



**HAL**  
open science

**Design and Optimisation Methods for Structures  
produced by means of Additive Layer Manufacturing  
processes**  
Giulio Costa

► **To cite this version:**

Giulio Costa. Design and Optimisation Methods for Structures produced by means of Additive Layer Manufacturing processes. Chemical and Process Engineering. Ecole nationale supérieure d'arts et métiers - ENSAM, 2018. English. NNT : 2018ENAM0035 . tel-02069224

**HAL Id: tel-02069224**

**<https://pastel.hal.science/tel-02069224>**

Submitted on 15 Mar 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

École doctorale n° 432 : Science des Métiers de l'ingénieur

## Doctorat ParisTech

# THÈSE

pour obtenir le grade de docteur délivré par

**l'École Nationale Supérieure d'Arts et Métiers**

**Spécialité " Conception "**

*présentée et soutenue publiquement par*

**Giulio COSTA**

le 22 octobre 2018

## **Design and Optimisation Methods for Structures produced by means of Additive Layer Manufacturing processes**

Directeur de thèse : **Jérôme PAILHÈS**  
Co-encadrement de la thèse : **Marco MONTEMURRO**

### **Jury**

<b>M. Philippe VERON</b> , Professeur des Universités, LSIS, UMR CNRS 7296, ENSAM Aix-en-Provence	Président
<b>M. Grégoire ALLAIRE</b> , Professeur des Universités, CMAP, UMR CNRS 7641, Ecole Polytechnique	Rapporteur
<b>M. Alberto PIRRERA</b> , Senior Lecturer, ACCIS, University of Bristol	Rapporteur
<b>M. Paolo VANNUCCI</b> , Professeur des Universités, LMV, UMR 8100, Université de Versailles	Examineur
<b>M. Frédéric VIGNAT</b> , Maître de Conférences HDR, G-SCOP, UMR CNRS 5272, Grenoble INP	Examineur
<b>M. Marco MONTEMURRO</b> , Maître de Conférences, I2M, UMR CNRS 5295, ENSAM Bordeaux	Examineur
<b>M. Jérôme PAILHÈS</b> , Professeur des Universités, I2M, UMR CNRS 5295, ENSAM Bordeaux	Examineur
<b>M. Stéven ROBIDOU</b> , Ingénieur Bureau d'Etudes, Ariane Group, Le Haillan	Invité



*Ai miei Genitori, a mio Fratello Gabriele, a Laura,  
le quattro persone più importanti della mia vita,  
sperando di renderle orgogliose di me.*





# Acknowledgements

I would like to thank my supervisors Jérôme Pailhès and Marco Montemurro for having given me the opportunity to improve as a researcher and as an engineer through this PhD thesis project. Beyond their scientific values, Jérôme's broad vision and Marco's incredible persistence in facing research problems constitute both exemplary attitudes for a young researcher. My sincerest thanks are addressed to the I2M Laboratory, to the industrial partners and to the Nouvelle-Aquitaine region for having supported the FUTURPROD project. I am particularly grateful to Stéven Robidou, Florine Weil and Thierry Gomez-Lavocat for their time and for their interest in my work.

Never before I needed to say "thank you" to my parents as at this moment. During bad moments in these last three years I joked with them by saying that it was their fault for my loneliness in Bordeaux, because they made me feel beloved. Always. For 25 years before leaving my place in Pisa and later on. Maybe I exaggerated, but it is not so far from truth. Thank you Sandro and Lucia for all.

Thank you also to you, Laura. I swear that I have never doubted about the strength of our love. However, these last three years were not just a prove of love; they have been a prove of respect and of sincere care for each other. I really want to say thank you for having been always on my side, for having laughed with me and for having cried with me. I will keep doing my best to deserve you.

Gabriele, my little big brother, you are someone very special to me, someone to estimate for his young genuineness. Thank you for your fresh optimistic and sometimes non-realistic way to look at the world.

I would like to thank my families. The first one is constituted of my relatives: my grandmother, my uncles and aunts, my cousins. You have been the kernel of my happiness in childhood, in adolescence and in adult life. A particular thought is dedicated to my uncle Pierluigi and my aunt Simona, which have followed this experience with passion, love and admiration. I am also extremely grateful to Francesca and Giuseppe for having always encouraged me and Laura in both our studies and in our life together. You all constitute my happiest memories. The second family is made of friends, especially those "ritals" which have shared this experience with me: Lorenzo, Michele, Pietro thank you for having been my strongest support in Bordeaux. A special mention is for you, Yohann. I have found a true friend with an unbelievable intellectual honesty, easy to misunderstand with ingenuity. Thank you for your infinite kindness.

Finally, I am sincerely grateful to those people who have somehow supported me along the way (no matter the eventual distance between us): thank you Anita, Antone, Antonio, Enrico, Giacinto Alberto, Giovanni, Grace, Luca, Maria Antonietta, Mélissande, Paolo, Torquato.



# Remerciements

Je voudrais remercier mes encadrants Jérôme Pailhès et Marco Montemurro pour la possibilité qu'ils m'ont donné de m'améliorer et de m'épanouir en tant que chercheur et ingénieur à travers cette thèse de doctorat. Au-delà de leur valeur scientifique, la vision d'ensemble de Jérôme et la persévérance de Marco quand il s'agit d'aborder un problème de recherche sont des qualités exemplaires pour un jeune chercheur. Mes remerciements les plus sincères sont adressés au Laboratoire I2M, aux partenaires industriels et à la région Nouvelle-Aquitaine pour avoir soutenu le projet FUTURPROD. Je suis particulièrement reconnaissant envers Stéven Robidou, Florine Weil et Thierry Gomez-Lavocat pour leur temps et pour l'intérêt montré envers mes travaux.

Maintenant plus que jamais, je sens le besoin de dire "merci" à mes parents. Pendant les mauvaises périodes je rigolais avec eux en disant que si je me sentais seul, et bien c'était leur faut parce qu'ils m'ont vraiment aimé. Toujours, pendant 25 années avant de laisser Pisa. Peut-être que j'exagère, mais cela n'est pas trop loin de la vérité. Merci Sandro et Lucia pour tout.

Merci à toi aussi, Laura. Je jure que je n'ai jamais eu des doutes concernant notre amour. Cependant, ces dernières années n'ont pas été seulement une démonstration d'amour; ils ont été une preuve de respect et de soin sincère de l'un pour l'autre. Je veux vraiment te remercier pour avoir toujours été de mon côté, pour avoir ri avec moi et pour avoir pleuré avec moi. Je vais continuer de faire de mon mieux pour te mériter.

Gabriele, mon petit grand frère, tu es quelqu'un de très spécial pour moi, quelqu'un à estimer pour son jeune authenticité. Merci pour ta vision du monde si fraîche, optimiste et parfois même un peu non réaliste.

Je voudrais remercier mes familles. La première est formée par mes proches: ma grand-mère, mes oncles, mes tantes et mes cousins. Vous avez été le noyau de mon bonheur pendant l'enfance, l'adolescence et la vie d'adulte. Une pensée particulière est adressée à mon oncle Pierluigi et ma tante Simona, qui ont suivi mon expérience avec enthousiasme, affection et admiration. Je suis aussi très reconnaissant à Francesca et Giuseppe pour avoir toujours encouragé Laura et moi pour nos études et pour notre vie ensemble. Vous tous constituez mes souvenirs les plus chers. La seconde famille est faite par des amis, notamment ceux "ritals" qui ont partagé cette expérience avec moi: merci Lorenzo, Michele et Pietro pour avoir été mon support le plus important à Bordeaux. Il faut faire une mention spéciale pour toi, Yohann. J'ai trouvé un vrai ami, avec une honnêteté intellectuelle incroyable, trop facilement confondue avec ingénuité. Merci pour ta gentillesse infinie.

Enfin, je suis vivement reconnaissant à toutes les personnes qui m'ont supporté en chemin de quelque manière (peu importe l'éventuelle distance entre nous): merci Anita, Antone, Antonio, Enrico, Giacinto Alberto, Giovanni, Grace, Luca, Maria Antonietta, Mélissande, Paolo, Torquato.



# Ringraziamenti

Vorrei ringraziare i miei inquadranti Jérôme Pailhès e Marco Montemurro per la possibilità che mi hanno dato di migliorare come ricercatore e come ingegnere attraverso questa tesi di dottorato. Al di là del loro valore scientifico, la visione d'insieme di Jérôme e la perseveranza di Marco nell'affrontare un problema di ricerca sono delle qualità esemplari per un giovane ricercatore. Inoltre, i miei ringraziamenti più sinceri vanno al Laboratorio I2M, ai partner industriali e alla regione Nouvelle-Aquitaine per aver supportato il progetto FUTURPROD. Sono particolarmente riconoscente a Stéven Robidou, Florine Weil e Thierry Gomez-Lavocat per il loro tempo e l'interesse mostrato verso il mio lavoro.

Adesso come non mai, sento il bisogno di dire “grazie” ai miei genitori. Durante i brutti periodi scherzavo con loro dicendo che erano loro la causa del mio sentirmi solo. Questo perché mi hanno fatto sempre sentire tanto amato. Sempre, durante 25 anni prima di lasciare Pisa e dopo anche di più. Forse esagero, ma non sono troppo lontano dalla verità. Grazie babbo Sandro e mamma Lucia per tutto.

Grazie anche a te, Laura. Giuro che non ho mai avuto dubbi sulla forza del nostro amore. Tuttavia, questi ultimi anni non sono stati solamente una dimostrazione d'amore ma una prova di rispetto e di cura dell'uno per l'altro. Voglio veramente ringraziarti per essere sempre stata al mio fianco, per aver riso con me e per aver pianto con me. Continuerò a fare del mio meglio per meritarti.

Gabriele, mio grande fratellino, tu sei qualcuno davvero speciale per me, qualcuno da stimare per la sua giovane genuinità. Grazie per la tua visione del mondo così fresca, ottimista e talvolta anche un po' non realista.

Vorrei poi ringraziare le mie due famiglie. La prima è formata dai miei parenti: mia nonna, i miei zii e le mie zie, i miei cugini. Siete stati il nucleo della mia felicità durante l'infanzia, l'adolescenza e nella vita adulta. Un pensiero particolare è rivolto ai miei zii Pierluigi e Simona, che hanno seguito questa mia esperienza con trasporto, affetto e ammirazione. Sono anche estremamente riconoscente a Francesca e Giuseppe per aver sempre incoraggiato me e Laura tanto nei nostri rispettivi studi quanto nella nostra vita insieme. Voi tutti costituite i miei ricordi più cari. La seconda famiglia è fatta da amici, in particolare da quei “ritals” che hanno condiviso questo percorso con me: grazie Lorenzo, Michele e Pietro per essere stati il mio maggiore supporto a Bordeaux. Una menzione speciale è per te, Yohann. Ho trovato in te un vero amico, con un'onestà intellettuale incredibile, troppo facilmente confondibile con l'ingenuità. Grazie per la tua gentilezza infinita.

Infine, sono davvero grato a tutti coloro che in qualche modo mi hanno supportato durante questo percorso, indipendentemente dalla distanza che ci divideva: grazie Anita, Antone, Antonio, Enrico, Giacinto Alberto, Giovanni, Grace, Luca, Maria Antonietta, Mélissande, Paolo, Torquato.



# Contents

<b>List of Figures</b>	<b>viii</b>
<b>List of Tables</b>	<b>xiv</b>
<b>Introduction</b>	<b>1</b>
The Thesis Context and the FUTURPROD Project . . . . .	1
Optimisation, Design and CAD in the ALM Framework . . . . .	3
Thesis Objectives . . . . .	6
Thesis Outline . . . . .	7
<b>Introduction</b>	<b>9</b>
Contexte de la thèse et le projet FUTURPROD . . . . .	9
Intégration de l’optimisation, de la conception et de la CAO . . . . .	11
Objectifs de la thèse . . . . .	14
Structure de la thèse . . . . .	14
<b>1 Literature Review</b>	<b>17</b>
1.1 Introduction to the Literature Review . . . . .	17
1.2 The Additive Layer Manufacturing Technology . . . . .	17
1.2.1 A classification of ALM processes . . . . .	18
1.2.2 The Selective Laser Melting Technology . . . . .	20
1.3 Topology Optimisation Methods . . . . .	23
1.3.1 Density-Based Methods . . . . .	23
1.3.2 The Level-Set Method . . . . .	29
1.4 Implementation of Manufacturing Constraints in Topology Optimisation . . . . .	36
1.5 Conclusions on the Literature Review . . . . .	38
<b>1 Revue de la littérature</b>	<b>41</b>
1.1 Introduction de la revue de la littérature . . . . .	41
1.2 La technologie de fabrication additive par couche (Additive Layer Manufacturing) . . . . .	41



1.2.1	Classification des processus ALM . . . . .	42
1.2.2	La technologie de fusion sélective par laser (SLM) . . . . .	43
1.3	Méthodes d’Optimisation Topologique . . . . .	45
1.3.1	Méthodes basées sur la densité . . . . .	45
1.3.2	La Méthode <i>Level-Set</i> . . . . .	47
1.4	Implémentation des contraintes de l’ALM dans l’OT . . . . .	51
1.5	Conclusions de la revue de la littérature . . . . .	53
<b>2</b>	<b>Fundamentals of Geometrical Modelling</b>	<b>55</b>
2.1	Introduction to the Fundamentals of Geometrical Modelling . . . . .	55
2.2	The NURBS curves theory . . . . .	55
2.3	The NURBS surfaces theory . . . . .	59
2.4	The NURBS hyper-surfaces theory . . . . .	61
2.5	Conclusions on the NURBS entities theory . . . . .	63
<b>2</b>	<b>Principes fondamentaux de la modélisation géométrique</b>	<b>65</b>
2.1	Introduction aux Principes fondamentaux de la modélisation géométrique . . . . .	65
2.2	La théorie des courbes NURBS . . . . .	65
2.3	La théorie des surfaces NURBS . . . . .	66
2.4	La théorie des hyper-surfaces NURBS . . . . .	68
2.5	Conclusions sur la théorie des entités NURBS . . . . .	69
<b>3</b>	<b>Optimisation Methods and Algorithms</b>	<b>71</b>
3.1	Introduction to Optimisation Methods . . . . .	71
3.2	Deterministic Methods for CNLPP . . . . .	74
3.2.1	Generalities on Deterministic Methods . . . . .	74
3.2.2	Optimality Conditions for CNLPP . . . . .	76
3.2.3	Deterministic Algorithms for CNLPP . . . . .	77
3.3	Meta-heuristic Methods for CNLPP . . . . .	81
3.3.1	Generalities on Meta-heuristics . . . . .	81
3.3.2	The Genetic Algorithm BIANCA . . . . .	82
3.3.3	The MATLAB version of BIANCA . . . . .	91
3.4	Conclusions on Optimisation Methods and Algorithms . . . . .	93
<b>3</b>	<b>Méthodes et algorithmes d’optimisation</b>	<b>95</b>
3.1	Introduction aux méthodes d’optimisation . . . . .	95
3.2	Méthodes déterministes pour les problèmes de type CNLPP . . . . .	97
3.2.1	Généralités sur les méthodes déterministes . . . . .	97
3.2.2	Conditions d’optimalité pour les CNLPP . . . . .	98
3.2.3	Algorithmes déterministiques pour les CNLPP . . . . .	98
3.3	Méthodes méta-heuristiques pour les CNLPP . . . . .	100

