UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

AUTOMATIC SKELETON CONSTRUCTION FROM BEAM-LIKE TOPOLOGY OPTIMIZATION RESULTS

MÉMOIRE PRÉSENTÉ
COMME EXIGENCE PARTIELLE DE LA
MAÎTRISE EN INGENIERIE

PAR
SHIMA SAMEALLAH

NOVEMBRE 2025

UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES
MAÎTRISE EN INGÉNIERIE (M. SC.)

Direction de recherche:

Vincent François directeur de recherche
Département de Génie Mécanique, Université du Québec à Trois-Rivières

---

Jean-Christophe Cuillière codirecteur de recherche
Département de Génie Mécanique, Université du Québec à Trois-Rivières

Jury d'évaluation

Vincent François directeur de recherche
Département de Génie Mécanique, Université du Québec à Trois-Rivières

---

Roland Maranzana Évaluateur externe
Département de Génie des Systèmes, École de Technologie Supérieure

---

Sasan Sattarpanah Karganroudi Évaluateur interne
Département de Génie Mécanique, Université du Québec à Trois-Rivières

Dedication

To my parents

To my dear sisters

To my dear friend Leila

To all my friends

Acknowledgements

I would first like to express my gratitude to my supervisors, Professors Vincent François and Jean-Christophe Cuillière for their quality and disciplined supervision, as well as their precious support and company during the most difficult period of my entire student life, a period that I was able to get through thanks to them.

I should thank UQTR for all services and supports that is provided during these years specially the exceptional chance that Adel Omar Dahmane, the dean of the time, gave me to reboot my academical path by accepting all passed courses in PhD duration as master's courses and the continuation of the exemption for the tuition fee like a PhD student.

I also thank the members of the jury, professors Roland Maranzana of École de Technologie Supérieure and Sasan Sattarpanah Karganroudi of Université du Québec à Trois-Rivières for the time they devoted to studying this work as well as for their constructive comments.

I express my deep gratitude to Professor Abolfazl Mohebbi for his help, friendship, and support during my period of remote work.

Finally, I would like to thank my dear sisters for their patience, their unwavering support, and their constant encouragement throughout my years of study. Their presence in my life has been a precious gift from God, and I am infinitely grateful to always be able to count on them.

Résumé

L’optimisation topologique (OT) est une technique permettant de déterminer la répartition optimale des matériaux lorsque l’espace de conception et les contraintes du problème sont définis. Les progrès récents dans le domaine de la fabrication additive ont rendu possible la reconstruction de ce résultat optimal. Cependant, son utilisation à des fins de production de masse pose plusieurs problèmes, notamment son coût élevé. La reconstruction des résultats de l’OT étant un objectif général, elle peut être divisée en plusieurs catégories en fonction de la nature du résultat de l’OT. Une stratégie alternative est de reconstruire les résultats de l’OT en géométries simplifiées, comme les structures de poutres, compatibles avec les méthodes de fabrication traditionnelles. L’objectif est de préserver les performances optimales tout en diminuant les coûts et en facilitant une production à grande échelle.

Parmi les différentes formes de résultats d’OT, ce travail se concentre sur les structures de poutre issues de l’OT, car elles sont simples à définir par leur axe central et leur section transversale et peuvent être reconstruites à l’aide de composants standardisés, rendant la transition vers une production de masse plus accessible. L’objectif principal est donc de développer une méthode automatique et précise pour extraire les informations essentielles de ces structures : les axes centraux des poutres et leurs sections transversales, afin de générer un modèle reconstructible.

Pour extraire ces axes, une stratégie est l’extraction de squelettes qui réduit une géométrie 3D en un ensemble de lignes fines, les centres des éléments structuraux. Cette méthode convient aux résultats d’OT de type poutre, mais nécessite de prendre en compte plusieurs aspects techniques pour garantir des résultats fiables. Un aspect clé est le centrage, car l’axe de la poutre se trouve en son centre. De plus, tout décalage pourrait altérer les propriétés mécaniques de la structure reconstruite, comme sa rigidité ou sa résistance. Le centrage est donc une propriété importante dans le choix ou le développement d’une méthode d’extraction de squelettes.

Dans cette thèse, une méthode d’extraction de squelettes bien centrée est développée, adaptée aux résultats lissés d’OT de type poutre. Dans cette méthode, une boîte d’identification est utilisée pour parcourir itérativement la structure. Elle identifie une branche qui est une portion de poutre, depuis son extrémité jusqu’à un point de jonction, en découpant la géométrie en sections transversales successives. Chaque section génère un point squelettique, et cette identification se poursuit d’une branche à l’autre, aboutissant à un squelette complet quand toutes les branches de résultat d’OT lissé ont été parcourues. Pour valider cette approche, deux méthodes d’extraction de squelettes existantes et reconnues pour leur bon centrage sont choisies. Cette validation n’est pas seulement qualitative, comme c’est le cas pour la plupart des méthodes d’extraction de squelettes, mais aussi quantitative. Un critère est sélectionné pour mesurer le centrage dans la partie validation quantitative, alors que les critères conçus à cette fin sont peu nombreux dans la littérature. De plus, une attention particulière est portée aux cas où le squelette pourrait sortir partiellement de l’objet original qui est un scénario sous-étudié, en expliquant en détail comment adapter le critère pour ces situations.

Le processus global de la méthode permet non seulement d’obtenir les axes des poutres, mais aussi d’estimer la section transversale en chaque point, facilitant ainsi la reconstruction complète de la structure. Plusieurs études de cas sont présentées pour illustrer l’efficacité de la méthode développée en ce qui concerne la précision, la robustesse et l’applicabilité. Parallèlement, les effets de paramètres influents, tels que la contraction des mailles et la taille des éléments de maillage, sont analysés. En fin de compte, cette thèse propose une méthode robuste et automatique d’extraction de squelettes bien centrés qui facilite l’utilisation des résultats d’OT de type poutre dans les processus de reconstruction pour la production industrielle, tout en introduisant des outils d’évaluation numérique rigoureux pour les méthodes d’extraction de squelettes.

Abstract

Topology optimization (TO) is a technique to find the optimal material distribution when the design space and problem constraints are defined. Recent advancements in additive manufacturing have made it feasible to reconstruct this optimal result. At the same time, there are several problems with its use for mass production targets, including its high cost. Since reconstructing the TO results is a broad objective, it can be broken down into several categories based on the nature of the TO result. Therefore, this work focuses on the beam-like TO results, as the beam structures are simple to define. In this regard, the skeletonization strategy is applicable to obtain the beam axes, but with some considerations. One of these considerations is centeredness, as the beam axis is at its center. So, centeredness is a prominent property in selecting or developing a skeletonization method. In this thesis, a well-centered skeletonization method is developed to skeletonize the beam-like TO results. In this method, an identifier box is used to identify a branch up to the junction point section by section, while the box makes the sectioning. Each section generates one skeletal point, and this identification continues from one branch to the other to discover the smoothed TO result. Two well-centered skeletonization methods are opted to validate the centeredness of the developed method. This validation is performed not only qualitatively, as is the case for most skeletonization methods, but also quantitatively. A criterion is selected to measure centeredness in the quantitative validation part, while the designed criteria for this purpose are limited in number in the literature. Moreover, the case of being out of the object is not well investigated in them. The implementation of the selected numerical criterion for this case is thoroughly explained. The overall process of the developed skeletonization method and its centeredness evaluation make it possible to have the beam structures' axes and the beam cross-section at each point of these axes. Several case studies are presented to illustrate the efficiency of the developed method. Meanwhile, the effects of contraction and mesh size are studied.

Table of contents

List of tables

List of figures

List of abbreviations

AESO	Additive Evolutionary Structural Optimization
ASOS	Automated Structural Optimization System
BCs	Boundary Conditions
BESO	Bidirectional Evolutionary Structural Optimization
B-Rep	Boundary Representation
CAD	Computer Aided Design
CS	Curve Skeleton
CSG	Constructive Solid Geometry
ESO	Evolutionary Structural Optimization
FE	Finite Element
FEA	Finite Element Analyses
FEM	Finite Element Methods
IASOS	Improved Automated Structural Optimization System
IGA	Iso-Geometric Analysis
LSM	Level Set Method
MCF	Mean Curvature Flow
NURBS	Non-Uniform Rational B-Splines
RAMP	Rational Approximation of Material Properties
ROSA	ROtational Symmetry Axis
SERA	Sequential Element Rejection and Admission
SIMP	Solid Isotropic Material with Penalization
TO	Topology Optimization
UTM	Unified Topological Model

List of symbols

α_0	The largest angle corresponding to the intersection point's ray, which is less than θ
α	The smallest angle corresponding to the intersection point's ray, which is more than θ
c	The identifier box center
C_i	An intersection point on the perpendicular plane on O_m and the point cloud
C_j	An intersection point which built a line with O_m and C_i
d_g	Global average element size for mesh generation
d_j^i	The distance from the i^{th} skeletal point to the shape boundary along the j^{th} ray
D_{ij}	A scalar distance field to color the distance of voxels in i^{th} skeleton from the voxels of j^{th} skeleton
d_{max}	Minimum length scale
d_{min}	Maximum length scale
DT_S	Euclidean distance transform of the skeleton S
E	Actual elastic modulus of the material
E_{center}	The off-center amount for a pair of rays
f	Volume fraction
lb	Half of the initial identifier box (cube) length
LC_i	The centeredness degree about point C_i
l_{min}	Shorter distance from O_m to C_i or C_j
l_{max}	Longer distance from O_m to C_i or C_j
$\lambda_{2\mathrm{max}}$	Maximum Lagrange multiplier of minimum length scale constraint
m	The number of skeletal points or compactness
n	Number of the nodes enclosed by an identifier box
O_m	A skeletal point
p	Penalization parameter
q	Penalization parameter

r	The radius of the bounding sphere of the shape
r_{minc}	The determined value for filtering the compliance sensitivity
r_{mind}	The determined value for filtering the relative density distribution
ρ	Relative density
$\rho(x,y,z)$	Relative density field
ρ_{th}	Threshold of the relative density
ρ_{void}	Relative density amount that replaces zero
S_i	The i^{th} (curve or surface) Skeleton
skl	A skeletal point
θ	The angle of the generating ray
x_m	Positive direction of x-axis
x_{min}	One face of the identifier box limiting its minimum values in x direction
x_{max}	One face of the identifier box limiting its maximum values in x direction
x_p	Negative direction of x-axis
y_m	Positive direction of y-axis
y_{min}	One face of the identifier box limiting its minimum values in y direction
y_{max}	One face of the identifier box limiting its maximum values in y direction
y_p	Negative direction of y-axis
z_m	Positive direction of z-axis
z_{min}	One face of the identifier box limiting its minimum values in z direction
z_{max}	One face of the identifier box limiting its maximum values in z direction
z_p	Negative direction of z-axis

CHAPTER 1 – INTRODUCTION

1.1. Background

In this technological world, every industry intends to present an optimized product with the best performance to avoid source depletion. In other words, products tend to be strong, lightweight, reliable, durable, efficient, and innovative. Thanks to automatic optimization tools, obtaining an optimized result for a specific application is no longer a dream, and it is also less time-consuming compared to the tedious design-analyze cycle that designers used to go through to reach a design close to the optimal result. In this regard, three main optimization categories are categorized depending on the objective set for the optimization. These categories are classified according to the optimization freedom, such as size, shape, and topology optimization, as illustrated in Figure 1.1. In size optimization, the shape and topology of the initial model are fixed, and user-defined dimensional parameters will be optimized. In shape optimization, the topology of the initial model is fixed, and the user-defined boundary geometries are optimized to achieve the desired dimension and shape. Finally, in topology optimization, the initial model's size, shape, and topology would be optimized to meet the most general set of objectives. TO is the optimal material distribution finding under applied loads and boundary conditions. It is noteworthy that the industry demands the optimal product.

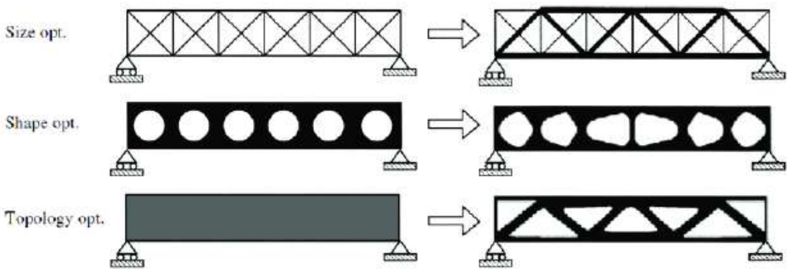


Figure 1.1- Size, shape, and topology optimization [1]

Besides opting for an optimization method in accordance with the objective set, there are other challenges to overcome to achieve the optimized product. In the first place, this optimized result is not in machine language, which means it is not manufacturable. It should be interpreted as a computer-aided design (CAD) model that is understandable to machines. However, the industry still needs to manufacture the optimized result at a mass-production scale and a low cost; satisfying this necessity is the ultimate goal. For this purpose, it should be manufacturable by one of the conventional manufacturing methods, including casting, forging, machining, and welding. Therefore, it will add other parameters to consider in CAD model reconstruction, in addition to those needed for additive manufacturing.

One of the simplest solutions coming to mind is to offer application-oriented solutions. Thus, the potential results, which can be beam-like structures, are the subjects of this study. For this specific subject, the neutral axes of the beams and the information about the sections should be sufficient to reconstruct the optimized result. To obtain the neutral axes of beams, skeletonization is a good aid. There are plenty of skeletonization methods, each targeting specific features of the skeleton. However, the reconstruction is not the sole application of the skeletons, and there are many other applications due to their capability in shape and boundary representation, such as animation, retrieval, matching, morphing, segmentation, and navigation.

1.2. Problem statement

This research focuses specifically on beam-like structures, where skeletonization methods become essential. By simplifying the reconstruction of complex topological optimization (TO) results, skeletonization helps extract the core structure, the beam axes, making it easier to work with these designs when they resemble beams. This simplification facilitates the creation of CAD models and ensures that the final design is manufacturable while maintaining optimized performance characteristics. Since this skeleton with beam cross-section information is part of a reconstruction process, it should be a well-centered curve skeleton. Researchers generated curve skeletons for diverse applications. Some of them are definitely automatic and well-centered, and a few can be used to generate a curve skeleton from a beam-like TO result. The main problem addressed in this research is the development of a robust, automatic, and well-centered curve skeleton from a beam-like TO result.

The most prominent obstacle to developing and assessing the desired skeleton is the lack of a precise definition for a curve skeleton and some of its features, like centeredness. That is why the qualitative comparison is standard in curve skeleton literature, and only a few researchers have done a quantitative assessment. When it comes to centeredness, there is a simple definition: a skeleton point should be located at an equal distance from at least two points of surface shape. However, in practice, it is neither easy to implement nor to assess. Thereafter, most researchers only limited the skeleton to be inside the initial shape.

1.3. Objectives

The primary objective of this project is to develop an automatic, well-centered skeletonization method that can transform raw 3D beam-like TO results into usable CAD models. To achieve this, we have set the following specific objectives:

Develop an Automatic Skeletonization Method: Design a process that automatically extracts skeletons from beam-like TO results, serving as the initial step toward converting them into accurate CAD models.

Validate the Method Visually and Numerically: Test and compare our new method against existing well-centered approaches through both visual assessments and numerical analyses. A suitable numerical framework should also be chosen to measure the centeredness of the skeletons, ensuring the effectiveness of the new method.

In this study, centeredness is a prominent feature of the skeleton, as it forms the axes of beams during the reconstruction phase. Therefore, if the skeleton is not well-centered or out of the object, the reconstructed model would also be out of the object. The centeredness is demonstrated in Figure 1.2, where the red skeleton went out of the object in the left sample, whereas the skeleton is visibly well-centered within the object in the right sample.

Figure 1.2- The centeredness of the skeleton [2]. Left) the red skeleton out of the object. Right) visually centered skeleton.

1.4. Methodology

To achieve these objectives, the methodology of this study is divided into four key points:

1. **Selecting a TO Method**: The study employs the Solid Isotropic Material with Penalization (SIMP) method as the TO approach, aiming to produce beam-like TO results. The optimization parameters of SIMP are fine-tuned to yield beam-like outputs, with adjustments made based on the specific characteristics of each example. However, the list of effective parameters remains consistent across all cases.
2. **Developing an Automatic Skeletonization Method**: This method involves scanning the object in successive steps and generating a skeletal point at each step. The cumulative set of these points forms the complete skeleton of the object.
3. **Establishing a Comparison Criterion for Skeletons**: Given the importance of centeredness and the precision offered by statistical measures, a numerical comparison criterion is adopted to evaluate the skeletons. While visual comparison is commonly used, numerical analysis provides a more objective and accurate evaluation.
4. **Selecting Additional Skeletonization Methods for Comparison**: For validation purposes, two well-centered skeletonization methods are chosen to benchmark against the newly developed method. This comparative analysis ensures the effectiveness and accuracy of the new approach.

This structured methodology ensures a comprehensive and systematic approach to achieving the research objectives, with a focus on developing and validating an automatic skeletonization method for beam-like TO results.

1.5. Structure of the present work

This chapter sets the stages for the work presented, offering a clear outline of the context, the core challenges, the objectives we aim to achieve, and the methodology we utilized. Chapter two, which goes into a thorough literature review, covering four central areas: TO, the interpretation of TO outcomes, curve skeletonization, and comparison tools for curve skeletons. Throughout this review, existing methods in each field are carefully categorized and critically analyzed, with attention given to their strengths and weaknesses. Chapter three introduces a new approach for converting TO results into skeletal CAD models. To help clarify this process, four case studies are employed, walking through the skeletonization procedure.

Chapter four discusses the choice of two skeletonization methods and a numerical criterion for evaluating centeredness. Then, we present the skeletons generated by each method across the case studies. These skeletons undergo both visual and numerical comparisons to assess the effectiveness of our proposed skeletonization technique. The chapter ends by explaining how the skeletons are evaluated.

In the final chapter, we discuss potential improvements to the skeletonization method and share some ideas for future research that can build upon this work.

CHAPTER 2 – LITERATURE REVIEW

2.1. Introduction

This chapter gives an overview of the literature review conducted for this project. The work structure is organized into three main fields, each of which is discussed in a dedicated section. The first section (2.2) focuses on TO methods, outlining popular categories along with their general advantages and limitations. Additionally, it explores manufacturing-oriented methods that harmonize with this research's focus. The second section (2.3) addresses techniques for interpreting TO results, with a review of several existing classes. The discussion then narrows to the specific orientation of this work: TO results that tend to form beam-like structures. The third section (2.4) introduces various curve skeletonization classifications, followed by an examination of key features of curve skeletons, which serve as the assessing parameters for comparing skeletonization methods. Finally, related comparisons are briefly discussed.

2.2. Topology optimization methods

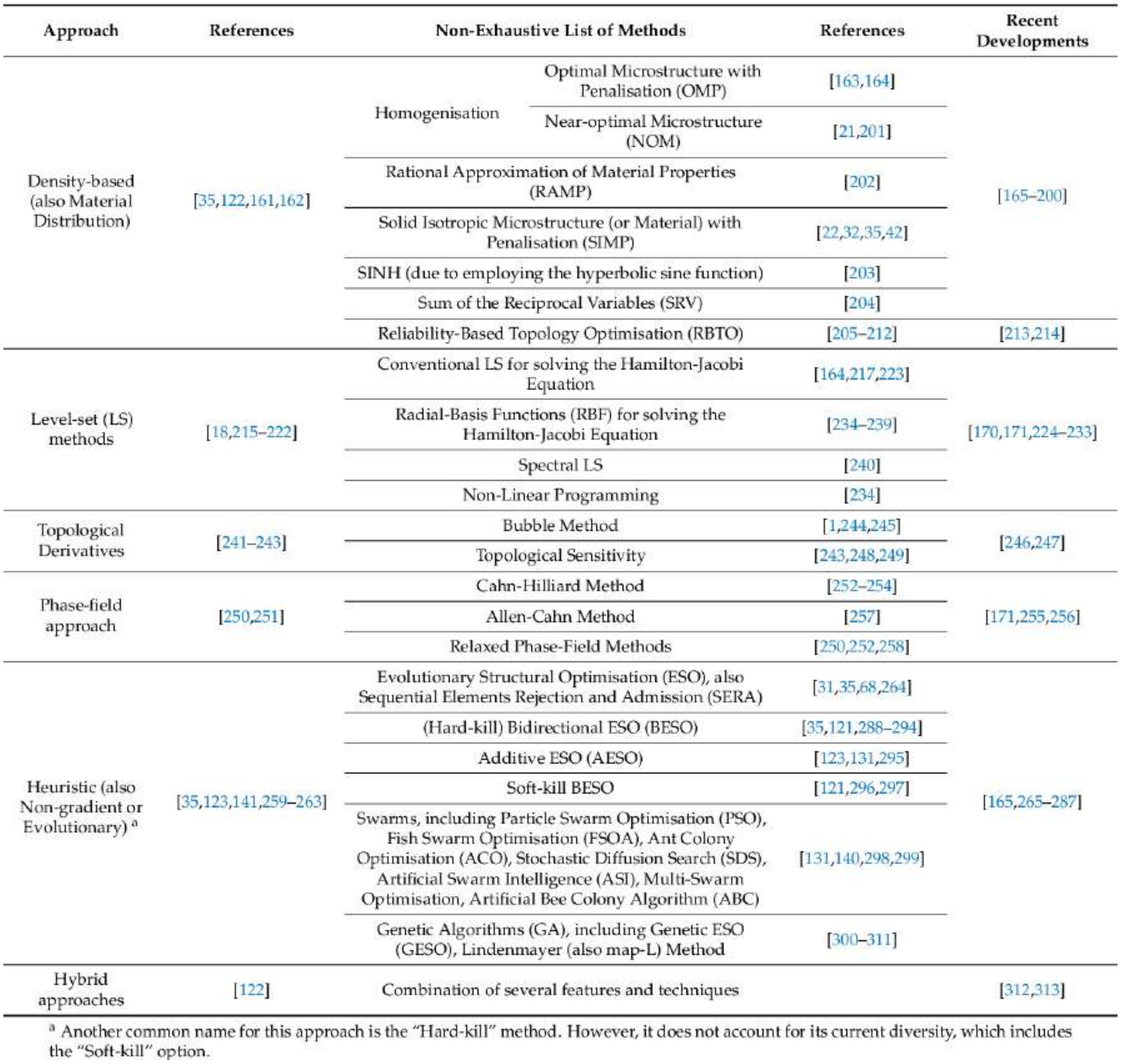
Topology optimization is an iterative process that aims to converge toward an optimal layout within a defined design space under specified loading conditions. The design space represents the volume where the final design must reside. Finite element methods (FEM) [3] are commonly used to perform the optimization calculations at each iteration. The objective of the optimization process is typically to minimize compliance, often guided by constraints on volume fraction. Unlike size and shape optimization, TO simultaneously provides designers with more freedom in terms of size, dimensions, and shape optimization. In recent decades, topology optimization has provided designers with greater freedom in many fields, including structural engineering. This section enumerates some popular TO classes: density-based, evolutionary, and boundary variation, while presenting some well-known methods for each class. As the approach of this work, this section is closed by presenting several manufacturing-oriented methods.

It has always been challenging to categorize TO methods into distinct frameworks, as evident in the discrepancies found in various review articles [4-6]. These articles group methods based on differing factors such as applicability and whether the TO is discretized or continuum-based. This challenge has only intensified with the increasing number of newly published papers on TO, some of which introduce new methods. In contrast, others combine existing methods to leverage their respective advantages.

In the structural domain, applicable TO approaches can be broadly subdivided into three branches:

- density-based methods (including homogenization and SIMP)
- evolutionary methods (like Evolutionary Structural Optimization (ESO))
- boundary variation methods (including level-set and phase-field approaches)

As shown in Figure 2.1, other classes, such as hybrid and biologically inspired methods [7], are not considered in this overview. The following subsections briefly describe each category, highlighting its advancements, advantages, and limitations.



Approach	References	Non-Exhaustive List of Methods		References	Recent Developments
Density-based (also Material Distribution)	[35,122,161,162]	Homogenisation	Optimal Microstructure with Penalisation (OMP)	[163,164]	[165–200]
			Near-optimal Microstructure (NOM)	[21,201]	
		Rational Approximation of Material Properties (RAMP)		[202]	
		Solid Isotropic Microstructure (or Material) with Penalisation (SIMP)		[22,32,35,42]	
		SINH (due to employing the hyperbolic sine function)		[203]	
		Sum of the Reciprocal Variables (SRV)		[204]	
		Reliability-Based Topology Optimisation (RBTO)		[205–212]	[213,214]
Level-set (LS) methods	[18,215–222]	Conventional LS for solving the Hamilton-Jacobi Equation		[164,217,223]	[170,171,224–233]
		Radial-Basis Functions (RBF) for solving the Hamilton-Jacobi Equation		[234–239]	
		Spectral LS		[240]	
		Non-Linear Programming		[234]	
Topological Derivatives	[241–243]	Bubble Method		[1,244,245]	[246,247]
		Topological Sensitivity		[243,248,249]	
Phase-field approach	[250,251]	Cahn-Hilliard Method		[252–254]	[171,255,256]
		Allen-Cahn Method		[257]	
		Relaxed Phase-Field Methods		[250,252,258]	
Heuristic (also Non-gradient or Evolutionary) [a]	[35,123,141,259–263]	Evolutionary Structural Optimisation (ESO), also Sequential Elements Rejection and Admission (SERA)		[31,35,68,264]	[165,265–287]
		(Hard-kill) Bidirectional ESO (BESO)		[35,121,288–294]	
		Additive ESO (AESO)		[123,131,295]	
		Soft-kill BESO		[121,296,297]	
		Swarms, including Particle Swarm Optimisation (PSO), Fish Swarm Optimisation (FSOA), Ant Colony Optimisation (ACO), Stochastic Diffusion Search (SDS), Artificial Swarm Intelligence (ASI), Multi-Swarm Optimisation, Artificial Bee Colony Algorithm (ABC)		[131,140,298,299]	
		Genetic Algorithms (GA), including Genetic ESO (GESO), Lindenmayer (also map-L) Method		[300–311]	
Hybrid approaches	[122]	Combination of several features and techniques			[312,313]

[a] Another common name for this approach is the "Hard-kill" method. However, it does not account for its current diversity, which includes the "Soft-kill" option.

Figure 2.1- TO methods classification [8]. Numbers within the figure are as cited in the original article.

2.2.1. Density-based

The methods concerning material or density distribution are considered in this class. The primary concept behind these methods is the discretization of the design domain and the evaluation of element density, as there are two ultimate extremes of the density range: zero and one, corresponding to void and solid, respectively. Confirming other TO methods process, the objective (minimizing compliance) and boundary condition constraints should be respected thoroughly. However, it is not always solid or void, and the interface between these will produce intermediate densities, which introduces inaccuracy due to the checkerboard effect. This effect is shown in Figure 2.2a and eliminated in Figures 2.2b and c. The principal solution considered in this class of TO methods is the penalization law. After penalization, elements with densities less than a definite threshold and the checkerboard effect will be eliminated. Some prevalent methods in this group are homogenization, SIMP, Rational Approximation of Material Properties (RAMP), and SINH, described below.

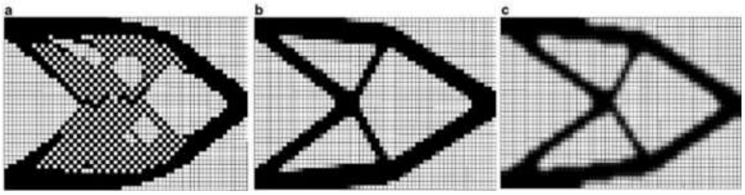


Figure 2.2- Topology plots. a) checkerboard effect, b, c) filtering the effect [9]

The homogenization method aims to set an infinite number of holes (voids) by utilizing porous microstructural materials (micro-perforated composites) to make a porous medium. Accordingly, TO with homogenization theory is the optimal porosity of the medium in the design domain when the absence of porosity (hole) is the solid presence, as reflected in Figure 2.3. The void and solid definition is the reason for being classified among density-based or material distribution-based methods. As the first TO method, the homogenization method was invented by Bendsøe and Kikuchi [10] and Suzuki and Kikuchi [11]. It is effectively applicable to compute mechanical properties.

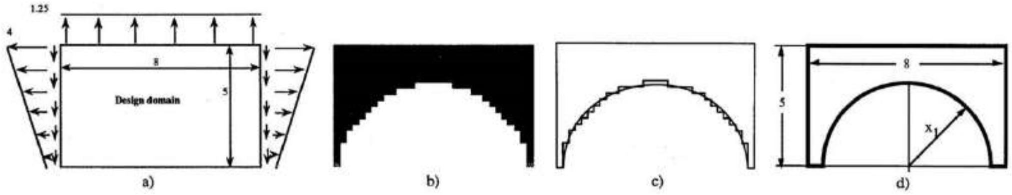


Figure 2.3- Homogenization theory example. a) design domain and boundary conditions. b) density distribution generated by homogenization. c) processed image. d) user interpretation and detailed design model. [12]

Although it has several benefits, including its simplicity in implementation and performance, it has some shortcomings: inaccuracy when the unit cell is larger than the macrostructure [13] and manufacturing difficulties due to infinitesimal pores generated through the TO process [14]. Figure 2.4 presents the definition of the unit cell.

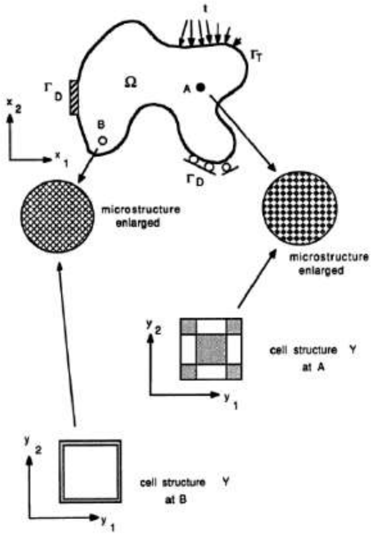
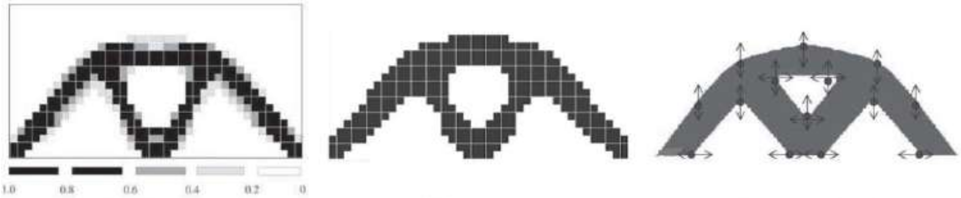


Figure 2.4- A structure with composite microstructure [10]

SIMP was independently developed by Bendsøe [15] and Zhou and Rozvany [16], but it was later named SIMP by Rozvany [17]. For the sake of simplicity, ease of implementation, and control over the size of structural features, it is widely accepted in both industry and academia; plentiful software is built on its foundation.

In SIMP methods, the design domain is discretized into finite elements with vivid densities as optimization variables. Meanwhile, a penalization parameter describes the relationship between density and Young's modulus. As the result is not binary (void or solid), intermediate densities in the binary interface are blurred and interpreted as grayscale or checkerboard patterns. The principal defect is numerical instabilities entailing the checkerboard effect, and mesh dependency from nonexistence as illustrated by Sigmund and Petersson [18]. Researchers' techniques to rectify this are perimeter control [19], sensitivity filter methods [20], restriction of gradients [21], density filter methods [22], user-defined length scale [23], and Shepard interpolation of the density field [24, 25].

To accede to the most prominent purpose, manufacture it, three post-treatment stages are inevitable [26]: topology design identification, structural boundary smoothing, and parametrization realization. Although the SIMP result is a mesh-based density field, the topology design identification and subsequent parametrization are performed by interpreting this field as a grayscale image and applying image-processing techniques, as illustrated in Figure 2.5.



(a) Grey level image (b) Binary map (c) Parameterized model

Figure 2.5- Parameterization through image processing [27]

RAMP was initiated by Stolpe and Svanberg [28]. One of its preponderances to SIMP is its nonzero sensitivity to zero density in the case of dependent loading. SINH (due to employing the hyperbolic sine function[7]) was schematized by Zhou and Rozvany [16], and a potential drawback is that intermediate density volumes after penalization inaccurately reflect the true volume. Figure 2.6 compares the intermediate density penalization functions for SIMP, RAMP, and SINH, where the density is ρ, the penalization parameter for SIMP and SINH is p and for RAMP is q. The evaluation metric is the penalized stiffness η(ρ); lower η at intermediate density indicates stronger suppression of gray elements. SIMP gives the steepest penalization, making it preferred for standard compliance density based TO problems due to simplicity and effectiveness.

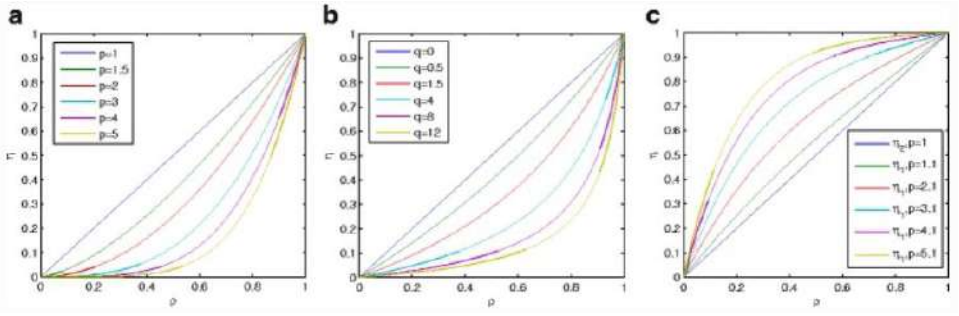


Figure 2.6- Comparison of a) SIMP b) RAMP c) SINH intermediate density penalization models [7]

2.2.2. Evolutionary

This branch of TO methods is named heuristic and hard-kill, as well as evolutionary, for several reasons. It includes these highlighted methods: ESO, Additive Evolutionary Structural Optimization (AESO), Bidirectional Evolutionary Structural Optimization (BESO), and Sequential Element Rejection and Admission (SERA). They are entitled to the hard-kill methods since they function by iteratively adding or removing finite materials from the design domain, and are denominated heuristic methods as the criteria for choosing materials to add or remove are heuristic, which may or may not be based on sensitivity information [7, 26].

Evolutionary methods were introduced by Xie and Steven [29, 30] with ESO to eliminate low-stressed materials and unify the structure in terms of stress. Volume fraction, *f*, adjustment and cross-section change are numerical regularization strategies to tune stress values after evaluation, as illustrated delicately in Figure 2.7 [31]. This method's disadvantage is its weakness in finding the final solution, as some eliminated material removed in the initial stages cannot be recovered. Thus, AESO [32] is introduced to add elements, and BESO [33] is introduced to add and remove elements. The alternative approach is to directly work on voids instead of solids, which involves soft-kill rather than hard-kill under the name of SERA [34].

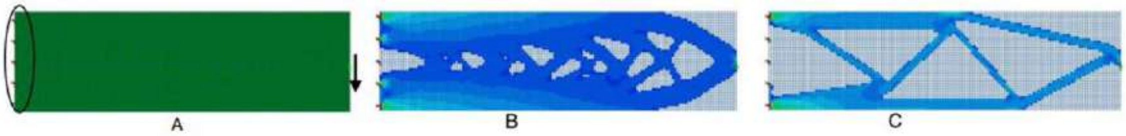


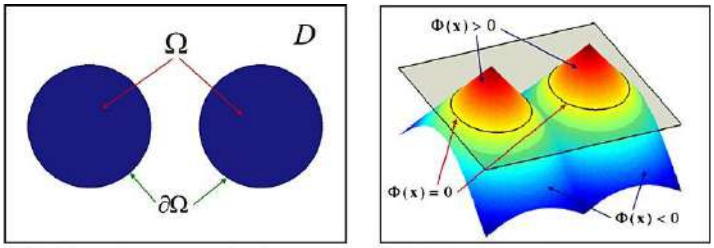
Figure 2.7- a) a finite element mesh of cantilever beam structural model with support and point load. b) Intermediate design during optimization process with $f=0.8$. c) Final optimization process with $f=0.35$. [31]

Its attractive features include simplicity, integration with Finite Element Analysis (FEA) solvers, freedom in selecting intermediate materials, and the capability to implement it in conjunction with another method only to benefit from its material obviation scheme [35]. However, among its most distinguished imperfections is the shortage of algorithm convergence and stopping criteria, which some researchers [36, 37] have attempted to address.

2.2.3. Boundary variation

On the contrary, boundary variation methods represent the boundary surface of the structure using high-dimensional implicit functions rather than utilizing explicit parameterization of the design domain. Since the computed result is crisp, it should undergo post-processing before interpretation. The level-set and phase-field methods are two well-known subcategories of this category.

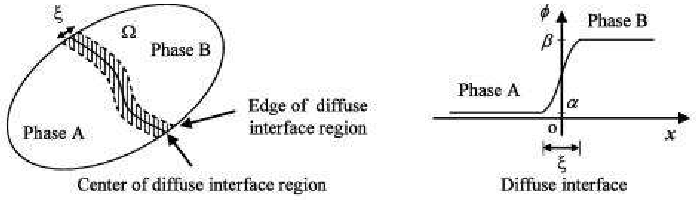
The level set method (LSM) was initially introduced by Osher and Sethian [38] and was preliminarily applied in structural topology optimization by Sethian and Wiegmann [39]. In this method, boundaries are defined as zero-level curves or contours of a level set function, which is the distance from the boundaries. However, the material domain is defined as a positive level, as demonstrated in Figure 2.8. Without voids inside the design domain, shape derivatives will cause shape deformation; thus, topology changes only stem from interior holes or topological derivatives. In this regard, specifying holes, boundary motion, and boundary merging are accomplished via a level-set function. The boundary shape can be modified by controlling the level-set motion in relation to the physical problem and optimization condition. Moreover, the detection of separated regions from boundary conditions and, consequently, computation is possible by this method.



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

Figure 2.8- Implicit level set representation at zero level set [40]

The phase-field method in topology optimization was first proposed by Bourdin and Chambolle [41], based on the surface dynamics representation of phase transitions in materials science, such as crack propagation or multiphase flow. Here, a phase-field function is specified over the design domain, and the boundary between phases ascertains structural boundaries (between void and solid), as illustrated in Figure 2.9. The most eminent of this method, declared in all relevant papers, is the redundant material phases interface definition. In comparison with level set methods, this interface is not tracked throughout the optimization. Thus, topology changes occur naturally during the optimization process, as shown in Figure 2.10. It is noteworthy that the computational cost of this method in structural topology optimization is considerably higher than that of level-set methods, which is why LSMs are more accepted and popular in this field.



(a) A 2D domain represented by the phase field function

(b) A 1D illustration of the phase field function

Figure 2.9- Examples of the phase-field function [42]

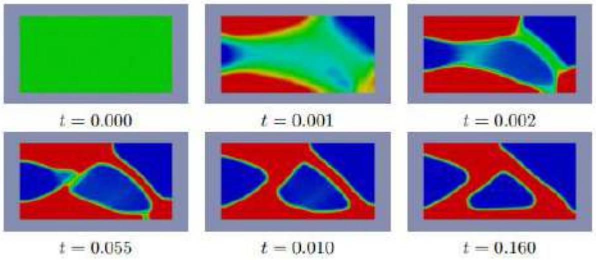


Figure 2.10- Allen-Cahn results for a cantilever beam computation at various times [43]

2.2.4. Manufacturing-oriented

Since topology optimization intends to optimize the design, increasing the manufacturability of TO results has aroused researchers' curiosity over the last two decades. To satisfy this desire, they devised several approaches, mainly in two canals. One is along the TO process, which can be rendered as direct machinable TO, and the other is within the interpretation phase. The direct machinable TO methods will be investigated concisely in the following section, while the other canal studies are discussed in the interpretation section. Furthermore, the TO algorithm should also note some manufacturing considerations to make the TO result come true. Among these, we can enumerate the minimum dimension of the part, the minimum size of the hole, the smoothness and flatness of the surfaces, the dimensional precision of the machine, the size of machining tools and more. Above all, most attempts are directed at preparing results for additive manufacturing rather than conventional manufacturing methods, since it has fewer limitations and considerations to follow.

In the third section of their paper, Liu and Ma did a valuable job [26] by remarking on two nontrivial aspects, length scale control and geometric feature-based design, concerning machinable or manufacturing-oriented TO methods.

In the length scale control, the void size will be checked to be bigger than the minimum cutter size to guarantee the result's manufacturability. This constraint is initially exerted in the TO process through filtering [20] and a local gradient constraint [21]. Additionally, Sigmund developed a morphology-based filtering technique described in Figure 2.11.

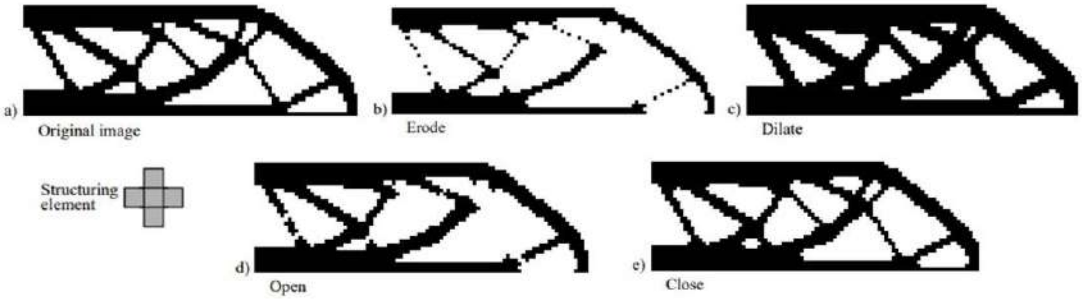


Figure 2.11- Demonstration of basic image morphology operators [44]

To address some recent attempts in this regard, Gao et al. [45] proposed a TO method in a LSM framework with length scale control. They defined a skeleton feature concept as a circle whose center is the skeletal point, and its radius is the length scale, as illustrated in Figure 2.12a. This length scale has an increasing Lagrange multiplier constraint with the maximum value of $\lambda_{2\max}$ and minimum-maximum limits of d_{min} and d_{max}. A signed distance function based on the skeleton feature is applied to obtain these minimum and maximum length scales. All these definitions are demonstrated in Figure 2.12.

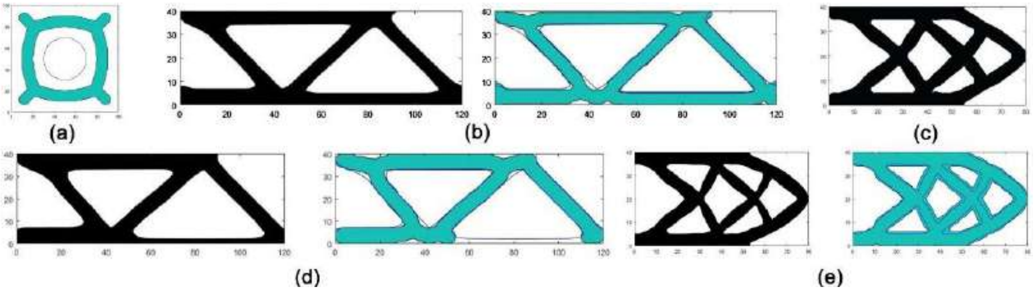


Figure 2.12- Resultant structures subjected to different length scale controls. a) skeleton feature based on a skeleton of a simple structure. In the rests, blacks are structure optimization results subjected to the specified parameters control and greens are corresponding skeleton features: b) λ_{2max}. c) d_{min} and d_{max}. d) d_{min}. e) d_{max}. [45]

Liu [46] actualized the piecewise length scale control for LSM, and he benefited from the image processing for the structural skeleton identification and segmentation and assumed unequal length scales for segments in necessity. So, the segmented skeleton (Figure 2.13b) is solicited from the structural topology (Figure 2.13a), then the design domain is decomposed (Figure 2.13c). The optimization and consequent decomposition results for a three-target length scale are shown in Figure 2.13d since the control is dynamic between segments. As an example, the level set contour and its structural decomposition are picturized in Figure 2.13e for a modified design domain.

