropic total perimeter with density (bulk I).

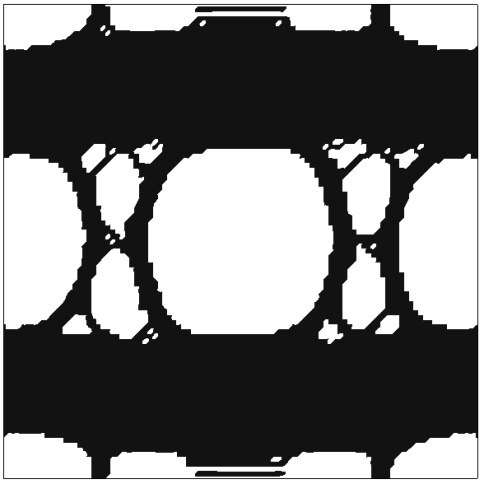


FIGURE 3.37: Isotropic total perimeter with level set (bulk II).

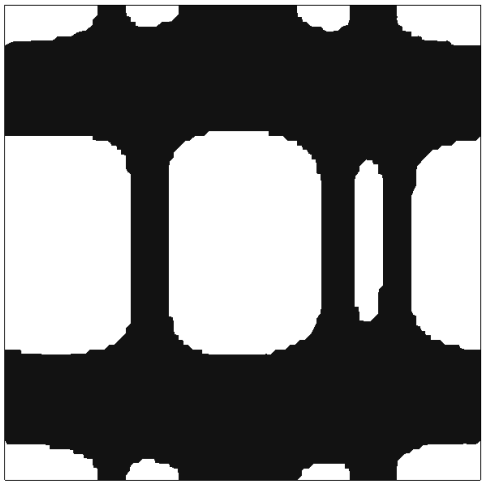


FIGURE 3.38: Anisotropic total perimeter with level set (bulk II).

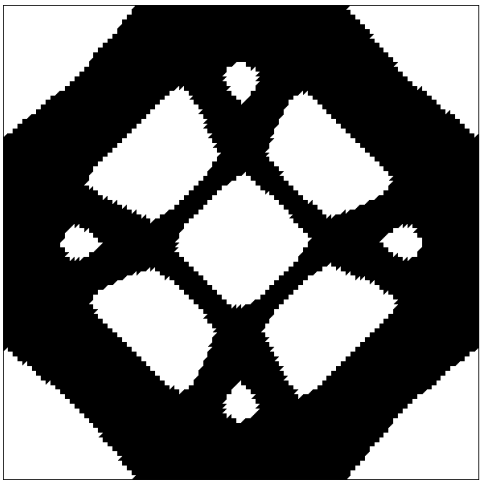


FIGURE 3.39: Isotropic total perimeter with density (shear).

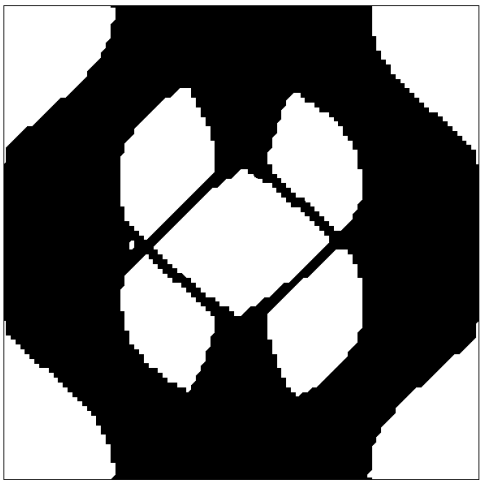


FIGURE 3.40: Anisotropic total perimeter with density (shear).

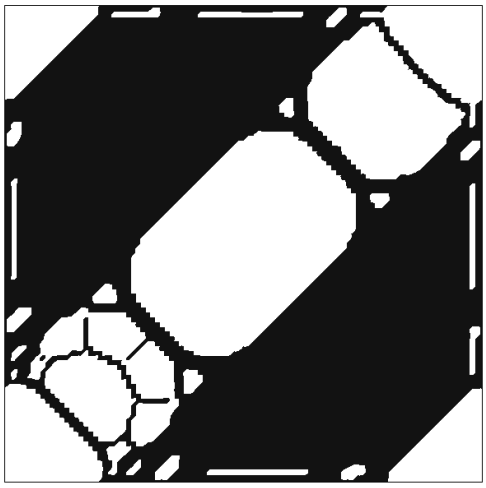


FIGURE 3.41: Isotropic total perimeter with level set (shear-bulk).

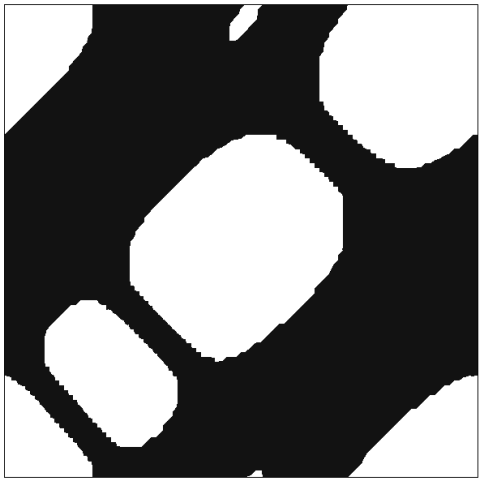


FIGURE 3.42: Anisotropic total perimeter with level set (shear-bulk).

Case:	**Single stage**							
Micro	**Cost**	**optUnconstr**	**optConstr**	**filter**	**nSteps**	**maxIter**	**Vi**	**Vf**
Isotropic total perimeter in Bulk I (density)	h + 0.052P	Projected Gradient	Dual Nested In Primal	P1	3	2000	1	0.6
Anisotropic total perimeter in Bulk I (density)	h + 0.052P	Projected Gradient	Dual Nested In Primal	P1	3	2000	1	0.6
Isotropic total perimeter in Shear (density)	h + 0.052P	Projected Gradient	Dual Nested In Primal	P1	3	2000	1	0.6
Anisotropic total perimeter in Shear (density)	h + 0.052P	Projected Gradient	Dual Nested In Primal	P1	3	2000	1	0.6
Isotropic total perimeter in Bulk II (level set)	h + 0.1P	SLERP	Dual Nested In Primal	P1	8	2000	1	0.6
Anisotropic total perimeter in Bulk II (level set)*	h + 0.1P	SLERP	Dual Nested In Primal	P1	8/ 10	2000/ 150	1/ 0.6	0.6/ 0.6
Isotropic total perimeter in Shear-Bulk (level set)	h + 0.1P	SLERP	Dual Nested In Primal	P1	8	2000	1	0.6
Anisotropic total perimeter in Shear-Bulk (level set)*	h + 0.1P	SLERP	Dual Nested In Primal	P1	8/ 10	2000/ 150	1/ 0.6	0.6/ 0.6

TABLE 3.8: Settings of different micro cases. Anisotropic cases labeled with * consist of two stages which values are separated by s_1/s_2. First stage is the same simulation as their complementary isotropic cases. Second stage is a postprocess of the result obtained from first stage, considering the anisotropic stiffness matrix.

The results of all variants regarding the microstructures case study are shown above, along with the settings configuration defined for each case.

In this case, just a single stage simulation strategy has been followed, which related cost functions are depicted in the *Cost* column of Table 3.8, along with other settings. This is because microstructures might allow the formation of very small features that implies the use of a larger number of iterations to obtain a converged result. Therefore, a penalty is crucial in these cases. Thus, the unit cell is optimized with a minimum compliance problem algorithm, combined also with a perimeter penalty term. Notice that the factor multiplying the perimeter penalty term might be different for each of the 8 variants in order to obtain a well-converged result. Nevertheless, there are two anisotropic cases (anisotropic total perimeter in Bulk II with level set, and anisotropic total perimeter in Shear-Bulk with level set) which use a 2-stage simulation strategy, the first one considering their related isotropic equivalent case in the cost function, and hence the second case considering the anisotropic perimeter, where the final converged result from first stage is numerically stored and this serves as the initial case of stage 2. Lastly, all variants are solved with SIMP-ALL as the interpolation method. Note that all obtained results fulfill periodic boundary conditions, as a microstructure unit cell.

In Figures 3.35 and 3.36 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the density as design variable and the Bulk I orthotropic case, respectively. Regarding the isotropic case, no local features with small length scales are appearing in the solution. The solution yields a black value distribution such that both main directions (horizontal and vertical) are optimized in material. In relation with the anisotropic case, the two previous vertical topologies have been split into two parts, as a virtual anisotropic stiffness matrix has been imposed with a priority in the global Y-axis, hence implying the material capability to withstand loads in the vertical direction with less cross section area, and satisfying the Bulk I optimization case simultaneously.

In Figures 3.37 and 3.38 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the level set as design variable and the Bulk II orthotropic case, respectively. Regarding the isotropic case, some local features with small length scales are appearing in the solution. Besides, the solution now yields a black value distribution such that the horizontal direction has more priority over the vertical one in terms of material distribution. In relation with the anisotropic case, the small features have dissapeared and the rest of reinforcing bars have converged into vertical topologies.

In Figures 3.39 and 3.40 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the density as design variable and the Shear orthotropic case, respectively. Regarding the isotropic case, no local features with small length scales are appearing in the solution. Besides, the solution yields a black value distribution such that the material is able to withstand pure torques, with double symmetry. In relation with the anisotropic case, horizontal features are severely penalized, whereas vertical ones have increased in its thickness.

In Figures 3.41 and 3.42 the results related with isotropic total perimeter and anisotropic total perimeter are shown for the level set as design variable and the Shear-Bulk orthotropic case, respectively. Regarding the isotropic case, some local features with small length scales are appearing in the solution. The solution yields a black value distribution such that the material is optimized in both main directions, and also with the capability to withstand torsional loads. In relation with the anisotropic case, small features have been penalized, whereas the other ones have increased in its thickness. It is noticeable the fact that no vertical orientation is produced of main

topologies, meaning that these converged orientations are needed in order to fulfill the shear contribution requirement.

3.2.6 Numerical methodologies comparison

SIMP vs SIMP-ALL

Finally, in this section a comparison between different numerical methodologies regarding interpolation methods and the target parameter ϵ is presented. First, the SIMP and SIMP-ALL interpolation methods are compared.