3.3.1	Généralités sur les méta-heuristiques . . . . .	100
3.3.2	L'algorithme génétique BIANCA . . . . .	100
3.3.3	La version MATLAB de BIANCA . . . . .	103
3.4	Conclusions sur les méthodes et algorithmes d'optimisation . . . . .	104
<b>4</b>	<b>A NURBS-based Topology Optimisation Algorithm</b>	<b>107</b>
4.1	Introduction . . . . .	107
4.2	Mathematical Formulation of the NURBS-based Topology Optimisation Method . . . . .	109
4.3	The algorithm <i>SANTO</i> (SIMP And NURBS for Topology Optimisation) . . . . .	115
4.4	Results Discussion . . . . .	122
4.4.1	The 2D benchmark . . . . .	122
4.4.2	Results for 2D problems: sensitivity to the NURBS surface degrees, number of control points and weights . . . . .	123
4.4.3	Results for 2D problems: comparison between classical and NURBS-based SIMP approaches . . . . .	128
4.4.4	Results for 2D problems: influence of Non-Design Regions . . . . .	129
4.4.5	Results for 2D problems: influence of a symmetry constraint . . . . .	130
4.4.6	The 3D benchmarks . . . . .	131
4.4.7	Results for 3D problems: sensitivity to the NURBS hyper-surface degrees, number of control points and weights . . . . .	133
4.4.8	Results for 3D problems: comparison between classical and NURBS-based SIMP approaches . . . . .	140
4.5	Conclusions and Perspectives on the NURBS-based TO Algorithm . . . . .	144
<b>4</b>	<b>Un algorithme d'Optimisation Topologique basé sur les entités NURBS</b>	<b>149</b>
4.1	Introduction . . . . .	149
4.2	Formulation Mathématique de la Méthode d'Optimisation Topologique basée sur les NURBS . . . . .	150
4.3	L'algorithme <i>SANTO</i> (SIMP And NURBS for Topology Optimisation) . . . . .	152
4.4	Discussion des Résultats . . . . .	154
4.4.1	Le benchmark 2D . . . . .	155
4.4.2	Résultats pour les problèmes 2D : sensibilité aux paramètres de la surface NURBS . . . . .	155
4.4.3	Résultats pour les problèmes 2D : comparaison entre la méthode SIMP classique et l'algorithme basé sur les NURBS . . . . .	156
4.4.4	Résultats pour les problèmes 2D : influence de Non-Design Regions . . . . .	156
4.4.5	Résultats pour les problèmes 2D : influence d'une contrainte de symétrie . . . . .	157
4.4.6	Les benchmarks 3D . . . . .	157
4.4.7	Résultats pour les problèmes 3D : sensibilité aux paramètres de l'hyper-surface NURBS . . . . .	157

4.4.8	Résultats pour les problèmes 3D : comparaison entre la méthode SIMP classique et l'algorithme basé sur les NURBS . . . . .	158
4.5	Conclusions et Perspectives concernant l'Algorithme d'OT basé sur les NURBS160	
<b>5</b>	<b>Geometrical Constraints Implementation in the Framework of the NURBS-based Topology Optimisation Algorithm</b>	<b>163</b>
5.1	Introduction to Geometrical Constraints in Topology Optimisation . . . . .	163
5.2	Poulsen's Formulation of the Minimum Length Scale Constraint . . . . .	167
5.2.1	Mathematical Statement of Poulsen's Minimum Length Scale constraint . . . . .	167
5.2.2	Poulsen's Minimum Length Scale constraint: numerical results . . . . .	169
5.3	Minimum Length Scale control in the NURBS-based TO Algorithm . . . . .	171
5.3.1	Minimum length scale resulting from B-Spline entities . . . . .	171
5.3.2	Some remarks about the proposed approach . . . . .	180
5.3.3	The effects of the NURBS weights on the minimum length scale . . . . .	181
5.3.4	Results: Minimum length scale in 2D . . . . .	184
5.3.5	Results: Minimum length scale in 3D . . . . .	186
5.3.6	The effects of a non-uniform knot vector on the minimum length scale	191
5.4	The Maximum Length Scale . . . . .	194
5.4.1	Mathematical Statement of the Maximum Length Scale constraint . . . . .	194
5.4.2	Results: Maximum Length Scale in 2D . . . . .	196
5.4.3	Results: Maximum Length Scale in 3D . . . . .	198
5.5	The Minimum Curvature Radius . . . . .	199
5.5.1	Mathematical Statement of the Minimum Curvature Radius constraint . . . . .	199
5.5.2	Results on the application of the Minimum Curvature Radius constraint . . . . .	201
5.6	Conclusions and Perspectives on Geometrical Constraints in the NURBS-based TO algorithm . . . . .	202
<b>5</b>	<b>Implémentation de Contraintes Géométriques dans l'Algorithme d'OT basé sur les NURBS</b>	<b>207</b>
5.1	Introduction aux Contraintes Géométriques en OT . . . . .	207
5.2	La formulation de Poulsen pour la contrainte de l'épaisseur minimale . . . . .	208
5.2.1	L'énoncé mathématique de la contrainte de l'épaisseur minimale selon Poulsen . . . . .	208
5.2.2	La contrainte de l'épaisseur minimale selon Poulsen : résultats . . . . .	209
5.3	Le contrôle de l'épaisseur minimale dans l'algorithme d'OT basé sur les NURBS . . . . .	210
5.3.1	L'épaisseur minimale résultant des entités B-Spline . . . . .	210
5.3.2	Quelques remarques sur l'approche proposée . . . . .	211

5.3.3	Les effets des poids des NURBS sur l'épaisseur minimale . . . . .	212
5.3.4	Résultats : l'épaisseur minimale en 2D . . . . .	212
5.3.5	Résultats : l'épaisseur minimale en 3D . . . . .	213
5.3.6	Les effets d'un <i>knot vector</i> non uniforme sur l'épaisseur minimale . .	214
5.4	L'épaisseur maximale dans l'algorithme d'OT basé sur les NURBS . . . . .	214
5.4.1	L'énoncé mathématique de la contrainte de l'épaisseur maximale . .	214
5.4.2	Résultats : l'épaisseur maximale en 2D . . . . .	215
5.4.3	Résultats : l'épaisseur maximale en 3D . . . . .	216
5.5	Le rayon minimal de courbure . . . . .	216
5.5.1	Enoncé mathématique de la contrainte sur le rayon minimal de courbure	216
5.5.2	Résultats : application de la contrainte sur le rayon minimal de courbure	217
5.6	Conclusions et Perspectives concernant les contraintes géométriques dans l'Algorithme d'OT basé sur les NURBS . . . . .	217
<b>6</b>	<b>Eigenvalue Problems in the Framework of the NURBS-Based Topology Optimisation Algorithm</b>	<b>219</b>
6.1	Introduction . . . . .	219
6.2	Eigenvalue Buckling Problems . . . . .	223
6.2.1	First buckling load maximisation in the classic SIMP framework . .	223
6.2.2	Mathematical formulation of buckling problems in the NURBS- based TO algorithm . . . . .	226
6.2.3	Discussion on numerical aspects . . . . .	226
6.2.4	Results . . . . .	229
6.3	Eigen-frequencies Problems . . . . .	235
6.3.1	First Eigen-frequency maximisation in the classic SIMP framework	235
6.3.2	Mathematical formulation of eigen-frequencies problems in the NURBS- based TO algorithm . . . . .	237
6.3.3	Discussion on numerical aspects . . . . .	237
6.3.4	Results . . . . .	238
6.4	Conclusions and Perspectives . . . . .	243
<b>6</b>	<b>Problèmes aux valeurs propres dans le cadre de l'algorithme d'OT basé sur les NURBS</b>	<b>247</b>
6.1	Introduction . . . . .	247
6.2	Problèmes de flambage aux valeurs propres . . . . .	249
6.2.1	Maximisation de la première charge critique de flambage dans la méthode SIMP classique . . . . .	249
6.2.2	Enoncé mathématique des problèmes de flambage dans l'algorithme d'OT basé sur les NURBS . . . . .	250
6.2.3	Discussion sur les aspects numériques . . . . .	251
6.2.4	Résultats . . . . .	251

6.3	Problèmes de fréquences propres . . . . .	253
6.3.1	Maximisation de la première fréquence propre dans la méthode SIMP classique . . . . .	253
6.3.2	Énoncé mathématique des problèmes de fréquences propres dans l’algorithme d’OT basé sur les NURBS . . . . .	254
6.3.3	Discussion sur les aspects numériques . . . . .	254
6.3.4	Résultats . . . . .	254
6.4	Conclusions et Perspectives . . . . .	256
<b>7</b>	<b>General Fitting Techniques for Curves and Surfaces Reconstruction</b>	<b>259</b>
7.1	Introduction to Curve and Surface Reconstruction . . . . .	259
7.2	A General Hybrid Optimisation Strategy for Curve Fitting in the NURBS Framework . . . . .	263
7.2.1	Mathematical Formulation of the Curve Fitting Problem . . . . .	263
7.2.2	Numerical Strategy . . . . .	268
7.2.3	Studied Cases and Results for curve fitting . . . . .	269
7.3	Surface Reconstruction in the NURBS Framework . . . . .	280
7.3.1	Surface Parametrisation . . . . .	281
7.3.2	Surface Fitting . . . . .	282
7.3.3	Results on Surface Fitting . . . . .	284
7.4	Conclusions and Perspectives on Approximation Problems . . . . .	285
<b>7</b>	<b>Techniques de Fitting générales pour la Reconstruction CAO de Courbes et de Surfaces</b>	<b>291</b>
7.1	Introduction à la Reconstruction de Courbes et Surfaces . . . . .	291
7.2	Une stratégie d’Optimisation Hybride Générale pour le Fitting des Courbes dans le cadre NURBS . . . . .	293
7.2.1	L’énoncé mathématique du problème de Fitting de Courbes . . . . .	293
7.2.2	Stratégie Numérique . . . . .	295
7.2.3	Cas d’étude et Résultats pour le Fitting de Courbes . . . . .	295
7.3	Reconstruction de Surfaces le cadre des NURBS . . . . .	298
7.3.1	Paramétrage de Surfaces . . . . .	298
7.3.2	Fitting de Surfaces . . . . .	299
7.3.3	Résultats du Fitting de Surfaces . . . . .	300
7.4	Conclusions sur les Problèmes d’Approximation . . . . .	300
	<b>Conclusions and Perspectives</b>	<b>303</b>
	General Conclusions . . . . .	303
	Perspectives of this Thesis . . . . .	305

<b>Conclusions et Perspectives</b>	<b>309</b>
Conclusions Générales . . . . .	309
Les Perspectives de la Thèse . . . . .	311
<b>Annexe A</b>	<b>315</b>
NURBS entities : derivatives computation . . . . .	315
<b>Annexe B</b>	<b>317</b>
Sensitivity analysis of the minimum member size constraint (Poulsen's formulation)	317
<b>Annexe C</b>	<b>319</b>
Sensitivity analysis of the maximum member size constraint . . . . .	319
<b>Annexe D</b>	<b>321</b>
Sensitivity analysis of the minimum curvature radius constraint . . . . .	321
<b>Annexe E</b>	<b>323</b>
Details of the NURBS curve of the flame problem . . . . .	323
<b>Bibliography</b>	<b>325</b>
<b>List of Publications issued from this thesis</b>	<b>339</b>



# List of Figures

1	ALM Structures Design Loop . . . . .	5
1.1	Critical Angle for supported overhangs. . . . .	22
1.2	Representation of a general 3D structure through the SIMP fictitious density field. . . . .	25
1.3	Classic SIMP algorithm. . . . .	28
1.4	Example of Level Set Function. . . . .	30
1.5	Overview of the LSM. . . . .	31
1.6	Boundary evolution with (left) or without (right) “viscous term” [1]. . . . .	34
2.1	Example of Blending Functions for B-Spline/NURBS curves of degree $p_1 = 3$ and number of control points $n_1 = 6$ . . . . .	57
2.2	Example of B-Spline and NURBS curves of degree $p_1 = 3$ and number of control points $n_1 = 6$ . . . . .	58
2.3	B-Spline and NURBS surfaces with same degrees, control points and knot vectors. . . . .	60
2.4	B-Spline and NURBS hyper-surfaces with same degrees, control points and knot vectors. . . . .	64
2.5	Iso-level corresponding to 10 of B-Spline and NURBS hyper-surfaces. . . . .	64
3.1	Classification of optimisation algorithm. . . . .	73
3.2	Overview of the SQP algorithm. . . . .	78
3.3	Overview of the IP algorithm. . . . .	80
3.4	GAs: an example. . . . .	83
3.5	Individual Structure for problem (3.9). . . . .	84
3.6	Example of optimisation of modular system. . . . .	86
3.7	The GA BIANCA. . . . .	87
3.8	Reproduction phase in a standard GA. . . . .	89
3.9	Modified reproduction phase between different species for optimisation of modular systems. . . . .	90
3.10	The general individual’s structure for the MATLAB version of BIANCA . . . . .	92



4.1	SIMP pseudo-density representation through NURBS entities. . . . .	110
4.2	SANTO flowchart . . . . .	116
4.3	Examples of possible mirror symmetries for 3D TO problems . . . . .	116
4.4	Example of embedding domain, reference system and control points for a 3D component. . . . .	117
4.5	Postprocessing procedure for 2D structures. . . . .	120
4.6	Postprocessing procedure for 3D structures. . . . .	121
4.7	Cantilever plate problem - $a_1 = 320$ mm, $a_2 = 200$ mm, Thickness $t = 2$ mm, Young's Modulus $E = 72000$ MPa, Poisson's ratio $\nu = 0.33$ , Load $P = 1000$ N. . . . .	123
4.8	B-Spline results for $p_1, p_2 = 2$ . . . . .	124
4.9	NURBS results for $p_1, p_2 = 2$ . . . . .	124
4.10	B-Spline results for $p_1, p_2 = 3$ . . . . .	125
4.11	NURBS results for $p_1, p_2 = 3$ . . . . .	125
4.12	B-Spline results for $p_1, p_2 = 4$ . . . . .	125
4.13	NURBS results for $p_1, p_2 = 4$ . . . . .	126
4.14	Cantilever plate problem - True compliance trends. . . . .	126
4.15	Cantilever plate problem - Projected compliance trends. . . . .	127
4.16	Hyperworks-OptiStruct <sup>®</sup> solution of the cantilever plate problem: $c =$ $398.66$ Nmm, $V/V_{ref} = 0.3992$ . . . . .	129
4.17	Cantilever plate problem with NDRs: design domain in white, prescribed material NDR in red, prescribed void NDR in yellow. . . . .	129
4.18	Cantilever plate problem with NDRs. . . . .	130
4.19	Cantilever plate problem with symmetry constraint. . . . .	131
4.20	Problem settings of the first benchmark . . . . .	132
4.21	Problem settings of the second benchmark . . . . .	133
4.22	Benchmark 1, $p_1 = p_2 = p_3 = 2$ , $n_{tot} = 36 \times 6 \times 10$ . . . . .	135
4.23	Benchmark 1, $p_1 = p_2 = p_3 = 2$ , $n_{tot} = 40 \times 8 \times 12$ . . . . .	135
4.24	Benchmark 1, $p_1 = p_2 = p_3 = 2$ , $n_{tot} = 46 \times 10 \times 14$ . . . . .	135
4.25	Benchmark 1, $p_1 = p_2 = p_3 = 3$ , $n_{tot} = 36 \times 6 \times 10$ . . . . .	136
4.26	Benchmark 1, $p_1 = p_2 = p_3 = 3$ , $n_{tot} = 40 \times 8 \times 12$ . . . . .	136
4.27	Benchmark 1, $p_1 = p_2 = p_3 = 3$ , $n_{tot} = 46 \times 10 \times 14$ . . . . .	136
4.28	Benchmark 1, $p_1 = p_2 = p_3 = 4$ , $n_{tot} = 36 \times 6 \times 10$ . . . . .	137
4.29	Benchmark 1, $p_1 = p_2 = p_3 = 4$ , $n_{tot} = 40 \times 8 \times 12$ . . . . .	137
4.30	Benchmark 1, $p_1 = p_2 = p_3 = 4$ , $n_{tot} = 46 \times 10 \times 14$ . . . . .	137
4.31	Synthetic results: Objective function vs Iterations . . . . .	138
4.32	Objective function ratio vs hyper-surfaces degrees . . . . .	139
4.33	Solutions of the first benchmark provided by the commercial software OptiStruct <sup>®</sup> 140	
4.34	Comparison of the proposed methodology and the commercial software OptiStruct <sup>®</sup> . . . . .	142