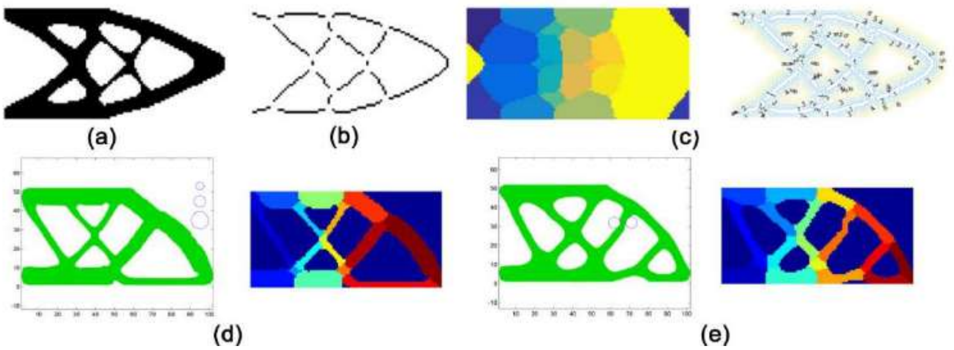


Figure 2.13- Piecewise length scale control. a) structural topology, b) segmented skeleton, c) design domain decomposition, d) TO and structural decomposition for MBB structure with three length scale targets, e) level set contour and structural decomposition for a modified design domain. [45]

In a Non-Uniform Rational B-Splines (NURBS)-based SIMP method [47], the NURBS entity's (knot vectors) properties can result in a minimum length scale forecast plus no compulsion to add further optimization constraints. The minimum length scale is independent of the mesh size but controllable in Finite Element (FE) or any reassembled geometry. Figure 2.14 reports the NURBS surface after threshold operation, the minimum length scale, and the effect of increasing the number of control points on the result.

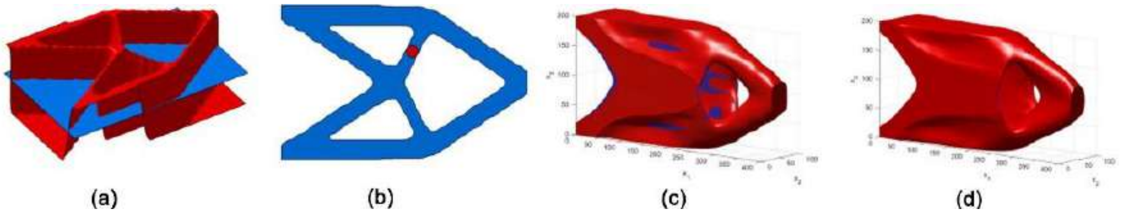


Figure 2.14- Minimum length scale determination and the effect of increasing control points. a) threshold operation on the NURBS surface, b) actual minimum member size in the 2D result, c) 30×12×18 control points, d) 36×18×24 control points. [47]

Concerning the second remarked aspect of manufacturing-oriented, Geometric feature-based design deals with design methods that deliberate manufacturing requirements from the early design stages. In the TO framework, SIMP and LSM have the potential to be grouped in this category in addition to their significant role in expanding manufacturing-oriented TO methods. Their potential stems from voids or solids' freedom of movement, scaling, and rotation as geometric features. Among works in this field, Chen et al. [48] parameterized level set functions to model geometric features implicitly. Figure 2.15 includes both parametric and free-form shapes, where the parametric shape is the circle, and the free-form shape is the outer shape represented by a linear B-spline.

In summary, Table 2.1 provides a comparative synthesis of the studied TO methods specifically evaluated for beam-like structure generation, including advantages, limitations, and suitability analysis.

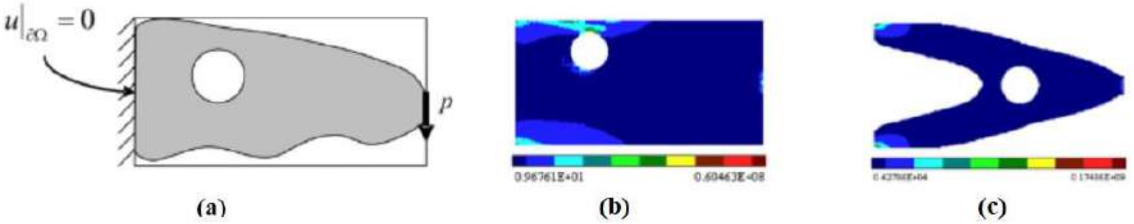


Figure 2.15- Strain energy distribution of the initial and optimal shapes. The design variables are the center of the circular hole and the B-spline coefficients. a) the optimization problem, b) the initial design, c) the optimal shape. [48]

Table 2.1- Comparative analysis of the discussed TO methods for beam-like structures.

Method category	Density-based	Evolutionary	Level-set	Phase-field
Output type	Density field	Discrete elements	Implicit boundaries	Phase-field function
Advantages	Simple implementation, Widely used in industry	Simple concept, FEA Integration	Crisp boundaries, Natural topology changes	Natural topology evolution, No boundary tracking
Limitations	Checkerboard effect, Grayscale intermediates, Requires post-processing	Convergence issues, Lack of stopping criteria	Requires initial holes, Computational complexity	High computational cost
Suitability	High	Medium	High	Medium
References	[9-27]	[29-37]	[38-40]	[41-43]

Recent manufacturing-oriented TO methods have focused on tighter integration between optimization and fabrication processes. Zou et al. [49] developed explicit self-supporting constraints for additive manufacturing within the SIMP framework, while Deng et al. [50]

proposed a method that simulates multi-axis machining within the optimization loop. Pan et al. [51] recently introduced an automated approach for converting TO results into parametric models compatible with three-axis CNC machining. Despite these advances, automated reconstruction of beam-like TO structures as CAD models remains challenging.

2.3. Interpretation of TO results

As mentioned in detail in the last section, a variety of TO methods give TO results that need post-processing to make them manufacturable, except those directly machinable in the 2.2.1.4 subsection. From an ambitious point of view, designers seek 3D CAD models as a TO result to be able to either fabricate or edit them. The result should be editable since some manufacturing constraints should be applied after the optimization process, and more importantly, it will be indispensable during the product development phases. Furthermore, manual CAD reconstruction is costly, tedious, and sometimes hard to implement, so automatic CAD reconstruction is the main target. The importance of the automatic reconstruction cycle is particularly evident in SIMP results. Its density-based results require geometry reconstruction, after which stress must be recalculated to confirm mechanical performance matches the optimized design, often requiring parameter adjustments and redoing optimization. While this validation loop applies to many topology optimization methods, it is especially critical for SIMP due to its reliance on density thresholding and geometry extraction. SIMP is among the common methods in software. The obtained results in each stage of this cycle for a bike’s suspension rocket with the SIMP method before validation are presented in Figure 2.16.

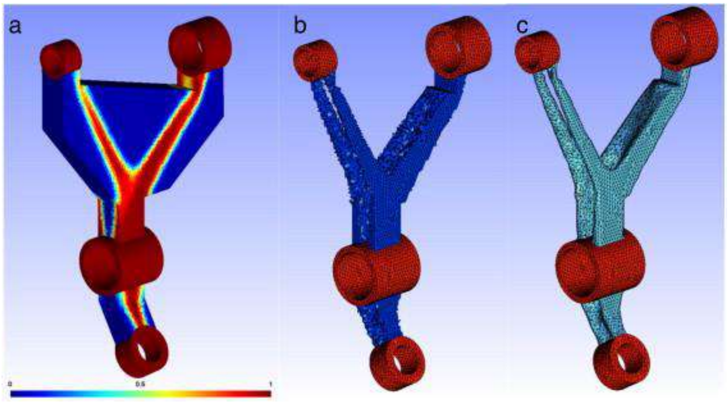


Figure 2.16- Raw SIMP result, rough and final shape. [52, 53]

A CAD model is responsible for modelling all the needed information of a product throughout its life cycle, comprising spatial and non-spatial information. Non-spatial information includes material, hardness, and other non-geometric attributes. However, a CAD model focuses primarily on spatial information, and spatial information modelling is sometimes called geometric modelling. In the following section, the development stages of CAD modelling, along with the distinct approaches of representing a model, are presented. Then, three categories of techniques for producing a CAD model from a TO result are counted, along with their privileges, deficits, and limitations. Afterward, some similar works investigating beam-like structures are scrutinized.

2.3.1. Geometric modelling techniques

The development stages of CAD modelling may be divided into three categories: Solid modelling, Parametric modelling, and Direct modelling. Indeed, it was ignited by wireframe modelling, extended by surface modelling, and completed by solid modelling to represent any possible geometric queries. Two approaches are introduced to represent a solid model: Constructive Solid Geometry (CSG) and Boundary Representation (B-Rep). CSG represents a solid as a structure tree with primitive shape leaves and Boolean operations' intermediate nodes. In contrast, B-Rep represents it by describing the boundary between the solid and non-solid. It would be more transparent in Figure 2.17.

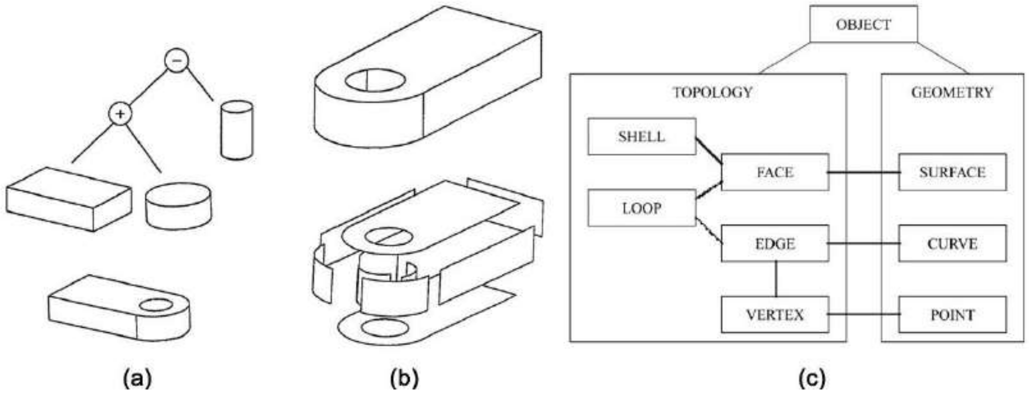


Figure 2.17- Representing solids with a) CSG and b) B-rep. c) basic elements of the data structure. [54]

In CSG, it is evident that more than one structure tree can be used to represent a solid, and this is addressed as one of this technique's drawbacks because it is infeasible to compare shapes by comparing their trees. On the other hand, it guarantees model validity thanks to using Boolean operations.

B-Rep models comprise topology (model's structure) and geometry (model's shape) parts. The relationship and equivalency between topological elements, such as faces, edges, and vertices, and geometric elements, including surfaces, curves, and points, are described in Figure 2.17c. In fact, topology elements rest on geometrical ones, each bound by the other. In other words, the face lies on the surface, bounded with edges where an edge lies on a curve and bounded with vertices. On the other hand, geometric elements carry mathematical information, so objects with different geometries can still share the same topology, like a cylinder and a cube.

These two approaches motivated the subsequent development stages of CAD modelling, including parametric modelling and direct modelling. Parametric modelling stems from CSG and results from a combination of associative modelling and solid modelling to achieve the goal of automatic change propagation. In this modelling, primitive shapes are replaced by features, which makes it easier to embed the associativity in geometric entities or shapes. Meanwhile, the structure tree in CSG evolves into a construction tree or construction history in parametric modelling. However, the construction history needs to be carefully considered, as it determines what can be edited in the model and will not be modified after finishing a design.

Conversely, direct modelling based on B-rep provides complete freedom for a user to edit or manipulate the boundary faces. The model has several benefits, such as flexible edits and fast updates, but it has some consequences, including losing associativity and parameter-driven modification. The complementary nature of these latter models brings the idea of integrating them to benefit from their advantages. To date, hybrid approaches have been introduced to integrate parametric and direct modelling within the same modeller [55]. This section aims to provide an overview of CAD model history and strategies, as a TO result needs to be converted into a CAD model.

2.3.2. Interpretation methods

Subsequently, indirect tactics of TO results transformation to CAD models are generally filed in three sets: multi-sectioning, geometric approximation, and reverse engineering, leaning on the chosen tactic to extract geometric information from the TO result. Each has its merits and restrictions referred to in the relevant part.

2.3.2.1. Multi-sectioning

Many approaches have been dedicated to forming a CAD model from TO results. Among them, iso-density contours are widely used in the literature. However, the idea behind iso-density contours is essential, as this group's technique involves sectioning the TO result and sweeping the sections in accordance with the direction of the branch. The availability of cross-sections makes it appropriate for layer-by-layer manufacturing methods, mainly additive manufacturing. In this regard, some works are reviewed hereunder.

Marsan and Dutta [56] developed a CAD model from the TO result done by the homogenization method in continuing Chirehdast's work [57]. The input is 3D rectangular prismatic volume elements, which could be considered like voxels accompanying density values. A set of planar contours are extracted from the volume and interpolated with parametric surfaces. Several processes are applied on each layer to remove grayscale, noise, and checkerboard effect, as illustrated in Figure 2.18a, to obtain Figure 2.18b. The only remaining step in having a 3D CAD model is shaping transition contours, as shown in Figure 2.18c. This method is general enough to suit complex structures with branches and holes; however, user intervention for defining density threshold value and degree of the parametric surface is inevitable. Moreover, it is not suitable for traditional manufacturing due to the complexity of structures designed by the homogenization method.

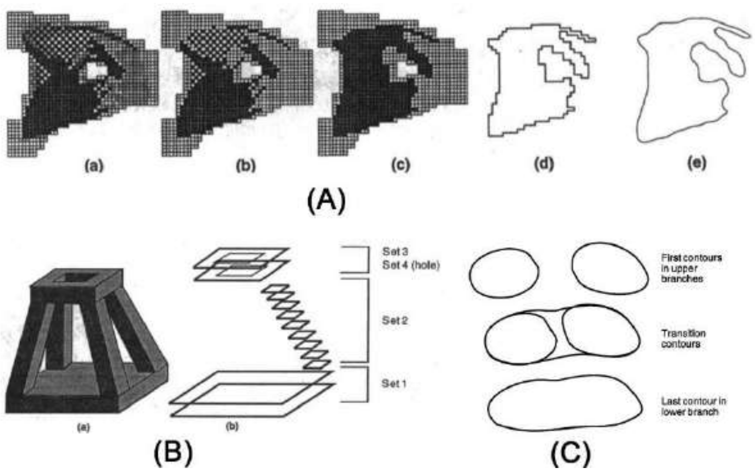


Figure 2.18- A) a layer of a 3D homogenized structure: a) unthresholded, b) thresholded, c) noise removal, d) extracted edges, e) smoothed edges. B) a) a branched 3D object, b) the object broken down into sets of contours. C) transition contours. [56]

Later, they [58] surmounted user intervention by implementing it in a computer environment and applied tiling or skinning strategies to connect planar contours with polygonal facets or parametric spline surfaces, as illustrated in Figure 2.19. This method is convenient for multi-branched objects with convex or non-convex contours which can vary remarkably from one to the other.

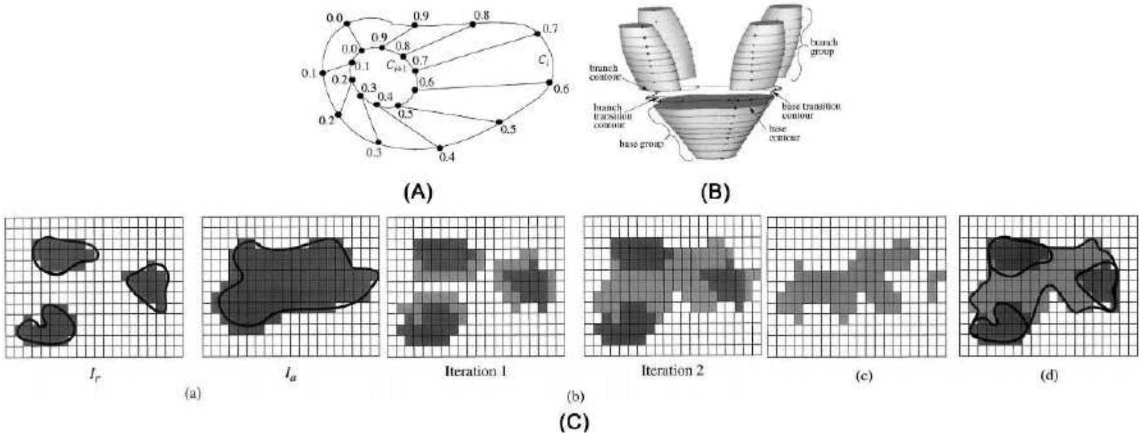


Figure 2.19- A) point correspondence between two contours determined by parameter value, B) terminology for transition algorithm, C) formation of transition contours. [58]

Tang and Chang [59] proposed an interpretation method not on iso-density contours extraction but on a voxel-based model. The B-spline parametric curves and surfaces helped to import the TO result into the parametric CAD model. The processing procedure for a tracked vehicle road arm can be followed in Figure 2.20. Therefore, CAD model can be obtained just by applying subtraction Boolean operation or union Boolean operation application. This method is feasible for structural optimization but has several deficiencies, including significant user interactions and decision-making, mesh distortion, and necessary treatments.

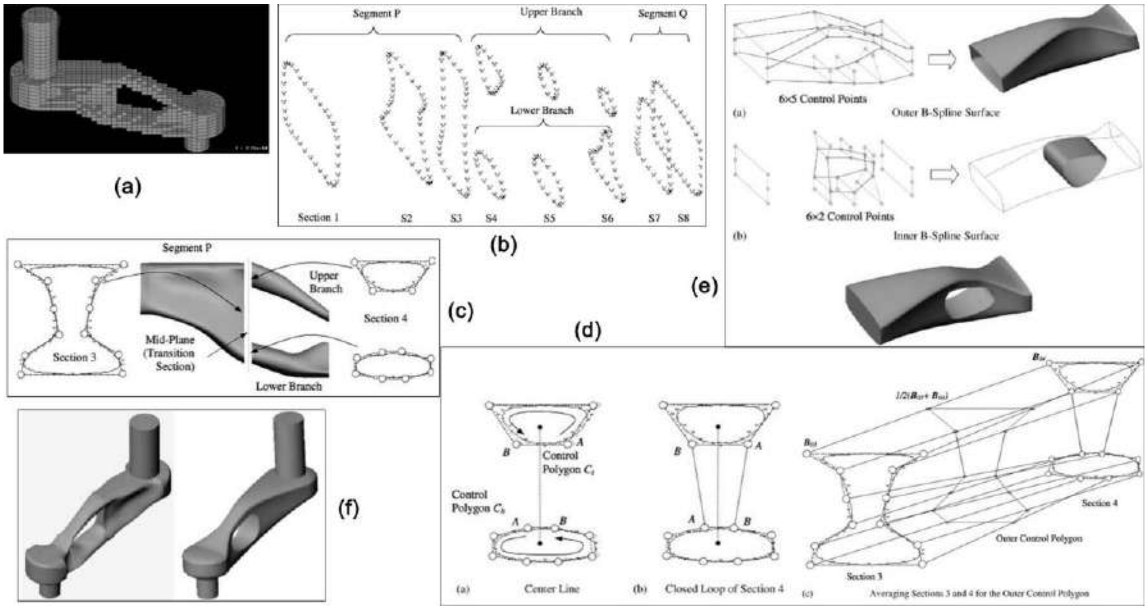


Figure 2.20- a) topologically optimized road arm, b) smoothed geometric points of cross-sections, c) transition between segment P and branches, d) construction of control polygons at the transition section, e) solid model reconstruction by subtraction Boolean, f) reconstructed solid models by Boolean union and subtraction operations. [59]

To prepare the TO result for a simple reconstruction implementation, Hsu and Hsu [60] utilized tuning and filtration before interpretation. These tuning and filtration steps are applied to omit trivial solids and voids as shown in Figure 2.21. After sectioning, the FE TO result is converted to a B-spline density contour on each section. Afterward, sweeping these sections provides the reconstructed model. Concerning this method, the finer mesh raises a more complex topology, and the checkerboard effect will nullify the effect of filtration. At the same time, applying higher-order FE solves the checkerboard problem.

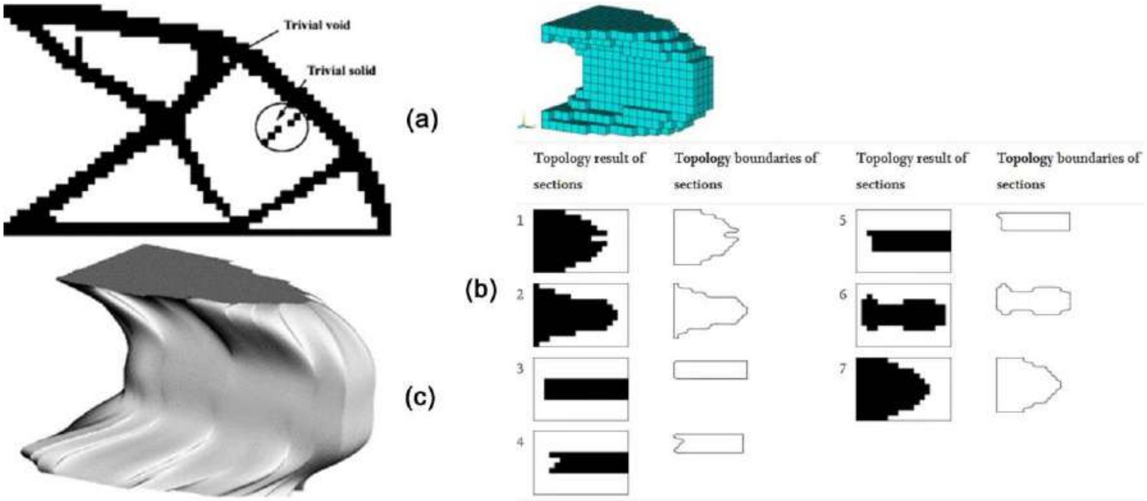


Figure 2.21- a) trivial solid and void in the cantilever, b) TO result and boundaries of the selected sections of the cantilever along y direction, c) reconstructed 3D model. [60]

Recently, Bender and Barari [61] mentioned that FE TO results can be directly sliced to give sections just like commercial software taking STL format TO results and providing triangulated surfaces for layer-by-layer fabrication, as demonstrated in Figure 2.22. These sorts of software benefit from algorithms like marching cubes. Therefore, it is possible to eliminate the redundant intermediate steps and construct a bridge between the TO result and additive manufacturable result, just like software. The result can be fed back to the TO process using this bridge, and the final product would be qualified.

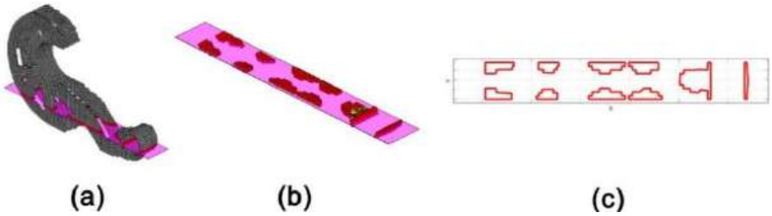


Figure 2.22- a) the elements lie on the slice plane, b) the filter set of the elements, c) the contours of the slice. [61]

2.3.2.2. Geometric approximation

This method of interpretation takes benefits from image processing to turn TO grey image into a binary (black and white) map. Then, it exploits trusses, shape templates, or B-splines, depending on the application, to smooth and parameterize these maps. The researchers employed various approaches to interpretation. Bremicker et al. [62] extracted a truss or a continuum structure from the binary map by skeletonization or boundary smoothing. Lin and Chao [27] interpreted the external boundary of the binary image by B-splines and the interior holes by shape templates like parametric spheres, cylinders, and rectangles. However, the number of templates is limited, and increasing the fitting accuracy is another challenge in itself. Meanwhile, Yildiz et al. [63] profited from the neural network-based image processing to identify hole features. Recently, Chou and Lin [64], Yi et al. [65], and Yi and Kim [66] identified geometrical features from topology, shape, and section-optimized results, which the first and last explained more hereinafter.

B-spline curves are employed to determine exterior boundaries and interior holes in this fully automated 2D method [64]. Simultaneously, the performance of this new proposed strategy improved the automated structural optimization system (IASOS) over the automated structural optimization system (ASOS) [27], as presented in Figure 2.23. This enhancement is obtained by improving hole shape accuracy, since IASOS applies the polygonal image-interpreting technique (with any polygonal shape) instead of the characteristic value-based one (with only four polygonal templates) in ASOS. Thanks to this method, the shapes of interior holes are more accurate, and the computational efficiency and power of volume diminution are improved.

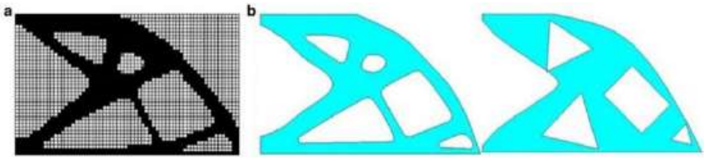


Figure 2.23- a) TO result, b) configuration after image interpretation (middle: IASOS, right: ASOS) [64]

The active contour method and optimization techniques represent the boundary of the TO result by basic geometric features [66]. Firstly, the geometric boundaries of the TO result are interpreted in the form of points defined by their coordinates, as shown in Figure 2.24. Then, section and shape optimization are executed on this result. It is note-taking that users should define the design variables and geometric constraints of optimization problems. On the one hand, benefiting from B-splines is possible, but conventional manufacturing finds it troublesome. On the other hand, basic geometric features such as lines, arcs, circles, and fillets require plain zero-one TO results without thin parts or isolated islands. They cannot describe complex geometries even in 2D, the subject of this study.

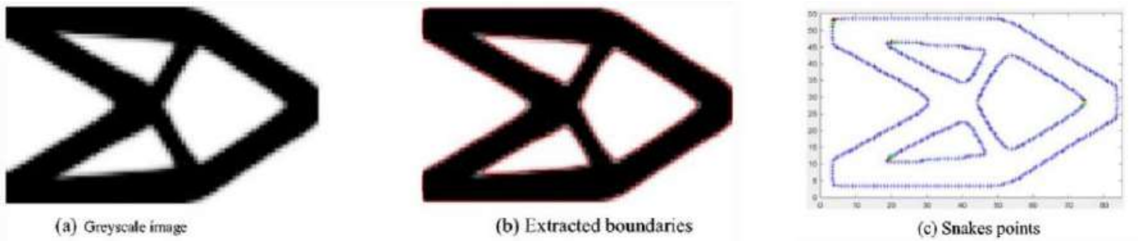

(a) Greyscale image (b) Extracted boundaries (c) Snakes points

Figure 2.24- Image interpretation by the active contour method. [66]

2.3.2.3. Reverse engineering

Reverse engineering or reverse modelling directly obtains a geometric model from a physical one. Such a procedure, model reconstruction, can be partitioned into surface-based and feature-based methods, with the former being more thorough and affiliated with commercial software like CATIA®. Accurate reconstruction enables reliable numerical analysis (FEA or iso-geometric analysis) of the model's performance. Iso-surface generation (like the marching cube algorithm) is the most important part of this method; for example, Koguchi and Kikuchi [67] used it to identify iso-density surfaces and model boundary segments by spline surface patches. Thus, this method is more accurate than the contour sweeping method, as the resultant precise spline surface boundaries are more accurate than approximated surfaces. Geometric reverse engineering attempts to reconstruct parametric shapes, like polygon meshes and volumetric geometries, from physical objects. A critical part of this attempt is mesh segmentation, which includes splitting boundaries into patches suitable for CAD modelling. Well-known segmentation methods include spectral analysis (graph-based partitioning), clustering (grouping similar elements), region growth (expanding from seed points), skeleton-based methods (using geometric skeletons to define patches), and machine learning (data-driven segmentation) applications. Several reverse engineering approaches are presented below.

With the low member size setting, TO leads to frame-like polygon meshes with slender parts and can be reconstructed as beams by extrusion of cross-sections along beamlines. Denk et al. [68] benefited from the homotopic thinning method to obtain curved skeletons as beamlines. They successively transformed them into a subdivision surface control grid thanks to Euclidean distance transformation, as demonstrated in Figure 2.25a. Junctions pose challenge, while the proposed solution is presented in Figures 2.25b and 2.25c.

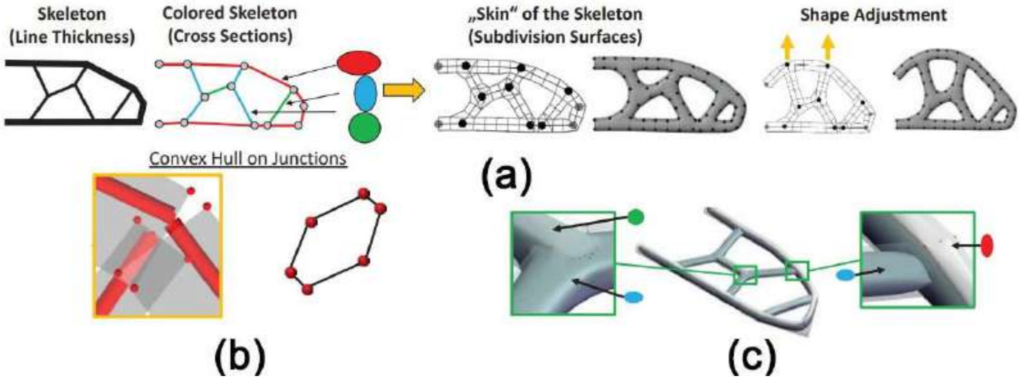


Figure 2.25- a) colored shape determination, b) the convex hull on the junction from the control grid created for beams, c) the connection of the different beams at junctions. [68]

It is feasible to profit from the physics of the problem rather than geometry to interpret geometric TO results by clustering process where the physics of the problem implies stress information of TO result [69]. The clustering process displayed in Figure 2.26a includes beam/linking identification via principal stresses, tension/compression division, histogram clustering, and linking zones. Consecutively, a parametric CAD model with cylinder extrusion profiles and spherical structural links is concluded, as shown in Figure 2.26b.

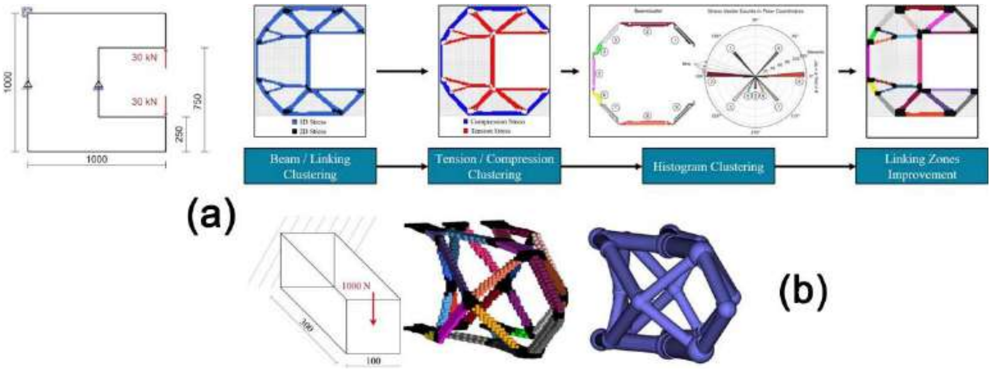


Figure 2.26- a) 2D design space and its complete clustering process, b) 3D cantilever beam: design space (left), clustering result (middle), parametric CAD model (right). [69]

Doutre et al. [53] separated reverse engineering approaches into two principal groups. First, designers manually produce CAD models from the polygonal model resulting from the optimization software (using standard CAD features in CATIA or poly NURBS technology in Evolve™), as illustrated in Figure 2.27a-c. Second, directly apply automatic surface or volume generation tools on the polygonal model (shrink wrapping approach in Magic's, developing-remeshing uniform density point cloud in MeshLab, or handling CATIA module in STL), as demonstrated in Figure 2.27d. In manual methods, designer can apply changes in the model, manufacturing integration, and post-reconstruction optimization. However, automatic methods limit parametric adjustments and complicate FEA due to numerous generated surfaces during reconstruction.

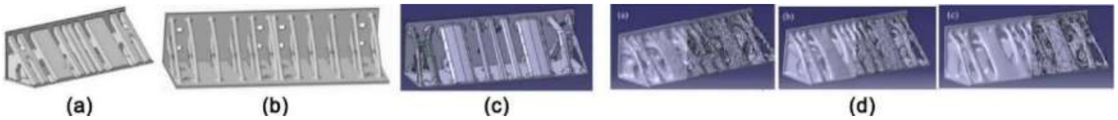


Figure 2.27- CAD models with: a) CATIA, b) CATIA generative shape design, c) Evolve 2015, d) different approaches. [53]

It is evident that the classification done previously in this section is not the only one, and it depends on the respect for categorization. In respect of dimension, reconstruction methods are divided into three groups [70]: 1D skeleton-based (well-suited for beam-like structures, limited in handling junctions), 2D surface-based (advanced via remeshing, subdivision, NURBS surface fitting; as demonstrated in Figures 2.28a-c), and 3D volume-based (formed from volume decomposition and Boolean operations, as pictured in Figure 2.28d). 2D surface-based is challenged in automation, gap stitching, and retaining sharp features. 3D volume-based is excellent in suppressing small features and retaining critical features, but it is unsuitable for complex TO models and automation.

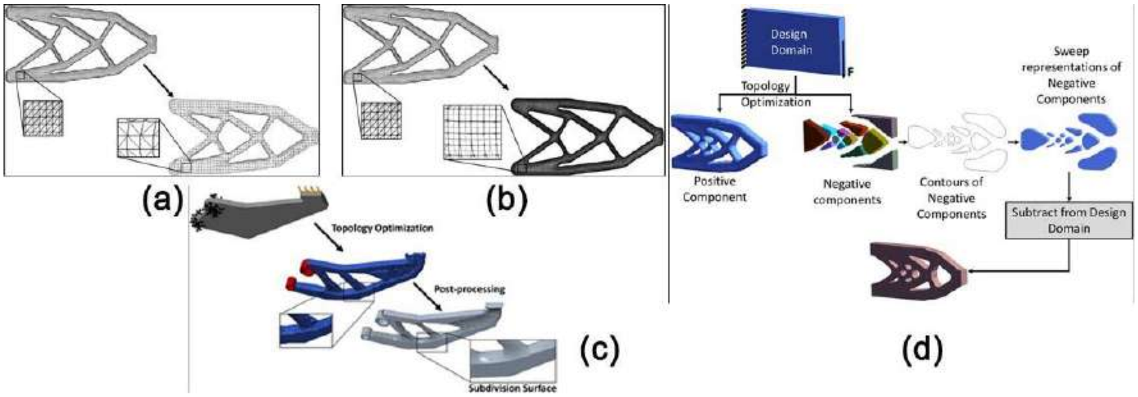


Figure 2.28- a) remeshing triangular meshes using screened Poisson surface reconstruction, b) NURBS surface fitting with control points, c) geometry reconstruction with subdivision, d) a cantilever TO problem with geometry reconstruction. [70]

Regarding surface-fitting, in recent research [51], a parametric sketch contours generation method is employed voxelization and accessibility to convert the TO result to a parametric model by removing geometries layer by layer for a three-axis CNC machining application. Several stiffness constraints are applied, and the redundant geometries are classified into 10 different categories. Remove geometries of categories 1 and 3 from the raw material for a specific example is shown in Figure 2.29.

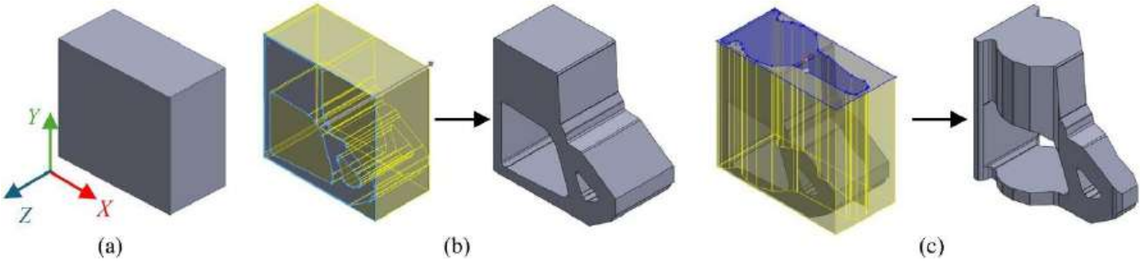


Figure 2.29- Removing categories 1 and 3 redundant geometries. a) the raw material, b) removing category 1, c) removing category 3 [51].

2.3.3. Beam-like structures

Beam-like structures always attract researchers' attention since one of three groups categorizing approaches for CAD model reconstruction from TO results is devoted to them, referring to the last part of the precedent section. A huge number of works focused on this kind of structure have been reviewed before [68, 69] and will be overviewed in this section specifically.

In a fully automated procedure, Nana et al. [71] reconstructed beam-like structures resulting from SIMP optimization, where stages constitute noise removal, smoothing, skeletonization, normalization and circular cross-section, as plotted in Figure 2.30. This reconstructed structure is validated successively as part of this procedure. To generate a skeleton from the smoothed TO result, a mesh-contraction Laplacian-based algorithm developed for dealing with point clouds [72, 73] is applied. Then, the curved skeleton segments are converted to straight lines by normalization. A uniform circular cross-section by mean local Euclidean distance is considered for each skeleton segment or branch. These two latter augment manufacturability and shrink design cost. Afterwards, the mini-beam rigid connection [74] solved the FEA problem in connecting the normalized skeleton beam element and the non-design part's tetrahedron to make FEA validation possible. Different proposed configurations of mini beams are brought in Figure 2.31. The option of generating a mini beam with and without contour is considered in this work. The workflow of this method, from applying loading conditions to validation, is represented in Figure 2.32 for a footbridge [75]. Among the limitations of this method are its weaknesses in the centeredness of beam axes, junction point accuracy, and confining the beam cross-section to the circular one [76].

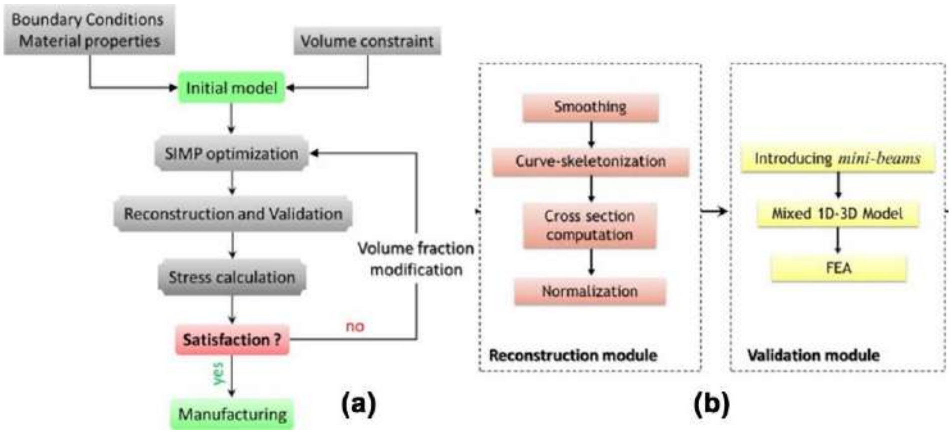


Figure 2.30- a) volume fraction loop based on the stress validation, b) reconstruction and validation stages. [71]

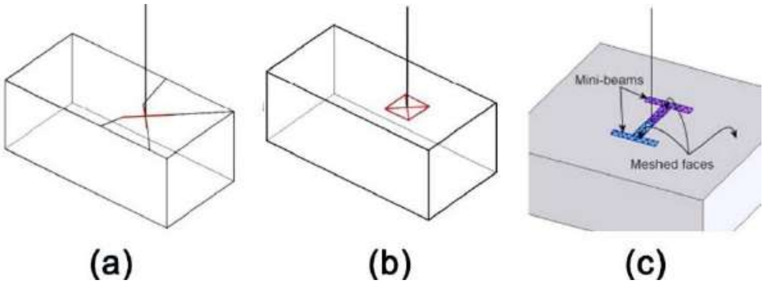


Figure 2.31- a) the first approach without contour mini beams, b) the first approach with contour mini beams, c) addition of mini beams over the internal faces' mesh in the second approach. [74]

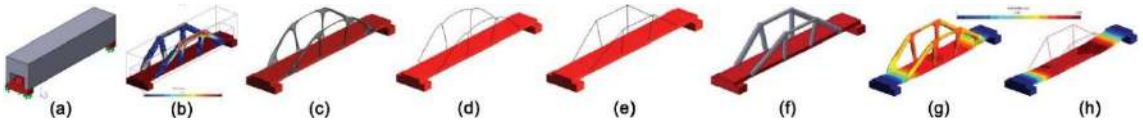


Figure 2.32- a) model with boundary conditions (BCs), b) TO result derived by SIMP method, c) smoothed result by Laplacian method, d) curve skeleton from TO result, e) skeleton after normalization, f) reconstructed 3D CAD model, g) displacement distribution for TO result, h) displacement distribution for the reconstructed beam structure. [75]

An automatic method is proposed by Kresslein et al. [77] to extract cross-sectional shape from a triangulated surface mesh, assuming this mesh is generated from a scanned point cloud. Their method can be considered a reverse engineering interpretation, as a point cloud can represent a physical object, and the result is a CAD model. Figure 2.33a abstracts the stages of section recovery in this methodology. A contraction-based skeletonization method using mean curvature flow (MCF) is applied to draw a curve skeleton from the surface mesh. Then, the skeleton is segmented into branches fitted by B-splines. Thereafter, the orthogonal planes defined on fitted segments intersected surface mesh edges and provided cross-section points, while profiting from a clustering filter to ignore noise points, as illustrated in Figure 2.33b. Several post-processing operations are suggested, such as area and moment of inertia calculation after getting cross-sections. Figure 2.33c shows the skeleton and cross-sections of a frame bike developed by this method. Among its limitations, we can nominate a lack of criteria for the number of planes according to the specific application, inaccurate centeredness of the skeleton, and extension of cross-sections into neighbour members near junctions and holes. It should be noted that the excessive number of cross-sections leads to complex shape reconstruction, and the non-centered skeleton segment makes the plane non-orthogonal and diminishes the method's efficiency.

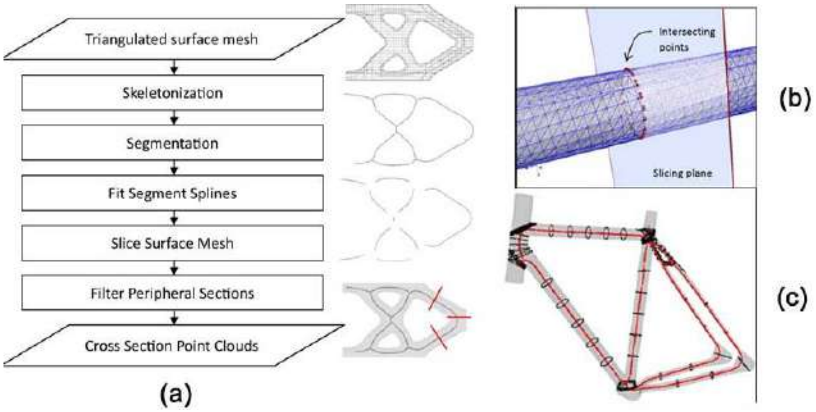


Figure 2.33- a) section recovery methodology, b) plane intersecting surface mesh edges, c) bike frame skeleton and cross sections. [77]

Yin et al. [78] proposed a fully automatic and topologically accurate strategy to solicit a compact and editable structural parametric CAD model from a voxelized solid-void binary image resulting from TO. The process of this method consists of five phases: SIMP-based TO, homotopic skeletonization, structural frame model generation, size and layout optimization, and CSG tree CAD model generation, as depicted in Figure 2.34. The voxel model and skeleton are both presented in Figure 2.35. Since mechanical considerations are not figured on in skeletonization and graph algorithms, the frame structure needs to go through size and layout optimizations to adjust the cross-section of the frame members and the joints' coordinates, respectively. Eventually, the structure is converted to a compact CAD model as a CSG tree, making it editable. The employed primitives in this study are cylinders and spheres; applying others would be straightforward. The final solid model is a B-rep of NURBS curves and surfaces.