Consider the cantilever beam case study with the following settings configuration:

- Cost function, $J = c + 10^{-4}P$ (isotropic).
- Design variable: density.
- Optimizers: Projected Gradient (unconstrained) + Dual Nested In Primal (constrained).
- Filter: P1.
- 4 steps with 2000 maximum total iterations.
- $V_f = 0.5$.

The solution with a single stage simulation for three different methods - SIMP with an exponent $p = 2$, SIMP with an exponent $p = 3$ and SIMP-ALL- are presented in the following figures.

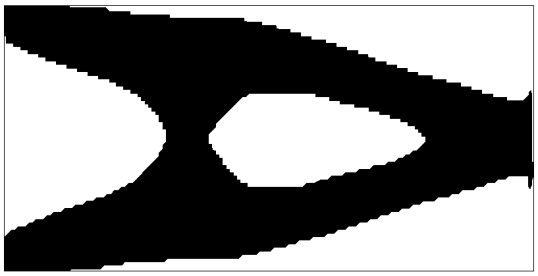


FIGURE 3.43: Converged solution of the cantilever beam example with the interpolation SIMP method ($p = 2$).

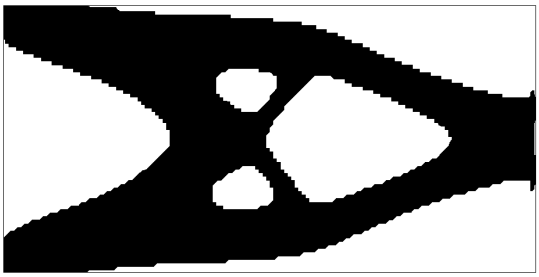


FIGURE 3.44: Converged solution of the cantilever beam example with the interpolation SIMP method ($p = 3$).

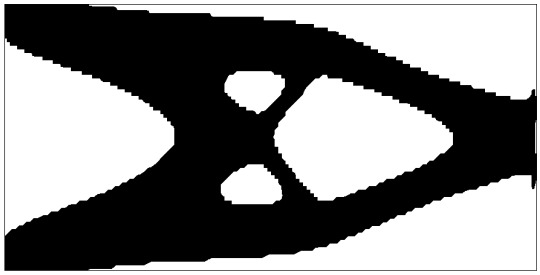


FIGURE 3.45: Converged solution of the cantilever beam example with the interpolation SIMP-ALL method.

In previous results one checks the strong dependence of the exponential parameter of the SIMP method, recalling that such parameter accounted for the degree of gray regions penalization. Therefore, the solution related with $p = 2$ shows a cantilever beam with any cross reinforcing bars, since with this exponential order gray regions are not penalized with the same importance as the SIMP-ALL method, hence implying that the later regularization induced in a larger region of black values. Moreover, the solution with $p = 3$ yields a design variable distribution very similar to the one obtained with the most generic SIMP-ALL method, since with a cubic expression gray regions are severely penalized. Besides, it is justified the reason of selecting $p = 3$ in SIMP method as a heuristic model for optimization problems.

High ϵ vs low ϵ

Lastly, in second place the target parameter ϵ influence in optimization problems is assessed, due to its crucial importance.

Recall the definition of the perimeter shape functional in Equation (3.5).

$$P = \frac{1}{2\epsilon} \int_{\Omega} (1 - \rho_{\epsilon})\rho \cdot d\Omega \tag{3.5}$$

One shall check that ϵ dependence accounts for terms $\frac{1}{2\epsilon}$ and ρ_{ϵ}. More specifically, the smaller the ϵ, the poorer the shape and topological derivatives due to the regularized density. Nevertheless, the smaller the ϵ the greater the contribution of perimeter inside the optimization problem.

When dealing with a cost function where both compliance and perimeter are participating, it is desirable to impose a ϵ enough small to ensure enough penalty of perimeter constraints. Thus, one option is to set ϵ as the minimum mesh size Δh (employed throughout all previously analyzed case studies). Nevertheless, if this target parameter is even smaller, no changes in the topology may be observed as the topological derivative would have lost its sense, although the penalty term was noticeable. Hence, some agreement must be established such that the selected ϵ is enough small to ensure the appearance of perimeter as a penalty term, and enough large to avoid the meaningless of the topological derivative.

Therefore, a small case study is proposed below, considering the pure minimum perimeter problem applied to the microstructure with a void square inclusion of 30% of volume, similar to the one that has been previously studied in Section 3.2.1.

The initial case is submitted to the minimum perimeter problem in two different cases: one with a single step and $\epsilon = 2\Delta h$; another with a single step and $\epsilon = 20\Delta h$. Final results are depicted below.

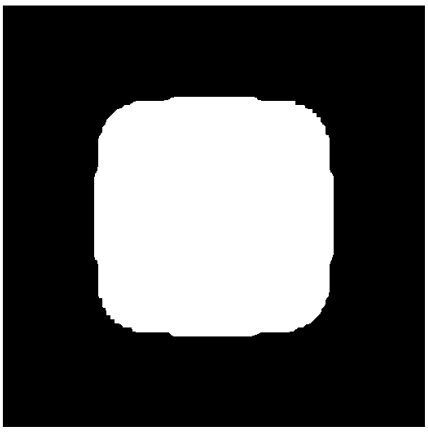


FIGURE 3.46: Converged solution of the microstructure example with $\epsilon = 2\Delta h$.

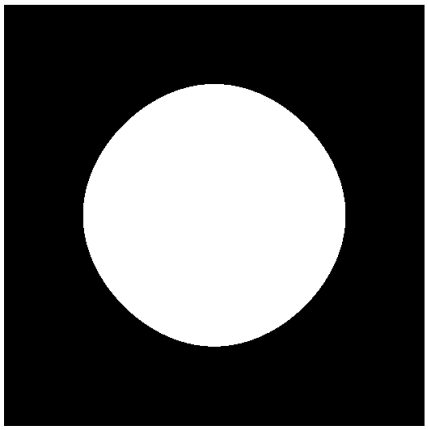


FIGURE 3.47: Converged solution of the microstructure example with $\epsilon = 20\Delta h$.

Note how the solution which uses the small value of ϵ has converged into a square with curved boundary, rather than the circle, which is clearly the shape that minimizes the interior perimeter given a certain quantity of material. This is because the topological and shape derivatives are so poor that the steepest descent algorithm finds a wrong local minimum before transforming the domain into a circle. Nevertheless, with the case of larger ϵ hence the solution converges into a perfect circle. Although $\epsilon = 20\Delta h$ is relatively high, regarding this particular case this does not matter, since the perimeter is not acting as a penalty term to the compliance. Large ϵ considers larger regularization and thus the problem becomes more *global*, helping the optimizer on finding better results. However, for real optimization problems ϵ must be selected carefully, as well as its target parameter evolution over computational steps.

Chapter 4

Project impact

4.1 Environmental impact

The intangibility of this project directly implies the definition of an environmental impact to be relatively complex, since the exposed development in previous chapters has been based in a pure theoretical field, showing the obtained results via numerical simulations. Nevertheless, some aspects regarding the complementary part of the design, Additive Manufacturing, can be mentioned.

Due to the rapid prototyping achieved by means of Additive Manufacturing techniques and following the growing interest in environmentally benign manufacturing, researchers compared their performance with more conventional routes such as machining or injection molding processes. Most studies focus on energy consumption, which for Additive Manufacturing unit processes this energy is 1 to 2 orders of magnitude higher compared to conventional routes. Therefore, from environmental perspective, this consumption shall be compensated by functional improvements during the use stage of manufactured parts. However, when assessing potential fuel consumption reduction coefficients, the application of lightweighting components in aircrafts only implies environmental benefits if significant weight reduction can be reached. [22]

4.2 Economic impact

Indeed, this project contributes positively to the economic impact of all industries that shall obtain benefits from topology optimization techniques, combined with Additive Manufacturing. More specifically, the heuristic designs obtained, consisting of complex shapes and unintuitive holes, can only be created by means of Additive Manufacturing with a reasonable cost. Furthermore, although such processes are usually applied only to thermoset materials, researchers are currently investigating new machinery to allow the creation of these designs with new kind of materials, such as metals.

Moreover, since topology optimization reduces the weight of an initial design, hence implying a reduction of the quantity present inside its bill of materials, the structural cost also is reduced, allowing companies to offer products with the same capabilities and with a minimum production cost.

Finally, related to the budget of this project, this has been relatively low, as seen in Section 5.1. Specifically, the total budget of this project is **11412.00 €**.

4.3 Safety considerations

Regarding safety considerations, few aspects can be taken into account, more related to future lines of investigation.

The resulting manufacturing of the optimized parts must be tested with a real experiment, under the same load distribution and constraints as the numerical experiment, in order to assure the capability of the design and material to withstand the related stress distribution.

Lastly, recall the importance to fulfill Additive Manufacturing constraints. Therefore, it is mandatory to check the sensitivity of the machine previous to perform the numerical optimization, so as to ensure manufacturability by controlling the minimum length scale. Besides, the part shall be printed in the same orientation as how overhang constraints where assessed numerically.

Chapter 5

Budget and project schedule

5.1 Budget

The budget of this project, which gives dimension to its initial inversion, has included the following kind of costs: Human Resources, Software and other individual fixed costs.

More specifically, in Tables 5.1, 5.2 and 5.3, the Human Resources Plan, Software and fixed costs are depicted, respectively.

Finally, in Table 5.4 the total budget of the project is computed.

Human Resources Plan

ID	Components	Hours	Cost/u (€)	Cost (€)
1.1	**Project management.** Costs relative to project development time.	300.0	20.00	6000.00
			Total	**6000.00**

TABLE 5.1: Cost of project Human Resources. Own elaboration.

Software

ID	Components	Cost (€)
2.1	**Permanent licenses.**	3100.00
2.1.1	*Matlab & Simulink.*	*2000.00*
2.1.2	*GiD Simulation.*	*1100.00*
	Total	**3100.00**

TABLE 5.2: Cost of Softwares. Own elaboration.

Fixed costs

ID	Components	Hours	Cost/u (€)	Cost (€)
3.1	**Material.**	-	-	2000.00
3.1.1	*Computer*	-	-	1500.00
3.1.2	*Office material*	-	-	500.00
3.2	**Additional costs.**	-	-	312.00
3.2.1	*Light*	300.0	0.17	51.00
3.2.2	*Telephony*	300.0	0.38	114.00
3.2.3	*Commuting*	300.0	0.49	147.00
			Total	**2312.00**

TABLE 5.3: Individual fixed costs. Own elaboration.