4.35	Comparison for the second benchmark case. . . . .	143
5.1	Cantilever plate problem, $d_{min} = 16$ mm. . . . .	170
5.2	Cantilever plate problem, $d_{min} = 20$ mm. . . . .	170
5.3	Cantilever plate problem, $d_{min} = 25$ mm. . . . .	170
5.4	Simulation of 2D minimum length scale . . . . .	172
5.5	Simulation of 3D minimum length scale. . . . .	173
5.6	$d_{min}$ trends in 2D, $p_j = 3$ . . . . .	175
5.7	$d_{min}$ trends in 3D, $p_j = 3$ . . . . .	175
5.8	$d_{min}$ vs. $\Delta X$ , $p_j = 3$ . . . . .	176
5.9	$d_{min}$ vs. $\Delta X$ , $p_j = 2$ . . . . .	176
5.10	$d_{min}$ vs. $\Delta X$ , $p_j = 4$ . . . . .	177
5.11	$d_{min}$ vs. $n/a$ , $p_j = 2$ . . . . .	177
5.12	$d_{min}$ vs. $n/a$ , $p_j = 3$ . . . . .	178
5.13	$d_{min}$ vs. $n/a$ , $p_j = 4$ . . . . .	178
5.14	$d_{min}^B$ vs. $\Delta X_0$ , $p_j = 2$ . . . . .	179
5.15	$d_{min}^B$ vs. $\Delta X_0$ , $p_j = 3$ . . . . .	179
5.16	$d_{min}^B$ vs. $\Delta X_0$ , $p_j = 4$ . . . . .	180
5.17	The 2D optimised topology provided by the NURBS-based SIMP method. . . . .	182
5.18	Weights and Control Points trends for a NURBS solution. . . . .	183
5.19	2D benchmark - Geometric parameters: $a_1 = 320$ mm, $a_2 = 200$ mm, thickness $t = 2$ mm - Material parameters: $E = 72000$ MPa, $\nu = 0.33$ - Mesh: $96 \times 60$ SHELL181 elements - Load: $P = 1000$ N. . . . .	184
5.20	Procedure to measure the actual minimum member size. . . . .	185
5.21	2D solutions - minimum length scale, $p_j = 3$ . . . . .	186
5.22	3D benchmark - Geometric parameters: $a_1 = 400$ mm, $a_2 = 100$ mm, $a_3 = 200$ mm - Material parameters: $E = 72000$ MPa, $\nu = 0.33$ - Mesh: $64 \times 16 \times 32$ SOLID185 elements - Load: $P = 2000$ N. . . . .	187
5.23	Highlight of the minimum length scale in 3D, B-Spline solution, $p_j = 3$ , $24 \times 6 \times 12$ control points. . . . .	189
5.24	Highlight of the minimum length scale in 3D, B-Spline solution, $p_j = 3$ , $30 \times 12 \times 18$ control points. . . . .	189
5.25	Highlight of the minimum length scale in 3D, B-Spline solution, $p_j = 3$ , $36 \times 18 \times 24$ control points. . . . .	190
5.26	Highlight of the minimum length scale in 3D, NURBS solution, $p_j = 3$ , $24 \times 6 \times 12$ control points. . . . .	190
5.27	Highlight of the minimum length scale in 3D, NURBS solution, $p_j = 3$ , $30 \times 12 \times 18$ control points. . . . .	191
5.28	Highlight of the minimum length scale in 3D, NURBS solution, $p_j = 3$ , $36 \times 18 \times 24$ control points. . . . .	191
5.29	Knot vectors distributions. . . . .	192

5.30	B-Spline solutions with different knot vector distributions. . . . .	193
5.31	NURBS solutions with different knot vector distributions. . . . .	193
5.32	Traction plate problem - $a_1 = 400$ mm, $a_2 = 200$ mm, Thickness $t = 2$ mm, Young's Modulus $E = 72000$ MPa, Poisson's ratio $\nu = 0.33$ , Load $P = 1000$ N. . . . .	196
5.33	NURBS-based SIMP solutions for the traction plate problem (a) without and (b) with maximum member size constraint. . . . .	197
5.34	OptiStruct <sup>®</sup> solution of the traction plate problem (a) without and (b) with maximum member size constraint. . . . .	197
5.35	3D benchmark - Geometric parameters: $a_1 = 500$ mm, $a_2 = 100$ mm, $a_3 = 200$ mm - Material parameters: $E = 72000$ MPa, $\nu = 0.33$ - Mesh: $60 \times 16 \times 24$ SOLID185 elements - Load: $P = 5000$ N. . . . .	199
5.36	3D TO result - Compliance minimisation with equality constraint on the volume. . . . .	199
5.37	3D TO result - Compliance minimisation with equality volume constraint and maximum member size constraint. . . . .	200
5.38	Solution of the cantilever plate problem (a) without and (b) with the minimum local curvature radius constraint . . . . .	202
6.1	The $f_\sigma/f_\lambda$ ratio. . . . .	228
6.2	The base-thickness as Non-Design Region. . . . .	228
6.3	Geometric parameters: $a_1 = 900$ mm, $a_2 = 300$ mm, $d = 20$ mm, thickness $t = 2$ mm, symmetry plane $x_2 = a_2/2$ - Material Parameters: Steel, Young's modulus $E = 200000$ MPa, Poisson's ratio $\nu = 0.3$ - Loads: Unit vectors . . . . .	229
6.4	Double penalisation scheme: B-Spline results, $p_1, p_2 = 2$ . . . . .	231
6.5	Double penalisation scheme: B-Spline results, $p_1, p_2 = 3$ . . . . .	231
6.6	Double penalisation scheme: B-Spline results, $p_1, p_2 = 4$ . . . . .	231
6.7	Double penalisation scheme: NURBS results, $p_1, p_2 = 2$ . . . . .	231
6.8	Double penalisation scheme: NURBS results, $p_1, p_2 = 3$ . . . . .	231
6.9	Double penalisation scheme: NURBS results, $p_1, p_2 = 4$ . . . . .	232
6.10	Double penalisation technique: synthesis of results. . . . .	232
6.11	Base-thickness strategy: B-Spline results for $p_1, p_2 = 2$ . . . . .	233
6.12	Base-thickness strategy: B-Spline results for $p_1, p_2 = 3$ . . . . .	233
6.13	Base-thickness strategy: B-Spline results for $p_1, p_2 = 4$ . . . . .	233
6.14	Base-thickness strategy: NURBS results for $p_1, p_2 = 2$ . . . . .	233
6.15	Base-thickness strategy: NURBS results for $p_1, p_2 = 3$ . . . . .	233
6.16	Base-thickness strategy: NURBS results for $p_1, p_2 = 4$ . . . . .	234
6.17	Base-thickness strategy: synthesis of results. . . . .	234
6.18	The $f_\omega/\rho_e$ ratio. . . . .	238

6.19	Geometric parameters: $a_1 = a_2 = 160$ mm, $d = 20$ mm, thickness $t = 2$ mm - Material Parameters: Aluminium Alloy, Young's modulus $E = 72000$ MPa, Poisson's ratio $\nu = 0.33$ - Non-Structural Mass: $m_{NS} = 40$ Kg. . . . .	239
6.20	Eigen-frequencies problem: B-Spline results, $p_1, p_2 = 2$ . . . . .	241
6.21	Eigen-frequencies problem: B-Spline results, $p_1, p_2 = 3$ . . . . .	241
6.22	Eigen-frequencies problem: B-Spline results, $p_1, p_2 = 4$ . . . . .	241
6.23	Eigen-frequencies problem: NURBS results, $p_1, p_2 = 2$ . . . . .	242
6.24	Eigen-frequencies problem: NURBS results, $p_1, p_2 = 3$ . . . . .	242
6.25	Eigen-frequencies problem: NURBS results, $p_1, p_2 = 4$ . . . . .	242
6.26	Eigen-frequencies problem: synthesis of results. . . . .	243
7.1	Trend $\kappa_{\Gamma_{max}}$ vs $p$ for the curve $\Gamma$ . . . . .	266
7.2	Overview of the global numerical strategy for the curve fitting problem . . . . .	268
7.3	The individual's structure for the curve fitting problem . . . . .	269
7.4	The Descartes' Folium . . . . .	272
7.5	The four-leaf clover . . . . .	273
7.6	The flame . . . . .	274
7.7	Detail on the NURBS approximating the flame . . . . .	274
7.8	The tennis ball stitching . . . . .	275
7.9	Starting data set for the paddle problem . . . . .	276
7.10	The paddle . . . . .	277
7.11	Sensitivity to the number of TPs: approximating curves . . . . .	279
7.12	Overview of the Surface Reconstruction scheme. . . . .	281
7.13	Surface Fitting: Paraboloid. . . . .	285
7.14	Surface Fitting: Shell Surface. . . . .	285
7.15	Surface Fitting: Klein's Surface. . . . .	286
7.16	Surface Fitting: Trefoil Surface. . . . .	286



# List of Tables

- 5.1 Minimum length scale for 2D B-Spline solutions . . . . . 185
- 5.2 Minimum length scale for 2D NURBS solutions . . . . . 187
- 5.3 Minimum length scale for 3D B-Spline solutions . . . . . 188
- 5.4 Minimum length scale for 3D NURBS solutions . . . . . 189
  
- 7.1 Setting of genetic parameters . . . . . 270
- 7.2 Setting of variables boundaries. . . . . 270
- 7.3 Genetic Algorithm: Numerical Results. . . . . 271
- 7.4 Gradient Algorithm: Numerical Results. . . . . 271
- 7.5 Maximum allowed curvature values . . . . . 276
- 7.6 Sensitivity to the number of TPs - Numerical results . . . . . 280
- 7.7 Surface Fitting - Data . . . . . 285
- 7.8 Surface Fitting - Numerical results . . . . . 286



# Introduction

## The Thesis Context and the FUTURPROD Project

From 1980s, the *mass production* (also known as *flow production*) concept started to turn into *mass customisation*: companies have to stimulate and attract new customers by proposing a large gamma of products in order to answer to different needs. In the 2000s, this phenomenon becomes really important, as the market undergoes saturation and direct contacts between customers and producers become possible: customers expect exactly the product that they are asking for and in a limited time lapse (*personalised production*) [2]. The emerging *Additive Layer Manufacturing* technology (ALM) seems to meet all these market requirements. Producing mechanical parts starting from a numerical model by adding material has totally changed the way to conceive the manufacturing process. In the last decade, many companies invested in ALM worldwide and the growth of ALM machine implantation has been exponential during the years [3]. Nowadays, ALM offers many opportunities for diverse applications which can be summarised in *rapid prototyping (3D printing)*, *rapid tooling* and *direct manufacturing*. The most of the ALM applications still concern the *rapid prototyping* [3]: functional parts, i.e. parts that are designed and produced by means of ALM to be directly employed, are about 20% and this percentage is restricted almost only to aeronautic, automotive and biomedical fields. Nevertheless, an increasing effort coming from both the scientific and industrial world is put on the development of the *direct manufacturing* [3]. This is due to four main opportunities that ALM offers. Products can have a really complex shape or a multi-material composition (**Design Freedom**). Furthermore, ALM ensures **Environmental Efficiency**, since it needs only the material which actually constitutes the final product, so the mass/energy flow is minimised. ALM does not need specific tools depending on the different productions, so it is suited to small series, eliminating the stocking problem (**Flexibility**). Finally, Design Freedom together with Flexibility allow for producing highly personalised components without additional costs (**Personalisation**).

Although ALM offers undeniable advantages, it exhibits some limitations hampering the systematic use of this technology in the industrial world. These limitations concern both the manufacturing process itself and the lack of know-how which are of paramount importance in order to integrate ALM in a standard industrial numerical chain, i.e. from



preliminary design to certification. New questions naturally arise when dealing with ALM. For instance, what is the surface quality of the obtained structures? How is it possible to keep control of bulk material defects? How does the micro-structure affect the macroscopic mechanical properties? Which are the optimum machine parameters to produce components by ALM? In what phase of the design flow should optimisation intervene in order to fully exploit the potential of ALM? Answering these questions is a key-point in order to fill the gap between the requirements demanded by technical specifications and the effectively produced piece. Therefore, it seems evident that the industrialisation of ALM process passes through the development of mathematical models and algorithms capable of integrating the specificity of the manufacturing process as well as the related physical phenomena since the first (preliminary) stage of the design process (*Design for Manufacturing Methods*).

The work related to this Ph.D. thesis has been carried out in the framework of the FUTURPROD project, which has been co-founded by the Nouvelle-Aquitaine region and by several industrial partners: AGB, Ariane Group, Polyshape and Stelia Aerospace. The main objective of the FUTURPROD project consists of encouraging the development of the ALM technology for metal materials in Nouvelle-Aquitaine region by establishing a collaboration between industrial partners and local research laboratories, as the “Institut de Mécanique d’Ingénierie” (I2M) of Bordeaux. The project has been conceived, according to the industrial partners requirements, for the *Selective Laser Melting* (SLM) technology. Thus, some tasks of the project are really specific for this additive manufacturing technology, whilst others are really general. Particularly, three research axes have been identified (each objective corresponds to a Ph.D. thesis):

1. *Characterising and optimising the SLM manufacturing process.* Since SLM technologies involve complex phenomena (metal melting, temperature gradients, high chemical reactivity of metals with oxygen, etc.), it is basic to understand which main parameters have to be controlled in order to produce high-quality and reliable mechanical components. As a matter of fact, the laser power, the speed of the laser beam, the powder quality in terms of grain density, geometry and oxidation are all important parameters affecting the final result. Moreover, this work should clarify how much oxidation is critical in SLM and if it is possible to control the oxidation rate in built up structures.
2. *Fatigue phenomena of SLM structures.* Nowadays there is an increasingly strong interest in producing functional parts by ALM (*direct manufacturing*). Since fatigue is still one of the most common causes of failure for in operation structures, it is basic to understand how the SLM process affects the behaviours of mechanical components in this context. The challenge of this research axis is to set up a numerical model capable of taking into account the micro-structure resulting from the SLM process (presence of defects, heterogeneity) and it should simulate the damage mechanism.

Then, correctly forecasting the fatigue limit and comparing it with the fatigue limit of the base material are fundamental tasks.

3. *Design of ALM structures.* The consequence of ALM is twofold. On the one hand, ALM allows for producing very intricate shapes and topologies, which often are the result of a dedicated optimisation process; on the other hand, increasing the design freedom makes the design flow trickier because new requirements and constraints related to the specific ALM process and to the related physical phenomena arise. Identifying the critical points in the design flow of a structure obtained by means of a given ALM process and proposing a consistent strategy to overcome these difficulties becomes essential. This task can be summarised through the following question: is it possible to develop a suitable optimisation method able to properly integrate, since the early stage of the design process, the features of the ALM technology? When dealing with ALM structures, the bound between design and optimisation is so intimate that it would be better to introduce the “Optimum Design for Additive Manufacturing” concept instead of the simple “Design for Manufacturing” [4].