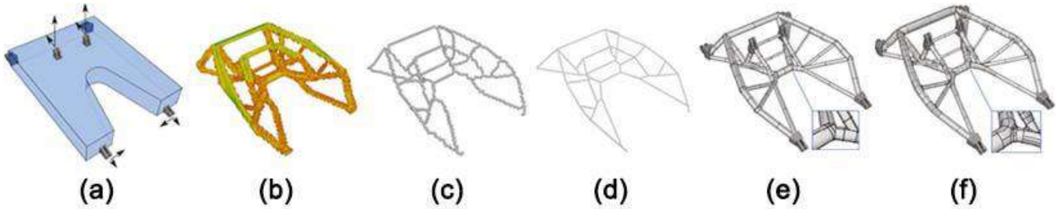


Figure 2.34- a) design space, b) TO geometry, c) thinned voxel chain model, d) spatial frame model, e) reconstructed CAD model by recursively combining primitive solids, f) reconstructed CAD model by recursively combining primitive solids and blended surfaces. [78]

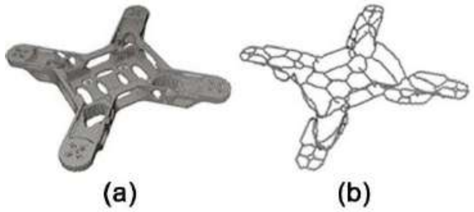


Figure 2.35- Skeletonization of the quadcopter frame. a) the voxel model from the STL mesh, b) the voxel chain skeleton. [78]

Amroune et al. [76] propounded a fully automated B-rep CAD reconstruction process from SIMP result. It should be mentioned that the TO result is smoothed with Taubin approach [79], and the skeleton is provided by using Tagliasacchi et al. [80] method. The principal echelons of this process are skeleton postprocessing, junction surface generation, branch surface generation, and solid model construction, as elucidated in Figure 2.36. Cross-sections are computed via B-spline interpolation and lofted to establish CAD faces in branches and junctions while the remaining opening area is closed by filling faces. Therefore, sewing two latter types of faces together provided the wanted CAD model. The gained model was validated by FEA. The complete process for a cantilever is pictured in Figure 2.37. This strategy has contributed significantly to integrating TO, and the outcome

CAD model can be editable and/or refinable. However, it has limitations like being bounded to beam-like TO and obtaining G0 continuity across connections between branches and junction surfaces instead of G1.

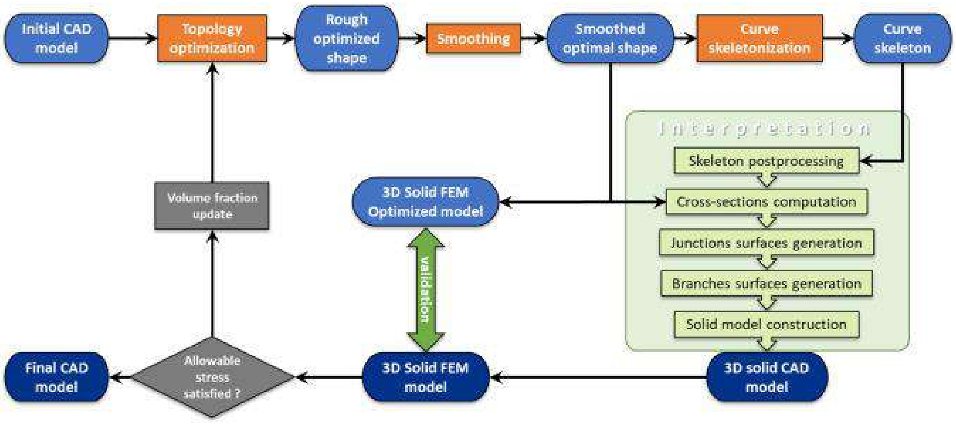


Figure 2.36- The flowchart of the reconstruction process. [76]

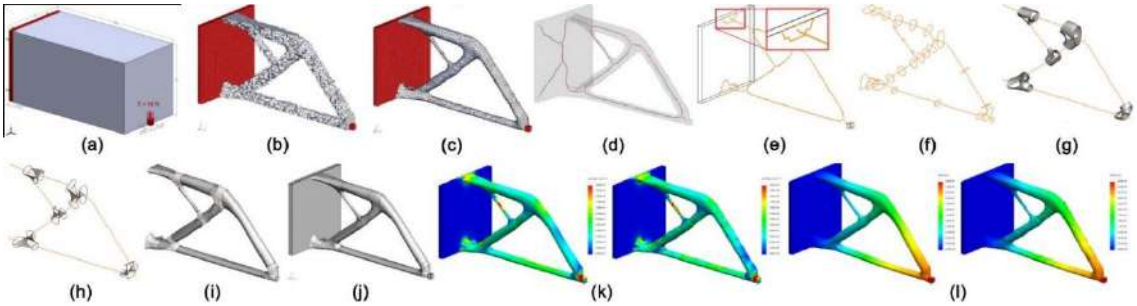


Figure 2.37- a) design space with BCs, b) TO result, c) smoothed TO result, d) curve skeleton, e) smoothing connections with non-design interfaces, f) 3 cross-sections per branch, g) generated connecting B-spline surfaces, h) generated filling surfaces on junctions, i) branches lofting surfaces, j) reconstructed CAD model, k) Von Mises stress distribution and l) displacement distribution for TO result and reconstructed model. [76]

The interpretation methods reviewed above are compared in Table 2.2 with respect to their applicability to beam-like structures, automation level, and compatibility with manufacturing requirements.

Table 2.2- Comparison of the discussed interpretation methods for beam-like TO results.

Approach	Multi-sectioning	Geometric approximation	Reverse engineering	Skeleton-based
Key techniques	Iso-density contours, sweeping, B-spline	Skeletonization, template fitting	Skeleton extraction, surface fitting	Curve skeleton, cross-section extraction
Advantages	Natural for elongated members, clear cross-sections	Truss extraction, simple templates	Physics-based, parametric CAD output	Direct beam representation, structural logic preserved
Limitations	User-defined thresholds, junctions complexity	Limited template library, accuracy issues	Centeredness issues, circular cross-section limit	Junction accuracy, centeredness challenges
Automation level	Semi-automated	Semi-automated	Automated to manual	Fully automated
Manufacturing compatibility	High for AM	Medium	Variable	High potential
References	[56-61]	[27, 62-66]	[51, 53, 68-70]	[71, 76-78]

Recent progress in reconstructing beam-like structures has made the process both more automated and more precise. Denk et al. [81] combined curve skeletons with Meta Balls to create smooth surface models of beam networks, which also made it possible to adjust beam sizes automatically to meet manufacturing limits. In another approach, Morris et al. [82] introduced BeNTO, a method that merges beam network extraction with level-set TO so that beam structures evolve alongside the optimization process. Polak and Nowak [83] built a skeleton-based system for detecting truss forms in biomimetic optimization. They showed that the results can be turned directly into parametric CAD models for industrial applications. These studies mark a steady move toward more automated and manufacturing-aware reconstruction methods.

In conclusion from the literature, full integration of TO into the CAD model has not yet been achieved and still requires profound investigations and research. Since a comprehensive fulfillment of this appeal appears to be a widespread desire, TO results should be classified to make this integration feasible step by step, as the approach of this work, studying beam-like TO results. As mentioned in the last paragraph of reverse engineering, interpretation via skeleton can be considered 1D and somehow the simplest; however, studies on it are still imperfect and have limitations, as discussed in the previous section. Therefore, the present work is oriented toward beam-like structures, and it is evident that providing a skeletonization method that best fits the needs of the work is crucial. Accordingly, the skeletonization methods, their features and deficiencies are overviewed in the next section.

2.4. Curve skeletonization

Nowadays, skeletonization plays a key role in computer graphics and vision, pattern recognition, and machine intelligence by representing the topology and geometry of 3D objects. The skeletons are utilized in various applications due to their ability to represent effectively an object's shape and boundary, such as animation, retrieval, matching, morphing, segmentation, navigation, and reconstruction. The concept of a skeleton is diversely defined; four definitions are portrayed in Figure 2.38, and a well-known definition is the medial axis, which consists of the centers of maximally inscribed spheres within an object. Depending on the dimension of this object, two types of skeletons may be generated: for 2D objects, a curve skeleton in 1D will be obtained, while for 3D objects, either a curve skeleton or a surface skeleton in 2D can be obtained. A curve skeleton can be derived either directly from 3D objects or by simplifying a surface skeleton. Many methods have been developed to extract skeletons from an object and categorized based on different criteria, which this section reviews.

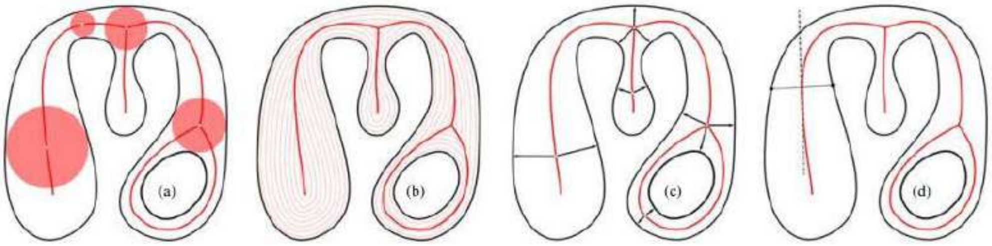


Figure 2.38- Alternative definitions of medial skeleton: a) centers of maximally inscribed balls, b) shock graph of the grassfire surface flow, c) points with more than one corresponding images on the surface, d) local axis of reflectional symmetry. [84]

2.4.1. Methods

Several works are devoted to surveying a collection of skeletonization methods to execute and compare them and emphasize their strengths and shortcomings together with the categorization of rendering methods. However, most rely on curve skeletonization methods since the surface skeleton structure is more complex and slower to compute than curve skeletonization. Moreover, the present work concentrates on beam-like structures and subsequently deals with curve skeletons, which should be kept in mind.

2.4.1.1. Categories

As mentioned, various kinds of classification have been proposed for curve and surface skeletonization methods, some of which bring particular attention to curve skeletonization. In a comprehensive classification [84], methods are divided into curve and surface skeletonization according to their dimension, with further subdivision into spatial sampling, image, and analytics. Image methods use a uniform discrete voxel sample, and analytic ones use a continuous and typically non-uniform sample. Although both classes converge to a skeleton, their output have distinct properties suited to specific applications. Image methods include topological thinning and distance-field techniques for surface skeletonization, with projection being an additional technique for curve skeletonization. On the other hand, analytic methods for surface skeletonization are based on techniques such as the Voronoi diagram (partitioning of space into regions based on proximity to points in a sample), bisector, and shrinking ball. For curve skeletonization, analytic methods encompass medial-surface-based, generalized field, contraction, mesh decimation, property grouping, and topology-driven techniques. In subsequent sections, only curve skeletonization methods are investigated. An example of a Voronoi diagram with accuracy enhancement of the skeleton by increasing sampling is represented in Figure 2.39, while a topology-driven Reeb graph (a topological structure that captures the evolution of level sets of a function on a surface) and reconstructed object from it are shown in Figure 2.40.

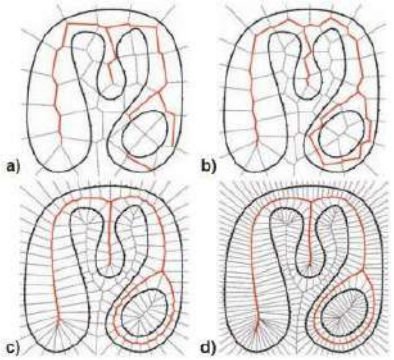


Figure 2.39- Voronoi diagram of a boundary with increasing and uniform sampling density. [84]

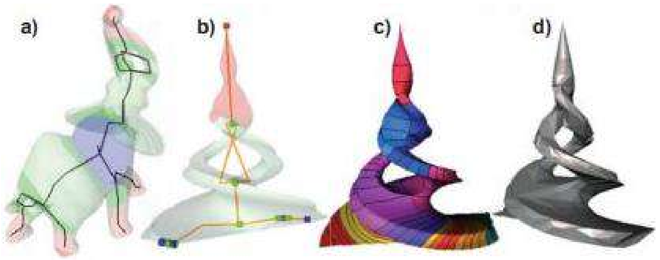


Figure 2.40- a) Reeb graph of a 3D shape built by the integral geodesic distance. d) Shape reconstruction from b) its Reeb graph using c) level set tiling. [84]

Tierny et al. [85] proposed a curve skeletonization method, particularly suited for computer graphics applications such as shape deformations using constriction approximation on polygonal meshes to refine the Reeb graph. Figure 2.41 reports this flow. The developed method is insensitive to noise, robust against mesh sampling variation, and invariant to geometrical transformations.

(a) Feature points. (b) Mapping function. (c) Dual Reeb graph. (d) Constriction approximation. (e) Constriction enhanced graph. (f) Application to deformation.

Figure 2.41- Main steps of the method on a standard model. [85]

Four classes are presumed for curve skeletonization [86], including thinning and boundary propagation, distance field, geometric field, and general field, where thinning is illustrated in Figure 2.42. Potential-based skeletonization as a general field is demonstrated in Figure 2.43. The Voronoi diagram and Reeb graph are employed as geometric fields, and functions other than distance are treated as general fields consisting of potential field, electrostatic field, repulsive force, and radial basis function.

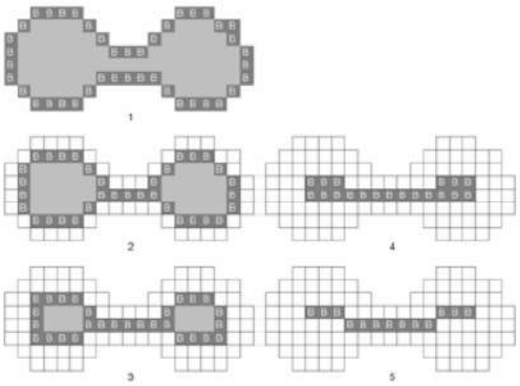


Figure 2.42- The thinning process on a 2D shape. [86]

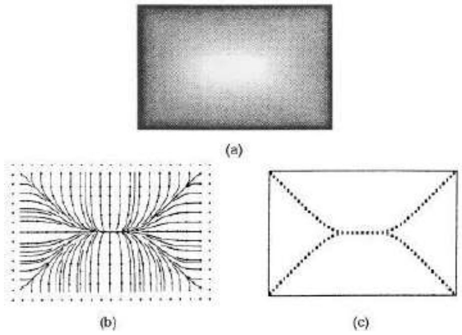


Figure 2.43- Newtonian potential-based skeletonization. a) the potential function, b) the streamline of the force fields, c) the derived skeleton. [87]

Farag et al. [88] developed a general field function skeletonization method by introducing an antipodal point computation algorithm as the key part of their method. The antipodal point of a boundary mesh vertex is its farthest point located on the boundary, whereas beforehand, their connecting line should be entirely inside the mesh. Thus, the center of this line is a skeletal point, and a set of them clarifies the curve skeleton. The smoothing and center fitting as post-processing is carried out afterward. In addition to centeredness, it preserves details, making it convenient for reconstruction. The quality of this method's performance is revealed in Figure 2.44.

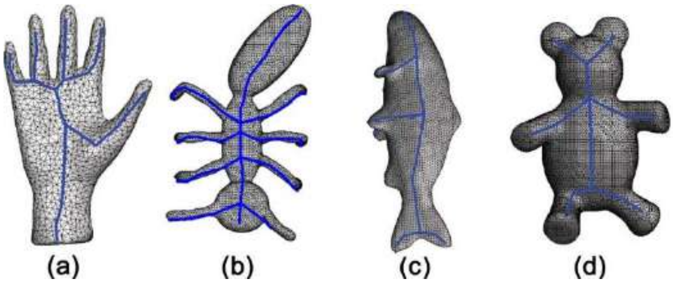


Figure 2.44- skeletons generated by the method for 3D objects. [88]

In another taxonomy [2] based on Blum's grassfire analogy, Blum can be placed as a pioneer in defining the skeleton concept; curve skeletonization methods are assumed in four branches: thinning, advection-based, erosion-based, and contraction-based methods. In advection-based methods, an important factor is attributed to each point of the shape, and then, by applying a threshold and advection principle, the shape will turn into a surface and consequently, a curve skeleton.

In the last grouping, two main subdivisions, volumetric and geometric, are propounded [72, 73] for curve skeletonization methods. Whereas voxelized, thinning, distance, and general fields are subspecies of volumetric methods, and Voronoi diagram, Reeb graph, and contraction methods are taken in the geometric methods class. It is noted that there are methods that fall in neither volumetric nor geometric groups. Figure 2.45 illustrates how the skeleton is derived from the grassfire model through a distance map under a distance-based method.

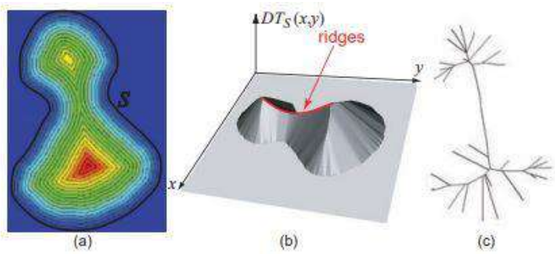


Figure 2.45- a) 2D shape boundary S with its distance transform shown by color-coding and level sets. b) Ridges of the distance transform plot. c) Corresponding shape skeleton. [84]

Overall, the curve skeletonization methods can be apportioned into volumetric and geometric (image and analytic), in which volumetric employs discretized field function or voxelized representation, where geometric enforces directly on point clouds or polygon meshes. Therefore, it gives them some positive and negative features and makes researchers nominate desired curve skeleton (CS) properties to decide which are more prominent in the target application. The common probable bugs in volumetric methods are detail loss and numerical instability originating from inappropriate discretization resolution, in addition to the necessity of defining the shape of interior parts well. Low grid resolution gives a coarse skeleton; a high one is costly in implementation [2]. Due to the grid limitation, they can offer accurate reconstruction but of lower quality than geometric methods. On the other side, analytical reasoning of the extracted skeleton's properties or finding the relation between the result and desired properties is more complicated in geometric methods due to their complex algorithm [89]. To count the apparent drawbacks of geometric methods, the Voronoi diagram is sensitive to noise and sampling conditions; the Reeb graph can be sensitive to the model's orientation or require boundary condition specification depending on the chosen function on the model's surface to cluster iso-valued regions and connect them. Moreover, it has not been proven that the geometric methods are more efficient than the volumetric ones since they are compared. It would be interesting to report comparing a volumetric method with both volumetric (distance-based renowned as well-centered method) and geometric methods based on medial surfaces and inside-out evolution schemes illustrated in Figure 2.46. Due to simplifying the finding of the center points, it is several times faster than the most accelerated method of the geometric class, contraction-based. In addition, it does not have the centration problem as shot in Figure 1.2 where the left generated by a contraction-based method but still has the common deficiencies of volumetric methods.

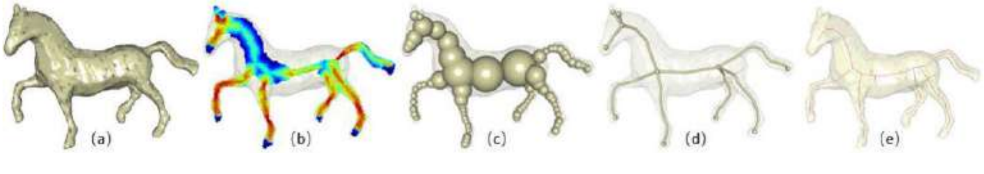


Figure 2.46- a) a 3D shape, b) computed centeredness of the points on its medial surface, c) significant shape features, d) the clean skeleton by this method, e) the skeleton extracted by ET methods. [2]

2.4.1.2. Contraction-based methods

This section explores contraction-based methods to select well-centered CS methods, avoid the limitations and imperfections of voxelized methods as volumetric or image methods, and still employ an accelerated method. Moreover, it is claimed that the state-of-the-art methods for curve skeleton extraction from high-resolution shapes are currently contraction ones [80].

CAO et al. [72] proposed a contraction procedure for reconstructing 3D models from incomplete point clouds, by building a skeleton graph, edge contraction, location correction, and modification by radius field, as illustrated in Figure 2.47. Meanwhile, contraction and developing skeleton graphs are carried out using the Delaunay Laplacian with cotangent weights and farthest point sampling, and a curve skeleton is extracted through topological thinning. Then, every vertex is relocated to the center of its local neighbourhood and ameliorated centeredness. A radius field is used to refine the skeleton, with adjustments to prevent branch collapses or hole artifacts.

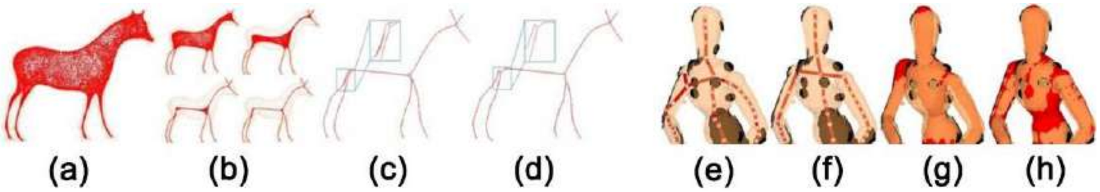


Figure 2.47- The contraction algorithm plus the editing process. a) input point cloud and the contracted points after the first iteration, b) contracted points after 2-5 iterations, c) skeleton graph constructed by farthest-point sampling, d) curve skeleton after topology thinning by contracted edge, e) extracted skeleton by this contraction algorithm, f) edited skeleton by location correction, g) the radius field defined on the skeleton depicted as a conical shape approximation, h) modified radius field. [72]

The stages stated in the best-known mesh-based curve skeletonization method [73], Au, embrace geometry contraction, connectivity surgery via edge collapse, skeleton-mesh mapping, and embedding refinement epitomized in Figure 2.48. The constrained Laplacian smoothing performs the geometry contraction. The skeletonization process yields a skeleton-mesh mapping, which provides precious information for centeredness.

Although the embedding refinement output can solve the out-of-center trouble, the method can still not guarantee that the skeleton is inside the object.

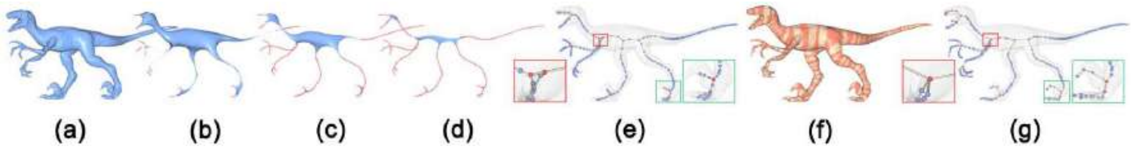


Figure 2.48- The skeleton generated by geometry contraction. a) the original mesh, b, c, d) the contraction iteration results, e) the connectivity surgery on the contracted mesh, f) the induced skeleton-mesh mapping, g) the curve-skeleton after embedding refinement. [73]

A discrete form of MCF is inspired by the Au method [73] to expand a contraction-based and simultaneously mesh-based skeletonization method, which is claimed to be robust, well-centered, and topology-preserving [80]. It is stable and efficient because of re-meshing and connectivity updates added to MCF to elide numerical instabilities and control the flow. Before igniting the process, the Voronoi diagram of the triangular mesh vertices and successively medial poles are computed thanks to QHull. An implicit constrained Laplacian solver is utilized to smooth the mesh repeatedly when the triangular mesh is re-meshed locally by splitting poorly shaped edges and collapsing short edges to fade the volume throughout meso-skeletons. Due to the contraction, local surface curvature changes; therefore, Laplacian should be updated. The MCF makes it possible to consider an energy term in the algorithm to pull the vertex to Voronoi poles after each edge collapse and improve the centeredness. They provided free access to their algorithm at http://code.google.com/p/starlab-mcfskel. This method proceeding is demonstrated in Figure 2.49.

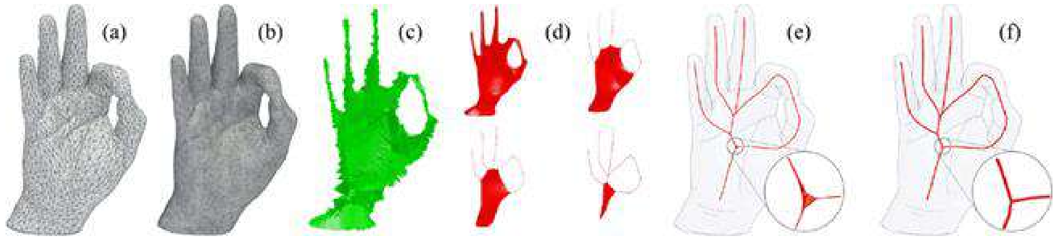


Figure 2.49- An overview of the skeletonization algorithm. a) input mesh, b) remeshing to obtain better quality Voronoi poles, c) medial skeleton, d) meso-skeleton by iterative mesh contraction, e) thin skeletal structure, f) curve skeleton via edge collapse. [80]

The curve skeleton can be extracted from the surface skeleton formed from either a 3D object or a point cloud due to the TJ method [90] represented in Figure 2.50. It works based on an explicit advection strategy applied to the surface skeleton in the implicitly computed gradient of its distant-transform field. This makes curve skeletal points on the medial surface located at a maximal geodesic distance from at least two medial surface boundary points, just as defined in the definition of the centeredness property.

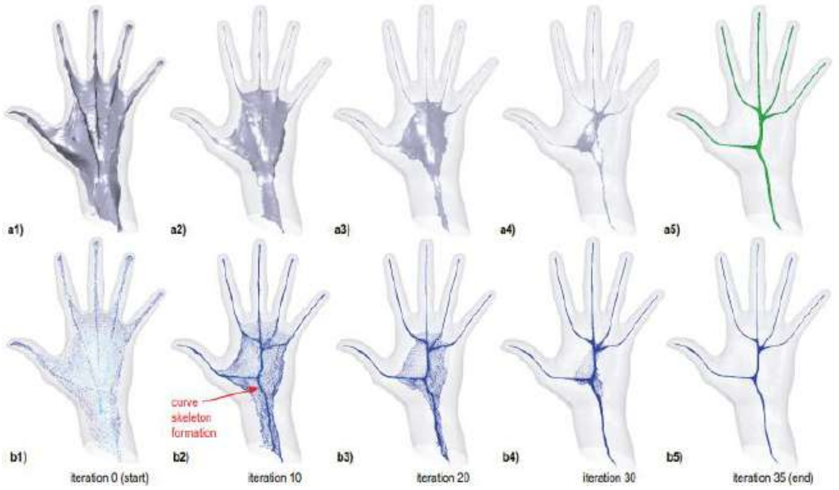


Figure 2.50- Advection steps for curve skeleton extraction. a1-a4) surface skeleton mesh, b1-b5) medial point clouds, a5) curve skeleton. [90]

A technique founded on an oriented point set’s generalized rotational symmetry axis (ROSA) misses some data and cylindrical regions for the skeleton computation part [91]. Figure 2.51A, B illustrates ROSA performance facing missing data and two cluster point clouds. An iterative ROSA algorithm starts the procedure point computation via planar cuts and finds the center point for the non-cylindrical joint region. Then, the skeletal samples are transferred to computed ROSA points. This process is picturized in Fig 2.51C.

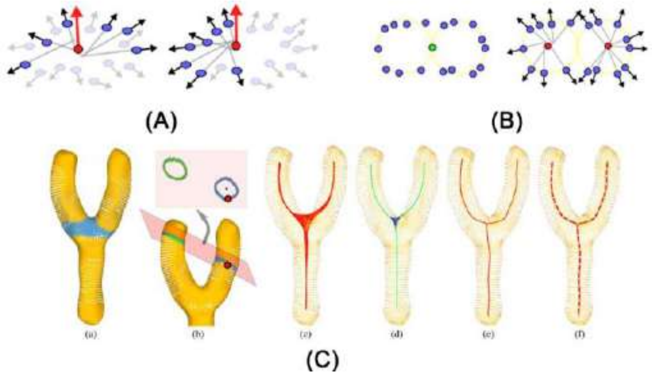


Figure 2.51- A) stability of ROSA point position and orientation in case of the points missing, B) Orientation information compensates for missing data: left) without normal leads to a wrong interpretation, right) With normal reveals two ROSA points, C) the algorithm overview. a) input point cloud with a joint (blue), b) optimal cutting plane and relevant neighborhood points, c) skeletal cloud after ROSA and joint recovery, d) after thinning with branch (green) and joint (blue) identification, e) after re-centering, f) curve skeleton. [91]

Recent developments in contraction-based skeletonization have improved robustness and application-specific adaptations. Hui et al. [92] presented a revised Laplacian contraction method along with a weight adjustment factor for tree structures. In contrast, Zhou et al. [93] proposed a skeletonization algorithm to contract points via optimizing distance summation to k-nearest-neighbours. They demonstrated faster convergence of their method than L_1-median methods. In another approach, Chen et al. [94] combined a measure of local surface variation with directional contraction to use multiscale features coming from raw point clouds, while Jiang et al. [95] developed a graph contraction approach based on the geodesic distance metric in order to retain the topology of more complex branched forms. This attempt to further develop contraction-based methods demonstrates a continuing evolution toward greater automation and robustness.

2.4.2. Comparison

Unlike surface skeleton, there is no precise definition for curve skeleton to assess the result; therefore, the only way to compare curve skeletonization methods is by comparison via the target properties of the desired curve skeleton. Thus, some common CS properties in the relevant literature with their concept are first presented to prepare the field of comparison. In continuation of this section, some qualitative and quantitative comparisons are reviewed. It should be noted that the number of comparison papers is less than the developed skeletonization methods; in addition, due to the lack of a clear definition for curve skeleton and some of its properties, the quantitative ones are more limited in number when it comes to qualitative ones. Thanks to the researchers' attempts, we overcame the challenges of 2D skeletonization. Thus, 3D is the matter of this study, although it is not mentioned.

2.4.2.1. Properties

Among all the existent properties for a curve skeleton, the focus is on the desired ones for the reconstructed TO results, as there is a conflict between some properties, and all CS properties are not essential for a specific application. Regarding the conflict between properties, it is not possible to have thinness and reconstructability or centeredness and robustness simultaneously [86]. The target properties and their definition are counted followingly [84].

Homotopy: Two continuous functions in the same topological space are homotopic if one can be deformed continuously to the other. In other words, the CS and the input shape are topologically equivalent if they have the same number of connected components and tunnels. So, small-scale tunnels (small features) should be discarded as they affect topology-based shape analyses.

Invariance: The skeleton depends only on shape, not its position or size; therefore, it should be invariant under isometric transformation. It is typically satisfied in the case of analytical (geometric) methods due to their computation procedure. However, this cannot be true for voxel-based methods because of their constraints on the fixed voxel grid.

Thinness: The skeleton is inherently thin; hence, it should be as thin as the space sampling allows, one voxel thick for voxel-based and containing just lines (no polygon or loose point) for mesh-based curve skeleton. As a consequence of grid resolution, situations like Figure 2.52 may happen in conflict of thinness and centeredness.

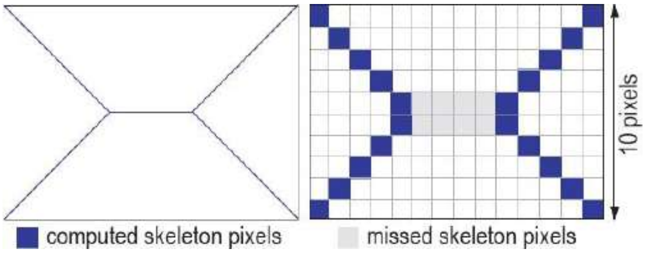


Figure 2.52- Thickness and centeredness issues. left) Formal skeleton, right) computed by a fixed pixel grid. [84]

Centeredness: In a simple definition, a skeleton point should be located at an equal distance from at least two points of surface shape, but in practice, it is not that easy to implement or assess. The issues that come up for assuring this property are the employed sampling space, as mentioned before, and the lack of a centeredness definition for curve skeletons since the commonly applied limitation is being inside the initial shape.

Smoothness: Like centeredness, smoothness is also hard to define. Surface skeletons and piecewise curve skeletons (per branch) should be at least C2 continuous to ensure smooth curvature without abrupt changes. Additionally, there are some issues regarding this property, including the quantitative assessment of sufficient smoothness and its limitation due to sampling resolution. For example, in mesh-based methods, denser sampling can complicate achieving this level of continuity.

Detail preserving/ resolution: The skeleton should be capable of accurately capturing all shape topology and geometry, while allowing users to control the scale of input shape details to preserve and eliminate. The momentous issue is distinguishing between small-scale details to preserve and noises to remove.

Sampling robustness/ regularization: Small input sampling differences should not lead to large dissimilarities in curve skeletons or sensitivity to small shape changes should be omitted. Here, the key issue is controlling the smoothing amount to achieve a satisfactory level of noise removal.

Reconstructability and computational scalability are the other properties that can be conducted to realize more accurate results at a lower computational cost. Since the goal of the current comparison is to validate the developed skeletonization method, several comparison approaches are briefly represented.

The curve skeletonization methods discussed above are summarized in Table 2.3, evaluated against properties essential for accurate beam structure reconstruction. This comparison informed the selection of centeredness as the primary criterion for the method developed in Chapter 3.

Table 2.3- Comparison of the discussed curve skeletonization methods.

Category	Geometric/Analytic		Volumetric		Contraction		
Method	Voronoi-based	Reeb graph	Distance field	Thinning	CAO	Au	MCF
Centeredness	High	Medium	Medium-High	Low-Medium	Medium-High	Low-Medium	High
Smoothness	Medium	High	Medium	Low	Medium	Low-Medium	High
Robustness	Low	Medium	High	High	Medium	High	High
Detail preservation	High	Medium	Medium	High	Medium	High	Medium
Computational efficiency	Medium	Medium	High	High	Medium-High	Medium	Medium
Junction handling	Challenging	Good	Medium	Medium	Medium	Good	Good
References	[84]	[84, 85]	[2, 84]	[86, 87]	[72]	[73]	[80]

2.4.2.2. Qualitative and quantitative comparison

As mentioned earlier, due to the lack of an exact definition for CS, methods are compared based on desired CS properties in the literature. Among these properties, centeredness is ambiguous since no exact definition exists. However, exact centeredness is not required for most applications, and approximate or relaxed centeredness is also satisfying [86]. Comparison studies in this field, which are essential for reconstructing TO results, are divided into qualitative and quantitative ones; subsequently, some of these studies are reviewed.

a) Qualitative comparison

Cornea et al. [86] qualitatively compared four curve skeletonization methods against eight properties of curve skeletons.

Sobiecki et al. [89] qualitatively compared six contraction-based curve skeletonization methods against six properties. Four techniques (AU, CAO, TJ, and ROSA) are discussed in section 2.4.1.2. The two more, AUI and AUS, are outstretched to improve the Au method and rectify its significant drawbacks, centeredness, and smoothness. In AUS, by exerting Laplacian contraction on the surface skeleton instead of a mesh-based model, smoothness and centeredness are improved even better than the AUI result. To summarize the strengths and weaknesses of each of these, TJ is good in smoothness and centeredness but weak in robustness and detail preserving; ROSA and CAO show a deficiency in homotopy and detail preserving. Additionally, AUS displays satisfactory performance in smoothness, centeredness, and robustness but not in detail preserving; AUI operates well in robustness and detail preserving, and AU gives a proper skeleton in detail preserving. However, the last three acts are acceptable regarding centeredness, and AUS is the most accurate, thanks to the engaging surface skeleton. Ultimately, among counted properties, centeredness and data preservation are essential for having a reconstructable model, and accordingly, Figures 2.53 and 2.54 represent the compared results of these methods respecting these properties. Then, they extended their work [96] to qualitatively and quantitatively comparison of six curve- and four surface skeletonization methods against those six properties.

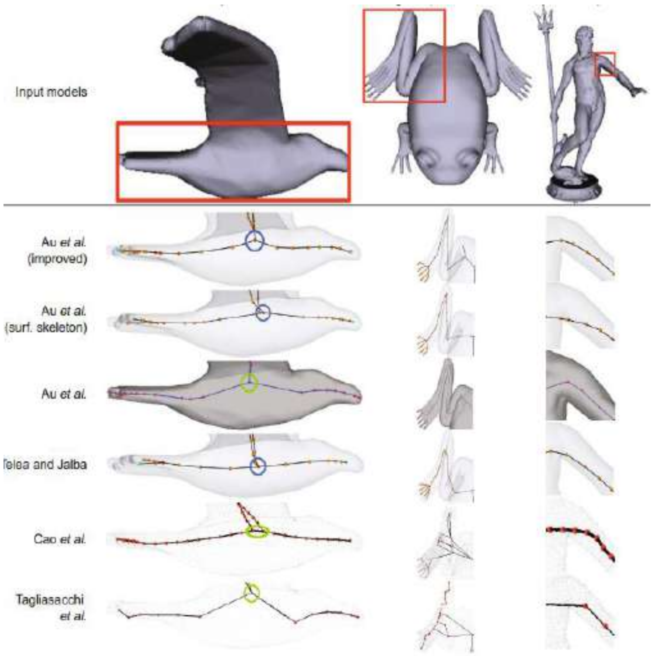


Figure 2.53- Centeredness comparison. [89]

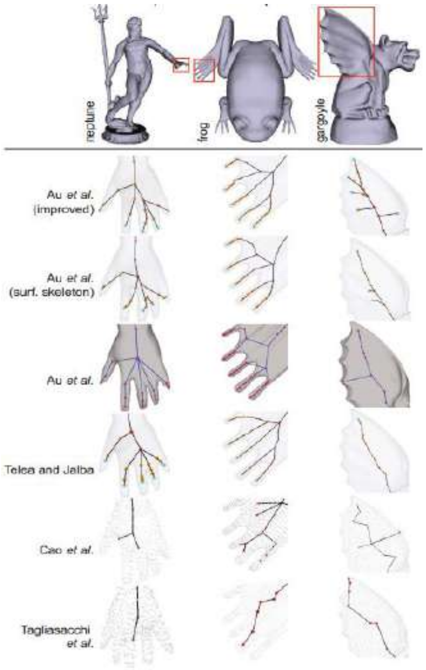


Figure 2.54- Detail preservation comparison. [89]

Tagliascchi et al. [84] compared six curve and six surface skeletonization methods; meanwhile, they criticized different properties of CS. Figure 2.55 represents the comparison results of CS methods.

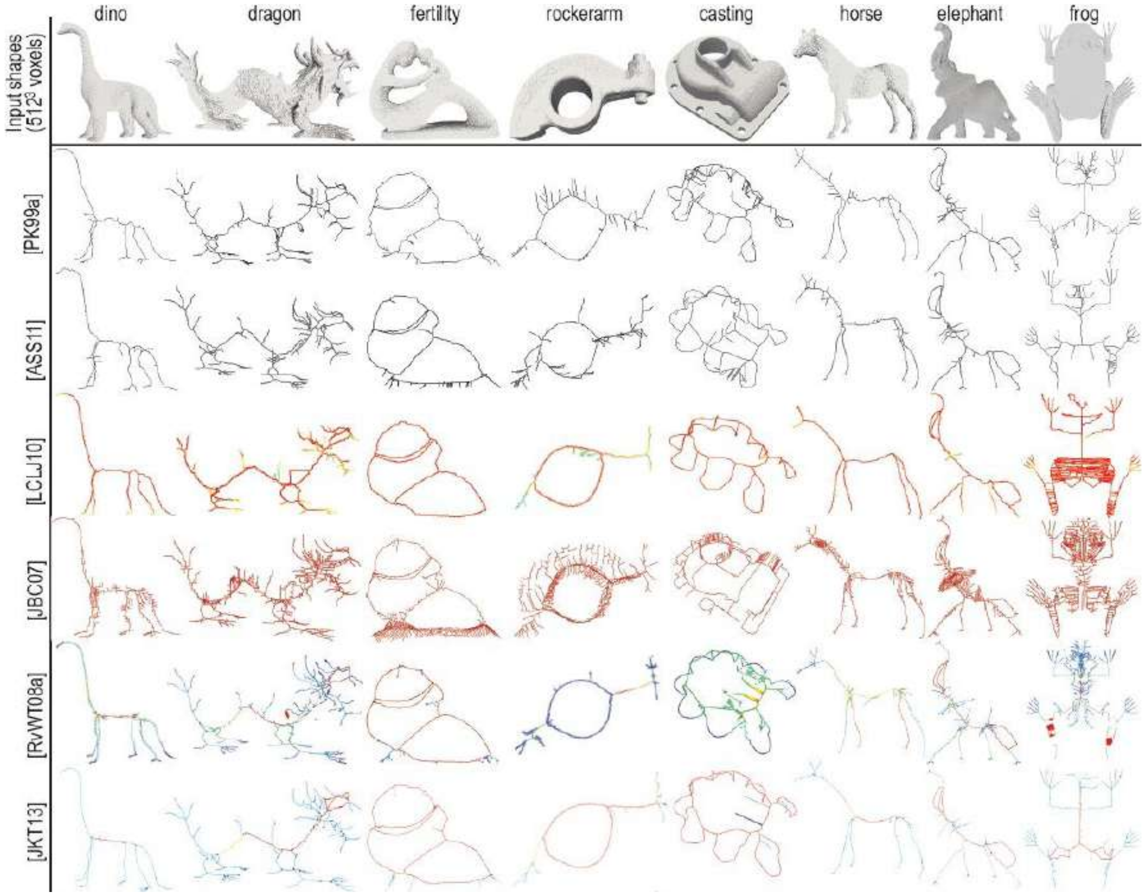


Figure 2.55- Qualitative comparison of six CS methods. [84]

b) Quantitative comparison

There are a few articles on quantitative comparison of CS techniques since almost all researchers compare methods qualitatively. Moreover, centeredness attracts more attention as it is crucial when it comes to shape reconstruction. Some quantitative assessments in terms of centeredness are counted in this subsection.

Measuring distance from skeletal points to their closest surface points of objects gives absolute centeredness while comparing different techniques applying Hausdorff distance metrics provides the relative [84].

Tagliascchi et al. [80] considered a skeleton well-centered and compared the centeredness of other methods with it. They measured average asymmetric Euclidean distances to the considered standard skeleton and normalized them.

As mentioned in the qualitative comparison section, Sobiecki et al. [96] also performed a quantitative comparison. However, this comparison is in terms of speed and memory requirements, not CS properties. Moreover, they did a centeredness assessment, although it is visual [96, 97], as in Figure 2.56. To do so, they defined a scalar distance field for (curve or surface) skeletons S_1 and S_2; they utilized a 3D Euclidean distance transform (*DT*) in this direction.

$$D_{12}(x \in \mathbb{R}^3) = \begin{cases} \min\limits_{y \in S_2} \|x - y\| = DT_{S_2}(x) & if\ x \in S_1 \\ 0 & if\ x \notin S_1 \end{cases} \tag{2.1}$$

To compare two same-kind (curve or surface) skeletons, they draw the field D_{12}+D_{21} over the voxels $S_1 \cup S_2$, normalized by their maximum value, using a rainbow colormap [96, 97]. When the skeletons are close, the colour code is blue, while for the large difference of skeletons, the colour code is red. Generally, voxels in S_1 are colored by D_{12} where voxels in S_2/S_1 are grey. Therefore, voxels in $S_1 \cap S_2$ are blue, and voxels in S_1 far from S_2 are red. For the case of comparison between curve and surface skeletons, if the surface contains the curve, the colour would be blue.