Total budget

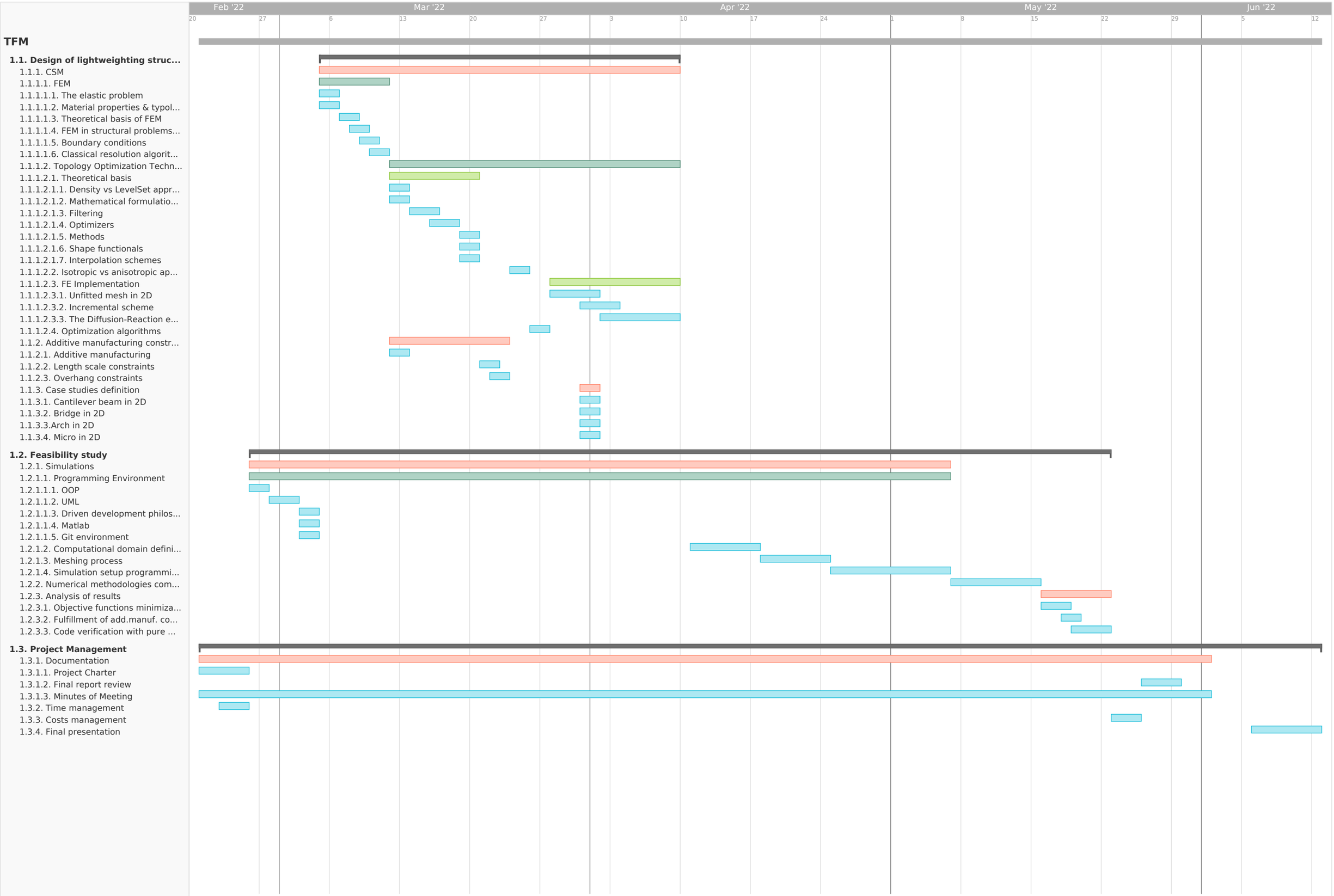
ID	Components	Cost (€)
1	**Human Resources Plan.**	6000.00
2	**Software.**	3100.00
3	**Fixed costs.**	2312.00
	Total	**11412.00**

TABLE 5.4: Total budget cost. Own elaboration.

Thus, a total budget of **11412.00 €** is obtained.

5.2 Schedule

Moreover, in next page the Gantt Chart followed throughout this project is attached.



Chapter 6

Conclusions

6.1 Project conclusions

The present project has accomplished the main objectives planned in the beginning. The feasibility of designs under additive manufacturing constraints - and considering numerical topology optimization techniques - has been properly analyzed, so as to validate the anisotropic perimeter implementation and giving a positive answer to the creation of lightweighting structures and the related fulfillment of 3D printing constraints. More specifically, the methodology followed and the results obtained have led to the key points depicted hereunder.

The programming environment defined for this project has been useful throughout all stages, using the test-driven development philosophy, along with Object-Oriented Programming techniques to obtain higher reusability dynamism of all implemented objects, unlike traditional successor functions, and thus overall optimizing code development time. Besides, Git Version Control System has been helpful to manage multiple versions of the Swan project, hence combining the code developed with further improvements of the generic code repository.

Moreover, the use of Finite Element Methods in structural problems allows the creation of a robust numerical approach that gives a modelization of a mathematical model, under a set of loads and constraints. Therefore, such numerical methodology combined with topology optimization has led to the obtaining of a heuristic model that creates new designs which weight have been reduced by the creation of voids near the less stressed areas, fulfilling an objective function and a set of constraints, considering two main design variables: density and level set.

Therefore, in order to fulfill additive manufacturing constraints - overhang and minimum length scales requirements - an additional shape functional has been defined as a penalty contribution to the main objective function (mainly the compliance and orthotropic effective properties): the anisotropic perimeter tool, being the evolution of the standard (isotropic) perimeter. Both perimeter approaches lead to the resolution of a Diffusion-Reaction equation. Hence, two possibilities to define at which boundaries the perimeter contributes appear, being total (Robin boundary conditions) and relative (Neumann boundary conditions) perimeter.

The known anisotropic perimeter solves both length scale and overhang constraints, so as to allow manufacturability with additive manufacturing technology. On one hand, considering the perimeter as objective function penalty term self-penalizes the creation of boundaries (gray regions), thus avoiding the formation of small geometrical scales.

On the other hand, defining an anisotropic stiffness matrix by including a conductivity matrix in the integration of $\underline{\underline{K}}_{el}$ has defined virtually in which direction the material has the maximum stiffness intensity, hence keeping control in overhanging phenomena by orienting the topologies mainly in the vertical direction.

Regarding the results obtained, first the anisotropic perimeter has been validated by means of a pure minimum perimeter problem, where a square shape has converged into an ellipse with higher eccentricity as the anisotropic scale factor increased, and which major axis orientation is related with the virtual anisotropic stiffness matrix eigenvectors. Lastly, in relation with the case studies, the apparition of local features with small length scales has been avoided when including either isotropic or anisotropic perimeter as penalty term. Furthermore, vertical tendency orientation of topologies has been generally obtained in relation with the anisotropic cases, along with the penalization of horizontal features. It is also worth mentioning the fact that interior perimeter cases severely penalizes the obtained topologies, when compared with their total perimeter complementary cases, even splitting the initial configuration in two parts, as it occurs in the bridge and arch case studies. It must be highlighted also the importance of selecting an accurate interpolation method for any case study, as well as the target parameters settings to allow the precise functioning of shape and topological derivatives, whereas the perimeter acts as a non-negligible penalty term.

Overall, this project has become clearly relevant for the exploration of new lightweighting structures and the ulterior limitation of current environmental impact at all industrial sectors, as the aerospace industry. Indeed, the use of numerical methods concerning topology optimization techniques and additive manufacturing constraints shall allow the minimization of both design time and costs. Besides, as the code has been written to be compatible with collaborative projects, further exploration of future lines of research will be allowed in the course of PhD professionalization, specially when considering phase-field models, high-performance computing and large-scale optimization inside the non-linear regime.

6.2 Future lines of research

Although the main objectives of this project have been successfully accomplished in order to obtain heuristic designs computationally, some aspects in which future work is needed must be taken into consideration. More specifically:

- To expand the anisotropic perimeter code to optimize 3D structures.
- To implement the minimum length scale constraint functional with a density based projection method, inside the Swan project.
- To implement adaptive algorithms where white values does not account for the stiffness matrix of the structure (continuous refactoring of the stiffness matrix), along with selective geometry methods.
- To test experimentally the resulting manufacturing of the optimized parts, under the same boundary conditions as the numerical experiment.
- To move further and design optimal structures in the non-linear regime, specially when considering plasticity and damage models.

Appendix A

Programming Environment

Prior to gain knowledge inside the field of topology optimization methods, in this chapter a definition of the programming environment which will be used throughout the project evolution is provided. The importance of developing this code following the guidelines and techniques depicted in the following sections is crutial, above all taking into consideration that all the work carried out shall be able to be combined with future code.

Therefore, the Object-Oriented Programming technique is employed, along with the test-driven development philosophy and the Unified Modeling Language representation method, which will schematize the code algorithm at user level. Finally, an introduction to the Git Environment will be exposed.

A.1 Object-Oriented Programming

Software technology has been used since the 1950s in order to solve a complex problem by means of an assembled machine language. Over the last five decades, the evolution of software technology has defined two different types of programming techniques, firstly a procedure-oriented and hence an object-oriented programming. [23]

Procedure-Oriented Programming. Regarding the first improvement or phase, defined as Procedure-Oriented Programming (POP), the complex problem is solved by following a cartesian-based method, where the main program is divided into a sequence of secondary tasks that shall be done in order to obtain the last result. Besides, these new tasks may be also divided into smaller tasks, even related with several predecessors, and thus keeping this procedure working until atomic tasks are reached. At software technology level, these tasks are known as functions. Hence, atomic functions are activated separately in such a way that more complex functions can be accessed until the main program is completely solved. This hierarchical decomposition technique is the most common one to maintain a first contact with a determined programming language, as it is simple to implement.

In Figure A.1 a common algorithm decomposition scheme is depicted. Although this programming technique is widely used at user level, POP presents a clear disadvantage in relation with its exploitation to external programmers. Therefore, a procedure-oriented code has a lack of dynamism when considering its reusability.