This Ph.D. thesis deals with the last research axis of the FUTURPROD project.

## Issues related to the Integration of Optimisation, Design and CAD in the ALM Framework

The wide possibilities in terms of shapes and topologies allowed by ALM as well as its high costs advise to make use of this technology for high-performances, optimised components. Therefore, it seems natural to take into account an optimisation task during the design process. This optimisation step can be carried out at different levels, according to the structure nature. Classically, structural optimisation is divided in three categories [5]. The first one is the *Sizing* (or Parametric) *Optimisation* [6]: some geometric/mechanical quantities are identified as design variables and the optimisation consists of finding the optimum values of these variables in order to minimise/maximise a certain response. An example in this sense is the minimisation of the production (forming, welding) costs of a cylindrical vessel, capped at both ends by hemispherical heads. Shell thickness, heads thickness, inner radius and length of the cylindrical part can be considered as design variables [7]. The second family of structural optimisation problems is referred as *Shape Optimisation* (SO) [8]: as suggested by the name, the boundaries of a structure of a given topology can evolve during optimisation in order to reach a precise goal, but the topology of the structure remains unchanged. This kind of optimisation is more challenging than *Parametric Optimisation* because keeping control of the boundaries and avoiding intersections, overlapping and singularities is a really complicated task. The last structural optimisation category is *Topology Optimisation* (TO). Generally speaking,

TO aims at determining the optimum “properties” defining the *topological space* (e.g. *connectedness* and *compactness*). The TO problem is very hard to be solved by means of a purely geometric approach. The most common practice is to transform such a problem into a simpler one: the problem of finding the “optimum topology” of a given space can be reformulated into the problem of determining the optimum material distribution over a given set [5,9,10]. In this background, at least two material phases (“solid” and “void”) are distributed in a prescribed design domain to minimise a given cost function subjected to some constraints, characterising the problem at hand.

Currently, well-established software packages (for instance, Altair OptiStruct<sup>®</sup> [11], TOSCA [12]) allow for performing both SO and TO of structures. Even though industrial companies can efficiently carry out optimisation through these software, integrating the specificity of the ALM process is far to be an easy task because of several reasons. Firstly, the results interpretation is imprecise, since the optimised geometries are described through Finite Elements (FE) and not by means of geometric entities. This fact does not permit to uniquely retrieve the boundaries of the optimised structure in a post-processing phase. Secondly, the optimisation constraints of different nature (geometric, mechanic, technological, etc.) are not necessarily met on the actual geometry at the end of the optimisation process. Of course, the numerical procedure converges towards a feasible solution when optimisation is carried out by means of a FE calculation performed on the mesh of the overall domain, wherein the optimum solution is embedded. Actually, optimisation constraints are not met on the final, reassembled design, wherein the structure boundary must be represented through suitable geometric entities. Finally, although research is really proactive on these topics, at this stage it is not possible to impose specific, ALM-oriented, constraints, e.g. considering overhangs and support structures according to the building direction, temperature gradients, minimum/maximum dimensions of structural features: indeed, these constraints are either not supported by commercial software or not consistent on the final reassembled topology, as stated. All these aspects make the design flow ineffective and several design - optimisation - CAD loops are required before converging on an “optimal and manufacturable” solution (Fig. 1 offers an idea of the overall design flow):

- 1) A first design of a structure is considered. Let it be the “current” design.
- 2) A TO task is performed on the current design by means of a standard commercial software.
- 3) The optimisation provides a FE-based design that is far from being compatible with a CAD environment, where the design phase takes place.
- 4) A reconstruction/reassembly phase is necessary in order to transform pixels and voxels into surfaces and volumes, and to convert the jagged boundary into curves and surfaces, respectively.

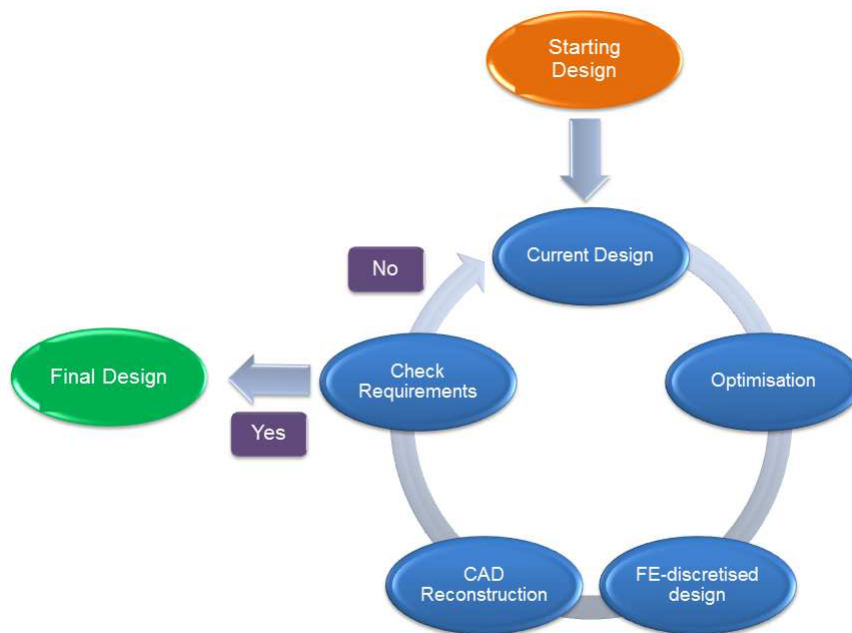


Figure 1 – ALM Structures Design Loop

- 5) After the reconstruction phase, the technical requirements are checked and, usually, several design loops are needed to obtain an optimised structure meeting the full set requirements.

The CAD reconstruction can be identified as the most important weakness of the loop of Fig. 1. Currently, some dedicated tools are available to achieve a CAD compatible design. For instance, the OSSmooth module has been implemented in Altair OptiStruct<sup>®</sup>. The boundary of a 2D structure can be exported in standard CAD exchange file format, like IGES (Initial Graphics Exchange Specification) or STP (STandard for the Exchange of Product model data). In 3D, the boundaries (which should be surfaces in a CAD environment) cannot be easily retrieved. If an IGES file is demanded, OSSmooth returns a set of disconnected patches that only apparently reproduces the solid boundaries. Otherwise, an STL file (Standard Tessellation Language) can be obtained: in this case the boundary of the 3D optimised structure is approximated *via* triangles, which are defined through their vertices and the external normal components. Anyway, reassembling the optimum final solution into the CAD environment requires a consistent amount of arbitrary decisions taken by the designer and the resulting design could be really far from that provided by the TO software [13]. Of course, passing from a FE-discretised design to a CAD-compatible geometry introduces some kind of approximation and, consequently, an

inconsistency between the two configurations is not surprising. Indeed, the direct result of TO process, i.e. the optimum distribution of the material phases over a given design domain, is not CAD-compatible. Moreover, since ALM constraints are extremely specific and induce deep, non-intuitive modifications in the structure lay-out, they should be taken into account from the starting phases of the design loop, i.e. at the stage of TO. Considering ALM constraints only in a post-processing phase could imply to completely overturn the optimised configuration, making the TO algorithm result useless.

These are the reasons why nowadays widespread TO software like OptiStruct<sup>®</sup> or TO-SCA are mainly used to achieve a pseudo-optimum design that is taken as guideline by the designer in order to improve somehow the current performances of a specific component.

## Thesis Objectives

In the light of the aforementioned issues, this Ph.D. thesis aims at providing a general methodology to design and optimise components that are supposed to be produced by means of the ALM technology. In this context, the following main objectives can be identified:

1. The first requirement to be satisfied is the mathematical formulation and the development of an innovative TO algorithm, whose basic feature is giving a description of the optimum topology by means of purely geometric entities and not on finite elements. The new TO algorithm should have an intrinsic CAD compatibility because of two main reasons: on the one hand, the CAD reconstruction phase would be simplified and, on the other hand, the optimised configuration would be consistent with respect to optimisation constraints. To this purpose, B-Spline and NURBS entities have been naturally chosen to constitute the fundamentals of the new TO method presented in this thesis.
2. Secondly, it is sought to formulate suitable optimisation constraints which should be capable of taking into account the basic requirements of the ALM technology. These constraints should be met on the final, reassembled geometry.
3. Furthermore, the new mathematical formulation of the TO problem should not introduce negative side effects. Therefore, it must guarantee to take into account several mechanical quantities which are already considered in standard TO software, without any added restriction.
4. The intrinsically CAD compliant nature of the algorithm to be developed should simplify the CAD reconstruction task, but it needs to be supported: the final objective of this work is to develop suitable mathematical techniques and tools which

should be capable of reducing the dependence of the reconstruction phase on the designer's arbitrary decisions. These devices should lead the designer to make choices which are based on optimisation criteria and not on "engineering sensitivity".

It is not possible to separate and categorise the scopes of this work in "scientific objectives" and "industrial objectives": it is evident, from this preliminary discussion, that the development of an effective TO method able to integrate ALM technology specificity is a basic industrial need that implies both fundamental and applied research activities.

## Thesis Outline

The structure of this manuscript has been conceived in order to clearly answer to the four objectives characterising this Ph.D. thesis.

As far as the bibliographic study is concerned, the following choice has been made. A literature review is provided in Chapter 1, whose main aspects are essentially general features of ALM technologies, TO methods, and the formulation of some ALM constraints in TO algorithms. These topics have been gathered together either because they are basic to better understand the thesis framework or because the discussed ideas are frequently recalled all along the manuscript. More specific and shorter states of the art are proposed at the beginning of Chapters 4-7 in order to provide the main references for the problem at hand.

Chapter 2 is dedicated to the NURBS theory. NURBS entities are clearly defined and their main properties are discussed through some meaningful example.

Since this Ph.D. thesis deals with optimisation in general and not only with TO techniques, a brief overview on deterministic and on meta-heuristic methods is provided in Chapter 3.

Chapter 4 aims at answering to the needs expressed by objective 1. Particularly, the bases of surfaces and hyper-surfaces NURBS theory of Chapter 2 are suitably combined with the gradient algorithms described in Chapter 3 in order to provide an effective and CAD-compatible TO tool. The efficiency of the proposed approach is tested on the well-established problem of compliance minimisation subject to a volume equality constraint.

Chapter 5 is related to objective 2. Some constraints of geometric nature (length scale control, curvature radius), which are essential when the ALM technology is used, have been formulated from of a mathematical viewpoint in the framework of the algorithm developed in Chapter 4.

Objective 3 is considered in Chapter 6: the proposed NURBS-based TO algorithm is enhanced with two other mechanical quantities, which are often considered in the design of structures: buckling loads and eigen-frequencies.

Chapter 7 deals with the topics issued from objective 4. A significant contribution has been provided in the context of the curves approximation problem and its generalisation to the surface approximation problem is discussed as well.

Finally, Conclusions and Perspectives of this work are discussed.

The scientific production related to this Ph.D. thesis is provided at the end of this document.

# Introduction

## Contexte de la thèse et le projet FUTURPROD

A partir des années 1980, le concept de production de masse a commencé à se transformer vers un concept de personnalisation de la production : les entreprises doivent attirer de nouveaux clients en proposant un large éventail de produits pour répondre au mieux aux différents besoins. Dans les années 2000, ce phénomène devient encore plus prépondérant, le marché étant très concurrentiel et des liens directs entre clients et producteurs devenant nécessaires : les clients sont en attente d'un produit qui réponde exactement à ce qu'ils demandent et dans un temps limité (production personnalisée) [2].

La technologie émergente de la fabrication additive par couche (*Additive Layer Manufacturing* - ALM) semble pouvoir répondre à toutes les exigences du marché. En effet, produire des pièces mécaniques à partir d'un modèle numérique en utilisant du matériau de manière additive a radicalement changé la façon de concevoir les pièces et d'imaginer le processus de fabrication. Au cours de la dernière décennie, de nombreuses entreprises ont ainsi investi dans l'ALM dans le monde entier et la croissance de l'implantation des machines ALM est exponentielle [3]. De nos jours, l'ALM offre de nombreuses opportunités pour diverses applications qui peuvent être catégorisées de la manière suivante : le *prototypage rapide* (impression 3D), l'*outillage rapide* et la *fabrication directe*. La plupart des applications ALM concernent encore le prototypage rapide [3] : les pièces qui sont actuellement en fonctionnement, conçues et produites au moyen d'ALM, représentent environ 20% de l'ensemble des pièces produites par le procédé ALM. Ce pourcentage est presque exclusivement réalisé dans les secteurs de l'aéronautique, de l'automobile et du biomédical. Néanmoins, un effort croissant venant du monde scientifique et industriel est mis sur le développement de la fabrication directe pour des pièces de structure [3]. Cela est dû aux quatre principales opportunités qu'offre l'ALM. Ainsi, les produits peuvent avoir une forme très complexe ou une composition multi-matériaux (*Liberté de Conception*). Egalement, l'ALM garantit l'*efficacité environnementale*, car elle n'utilise que le matériau qui constitue réellement le produit final, ce qui minimise le flux massique / énergétique. De plus, l'ALM n'a pas besoin d'outils spécifiques en fonction des différentes productions, il est donc adapté aux petites séries, éliminant ainsi le problème de stockage (*flexibilité*). Enfin, la conception libre et la flexibilité de l'ALM permettent de produire des composants



hautement personnalisés sans coûts supplémentaires (*personnalisation*).

Bien que l'ALM offre des avantages indéniables, il présente certaines limites qui entravent l'utilisation systématique de cette technologie dans le monde industriel. Ces limitations concernent à la fois le processus de fabrication lui-même et le manque de savoir-faire. Ces aspects sont d'une importance capitale pour intégrer l'ALM dans une chaîne numérique industrielle standard, c'est-à-dire de la conception préliminaire à la certification. De nouvelles questions se posent naturellement lorsqu'il s'agit d'ALM. Par exemple, quelle est la qualité de surface des structures obtenues ? Comment est-il possible de garder le contrôle des défauts de matériaux in-situ ? Comment la microstructure affecte-t-elle les propriétés mécaniques macroscopiques ? Quels sont les paramètres machine optimaux pour produire des composants au moyen de l'ALM ? Dans quelle phase de conception doit-on faire intervenir l'optimisation afin d'exploiter pleinement le potentiel de l'ALM ? Répondre à ces questions est un point clé pour combler l'écart entre les exigences fixées par les spécifications techniques et la pièce effectivement produite. Par conséquent, il semble évident que l'industrialisation du processus ALM passe par le développement de modèles mathématiques et d'algorithmes capables d'intégrer la spécificité du processus de fabrication ainsi que les phénomènes physiques connexes depuis la première étape (préliminaire) du processus de conception jusqu'à la mise en œuvre finale du procédé.