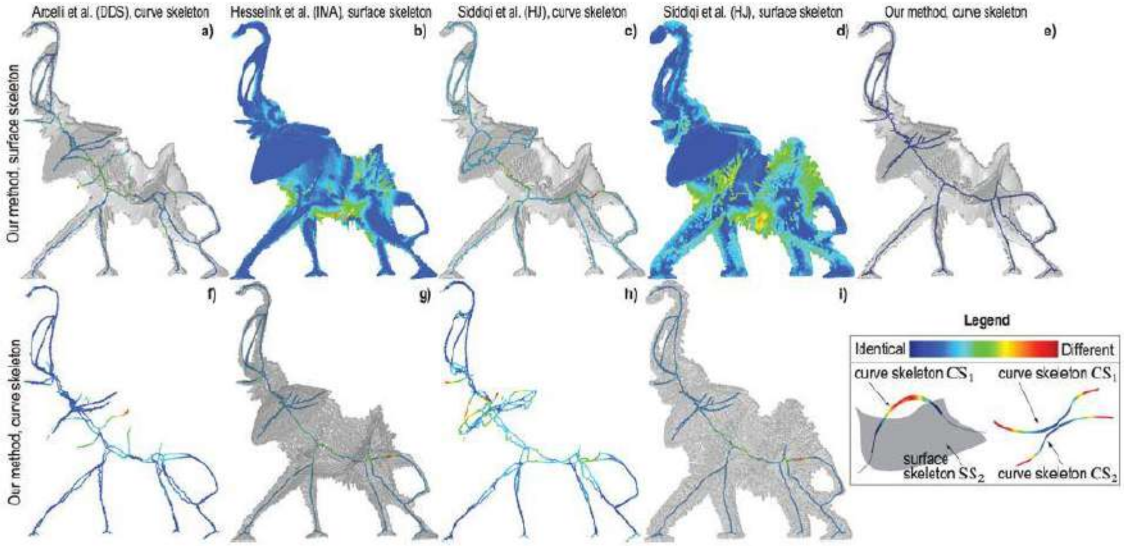


Figure 2.56- Quantitative comparison of surface and curve skeletons by mean a defined distance field. [97]

Hu et al. [98] designed a criterion based on the suggestion of Cornea et al. [86] and compared their developed method with three others regarding centeredness, as in Figure 2.57. A plane perpendicular to the skeletal direction is built for each skeletal point, O_m, while this direction can be defined using the neighbouring skeletal points. Then, the distance between O_m and each intersection point, C_i, of this plane and the object is calculated. In this intersection set, the other intersection point, C_j, is searched so that C_i and C_j are almost in a straight line. The shorter and longer distances in $|C_i O_m|$ and $|C_j O_m|$ are named l_{min} and l_{max}, while l_{min}/l_{max} is recorded as LC_i, the centeredness degree about point C_i. The average of these LC_i would give a centeredness degree of the skeletal point O_m. Finally, the average amount for all skeletal points is the centeredness degree of this skeleton. Therefore, the closer to one for a centeredness degree, the more centered a skeleton. Concerning the junction point, the centeredness degree can be computed by averaging the centeredness degree of the neighbouring skeletal point at each branch connected to this junction.

Method	ROSA method	L_1-medial method	distance field guided L_1-medial method	Our method
Model and skeleton				
Centeredness	0.980422325	0.979861725	0.979247365	0.985610647

Figure 2.57- Visual and numerical comparison results of their proposed method with three others for a cylinder. [98]

Li et al. [2] validated their proposed method by comparing it with three other methods in terms of centeredness. They began with a visual comparison, which led to the exclusion of two methods [97, 99] due to their unsatisfactory results, as shown in Figures 2.58 and 1.2, where the rightmost image in Figure 1.2 is generated by a contraction-based method [99]. Subsequently, they employed a centeredness evaluation criterion [100] to implement a numerical comparison for the remaining method [101], demonstrating that their results were comparable to or superior to those of the third method. They utilized a compactness definition to indicate the number of skeletal points forming the skeleton.

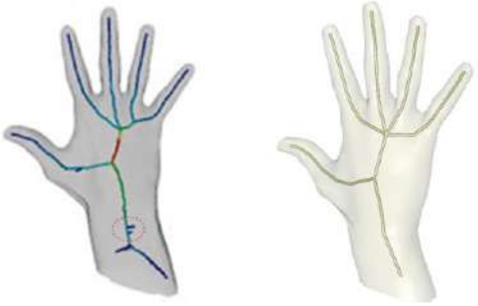


Figure 2.58- Centeredness comparison. Left- advection-based [97]. Right- proposed method [2]

Regarding the criterion, the main idea is to seed several uniformly distributed radial rays in a plane normal to the CS direction at each skeletal point and measure the distance to the boundary along each of these rays [86]. They sampled 36 radial rays at each skeletal point [100], which means 18 pairs, as each two of them is in the opposite direction. The junction points are not considered in this criterion. The following formula computes the value of centeredness:

$$E_{center} = \frac{1}{r}\sum_{i=1}^{m} \max_{1\le j\le 18} \left| d_j^i - d_{j+18}^i \right| / m \qquad (2.2)$$

Where m is the number of skeletal points or compactness, d_j^i is the distance from the i^{th} skeletal point to the shape boundary along the j^{th} ray, and the $\left| d_j^i - d_{j+18}^i \right|$ is the distance difference or off-center for a pair of rays. Finally, r is the radius of the bounding sphere of the shape. It should be noted that the smaller E value means a more centered CS.

2.5. Conclusion

Conforming to the target of the current study to reconstruct a 3D model from a topologically optimized beam-like structure, a review of the literature in this chapter is accomplished via three major sections: topology optimization, interpretation of TO results, and curve skeletonization.

The TO methods are grouped into three essential branches in the first section, including density-based (e.g., SIMP, RAMP, SINH), evolutionary (e.g., ESO, BESO, AESO, SERA), and boundary variation (e.g., level-set, phase-field). Additionally, since the desire is to obtain a manufacturable reconstructed result, manufacturing-oriented methods are investigated, which provide vital insights into integrating manufacturing in TO through specific considerations, including length scale control and geometric feature-based design. For this project, density-based and level-set methods are more relevant thanks to their ability to produce beam-like topologies that align with the structural requirements of the reconstruction process. However, the review highlights a gap in integrating these methods with automated post-processing techniques for 3D reconstruction, focusing on the necessity of exploring curve skeletonization, the concern of the current study.

In the second section, three popular techniques for interpreting TO results in accordance with the flow of the reconstruction procedure are studied, including multi-section, geometry approximation, and reverse engineering. These approaches aim to transform TO outputs into 3D CAD models suitable for fabrication or analysis. While these methods help to extract geometric information, they often lack specificity for beam-like structures, which is why interpretation strategies on these structures are reviewed separately. This review affirms that sectional analysis and precise skeletal representations are essential but underexplored for accurate reconstruction of beam-like topologies, giving an idea of proposing a curve skeletonization method providing a centered skeletal representation.

The third section on curve skeletonization clarified its pivotal role in interpreting beam-like TO results for 3D reconstruction. Among the curve skeletons features, centeredness is a critical feature for ensuring the integrity of beam-like structures in reconstructed models. So, contraction-based CS methods are highlighted as they can produce well-centered skeletons. Then, the review of qualitative and quantitative centeredness criteria highlighted the importance of numerical validation, which this project adopts to evaluate the effectiveness of the proposed method. However, existing CS methods are rarely validated numerically; this study addresses this limitation in curve skeletonization methods.

In conclusion, TO methods, interpretation techniques, and curve skeletonization approaches are essential but still limited in their integration for the automated and accurate 3D reconstruction of beam-like TO results. This gap is filled by developing a well-centered curve skeletonization method as a step toward achieving a robust 3D reconstruction model that can be manufactured using traditional methods.

CHAPTER 3 – AUTOMATIC CURVE SKELETONIZATION METHOD

3.1. Introduction

Curve skeletonization is essential to reconstructing a beam-like TO result as a matter of this study, while it is needed for other before-mentioned subjects in the first paragraph of section 2.4. This chapter presents our novel automatic skeletonization method specifically developed for beam-like TO results, with original contributions detailed in section 3.1.1. Then, the procedure for developing the proposed curve skeletonization is presented; meanwhile, the figures are used to illustrate how the method works in different steps of this development, as well as facing special conditions to ensure the generality of the method.

In this regard, the chapter begins by describing the Unified Topological Model (UTM) [102], which maps topological entities to mesh and CAD representations for integrated CAD/FEA analysis. Then, since the method's input is a beam-like TO result, it is declared how it can be generated. Accordingly, the employed TO method and the required post-processing steps are introduced, while the effective parameters for obtaining a suitable beam-like result for our proposed method are highlighted.

At this point, with the beam-like TO result generated and post-processed, the methodology description can begin. The first step is the generation of non-design skeletal points, where the non-design area is where the boundary conditions are applied. The design and non-design areas are illustrated in Figure 3.1. Skeletal points are the points of the skeleton, and a non-design skeletal point is a skeletal point on the interface of design and non-design, as displayed in Figure 3.2 in red. The following section demonstrates how the first design skeletal point is generated. Then, the steps considered for developing a branch from the non-design point to the junction are illustrated, where the criterion for identifying the junction is clearly stated. A branch is part of a skeleton where each point has only two neighbour points and is drawn in black in Figure 3.2, while a junction point is the intersection point of the branches, so it has more than two neighbour points, as marked by a green point in Figure 3.2. Afterward, it involves the steps for generating a branch between two junctions and its subsections. Therefore, when all branches between two junctions are also built, a curve skeleton is developed.

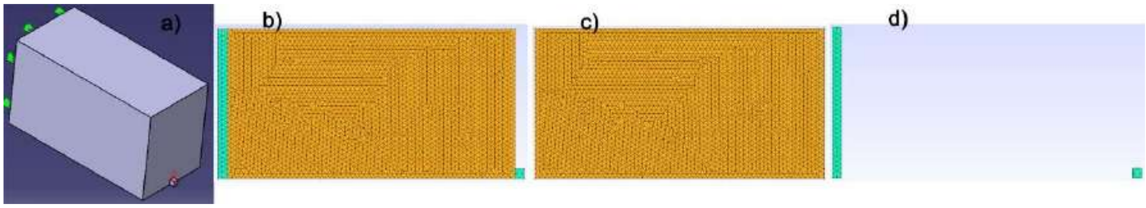


Figure 3.1- Design and non-design areas- a) cantilever loading in 3D, b) whole model in 2D, c) design area of model, d) non-design area of model.

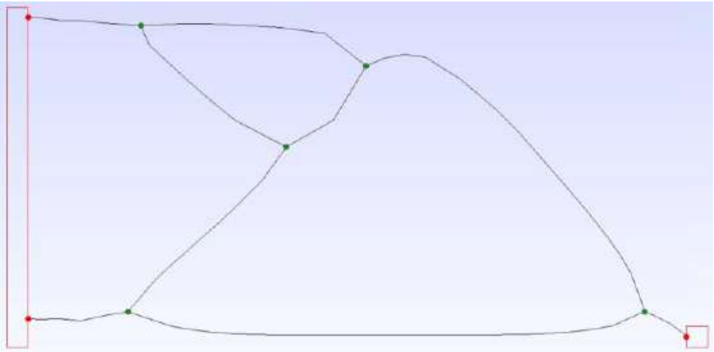


Figure 3.2- The essential definitions in this methodology.

3.1.1. Novel contributions of this work

Getting meaningful curve skeletons from topology-optimized beam structures has proven trickier than it might seem at first glance. Sure, there are skeletonization methods out there, but they were not really built with topology optimization in mind. The skeletons they produce are often not well-centered enough for the CAD reconstruction we need afterward. This thesis addresses this gap through the following original contributions:

Automatic Marching Identifier Box Method for TO Results: This thesis introduces an automated scanning approach that uses an identifier box to march through the design domain and generate skeletal points. What makes it different from existing methods is that it does not need manual intervention or parameter tweaking for each new geometry. The identifier box adjusts its dimensions based on what it encounters locally, allowing it to handle diverse beam-like structures without constant user input. Unlike existing skeletonization methods that require predefined parameters, this automatic adaptation represents a novel contribution for TO-specific applications.

Integration of TO-Specific Knowledge into Skeletonization: Most skeletonization algorithms treat everything as a generic 3D shape; they do not care where it came from. However, topology optimization results have built-in information that's actually really useful. This work proposes an approach that explicitly leverages the inherent distinction between design and non-design in TO results, a TO-aware strategy not found in generic skeletonization methods. By generating skeletons from non-design boundaries and using the SIMP density distribution, the method naturally respects the physical constraints and boundary conditions from the original optimization problem.

Dynamic Junction Detection with Multiple Strategies: Branch points are notoriously tricky to identify reliably. While traditional methods use simple geometric criteria, this thesis develops three complementary strategies: a) width proportion analysis in section 3.4.2.6, b) distance-based detection in section 3.4.3.2b, and c) identifier box overlap detection in section 3.4.3.2c. The originality lies in the multi-strategy approach: ensuring reliable junction identification even in complex geometries where relying on a single criterion might fail.

Quantitative Centeredness Validation Framework: Much of the skeletonization literature relies on qualitative visual comparisons and considers them good enough. This thesis establishes a numerical framework for evaluating skeleton centeredness, detailed in Chapter 4. This enables objective comparison between different skeletonization methods and provides a measurable quality metric for CAD reconstruction accuracy. While centeredness is conceptually understood in the field, a quantitative validation framework for TO-derived skeletons has not been reported previously.

The combination of these contributions yields a fully automatic method designed explicitly for beam-like TO results. It produces well-centered skeletons suitable for downstream CAD reconstruction, addressing the gap between TO results and manufacturable geometric models.

3.2. Unified Topological Model

UTM is an original environment and unique database organization developed to gather and integrate all relevant information from CAD and FEA, plus possessing tools to automatically maintain the consistency of this integrated data structure during design modifications [102]. UTM supports CAD/FEA integration through capabilities including automatic mesh pre-optimization, automatic remeshing, automatic CAD reconstruction, and mixed-dimensional analysis. The automatic mesh pre-optimization performs a mesh preparation for FEA from a CAD model by automatically defining analysis constraints. This is achieved by utilizing and improving advancing front mesh generation techniques within the UTM's extended BREP structure. As a CAD model evolves during the design process, UTM enables automatic remeshing by identifying geometric and topological differences between the original and modified models, reusing existing mesh elements, and generating new elements to ensure consistency with the updated CAD model. On the other hand, an automatic CAD reconstructed model can be derived from the deformed FEA mesh in UTM. It integrates beam, shell, and solid elements into its data structure to ensure efficient analysis of complex systems which require mixed-dimensional FEA.

UTM data structure consists of entities that form a B-rep, FEA entities, and physical properties, as demonstrated in Figure 3.3. Topological and co-topological entities, in addition to geometric entity pointers, are the entities that form B-rep. B-rep is chosen since most commercial CAD systems are based on it. The integration of FEA with CAD is implemented in B-rep by linking mesh entities to geometric ones with features in the B-rep's topological structure.

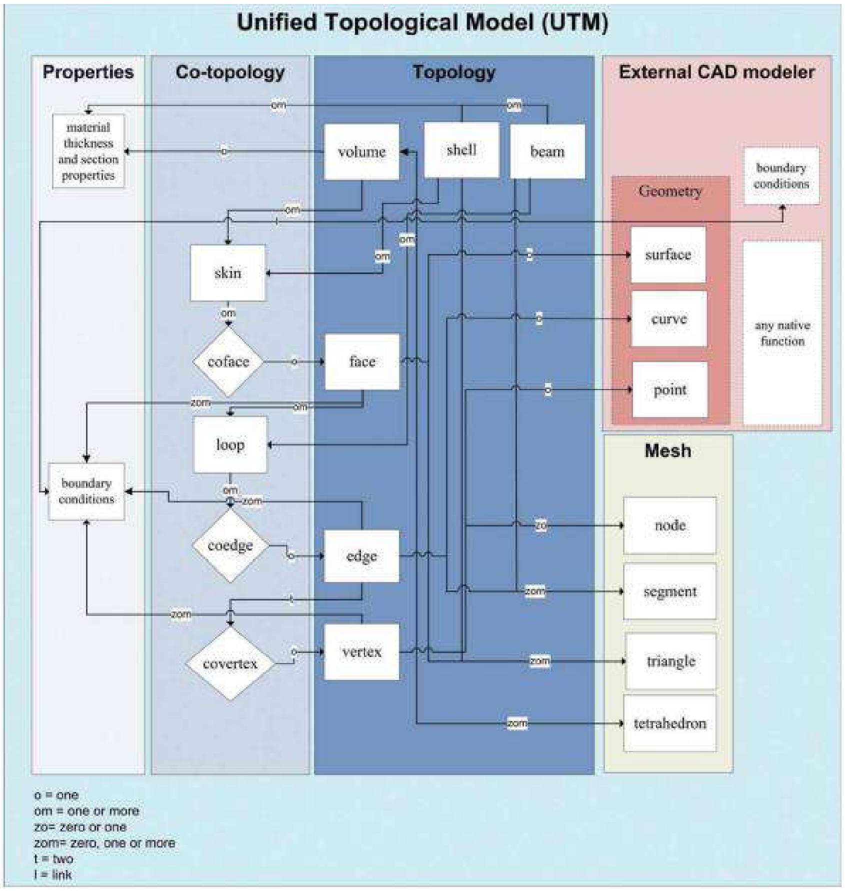


Figure 3.3- General architecture of the UTM. [102]

TO methods are integrated into UTM [52] using two B-rep models to define the complete model and non-design areas since non-design should remain unchanged during optimization. The first model represents the complete model, which includes both design and non-design components, as in Figure 3.4a, while the second contains only the non-design components, as in Figure 3.4b. So, the non-design component can be derived from the whole model easily. All finite elements are tagged as design or non-design, consequently ensuring mesh conformity at the interface of design and non-design, as in Figure 3.4c.

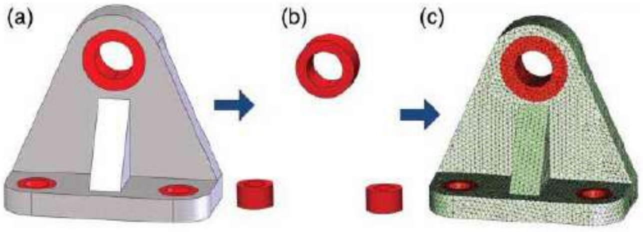


Figure 3.4- TO integration in UTM. a) whole model, b) non-design model, c) resultant meshed model. [102]

3.3. Generation of beam-like TO results

3.3.1. CAD model under BCs

In order to begin the TO process by UTM, two whole and non-design CAD models of the problem should be imported into UTM as inputs. The non-design model defines the non-design domain, enabling UTM to distinguish it from the design domain, as explained in section 3.2. The CAD model (whole design and non-design) with its BCs is presented in Figure 3.1a. The material features, Young's modulus and Poisson coefficient, are added to the model simultaneously during BCs application.

3.3.2. SIMP as the TO method

The selected TO method in UTM is SIMP, the TO result is obtained, and the model is meshed. So, an automatic 3D mesh should be generated and adapted to SIMP method, which is implemented by Cuilliere et al. [103]. The current section briefly presents the adaption process to obtain a proper beam-like TO result.

The first challenge is the conformity and continuity of the resultant mesh, as the non-design domain remains unchanged throughout the optimization process. In contrast, the design domain evolves with each iteration, requiring maintaining mesh integrity in the interface of design and non-design regions. As two subdomains are defined in UTM, one is a whole model, and the other is non-design; a back-and-forth process is applied to keep the conformity and continuity by transferring the mesh elements of one B-rep model to the other, achieved by generating a 3D unstructured mesh for the SIMP method, which needs several adaptations. Applying the required adaptations to use the unstructured mesh with SIMP provides other valuable benefits, such as non-constant element sizes, local control over a TO result, and the ability to refine or coarsen geometry details. This unstructured mesh, along with BCs, material data, and TO parameters, is the TO process input.

SIMP optimizes the material distribution within a predefined design domain to get the minimum compliance or maximum stiffness under a volume constraint. A relative density field, $\rho(x,y,z)$, represents the material distribution, varying from 0 to 1, corresponding to no material to full material, respectively. Through the optimization process, intermediate values are penalized to obtain a binary result, solid or void. Regarding the non-design subdomain, ρ is always 1 to keep it solid and unchanged during optimization. The relative density is employed to define the distribution of a virtual elastic modulus through the penalization law, $\tilde{E}(x,y,z) = E.\rho(x,y,z)^p$, where E is the actual elastic modulus of the material, and p is the penalization factor, which varies between 1 and 3 as one of the optimization parameters, and its value is set 3 in this study, as this is a widely adopted value in SIMP method to effectively suppress intermediate densities, while maintaining numerical stability and convergence. The set minimum value of density, ρ_{void}, is the amount that replaces 0 as an optimization parameter to avoid singularity, and it is 0.001 in this work. The imposed constraint is to reach a predetermined volume fraction, which is the fraction of the resultant design volume to the initial one, and it is another optimization parameter. The global compliance, $\tilde{C} = \sum_{e=1}^{N}(\rho_e)^p.\{\widetilde{U}\}^t.[K_e].\{\widetilde{U}\}$, is calculated by FEA in each iteration to eliminate elements with intermediate density values and reach the minimum amount, where $\{\widetilde{U}\}$ is the global displacement vector and $[K_e]$ is the stiffness matrix of the element. The evolution of elements' densities and global compliance values during optimization is demonstrated in Figure 3.5. This iterative process stops when the relative variation in compliance between two consecutive iterations reaches a threshold, which is also an optimization parameter set to 0.5% in this study.

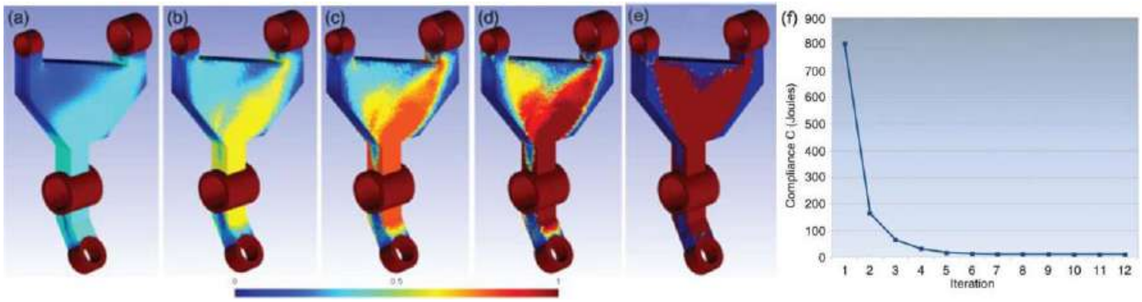


Figure 3.5- Evolution of relative density field after a) 1 iteration, b) 2 iterations, c) 3 iterations, d) 4 iterations, e) 12 iterations, and f) resultant evolution of global compliance. [103]

Two filters are introduced to apply during optimization to improve TO results concerning the checkerboard effect. They are filtering the compliance sensitivity and relative density distribution, as illustrated in Figure 3.6. These filters are applied only within the design subdomain, affecting elements within a determined radius of the element under consideration. These radii, noted r_{minc} for compliance sensitivity filtering and r_{mind} for filtering relative density distribution, can be changed independently of each other. Each of these radii is calculated by multiplying a coefficient by the local tetrahedron size, which is almost constant across the mesh. The coefficients for calculating both radii are set to 1.25 in this study, as another optimization parameter.

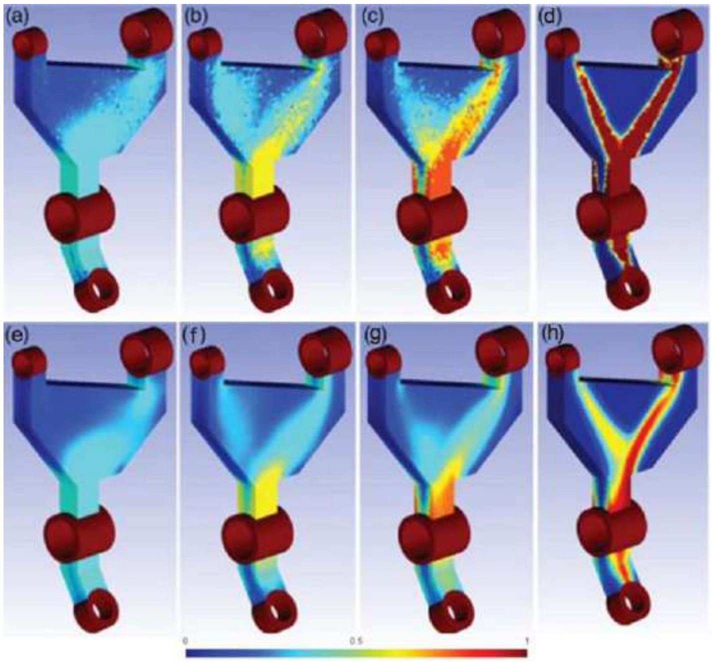


Figure 3.6- Evolution of relative density field after a, e) 1 iteration, b, f) 2 iterations, c, g) 3 iterations, d, h) 11 iterations with a filter on compliance sensitivity (first raw), and relative density (second raw). [103]

Another post-processing step is required to prepare the TO result as input for the skeletonization method, which involves generating a 3D geometry from the TO result. There are two approaches to do so: setting a density threshold, ρ_{th}, and computing iso-density surfaces. The parameter of ρ_{th} is employed to smooth the boundary of the result and remove the non-manifold patterns. Therefore, employing filtering on the relative density distribution yields better results when using this approach. The preserved design elements have densities equal to or greater than the determined value of ρ_{th} in this approach. On the other hand, to compute iso-density surfaces, the relative density distribution should be transformed into a continuous density distribution field. This continuous field can be computed by using a linear piecewise interpolation in each finite element. For this case, selecting an appropriate value for ρ_{th} is critical because of size differences in neighbouring elements after the cutting process, which necessitate a compromise in continuity of relative density distribution, especially at very low volume fractions. When computing iso-density surfaces, the boundary of the TO result is cut by an iso-density surface corresponding to a specified density threshold. In contrast, when applying a density threshold, the entire mesh is retained, with only elements below the density threshold being filtered out. Both approaches are implemented as part of postprocessing, independent of the topology optimization calculation, as mentioned in the following section.

Some parameters should be tuned to get the beam-like TO results from this process as they are more effective in forming the TO results. These parameters are ρ_{th}, f, d_g, where f is the volume fraction, and d_g is the mesh size in mm.

Concerning the TO result, there are two types of elements, one with a density equal to or more than ρ_{th} and the other with a density less than it. So, the density distribution is among the TO result, as illustrated in Figure 3.7 left. Utilizing this density distribution and the amount of ρ_{th}, beside the design and non-design labelled elements, the final TO result can be divided into three partitions: the non-design elements with a density value of one, the design elements with the density amounts equal or more than ρ_{th}, and the design elements with the density amounts less than ρ_{th}. Thus, the desired TO result with the density equal or more than the density amounts more than ρ_{th} includes the two first partitions as shown in Figure 3.7 right.

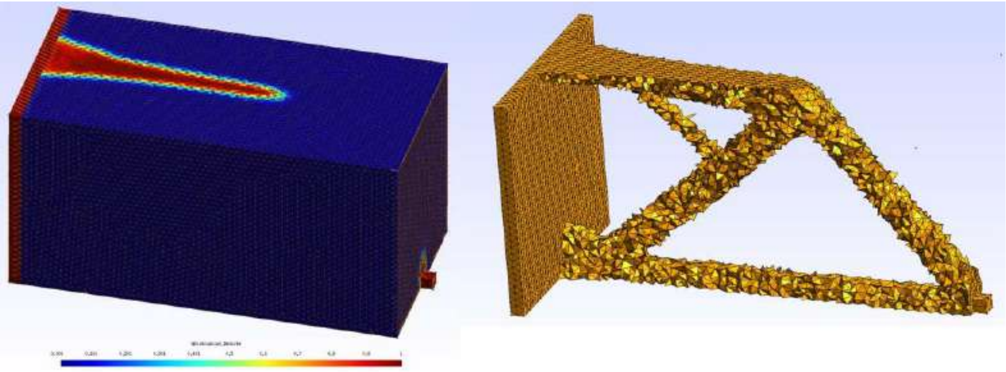


Figure 3.7- TO result. Left) The density distribution. Right) The elements with densities equal or more than ρ_{th}.

3.3.3. Post-processing of TO results

Both the cut by iso-density surface and applying a density threshold are counted as post-processes. In this study, the iso-density surface cut is not applied, and instead, the density threshold is adjusted to enforce the target volume fraction f in the topology optimization result. While the TO result should be contracted in addition to being smoothed to obtain a post-processed result that resembles a skeleton, a smoothing is considered post-processing in this work. Several smoothing methods are investigated, including Taubin 1995 [104], Chen 2005 [105], Chen 2008 [106], and Jiao 2012 [107]. The smoothing methods of Taubin and Jiao have been found effective in getting smooth results, with Jiao applied after Taubin. Taubin's smoothing technique, applied with 10 iterations on all TO results, provides initial surface refinement without significant contraction. Jiao's smoothing, with a reported number of iterations, is used for contraction to approach a skeletal form. It should be noted that after smoothing, the TO result is represented as a single closed surface shell formed by the connected boundary surfaces of the design and non-design domains. Figure 3.8 demonstrates the necessity of smoothing a TO result and how more Jiao's smoothing iterations works as contraction. Both approaches represent valid postprocessing choices for model extraction, with iso-density surfaces potentially yielding smoother boundaries prior to additional smoothing iterations.

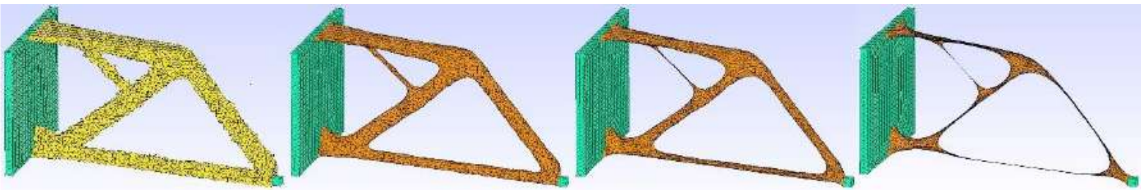


Figure 3.8- Jiao's smoothing application on a TO result- from the left: TO solid result, 5 iterations of smoothing, 20 iterations of smoothing, and 100 iterations of contracting.

Moreover, the curve skeleton is a curve; thus, the more contracted a TO result, the closer a TO result is to the skeleton, as illustrated in Figures 3.8 and 3.9. In this work, five iterations of smoothing are addressed as a post-processing step to smooth a TO result surface, while the contraction would replace it for more iterations as the first step of skeletonization. Employing this contraction technique to obtain a skeleton is effective up to a point since the object in branch areas converges to a curve while it remains volumetric in the junction areas; the branch and junction definitions are explained in the next section. The contraction effect's saturation point seems to be around 100 iterations of Jiao smoothing, according to Figures 3.8 and 3.9. This saturation point indicates the number of Jiao's iterations in which the contracted result is sufficiently converged to a curve in branches. The contracted result changes significantly up to the saturation point (~100 iterations) but stabilizes thereafter, as observed at 20 and 200 iterations.

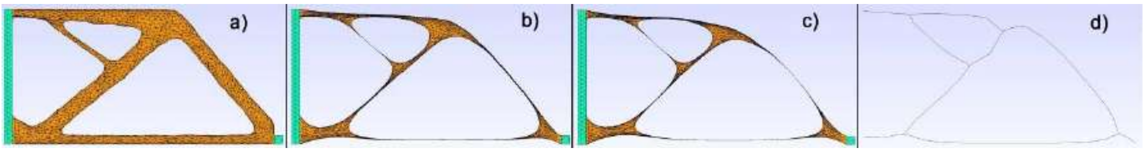


Figure 3.9- The contraction effect on the skeleton generation. a) 5 iterations of smoothing, b) 100 iterations of contraction, c) 200 iterations of contraction, d) the curve skeleton.

This study reports the amount of ρ_{th} for each case study. Then, the starting iteration number for each technique is used to show the sequence of the methods' application, Jiao [107] after Taubin [104], the number of iterations to which each method is applied among the inputs in post-processing. In this chapter, the number of 100 iterations is chosen for Jiao to obtain a sufficient contracted result to conclude the skeleton. The smoothing parameters are unified to facilitate the implementation and minimize the variables to make the comparison more precise. The utilized contracted result in this chapter is illustrated in the rightmost image of Figure 3.8.

3.4. Methodology

The proposed method focuses on analyzing the beam-like design region, which is enclosed by non-design regions forming its boundaries. An adjustable marching identifier box reads the nodes within this design region. Starting at the non-design region boundaries, the box moves through the design region in discrete steps. At each step, the identifier box collects the nodes within its boundaries and calculates a skeletal point at their center. If this skeletal point lies within the design region, it is connected to the previous skeletal point to form a skeletal structure.

As a matter of clarification, it is essential to declare several definitions before stating the assumed process steps. An identifier box is a cuboid that captures and analyzes a part of the object, including some mesh entities, as shown by a pink rectangle in Figure 3.10. In this context, scan refers to the process of the identifier box systematically collecting data from a specific region of the object and analyzing it. The marching direction, chosen exclusively from the global reference frame directions (x, y, z), is the direction in which the identifier box moves to scan the next part of the object, as marked by a yellow arrow after the non-design set in Figure 3.10. A junction box is the identifier box, including the junction point, as shown by a green rectangle in Figure 3.10. In this figure, the non-design sets are highlighted in green.

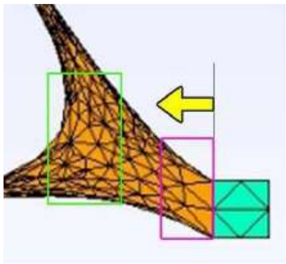


Figure 3.10- The essential definitions in this methodology.

The process begins with non-design sets, as shown in Figure 3.11, where each set serves as the starting point for a branch. The first step generates a non-design skeletal point at the interface of design and non-design parts. Next, the branch from this skeletal point to a junction point is created through steps like identifying the marching direction of an identifier box, constructing an identifier box, adjusting its dimensions, generating the first design point of a branch, and determining the following points. However, these steps can vary in order or necessity depending on the case. The JC’s info and dir, as depicted, include the junction box border, and the directions identifier box has marched or will march to form new branches from this junction point. Finally, the process concludes by generating a branch between two junction points, involving distinct considerations such as identifying upcoming branches at a junction and updating the junction coordinates. The subsequent sections clarify each part of this procedure while the basics of the method are explained. The collocation of three directions in these sections refers to the global reference frame directions: x, y, and z.

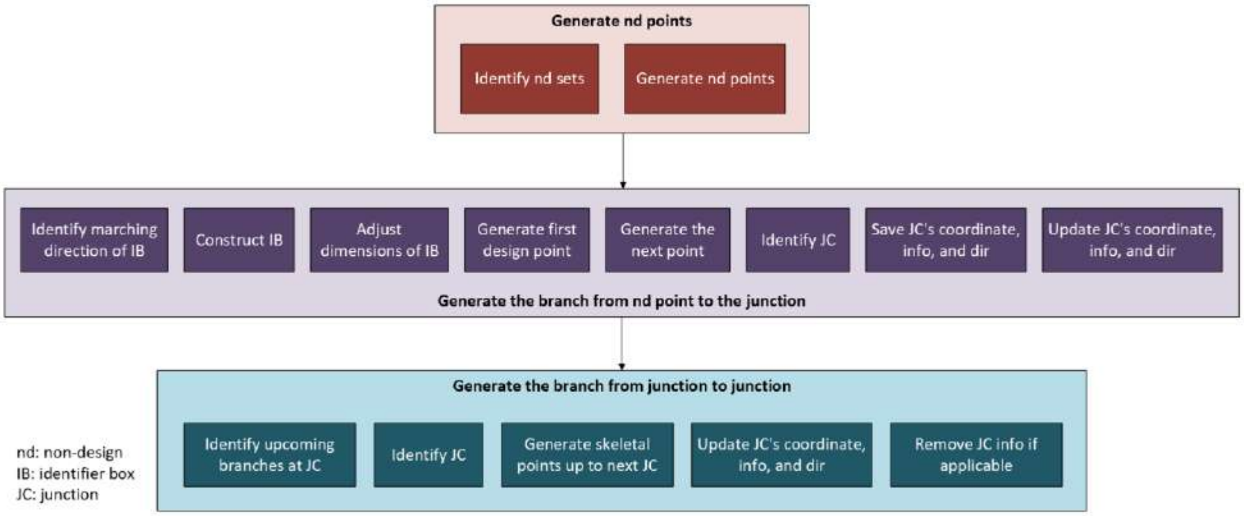


Figure 3.11- The methodology mechanism.

3.4.1. Generation of non-design skeletal points

In the beginning stages, the non-design skeletal points should be created to start the branches from non-design to the junction point. As demonstrated in Figure 3.12, the labelled non-design mesh elements are found and classified in different sets wherever they are connected. Thus, the interface sections of design and non-design are identified. The center point of each interface section is the non-design skeletal point. To facilitate understanding and following the method's logic, the algorithm's steps are picturized in 2D. However, 3D captures are provided when they are more helpful.

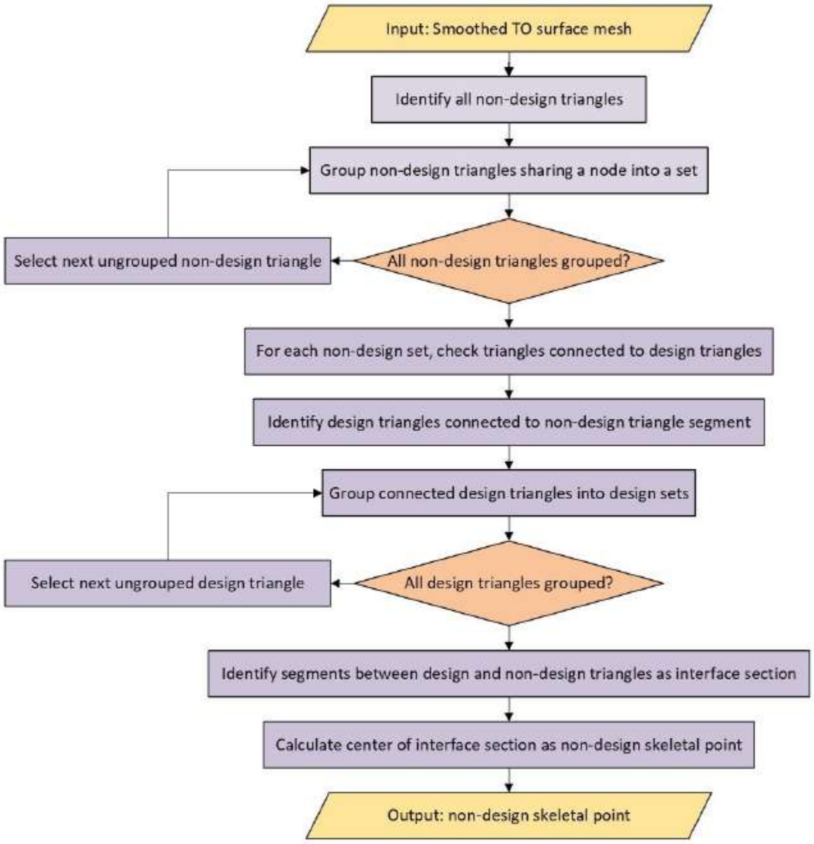


Figure 3.12- The methodology mechanism in generating non design points.

3.4.1.1. Identifying non-design sets

The method's input is a smoothed beam-like TO result in the form of a surface mesh including triangles, segments, and nodes. Initially, all non-design triangles are identified. The algorithm identifies adjacent triangles that share a common node and groups them into a single set. This process is performed for each triangle in the set till all connected triangles are included. The procedure repeats for the remaining ungrouped triangles until the entire non-design triangles are partitioned into split sets of connected triangles, as illustrated in Figure 3.13.

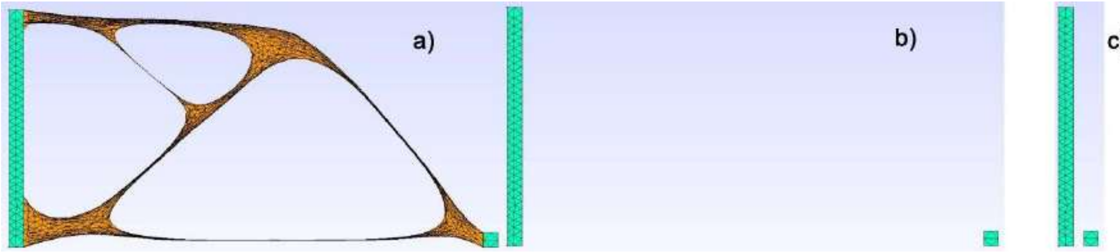


Figure 3.13- Non-design sets identification. a) the smoothed TO result. b) non-design triangles. c) two non-design sets.

3.4.1.2. Generation of non-design skeletal points

In each non-design set, all triangles are studied to determine whether they are connected to the design triangles. So, the triangles connected to each segment of the non-design triangles are studied, and if it is a design triangle, it is put in a design set. Starting from one of these triangles, the connected design triangles to this one through a node are collected in another design set and removed from the source design set. Like classifying non-design sets, it continues to find all connected design triangles and classifying all found design triangles in different sets, as presented in Figure 3.14a. In contrast with the number of non-design sets, visible in Figure 3.13c, two, the number of design triangles sets is three, as two design sets are connected to the same non-design set. The segments in the connection between design and non-design triangles form the interface section of design and non-design, as the orange parts in Figure 3.14b in 2D view. Finally, the center of this interface section is the non-design skeletal point, as highlighted in yellow in Figure 3.14c. This center is the average of all section node coordinates.

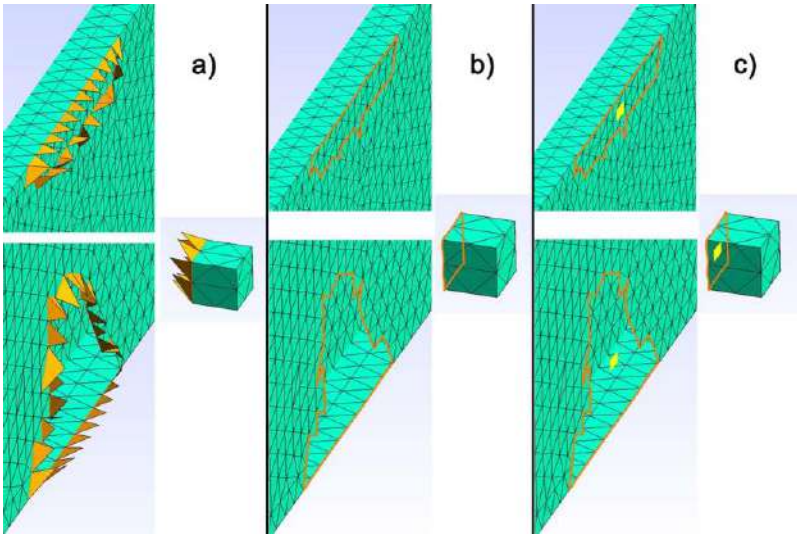


Figure 3.14- Non-design skeletal points generation. a) The design triangles connected to non-design sets. b) The interface section of design and non-design in orange. c) the yellow non-design skeletal points as the centers of the interface sections.

The interface section is a contour in 3D, and the non-design skeletal point is the center of this contour, as illustrated in Figure 3.15.

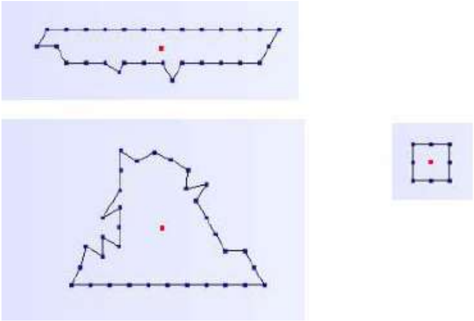


Figure 3.15- The interface sections in 3D with locating their centers.

3.4.2. Generation of the branch from non-design skeletal point to the junction point

The steps of generating a branch from a non-design skeletal point to a junction are mentioned in Figure 3.16.