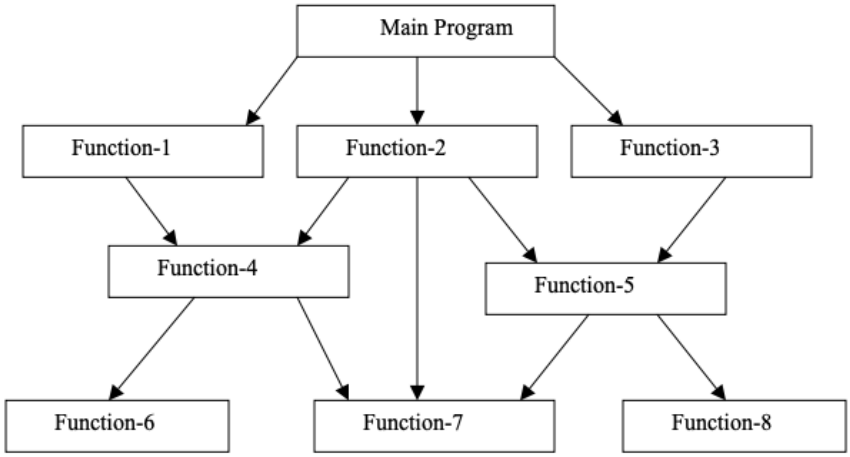


FIGURE A.1: Typical hierarchical decomposition of procedural oriented algorithms. [23]

Furthermore, in this multi-functional problem some data shall be placed locally inside each function as volatile variables, and also other data may be defined globally, such that all the functions can access it. Therefore, a clear drawback appears when considering the vulnerability of global data to an inadvertent change by a function. Thus, the motivating factor aiming to eliminate some of conventional programming disadvantages defined the second improvement, also with the purpose of incorporating the best POP features with other new powerful concepts.

Object-Oriented Programming (OOP). Treats data as a critical element, since it does not allow it to be placed freely around the complex problem decomposition, hence locating data closely to the function that operate on it. In fact, OOP allows the decomposition into a number of entities called objects and thus stores properties and functions around these objects. In Figure A.2 the organization scheme of OOP is exposed. The public functions associated with a certain object can access its properties, as well as the functions of other objects.

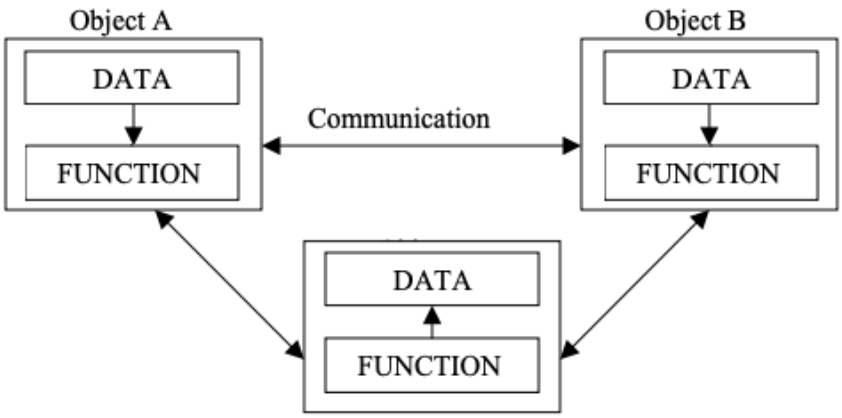


FIGURE A.2: Data and functions organization in Object-Oriented Programming. [23]

This new programming technique implies a higher reusability dynamism, since some of the objects into which a program is divided may be also complement the composition of another program, unlike traditional successor functions. In fact, a determined

object does not serve a specific complex problem, and it shall be used for a large variety of complex problems, relatively similar between them. This feature helps to optimize code development time regarding reusability and tangibility by external programmers. Other features of OOP are the following:

- Main focus on data rather than procedure.
- Properties and functions structures characterize the object to which they belong.
- Some part of data can be hidden.
- It is possible to create multiple instances of an object without any interference.
- OOP can be easily upgraded from small to large problems.

In Object-Oriented Programming there are two concepts extensively used: Objects and Classes. Objects consists in the basic run time entities in an OOP system, and they shall be chosen so that real-world objects are matched. The complex problem is analyzed throughout objects and the communication rule between them, being able to interact without knowing details of each other's code. Moreover, Classes are the basic reflection of a certain collection of brother objects. In fact, objects are variables of the type class. Once a class is defined, any number of objects belonging to that class can be created with different purposes and properties.

Finally, OOP is not the right of any specific programming language and it can be implemented using languages such as C and Python. Nevertheless, in this project the language used is MATLAB, as it is a C-based one, hence offering an easier interface to implement all the code since C is a language specially designed to support OOP.

A.2 Test-driven development philosophy

During the development of any complex project by means of a coupled object-oriented code, it is important to obtain a composition of objects with a high degree of reusability, thus implying the creation of objects without any duplicated set of functions performing the same task in different classes. Therefore, the goal is to obtain clean code that works. [24]

Clean code that works improves the lives of programmers, as it is a predictable way to develop. The method that allows users to get clean code without taking into consideration the addition of possible imperceptible sources of errors is called Test-Driven Development (TDD). This development style will be used throughout the evolution of this project.

The two simple rules in TDD consists in writing new code for a failing automated test, and hence eliminating duplication. The development environment must provide rapid response to small changes by providing feedback between decisions. Finally, such two rules imply an order to the programming tasks, as listed below:

- Red: creation of a test that does not work.
- Green: making the test to work quickly.
- Refactor: elimination of all the duplication created during the Green step.

A.3 Unified Modeling Language

The last step to clearly define the Programming Environment, beyond the creation of a set of objects that together solve the initial complex problem, consists in exposing the way in which these objects are communicating between them, at user level.

Unified Modeling Language (UML) is a graphical language officially defined by the Object Management Group with the purpose of providing visualization, specification, construction and documentation of objects inside a software system. UML offers a standard way to schematize all system functions, along with programming language statements, database schemes and reusable components. Hence, UML represents basically the projection into the complex problem system. Additionally, these diagrams might be developed in different abstraction levels depicting the most important ideas in each one of the level. [25]

The development of an UML involves both structural and behavioral modeling. On the one hand, structure diagrams aim to schematize the static aspect of a system, defining for instance diagrams of each of the classes and objects. On the other hand, behavioral diagrams are used to represent the dynamics aspects of a system, such as interaction diagrams, which aim to define the sequence, communication and interaction between classes. By using interaction diagrams, it is easier to understand the control flow within components.

In Figure A.3 a class representation scheme is depicted. This representation illustrates the structural or static viewpoint of a system. Furthermore, it also clearly defines the template for creating its corresponding objects.

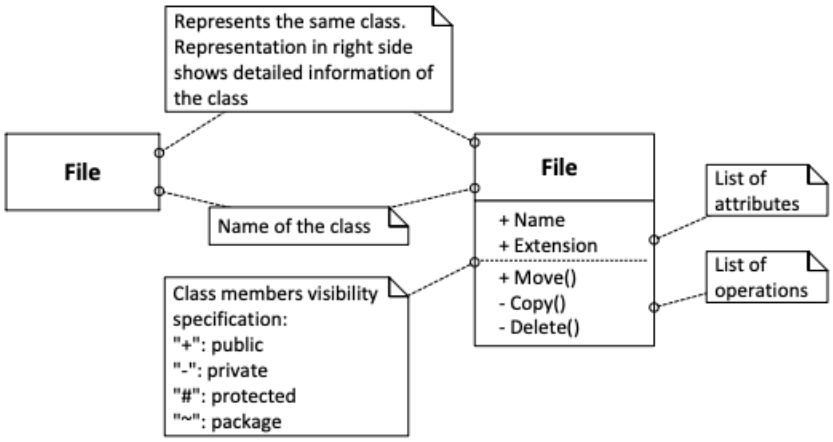


FIGURE A.3: Class representation example. [25]

Moreover, the most important relationships between these elements, exposing either physical or logical relations, are association and dependency. Association shows the physical structure of things, whereas dependency states which entity is using the information and services of another entity. In Figure A.4 an example is shown considering these relationships, applied to an enterprise data synchronization system. Association relationships are depicted with a continuous arrow, whereas dependency is defined with a non-continuous arrow.

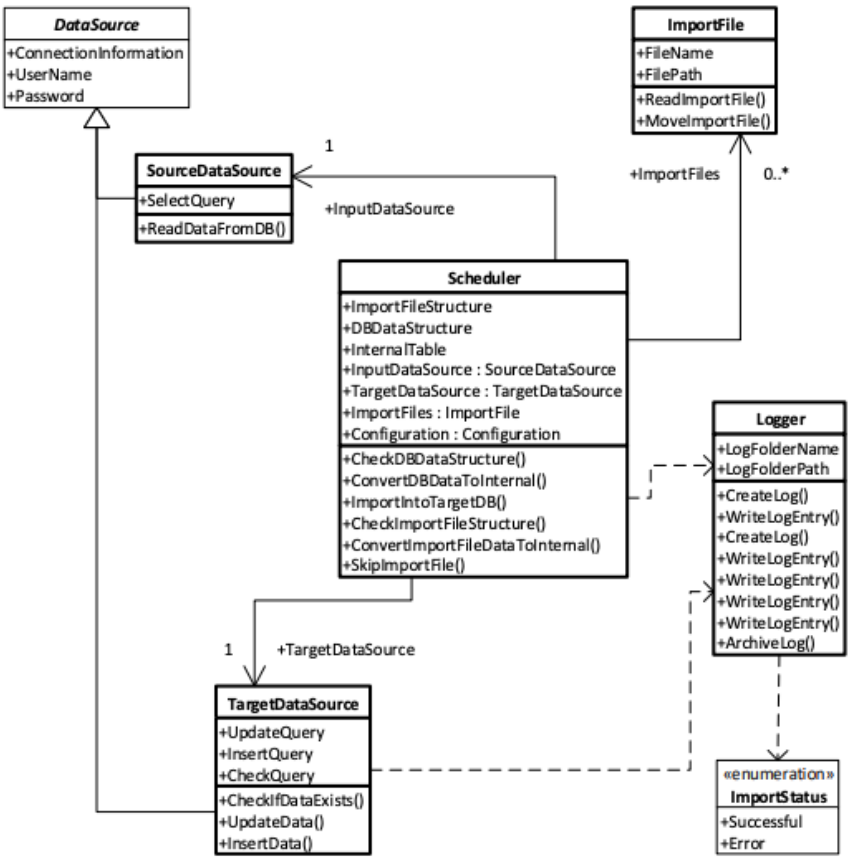


FIGURE A.4: Class diagram of enterprise data synchronization system. [25]

Finally, other common instance-level relationships that may appear in a generic UML class diagram are defined below:

- Inheritance: a super type class represents the specialized form of other subclasses that share the properties and functions from the predecessor class, but may define an additional function, depending on the specific purpose. Therefore, an instance of the subtype class is also an instance of the superclass. This is depicted with a hollow-triangle ended arrow (the arrow's origin contains the subclass and the hollow triangle the superclass).