Cette thèse a été réalisée dans le cadre du projet FUTURPROD, cofinancé par la région Nouvelle-Aquitaine et par plusieurs partenaires industriels : AGB, Ariane Group, Polyshape et Stelia Aerospace. L'objectif principal du projet FUTURPROD vise à encourager le développement de la technologie ALM pour les matériaux métalliques en Nouvelle-Aquitaine, en établissant une collaboration entre des partenaires industriels et des laboratoires de recherche locaux, comme l'"Institut de Mécanique et d'Ingénierie" (I2M) de Bordeaux. Le projet a été conçu avec comme exigence l'étude de la technologie SLM (Selective Laser Melting). Ainsi, certaines tâches du projet sont vraiment spécifiques à cette technologie de fabrication additive, alors que d'autres sont vraiment générales. En particulier, trois axes de recherche ont été identifiés (chaque objectif fait l'objet d'une thèse de doctorat) :

- a) Caractériser et optimiser le processus de fabrication en SLM. Comme les technologies SLM impliquent des phénomènes complexes (fusion des métaux, gradients de température, réactivité chimique élevée des métaux avec l'oxygène, etc.), il est donc fondamental de comprendre quels paramètres principaux doivent être contrôlés pour produire des composants mécaniques fiables et de haute qualité. En effet, la puissance du laser, la vitesse du faisceau laser, la qualité de la poudre en termes de densité de grain, de géométrie et d'oxydation sont autant de paramètres importants affectant le résultat final. De plus, ce travail devrait clarifier les cinétiques d'oxydation.
- b) Phénomènes de fatigue des structures SLM. De nos jours, on s'intéresse de plus en plus à la production de pièces fonctionnelles par ALM (fabrication directe). La

fatigue étant toujours l'une des causes les plus fréquentes de défaillance des structures en fonctionnement, il est fondamental de comprendre comment le processus SLM affecte les comportements des composants mécaniques dans ce contexte. Le défi de cet axe de recherche est de mettre en place un modèle numérique capable de prendre en compte la microstructure issue du processus SLM (présence de défauts, hétérogénéité) et de simuler le mécanisme d'endommagement. Ensuite, les modèles devront permettre de prévoir correctement la limite de fatigue et la comparer à la limite de fatigue du matériau de base.

- c) Conception de structures ALM. L'enjeu pour l'ALM est double. D'une part, l'ALM permet de produire des formes et des topologies très complexes, qui sont souvent le résultat d'un processus d'optimisation dédié. D'autre part, l'augmentation de la liberté de conception rend le processus de conception plus délicat car de nouvelles exigences, contraintes spécifiques et phénomènes physiques associés à l'ALM apparaissent. Il devient essentiel d'identifier les points critiques dans le processus de conception d'une structure ainsi obtenue afin de proposer une stratégie cohérente pour surmonter ces difficultés. Cette tâche peut être résumée à travers la question suivante : est-il possible de développer une méthodologie d'optimisation adaptée capable d'intégrer correctement, depuis le début du processus de conception, les fonctionnalités de la technologie ALM ? En ce qui concerne les structures ALM, la limite entre la conception et l'optimisation est tellement intime qu'il serait préférable d'introduire le concept "Optimum Design for Additive Manufacturing" au lieu du simple "Design for Manufacturing".

Cette thèse porte sur le dernier axe de recherche du projet FUTURPROD.

## **Problèmes liés à l'intégration de l'optimisation, de la conception et de la CAO dans le cadre de l'ALM**

Les larges possibilités en termes de formes et de topologies offertes par l'ALM, ainsi que ses coûts élevés, conseillent d'utiliser cette technologie seulement pour des composants optimisés pour de hautes performances. Par conséquent, il semble naturel d'introduire la notion d'optimisation au cours du processus de conception. Cette étape d'optimisation peut être réalisée à différents niveaux, selon la nature de la structure. Classiquement, l'optimisation structurelle est divisée en trois catégories [5]. La première est l'optimisation de dimensionnement (ou paramétrique) [6] : certaines grandeurs géométriques / mécaniques sont identifiées comme des variables de conception et l'optimisation consiste à trouver les valeurs optimales de ces variables afin de minimiser / maximiser une certaine réponse. Un exemple dans ce sens est la minimisation des coûts de production (forme, soudage) d'un récipient cylindrique, coiffé aux deux extrémités par des têtes hémisphériques. L'épaisseur

de la coque, l'épaisseur de la tête, le rayon intérieur et la longueur de la partie cylindrique peuvent être considérés comme des variables de conception [7].

La deuxième famille de problèmes d'optimisation structurelle est appelée Optimisation de Forme (OF) [8] : comme son nom l'indique, les limites d'une topologie donnée peuvent évoluer pendant l'optimisation pour atteindre un objectif précis, mais la topologie de la structure reste inchangée. Ce type d'optimisation est plus difficile que l'optimisation paramétrique parce que garder le contrôle des limites et éviter les intersections, les chevauchements et les singularités est une tâche vraiment complexe. La dernière catégorie d'optimisation structurelle est l'optimisation de la topologie (OT). D'une manière générale, l'OT vise à déterminer les "propriétés" optimales définissant l'espace topologique (par exemple, la connexité et la compacité). Le problème de l'OT est très difficile à résoudre au moyen d'une approche purement géométrique. La pratique la plus courante consiste à transformer le problème initial en un problème plus simple. Ainsi, le problème d'obtention de la "topologie optimale" d'un espace donné peut être reformulé en un problème de détermination de la distribution optimale du matériau sur un espace donné [5, 9, 10]. Dans ce contexte, au moins deux phases matérielles ("solide" et "vide") sont distribuées dans un domaine prédéfini (dit de conception) pour minimiser une fonction de coût soumise à certaines contraintes (caractérisant le problème en question). Actuellement, des logiciels disponibles dans le commerce (par exemple, Altair OptiStruct [11], TOSCA [12]) permettent d'exécuter à la fois les activités d'OF et d'OT de structures. Même si les industriels peuvent efficacement effectuer l'optimisation à travers ces logiciels, l'intégration de la spécificité du processus d'ALM est loin d'être une tâche facile pour plusieurs raisons. Premièrement, l'interprétation des résultats est imprécise, car les géométries optimisées sont décrites par des éléments finis (EF) et non par des entités géométriques. La nature de ce résultat ne permet pas de retrouver de manière unique les limites de la structure optimisée dans une phase de post-traitement. Deuxièmement, les contraintes d'optimisation de nature différente (géométrique, mécanique, technologique, etc.) ne sont pas nécessairement satisfaites sur la géométrie. En effet, ces contraintes ne sont pas supportées aujourd'hui par aucun logiciel commercial, où elles ne sont pas cohérentes avec la topologie finale. Tous ces aspects rendent le processus de conception inefficace et plusieurs boucles de conception-optimisation - CAO sont nécessaires avant de converger vers une solution "optimale et fabricable" (la Fig. 1 donne une idée du processus global de conception) :

1. On considère une première conception d'une structure réalisée, que l'on nommera "design courant".
2. Une tâche d'OT est effectuée sur la conception actuelle au moyen d'un logiciel commercial standard.
3. L'optimisation fournit une conception basée sur les EF qui est loin d'être compatible avec un environnement CAO, dans lequel la phase de conception se déroule

classiquement.

4. Une phase de reconstruction est nécessaire pour transformer les pixels et les voxels en surfaces et en volumes, et pour convertir la frontière irrégulière respectivement en courbes et en surfaces.
5. Après la phase de reconstruction, les exigences techniques sont vérifiées et, généralement, plusieurs boucles de conception sont nécessaires pour obtenir une structure optimisée répondant aux exigences de l'ensemble complet.

La reconstruction CAO peut être identifiée comme la faiblesse la plus importante de la boucle de la Fig. 1. Actuellement, certains outils dédiés pour réaliser une conception compatible CAO sont disponibles. Par exemple, le module OSSmooth a été implémenté dans Altair OptiStruct. Le contour d'une structure 2D peut être exportée dans un format de fichier d'échange CAO standard, comme IGES (Initial Graphics Exchange Specification) ou STP (STandard pour les données de modèle d'échange de Produit). En 3D, les limites (qui devraient être des surfaces dans un environnement CAO) ne peuvent pas être facilement récupérées. Si un fichier IGES est utilisé, OSSmooth renvoie un ensemble d'éléments déconnectés qui ne reproduisent seulement qu'apparemment les limites des solides. Sinon, un fichier STL (Standard Tessellation Language) peut être obtenu : dans ce cas, la frontière de la structure optimisée 3D est approximée via des triangles (définis par leurs sommets et les composants normaux externes). Quoi qu'il en soit, la reconstruction de la solution finale optimale dans l'environnement CAO nécessite un nombre conséquent de décisions arbitraires prises par le concepteur et la conception résultante peut potentiellement être très éloignée de celle fournie par le logiciel d'OT [13]. Bien sûr, passer d'une conception discrétisée par EF à une géométrie compatible CAO introduit une approximation et, par conséquent, une incohérence entre les deux configurations n'est pas surprenante. En effet, le résultat direct obtenu par OT, c'est-à-dire la distribution optimale des phases de matériau sur un domaine de conception donné, n'est pas CAO compatible. De plus, étant donné que les contraintes ALM sont extrêmement spécifiques et induisent des modifications profondes et non intuitives dans la structure, elles doivent être prises en compte dès les phases de départ de la boucle de conception, c'est-à-dire au stade OT. Considérer les contraintes ALM uniquement dans une phase de post-traitement pourrait impliquer de revoir complètement la configuration optimisée, rendant inutile l'algorithme d'OT. Ce sont les raisons pour lesquelles de nos jours des logiciels largement répandus tels qu'OptiStruct ou TOSCA sont principalement utilisés pour obtenir un design pseudo-optimal qui sert de ligne directrice au concepteur afin d'améliorer en quelque sorte les performances actuelles d'un composant spécifique.

## Objectifs de la thèse

À la lumière des questions énoncées ci-dessus, cette thèse vise à fournir une méthodologie générale pour concevoir et optimiser des composants qui sont censés être produits au moyen de la technologie ALM. Dans ce contexte, les principaux objectifs suivants peuvent être identifiés :

1. La première exigence à satisfaire est la formulation mathématique et le développement d'un algorithme d'OT innovant, dont la caractéristique fondamentale est de donner une description de la topologie optimale au moyen d'entités purement géométriques et non pas des entités de type éléments finis. Le nouvel algorithme d'OT doit avoir une compatibilité CAO intrinsèque pour deux raisons principales : d'une part, la phase de reconstruction CAO sera simplifiée et, d'autre part, la configuration optimisée sera cohérente avec les contraintes d'optimisation. A cet effet, les entités B-Spline et NURBS seront naturellement testées pour constituer les fondamentaux de la nouvelle méthode d'OT présentée dans cette thèse.
2. On cherchera à formuler des contraintes d'optimisation adaptées qui devraient pouvoir prendre en compte les exigences de base de la technologie ALM. Ces contraintes devront être satisfaites sur la géométrie finale reconstruite.
3. En outre, la nouvelle formulation mathématique du problème d'OT devra garantir l'obtention des solutions optimales. Elle devra garantir la prise en compte des grandeurs mécaniques déjà prises en compte dans les logiciels d'OT standard, sans aucune restriction supplémentaire. Le choix des hypothèses liées à notre formulation doit assurer des solutions d'une qualité au moins équivalente à celle obtenue avec les logiciels du commerce.
4. La nature intrinsèquement CAO compatible de l'algorithme à développer, devra simplifier la tâche de reconstruction CAO : l'objectif final de ce travail sera de développer des techniques et des outils mathématiques appropriés qui devront être capables de réduire les tâches lors de la phase de reconstruction.

## Structure de la thèse

La structure de ce manuscrit a été conçue pour répondre clairement aux quatre objectifs caractérisant ce travail.

En ce qui concerne l'étude bibliographique, une revue de la littérature est fournie au chapitre 1, dont les principaux aspects sont essentiellement les caractéristiques générales des technologies ALM, les méthodes d'OT et la formulation de certaines contraintes ALM dans les algorithmes d'OT. Ces sujets ont été rassemblés soit par leurs caractères fondamentaux pour mieux comprendre le cadre de la thèse, soit par leurs caractères illustratifs des idées

discutées tout au long du manuscrit. Des techniques et méthodologies plus spécifiques sont proposées au début des chapitres 4 à 7 afin de fournir les principales références pour le problème en question.

Le chapitre 2 est consacré à la théorie NURBS. Les entités NURBS sont clairement définies et leurs propriétés principales sont discutées à travers un exemple significatif.

Cette thèse traite de l'optimisation en général et pas seulement des techniques d'OT, un bref aperçu des méthodes déterministes et méta-heuristiques est présenté au chapitre 3. Le chapitre 4 vise à répondre aux besoins exprimés par l'objectif 1. Notamment, les bases des surfaces et hyper-surfaces de la théorie NURBS du chapitre 2 sont convenablement combinées avec les algorithmes de gradient décrits au chapitre 3 afin de fournir un outil d'OT efficace et CAO compatible. L'efficacité de l'approche proposée est testée sur le problème bien établi de la maximisation de la rigidité soumise à une contrainte d'égalité de volume.

Le chapitre 5 est relatif à l'expression des contraintes de fabrication liées à l'ALM. Certaines contraintes de nature géométrique (par exemple le contrôle d'échelle de longueur, rayon de courbure, . . .), essentielles lors de l'utilisation de la technologie ALM, ont été formulées d'un point de vue mathématique dans le cadre de l'algorithme développé au chapitre 4.

L'objectif de robustesse de notre outil est considéré dans le chapitre 6 : l'algorithme d'OT est amélioré en travaillant sur deux autres grandeurs mécaniques, qui sont souvent nécessaires dans la conception des structures : les charges de flambement et les fréquences propres.

Le chapitre 7 traite des sujets issus de la reconstruction CAO. Une contribution significative a été fournie dans le cadre du problème d'approximation des courbes, et la généralisation à un problème d'approximation de surface est également discutée.

Enfin, les Conclusions et Perspectives de ce travail sont discutées.

La production scientifique liée à cette thèse est fournie à la fin de ce document.



# Chapter 1

## Literature Review

### 1.1 Introduction to the Literature Review

This first chapter aims at providing a state of the art about three main topics related to this Ph.D. thesis. Firstly, the ALM technology is described from a manufacturing viewpoint. The objective of this non exhaustive literature review is to highlight the complexity of the physical phenomena involved in ALM process and to emphasise that the realisation of components by means of ALM technology should necessarily take into account several aspects from the early stages of the design process. Secondly, the two most established methods used to perform TO are described: advantages and drawbacks of each one are underlined. Finally, a state of the art about some recent techniques of ALM-oriented constraints implementation in TO algorithms is provided.

### 1.2 The Additive Layer Manufacturing Technology

ALM has experienced an important spread in the last decades and several technologies have been developed. Each technology is suited for a specific application and for well-defined materials. Beyond the peculiar differences characterising the ALM technologies, the procedure that allows for obtaining a component is composed of several steps, as described here below [14].

1. **CAD Model.** The CAD model of the component to be produced is considered. It could be the result of an optimisation process or simply the best trade-off choice made by the designer in order to meet the technical specifications.
2. **Conversion to standard file format.** The CAD model boundary should be converted into a standard file format containing its discretised form. Generally, an STL file is chosen but other, more complete file formats are available (e.g. the AMF



file, which relies on the definition of curved triangles and it has been conceived for ALM applications).

3. **Transfer to ALM Machine and File Manipulation.** The standard file is transferred to the ALM machine computer. Here, the file is modified according to precise requirements such as orientation, facets “cleaning” (i.e. suppression of degenerated triangles), etc.. The eventual volume of support structures and its topology are established during this phase as well.
4. **Machine Set-up.** This task consists of assigning all the parameters characterising the ALM technology at hand. Some among them are easy to set but others have contrasting effects on the final component, as remarked later on.
5. **Build.** The actual building phase is an automatic task for the most of ALM machines nowadays.
6. **Removal and Post-processing.** Once removed from the machine, the piece should be carefully cleaned up (typically, surface treatments are necessary) and, eventually, useless support structures must be removed.

As it can be deduced from the previous list, the production of a component through ALM is not only matter of manufacturing. It requires a deep and skilled knowledge of the machine parameters and of their impact on the physical phenomena involved in the process. Nevertheless, handling CAD files and managing the correct information flow to an STL/AMF file is a fundamental task as well. Moreover, the previous scheme should be tailored for each specific ALM technology, as it can be easily deduced from the high amount of variants for this manufacturing process, discussed in the following.