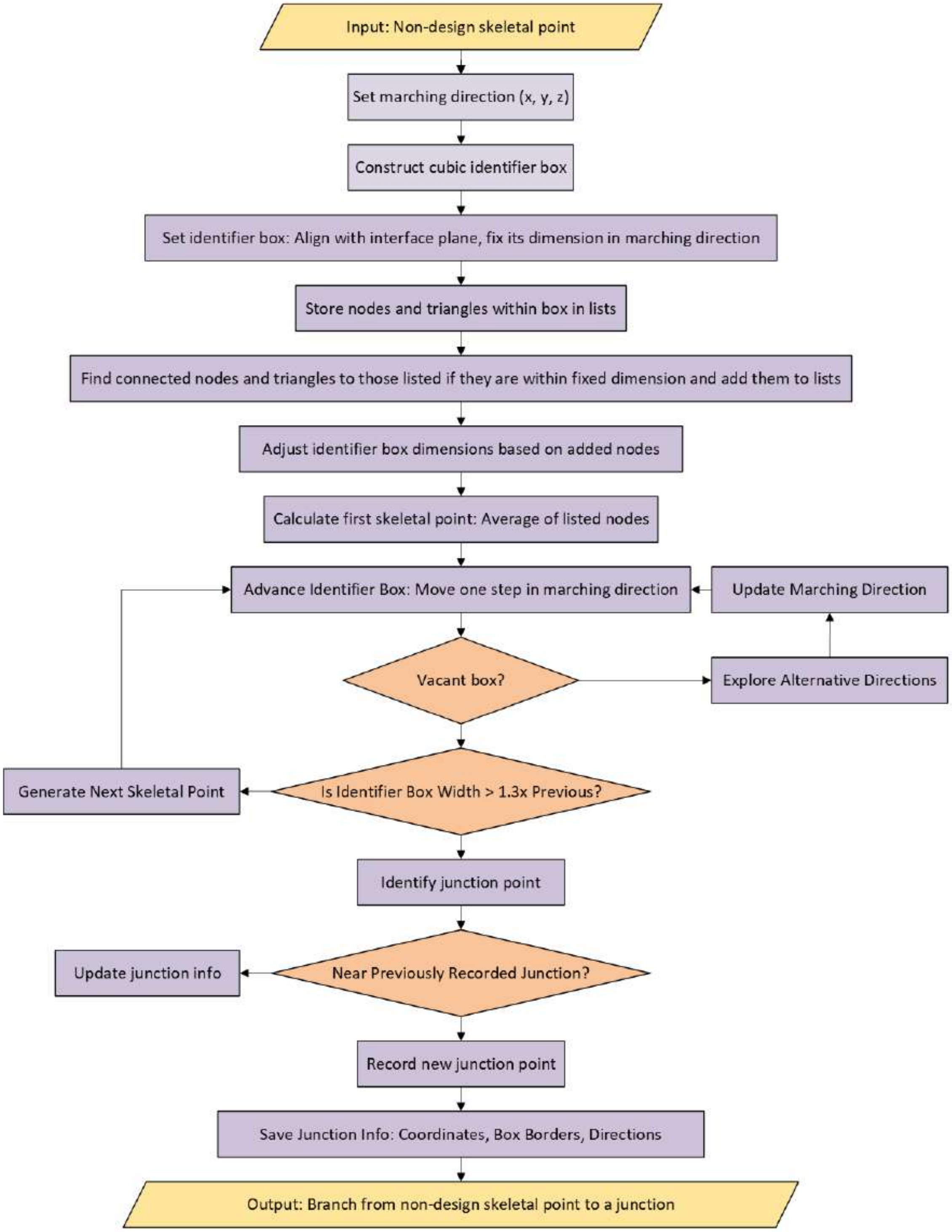


Figure 3.16- The methodology mechanism in generating a branch from non-design skeletal point to junction point.

Accordingly, this branch generation necessitates three guides: two to begin and one to terminate the branch. The beginning guides are the marching direction and the construction of the identifier box as scanning tools. The termination guide is the junction point identification to finish this generation cycle. Therefore, this section has subsections in addition to the generation of the first and next design skeletal points, which indicate how they are generated. Moreover, since the junction points are prominent in the skeleton, their information should be saved to make them accessible whenever required.

3.4.2.1. Identifying marching direction of identifier box

It should go back to section 3.4.1.2 and employ the design triangles set related to the understudy non-design set. In each of these design triangles, there is one design node since they are connected to non-design triangles with a segment. These design nodes are analyzed to determine the direction in which they cross the boundaries of the corresponding non-design set, as illustrated in Figure 3.17a. A single marching direction, chosen from the global reference frame directions (x, y, z), must be selected for each design triangle set. If multiple crossing directions are identified, the algorithm selects the direction perpendicular to the design and non-design interface plane, as shown in Figure 3.17b. Multiple crossing directions can occur when the non-design set is too narrow to fully cover the design boundaries in directions other than the marching direction. For example, one non-design set may bound the design part in the y and z directions, while another is significantly smaller, leading to multiple possible crossing directions, as shown in Figures 3.13 and 3.17. The criterion for selecting the perpendicular direction ensures a consistent choice and rescues the algorithm from blocking. If the perpendicular direction is not viable, the algorithm adapts by selecting an alternative direction from the global reference frame axes (x, y, z). It is worth mentioning that in this study, the perpendicular direction is undoubtedly one of the principal axes of the global reference frame, as the input CAD model is generated based on this orientation.

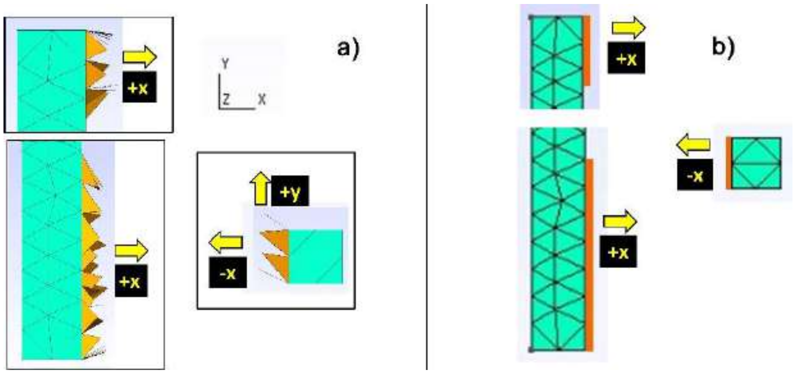


Figure 3.17- Marching direction identification. a) the directions which the design triangles cross the non-design sets. b) the final marching direction for each non-design sets.

3.4.2.2. Constructing the identifier box

The identifier box is a cube at first, a regular hexahedron with equal edge lengths of *lb*, centered at a point offset by half its edge length and aligned with the global reference frame axes. A cube is chosen over a cylinder or sphere because its alignment simplifies min/max bounds checks, enables efficient face-based overlap tests, and ensures uniform resolution in all directions. To determine *lb*, the minimum and maximum coordinates of all design triangles nodes of this set in the x, y, and z directions are identified, as picturized in Figure 3.18a. Let these minimum and maximum be: x_{min}, x_{max}, y_{min}, y_{max}, z_{min}, and z_{max}. The minimum difference between minimum and maximum values in three directions gives the *lb* value, as in equation (3.1). The identifier box is then set by adjusting its center *c* in such a way that one side of the box is placed on the interface section of design and non-design, as in equation (3.1) for this figure. So, the minimum and maximum values in the marching direction are revised based on this placement and *lb* value. Therefore, the cube center coordinate averages minimum and maximum values in three directions, as these are border faces of the identifier box, which are aligned with the global axes. Having the cube center and *lb* value, the identifier box can be set, as in Figure 3.18b.

$$
\begin{gathered}
lb = \frac{1}{2} min(x_{max} - x_{min}, y_{max} - y_{min}, z_{max} - z_{min}) \\
c = (x_{min} + \frac{lb}{2}, \frac{(y_{min} + y_{max})}{2}, \frac{(z_{min} + z_{max})}{2}) \qquad (3.1)
\end{gathered}
$$

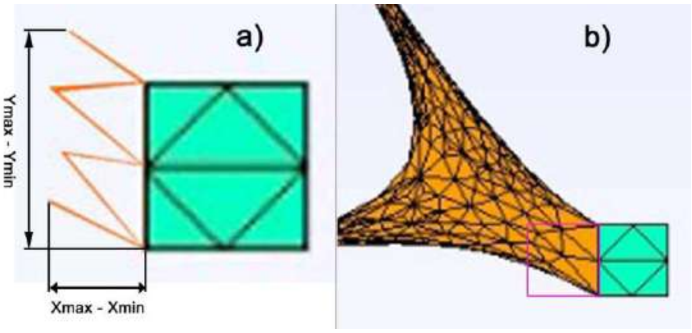


Figure 3.18- Identifier box construction. a) finding minimum and maximum coordinates of design triangles nodes. b) placement of the identifier box.

3.4.2.3. Dimension adjustment of the identifier box

To follow the methodology procedure, the enclosed part of the object should be scanned thoroughly. Thus, the dimension of the pink identifier box in the marching direction is fixed (highlighted in blue), while in the other directions (for real 3D cases) it should be adjusted. For 2D cases used for illustration, as in Figure 3.19, there is one other direction besides the marching direction.

Before processing, nodes and triangles within the cube are stored in dedicated lists. To scan a specific portion of the object between two planes along the marching direction, additional design triangles within this region, connected to those already in the list, are added. The nodes of these new triangles are also included in the node list. These lists have a feature to avoid duplication to ensure no node or triangle can be added more than once. The identifier box dimensions are adjusted according to the added nodes' coordinates, as in Figure 3.19.

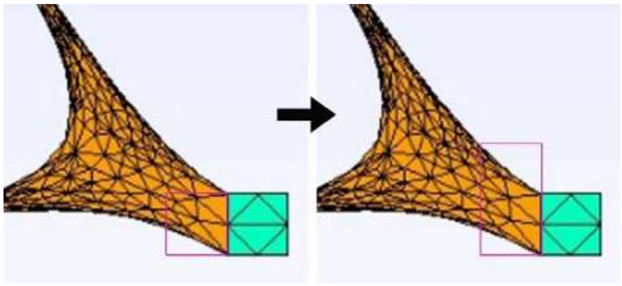


Figure 3.19- Dimension adjustment of the identifier box.

3.4.2.4. Generation of first design skeletal point

Due to this work idea, a skeletal point, denoted *skl*, is the barycenter of *n* nodes enclosed by an identifier box, calculated as the average of the collected nodes' coordinates in the list, as shown in equation (3.2). This yellow skeletal point located in the barycenter of the nodes within the pink identifier box is illustrated in Figure 3.20.

$$skl = \left(\frac{\sum_{i=1}^{n} x_i}{n}, \frac{\sum_{i=1}^{n} y_i}{n}, \frac{\sum_{i=1}^{n} z_i}{n}\right) \tag{3.2}$$

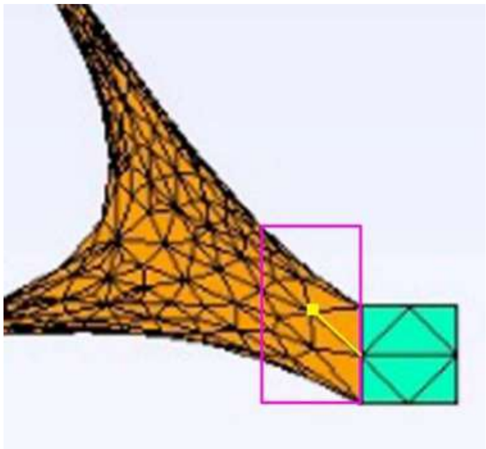


Figure 3.20- First design skeletal point generation.

3.4.2.5. Generation of the next design skeletal point

As a matter of procedure, the identifier box should advance in the marching direction by a fixed step size (one *lb*) to activate the process for the next skeletal point, as in Figure 3.21a. Like what was done for the first design skeletal point, this identifier box is adjusted dimensionally in directions other than marching direction, and the skeletal point is generated, as shown in Figures 3.21b and 3.21c, respectively. The identifier box is no longer a cube after adjustment, but its length in the marching direction is still *lb*.

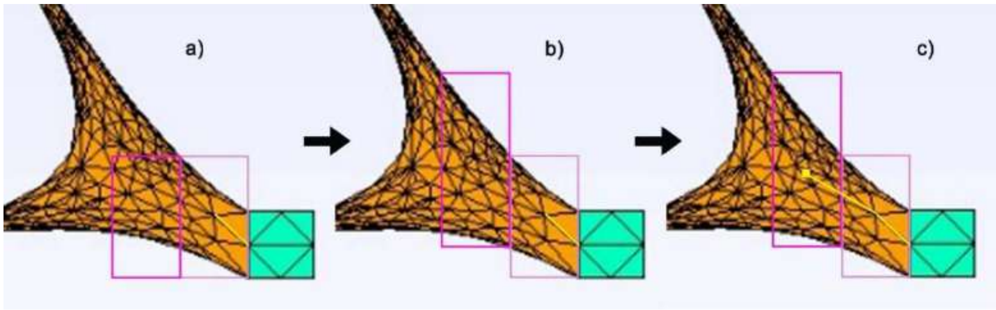


Figure 3.21- Next design skeletal point generation. a) marching the identifier box respecting the marching direction for 1lb. b) adjusted identifier box. c) skeletal point generation.

3.4.2.6. Identifying the junction point

Since the junction point is the meeting place of several branches, its identifier box is wider in at least one of the two directions other than marching direction in comparison with those of the other skeletal points in the branch, as illustrated in Figure 3.22. To identify junctions, the width of the current identifier box is compared with that of the previous one, separately for each of the two directions other than the marching direction. A point is classified as a junction if this width ratio exceeds 1.3 in either direction. The 1.3 proportion limit is a fixed geometric criterion in this study.

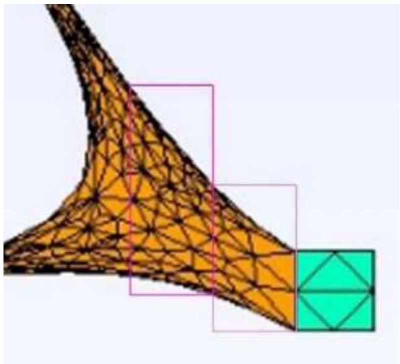


Figure 3.22- The junction point's identifier box in comparison with other skeletal point's identifier box.

Notably, the junction point can occur at any time, even right after the non-design skeletal point. To improve detection reliability, future work should align identifier boxes with local branch directions rather than global axes, minimizing void space in boxes.

3.4.2.7. Saving junction point coordinate, junction box info, and direction

Junction points are special in this methodology since they are the breaking points of stages, either to terminate a branch or to start it, or even to start and finish both. Therefore, saving any information related to them is essential, including their coordinates, the border of their identifier box, and the marching directions of the connected branches. The border of the identifier box is defined by its minimum and maximum coordinates along the x, y, and z axes. At this stage, the junction's direction information corresponds to the marching direction of the current branch. Additionally, the direction in which the identifier box is widest is included in the direction information, as identified during the junction point identification process.

3.4.2.8. Updating saved junction coordinate, info, and direction

When a junction point, previously discovered and recorded, is encountered near the current point, the current point is not considered a new junction. The only action required is to update the information related to that junction point. This case is discussed in detail in Section 3.4.3.2 and is highlighted here as a key step in the procedure.

Since this methodology step is over, the branches from the non-design skeletal points to the junction points are generated, as in Figure 3.23.

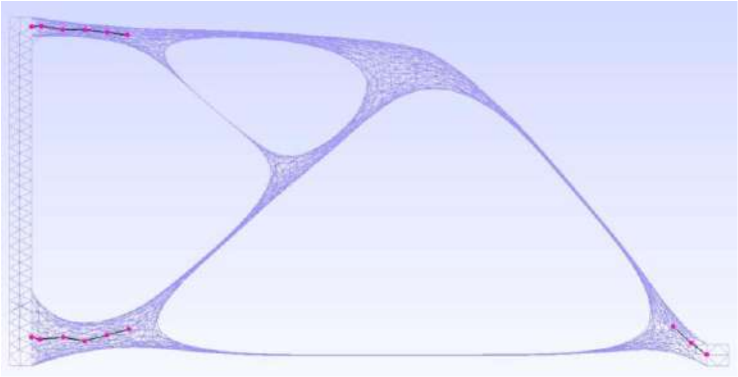


Figure 3.23- The generated branches from the non-design skeletal points to the junction points.

In a condition like that illustrated in Figure 3.17a, a branch can follow one of two possible marching directions. After a while, the identifier box in the selected direction may become vacant, as shown in Figure 3.24b. To address this, the method dynamically updates the marching direction to continue generating the branch, as illustrated in Figure 3.24c. Specifically, the technique allows the identifier box to explore alternative directions other than the current, z direction in this figure, to locate mesh elements and resolve the vacancy. Once a new marching direction, y direction in this figure, is identified, the branch generation process resumes.

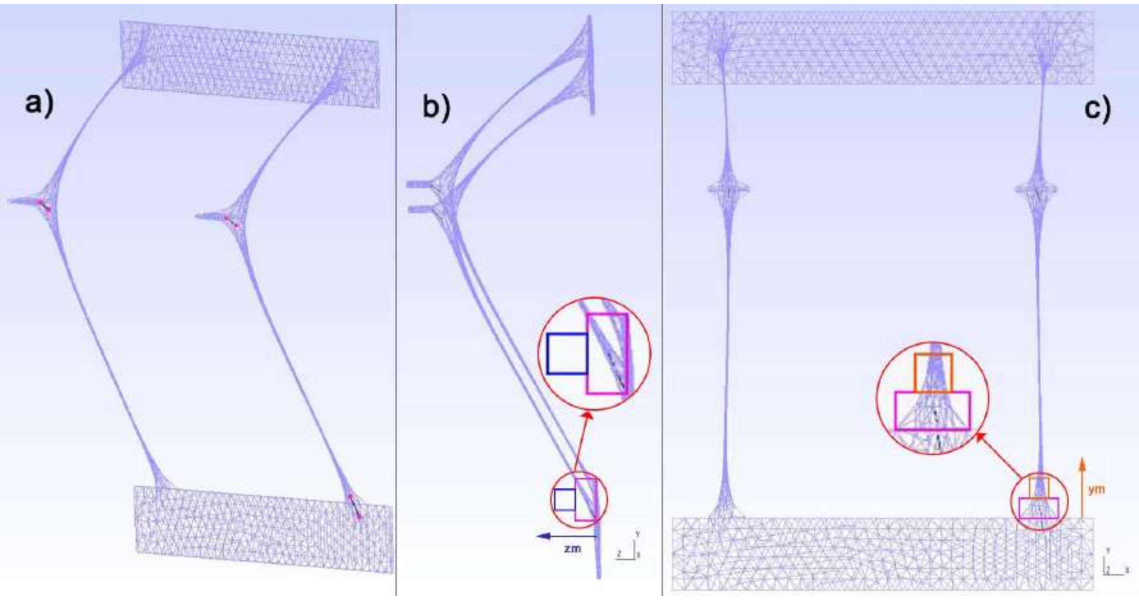


Figure 3.24- Changing marching direction due to the vacant initial identifier box. a) the generated skeleton up to the vacancy. b) the vacant initial identifier box in the marching direction, z_m. c) Changing the marching direction, y_m, to solve the vacancy.

3.4.3. Generation of the branch from one junction to the other

To skeletonize the complex objects with more than one junction or, in other words, to establish a general method, the branches from one junction to the other must be generated, and the designed process is demonstrated in Figure 3.25.

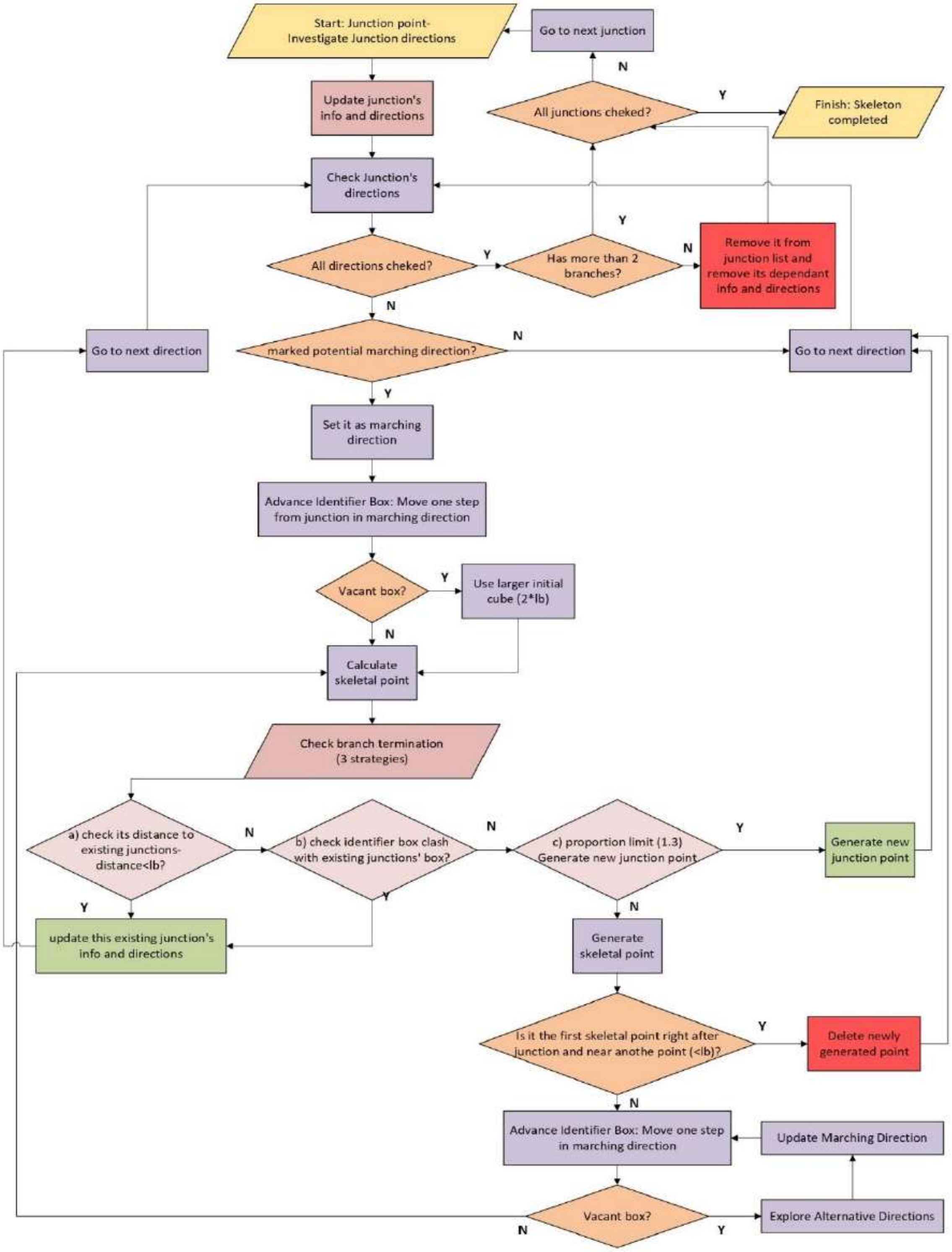


Figure 3.25- The methodology mechanism in generating a branch from a junction point to the other.

Like the initiation of the branch generation from the non-design point to the junction point, knowing the marching direction and the length of the initial cube is essential here, too. The length of the initial cube, $2lb$, stays constant. To find out the potential marching directions, all directions should be explored to mark the potential ones, as illustrated in Figure 3.26. Afterward, these marked potential directions are investigated, and the branches connected to the understudy junction are generated. This procedure is scrutinized in the current section to clarify this part of the method.

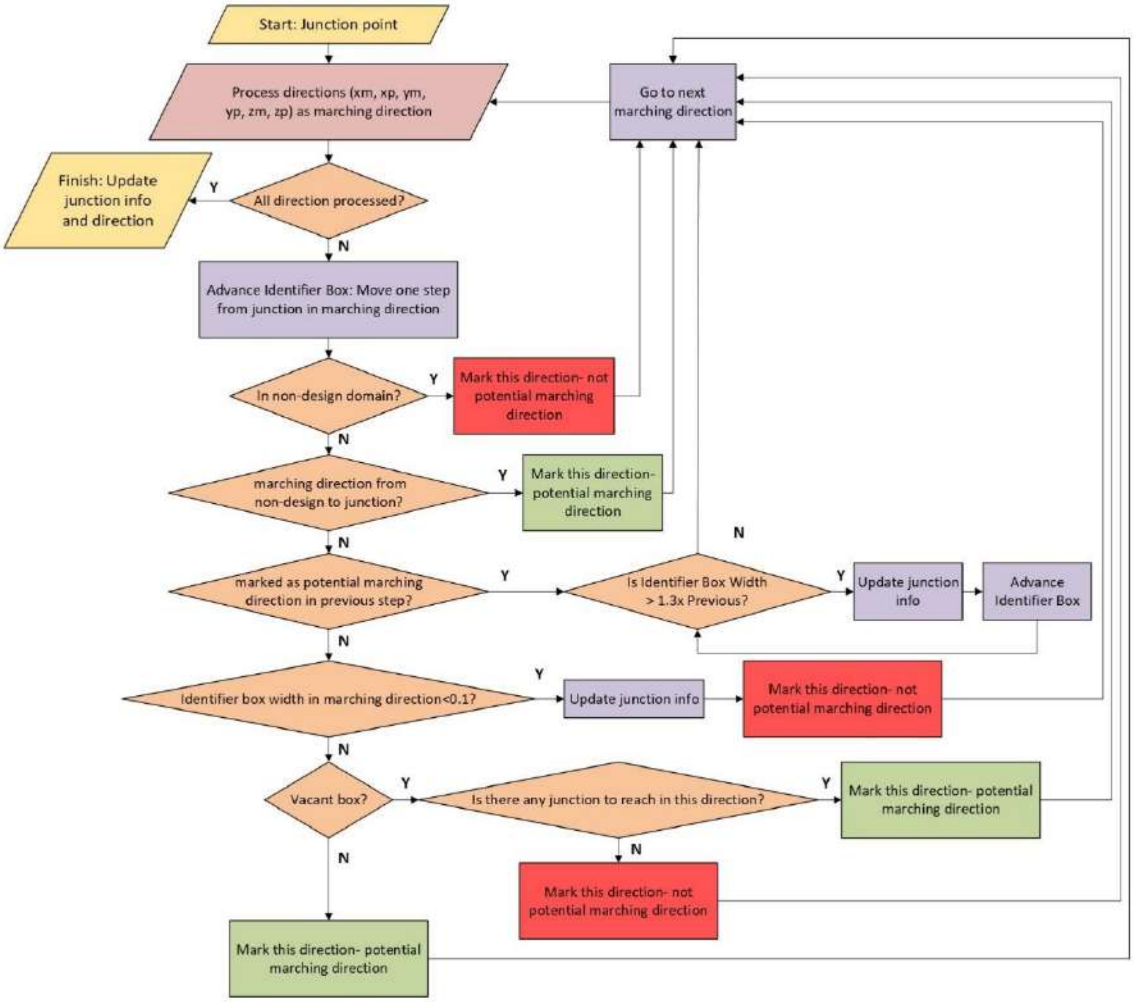


Figure 3.26- The methodology mechanism in updating a junction info and direction.

3.4.3.1. Identifying upcoming branches at junction and update junction box

To identify the upcoming branches, all directions shown in Figure 3.27 should be investigated to mark the potential marching directions, as mentioned in section 3.4.3. The junction's direction information is used to store the marked one. This investigation starts from x_m and ends with z_p, as indicated in Figure 3.27.

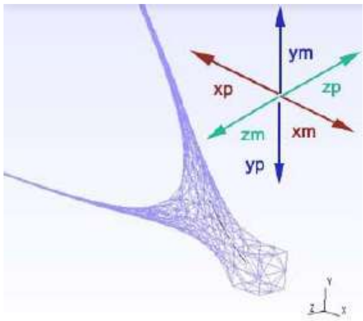


Figure 3.27- All directions to be studied to find the potential marching directions at a junction point.

To start the investigation, place the identifier box one step beyond the junction point in the marching direction. This step determines the directions to study after a junction without generating skeletal points, solely to check for potential branches. For the first direction, x_m, the identifier box is placed at a center $c(x_{max}+\frac{lb}{2},\frac{(y_{min}+y_{max})}{2},\frac{(z_{min}+z_{max})}{2})$. Since this identifier box lies in the non-design area, this direction is not a potential marching direction.

The next potential marching direction, x_p, is the path from a non-design point to the current junction point. This direction is already marked as a potential marching direction for further exploration. Before proceeding to investigate the next direction, y_m, we must verify whether the junction information, which reflects the properties of a junction box, needs updating. The identifier box's dimensions are adjusted in all directions except x_p, where its length, lb, remains fixed. Thus, the junction information may only need updates in this direction. A junction box may contain multiple identifier boxes in this direction, as shown in Figure 3.28. The parameter lb does not directly influence the junction box's size, so it cannot predict how many identifier boxes the junction box contains. An identifier box is considered to have crossed the junction box's borders if the proportion falls below 1.3, consistent with the junction identification process. Otherwise, it is still inside the junction box, and we proceed to the next identifier box in this direction. In Figure 3.28a, the red identifier box represents the initial junction box. The blue identifier box remains inside the junction box's borders (proportion ≥ 1.3), while the orange one lies outside. As shown in Figure 3.28b, the updated junction box, highlighted with green borders, contains two identifier boxes. The junction information is then updated based on this revised junction box configuration.

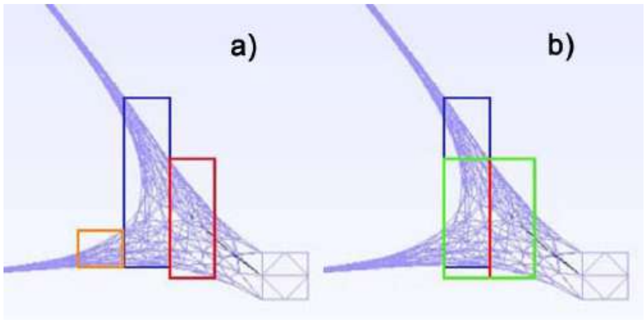


Figure 3.28- Investigation in the marching direction of the generated branch from non-design to this junction. a) The generated identifier boxes around the junction point. b) The updated identifier box of the junction, the junction box.

The next direction to investigate is y, as shown in Figure 3.29. This direction, where the identifier box is widened, as demonstrated in Figure 3.28, is recorded in the junction's direction information as a potential marching direction. For this type of direction, compare the minimum or maximum of the collected points' y-coordinates, with the corresponding value in the junction's information. If the difference is less than 0.1, which is considered negligible in this study, update the junction's information. Otherwise, mark the direction as a potential marching direction in the junction's direction information. Note that the y-direction indicates where the identifier box widens, but the potential marching direction to be marked is either y_m or y_p. For the case illustrated in Figure 3.29, the value of y_{min} is updated in the information of the junction during investigation in the y_p direction, and y_m is marked as the potential marching direction in the junction's direction information.

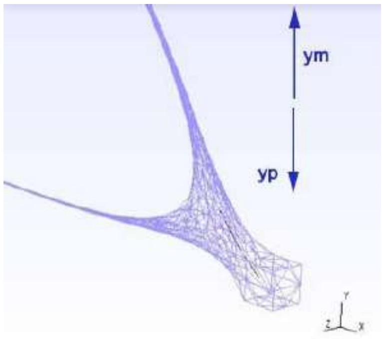


Figure 3.29- Investigation in y direction.

In the global z direction, no branch exists, and the identifier boxes in both the z_m and z_p directions are vacant. This six-directional investigation using globally aligned identifier boxes ensures complete coverage by capturing any branch segment as it has indexes in three global axes, even for arbitrarily oriented junctions. However, this approach may reduce detection sensitivity for branches aligned near the diagonals of the global frame, which is addressed in section 5.2.1. Therefore, the algorithm proceeds to the next junction, as all six directions are investigated for this junction.

An additional condition, illustrated in Figure 3.30, requires checking if the identifier box in the direction where it previously widened is vacant. Since the upcoming branches from this junction should flow to the other existing junctions, the algorithm checks for junctions' presence in this direction. Towards verification, in the y_m direction, no junctions have a y-coordinate greater than the current junction's y-coordinate, as shown in Figure 3.30c. However, in the y_p direction, three junctions have y-coordinates less than the current junction's y-coordinate, making y_p a potential marching direction. Following the defined procedure, the information is updated by modifying y_{max} in the y_m direction. The isometric and xz views in Figures 3.30a and 3.30b illustrate why the identifier box appears vacant on the first check: the upcoming branch is narrow and extends in both the z_p and y_p directions. The logic that the upcoming branches from this junction should flow to the other existing junctions is demonstrated in Figure 3.30c, to mark y_p as a potential marching direction. These potential directions may lead to a branch, but during branch generation, the algorithm verifies their validity. Thus, a branch may not exist in the potential marching direction.

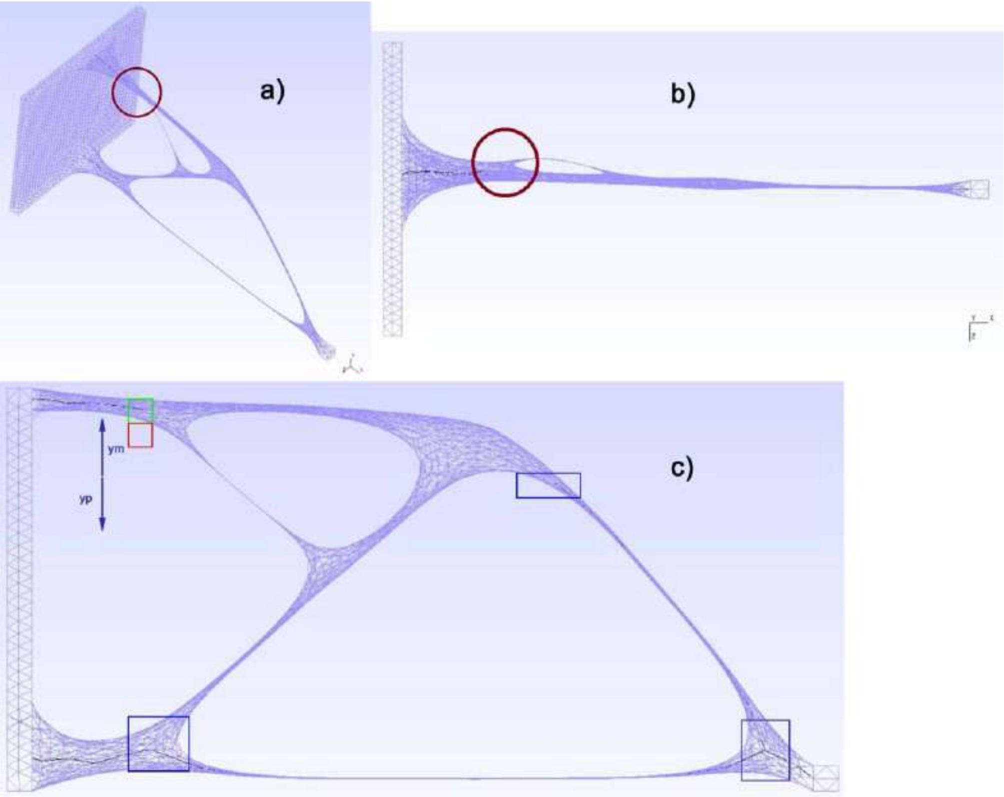


Figure 3.30- Vacant identifier box on the first try in y_p direction immediately after the junction point, while there is a branch. a) isometric view, b) xz view, c) generated junctions to compare their y-coordinates.

After completing the investigation of potential marching directions, the junction's coordinates are updated based on the junction's information. To do so, an initial cube is constructed with the smallest possible edge length, determined as the minimum of the differences between the maximum and minimum values in the updated junction's information, $\min(x_{max} - x_{min}, y_{max} - y_{min}, z_{max} - z_{min})$. The initial cube illustrated in Figure 3.31 has an edge length equal to $z_{max} - z_{min}$. Next, the dimensions of this identifier cube are adjusted, similar to the process used for generating a skeletal point. However, unlike that process, constraints are applied in all directions, as junction's information indicates the borders of this junction box. The updated junction is defined as the center of this modified identifier box, highlighted in green in Figure 3.28b.

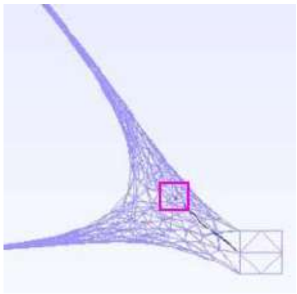


Figure 3.31- The initial cube to update the junction.

All required data to begin the generation of the next branch from one junction to the other are collected.

3.4.3.2. Identifying the junction point

Concerning the termination of the branch, three strategies to identify the junction point are extended in this step: one to generate a new junction point and two others to detect the existing one.

a) Utilizing the proportion limit, 1.3, to predict the junction. This strategy is the same as what is applied in the first step of the methodology in section 3.4.2.6. This strategy leads to generating a new junction point, as illustrated in Figure 3.32. In this figure, the branch starts from JC0, the junction marked in the figure in green, and the marching direction is y_m. The other junctions, JC1 and JC2, generated in the previous process step, are highlighted in blue in this figure.

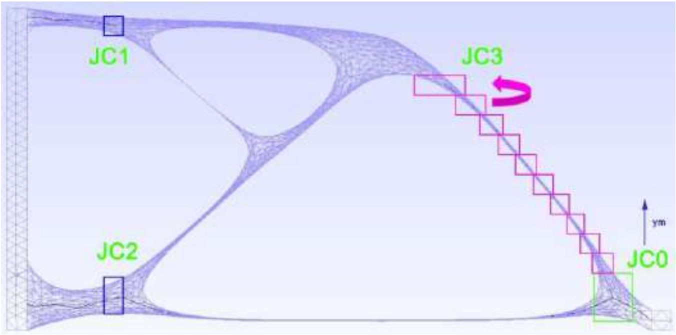


Figure 3.32- The proportion limit technique to identify the junction point while generating the branch between two junctions.

b) Checking the distance to the discovered junctions. Before generating a point, its distance to each junction is computed. If this distance is less than *lb*, the point lies too close to a junction to qualify as a skeletal point, so the nearby junction is updated instead, as demonstrated in Figure 3.33. Here, the branch comes from green JC0 in the x_p direction and ends in updated orange JC2, while it was in blue previously.

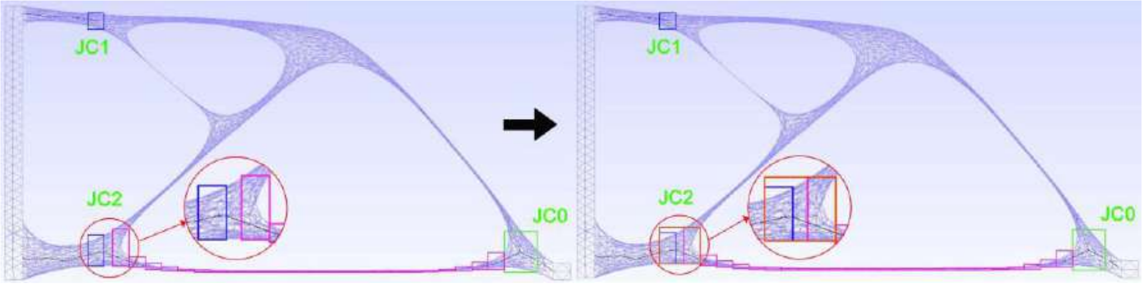


Figure 3.33- The checking distance technique to identify the junction point while generating the branch between two junctions.

c) Comparing the borders of the identifier box with the borders of the discovered junctions. If there is any clash or overlap between the identifier box and a junction box, it is the final of the branch, and this junction and its information should be updated. Three possibilities of clash or overlap conditions between these boxes just in the x direction are shown in Figure 3.34.

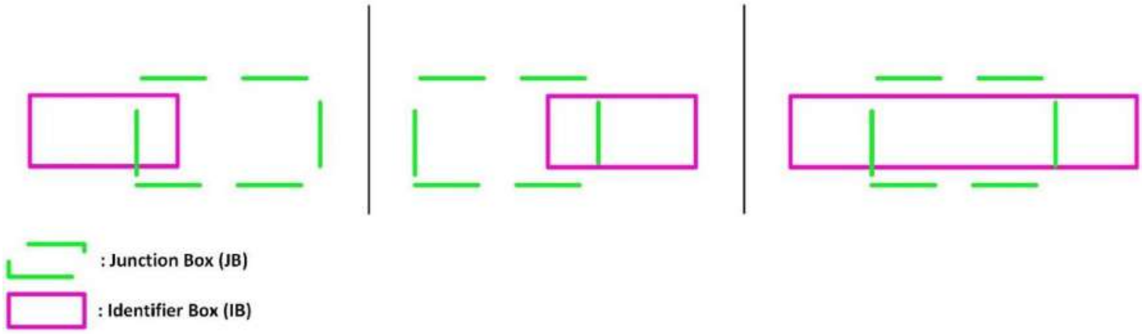


Figure 3.34- The technique of comparing the identifier box border with the junction box border to identify the junction point while generating the branch between two junctions.

These three strategies for identifying the junction point operate independently. Therefore, even if a junction point is close to the current identifier box, the strategies can still function effectively without requiring the identifier box to be widened.

3.4.3.3. Generation of skeletal points up to the next junction point

The data respecting the start and finish of the branch is gathered in previous sections. Thus, the skeletal points of the branch are generated similarly to those reviewed in section 3.4.2.5. Two more considerations are added here regarding the difference in nature of these branches.

First, regarding the vacancy, as mentioned in Figure 3.30, an initial cube twice the routine is constructed, and the process is continued as the routine. The reason for this vacancy is the vacancy of the initial cube because there was nothing to extend and find the connecting triangles and nodes. So, beginning with a bigger initial cube ($2lb$) is sufficient to avoid the vacancy, as shown in Figure 3.35. The pink initial cube is vacant as the upcoming branch is in the -z direction, as shown in Figure 3.30b. In contrast, the violet initial cube is not vacant, since it includes a part of the junction box. The elements outside the two limitations in the y direction, marked by dotted lines, are neither extended nor added to the investigated lists according to the methodology.

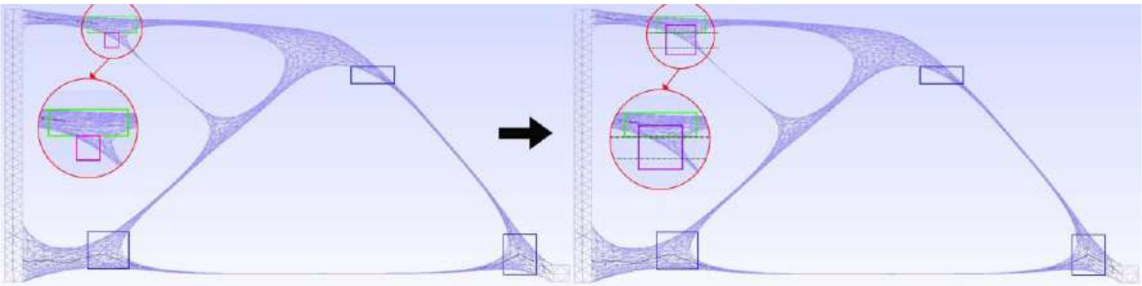


Figure 3.35- Replacing a vacant initial identifier box with a bigger one.

The second is about generating a branch which was generated before. However, how can it happen? Since the principal directions are chosen as the marching directions, one branch can be in more than one direction, as faced at the beginning of the process to initialize the marching direction in Figure 3.17a. To avoid a redundant branch generation, if the first skeletal point of a branch is near to the other skeletal point, this newly generated point is deleted, and the branch generation in this direction is terminated initially. The determined measure of this closeness is the distance less than *lb*. The occurrence of this condition is illustrated in Figure 3.36. This branch is already generated in the y_p direction, and this x_m is redundant.

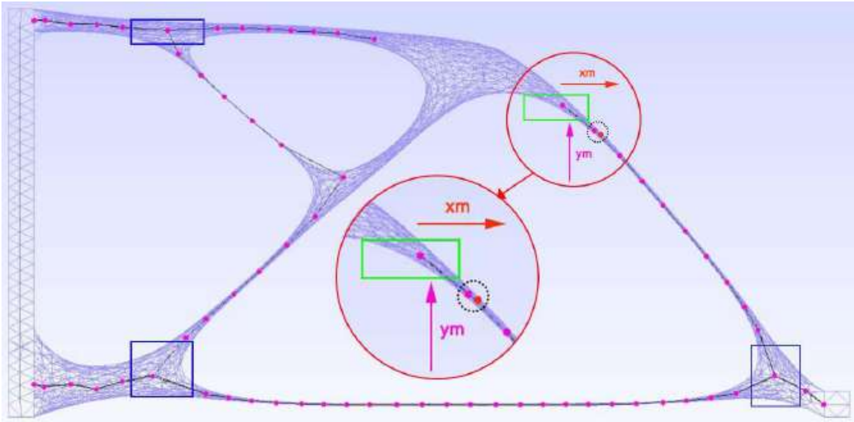


Figure 3.36- A branch regeneration avoidance.

The generated branches from the first-generation junctions are represented in Figure 3.37. The first-generation junctions are those generated through the first process step, the branch generation from non-design skeletal points to the junction points.