- Aggregation: variant of the association relationship, being more specific. An aggregation occurs when a class is a collection of other classes in its input. This is represented as a hollow diamond shape on the containing class with a single line to the contained one.

- Composition: a determined class calls at some function an instance of another class. This is represented with a filled diamond shape on the containing class, connected with a single line to the contained one.

- Association: structural relationship specifying that objects of one class are connected to objects of another class if needed (data-driven view). This is represented with a dotted line with arrows on both sides, connecting two classes.

A.4 Git Environment

A.4.1 Version Control Systems

Git is a Version Control System (VCS) which is able to manage multiple versions of a collaborative project, by tracking each change to the different files, offering a way to undo or roll back each project stage. [26]

With Git, multiple people can work on their copy of the project, referred as branches, so that their work can be finally merged into the main version when the team members are satisfied with the work. Specifically, the following functionalities are found:

- Go back and forth between multiple versions.
- Review differences between versions.
- Check file change history.
- Review changes made by others.

The main feature of Git is its *Three States* system, more specifically:

- Working directory - current snapshot in which developer is working on.
- Staging area - modified files are marked in their current version.
- Git directory - history database.

Therefore, one may modify the corresponding files in order to add them to the staging area, thus taking the snapshot and finally adding them to the database.

Lastly, the present project's source code is stored in *GitHub* server repository *SwanLab/Swan*. [27] Now, during this project, the main characteristics and utilities that Git has in order to fulfill its functionalities are presented.

A.4.2 Branches

A branch is just the developer independent copy of the project at a certain time, being the main feature behind code reviews. In this copy, many changes might be made such that other people's work will not result affected. Branches are usually created in order to develop new features of the master code.

Furthermore, the main or master code is itself the production branch, where future customers would be able to obtain the last stable version of the software. Therefore, several development branches are created from the master one, where all the progress and commits will be recorded, and keeping the branch updated with master. These patching branches continuously live and die with a pull request into the master branch. It is very important to remark that a developer must not commit never to the master branch directly, since making pull requests from patching branches allow the project directive team to review and test all changes before merge them into the master code.

Lastly, although branches are mainly created to work locally, it is also possible and advisable to push them to remote, so that these branches will be published to the remote repository and everyone inside the project will be able to see the corresponding progress.

A.4.3 Commits

When working inside a patching branch, the new files created to define the corresponding features are set to *untracked files* by default. These files will remain ignored until the developer decide to stage and commit them. Besides, when a existing file tracked by git is modified, this is classified as *modified/staged* until the developer again commit it in order to take a snapshot of the entire project at this instant.

Thus, as it has been exposed earlier during the development of this section, a commit is defined as a snapshot of the entire project at a certain time. Commits allow git to compare different versions of the project by just identifying the code lines which are different between such versions, without recording specifically individual changes done to the files. Commits also contain information about the author and time. All commits have an heredity link with previous ones, so that the ensemble of all the parentage is also known as the branch.

A.4.4 Pull requests

Pull requests consist in submitting a permission to apply all the commits in a branch to another branch (usually, the master branch). The *pull* concept is completely the opposite of *push*, hence the permission asked deals with executing the pull action on a remote repository. Nevertheless, it is also necessary to later merge the branches together, which is also included in the permission. Thus, a pull request just allows the developers to finally include their work in the master branch. This assures that every fix committed in the patching branches is tested and reviewed.

Appendix B

The Finite Element Method

B.1 The Finite Element Method

B.1.1 The linear static elastic problem

Structural static analysis does not take into consideration time as an independent variable and it is useful when deflections are constant or change slowly. These *quasistatic* problems may include steady inertia loads and also exclude plastic action. In this section a formulation for the linear static elastic problem will be provided, along with the relationship with its properties. Firstly, an abstraction of the general solid element into a bar and beam element will be performed in order to introduce the problem, and hence the behaviour of a generic 3D solid element will be provided.

The importance to study bars and beams prior to the generic 3D element, in addition to providing a simple introduction to linear static analysis, deals with the initial modeling of a complicated structure, which may give useful information before building the most realistic model with a small effort. Bars and beams are already separate elements, just differing in how the internal distribution of forces is defined. Bar elements are hinged together at connection points resisting only axial forces, whereas beam elements are welded at such connection points, thus resisting axial forces, transverse forces and bending moments.

The elastostatics applied to a uniform prismatic bar of length L with Young Modulus E and cross-sectional area A are easily calculated from the elementary expression for stretching a bar an amount $\delta = FL/(AE)$. [28]

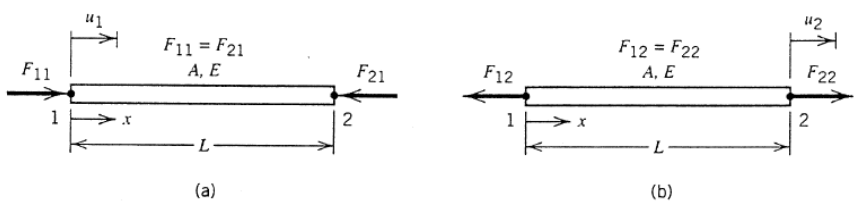


FIGURE B.1: Nodal forces due to deformation of a two-node bar element. (a) Node 1 displaced u_1. (b) Node 2 displaced u_2. [28]

Following the scheme depicted in Figure B.1, a node is located at each end, allowing only axial displacements. If each of these nodes are displaced separately, one gets the force that shall be applied to nodes, such that the displacement state is guaranteed, as shown in Equation (B.1).

$$F_{11} = F_{21} = \frac{AE}{L} u_1 \qquad \text{and} \qquad F_{12} = F_{22} = \frac{AE}{L} u_2 \tag{B.1}$$

Therefore, allowing both nodes to displace simultaneously and considering the positive sign convention when forces are oriented to the right, the equivalent resultant forces F_1 and F_2 applied to the bar at both nodes are shown in Equation (B.2).

$$\begin{bmatrix} F_{11} & -F_{12} \\ -F_{21} & F_{22} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} \tag{B.2}$$

$$\frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} \tag{B.3}$$

Or equivalently, as depicted in Equation (B.3). In this last expression the square matrix, including the factor AE/L, is defined as the element stiffness matrix $\underline{\underline{K_{el}}}$ for a single bar element. Hence, the expression is also written as $\underline{\underline{K_{el}}} \cdot \underline{u} = \underline{F}$.

The methodology already provided for bars can produce a stiffness matrix only for simple elements. For most elements, such as beams, a general or formal procedure shall be used instead. As exposed in [28], the general expression to compute an element stiffness matrix is depicted in Equation (B.4),

$$\underline{\underline{K_{el}}} = \int_{\Omega} \underline{\underline{B}}^T \cdot \underline{\underline{C}} \cdot \underline{\underline{B}} \cdot dV \tag{B.4}$$

where $\underline{\underline{B}}$ is the strain-displacement matrix, $\underline{\underline{C}}$ the constitutive matrix and Ω the element domain in which the stiffness is computed. In fact, this expression states that internal work stored inside the element is due to elastic strain energy. In the particular case of a plane beam element, $\underline{\underline{C}} = EI$, where I is the centroidal moment of inertia related to its cross-sectional area. Besides, for beams the strain energy depends on curvature d^2v/dx^2, being $v = v(x)$ the lateral displacement of a plane beam element.

The interpolation of lateral displacement of the plane beam elements from its nodal degrees of freedom (related to in-plane bending and transverse shear force) is proposed in Equation (B.5),

$$v = \begin{bmatrix} N_1 & N_2 & N_3 & N_4 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ \theta_{z1} \\ v_2 \\ \theta_{z2} \end{bmatrix} = \underline{\underline{N}} \cdot \underline{d} \tag{B.5}$$

where each N_i is defined as shape functions, which describe how the displacement changes with the local coordinate when the corresponding degree of freedom is unity, while the other is zero. A relationship between matrices $\underline{\underline{N}}$ and $\underline{\underline{B}}$ can be set through spatial derivative of shape functions. In Figure B.2 the definition of each shape function is provided for a simple plane beam element.

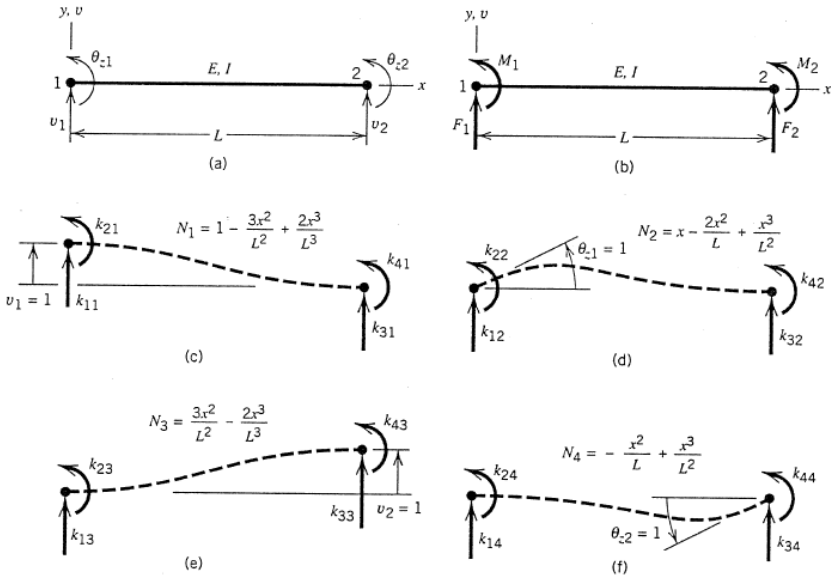


FIGURE B.2: (a) Simple plane beam element within its nodal degrees of freedom. (b) Nodal loads related to degrees of freedom. (c-f) Deflected shapes and shape functions associated with activation of each degree of freedom. [28]