### 1.2.1 A classification of ALM processes

In this brief bibliography on ALM technologies, the classification discussed in [14] is considered. The adjectives “direct” and “indirect” are used according to the definition given in [15]. In indirect technologies the bonding function is performed by a binder material, different from the bulk material. Conversely, in direct methods, the bulk material is also responsible of the binding function.

*Vat photo-polymerization processes* exploit the reactivity of photo-polymers to radiation in the ultra-violet (UV) range of wavelengths: under these conditions, photo-polymers undergo a complex suite of chemical reactions leading to solidification.

*Powder bed fusion processes* are perhaps the most suited for structural applications. Many processes belong to this category, as Selective Laser Sintering (SLS), Selective Laser Melting (SLM) and Electron Beam Melting (EBM). Regardless to the specific technology, usually an energy source is used to induce the partial or total melting of powder particles

and prescribed mechanisms deal with the uniform spreading of the powder and with the provision of new powder layers (generally a screaper). Various materials can be employed in powder bed systems, such as metals, thermoplastics and ceramics.

*Extrusion-based systems* are commonly used mainly because of the relatively low cost of the machine and because of the simplicity of the set-up. The principle consists of melting the bulk material in a suitable chamber and pushing it out through a nozzle. The bonding mechanism can be realised by means of the simple solidification of the material itself (direct method) or by means of secondary build materials (indirect method). The most common extrusion-based ALM technology is the Fused Deposition Modelling (FDM) to produce ABS (Acrylonitrile Butadiene Styrene) components.

*Sheet Lamination* processes, as LOM (Laminated Object Manufacturing), involve a layer-by-layer lamination of material sheets and some bonding mechanism is exploited (adhesives, thermal bonding, clamping, ultrasonic welding). Obviously, this method is really sensitive to the variation of strength in the direction normal to the layers.

Promising processes are those belonging to the class of *Material Jetting* and *Binder Jetting*. In the former case, the bulk material is directly dispensed from a print-head, whilst in the latter case the bulk material is a powder substrate and a suitable binder is printed on it.

*Directed Energy Deposition* (DED) machines allow for producing parts by melting materials as they are being deposited: to this aim, the print-head is constituted of a heat source (e.g. a laser beam or an electron beam) and a material powder feeder. This technology is mostly employed for metal parts, but polymers and ceramics can be used as well. When compared to powder bed fusion processes, DED machines can achieve directional solidified and single crystal structures, multi-material components and fully dense parts. However, the accuracy is not very high and surface roughness is not satisfying for engineering applications. Moreover, DED process are not able to produce small-scale features.

*Direct Write* (DW) technologies are well suited for very small scale components. Typical applications of this emerging additive manufacturing process involve the thermal and strain sensors production and antenna fabrication.

In this manuscript, the attention is focused on the metal powder bed technology, since it is the most suited for engineering automotive and aerospace applications. All metal materials that can be welded are prone to ALM [15]. Among the powder bed processes, the most widespread are undoubtedly SLS, SLM and EBM. In SLS, the metal powder is firstly spread in a layer and then selectively scanned by a laser to build each layer. SLS can work either as direct method by partially melting the metal particles or as indirect method by melting a low-melting-point binder (phenolic polymers or low-melting-point metals). Anyway, a post-processing phase concerning the removal of polymer binder, such as thermal sintering, liquid-metal infiltration or hot isostatic pressing (HIP), is necessary. The most important issue related to SLS is the impossibility to reach fully dense parts:

consequently, porosity heavily affects mechanical properties of SLS structures. SLM and EBM can partially overcome this issue because of the improved energy flow through the heat source, which is a laser beam for SLM and an electron beam for EBM. In any case, the bonding mechanism is the melting metal itself. Nearly fully dense metal parts exhibiting as good mechanical properties as the bulk material can be produced.

### 1.2.2 The Selective Laser Melting Technology

Here below, a more detailed description of the SLM technology is provided because it has been chosen as a reference ALM technology in the context of the FUTURPROD project. In the framework of SLM, researchers have investigated the mechanical properties of components obtained by using different materials: stainless steels [16–18], several aluminium alloys (often silicon-based alloys) [19,20], titanium alloys [21], hastelloys [22] and Inconel [23]. It is evident that the physical-chemical phenomena involved in the SLM process have a strong effect on the properties of the manufactured part. The parameters controlling the process have been identified by Spears and Gold [24]: 50 parameters can be identified and only 12 can be controlled by the user. The parameters are classified as laser and scanning parameters (e.g. the laser power supply, scan speed and path), powder material properties (bulk density, material absorptivity), powder bed properties (layer thickness, powder bed temperature, thermal conductivity) and build environment parameters (inert atmosphere gasses). Therefore, the produced component is always the result of the interaction of all these parameters. Being impossible to provide a complete picture of the whole set of process parameters and the related influence on the resulting part, it has been decided to take a pragmatic approach and to summarise the most important defects appearing in SLM structures, with a particular attention on their causes.

- **Porosities.** The presence of porosities cannot be totally eliminated in SLM and they constitute internal defects, prone for crack onset. The porosity is primarily affected by the laser parameters (scanning speed, laser power, lasing strategy) and powder parameters (preheating temperature, the layer thickness). Too low laser energies do not allow for a complete melting of the powder particles, which can be trapped in the surrounding melting pot, whilst a too high laser energy could lead to trap some gas portions, resulting in spherical cavities. The quality of the powder itself is important (impurities should be eliminated, especially for those technology that make use of recycle), together with the absorption rate of the laser energy.
- **Hot Cracking Defects.** This phenomenon is typical of welded aluminium alloys and it depends on the severe thermal gradient undergone by the heat affected zone. Limiting the temperature gradient would be useful to solve this problem. Improvements can be achieved thanks to the composition of the alloy at hand: typically, low melting temperature elements lead to a decrease of the hot cracking

phenomenon. It should be pointed out that hot cracking is an evident consequence of the thermal history of the component but other possible issues are related to the strong thermal gradient. For instance, the built part can exhibit deformations or residual stresses [25]. Thermal effects can involve very local phenomena (such as the material delamination when a zone is scanned by the laser) or global phenomena: the latter is the case of variation of processing temperature. Generally, cross-sections of the part are scanned with paths parallel to each other. In order to achieve a compromise between material wetting conditions and uniform temperatures, Kruth *et al.* [16] have demonstrated that better results can be achieved in terms of distortion by scanning a steel plate dividing it in small rectangular sectors rather than following parallel paths.

- **Anisotropy.** It is not difficult to guess that ALM structures, being produced according to a preferential build direction, exhibit an overall anisotropic behaviour. Indeed, this fact is not crucial in the case of SLM structure, whilst it is much more critical and evident in extrusion-based technologies. In any case, reducing thermal gradients and suitably orienting the component in the machine chamber are good practices allowing for an almost isotropic behaviour.
- **Surface Roughness.** The surface quality is far to be appropriate for a structure produced by SLM and supposed to satisfy durability requirements. Generally speaking, finishing and polishing post-processing phases are necessary. The porosity is one of the main causes of the poor quality of surfaces. Poor surface quality is also due to laser parameters, as the power and the scanning rate. Moreover, the orientation of the piece in the machine has a strong impact on the surface quality as well: the *scale effect* occurs for all those surfaces which are not perpendicular or parallel to the basement plane [26,27]. Finally, surface roughness is also affected by the *balling* phenomenon, occurring when the molten material does not wet the underlying substrate because of the surface tension, which tends to deform the liquid into a sphere. This results in a rough and bead-shaped surface, hindering a smooth layer deposition and decreasing the density of the produced part.

Beyond the parameters directly related to the machine, there are other parameters that depend on the designer's sensibility and experience, such as the support material creation and the piece orientation with respect to both the scraper and the machine basement. The effect of the orientation of the component in the machine chamber has already been mentioned for the surface roughness and for the occurrence of anisotropy. However, the orientation has a strong impact also on other features characterising the process and the quality of the final component. If the part has a too broad surface perpendicular to the scraper sweep direction, this one could hit the component during production and could make it move (*shock*), being the powder bed absolutely not stiff enough to prevent this movement. The orientation of the part is still basic to properly cover it with powder. Its

compactness is about 60%, so a material withdrawal occurs during melting. Then, an amount of excess powder is required but, if the highest dimension is disposed along the scraper sweep direction, the powder exhaustion will be really probable. Accordingly, a good practice would be tilting the mechanical component with respect to the sweeping direction of the scraper. Manufacturing time is strongly affected by the part positioning in the machine as well: it depends on the laser power and on the number of layers to be deposited. It is worthy to dispose the part in such a way that its highest dimension lies on the basement plane.

Further technological requirements to be investigated are those related to the presence of support structures. When one zone of the mechanical part overcomes a *critical angle* of inclination (refer to Fig. 1.1), a support structure is necessary in order to avoid material breakdown or instability. Cloots *et al.* [28] distinguish *supported overhangs* and *unsupported overhangs*.

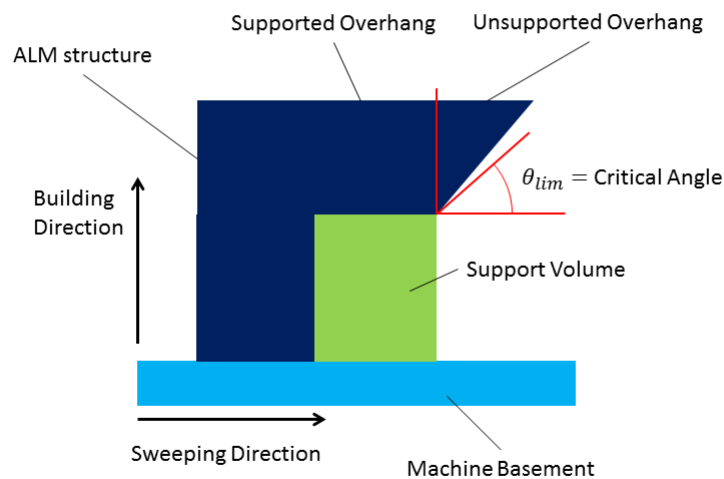


Figure 1.1 – Critical Angle for supported overhangs.

Besides the effective support role, these structures are really useful in order to improve the quality of the final part:

- supports can be used as interfaces between the base-plate and the part; in this way, efforts are better distributed inside the structure, the part can be removed by the

machine without being marked and, finally, it can be lifted up on the basement (so the irregular first layers are not included in the piece manufacturing);

- the centre of gravity of the mechanical part changes during ALM production, so a support structure could be basic to stabilise the component;
- concerning residual stresses induced by the gradient of temperature, a suitable placing of supports during manufacturing can stiff some critical regions in order to meet geometrical constraints after the cooling phase; therefore, a thermal treatment for stress relaxation is applied, holding the support in the initial position and, under these conditions, the support structure can be removed since no high residual stresses are in the mechanical part.

Often a lattice structure is chosen to create supports [29]: selective connection points permit the efforts transmission, heat dissipation and an easier separation in the post-processing phase. Moreover, the grid, no-walled, structure assures a suitable stiffness without wasting powder which is not embedded. Because of necessary post-processing, functional surfaces have not to be put in contact with the support but they must be perpendicular or parallel to the scraper sweeping direction. However the optimisation of the support structure (in terms of minimum support volume, best heat conductivity) is an active research topic [28,30,31] and it should be included in the design flow of an ALM structure.

## 1.3 Topology Optimisation Methods

In the last three decades, Topology Optimisation (TO) has gained an increasing degree of interest in both academic and industrial fields. The aim of TO for structural applications is to distribute one or more material phases in a prescribed domain in order to satisfy the requirements for the problem at hand. Usually, the design problem is formulated as a Constrained Non-Linear Programming Problem (CNLPP), wherein a given cost (or objective) function must be minimised and, meanwhile, the full set of optimisation constraints has to be met.

In this literature review, only density-based methods and the Level Set Method (LSM) are discussed. Other TO methodologies, such as the Evolutionary Structural Optimisation (ESO) [32] and the Phase-Field method [33] are not considered here.

### 1.3.1 Density-Based Methods

Classically, first TO methods were based on a Finite Elements (FE) description of the design domain [34]. The basic idea consists of defining a continuous fictitious density function (or pseudo-density function) varying between zero and one on the computation

domain. The pseudo-density function is evaluated at the centroid of each element of a predefined mesh and provides information about the topology: “void” and “solid” phases are associated to the lower and upper bounds of the density function, i.e. zero and one, respectively. Meaningless “gray” elements (related to intermediate values of the density function) are allowed but penalised during optimisation in order to achieve a “clear” solid-void final design. Thus, mechanical properties of each element are computed (and penalised) according to the local density value. Several penalisation schemes have been developed for evaluating mechanical properties, e.g. Solid Isotropic Material with Penalisation (SIMP) or Rational Approximation of Material Properties (RAMP) [5].

The mathematical statement of the classic SIMP method is briefly described in the following. Without loss of generality, the discussion focuses on 3D TO problems. More precisely, the mathematical formulation is here limited, for the sake of clarity, to the problem of minimising the compliance of a structure, subject to an equality constraint on the volume.

Let  $D \subset \mathbb{R}^3$  be a compact subset in the 3D Euclidean space, in which a Cartesian orthogonal frame  $O(x_1, x_2, x_3)$  is defined:

$$D = \{(x_1, x_2, x_3) \in \mathbb{R}^3 | x_1 \in [0, a_1], x_2 \in [0, a_2], x_3 \in [0, a_3]\}, \quad (1.1)$$

where  $a_1$ ,  $a_2$  and  $a_3$  are three reference lengths of the domain (related to the problem at hand), defined along  $x_1$ ,  $x_2$  and  $x_3$  axes, respectively. The distribution of a given isotropic “heterogeneous material” (i.e. the definition of void and material zones) in the design domain  $D$  is sought in order to minimise the virtual work of external loads applied to the structure and, meanwhile, to meet a suitable volume equality constraint. Let  $\Omega \subseteq D$  be the *material domain*. In the SIMP approach,  $\Omega$  is determined by means of a fictitious density function  $\rho(x_1, x_2, x_3) \in [0, 1]$  defined over the whole design domain  $D$ . Such a density field is related to the material distribution:  $\rho(x_1, x_2, x_3) = 0$  means absence of material, whilst  $\rho(x_1, x_2, x_3) = 1$  implies completely dense bulk material (refer to Fig. 1.2).

The density field affects the stiffness tensor  $E_{ijkl}(x_1, x_2, x_3)$ , which is variable over the domain  $D$ , according to the following formula:

$$E_{ijkl}(\rho(x_1, x_2, x_3)) = \rho(x_1, x_2, x_3)^\alpha E_{ijkl}^0, \quad i, j, k, l = 1, 2, 3, \quad (1.2)$$

where  $E_{ijkl}^0$  is the stiffness tensor of the bulk isotropic material and  $\alpha > 1$  is a suitable parameter that aims at penalising all the meaningless densities between 0 and 1. The power law of Eq. (1.2) is the most widely used in the SIMP framework when the problem of minimum compliance with an equality volume constraint is faced. The choice of other penalisation schemes should be carefully assessed according to the particular problem at hand [5].