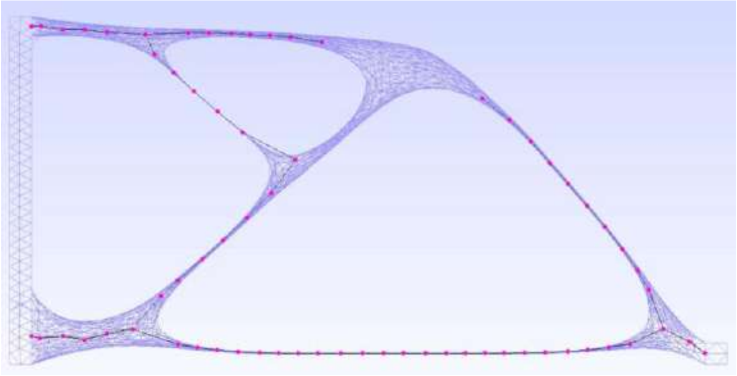


Figure 3.37- The generated branches from the first-generation junctions to the next-generation junctions.

3.4.3.4. Updating junction coordinate, info, and direction

At the final stage of generating a branch from one junction to another, when the junction is identified, all its information should be updated if it has already been generated. This update is about the border of the junction box, information, the direction information of the junction, and finally, the junction's coordinate following the updated information, as mentioned in the last part of section 3.4.3.1. This update needs to be done when the existing junction is detected and one of the last two strategies in section 3.4.3.2 is activated.

3.4.3.5. Removing the junction info for the case of one upcoming branch

When all potential marching directions are surveyed and all branches coming from the understudy junction are generated, it is time to ensure that it was a junction. Since the junction point is the meeting place of more than two branches, if it has only two branches, it was not a junction. Thus, it should be removed from the list of junction points. Its dependant information should be also removed successively. This condition is illuminated in Figure 3.38.

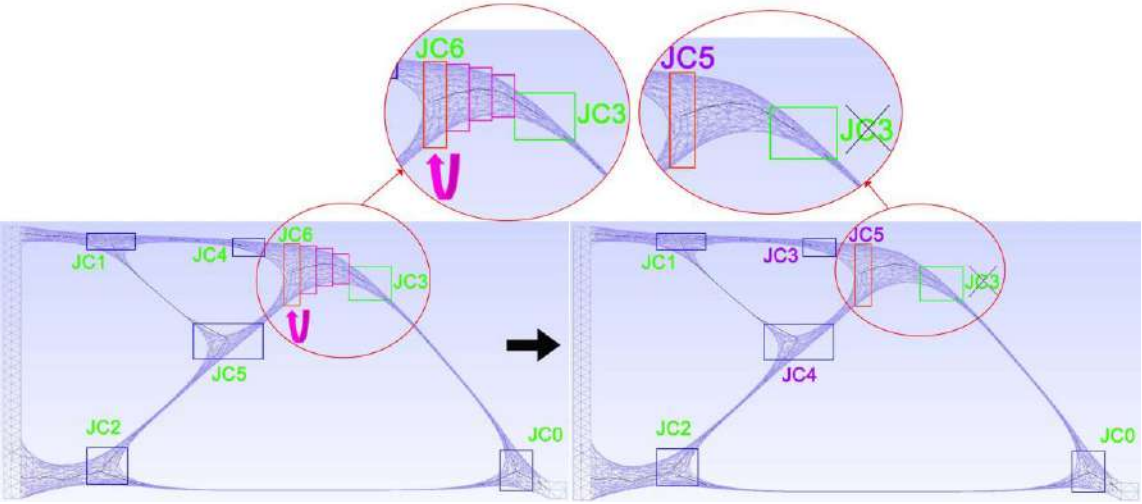


Figure 3.38- Removing the misidentified junction and its relevant information.

The branches starting from the first-generation junctions are generated, as illustrated in Figure 3.37. For the next generation junctions, the identifier box starts from JC3 and moves in the x_p direction to generate JC6. Meanwhile, JC3 and its dependent information were removed, as demonstrated in Figure 3.38. This figure shows three remaining junctions (JC3, JC4, and JC5 in purple) to be processed for branch generation. From JC3, the identifier box moves in the x_m direction to reach JC5. JC3 (purple) and its dependent information are removed, as was JC3 (green) in Figure 3.38. Next, starting from the updated JC3 in Figure 3.39, the identifier box moves in the x_m direction to reach JC4, as investigations in the y_m and y_p directions are complete. Finally, JC4's investigation confirms all branches are generated, leaving no further junctions to process. The resulting skeleton is shown in Figure 3.39.

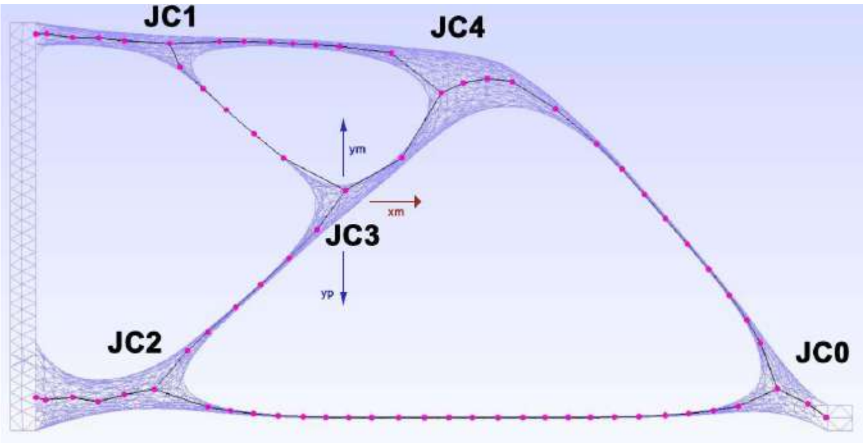


Figure 3.39- The generated curve skeleton by the proposed methodology.

3.5. Conclusion

This chapter presents the stages of the development of an automatic curve skeletonization approach. The required steps to prepare the TO result as an input, including effective parameters in SIMP and post-processing to obtain a beam-like result, are overviewed. Therefore, the methodology containing three principal blocks is explained. The first block is the generation of non-design skeletal points located at the center of the interface between design and non-design regions. The other two blocks deal with the branch's generation; one generates a branch connecting a non-design skeletal point to a junction point, and the other generates a branch between two junctions.

Meanwhile, several techniques are added to identify the junction, and successively, the junction and its related information are saved. The junction's nature is verified at the end of the third block. If the verification confirms that it was not a junction, this junction and its related information are removed from the lists of junctions and their related information.

CHAPTER 4 – NUMERICAL AND VISUAL VALIDATIONS

4.1. Introduction

The developed method produces locally centered skeletal points by design, with each point defined as the barycenter of nodes within a topologically meaningful identifier box. Thus, centeredness is enforced by construction rather than being an emergent property. However, global medial alignment in complex geometries depends on the placement of the identifier box and boundary effects, requiring quantitative validation. Given the literature's limited focus on quantitative centeredness in Section 2.4.2.2, this chapter prioritizes numerical comparison, while including qualitative assessment. In addition to centeredness, compactness is reported for each method. For beam reconstruction, lower compactness is beneficial, as fewer skeletal points per branch yield straighter skeleton segments that more closely approximate the linear beam geometry required for manufacturing. This reduces the geometric adjustment needed during the normalization step in section 5.2.3, where branches are converted to straight beam axes for CAD reconstruction.

First of all, the employed platform is introduced in this chapter. Then, the selected methods to be compared with our method are overviewed. Accordingly, the tools to implement them and the necessary post-processes to obtain the desired skeleton in our platform are presented. The desired skeleton is a skeleton started from non-design parts. Afterward, a quantitative criterion to calculate the centeredness of a skeleton is chosen, and its implementation and functionality are described as well.

Finally, the developed method is validated through the case studies. The CAD model, TO result, and post-processed result are captured for each of them. In addition, the resultant skeletons are compared numerically and visually. Meanwhile, the advantages and disadvantages of the methods revealed through these case studies are highlighted.

4.2. Our platform

The main platform of this project is MAGiC (Maillage Automatique de Géométrie intégré à la CAO (Automatic geometry meshers integrated into CAD)), originally designed as an automatic meshing tool by V. François and J. C. Cuillière. Over the years, research carried out through it by ERICCA members induced an original development environment and database organization, UTM. This environment is designed around object-oriented programming principles (particularly polymorphism and encapsulation) and B-rep concepts. It facilitates CAD/FEA integration and mixed-dimensional modelling and analysis. The integration, linking CAD/FEA entities, is made possible by linking FEA entities with the features in the B-rep topological structure by UTM, as demonstrated in the 3.2 section.

Automatic mesh generation in UTM is developed by our team [52] based on the advancing front technique with the ability to respect density adaptation constraints, guaranteeing design/non-design tags are kept in the case of topology optimization. A SIMP (TO) method based on Bendsøe and Sigmund's work [4] is integrated into the UTM via another attempt in our team [103] while design/non-design subdomains are thoroughly specified. The primary programming language is C++ [108]. Some software is encapsulated in UTM, including Open CASCADE® [109] as a CAD modeller, Code Aster [110] as a finite element solver, and GMSH [111] as a visualization facility. Moreover, as input to the process, STEP files generated from any CAD system are used.

The UTM development in MAGiC makes it feasible to accomplish topology optimization, mixed-dimensional modelling and analysis, automatic simplification of a CAD model to improve mesh quality, and automatic FE model re-meshing following a CAD model modification.

4.3. The selected curve skeletonization methods for comparison

Two curve skeletonization methods are selected to be compared with our method or, in other words, to validate our method. These methods, along with their open-source codes and tools, are presented in the following. Several post-processes are performed on the results that came from the tools to make them somehow equivalent to the skeleton that came from ours and prepared them for comparison.

4.3.1. Method 1: Mean curvature flow

The method [80] is developed by taking advantage of the surface fairing application of the mean curvature flow while minimizing the area, and that is why the same name is adopted for it. The area-minimizing characteristic leads the input mesh to collapse, while local re-meshing at each of these collapses makes the result stable. On the other hand, incorporating the summation of energy terms of smooth, velocity, and medial leads to obtaining a constrained flow and a smoothed medially centred skeleton since the velocity terms fade over time. To explain the reason for achieving a skeleton from the area-minimizing characteristic, the differential geometry of the flow is analyzed. Then, it is demonstrated that the surfaces contract toward the largest curvature in MCF, so MCF bolds local shape anisotropy. Since the method starts from a surface mesh and terminates with a curve, a mixture of surfaces and curves is formed in between, called meso-skeleton, as described in Figure 4.1.

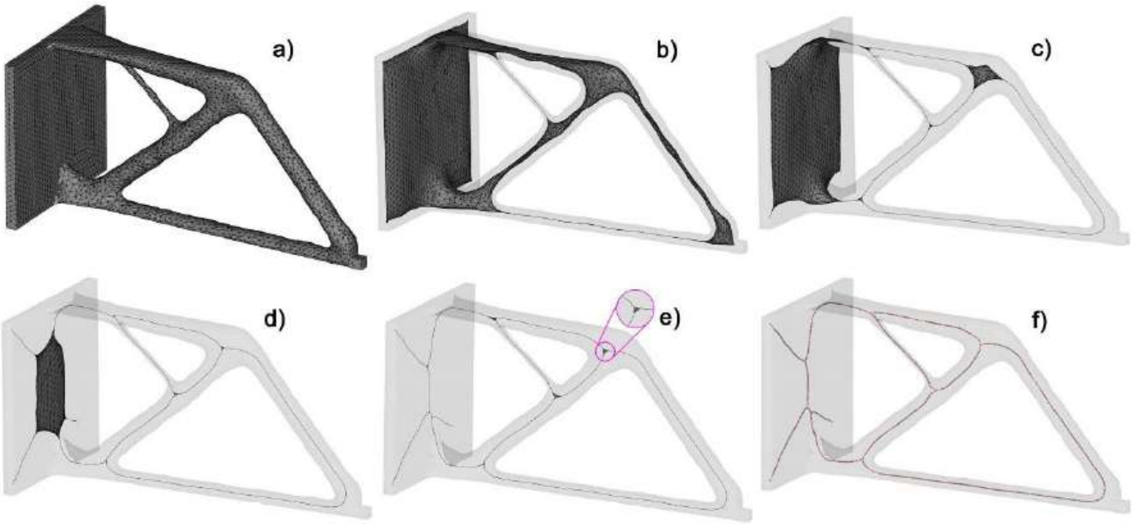


Figure 4.1- The process of obtaining a skeleton from a surface mesh. a) input surface mesh, b, c, d, e) contracted mesh which named meso-skeleton [80], f) the skeleton.

During the review of the literature in the second chapter, this method is overviewed in section 2.4.1.2 and classified among the contraction-based methods, while its well-centeredness is highlighted. Since an open-source tool, Starlab [112], is developed by Tagliasacchi, one of this method's developers, to implement it, this tool is employed here to extract the skeleton for the purpose of comparison. The environment of Starlab, the steps to obtain the skeleton, and the parameters that need to be tuned are introduced in this section. Moreover, the post-processes to read the consequent skeleton from Starlab in MAGiC in the form of a skeleton connected to the center of the non-design sections are brought afterwards.

4.3.1.1. Development tool

Starlab v1.0 [112] is the tool to implement MCF skeletonization, and the package is ready to download as an open-source tool. The package is prepared as pre-compiled Windows, OSX, and Ubuntu binaries. A tutorial video of extracting a skeleton from a surface mesh is also there to guide the user. However, the required steps leading a mesh surface to converge to a skeleton are counted and captured in this section.

a) Input file format

The input file in .off format is produced in MAGiC. It is a smoothed design part connected to the non-design parts, where the design part is smoothed by applying five iterations of Jiao. It is converted to a .off file to prepare it for this method and method 2. Figure 4.2 shows a smoothed triangulated surface input.

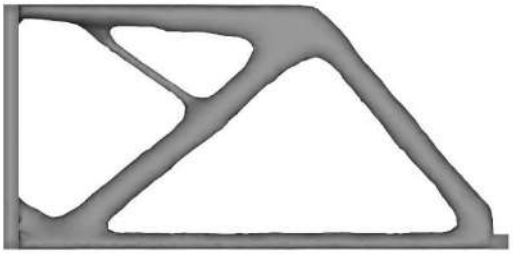


Figure 4.2- An input in .off format.

b) Voronoi based MAT

This method computes Voronoi poles to guide surfaces towards medial axes and generate well-centered skeletons.

c) MCF Skeletonization

In this stage, the skeletonization parameters should be tuned, while how to tune them is debated in the next section, 4.3.1.2. Then, the skeletonization process performs iteratively by each click of the user, and in a few seconds, a surface mesh in the configuration of a skeleton is realized.

d) Conversion of Surface Mesh to Skeleton

Then, the user selects this surface mesh to be converted to a skeleton.

e) Output file format

This skeleton should be saved in .cg format.

4.3.1.2. Parameters

This method introduces four crucial parameters: w_L, w_H, w_M, and ε [80]. However, three of them need to be tuned as the three first parameters are related. The ratio of w_L/w_H controls the velocity of the contraction, and the ratio of w_L/w_M controls the smoothness of the medial. The parameter w_M controls the intensity of surface attraction to the medial axis. As mentioned before, the energy terms control the curve skeleton, and these parameters are related to these terms. The fourth parameter, ε, defines the limit length of the edge which can survive; edges shorter than this amount will collapse during the process. So, it controls the resolution of the resultant skeleton.

The Starlab tool has six parameters to modify, and the tool proposes a default value for each of them. The parameters Omega_L, omega_H, and omega_P are equivalent to w_L, w_H, and w_M as energy parameters, and edgelength_th is ε.

The effect of parameters on the centeredness of the result is studied in this section. This study deals with energy parameters where the value of w_L is always kept at one, and the parameters of w_H and w_M are changed to study attraction, contraction, and smoothness. Since the ε affects only the resolution and does not significantly influence the centeredness, there is no need to study its effect.

Like the main source [80], the effect of changing w_H is studied first, and the result is illustrated in Figure 4.3. The skeleton on the right with the value of one is the best centred, while it is out of the object in the connection of design and non-design on the upper side. So, the value of 0.8 following the middle picture in the figure is selected for it since it is well-centred, and the skeleton is still in the object.

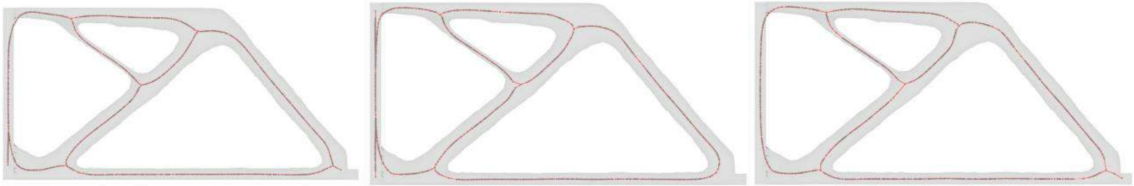


Figure 4.3- The effect of w_H - From the left, their values are 0.4, 0.8, and 1.

The effect of changing the value of w_M is studied and shown in Figure 4.4. As is visible in the pictures of this figure, the difference in the resultant skeletons is not magnificent, while the right one with the value of 0.2 is well-smoothed and well-centred. The left picture should be smoother in the junctions, and the middle and left should be smoother and more centred in the branches.

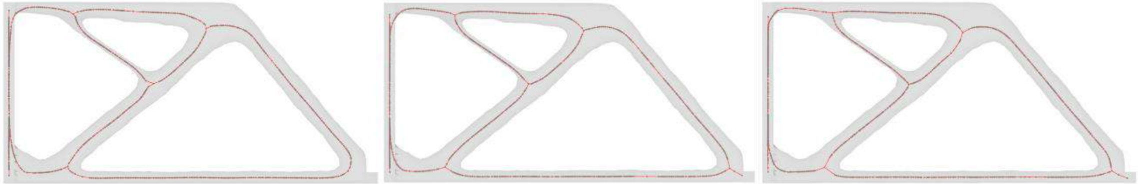


Figure 4.4- The effect of w_M - From the left, their values are 0.2, 0.5, and 0.8.

4.3.1.3. Post-processes

Several post-processes are required to obtain a skeleton connected to the interface of design and non-design and ready to evaluate and compare. These processes include reading the skeleton result in MAGiC, removing the non-design parts, and connecting to the interface of design and non-design. Since this method is used by Amroune, a team member of ERICCA, these post-processes have already been implemented [113]. However, the two first are only employed here.

a) Reading the skeleton result

Since this method is used for validation, the results should be read on our platform where the validation is performed. All mesh elements of the skeleton are read and added to a new mesh, as illustrated in Figure 4.5.

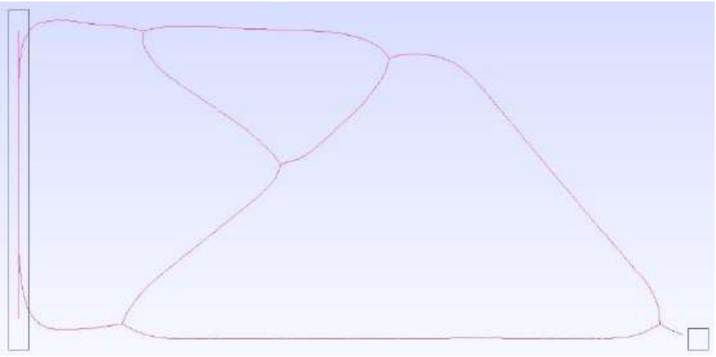


Figure 4.5- Post-processes of method 1 [80]- Reading cg file.

b) Removing the non-design parts

Since the input of this method is the whole model, and obtaining the skeleton of the design part is the objective of this study, the part of the skeleton which stems from the non-design should be removed. As the input of the skeletonization process is produced in MAGiC and the non-design is distinguishable there, removing the non-design part is not a big deal, as shown in Figure 4.6. The black lines in this figure and the left picture indicate the non-design borders where two skeleton branches end on the non-design plane. However, on the right part of the figure, it demonstrated that for one of these two branches, the branch is about to be out of the object, which is the meshed part.

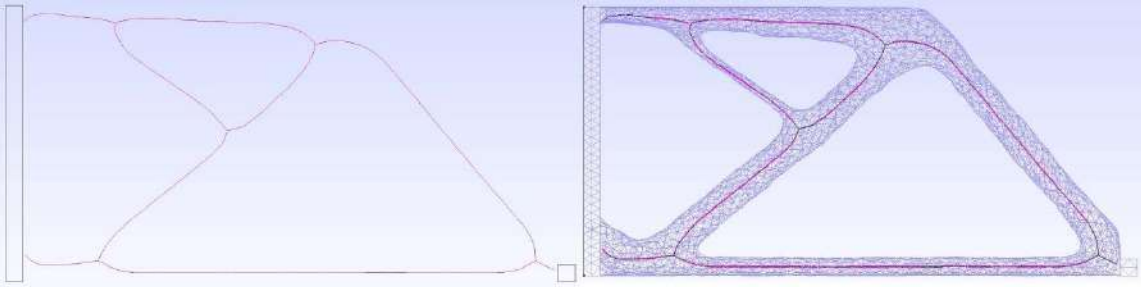


Figure 4.6- Post-processes of method 1 [80]- Removing non-design parts. Left) the skeleton position in comparison with the borders of non-design. Right) the skeleton position in comparison with the meshed whole model.

c) Connecting to the interface of design and non-design

The center points of the interface sections of design and non-design shown in Figure 3.14 are added to the skeleton mesh. Then, each center is connected to its closest point, as in Figure 4.7 left. Afterward, the nodes with one connected segment are searched to delete the short, excessive branches from these closest points; if the found node is not the center point of the interface section, the node and its connected segment are deleted, and it continues to have only the center points with one segment connected. The resultant skeleton is shown in Figure 4.7 right.

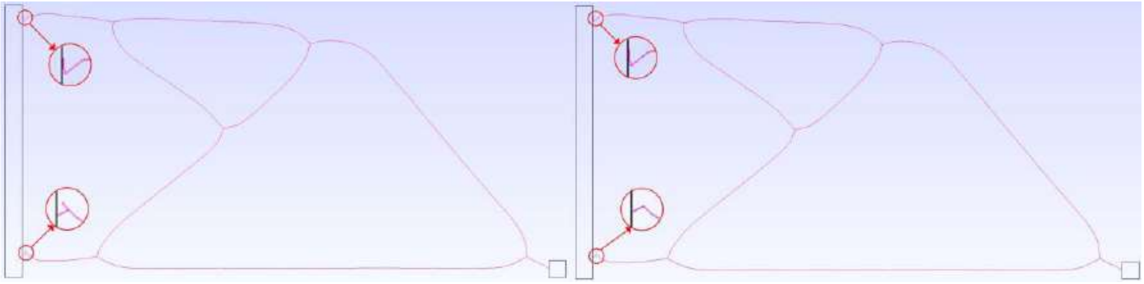


Figure 4.7- Post-processes of method 1 [80]- Connecting to the interface of design and non-design. Left) Connecting center points of the interface sections to the closest points. Right) Removing excessive branches.

4.3.2. Method 2: Laplacian-based mesh-contraction

This method is developed based on two contraction-based works [72, 73], overviewed in section 2.4.1.2. This open-source method can be accessed on GitHub [114]. However, a summary of this method is provided in this section.

The method comprises two successive blocks: geometry contraction and topological thinning. This geometry contraction technique inherits from the Au approach [73], which is iterative quadratic energy minimization. The terms of this energy are Laplacian contraction and an attraction, which is defined by the related weights, w_L and w_H. As demonstrated in Figure 4.8 from the same source, the contraction term removes the geometry details while the attraction term preserves the geometry of the shape. Thus, tuning these two parameters well leads to good results. The initial value of w_H is one and the equivalent value for w_L is 0.2 of the mean neighborhood extent of the point cloud.

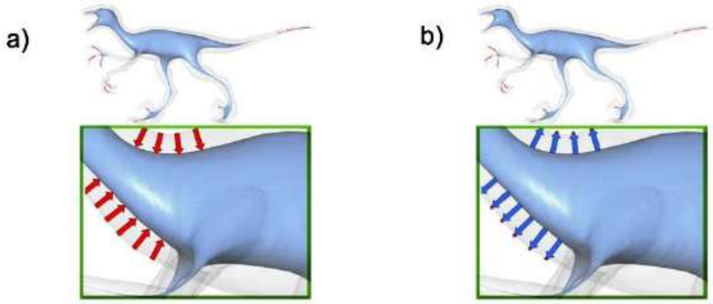


Figure 4.8- Geometry contraction in method 2 [72, 73]. a) Contraction, b) Attraction. [114]

To construct Laplacian for the point cloud, a planar triangulation of the point cloud within a distance of R is needed, and then this triangulation is used to define a weighting scheme. However, it is an iterative function, and it would not be easy to compute at each iteration. So, the local planar triangulation is done once, and the built 1-ring structure is kept. Then, using K nearest neighbours of a point, a tangent plane is built, and Laplacian is constructed through 1-ring of triangulation. Therefore, the cotangent weights Laplacian is used for the contraction. The number of K nearest neighbours, KNN, can be selected from 8 to 30, where 30 is selected here.

In the next block, topological thinning, the contracted points are sampled employing a farthest-point sampling and a ball with radius r, as in Figure 4.9a. The parameter r is 0.02 times the most extended diagonal length of the whole model's bounding box. Then, the samples are connected if their points share common 1-ring neighbours, as in Figure 4.9b. Therefore, since unnecessary triangles are constructed in the last step, these triangles collapse to get the final skeleton, as in Figure 4.9d.

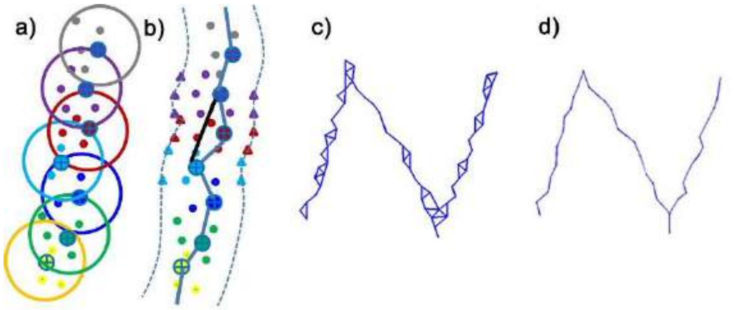


Figure 4.9- Topological thinning in method 2 [72, 73]. a) Sampling the contracted points. b) Connecting the samples. c) Unnecessary triangles constructed in step b. d) Collapse edges of triangles to obtain the final skeleton. [114]

4.3.2.1. Development tool

The code package for this method is available to download on GitHub [114] and can be executed in MATLAB®. It includes MEX files, which are C++ functions callable from MATLAB. These functions are pre-compiled to improve processing speed, and their source code can be hidden. However, the authors provide the source codes for these MEX files [114]. Users should ensure their MATLAB environment includes a compatible compiler to run the code, with further setup details available in the GitHub repository.

4.3.2.2. Required update

The code is updated in three steps: the first is to solve the errors, and the second is regarding the comparability since the skeleton is generated in scale in the main code. The third is related to the input to obtain a precise final skeleton.

a) Update the code to solve errors in recent version of MATLAB

Although the recommended update by the author is applied and the compatibility of MATLAB and the compiler is verified, the code still generates errors. Investigation of errors clarifies that the problem comes from the pointers and their allocation in memory. Since most systems are 64 bits nowadays, and the type of Long for a pointer is not large enough to hold it, Uintptr_t is replaced with Long to ensure the entire address range can be saved. Likewise, for the type conversion and pointer assignment, static_cast<uintptr_t> and reinterpret_cast are employed to ensure that the pointer address is preserved thoroughly, and the pointer is interpreted correctly. Therefore, the code can be run without any problem.

b) Reverse scaling

In the original code, the model is scaled to convert the boundary box of the model to a unit box and move it to the origin. In Figure 4.10, the comparison of the skeleton resulting from the scaled model with the original model is captured.

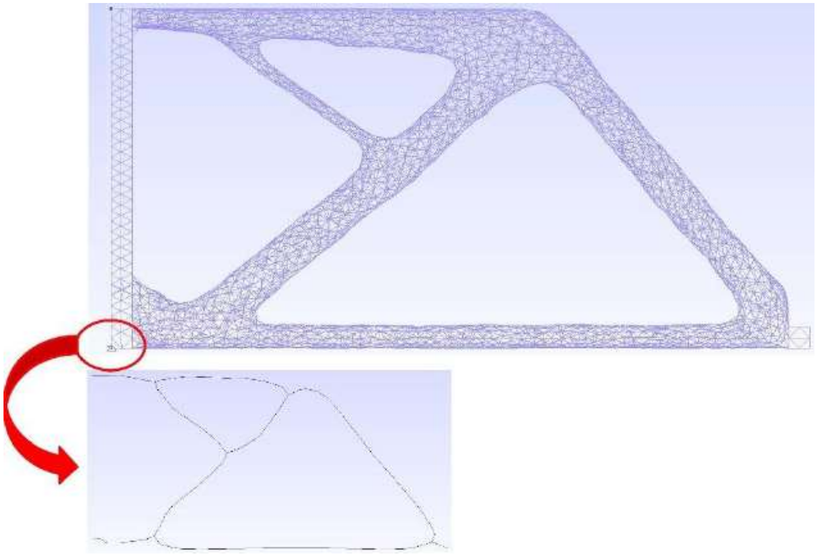


Figure 4.10- Scaling in method 2 [72, 73].

This scaling should be reversed to compare the skeleton with the original model visually. This reverse scaling update is applied to the result [115], while scaling was the first function of the code after reading the input file. To do so, the scale parameters are saved through scaling.

c) Adapt input file

The input file should be in off format regarding generated examples. Nana, a member of the ERICCA team, employed this method for skeletonization [115]. However, he did not use the complete model, including design and non-design. Instead, he constructed a new model by adding the design and non-design interface sections to the design part. The results generated by both models are investigated, as illustrated in Figure 4.11. In the left one, which resulted from the whole model, a part of the skeleton near non-design is out of the object, while this problem is not there in the right one, plus the left one is smoother. Moreover, since the non-design parts are also there, removing triangles and collapsing edges takes more attempts. So, the same technique is employed to obtain smoother results.

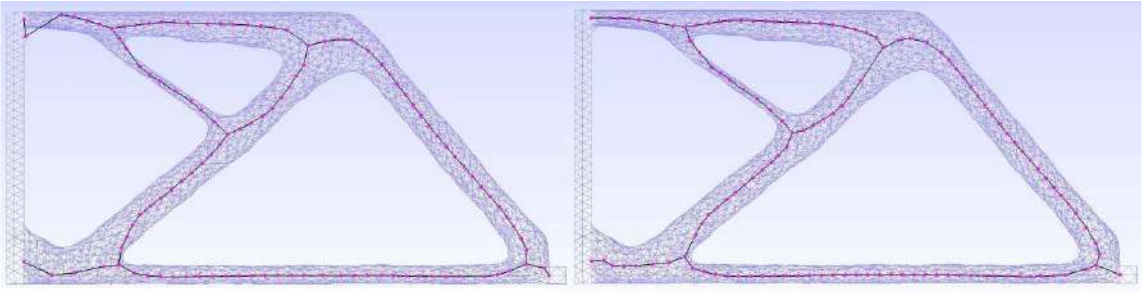


Figure 4.11- The skeletons of TO result generated by method 2 [72, 73]. Left) The whole model. Right) The closed design model, the technique of Nana [115].

4.3.2.3. Post-processes

The raw skeleton generated in the last step still needs some post-processes to finalize. The first is saving mesh elements generated in MATLAB as output to read it in MAGiC. The second is its connection to the interface sections’ center.

a) Saving mesh elements as output

The code gives the result in the figure format, as shown in Figure 4.12. So, the information related to nodes and segments should be read and saved to import in MAGiC and continue the comparison process there. The nodes' information is saved with their coordinates in a text file. The information of segments is saved under the name of connectivity in a text file [115], where each row belongs to a node with the same number, and the numbers of nodes connected to this node are presented in columns in front of this node. So, the number of columns at each raw indicates the number of segments for this node. It is probable to get NAN nodes, but they can be ignored without any consequences since they do not even have any connectivity.

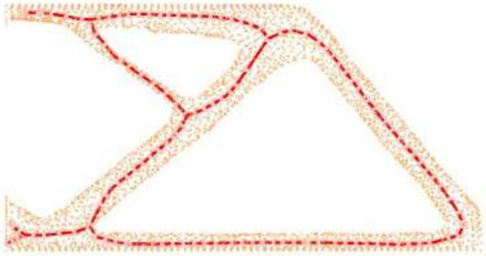


Figure 4.12- The skeleton result of MATLAB code for method 2 [72, 73].

The node and connectivity text files are read in MAGiC to generate a new mesh for this skeleton. Concerning the mesh elements, there are just nodes and segments for a skeleton. The nodes are added to the mesh if the node is not NAN. The segments are added to the mesh by using the connectivity text file. The resultant skeleton is in Figure 4.13.

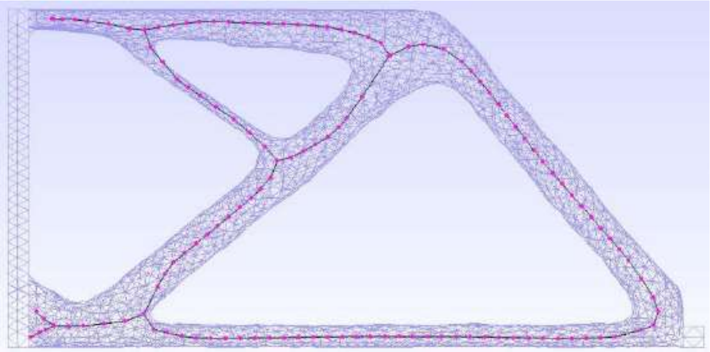


Figure 4.13- The skeleton result for method 2 [72, 73]- not connected to non-design.

b) Connecting to design and non-design interface section center

Like the same section for method 1, at the beginning, the interface section center points are added to the mesh. Then, each center is connected to its closest point. Thereafter, the nodes with one connected segment are searched, and if it is not a section center point, this node and its connected segment are deleted since it is a part of an excessive branch. This search-deletion cycle continues until the only points with one connected segment are section center points. Thus, the skeleton is in Figure 4.14.

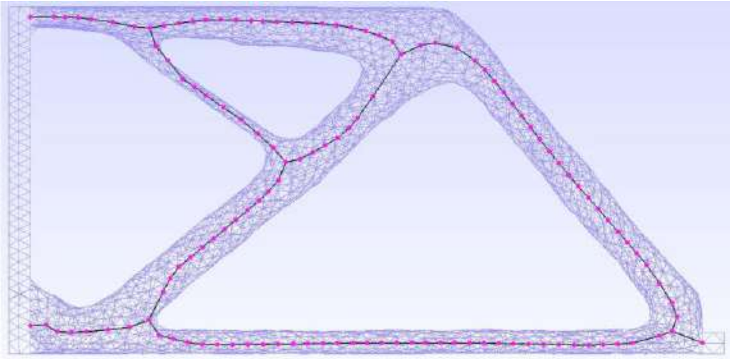


Figure 4.14- The skeleton for method 2 [72, 73] with applying post-processes.

4.4. The numerical validation criterion

In this study, the well-centred is pointed like a precious value for a skeleton aside from the routine, being in the object. That is why quantifying the centeredness is determined as the validation measure even for validating the developed method. The selected criterion [100] to measure the centeredness is among the quantitative comparison methods in section 2.4.2.2. b), the last one [2].

In summary, the criterion is uniformly seeding radial rays in a local normal plane to the CS at each skeletal point, measuring the off-centre for each pair of these rays, and computing the centeredness by equation (2.2). Since the number of skeletal points clearly impacts the centeredness value regarding this formula, it is accentuated under the label of compactness. The criterion is scrutinized below.

To start the procedure, a local normal plane to the CS is constructed on the skeletal point just by defining the normal direction of this plane since the skeletal point is a point on this plane. The previous skeletal point is connected to the next one to obtain this direction, as shown in Figure 4.15a. Then, to seed radial rays, the segments of triangles in the corresponding neighbourhood are collected, and their intersection point with the normal plane is saved if they had any intersection, as in Figure 4.15b. To utilize these intersection points in the computation of equation (2.2), they should be arranged in order on this intersection contour around a point inside the contour. The choice of this point in this step is challenging, although the skeletal point is where the 36 rays are constructed afterwards. However, it would be problematic if it is not located inside since the contour obtained with arranged points would be chaotic, as shown in Figure 4.16. The most prominent challenge in implementing this validation criterion is the occasion of being out of the object, and it has already tried to be avoided several times in this chapter by adopting different strategies. The mean point of the intersection points is a safe option since it is guaranteed to be inside the contour, so it can arrange the intersection points in order.

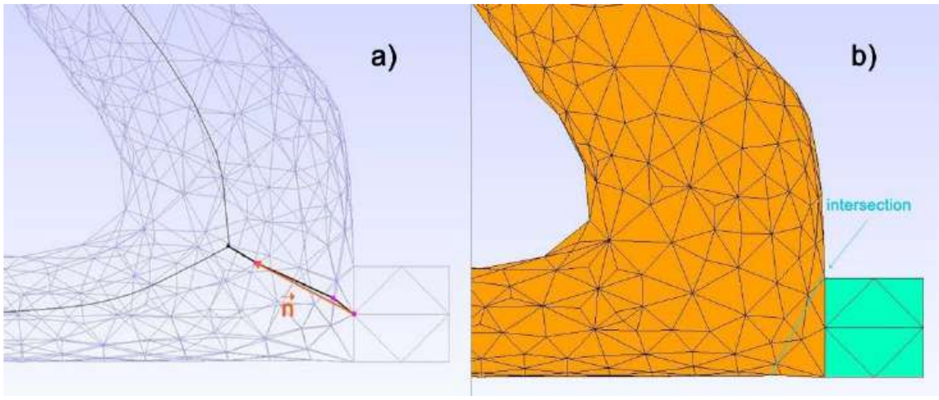


Figure 4.15- The centeredness measure. a) normal direction of the normal plane. b) intersection with the normal plane.

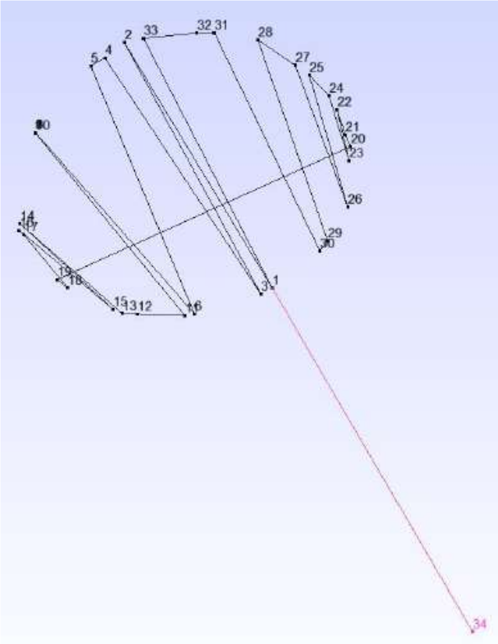


Figure 4.16- The chaos due to choosing the skeletal point to arrange intersection points.

One of them is considered to be the first point of the contour to arrange the intersection points, while the first ray connects the mean point to this first point. Therefore, the other points of contour are arranged according to the angle of their ray with the first ray. A proper contour is constructed as in Figure 4.17a, and it is time to generate the 36 rays. However, these rays in the constructed contour connect the mean point, not the skeletal point, to the intersection point. It matters whether the skeletal point is inside the contour to generate the 36 rays since a special caution is considered in the criterion for being out of the object. If it is inside, generating 36 rays is straightforward, as shown in Figures 4.17a and b; the first ray connects the skeletal and first point of the contour. Theta, θ, is the angle between the generating ray and the first one, and it increases by 10 degrees in each iteration to have 36 rays in 360 degrees. The point corresponding to theta is generated by interpolation between two successive intersection points corresponding to rays with α_0 and α angles, less and more than theta, respectively, as in Figures 4.17a and b.

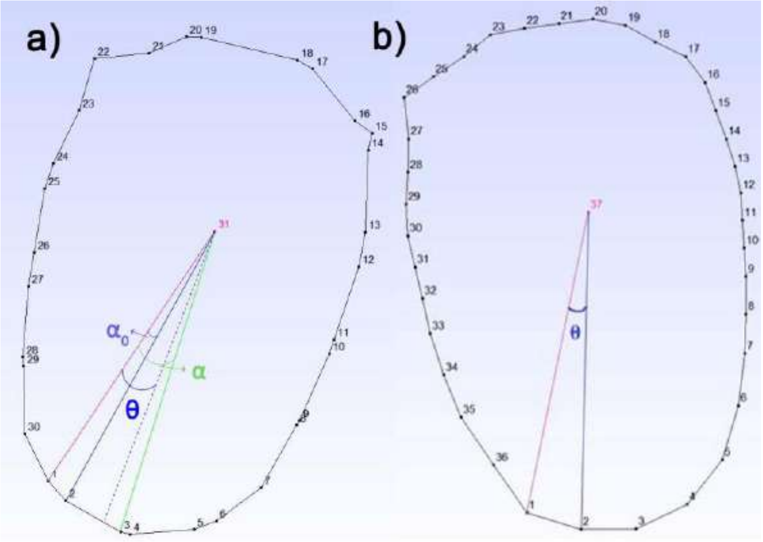


Figure 4.17- The 36 rays' generation for the case of the skeletal point inside the object. a) the constructed contour with intersection points. b) the 36 rays.

If it is out of the object, like in Figure 4.18, a contour slice can be seen as a contour around the skeletal point, and the rest is excessive. As declared in this figure, the intersection points in the concave slice are labelled to preserve in generating 36 rays. The same interpolation technique generates 36 rays while adding two cautions. The first is regarding the first ray; if the corresponding point is not in the concave slice or, in other words, it is not labelled to preserve, it continues to go to the next point to find a point in this slice. The other is when the next point is in the convex slice, the current point in the concave slice defines α_0, and to find α, it continues to go to the next point as long as it is still in the convex slice to exit it and find a point in the concave slice. For all θ amounts between these α_0 and α, it is necessary to go back to the criterion to follow its caution. The caution in the criterion is "when a ray is not intersected with the boundary, its related d_j^i is assigned 0.0, because the extracted curve skeleton with MCS may be outside the shape" [100]. So, the length of this ray is zero, which means the corresponding point of this ray is the skeletal point. Therefore, 36 or 18 pairs of rays can be generated whether the skeletal point is inside the object.

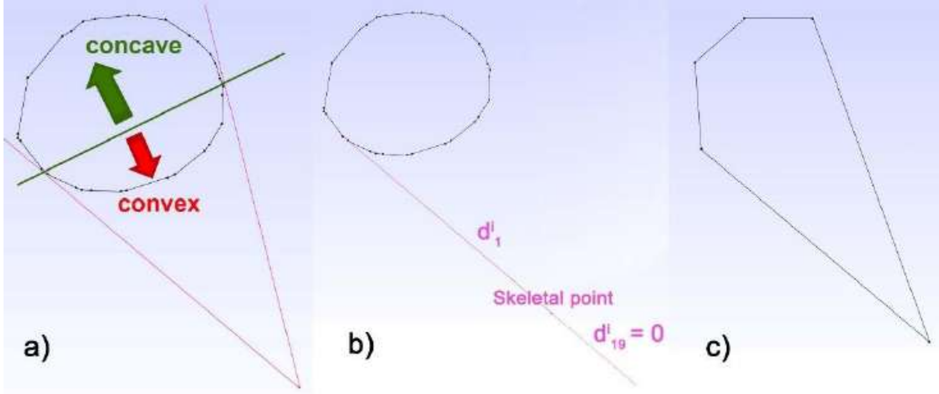


Figure 4.18- The 36 rays' generation for the case of the skeletal point out of the object. a) the convexity of the intersection points regarding the skeletal point. b) the caution in the criterion for being out of the object occasion. c) the 36 rays.

Thereafter, the length of rays is computed, and the difference of ray lengths in a pair is concluded; then, the maximum value for this difference among these 18 pairs is saved for this skeletal point. This value is computed for each skeletal point, and then the mean of these values is calculated and divided by the r value to obtain the centeredness. It is mentioned before that r is the radius of the bounding sphere of the original object.

Some notes should be mentioned: first, junctions and non-design skeletal points are excluded from computation regarding the definition of the normal direction of the local normal plane. Second, the corresponding neighbourhood is a cube in which the skeletal point is in its center and has a defined length. To calculate this length, the distance between the interface section points and the center of the interface section, as in Figure 3.15, is measured for each interface section and the largest amount among all of them is obtained, then twice this value is defined as this cube's length. Concerning the accuracy of the reported centeredness values, four decimal places are considered in this work, as seen in the literature [2, 100].