Hence, curvature of the beam element is obtained in Equation (B.6).

$$\frac{d^2v}{dx^2} = \left[\frac{d^2}{dx^2}\underline{\underline{N}}\right] \cdot \underline{d} = \underline{\underline{B}} \cdot \underline{d} \tag{B.6}$$

Using now Equation (B.4) with all the definitions exposed, one gets the element stiffness matrix for a simple plane beam element, shown in Equation (B.7).

$$\underline{\underline{K_{el}}} = \begin{bmatrix} 12EI/L^3 & 6EI/L^2 & -12EI/L^3 & 6EI/L^2 \\ 6EI/L^2 & 4EI/L & -6EI/L^2 & 2EI/L \\ -12EI/L^3 & -6EI/L^2 & 12EI/L^3 & -6EI/L^2 \\ 6EI/L^2 & 2EI/L & -6EI/L^2 & 4EI/L \end{bmatrix} \tag{B.7}$$

Thus, considering for instance a 2D beam element, which is a combination of a bar element and a simple plane beam element, now the structure withstands axial stretching, transverse shear force and bending in one plane. By combination of Equations (B.3) and (B.7), the stiffness matrix of a 2D beam element is depicted in Equation (B.8).

$$\underline{\underline{K_{el}}} = \begin{bmatrix} AE/L & 0 & 0 & -AE/L & 0 & 0 \\ 0 & 12EI/L^3 & 6EI/L^2 & 0 & -12EI/L^3 & 6EI/L^2 \\ 0 & 6EI/L^2 & 4EI/L & 0 & -6EI/L^2 & 2EI/L \\ -AE/L & 0 & 0 & AE/L & 0 & 0 \\ 0 & -12EI/L^3 & -6EI/L^2 & 0 & 12EI/L^3 & -6EI/L^2 \\ 0 & 6EI/L^2 & 2EI/L & 0 & -6EI/L^2 & 4EI/L \end{bmatrix} \begin{matrix} u_1 \\ v_1 \\ \theta_{z1} \\ u_2 \\ v_2 \\ \theta_{z2} \end{matrix} \tag{B.8}$$

Lastly, the term 3D Solid will be used to mean a three-dimensional solid that is unrestricted as to shape, loading, material properties and boundary conditions. Hence, all six possible stresses (three normal and three shear) must be taken into consideration, as shown in Figure B.3.

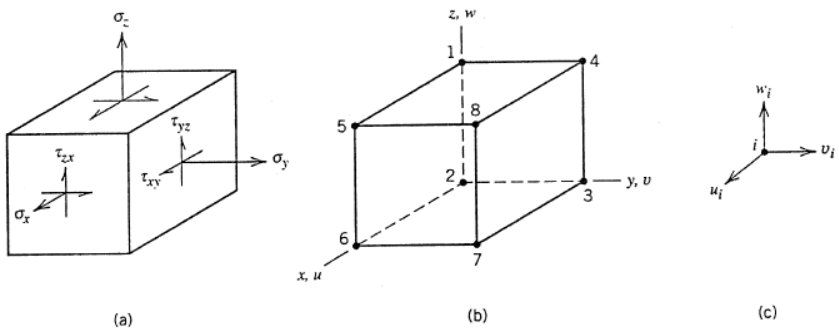


FIGURE B.3: (a) Three-dimensional stress state. (b) 8-node hexahedron element. (c) Degrees of freedom at any node. [28]

Furthermore, the displacement field involves all three translational components. 3D models are the hardest to prepare, thus defining the most general case of analysis. Nevertheless, they also are the ones demanding the highest computer resources.

In Equation (B.9) the strong form of elastostatics problem is depicted.

$$\begin{array}{ll} \frac{\partial \sigma_{ij}}{\partial x_j} + f_i = 0 & \text{in } \Omega \\ u_i = \hat{u}_i & \text{on } \Gamma_u^i \\ \sigma_{ij} n_j = \hat{t}^i & \text{on } \Gamma_\sigma^i \end{array} \tag{B.9}$$

Where a constitutive relation for these 3D elements is also provided in Equation (B.10).

$$\sigma_{ij} = C_{ijkl} \epsilon_{kl} \tag{B.10}$$

Noting that C_{ijkl} is also referred to the constitutive matrix previously seen in Equation (B.4), or elasticity matrix. Using Voigt notation, the elasticity matrix for isotropic materials (those ones which mechanical behaviour does not depend in the direction) is exposed in Equation (B.11).

$$\underline{\underline{C}} = \begin{bmatrix} \lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda + 2\mu & \mu & 0 & 0 & 0 \\ \lambda & \lambda & \lambda + 2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix} \begin{matrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ \gamma_{23} \\ \gamma_{13} \\ \gamma_{12} \end{matrix} \tag{B.11}$$

Where λ and μ are the Lame parameters, which are functions of the Young's modulus and the Poisson ratio ν, as seen in Equation (B.12).

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)} \qquad \mu = G = \frac{E}{2(1+\nu)} \tag{B.12}$$

Therefore, only the definition of matrix $\underline{\underline{B}}$ remains to be completed in order to obtain a general approximation for the element stiffness matrix of a solid element. In order to complete the definition, the strain-displacement relations with displacement components u, v and w are exposed in Equation (B.13).

$$\begin{aligned} \epsilon_{11} &= \frac{\partial u}{\partial x} \quad \gamma_{12} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \\ \epsilon_{22} &= \frac{\partial v}{\partial y} \quad \gamma_{23} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \epsilon_{33} &= \frac{\partial w}{\partial z} \quad \gamma_{13} = \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \end{aligned} \tag{B.13}$$

Besides, displacements $\underline{u}$ within a solid element might be interpolated from nodal displacements $\underline{d}$, once again considering shape functions: $\underline{u} = \underline{\underline{N}} \cdot \underline{d}$. As it will be discussed later, the shape functions employed will differ depending on the type of element chosen for constructing all the structure. Nevertheless, in Equation (B.14) an interpolation between two nodes is provided.

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & ... \\ 0 & N_1 & 0 & 0 & N_2 & 0 & ... \\ 0 & 0 & N_1 & 0 & 0 & N_2 & ... \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ v_1 \\ w_1 \\ u_2 \\ v_2 \\ w_2 \\ ... \end{bmatrix} \tag{B.14}$$

Finally, substitution of $\underline{u} = \underline{\underline{N}} \cdot \underline{d}$ into the strain-displacement relations shown in Equation (B.13) yields the strain-displacement matrix $\underline{\underline{B}}$, which in consequence completes the definition of the stiffness matrix for a 3D solid element, as depicted in Equations (B.15) and (B.16), respectively.

$$\underline{\epsilon} = \underline{\underline{B}} \cdot \underline{d} \tag{B.15}$$

$$\underline{\underline{K_{el}}} = \iiint_{\Omega} \underline{\underline{B^T}} \cdot \underline{\underline{C}} \cdot \underline{\underline{B}} \cdot dxdydz \tag{B.16}$$

The mathematical formulation for 2D solid elements in the XY plane is completely analogous to the procedure depicted for 3D ones, defining the required basis to implement plane stress in optimization problems.

B.1.2 Theoretical basis of Finite Element Methods

The Finite Element Method (FEM) is a numerical approach used to solve a large variety of physical problems regarding engineering analysis and designs. This is usually employed in structural problems, given a model with a determined set of loads and constraints. Therefore, considering the idealization of the problem within a set of hypothesis, a mathematical model is created leading to the formulation of differential equations, also referred as the *strong form* formulation of the problem. Thus, FEM techniques aim to compute the corresponding solution to this kind of equations throughout a numerical procedure, taking into consideration that the solution will only be representative of the mathematical model, the related hypothesis

and the boundary conditions considered. [29]

The analysis process related to the Finite Element Method is defined through the steps depicted below.

1. **Physical problem definition**. FEM is applicable to a large variety of structural problems. For instance stress and strain analysis (mechanical properties assessment), or main modal shapes and natural frequencies determination (modal analysis).

2. **Mathematical model definition**. The governing differential equations of the problem must be determined ($f(u, \nabla^2 u) = 0$), stating hypothesis in relation with the following topics:

   - Geometry.
   - Kinematics.
   - Materials law.
   - External loads.
   - Boundary conditions. These conditions can define either a Dirichlet (imposition of the unknown u) or Neumann (imposition of the unknown gradient ∇u) constraint over a fragment of the domain's boundary.

3. **Finite Element Analysis**. The structural domain is discretized into a number of finite mesh elements, selecting the element shape, the mesh density and the unknown parameter u to be solved. Besides, all these parameters shall be representative of external loads and boundary conditions.

4. **Results convergence study**. Once the results corresponding to the differential equation (i.e, the representative mathematical model) are obtained, probably the u-field and its gradient would not represent the expected behaviour, since the problem has been solved with a discretization into finite elements. Nevertheless, as the mesh density increases, the field obtained would fit better with the exact solution. Therefore, considering a convergence criteria, while the results obtained do not satisfy the model validation, the domain must be meshed again with a higher density of elements and then returning to step 3. Once the convergence criteria is satisfied, the study would continue to step 5. An indicative parameter of this kind of study is the error norm evolution between the computed variable and a reference value, thus studying its stability in order to optimize computational cost.

5. **Interpretation of results**. Once the differential equation is completely solved by means of this numerical procedure, the related conclusions can be extracted. Therefore, some future improvements might be proposed. Mainly, one can consider to propose a harder mathematical model (step 2) either with a higher degree of the differential equations or an extension of the considered space dimensions. Moreover, it is also recommended to return again to step 1 and redefine the physical problem, hence implying the reduction of some hypothesis in order to obtain better feasibility in the design behaviour if the previous mathematical model was completely optimized.

Considering all aspects shown above, it can be concluded that, given a physical problem, the results obtained are effective and reliable when the following affirmations are fulfilled:

- The most effective mathematical model applied to the structural analysis fits the hypothetical model that represents the true solution, with enough precision and minimum computational cost.
- The mathematical model is reliable if the solution of the differential equation could be predicted with the most hardest version of such model.