Considering the FE formulation of the equilibrium problem for a linear elastic static analysis in the global reference frame, let  $\{d\}$  be the vector of the overall displacements

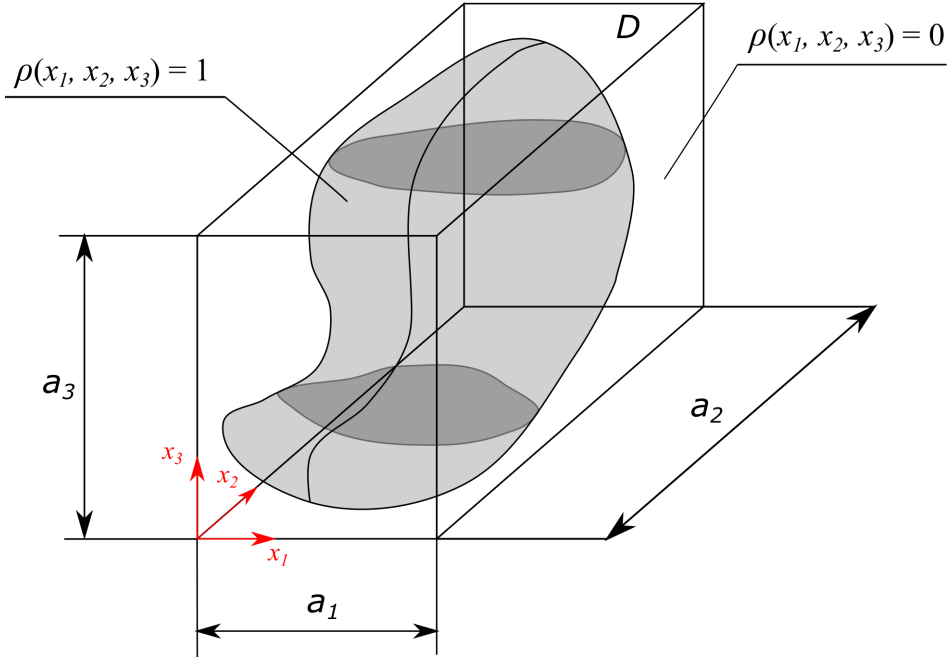


Figure 1.2 – Representation of a general 3D structure through the SIMP fictitious density field.

and rotations (referred to as *degrees of freedom* - DOFs) and  $\{f\}$  the vector of applied generalised nodal forces. The relationship between  $\{d\}$  and  $\{f\}$  is

$$[K] \{d\} = \{f\}, \quad (1.3)$$

where  $[K]$  is the global stiffness matrix of the structure. Accordingly, the compliance of the structure is computed as

$$c = \{d\}^T [K] \{d\}. \quad (1.4)$$

Taking into account Eq. (1.2),  $[K]$  can be expressed as

$$[K] = \sum_{e=1}^{N_e} \rho_e^\alpha [K_e^0], \quad (1.5)$$

where  $\rho_e$  is the fictitious density computed at the centroid of the generic mesh element  $e$ ,  $N_e$  the total number of elements, whilst  $[K_e^0]$  is the non-penalised element stiffness matrix expanded over the full set of DOFs of the structure.

The problem of minimising the compliance of a 3D structure subject to a constraint



on the overall volume can be stated as follows:

$$\begin{aligned} & \min_{\rho_e} c(\rho_e), \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} [\mathbf{K}]\{\mathbf{d}\} = \{\mathbf{f}\}, \\ \frac{V(\rho_e)}{V_{ref}} = \frac{\sum_{e=1}^{N_e} \rho_e V_e}{V_{ref}} = \gamma, \\ \rho_{min} \leq \rho_e \leq 1, \quad e = 1, \dots, N_e. \end{array} \right. \end{aligned} \quad (1.6)$$

In Eq. (1.6),  $V_{ref}$  is a reference volume,  $V(\rho_e)$  is the volume of the material domain  $\Omega$ , while  $\gamma$  is the fixed volume fraction;  $V_e$  is the volume of element  $e$  and  $\rho_{min}$  represents the lower bound, imposed to the density field in order to prevent any singularity for the solution of the equilibrium problem. The design variables of the TO problem in the classic SIMP framework are the fictitious densities defined at the centroid of each element: therefore the overall number of design variables is equal to  $N_e$ .

It is well-established that the classic SIMP problem (1.6) is ill-posed [5]. As matter of fact, the topologies proposed by the SIMP method change when a different mesh size is used; this is due to the fact that the greater the number of holes in the structure (by keeping constant its volume) the better the structure performance is. The limit of this process is a structural variation at the microscopic scale, that cannot be caught by an isotropic material description. It results in a numerical instability where a higher number of holes appears if a finer mesh is used. Several techniques can be adopted to overcome this problem [5]. The first one is trivially a *perimeter control*: a constraint on the maximum value of the perimeter results, *de facto*, in a limitation in the number of holes. Alternatively, a constraint on the spatial gradient of the density function plays a similar role. However, the most popular choice made by TO algorithms developers is to eliminate the mesh dependency by means of a filtering operation. Indeed, when a density value is associated to a mesh element, *a priori* there is no inter-dependence among contiguous elements. This fact leads to the well-known checker-board layout of material distribution, which is an artificial high stiffness solution (but totally meaningless from a physical viewpoint) [5]. Therefore, filters can be used in order to establish a fictitious dependence among adjacent elements. For instance, distance-based filters are very often employed. The use of filters in TO allows for overcoming both the mesh dependence issue and the occurrence of checker-board patterns. It can be shown that all the previous techniques result, at the end, in establishing a minimum length scale in the design, as discussed in [35] and in Chapter 5 of this manuscript.

It should be highlighted that TO problems are non-convex in general. Therefore, if a gradient-based algorithm is chosen in order to update the design variables at each iteration (i.e. to provide a new density distribution), probably the retrieved result will be a local

optimum and not a global optimum. Nevertheless, global strategies allowing for a better exploration of the design domain would fail and they are strongly not advisable because of the high number of design variables, usually characterising TO problems [36]. A trade-off solution to this problem could be employing so-called *continuation methods*, wherein an artificial convex or quasi-convex form of the problem is solved at the beginning and it is progressively changed into the original non-convex problem. In this case, gradient-based algorithms are well suited to perform the solution search.

The advantage of using mathematical programming in TO instead of zero-order algorithms (e.g. meta-heuristics) is the possibility to exploit the information provided by the derivatives of objective/constraints functions with respect to the whole set of design variables for the solution search. In the case of problem (1.6), the derivatives of the compliance and of the volume are reported here below for the sake of completeness (see [34] for more details). The partial derivative of the compliance reads

$$\frac{\partial c}{\partial \rho_e} = -\alpha \rho_e^{\alpha-1} \{\mathbf{d}\}^T [\mathbf{K}_e^0] \{\mathbf{d}\}, \quad e = 1, \dots, N_e. \quad (1.7)$$

If the compliance of the single mesh element is introduced as follows,

$$c_e = \rho_e^\alpha \{\mathbf{d}\}^T [\mathbf{K}_e^0] \{\mathbf{d}\}, \quad (1.8)$$

Eq. (1.7) can be simplified as

$$\frac{\partial c}{\partial \rho_e} = -\frac{\alpha}{\rho_e} c_e, \quad e = 1, \dots, N_e. \quad (1.9)$$

The partial derivative of the volume reads

$$\frac{\partial V}{\partial \rho_e} = V_e, \quad e = 1, \dots, N_e. \quad (1.10)$$

The derivatives computation is usually referred as *sensitivity analysis* in TO. An overview of the SIMP algorithm is shown in Fig. 1.3.

An interesting alternative to the more rigorous (and time consuming) mathematical programming strategy is the Optimality Condition [5]. The idea is to exploit the necessary conditions of optimality (as discussed in Chapter 3) in order to develop an efficient (although heuristic) updating scheme for element densities. These approaches have been widely tested in literature [37, 38]. Of course, the evident shortcoming of the optimality condition approach is that it is not general and an *ad hoc* rule should be provided for whatever constraint or objective function. In the case of compliance minimisation with a volume equality constraint, the criterion is easy to develop because the optimum design should be fully stressed or nearly fully stressed. However, developing heuristic criteria is not straightforward for other mechanical quantities or for manufacturing-oriented constraints. It is noteworthy that one of the most common algorithm used in literature is

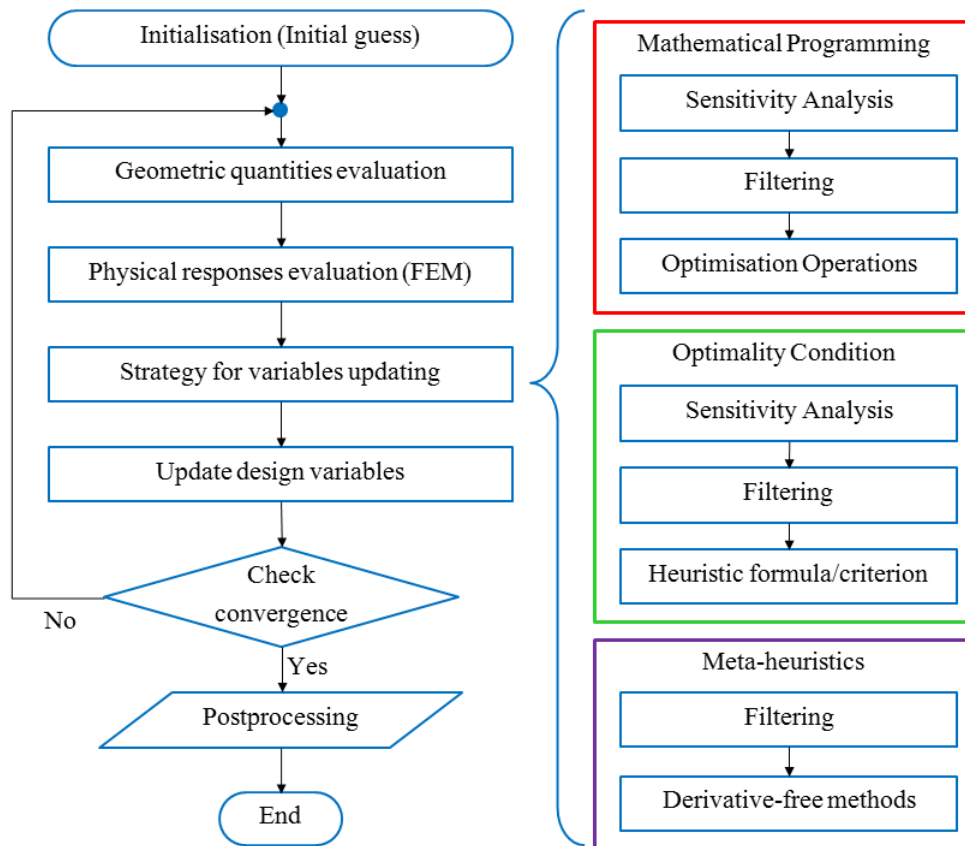


Figure 1.3 – Classic SIMP algorithm.

the well-known *Method of Moving Asymptotes* (MMA) [39, 40], but it represents just one among the possible optimisation algorithms suited for TO problems resolution. A more thorough discussion on mathematical programming algorithms can be found in Chapter 3 of this manuscript.

A summary of advantages and drawbacks of the SIMP method (that can be assumed as the reference density-based method) is provided here below.

### Advantages

- The SIMP method is relatively easy to understand and it can be implemented in very compact scripts [41].
- The robustness of density based methods has been widely tested in literature. The considerable amount of bibliographic references provided in the following of this manuscript proves that the SIMP method is extremely efficient and versatile for

several objective/constraint function implementation.

- The reliability of the SIMP method has made possible to embed this algorithm in well-established software packages (Altair OptiStruct<sup>®</sup> [11], TOSCA [12]), currently constituting the reference for TO in the industrial field.

#### Drawbacks

- Since the pioneering works on TO, different strategies were proposed during the years in order to overcome classic TO drawbacks, such as checker-board effect and mesh dependence. Projection methods have been used in [10] and their robustness has been investigated in [42]. In these methods, the design variables are the values of the fictitious density function at the mesh nodes. The element densities are obtained by means of a suitable Heaviside step function-based projection. Such a projection can be chosen in such a way to impose a minimum length scale or other kind of constraints (e.g. maximum length scale [43]). However, the well-posedness of the problem is subject to an artificial choice (the filter size, the filter type, the projection method, etc.).
- In spite of their relative simplicity, the SIMP method provides a FE-based description of the final geometry and suitable postprocessing must be forecast in order to obtain a smooth CAD-compatible design. This shortcoming involves both the mathematical nature and the lack of effective tools to interpret the final design in terms of CAD entities.
- There is no possibility to keep control of the boundary of the current topology during optimisation. This issue is related to the previous point and the common cause is the lack of a purely geometric entity describing the topology.

#### 1.3.2 The Level-Set Method

More recently, an alternative TO method, known as *Level-Set Method* (LSM) has been developed and successfully applied to solve TO problems [9, 44, 45]. Indeed, the LSM originates by the exigency of solving mathematical and physical problems involving boundary evolutions (evolution of interfaces in multi-phase flows, image segmentation, etc.) [1]. Several LSM-based algorithms have been investigated in the TO framework [46]. The basic idea is to describe the topology of the component through a suitable *Level-Set Function* (LSF). The sign of the LSF can be conventionally associated to material or void zones, while the zero value represents the boundary of the optimised structure:

$$\begin{cases} \psi(\mathbf{x}) > 0 \Leftrightarrow \mathbf{x} \in \Omega \setminus \partial\Omega, \\ \psi(\mathbf{x}) = 0 \Leftrightarrow \mathbf{x} \in \partial\Omega, \\ \psi(\mathbf{x}) < 0 \Leftrightarrow \mathbf{x} \in D \setminus (\Omega \cup \partial\Omega). \end{cases} \quad (1.11)$$

In Eq. (1.11),  $\psi$  is the LSF defined on the computational domain  $D$ ,  $\Omega$  is the material domain, whilst  $\partial\Omega$  is the boundary. An example of LSF for 2D structures, which is represented by means of a suitable 3D surface, is shown in Fig. 1.4. Analogously, the LSF is a 4D hyper-surface for the optimisation of 3D components and its 0-level set ( $\partial\Omega$  according to this notation) is constituted of 3D surfaces. Therefore, the LSM offers an implicit definition of the topology boundary, since the LSF is parametrised on the computational domain. Whatever transformation of a 2D or 3D material domain  $\Omega$  (eventually implying modifications of the connectivity of the domain) can be caught thanks to a LSF that can be represented in a domain whose dimension is increased by one with respect to that of the computational domain  $D$ .

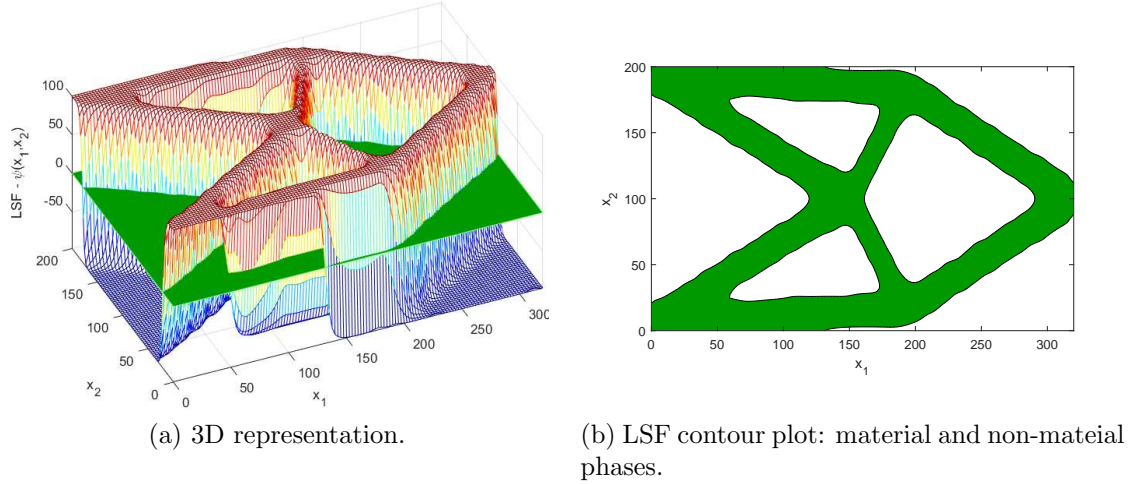


Figure 1.4 – Example of Level Set Function.