4.4.1. Simple demonstration of how the criterion operates

Due to the challenge for the skeletal point out of the object and as it is not zoomed in the literature how the centeredness can be measured for this case, a simple circular contour is opted to show how the criterion works in 2D when the position of the skeletal point regarding the contour changes. All probable positions are summarized in three categories: the skeletal point inside, on, and outside of the contour. So, a unit disk centred at (0.0, 0.0) is drawn like in Figure 4.19a. For the first category, the skeletal point (0.5,0.0) is placed inside the contour, as shown in Figure 4.19b, and 36 rays are generated. Therefore, the maximum difference for these 18 pairs is 1.0, which indicates the centeredness of this skeletal point. Concerning the second category, the skeletal point is pointed at (1.0,0.0) on the contour. As can be observed in Figure 4.19c, only 17 rays out of 36 have a non-zero length, which completely fits with the criterion. The centeredness amount for this case is 2.0, equal to the circle's diameter. To apperceive the out of the contour's category, the skeletal point (2.0,0.0) is drawn, and 36 rays are generated, which include seven non-zero length rays as in Figure 4.19d. Its centeredness value is 3.0, equal to the maximum distance of the skeletal point to the circle.

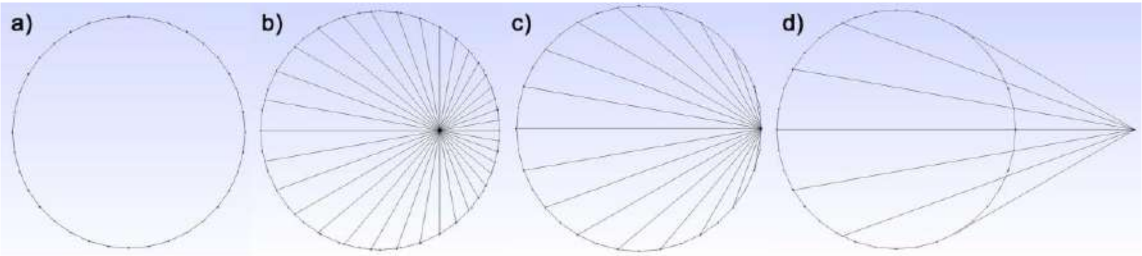


Figure 4.19- How the centeredness criterion works concerning the skeletal point position against the intersection contour. a) the circular intersection contour. b) the skeletal point inside the contour. c) the skeletal point on the contour. d) the skeletal point outside the contour.

4.5. Case studies

Four case studies including cantilever, y form, monster lite, and cantilever with four fixing constraints are employed to validate the developed method visually and numerically in the current section. Moreover, the centeredness of the skeletons generated by skeletonization methods 1 and 2, other than the developed method, is measured to evaluate these skeletons. These methods are described in sections 4.3.1 and 4.3.2, while the centeredness evaluation method is brought in section 4.4. It should be emphasized that the input of all methods is the smoothed TO result since the contraction is a part of the skeletonization process in methods 1 and 2. Thus, a few iterations of smoothing were applied to the optimized result to prepare a proper input. When it comes to visual comparison, the pink points are skeletal, and the black points are the ones out of the object. The effect of contraction on the results of the developed method is investigated in section 4.5.1.2, while the inputs of other methods are still smoothed. Moreover, the effect of mesh size is studied in section 4.5.1.3.

4.5.1. Cantilever

The dimension of the CAD model is 102mm in 51mm in 51mm, as in Figure 4.20. Regarding the boundary conditions, the outer broad face is fixed, and a downward pressure of 1MPa is applied on the upper face of the small cube, as demonstrated in Figure 4.20. The non-design parts are modelled separately to make them identifiable in the whole model. The cantilever is supposed to be aluminum, so Young's modulus and Poisson coefficient amounts are 69 GPa and 0.3, respectively.

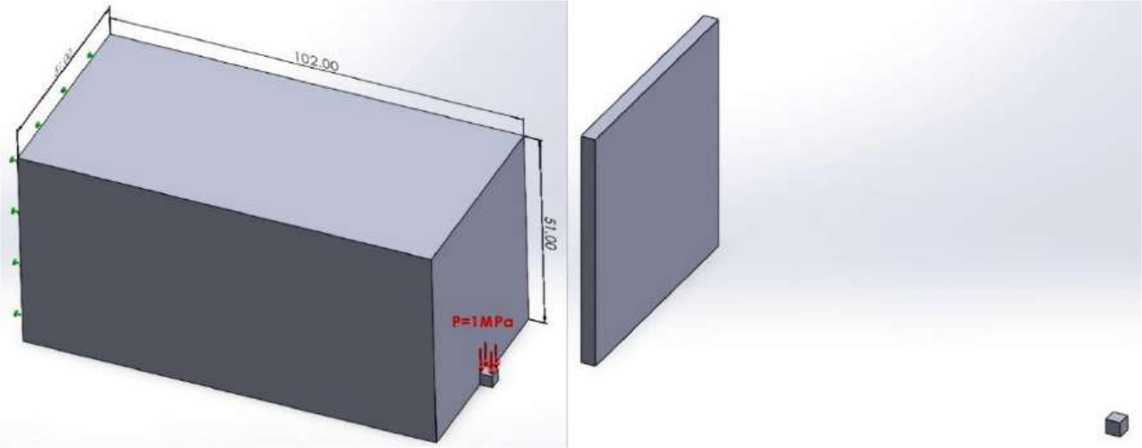


Figure 4.20- Cantilever CAD model. Right) non-design, Left) whole model under BC.

As mentioned in section 3.3.2, three optimization parameters should be tuned to get the beam-like TO result. The tuned values of these parameters are 1.6mm, 0.03, and 0.25 for d_g, f, and ρ_{th}, respectively. The resultant density distribution and the optimized cantilever with a density of more than 0.25 are obtained after 26 iterations and presented in Figure 3.7 since this is employed to capture the methodology procedure.

Figure 3.9a presents the smoothed TO result after applying 10 iterations of Taubin and five iterations of Jiao methods.

4.5.1.1. Numerical and visual validations with three methods

The smoothed TO result shown in Figure 3.9a is employed as the input of the newly developed method and method 1, while its design part closed by the intersection of design and non-design is the input of method 2. Therefore, the centeredness and compactness of each method are reported in Table 4.1. Moreover, the number of skeletal points which are out of the object is presented in this table. This table illustrates that the newly developed method generated the most centered skeleton with the fewest skeletal points since the amount of E_{center} is the least for this method without any point out of the object. In contrast, the resultant skeleton from method 2 has a skeletal point out of the object. The position of this point in the skeleton and how it is out of the object are observed in the following.

Table 4.1- The numerical centeredness result of the smoothed TO Cantilever- the inputs of all methods are smoothed with 10 iterations of Taubin followed by five iterations of Jiao.

	Our method	Method 1	Method 2
E_{center}	0.0120	0.0134	0.0165
Compactness	38	753	116
Out of object	0	0	1

Figure 4.21 captures the results generated by three methods in the x-y view. They are merged with the smoothed TO result to make a visual comparison possible. The black point identifies the point that is out of the object.

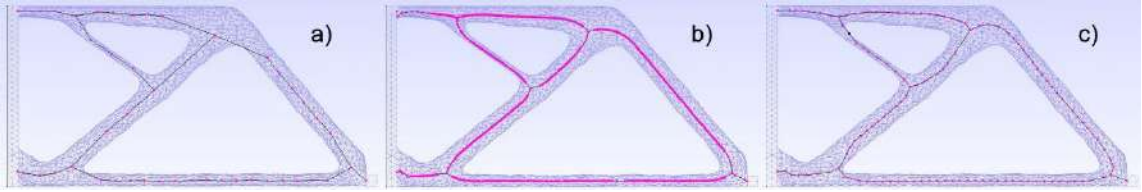


Figure 4.21- The generated skeletons by three methods for cantilever. a) the developed method, b) method 1, c) method 2.

Since it is not visible in Figure 4.21c that the black point is not inside the smoothed TO result, it is shown in Figure 4.22 by using different orientation and even magnifying the closed part. The skeletal point is pointed by a green arrow in this magnified part.

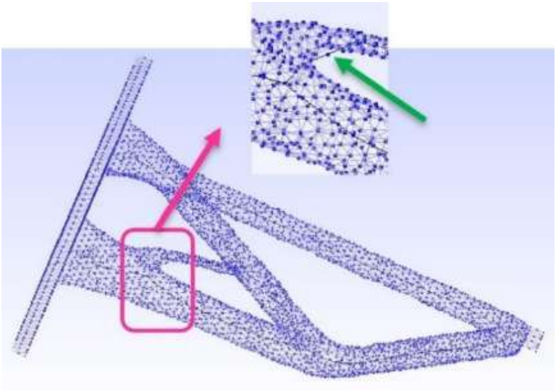


Figure 4.22- The out of the object skeletal point in the generated skeleton by method 2 for cantilever.

Due to the numerical and visual evaluation results presented in Table 4.1 and Figure 4.21, the skeleton generated by our method is the most centred one.

4.5.1.2. The effect of contraction

According to the reviewed literature in section 2.4.1.2, contraction can be taken into account as a key feature to skeletonize in order to gain a well-centered skeleton. Both selected skeletonization methods for validation are also categorized as contraction-based methods to ensure their results are well-centered. Therefore, the effect of contraction on the centeredness of the skeleton produced by the developed method is investigated in this section. In this regard, the smoothed TO result with five iterations of Jiao and the saturated contracted TO result with 100 iterations of Jiao are opted as the extremes of the contraction process due to Figure 3.9. Two other contracted results are also added to study this effect with the flow of contraction, in which 20 and 50 iterations of Jiao are applied. The numerical results for all cases, along with methods 1 and 2, are gathered in Table 4.2. At the same time, the centeredness evaluation measure for all cases is the smoothed TO result, obtained after five iterations of Jiao. This table reports the degradation in centeredness with increasing contraction iterations. The most evident reason to justify this fact is that the contraction is not an obligatory step of skeletonization in the developed method, and it functions either with the smoothed or the contracted TO result.

Table 4.2- The effect of contraction on the numerical centeredness results for the TO Cantilever- the inputs of all methods are smoothed with 10 iterations of Taubin followed by Jiao, where the number of Jiao iterations are indicated.

	E_{center}	Compactness	Out of object
Method 1- 5it	0.0134	753	0
Method 2- 5it	0.0165	116	1
Our method- 5it	0.0120	38	0
Our method- 20it	0.0138	57	0
Our method- 50it	0.0139	56	1
Our method- 100it	0.0215	62	2

So, as a matter of evaluation, it should be ensured that the contracted TO result is at the center of the smoothed TO result; to do so, the produced skeleton should be compared to the contracted TO result instead of the smoothed one. However, it is not logical to compare the centeredness values concluded from these contracted objects to the results that came from the smoothed objects for methods 1 and 2, as the order of amounts is greater for these two due to larger sections at the skeletal points. Therefore, the centeredness is measured twice for each contracted case study: once based on the smoothed TO result and the other time based on the contracted TO result. The reported results in Table 4.3 indicate that the resultant skeletons are well-suited in the center of the contracted TO result; however, this is not the case for the smoothed TO results.

Table 4.3- The effect of contraction on the numerical results stemming from the smoothed and contracted results for cantilever.

skeletonization	centeredness	E_{center}	Compactness	Out of object
Our method-20it	5it	0.0138	57	0
	20it	0.0094	57	0
Our method-50it	5it	0.0139	56	1
	50it	0.0059	56	2
Our method-100it	5it	0.0215	62	2
	100it	0.0055	62	2

To compare them visually, these results are depicted in Figure 4.23 for 20, 50, and 100 contraction iterations, respectively.

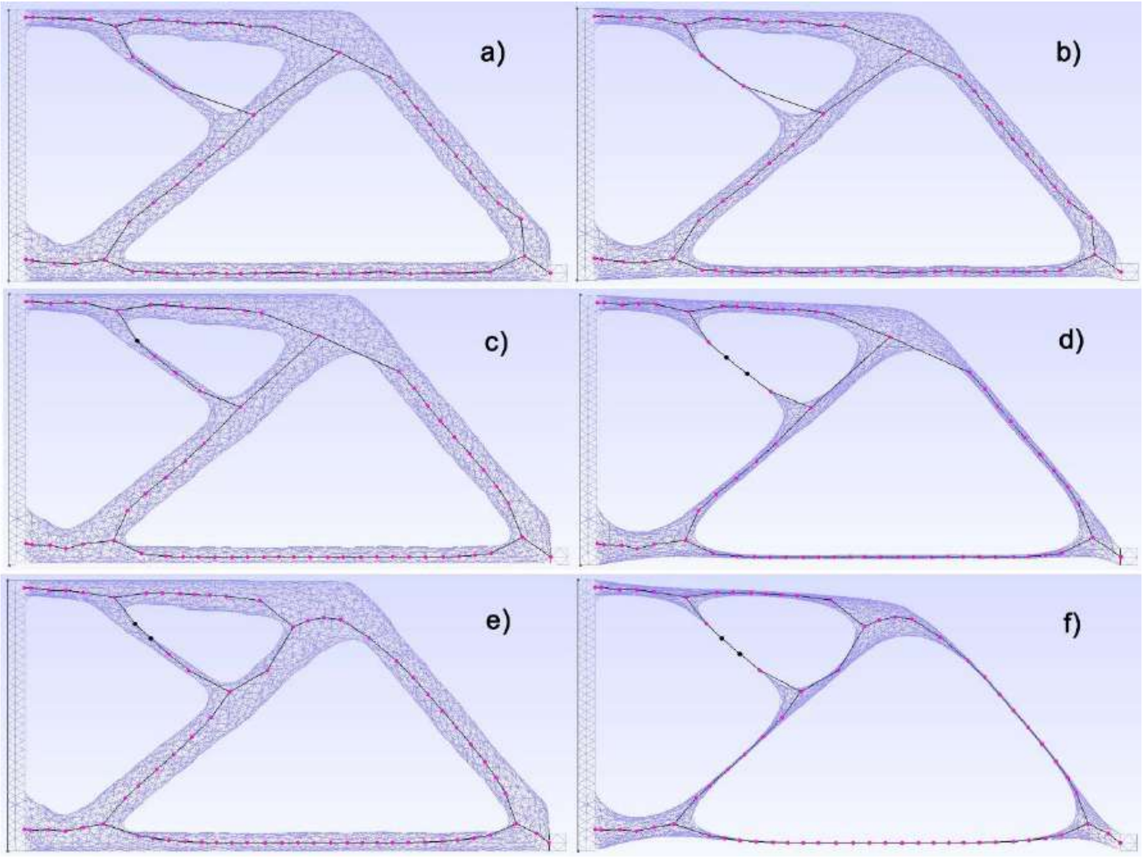


Figure 4.23- The contraction effect on the centeredness where the skeleton stemming from the contracted TO result is compared to a contracted TO result for cantilever. a, b) the skeleton of 20 times contracted TO result- a) compared to five times contracted TO result, b) compared to 20 times contracted TO result, c, d) the skeleton of 50 times contracted TO result- c) compared to five times contracted TO result, d) compared to 50 times contracted TO result, e, f) the skeleton of 100 times contracted TO result- e) compared to five times contracted TO result, f) compared to 100 times contracted TO result.

The centeredness values in Table 4.3 illustrate that the developed method functions well in centeredness since the values get better by applying more iterations of contraction when the centeredness is computed based on the contracted input. Thus, the degradation of the centeredness by applying more contraction in Table 4.2 may be because of the less centeredness of the contracted TO result compared to the smoothed TO result as the original input. It is checked in Figure 4.24 for 20, 50, and 100 contraction iterations, respectively.

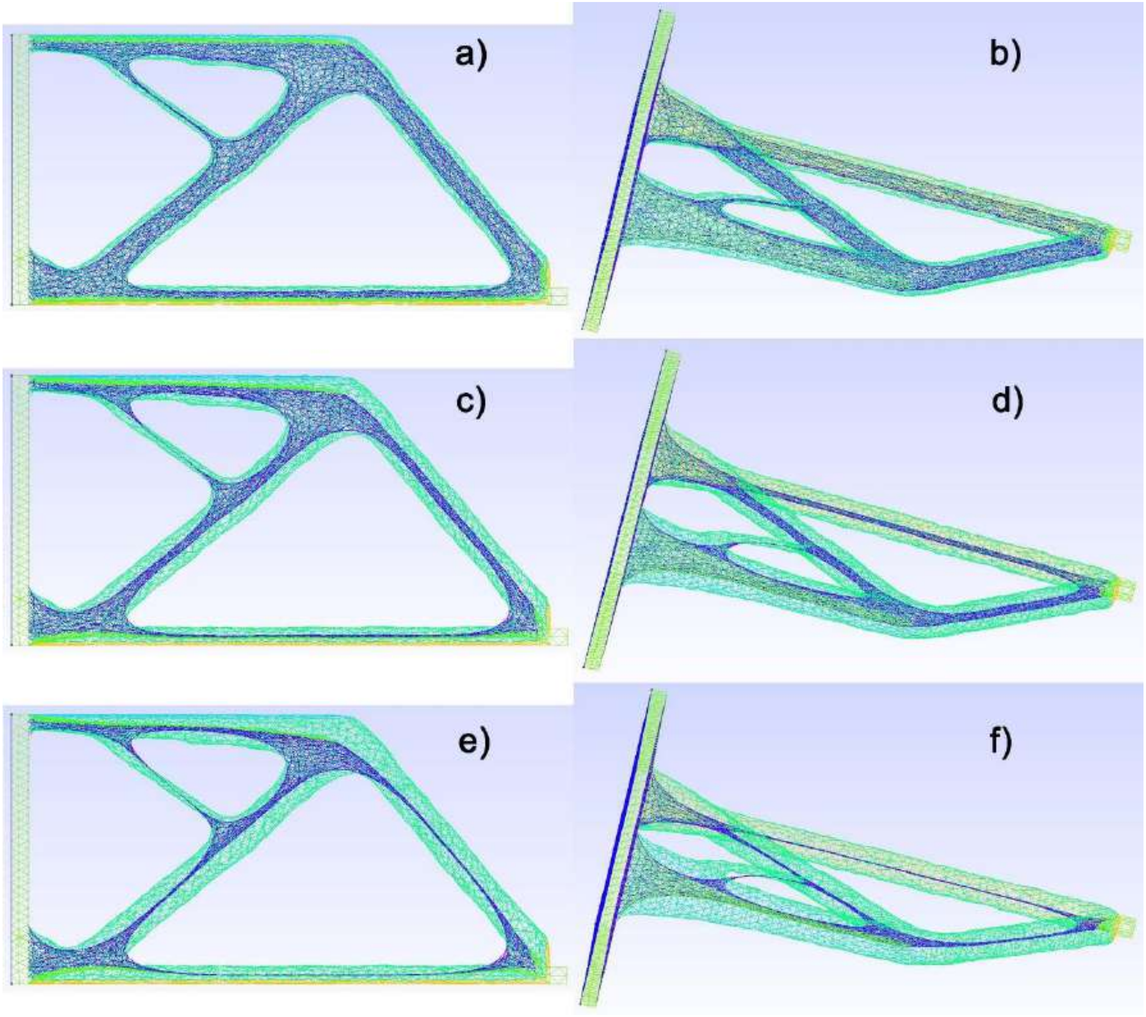


Figure 4.24- The centeredness of a dark color contracted TO result the light color smoothed result for cantilever. a, b) 20 iterations of contraction- a) x-y view, b) isometric view, c, d) 50 iterations of contraction- c) x-y view, d) isometric view, e, f) 100 iterations of contraction- e) x-y view, f) isometric view.

The thinnest branch is the most sensitive part; it went out of the center and even out of the object by increasing contraction, according to figures. So, this branch produced the skeletal points out of the object, as seen in Figure 4.23. It is observed in these figures that the contracted result is not well-centred in the neighbourhood of junctions since the junctions' areas are not contracted as much as the branches. The middle part of the branches seems to be well-centred. Compared to the smoothed TO result, the contracted TO result is less centred by increasing the contraction iterations.

The investigation of this section indicates that adding contraction to the stages of the methodology of the developed method does not make sense when the centeredness of the built skeleton matters and the centeredness of the contraction method is not assured.

4.5.1.3. The effect of mesh size

As already mentioned, the defined parameter for mesh size, d_g, is an effective parameter in the optimization process. Thus, changing mesh size affects the TO result as the input of the skeletonization methods. So, the inputs of all methods and the resultant skeletons change by modifying mesh size, and the produced skeletons by different mesh sizes can not be compared directly, even if they are by the same method. Generally, this parameter is tuned to get a TO result for a specific application, such as beam-like structures. So, it can still be changed slightly to obtain a beam structure TO result. In the current section, the mesh size effect on the obtained TO results and the consequent skeleton from the new input is studied, while how it affects the centeredness is observed.

Accordingly, the mesh size is changed from 1.6mm to 1.4mm, while the other optimization parameters stay unchanged, f is 0.03 and ρ_{th} is 0.025. Then, the TO result and density distribution are picturized in Figure 4.25.

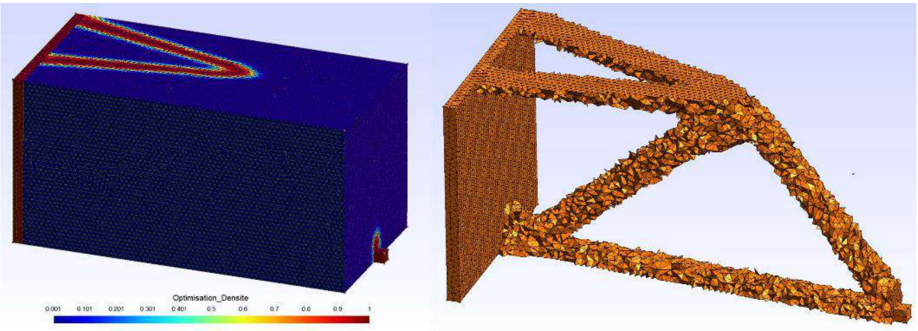


Figure 4.25- Cantilever TO result with the mesh size of 1.4mm. Left) density distribution. Right) elements with densities equal or more than 0.25.

It is noteworthy that the TO result is obtained after 23 iterations. The smoothed TO result after 10 iterations of Taubin followed by five iterations of Jiao is compared to the same smoothed one for the mesh size of 1.6mm in Figure 4.26.

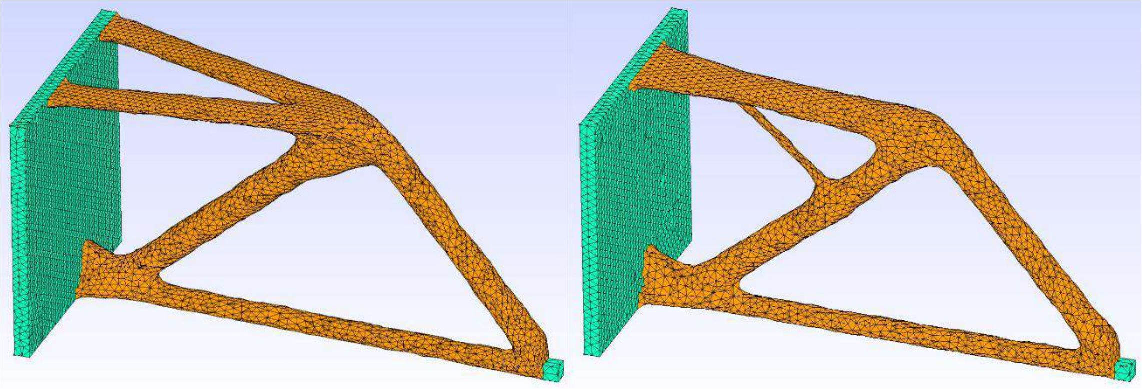


Figure 4.26- Cantilever smoothed TO result. Left) with the mesh size of 1.4mm. Right) with the mesh size of 1.6mm.

Therefore, the skeletons are generated using three methods for a mesh size of 1.4mm. Then, these skeletons merged with the smoothed TO result are compared visually with their equivalent for the mesh size of 1.6mm in Figure 4.27. According to this figure, there are two junctions instead of one using methods 1 and 2 for the case of the upper junction in the mesh size of 1.4mm. The only mentionable thing in comparing the resultant skeletons for one method by different mesh sizes is the fact that there is a skeletal point out of the object for method 2 in the mesh size case of 1.6mm.

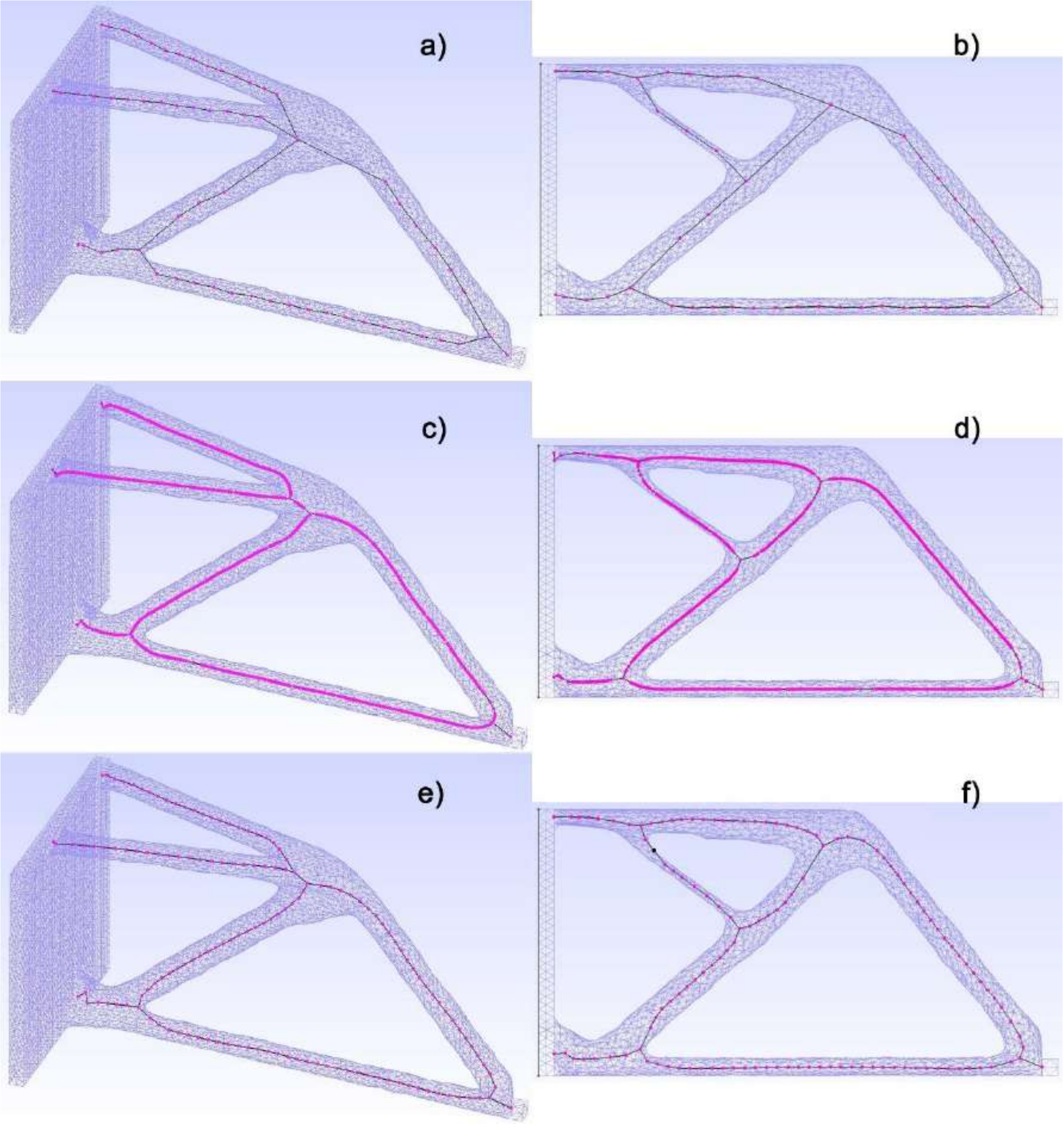


Figure 4.27- The mesh size effect on the centeredness of the generated skeletons by three methods for the smoothed (10 iterations of Taubin and five iterations of Jiao) TO cantilever as input for all of them. a, b) the generated skeletons by the developed methods - a) with the mesh size of 1.4mm, b) with the mesh size of 1.6mm. c, d)) the generated skeletons by method 1- c) with the mesh size of 1.4mm, d) with the mesh size of 1.6mm. e, f)) the generated skeletons by method2- e) with the mesh size of 1.4mm, f) with the mesh size of 1.6mm.

Followingly, the centeredness for three methods with both mesh sizes is reported in Table 4.4. This table indicates that the skeletons generated by all methods are more centered for the mesh size of 1.4mm, while they used more skeletal points in this case except for method 1. Moreover, the ranking of the methods to obtain well-centered skeletons is still the same: the developed method, method 1, and method 2.

Table 4.4- The effect of mesh size on the numerical centeredness results for the smoothed TO Cantilever- the inputs of all methods are smoothed with 10 iterations of Taubin followed by five iterations of Jiao.

		E_{center}	Compactness	Out of object
d_g = 1.6mm	Our method	0.0120	38	0
	Method 1	0.0134	753	0
	Method 2	0.0165	116	1
d_g = 1.4mm	Our method	0.0118	55	0
	Method 1	0.0125	715	0
	Method 2	0.0153	133	0

4.5.2. Y form

The CAD model is in Figure 4.28 with the dimensions of 150mm in 102mm in 10mm. A downward pressure of 1MPa is applied on the lowest face of the lowest part of non-design parts. In addition to this BC, both upper non-design parts are fixed in displacement and rotation, where this condition is applied on the upper face of each. The chosen material is aluminum, so Young’s modulus and Poisson coefficient amounts are 69 GPa and 0.3, respectively.

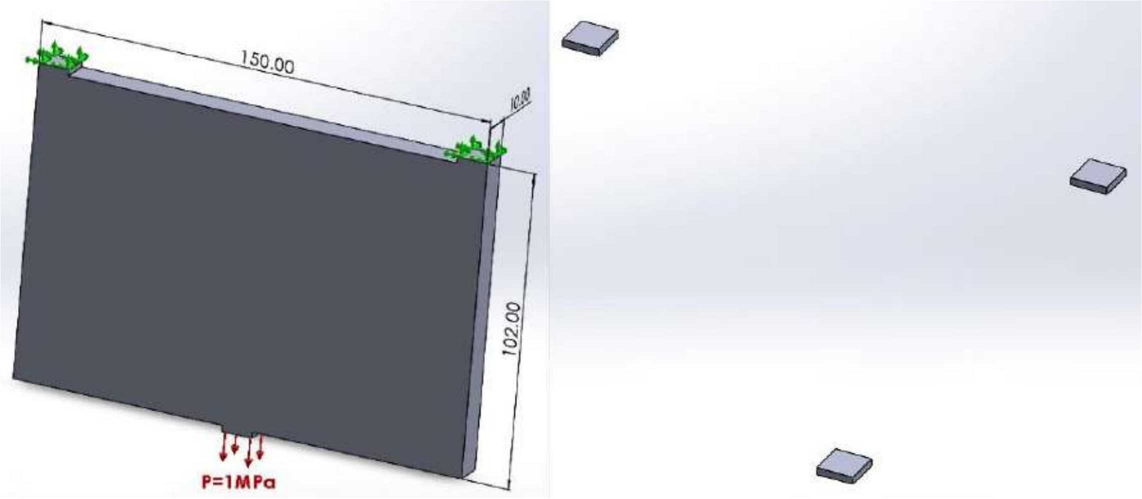


Figure 4.28- Y form CAD model. Right) non-design, Left) whole model under BC.

The amounts of effective optimization parameters in gaining beam-like TO result are 5mm, 0.1, and 0.4 for d_g, f, and ρ_{th} in this case. After 26 optimization iterations, the remaining mesh elements with densities of more than 0.4 are illustrated beside the density distribution capture of the model in Figure 4.29.

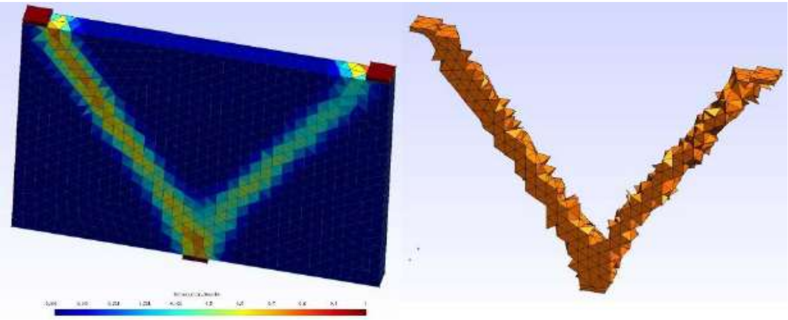


Figure 4.29- Y form TO result. Left) density distribution. Right) elements with densities equal or more than 0.4.

This result was then smoothed 10 times by Taubin and five times by Jiao methods. The smoothed or post-processed result is captured in Figure 4.30, where the green unchanged elements are non-design, and the orange smoothed ones are design.

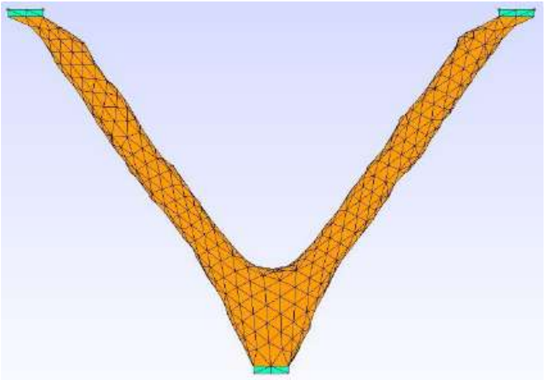


Figure 4.30- Y form smoothed TO result.

4.5.2.1. Numerical and visual validations with three methods

The smoothed TO result is the input of the developed method and method 1, while its closed design part by the intersection of design and non-design is the input of method 2. The smoothed TO result as the inputs for all methods passed the same smoothing process, 10 iterations of Taubin followed by five iterations of Jiao. The resultant skeletons from the three methods are evaluated by considering the smoothed TO result as a measure of evaluation, and the outcome is reported in Table 4.5. The lowest values for the developed method declare that its skeleton is the most centered one with the fewest skeletal points. In addition, none of the skeletons generates any skeletal point out of the object.

Table 4.5- The numerical centeredness result for the smoothed TO Y form- the inputs of all methods are smoothed with 10 iterations of Taubin followed by five iterations of Jiao.

	Our method	Method 1	Method 2
E_{center}	0.0093	0.0110	0.0135
Compactness	13	344	37
Out of object	0	0	0

Figure 4.31 demonstrates the generated skeletons by these three methods where they are merged with the smoothed TO result in the x-y view. It should be underlined again that the centeredness criterion evaluates the skeletal points as it seems in this figure that the

skeleton of method 1 is the best-centered while the numbers in Table 4.5 introduce the skeleton of the developed method.

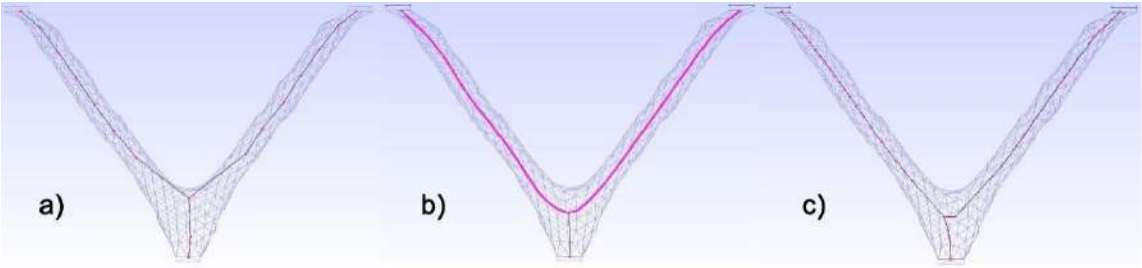


Figure 4.31- The generated skeletons by three methods for Y form. a) the developed method, b) method 1, c) method 2.

4.5.3. Monster lite

This case study is a gym tool that can be mounted on a wall and is a halter holder, as shown in Figure 4.32.

Figure 4.32- Wall-mounted monster lite [116].

It is made of steel A500 with Young's modulus of 206GPa and Poisson coefficient of 0.3. Its assumed dimension is 2306mm in 1245mm in 622.5mm, as depicted in Figure 4.33. Regarding the BCs, the mounting parts are fixed, and a downward pressure of 167.5KPa is applied on each holder since its actual load capacity is 1000lb.

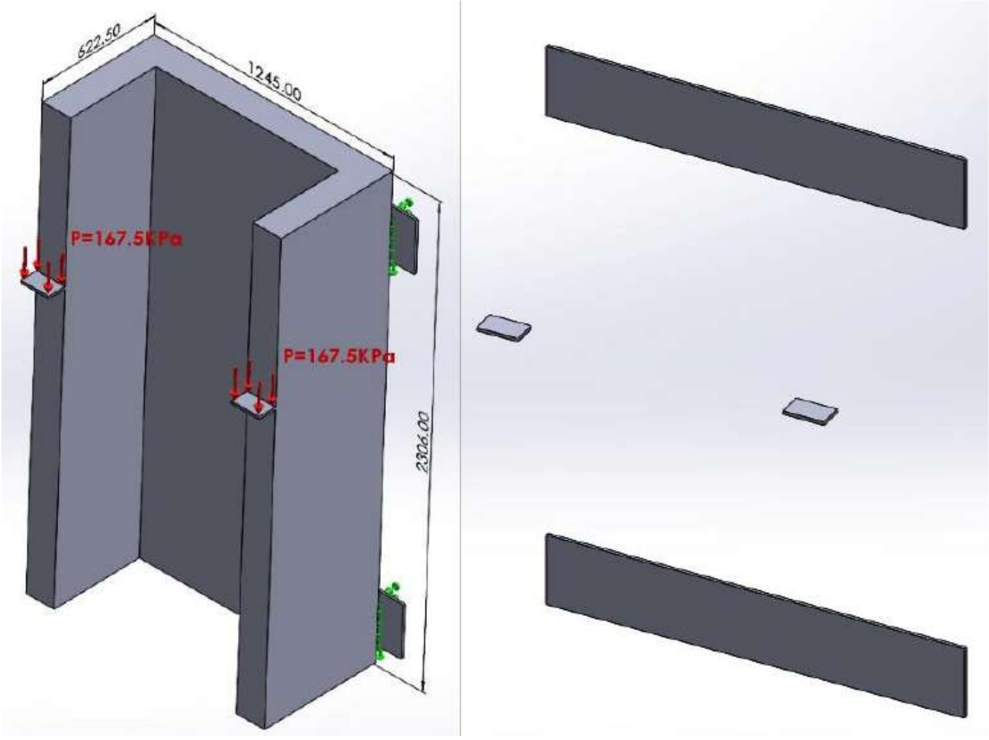


Figure 4.33- Monster Lite CAD model. Right) non-design, Left) whole model under BC.

The beam-like TO result is obtained after 15 iterations by choosing 45mm, 0.07, and 0.35 for d_g, f, and ρ_{th}. The elements with densities equal to or more than 0.35 are demonstrated in Figure 4.34, besides the density distribution of the whole model.

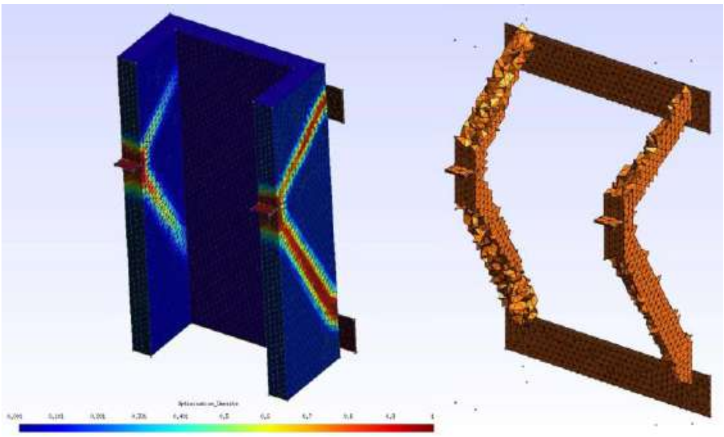


Figure 4.34- Monster lite TO result. Left) density distribution. Right) elements with densities equal or more than 0.35.

Therefore, this result is smoothed by considering the density threshold of 0.25 and without iso-density surfaces with 10 iterations of Taubin pursued by five iterations of Jiao, as emphasized in the beginning of this section. The smoothed result is shown in Figure 4.35.

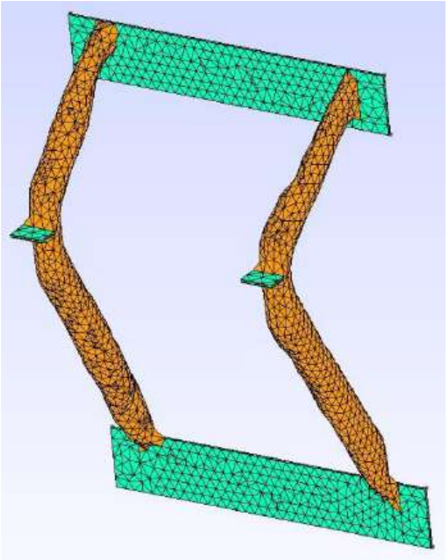


Figure 4.35- Monster lite smoothed TO result.

4.5.3.1. Numerical and visual validations with three methods

The smoothed TO result is employed to produce the skeletons through the developed method, methods 1 and 2. The smoothing process for the inputs of all methods is the same, 10 iterations of Taubin pursued by five iterations of Jiao, as in Figure 4.35. The same smoothed result is the measure of centeredness evaluation for these three methods. The evaluation results of these skeletons are recorded in Table 4.6. The lowest value of centeredness for the developed method indicates the best result for this skeleton, where it has the least compactness. To highlight the centering quality of these skeletons, there is no skeletal point out of the object among these skeletons.

Table 4.6- The numerical centeredness result for the smoothed TO Monster lite- the inputs of all methods are smoothed with 10 iterations of Taubin followed by five iterations of Jiao.

	Our method	Method 1	Method 2
E_{center}	0.0097	0.0118	0.0132
Compactness	20	470	90
Out of object	0	0	0

For the matter of visual validation, the merged version of these skeletons with the smoothed TO result is presented in Figure 4.36. This figure shows a considerable distance from the junctions to the nearest points in the branch for the developed method, which means a wide junction area. It agrees with the methodology since the identifier junction box is wide in the y direction.

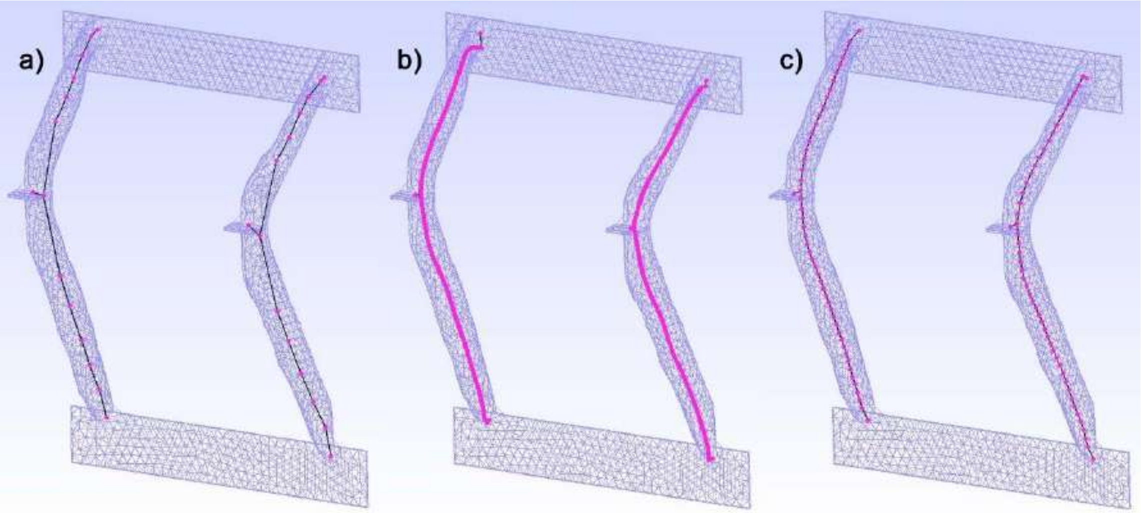


Figure 4.36- The generated skeletons by three methods for Monster lite. a) the developed method, b) method 1, c) method 2.