B.1.3 Finite Element Methods in structural problems

Once the theoretical basis of finite element methods is defined, along with the linear static elastic problem, both concepts shall be coupled in order to extend the application of finite element methods to solve the general elastostatics problem with solid elements. Therefore, the *strong form* of this phenomenon must be considered again, which is depicted in Equation (B.9). These differential equations are applied to a continuous system, representative of a structural differential element equilibrium (variables of state), which is complemented with the related boundary conditions.

The next step consists in finding a reformulation of the *strong form* since the direct resolution method can not always be applied. This will be performed considering the *Galerkin* formulation, the Virtual Works Principle and the numerical integration within the Gauss domain. [30]

Inside the field known as variational formulation (*weak formulation*) the following functions must be identified:

- *Trial functions* u verifying the same boundary conditions that state variables. I.e, $u_i = \bar{u}^i$ on Γ_u^i.
- *Test functions* v, which are the difference between two any trial functions, such that they value is equal to zero at the region where boundary conditions are set. I.e, $v_i = 0$ on Γ_u^i.

Moreover, the *Galerkin* formulation aims to approximate both functions with a similar procedure previously seen in Section B.1.1, using interpolating shape functions between nodal values, as depicted in Equation (B.17).

$$\begin{gathered} u \simeq \underline{\underline{N}} \cdot \underline{d} \\ v \simeq \underline{\underline{N}} \cdot \underline{c} \qquad \forall \underline{c} / \underline{c}_i = 0 \text{ on } \Gamma_u^i \end{gathered} \tag{B.17}$$

Considering again the *strong form* exposed in Equation (B.9), the Virtual Works Principle states that all virtual state variables field, which is compatible with the connections, will verify that external and internal virtual work are of the same magnitude. Therefore, the *weak form* formulation is obtained by integrating over the domain the product of the *strong form* equation with the test functions, as depicted in Equation (B.18).

$$\int_\Omega v_i \left(\frac{\partial \sigma_{ij}}{\partial x_j} + f_i \right) \cdot d\Omega = 0 \quad \rightarrow \quad \int_\Omega v_i \frac{\partial \sigma_{ij}}{\partial x_j} \cdot d\Omega + \int_\Omega v_i f_i \cdot d\Omega = 0 \tag{B.18}$$

As shown in [30], by applying integration by parts and considering the Divergence theorem, one transforms the left integral of previous equation into the expression exposed

$$\int_{\Omega} v_i \frac{\partial \sigma_{ij}}{\partial x_j} \cdot d\Omega = \int_{\Gamma} v_i (\sigma_{ij} n_j) \cdot d\Gamma - \int_{\Omega} \frac{\partial v_i}{\partial x_j} \sigma_{ij} \cdot d\Omega. \tag{B.19}$$

Considering both Dirichlet and Neumann boundary conditions, previous expression is transformed into

$$\int_{\Omega} v_i \frac{\partial \sigma_{ij}}{\partial x_j} \cdot d\Omega = \sum_{i=1}^{n_{sd}} \int_{\Gamma_\sigma^i} v_i \bar{t}^i \cdot d\Gamma - \int_{\Omega} \frac{\partial v_i}{\partial x_j} \sigma_{ij} \cdot d\Omega \tag{B.20}$$

where n_{sd} indicates the problem dimension and Γ_σ^i the region on Neumann boundary conditions are set. Inserting the preceding expression into Equation (B.18), and considering the symmetry tensor condition $\frac{\partial v_i}{\partial x_j} = [\nabla^s v]_{ij}$, thus Equation (B.21) is obtained.

$$\int_{\Omega} [\nabla^s v]_{ij} \sigma_{ij} \cdot d\Omega = \sum_{i=1}^{n_{sd}} \int_{\Gamma_\sigma^i} v_i \bar{t}^i \cdot d\Gamma + \int_{\Omega} v_i f_i \cdot d\Omega. \tag{B.21}$$

Finally, inserting the constitutive relationship $\sigma_{ij} = C_{ijkl} [\nabla^s u]_{ij}$ into previous expression, the complete weak formulation of the classic elastostatics problem is obtained, as shown

$$\int_{\Omega} [\nabla^s v]_{ij} C_{ijkl} [\nabla^s u]_{ij} \cdot d\Omega = \sum_{i=1}^{n_{sd}} \int_{\Gamma_\sigma^i} v_i \bar{t}^i \cdot d\Gamma + \int_{\Omega} v_i f_i \cdot d\Omega \quad \forall v_i / v_i = 0 \text{ on } \Gamma_u^i. \tag{B.22}$$

The last step consists in applying the *Galerkin* formulation onto the *weak form* expression in order to obtain the well known matrix equation. This is:

$$\int_{\Omega} \underline{\underline{B}} \cdot \underline{c} \cdot \underline{\underline{C}} \cdot \underline{\underline{B}} \cdot \underline{d} \cdot d\Omega = \sum_{i=1}^{n_{sd}} \int_{\Gamma_\sigma^i} \underline{\underline{N}} \cdot \underline{c} \cdot \bar{t}^i \cdot d\Gamma + \int_{\Omega} \underline{\underline{N}} \cdot \underline{c} \cdot \underline{f} \cdot d\Omega. \tag{B.23}$$

With a simple matrix product transformation, one gets Equation (B.24).

$$\underline{c}^T \left(\int_{\Omega} \underline{\underline{B}}^T \cdot \underline{\underline{C}} \cdot \underline{\underline{B}} \cdot d\Omega \right) \cdot \underline{d} = \underline{c}^T \left(\sum_{i=1}^{n_{sd}} \int_{\Gamma_\sigma^i} \underline{\underline{N}}^T \cdot \bar{t}^i \cdot d\Gamma + \int_{\Omega} \underline{\underline{N}}^T \cdot \underline{f} \cdot d\Omega \right) \tag{B.24}$$

Lastly, previous expression is analogous to the one depicted in Equation (B.25).

$$\underline{c}^T (\underline{\underline{K}} \cdot \underline{d}) = \underline{c}^T (\underline{F}) \quad \rightarrow \quad \underline{\underline{K}} \cdot \underline{d} = \underline{F} \tag{B.25}$$

Where $\underline{\underline{K}}$ is the stiffness matrix, $\underline{d}$ the nodal displacements vector and $\underline{F}$ the external forces vectors. Once again, it is obtained the same definition of the stiffness matrix as in Equation (B.16). Besides, the definition of $\underline{F}$ is depicted in Equation (B.26). The

vector of nodal forces is related to external loads, taking into consideration punctual, surface and body forces.

$$\underline{F} = \sum_{i=1}^{n_{sd}} \int_{\Gamma_\sigma^i} \underline{\underline{N}}^T \cdot \bar{t}^i \cdot d\Gamma + \int_\Omega \underline{\underline{N}}^T \cdot \underline{f} \cdot d\Omega \tag{B.26}$$

Nevertheless, at this stage of the mathematical development Equation (B.25) is not completed at a global point of view. Indeed, the whole domain may be composed of several types of elements due to mesh refinement and the presence of several distributed loads. Besides, different shapes shall be employed depending on the geometry of the domain. Therefore, a re-definition of all parameters of the matrix equation is provided in Equations (B.27) and (B.28).

$$\underline{\underline{K}} = A_{e=1}^{n_{el}} \left(\underline{\underline{K}}^e\right) = \sum_{e=1}^{n_{el}} \underline{\underline{L}}^{e^T} \cdot \underline{\underline{K_{el}}}^e \cdot \underline{\underline{L}}^e \tag{B.27}$$

$$\underline{F} = A_{e=1}^{n_{el}} \left(\underline{F}^e\right) = \sum_{e=1}^{n_{el}} \underline{\underline{L}}^{e^T} \cdot \underline{F}^e \tag{B.28}$$

Where $\underline{L}^e$ is a matrix consisting of integers 0 and 1 that establishes the relationship between elemental nodal variables with the global nodal vector $\underline{d}$. Once this assembly process is performed, now the matrix equation can be represented in its non-coupled form. This is performed since the original matrix equation is singular, hence not providing a direct resolution of the system of equations at this stage. This is:

$$\begin{bmatrix} K_{ff} & K_{fr} \\ K_{rf} & K_{rr} \end{bmatrix} \cdot \begin{bmatrix} d_f \\ d_r \end{bmatrix} = \begin{bmatrix} F_f \\ F_r \end{bmatrix} + \begin{bmatrix} 0 \\ \underline{R} \end{bmatrix} \tag{B.29}$$

where f represents the subset of *free* nodes and r the subset of *restricted* nodes. Equivalently, each subset corresponds to those nodes related to Neumann and Dirichlet boundary conditions, respectively. Therefore, each term of the type $\underline{\underline{\phi_{ij}}}$ defines a secondary term, born from the selection of rows and columns exposed in subsets i and j, respectively. Besides, $\underline{R}$ corresponds to the vector of reaction forces on Dirichlet nodes. Thus, the unknowns of the matrix problem are $\underline{d}_f$ and $\underline{R}$.

In order to solve the system of equations, each non-coupled matrix equation is considered, as follows.

$$K_{ff} d_f + K_{fr} d_r = F_f \;\rightarrow\; d_f = K_{ff}^{-1} \cdot (F_f - K_{fr} d_r) \tag{B.30}$$

$$K_{rf} d_f + K_{rr} d_r = F_r + R \;\rightarrow\; R = K_{rf} d_f + K_{rr} d_r - F_r \tag{B.31}$$

Now, matrix $\underline{\underline{K}}_{ff}$ is invertible.

B.1.4 Evaluation of integrals via Gauss quadrature

Lastly, we need a procedure to compute both the stiffness matrix and external force vector for each element, considering the integral definition (weak form). Since the dimensions of the elements of the mesh are completely arbitrary for different case studies, we will use an approach which allows to transform the elements into a reference domain, where $\underline{\underline{K}}$ and $\underline{F}$ will be finally computed. This is known as the Gauss quadrature approach. [30]

In general terms, let $f : \bar{\Omega}^e \rightarrow \mathbb{R}$ to represent some function that shall be evaluated over the physical domain $\bar{\Omega}^e$ (see Equation (B.32)).

$$I = \int_{\Omega^e} f \cdot d\Omega \tag{B.32}$$

The above integral must be computed in the *parent domain* by considering the following change of variables: $x = x(\xi)$. Considering elementary calculus, one transforms previous expression into Equation (B.33).

$$I = \int_{\Omega_\xi} J^e(\xi) f(\xi) \cdot d\Omega_\xi \tag{B.33}$$

where $d\Omega_\xi = d\xi d\eta$ for 2D problems, and $d\Omega_\xi = d\xi d\eta d\zeta$ for 3D problems. Finally, let ξ_g and w_g $(g = 1, 2...m)$ to represent the position and corresponding weight of the $g - th$ Gauss point. Therefore, the approximation of Equation (B.33) by Gauss quadrature reads the expression depicted in Equation (B.34). One-dimensional Gaussian rules are employed on each coordinate separately. In Appendix C, examples of finite elements in the parent domain are shown.

$$\int_{\Omega_\xi} J^e(\xi) f(\xi) \cdot d\Omega_\xi \simeq \sum_{g=1}^{m} w_g J^e(\xi_g) f(\xi_g) \tag{B.34}$$

Appendix C

Finite elements in Gauss quadrature

In this Appendix some examples of finite elements in the *parent domain* are shown in order to establish the procedure required to apply Gauss quadrature during the integration of the elementary stiffness matrix and external force vector. [30]

C.1 Bilinear quadrilateral element

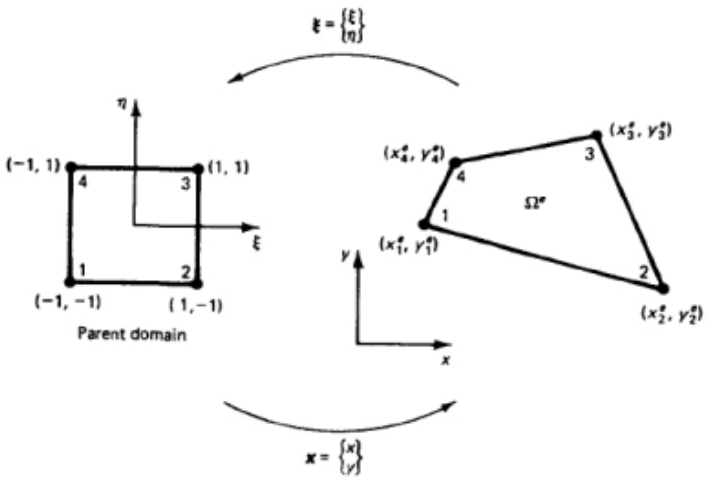


FIGURE C.1: Bilinear quadrilateral element domain and local node ordering. [30]

Matrix of shape functions: Equation (C.1).

$$\underline{\underline{N}}^e = \frac{1}{4}\left[(1-\xi)(1-\eta) \quad (1+\xi)(1-\eta) \quad (1+\xi)(1+\eta) \quad (1-\xi)(1+\eta)\right] \tag{C.1}$$

Element boundary tractions vector: Equation (C.2).

$$\underline{F}^e_{dis} = \frac{||\underline{x}^e_2 - \underline{x}^e_1||}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \cdot \begin{bmatrix} \bar{q}(\underline{x}^e_1) \\ \bar{q}(\underline{x}^e_2) \end{bmatrix} \tag{C.2}$$

2×2 Gaussian rule to integrate the stiffness matrix (4 Gauss points):

1. $\xi_1 = \left(-\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}\right)$, weight $w_1 = 1$.
2. $\xi_2 = \left(\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}\right)$, weight $w_2 = 1$.
3. $\xi_3 = \left(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right)$, weight $w_3 = 1$.
4. $\xi_4 = \left(-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right)$, weight $w_4 = 1$.

Gauss quadrature with $m = 2$ for integral evaluation of tractions loads:

1. $\xi_1 = -\frac{1}{\sqrt{3}}$, $w_1 = 1$.
2. $\xi_2 = \frac{1}{\sqrt{3}}$, $w_2 = 1$.

C.2 Linear triangular element

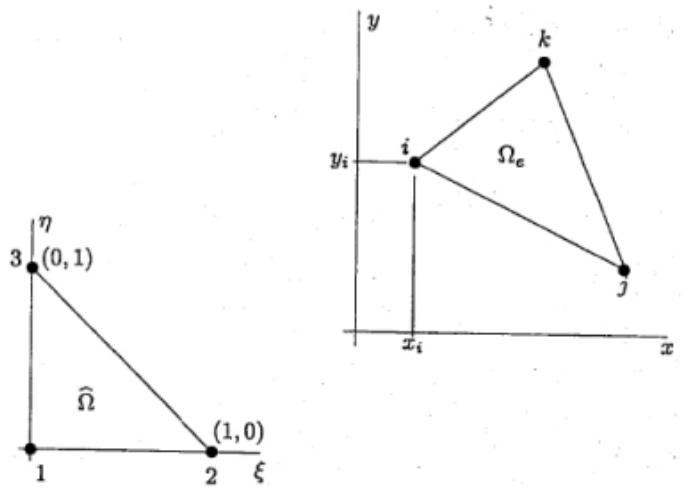


FIGURE C.2: Linear triangular element domain and local node ordering. [30]

Matrix of shape functions: Equation (C.3).

$$\underline{\underline{N}}^e = \begin{bmatrix} 1 - \xi - \eta & \xi & \eta \end{bmatrix} \tag{C.3}$$

As in the case of quadrilaterals, edges of linear triangles are straight lines defined by two nodes, thus obtaining the same expression for distributed loads, see Equation (C.2).

1×1 Gaussian rule to integrate the stiffness matrix (1 Gauss point):

1. $\xi_1 = \left(\frac{1}{3}, \frac{1}{3}\right)$, weight $w_1 = \frac{1}{2}$.

Bibliography

[1] L Zhu, N Li, and PRN Childs. "Light-weighting in aerospace component and system design". In: *Propulsion and Power Research* 7.2 (2018), pp. 103–119.

[2] J Sharma and S Simhambhatla. "Additive manufacturing of complex shapes through weld-deposition and feature based slicing". In: *The ASME 2015 International Mechanical Engineering Congress & Exposition (ASME/IMECE-2015). The Gorge bush Convention Center, Houston*. 2015.

[3] Jie Gao et al. "Topology optimization of micro-structured materials featured with the specific mechanical properties". In: *International Journal of Computational Methods* 17.03 (2020), p. 1850144.

[4] Martin Philip Bendsoe and Ole Sigmund. *Topology optimization: theory, methods, and applications*. Springer Science & Business Media, 2003.

[5] Àlex Ferrer Ferré. *Multi-scale topological design of structural materials: an integrated approach*. Universitat Politècnica de Catalunya, 2017.

[6] Ferran De La Fuente Pi. "Study of 3D printing in optimal design of structures and materials". MA thesis. Universitat Politècnica de Catalunya, 2017.

[7] S Amstutz et al. "Topological derivative for multi-scale linear elasticity models applied to the synthesis of microstructures". In: *International Journal for Numerical Methods in Engineering* 84.6 (2010), pp. 733–756.

[8] RJ Yang and CJ Chen. "Stress-based topology optimization". In: *Structural optimization* 12.2 (1996), pp. 98–105.

[9] Wolfgang Achtziger and Michal Kočvara. "Structural topology optimization with eigenvalues". In: *SIAM Journal on Optimization* 18.4 (2008), pp. 1129–1164.

[10] Blaise Bourdin. "Filters in topology optimization". In: *International journal for numerical methods in engineering* 50.9 (2001), pp. 2143–2158.

[11] Samuel Amstutz, Charles Dapogny, and Àlex Ferrer. "A consistent relaxation of optimal design problems for coupling shape and topological derivatives". In: *Numerische mathematik* 140.1 (2018), pp. 35–94.

[12] Alex Ferrer. "SIMP-ALL: A generalized SIMP method based on the topological derivative concept". In: *International Journal for Numerical Methods in Engineering* 120.3 (2019), pp. 361–381.

[13] Jorge Nocedal and Stephen J. Wright. *Numerical optimization*. Springer, 1995.

[14] Erik Burman et al. "CutFEM: discretizing geometry and partial differential equations". In: *International Journal for Numerical Methods in Engineering* 104.7 (2015), pp. 472–501.

[15] Seyed Arian and Samuel Amstutz. "Γ-convergence approximation of the anisotropic perimeter". In: (2021).

[16] K SRINIVASULU Reddy and Solomon Dufera. "Additive manufacturing technologies". In: *Int. J. of Man., Inf., Tech. and Eng* 4.7 (2016), pp. 89–112.

[17] Guo Liu et al. "Additive manufacturing of structural materials". In: *Materials Science and Engineering: R: Reports* 145 (2021), p. 100596.

[18] Florian Feppon. "Shape and topology optimization of multiphysics systems". PhD thesis. Université Paris-Saclay (ComUE), 2019.

[19] Robert B Haber, Chandrashekhar S Jog, and Martin P Bendsøe. "A new approach to variable-topology shape design using a constraint on perimeter". In: *Structural optimization* 11.1 (1996), pp. 1–12.

[20] Mingdong Zhou et al. "Minimum length scale in topology optimization by geometric constraints". In: *Computer Methods in Applied Mechanics and Engineering* 293 (2015), pp. 266–282.

[21] Fengwen Wang, Boyan Stefanov Lazarov, and Ole Sigmund. "On projection methods, convergence and robust formulations in topology optimization". In: *Structural and multidisciplinary optimization* 43.6 (2011), pp. 767–784.

[22] Karel Kellens et al. "Environmental impact of additive manufacturing processes: does AM contribute to a more sustainable way of part manufacturing?" In: *Procedia Cirp* 61 (2017), pp. 582–587.

[23] Dharminder Kumar and Pooja Chawla. "Introduction of Object Oriented Programming". In: *OOP with C++* (2017).

[24] Kent Beck. *Test-driven development: by example*. Addison-Wesley Professional, 2003.

[25] J Osis and U Donins. "Chapter 1-Unified modeling language: A standard for designing a software". In: *Topological UML modeling* (2017), pp. 3–51.

[26] Mariot Tsitoara. *Beginning Git and GitHub*. Springer, 2020.

[27] *SwanLab/Swan server | GitHub*. URL: https://github.com/SwanLab/Swan (visited on 05/09/2022).

[28] Robert D Cook. *Modeling for Stress Analysis*. New York: Wiley, 1994.

[29] Klaus-Jürgen Bathe. *Finite Element Procedures*. New Jersey: Prentice-Hall, Inc., 1996, pp. 3, 105, 225, 455, 485, 642.

[30] Joaquín A. Hernández. *Composite materials (part II: Structural Analysis) | Introduction and Finite Element Theory*. UPC, 2020.