Regarding the skeleton of method 1, the smooth flow of skeletal points in the branches is violated in the neighbourhood of non-design parts to connect to the non-design skeletal points. These non-design skeletal points on the centers of the interface of design and non-design are added during an additional step to method 1, back to section 4.3.2.3. This violation is because method 1 sees the whole design and non-design as an object, where the non-design sections of this skeleton are deleted later. However, the connections of these two parts are not in the center of their interfaces in the original skeleton.

4.5.4. Cantilever with four fixing constraints

The same Cantilever with some changes in non-design parts is utilized to obtain a more complicated TO result. The dimensions are the same, while the fixed outer board is replaced by four small cubes in the corners of the board, as reported in Figure 4.37. A downward pressure of 1MPa is applied on the upper face of the small cube, and the outer face of these four cubes is constrained in displacement and rotation due to the boundary conditions of the problem. The selected material is aluminum with Young's modulus and Poisson coefficient of 69 GPa and 0.3, respectively.

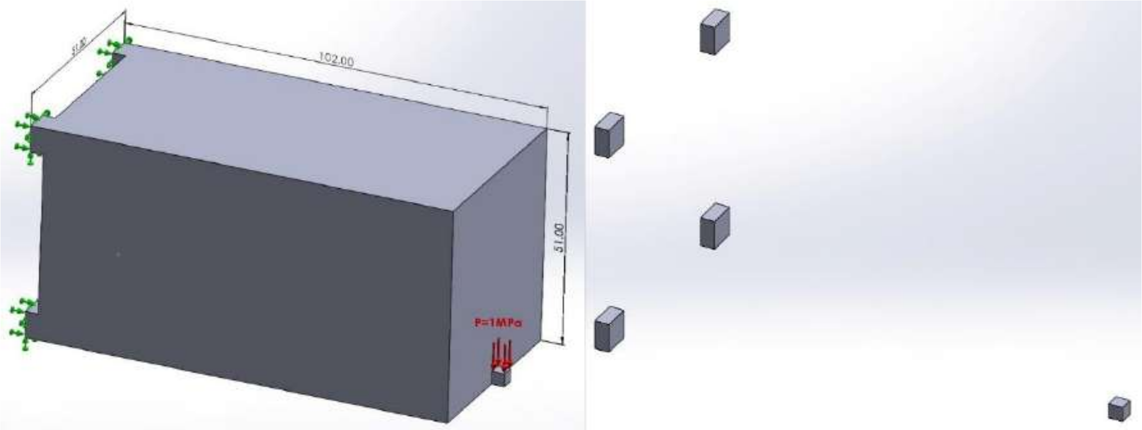


Figure 4.37- The CAD model of cantilever with four fixing constraints. Right) non-design, Left) whole model under BC.

Among the three effective optimization parameters, d_g and f are kept unchanged at 1.6mm and 0.03, while ρ_{th} value is turned to 0.3. The density distribution and the optimized result with a density of more than 0.3 are reached after 34 iterations and displayed in Figure 4.38.

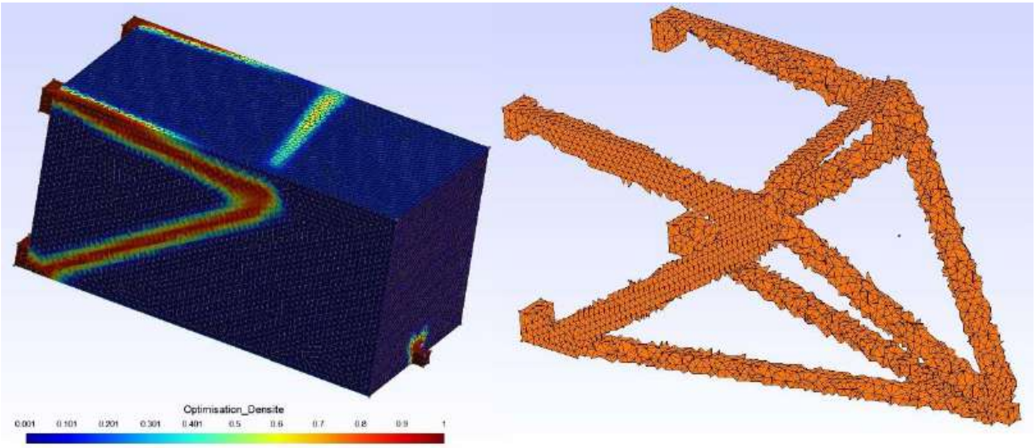


Figure 4.38- The TO result of cantilever with four fixing constraints. Left) density distribution. Right) elements with densities equal or more than 0.3.

The smoothed TO result with a density threshold of 0.3 after 10 iterations of Taubin and five iterations of Jiao method is brought in Figure 4.39.

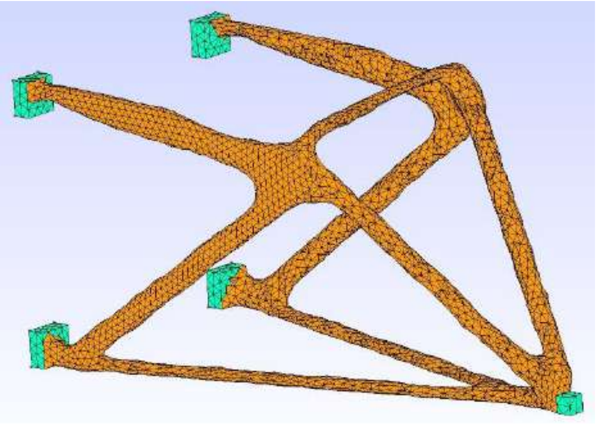


Figure 4.39- The smoothed TO result of cantilever with four fixing constraints.

4.5.4.1. Numerical and visual validations with three methods

Like the other case studies, the smoothed TO result is the input of the developed method, methods 1 and 2. The resultant skeletons from the three methods are assessed in terms of centeredness, and Table 4.7 records this assessment. It should be mentioned that the skeletons resulting from all methods are evaluated by comparing them to the smoothed

TO result, 10 iterations of Taubin and five iterations of Jiao. The lowest value of the centeredness accords with the best-centered result, and it belongs to the skeleton of the developed method, as stated in this table. Besides, method 1 hired a mass of skeletal points to form a well-centered skeleton. Therefore, no skeletal point is out of the object in any of these skeletons.

Table 4.7- The numerical centeredness result for the smoothed TO cantilever with four fixing constraints- the inputs of all methods are smoothed with 10 iterations of Taubin followed by five iterations of Jiao.

	Our method	Method 1	Method 2
E_{center}	0.0074	0.0089	0.0113
Compactness	90	1316	199
Out of object	0	0	1

The merged version of these skeletons with the smoothed TO result is established in Figure 4.40 to facilitate comparison. The large distance between the junction points to the closed point of the branches to the same junction happens for the developed method in this case study, like Monster lite, since the junction box is big for them. The skeleton of method 1 released two junction points instead of one for the two upper junctions with big junction areas since it contracts the branches to obtain the skeleton. It can be extended to the skeleton of method 2. However, method 2 could not resolve the occurred loop in one of these junctions while collapsing the triangles as a step in this method referred to in section 4.3.2. This magnified loop in the figure connects the two junctions in this junction area. Moreover, for the case of the junction close to the non-design part in the lowest rightest part of the object, the second junction generation stems from the technique of connecting the closed point of the skeleton to the non-design skeletal point. This technique is in the post-process of methods 1 and 2. So, these methods did not generate this junction naturally.

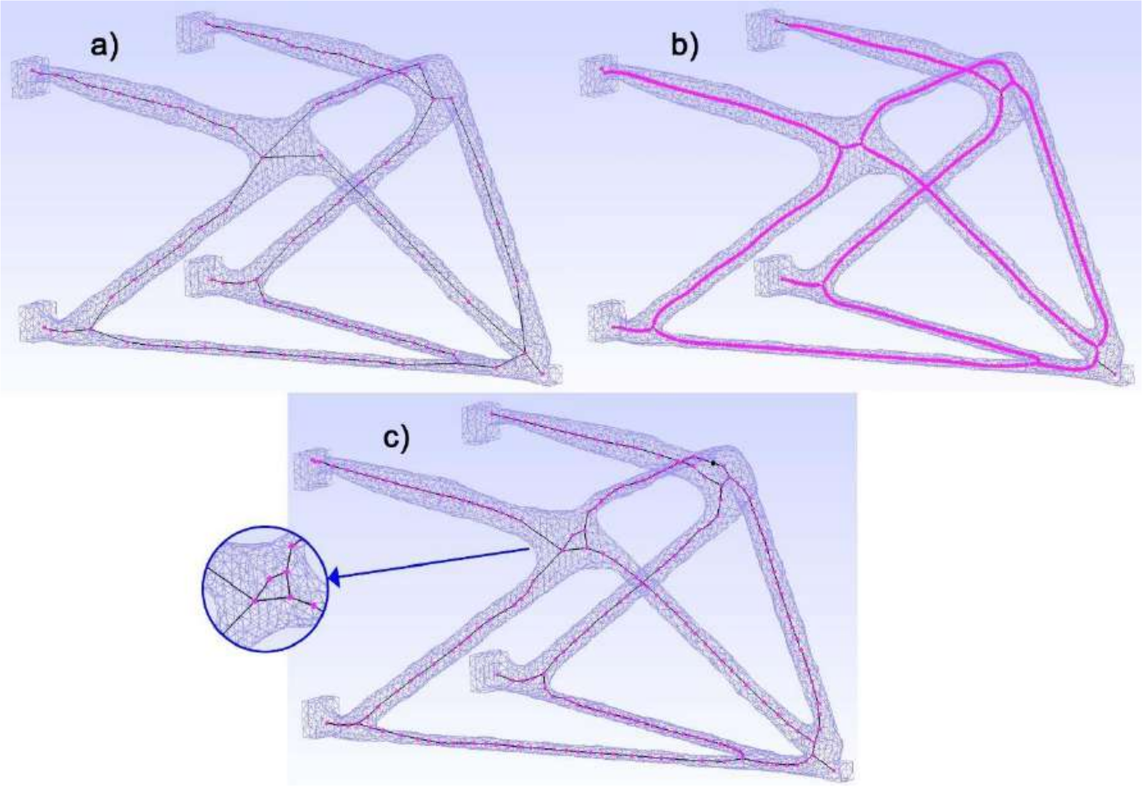


Figure 4.40- The generated skeletons by three methods for cantilever with four fixing constraints. a) the developed method, b) method 1, c) method 2.

4.6. Conclusion

In summary, MAGiC as the principal platform of this project is introduced, then the process of obtaining a proper skeleton from the selected skeletonization methods to compare with the developed method is overviewed. This process includes a brief look at the method, its development tool, its effective parameters, how they are tuned, and the needed post-process to prepare a proper skeleton for the comparison manner. Afterward, the chosen criterion for validation, how it is implemented, and how it operates are explained. Meanwhile, the vague state of the criterion regarding the points out of the object is dissected to add this state to the foreseen occurring conditions for generality. Finally, this chapter presented four examples: cantilever, y form, monster lite, and cantilever with four fixing constraints, which the developed method, as well as the other two methods, are utilized to skeletonize them. Having the validation goal in mind, the generated skeletons by these three methods are evaluated in terms of the centeredness, while they are also compared visually. Therefore, the presented examples proved that the developed method generates well-centered skeletons. Moreover, the skeletons generated by method 1 are better than method 2 in terms of the centeredness. Simultaneously, the effect of contraction and mesh size for the cantilever case study is investigated. It is concluded that the contracted TO result does not entail a better-centered skeleton, especially when the contraction is not toward the object’s center. Regarding the mesh size effect, the resultant skeleton varies since the TO result changes. The developed method can be applied to beam-like TO results specifically for the applications centeredness matters.

CHAPTER 5 – CONCLUSIONS AND PERSPECTIVES

5.1. Conclusions

In this thesis, an automatic skeletonization method is developed to convert the TO results tending to beam structures to skeletons. The method aims to align the skeleton with the neutral axis of the beam structure. This work is part of the process for converting TO results into a CAD model. Since the optimization affects only the design domain and the non-design part stays unchanged, this method focuses only on the design domain.

The second chapter presents a literature review divided into three main sections. The topology optimization methods are classified in the first section, where some popular classes, along with several known methods in each, are reviewed. To highlight the importance of manufacturing considerations, TO methods integrating manufacturing constraints were chosen to close this section. The second section focuses on the methods of interpreting TO results, where some developed methods are gathered in different classes. Due to the emphasis on beam-like structures in this work, some interpretation methods specific to these structures are surveyed. The third section analyzes curve skeletonization methods and approaches for their comparison. It is revealed that quantitative methods for comparing skeletons are limited in number.

An illustrative example is employed in the third chapter to demonstrate the approach of the proposed method after providing brief instructions on obtaining a beam-like structure in the SIMP TO result. This method is designed to identify the object's mesh elements section by section, following the branch flow by means of an identifier box. A skeletal point is generated at the center of the identified mesh elements in each section. Then, the identifier box goes further in the branch flow. It continues to the branch's end or junction point, and this process is repeated for all branches. Moreover, all junctions are investigated to ensure that there is no branch to process, since junctions are the start and end of a branch if this branch is not connected to the non-design domain.

Chapter four is devoted to evaluating the developed method through four examples. Since the neutral axis of a beam lies at its centroid, centeredness is used as the evaluation measure. Two contraction-based skeletonization methods are selected for validation, as these methods are recognized for producing well-centered skeletons. A criterion is chosen to quantitatively measure the off-centre beside the qualitative visual evaluation, which is routine in curve skeletonization. The results highlighted the strong performance and well-centeredness of the skeletons generated by the developed method.

Perspectives for future work arising from this research are enumerated in the next section.

5.2. Perspectives

5.2.1. Limitations and application domain

While the developed method demonstrates strong performance for well-defined beam-like structures, several limitations define its current application domain and may lead to reduced accuracy or failure in specific scenarios.

5.2.1.1. Geometric complexity challenges

The method faces difficulties when dealing with thin branches at junctions, particularly when these branches connect diagonally. As demonstrated in section 4.5.4 and Figure 4.40 for the cantilever with four fixing constraints, the upper branch connecting two symmetric parts is both thinner than adjacent branches and positioned diagonally at both junctions. This combination represents a worst-case scenario in which the identifier box struggles to capture enough points for accurate skeleton generation. More generally, structures with drastic directional changes or sharp curvature may challenge the section-by-section progression approach, as the marching direction may not adapt optimally to abrupt geometric transitions.

5.2.1.2. Global frame dependency

As outlined in section 3.4.2.1, the marching direction depends on the global reference frame, preventing local adjustment to the branch direction. This limitation requires careful alignment of CAD models with the global frame, restricting the method's flexibility for arbitrarily oriented structures.

5.2.1.3. Contraction method dependency

The method's centeredness performance is directly influenced by the quality of the contraction applied before skeletonization, as investigated in section 4.5.1.2. If the contraction method fails to maintain centering, particularly for thin or curved branches, the subsequent skeletonization inherits these deficiencies and can generate points which lie outside the object. For example, applying contraction to a thin branch mentioned above would likely transform it into a curve which diagonally connects to junctions, further complicating accurate skeleton generation.

While these limitations define the current application domain of the method, several refinements and enhancements have been identified to address these challenges and extend the method's robustness. These improvements are detailed in the following sections.

5.2.2. Enhancing Algorithm Independence through Global Frame Alignment

The marching direction depends on the global reference frame, and the algorithm cannot locally adjust to the perpendicular direction of the interface section, as outlined in section 3.4.2.1. In this work, CAD models are created to align the global frame with this perpendicular direction. This workaround enables the method to function correctly but requires user intervention and does not eliminate the fundamental dependency on frame alignment.

Achieving independence from global frame alignment is the most critical aspect of the proposed method that requires improvement. The current approach of aligning initial CAD models with the interface perpendicular direction is inefficient and limits automation. The required improvements include: (1) automatic detection of local reference frames based on interface geometry right after the non-design part; (2) Subsequently, adaptive marching strategies that adjust direction based on local branch orientation instead of a fixed global frame; and (3) robust handling of complex geometries through coordinate transformation techniques or domain decomposition when necessary.

5.2.3. CAD model reconstruction of the TO result

The proposed method supports CAD model reconstruction of TO beam-like structures by providing geometric and structural data for manufacturing via conventional techniques, as shown in Figure 5.1. To design a beam structure for manufacturing, the geometry of each beam's neutral axis and section is sufficient. In the proposed method, the resultant skeleton is considered as a guide to extract neutral axes of beams in the beam structure. Each beam's neutral axis is a branch limited by two junctions. The normalization step is a tuning process to convert a curvilinear branch to a linear one to generate a straight beam's neutral axis for manufacturing, as shown in Figure 5.2 for the cantilever case study.

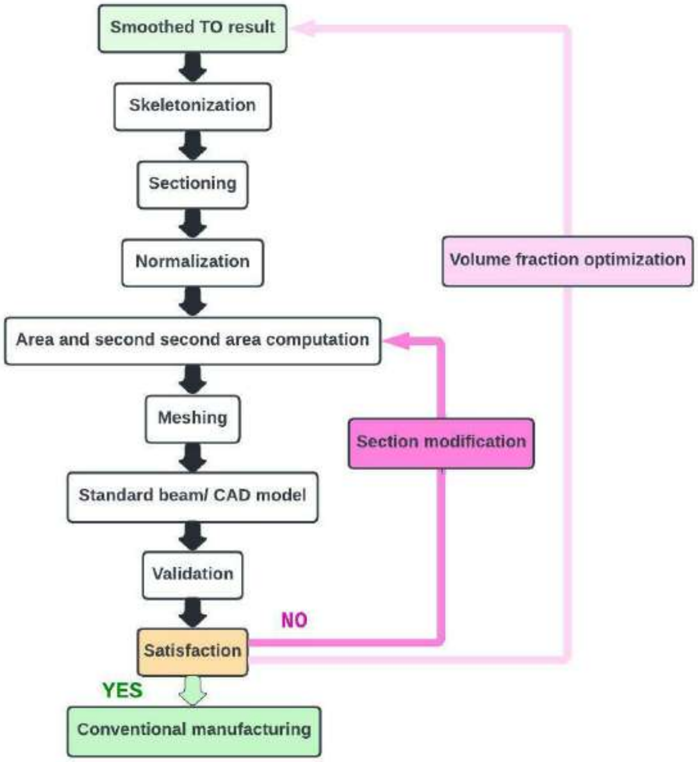


Figure 5.1- The proposed flowchart to reconstruct a CAD model from beam-like TO result which is manufacturable by conventional techniques.

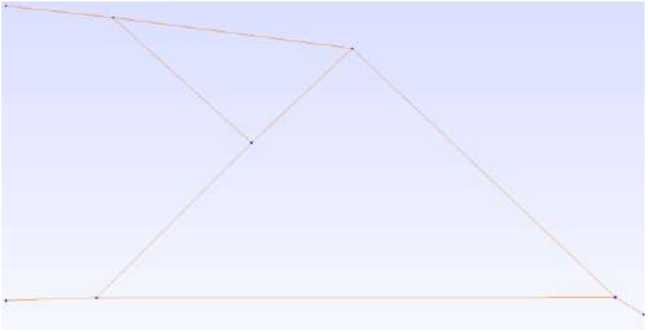


Figure 5.2- Normalization of branches in the resultant skeleton.

Regarding each beam's section, in the proposed method, at each skeletal point, the section is generated during the centeredness evaluation process. Moreover, other measures are required to obtain beam sections that match the TO result's mechanical properties due to reconstruction objective. Thus, the area and second moment of area of the section can be computed and saved on each skeletal point. Therefore, all required data for reconstructing CAD model of a TO result beam structure, including the neutral axes of beams (normalized skeleton) and beams' sections (area and second moment of area of section at each skeletal point) are computed.

Regarding the meshing step, three types of mesh elements are generated to ensure accurate finite element analysis: the tetrahedral elements to mesh non-design domain, the beam elements mesh for the skeleton, and the mini-beam elements for connecting the skeleton to the non-design. To solve the incompatibility of the degree of freedom in the connection of solid to beam elements (1D-3D), Bournival et al. [74] recommend benefiting from infinitely rigid mini beams. In the beam selection step, a standard beam is chosen for each branch to match the mean of the recorded area and second moment of area of its skeletal points, ensuring structural consistency. The validation step is inevitable if SIMP is the TO method since it ignores mechanical constraints during optimization. Thus, the analyzed results of the smoothed TO result and the reconstructed model should be compared in terms of Von Mises stress, displacement, strain energy, and compliance. If the reconstructed model's performance metrics (e.g., Von Mises stress, displacement) closely match the TO result, the model is considered satisfactory. However, if the stress analysis result of the reconstructed model is below the admissible stress the model is inadequate for the specified loading condition. To overcome this issue, two options exist to gain the desired result automatically. First, each beam should be modified (magnified or minified) separately, regarding the stress result of the validation. Second, the value of the volume fraction should be optimized due to the previous comparison result to lead to a satisfactory result.

5.2.4. Compatibility of the developed method with different TO methods

To make the skeletonization method work with more cases, we need to check how well it handles different kinds of TO output data. The results can come as continuous density fields (case of this study), binary meshes, or level set functions. This section quickly looks into how these data forms affect the skeletonization performance.

Density-based representation: regarding the penalization scheme used during optimization, RAMP provides the sharpest material-void transition among these penalization schemes, and its clear, robust result enhances the accuracy of the resultant skeleton. SINH also offers a sharp density transition, while the topology is less robust than RAMP's. SIMP produces softer transitions that may require more post-processing.

Mesh-based representations: the results of the evolutionary-based methods are compatible with this skeletonization method, with SERA performing best. Since its binary material-void output mesh has high connectivity, it needs almost no post-processing, is ideal for direct skeletonization input, and leads to high-quality skeletons. Thanks to its bidirectional optimization, its result is more robust than others. BESO also provides high connectivity, but the result is less refined than SERA due to simpler sensitivity updates. AESO and ESO need more smoothing and post-processing because of the checkerboard effect coming from the fragmented and disconnected topologies in binary mesh representations.

Level set representations: these can be compatible with an additional conversion step to a binary density field. The inherently smooth boundaries in level set methods may enhance skeleton quality.

These comparisons show that data representation significantly influences the performance in skeletonization, often more than the choice of TO methods. In this study, the density field is chosen.

5.2.5. Skeletal point position improvement to solve being out of the object

As post-processing, the skeletal points' positions can be improved, especially in the case of the skeletal point out of the object. It is suggested that the skeletal points be moved to the center of the section during the centeredness evaluation under certain conditions. These conditions are firstly when the point is out of the object and then when it is highly off-centre. It may require successive modifications to keep the skeleton smooth.

Regarding the highly off-centre, a scale should be defined to evaluate the off-centring, like the length of the longest ray in the section.

5.2.6. Refining the junction position

The junction position is critical for several reasons. First, its area remains volumetric even after contraction, and it is more likely to ruin the centeredness, as illustrated in section 4.5.1.2 and Figure 4.24. Moreover, the segments connecting the junction points to the skeletal points are risky in traversing the object, as observed in Figure 4.23a, b, and f. Furthermore, refining the junction position is essential in some interpretation methods, such as the one proposed in section 5.2.1, since the normalization step relies directly on precise junction points, and the more accurately the junction is positioned, the more effectively the beam aligns with the object.

5.2.7. Centeredness improvement by adding a well-centered contraction method

The TO result tends to the skeleton form by contracting it, while it matters to keep it in the center during this process. The centeredness may be amplified by employing the contracted TO result instead of the smoothed one. Concurrently, it should be well-centred compared to the smoothed TO result as the measure of centeredness evaluation. It has already been tried in this work in section 4.5.1.2, but it is not straightforward to keep the contraction result in the center. The combination of contraction-attraction may work like what is used in method 2. However, it is demonstrated that the skeleton stemming from the smoothed TO result is well-centred.

REFERENCES

1. Gebisa, A.W. and H.G. Lemu. *A case study on topology optimized design for additive manufacturing*. in *IOP conference series: materials science and engineering*. 2017. IOP Publishing.
2. Li, L., W. Wang, and Y. Chu, *A simple and stable centeredness measure for 3D curve skeleton extraction.* IEEE Transactions on Visualization and Computer Graphics, 2020. **28**(3): p. 1486-1499.
3. Cuilliere, J.-C., *Introduction a la methode des elements finis - 2e ed Cours et exercices corriges*. 2023, Dunod.
4. Bendsoe, M.P. and O. Sigmund, *Topology optimization: theory, methods, and applications*. 2013: Springer Science & Business Media.
5. Hassani, B. and E. Hinton, *A review of homogenization and topology optimization III—topology optimization using optimality criteria.* Computers & structures, 1998. **69**(6): p. 739-756.
6. Sigmund, O. and K. Maute, *Topology optimization approaches: A comparative review.* Structural and multidisciplinary optimization, 2013. **48**(6): p. 1031-1055.
7. Deaton, J.D. and R.V. Grandhi, *A survey of structural and multidisciplinary continuum topology optimization: post 2000.* Structural and Multidisciplinary Optimization, 2014. **49**: p. 1-38.
8. Ribeiro, T.P., L.F. Bernardo, and J.M. Andrade, *Topology optimisation in structural steel design for additive manufacturing.* Applied Sciences, 2021. **11**(5): p. 2112.
9. Bruns, T., *A reevaluation of the SIMP method with filtering and an alternative formulation for solid–void topology optimization.* Structural and Multidisciplinary Optimization, 2005. **30**: p. 428-436.
10. Bendsøe, M.P. and N. Kikuchi, *Generating optimal topologies in structural design using a homogenization method.* Computer methods in applied mechanics and engineering, 1988. **71**(2): p. 197-224.
11. Suzuki, K. and N. Kikuchi, *A homogenization method for shape and topology optimization.* Computer methods in applied mechanics and engineering, 1991. **93**(3): p. 291-318.
12. Bremicker, M., et al., *Integrated Topology and Shape Optimization in Structural Design∗.* Mechanics of Structures and Machines, 1991. **19**(4): p. 551-587.
13. Hoang, V.-N., et al., *Design of lattice structures with direct multiscale topology optimization.* Composite Structures, 2020. **252**: p. 112718.
14. Do, D.T. and J. Lee, *An automatically connected graph representation based on B-splines for structural topology optimization.* Structural and Multidisciplinary Optimization, 2019. **59**: p. 2023-2040.
15. Bendsøe, M.P., *Optimal shape design as a material distribution problem.* Structural optimization, 1989. **1**: p. 193-202.

16. Zhou, M. and G.I. Rozvany, *The COC algorithm, Part II: Topological, geometrical and generalized shape optimization.* Computer methods in applied mechanics and engineering, 1991. **89**(1-3): p. 309-336.
17. Rozvany, G.I., M. Zhou, and T. Birker, *Generalized shape optimization without homogenization.* Structural optimization, 1992. **4**: p. 250-252.
18. Sigmund, O. and J. Petersson, *Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima.* Structural optimization, 1998. **16**: p. 68-75.
19. Haber, R.B., C.S. Jog, and M.P. Bendsøe, *A new approach to variable-topology shape design using a constraint on perimeter.* Structural optimization, 1996. **11**: p. 1-12.
20. Sigmund, O., *On the design of compliant mechanisms using topology optimization.* Journal of Structural Mechanics, 1997. **25**(4): p. 493-524.
21. Petersson, J. and O. Sigmund, *Slope constrained topology optimization.* International Journal for Numerical Methods in Engineering, 1998. **41**(8): p. 1417-1434.
22. Bourdin, B., *Filters in topology optimization.* International journal for numerical methods in engineering, 2001. **50**(9): p. 2143-2158.
23. Guest, J.K., J.H. Prévost, and T. Belytschko, *Achieving minimum length scale in topology optimization using nodal design variables and projection functions.* International journal for numerical methods in engineering, 2004. **61**(2): p. 238-254.
24. Kang, Z. and Y. Wang, *Structural topology optimization based on non-local Shepard interpolation of density field.* Computer methods in applied mechanics and engineering, 2011. **200**(49-52): p. 3515-3525.
25. Kang, Z. and Y. Wang, *A nodal variable method of structural topology optimization based on Shepard interpolant.* International journal for numerical methods in engineering, 2012. **90**(3): p. 329-342.
26. Liu, J. and Y. Ma, *A survey of manufacturing oriented topology optimization methods.* Advances in Engineering Software, 2016. **100**: p. 161-175.
27. Lin, C.-Y. and L.-S. Chao, *Automated image interpretation for integrated topology and shape optimization.* Structural and Multidisciplinary Optimization, 2000. **20**: p. 125-137.
28. Stolpe, M. and K. Svanberg, *An alternative interpolation scheme for minimum compliance topology optimization.* Structural and Multidisciplinary Optimization, 2001. **22**: p. 116-124.
29. Xie, Y.M. and G.P. Steven, *A simple evolutionary procedure for structural optimization.* Computers & structures, 1993. **49**(5): p. 885-896.
30. Xie, Y.M., et al., *Basic evolutionary structural optimization*. 1997: Springer.
31. Hardjasaputra, H., *Evolutionary structural optimization as tool in finding strut-and-tie-models for designing reinforced concrete deep beam.* Procedia Engineering, 2015. **125**: p. 995-1000.

32. Querin, O., G. Steven, and Y. Xie, *Evolutionary structural optimisation using an additive algorithm.* Finite elements in Analysis and Design, 2000. **34**(3-4): p. 291-308.
33. Yang, X.Y., et al., *Bidirectional evolutionary method for stiffness optimization.* AIAA journal, 1999. **37**(11): p. 1483-1488.
34. Rozvany, G.I. and O.M. Querin, *Combining ESO with rigorous optimality criteria.* International journal of vehicle design, 2002. **28**(4): p. 294-299.
35. Xia, Q., T. Shi, and L. Xia, *Stable hole nucleation in level set based topology optimization by using the material removal scheme of BESO.* Computer Methods in Applied Mechanics and Engineering, 2019. **343**: p. 438-452.
36. Ansola, R., et al., *A simple evolutionary topology optimization procedure for compliant mechanism design.* Finite Elements in Analysis and Design, 2007. **44**(1-2): p. 53-62.
37. Ansola, R., et al., *3D compliant mechanisms synthesis by a finite element addition procedure.* Finite Elements in Analysis and Design, 2010. **46**(9): p. 760-769.
38. Osher, S. and J.A. Sethian, *Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations.* Journal of computational physics, 1988. **79**(1): p. 12-49.
39. Sethian, J.A. and A. Wiegmann, *Structural boundary design via level set and immersed interface methods.* Journal of computational physics, 2000. **163**(2): p. 489-528.
40. Luo, Z., et al., *Structural shape and topology optimization using a meshless Galerkin level set method.* International Journal for Numerical Methods in Engineering, 2012. **90**(3): p. 369-389.
41. Bourdin, B. and A. Chambolle, *Design-dependent loads in topology optimization.* ESAIM: Control, Optimisation and Calculus of Variations, 2003. **9**: p. 19-48.
42. Takezawa, A., S. Nishiwaki, and M. Kitamura, *Shape and topology optimization based on the phase field method and sensitivity analysis.* Journal of Computational Physics, 2010. **229**(7): p. 2697-2718.
43. Blank, L., et al., *Phase-field approaches to structural topology optimization.* Constrained optimization and optimal control for partial differential equations, 2012: p. 245-256.
44. Sigmund, O., *Morphology-based black and white filters for topology optimization.* Structural and Multidisciplinary Optimization, 2007. **33**: p. 401-424.
45. Gao, J., B. Song, and Z. Mao, *A novel approach for length scale control in structural topology optimization.* Engineering Optimization, 2019.
46. Liu, J., *Piecewise length scale control for topology optimization with an irregular design domain.* Computer Methods in Applied Mechanics and Engineering, 2019. **351**: p. 744-765.
47. Costa, G., M. Montemurro, and J. Pailhès, *Minimum length scale control in a NURBS-based SIMP method.* Computer Methods in Applied Mechanics and Engineering, 2019. **354**: p. 963-989.

48. Chen, J., M. Freytag, and V. Shapiro, *Shape sensitivity of constructively represented geometric models.* Computer Aided Geometric Design, 2008. **25**(7): p. 470-488.
49. Zou, J., Y. Zhang, and Z. Feng, *Topology optimization for additive manufacturing with self-supporting constraint.* Structural and Multidisciplinary Optimization, 2021. **63**(5): p. 2341-2353.
50. Deng, H., P.S. Vulimiri, and A.C. To, *Cad-integrated topology optimization method with dynamic extrusion feature evolution for multi-axis machining.* Computer Methods in Applied Mechanics and Engineering, 2022. **390**: p. 114456.
51. Pan, W., et al., *An automatic approach for generating parametric models from topology-optimization results for three-axis CNC machining.* Computer-Aided Design, 2025. **182**: p. 103863.
52. Cuillière, J.-C., V. Francois, and J.-M. Drouet, *Automatic mesh generation and transformation for topology optimization methods.* Computer-Aided Design, 2013. **45**(12): p. 1489-1506.
53. Doutre, P.-T., et al. *Comparison of some approaches to define a CAD model from topological optimization in design for additive manufacturing*. in *Advances on Mechanics, Design Engineering and Manufacturing: Proceedings of the International Joint Conference on Mechanics, Design Engineering & Advanced Manufacturing (JCM 2016), 14-16 September, 2016, Catania, Italy*. 2017. Springer.
54. Stroud, I. and H. Nagy, *Solid modelling and CAD systems: how to survive a CAD system*. 2011: Springer Science & Business Media.
55. Zou, Q., H.-Y. Feng, and S. Gao, *Variational direct modeling: A framework towards integration of parametric modeling and direct modeling in CAD.* Computer-Aided Design, 2023. **157**: p. 103465.
56. Marsan, A. and D. Dutta, *Construction of a surface model and layered manufacturing data from 3D homogenization output.* 1996.
57. Chirehdast, M., *An integrated optimization environment for structural configuration design*. 1992, University of Michigan.
58. Marsan, A.L. and D. Dutta, *Computational techniques for automatically tiling and skinning branched objects.* Computers & Graphics, 1999. **23**(1): p. 111-126.
59. Tang, P.-S. and K.-H. Chang, *Integration of topology and shape optimization for design of structural components.* Structural and Multidisciplinary Optimization, 2001. **22**: p. 65-82.
60. Hsu, M.-H. and Y.-L. Hsu, *Interpreting three-dimensional structural topology optimization results.* Computers & structures, 2005. **83**(4-5): p. 327-337.
61. Bender, D. and A. Barari. *Direct solid element slicing in topology optimization for additive manufacturing*. in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. 2019. American Society of Mechanical Engineers.
62. Bremicker, M., et al., *Integrated topology and shape optimization in structural design∗.* Journal of Structural Mechanics, 1991. **19**(4): p. 551-587.

63. Yildiz, A., et al., *Integrated optimal topology design and shape optimization using neural networks.* Structural and Multidisciplinary Optimization, 2003. **25**: p. 251-260.

64. Chou, Y.-H. and C.-Y. Lin, *Improved image interpreting and modeling technique for automated structural optimization system.* Structural and Multidisciplinary Optimization, 2010. **40**: p. 215-226.

65. Yi, G., B.D. Youn, and N.H. Kim. *Geometric feature identification from topology optimization results*. in *11th World Congress on Structural and Multidisciplinary Optimisation*. 2015.

66. Yi, G. and N.H. Kim, *Identifying boundaries of topology optimization results using basic parametric features.* Structural and Multidisciplinary Optimization, 2017. **55**: p. 1641-1654.

67. Koguchi, A. and N. Kikuchi, *A surface reconstruction algorithm for topology optimization.* Engineering with Computers, 2006. **22**: p. 1-10.

68. Denk, M., K. Rother, and K. Paetzold, *Beam-colored Sketch and Image-based 3D Continuous Wireframe Reconstruction with different Materials and Cross-Sections.* 2021.

69. Ramsaier, M., et al. *On a Physics-based Reconstruction Algorithm for Generating Clean Parametric Native CAD-Models from Density-based Topology Optimization Results*. in *Proceedings of the 13th World Congress on Structural and Multidisciplinary Optimisation*. 2019.

70. Subedi, S.C., C.S. Verma, and K. Suresh, *A review of methods for the geometric post-processing of topology optimized models.* Journal of Computing and Information Science in Engineering, 2020. **20**(6): p. 060801.

71. Nana, A., J.-C. Cuillière, and V. Francois, *Automatic reconstruction of beam structures from 3D topology optimization results.* Computers & Structures, 2017. **189**: p. 62-82.

72. Cao, J., et al. *Point cloud skeletons via laplacian based contraction*. in *2010 Shape Modeling International Conference*. 2010. IEEE.

73. Au, O.K.-C., et al., *Skeleton extraction by mesh contraction.* ACM transactions on graphics (TOG), 2008. **27**(3): p. 1-10.

74. Bournival, S., J.-C. Cuillière, and V. François, *A mesh-geometry based method for coupling 1D and 3D elements.* Advances in Engineering Software, 2010. **41**(6): p. 838-858.

75. Cuillière, J.-C., V. François, and A. Nana, *Automatic construction of structural CAD models from 3D topology optimization.* Computer-Aided Design and Applications, 2018. **15**(1): p. 107-121.

76. Amroune, A., J.-C. Cuillière, and V. François, *Automated lofting-based reconstruction of CAD models from 3D topology optimization results.* Computer-Aided Design, 2022. **145**: p. 103183.

77. Kresslein, J., et al., *Automated cross-sectional shape recovery of 3D branching structures from point cloud.* Journal of Computational Design and Engineering, 2018. **5**(3): p. 368-378.

78. Yin, G., X. Xiao, and F. Cirak, *Topologically robust CAD model generation for structural optimisation.* Computer methods in applied mechanics and engineering, 2020. **369**: p. 113102.
79. Taubin, G. *A signal processing approach to fair surface design*. in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. 1995.
80. Tagliasacchi, A., et al. *Mean curvature skeletons*. in *Computer Graphics Forum*. 2012. Wiley Online Library.
81. Denk, M., K. Rother, and K. Paetzold, *Truss Parametrization of Topology Optimization Results with Curve Skeletons and Meta Balls.* Proceedings of the Design Society, 2022. **2**: p. 363-372.
82. Morris, N.J., P.K. Jayaraman, and A. Butscher, *BeNTO: Beam Network Topology Optimization.* Computer-Aided Design, 2023. **156**: p. 103439.
83. Polak, J. and M. Nowak, *From structural optimization results to parametric CAD modeling—automated, skeletonization-based truss recognition.* Applied Sciences, 2023. **13**(9): p. 5670.
84. Tagliasacchi, A., et al. *3d skeletons: A state-of-the-art report*. in *Computer Graphics Forum*. 2016. Wiley Online Library.
85. Tierny, J., J.-P. Vandeborre, and M. Daoudi. *3D mesh skeleton extraction using topological and geometrical analyses*. in *14th Pacific Conference on Computer Graphics and Applications (Pacific Graphics 2006)*. 2006.
86. Cornea, N.D., D. Silver, and P. Min, *Curve-skeleton properties, applications, and algorithms.* IEEE Transactions on visualization and computer graphics, 2024. **13**(3): p. 530-548.
87. Chuang, J.-H., C.-H. Tsai, and M.-C. Ko, *Skeletonisation of three-dimensional object using generalized potential field.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000. **22**(11): p. 1241-1251.
88. Farag, S., et al. *Extracting 3d mesh skeletons using antipodal points locations*. in *2013 UKSim 15th International Conference on Computer Modelling and Simulation*. 2013. IEEE.
89. Sobiecki, A., et al. *Qualitative comparison of contraction-based curve skeletonization methods*. in *Mathematical Morphology and Its Applications to Signal and Image Processing: 11th International Symposium, ISMM 2013, Uppsala, Sweden, May 27-29, 2013. Proceedings 11*. 2013. Springer.
90. Telea, A.C. and A.C. Jalba. *Computing curve skeletons from medial surfaces of 3D shapes*. in *Theory and Practice of Computer Graphics*. 2012. Eurographics UK.
91. Tagliasacchi, A., H. Zhang, and D. Cohen-Or, *Curve skeleton extraction from incomplete point cloud*, in *ACM SIGGRAPH 2009 papers*. 2009. p. 1-9.
92. Hui, Z., et al. *A Robust Skeleton Lines Extraction Method for Individual Tree Modeling Using Terrestrial LiDAR Point Clouds*. in *2024 IEEE International Conference on Image Processing Challenges and Workshops (ICIPCW)*. 2024. IEEE.

93. Zhou, J., J. Liu, and M. Zhang, *Curve skeleton extraction via k-nearest-neighbors based contraction.* International Journal of Applied Mathematics and Computer Science, 2020. **30**(1).
94. Chen, H., et al., *Multiscale feature line extraction from raw point clouds based on local surface variation and anisotropic contraction.* IEEE Transactions on Automation Science and Engineering, 2021. **19**(2): p. 1003-1016.
95. Jiang, A., et al., *Skeleton extraction from point clouds of trees with complex branches via graph contraction.* The Visual Computer, 2021. **37**(8): p. 2235-2251.
96. Sobiecki, A., A. Jalba, and A. Telea, *Comparison of curve and surface skeletonization methods for voxel shapes.* Pattern Recognition Letters, 2014. **47**: p. 147-156.
97. Jalba, A.C., A. Sobiecki, and A.C. Telea, *An unified multiscale framework for planar, surface, and curve skeletonization.* IEEE transactions on pattern analysis and machine intelligence, 2015. **38**(1): p. 30-45.
98. Hu, H., et al. *Curve skeleton extraction from 3D point clouds through hybrid feature point shifting and clustering*. in *Computer graphics forum*. 2020. Wiley Online Library.
99. Li, L. and W. Wang. *Improved use of LOP for curve skeleton extraction*. in *Computer graphics forum*. 2018. Wiley Online Library.
100. Li, L. and W. Wang. *Contracting medial surfaces isotropically for fast extraction of centred curve skeletons*. in *Computer Graphics Forum*. 2017. Wiley Online Library.
101. Yan, Y., et al., *Erosion thickness on medial axes of 3D shapes.* ACM Transactions on Graphics (TOG), 2016. **35**(4): p. 1-12.
102. Cuillière, J.-C. and V. Francois, *Integration of CAD, FEA and topology optimization through a unified topological model.* Computer-aided design and applications, 2014. **11**(5): p. 493-508.
103. Cuillière, J.-C., V. Francois, and J.-M. Drouet, *Towards the integration of topology optimization into the cad process.* Computer-Aided Design and Applications, 2014. **11**(2): p. 120-140.
104. Taubin, G. *Curve and surface smoothing without shrinkage*. in *Proceedings of IEEE international conference on computer vision*. 1995. IEEE.
105. Chen, C.-Y. and K.-Y. Cheng, *A sharpness dependent filter for mesh smoothing.* Computer Aided Geometric Design, 2005. **22**(5): p. 376-391.
106. Chen, C.-Y. and K.-Y. Cheng, *A direction-oriented sharpness dependent filter for 3D polygon meshes.* Computers & Graphics, 2008. **32**(2): p. 129-140.
107. Clark, B., N. Ray, and X. Jiao. *Surface mesh optimization, adaption, and untangling with high-order accuracy*. in *Proceedings of the 21st international meshing roundtable*. 2013. Springer.
108. Delannoy, C., *C++ pour les programmeurs C*. 2007: Editions Eyrolles.
109. OpenCASCADE. *[Online]: Capgemini*. 2017 2017]; Available from: https://www.opencascade.com/.
110. Aster., C. *[Online]: Électricité de France.*; Available from: https://www.code-aster.org/.

111. Geuzaine, C. and J.F. Remacle, *Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities.* International journal for numerical methods in engineering, 2009. **79**(11): p. 1309-1331.
112. Starlab. *[Online].* Available from: https://github.com/taiya/starlab-mcfskel.
113. Amroune, A., *Automatisation de la construction de modèles géométriques à partir de résultats d'optimisation topologique*. 2022.
114. Cloudcontr. *[Online].* Available from: https://github.com/taiya/cloudcontr
115. Nana Takougoum, P.A., *Adaptation et transformation automatiques des résultats d'optimisation topologique en modèles CAO de structures de poutres*. 2018.
116. lite., M. *[Online].* Available from: https://www.roguecanada.ca/rogue-rml-3w-fold-back-wall-mount-rack.