Although several versions of the LSM have been proposed in literature [9, 46, 47], they share the same general logical procedure, whose flowchart is shown in Fig. 1.5. The most important feature is that the description of the topology is now related to the LSF and no more to the underlying mesh (decoupling effect): the FE model and the related mesh are necessary in order to perform the physical responses evaluation. Each step of the LSM is briefly described in the following.

### 1- LSF Parametrisation

The choice of the LSF directly affects the nature of the TO problem in terms of

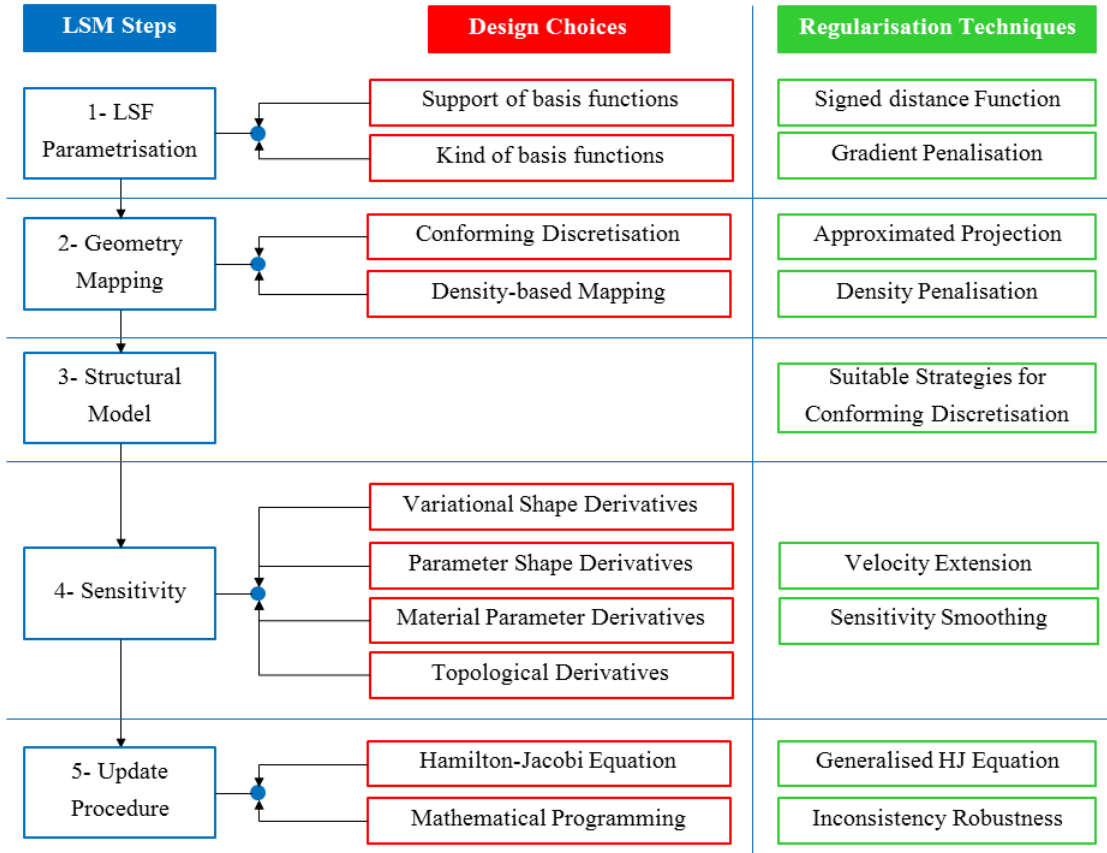


Figure 1.5 – Overview of the LSM.

non-linearity and convexity. Typical LSFs are in the form

$$\psi(\mathbf{x}, \mathbf{s}) = \sum_i^n \psi_i(\mathbf{x}, s_i) = \sum_i^n B_i(\mathbf{x})s_i \quad (1.12)$$

where  $\mathbf{s}$  is the design variables array and  $B_i(\mathbf{x})$  are the basis functions, which depends only on the position in the domain  $D$ . It is pointed out that the array  $\mathbf{s}$  collects some of (eventually all) the parameters defining the LSF. These parameters can intervene in the LSF definition in different ways: for instance, in Eq. (1.12) the simplest case is shown, wherein design variables just constitute the weights of basis functions. The LSF can be chosen among different sets of functions [46]. The *support size* of the basis functions  $B_i(\mathbf{x})$  is a first criterion to classify the LSF: the support is the region of the domain where the basis function is not null. Increasing the support size, i.e. passing from local basis functions to midrange functions or global functions means increasing the sensitivity of the responses to the LSF parameters: therefore, the design evolves significantly iteration by iteration but this process requires additional memory. Concerning the *kind* of basis

functions, often Radial Basis Functions (RBFs) are utilised because of their versatility and simplicity [47–49]. RBFs are typically defined on the overall design domain. However their mathematical statement can be changed in order to obtain compactly supported RBFs [48]. Other possible choices are FE basis functions or Fourier series functions. FE basis functions appear to be very promising, since they allow for taking into account the structural computation and the LSF evolution in the same manner. However, as remarked in [9], the steepness of the LSF gradient at the boundary is crucial to numerical stability and RBFs functions guarantee for a smoother gradient definition.

## 2- Geometry Mapping

Geometry Mapping is the operation that permits to transfer the information provided by the LSF to the underlying FE model for physical responses evaluation. This task can be performed by means of two methods. The first one is referred as *conforming discretisation* and it can be interpreted as a Lagrangian description of the boundary. Here, only the material phase  $\Omega$  is considered and the boundary is clearly identified. This method is well suited for those problems requiring the evaluation of local quantities (such as stress), geometry-dependent loads or particular non-linearities (e.g. contacts). Of course, the price to pay is the non-structured mesh generation at each iteration (a procedure that often fails) and the possibility of inconsistencies like material zones surrounded by the void phase. The second method is the *density-based mapping*. This is an Eulerian approach: a regular mesh is set up on the whole design domain  $D$  and the mechanical properties of each element can be affected by defining a pseudo-density function  $\rho$ , like in the SIMP method. A common practice is adopting the “black-and-white” description

$$\rho(\psi(\mathbf{x})) = \varepsilon + (1 - \varepsilon)H(\psi(\mathbf{x})), \quad (1.13)$$

where  $H(\psi)$  is the *Heaviside* function and  $\varepsilon \ll 1$  is in infinitesimal quantity. Then, the  $\rho$  distribution affects the material stiffness tensor  $E_{ijkl}$  according to the classic *Erstsz material model*, i.e.

$$E_{ijkl} = \rho(\psi(\mathbf{x}))E_{ijkl}^0. \quad (1.14)$$

This second approach is more robust and generally converges in less iterations but it suffers from the same drawbacks of the SIMP method.

## 3- Structural Model

Once the information concerning the topology has been transferred to the mechanical model, physical responses can be evaluated. The FE analysis can be carried out. There are not, *a priori*, restrictions about the responses or the non-linearity characterising the problem at hand. Of course, the user should be able to evaluate the sensitivity of objective/constraint functions with respect to the set of design variables. This task is far from being trivial in general cases, as explained in the following step.

#### 4- Sensitivity

The *sensitivity* in the LSM must be interpreted as the sensitivity of the responses to an elementary variation of the boundary in the normal direction. The most general method is referred as *variational sensitivity analysis* and it is strongly related to a particular updating procedure, that is the *Hamilton-Jacobi* (HJ) partial differential equation (PDE) [1], also known as transport equation, i.e.

$$\frac{\partial \psi(\mathbf{x}, t)}{\partial t} + V_n(\mathbf{x}, t) |\nabla \psi(\mathbf{x}, t)| = 0, \quad \mathbf{x} \in D. \quad (1.15)$$

In Eq. (1.15),  $t$  is a fictitious time parameter and  $V_n(\mathbf{x}, t)$  is the normal (to the boundary) component of a velocity field. One of the most important and challenging aspect of the LSM is relating the velocity field appearing in the HJ equation to the sensitivity of the objective/constraint functions to the boundary evolution. As stated in [9], it can be shown that an effective manner of managing the HJ equation is to identify the velocity field  $V_n(\mathbf{x}, t)$  as the shape derivative of the Lagrangian functional of the problem at hand (accounting for the objective function and the optimisation constraints). An exhaustive discussion of these mathematical concepts can be found in [8]. It is noteworthy that the shape derivative does not allow for the nucleation of new holes in the structure (pre-existing holes can only merge): this issue can be solved by evaluating the so called *topological derivative* [8], that is the sensitivity of the mechanical responses w. r. t. the nucleation of an infinitesimal hole in the structure. Such an approach has been successfully implemented in [50].

Other methods to evaluate the sensitivity have been developed mostly for those updating procedures which are based on mathematical programming (*parameter shape derivatives*). Let  $r(\mathbf{x})$  be a general response. It can be shown that the infinitesimal variation of the boundary can be related through the LSF gradient to the infinitesimal variation of the design variables ( $\delta s_i$ ), i.e. a subset of the LSF parameters. Therefore, in a mathematical programming method, wherein the LSF shape is tuned through these parameters, directly computing the gradient of responses w. r. t. design variables ( $\partial r(\mathbf{x})/\partial s_i$ ) is advantageous. Sometime this computation could become cumbersome but, if a density-based mapping strategy has been previously chosen, the derivative can be easily computed by the following simple chain rule:

$$\frac{\partial r(\mathbf{x})}{\partial s_i} = \sum_{e=1}^{N_e} \frac{\partial r(\mathbf{x})}{\partial \rho_e} \frac{\partial \rho_e}{\partial s_i}, \quad \forall i, \quad i = 1, \dots, n. \quad (1.16)$$

The last form is referred as *material parameter derivatives*.

#### 5- Updating Procedure



As previously anticipated, the two fundamental methods to update the current topology are the *solution of the HJ equation* and the evaluation of the descent direction (through suitable function derivatives) in the framework of *mathematical programming algorithms*. Solving the HJ equation is not trivial: several specificities should be considered. Firstly, the velocity field, related to the shape derivative and physically meaningful only at the boundary, should be extended at least on a strip around the boundary for regularisation reasons. Secondly, the “viscous term”  $\beta\Delta\psi(\mathbf{x}, t)$  should be added to the right hand side of Eq. (1.15). This term is supposed to regularise the solution and to avoid singularities, which otherwise could occur during the boundary evolution (an example is illustrated in Fig. 1.6). The smoothness of the solution is strongly dependent on the parameter  $\beta$  and its choice should be carefully assessed because it is problem-dependent.

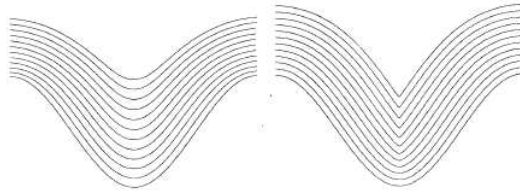


Figure 1.6 – Boundary evolution with (left) or without (right) “viscous term” [1].

Moreover, if the topological derivative has to be computed to enable the hole nucleation mechanism, it should be considered that the HJ equation must be modified by adding a reaction (sink/source) term. The HJ equation enhanced with the viscous term and the reaction term is referred as *generalised HJ equation*. Then, the solution of the HJ can be carried out thanks to efficient upwind schemes [1].

In an explicit marching scheme, the time interval  $\Delta t$  and the grid spacing  $h$  should be related to the normal velocity field  $V_n$  through the Courant-Friedrichs-Lewy (CFL) condition, i.e.

$$\Delta t \max(V_n) \leq h. \quad (1.17)$$

The fixed grid spacing must be small enough to allow a suitable resolution. Hence, the time step is limited and a significant amount of iterations is required to observe some topological change.

As far as mathematical programming is concerned, authors have solved TO problems in the LSM framework by using Sequential Quadratic Programming, MMA, Quadratic Approximation. In most of those cases, mathematical programming is coupled with simple LSFs, as, for instance, RBFs [48]. The advantage of using mathematical programming is essentially providing a rigorous and general method for constraints handling. Moreover, as discussed in the followings, the several regularisation phases required by the LSM could

result in some kind of inconsistency. Mathematical Programming tools are robust enough to cope with this numerical issues.

## 6- Regularisation

The LSM is a numerically tricky method, so several regularisation techniques should be taken into account. *Ad hoc* regularisation schemes take action at the different steps during the running of the LSM-based TO algorithm. It has been observed that the LSF gradient can strongly vary along the boundary and this hampers the convergence. In order to overcome this issue, the LSF is re-initialised to a signed-distance function, whose gradient norm is constant and equal to 1. In performing this operation, that is beneficial to the solution of the HJ equation, the zero-level contour is not exactly maintained and this fact could lead to inconsistencies in the optimisation process [1]. Other regularisation schemes consist of penalising the gradient of the LSF [46]. As for the SIMP method, a perimeter constraint can stabilise the solution and, most of all, guarantee for the well-posedness of the optimisation problem, making the solution robust with respect to numerical artifacts. A similar result can be obtained thanks to the *Tichonov regularization* [51]: the gradient of the LSF can be related to an energetic term and can be summed to the objective function. However, it has been shown that results depend on the introduced diffusion coefficient [50].

Some regularisation techniques can be adopted also in the geometry mapping phase. Namely, performing the projection from the LSF to the pseudo-density field through an approximated Heaviside function can increase the region of influence of the boundary and, therefore, the sensitivity is smoother and convergence improves. Meanwhile, it is advised to perform a similar penalisation to that of Eq. (1.2) to avoid large regions with intermediate densities (a similar result can be achieved by making use of the previously discussed signed distance reinitialisation).

As far as sensitivity and the updating steps are concerned, the regularisation techniques have been already discussed because their are intrinsic to the solution of the HJ equation. It must be underlined that, if the updating procedure is the HJ equation, an explicit scheme to smooth the sensitivity can be adopted.

In the light of the aforementioned aspects about the LSM, the following advantages and drawbacks can be identified.

### Advantages

- The LSF gives a clear implicit geometric representation of the boundary of the domain. The topology description is therefore uncoupled to the underlying mesh of the mechanical model. Theoretically, the mesh size can be tailored to obtain a satisfactory evaluation of the responses whilst the topology description depends only on the LSF parameters.

- The grey-scale effect (mesh elements with intermediate values of pseudo-density) characterising the SIMP method is overcome here.
- Very complex, even non-linear, problem as contacts or design-dependent loads can be solved by using a conforming mapping approach in natural way. A special attention should be paid to the mesh generation.

### Drawbacks

- The well-posedness of the TO problem is intrinsically violated when using the LSM. In fact, the topology depends only on the zero-level contour of the LSF, therefore its definition (and that of its gradient) is not unique. Moreover, since the final design strongly depends on the initial guess of the LSF function (when the hole nucleation mechanism is not considered), the convexity of the problem is not guaranteed even for the simple compliance minimisation problem.
- When the updating procedure relies on the solution of the HJ equation, the velocity field should be related to the sensitivity of the Lagrangian form of the problem at hand, considering the objective function together with the optimisation constraints with the respective Lagrange multiplier. Often, the handling of the Lagrange multipliers is not mathematically sound and they are simply set to constant values [9], or updated through heuristic rules [52]. The choice of the fixed Lagrange multiplier affects the results. When automatic updating schemes are used, the computational burden sensitively increases.
- The LSM requires a large amount of regularisation techniques. Often, such regularisation schemes have an intrinsically numeric/mathematical nature and are not related to any physical requirement.

## 1.4 Implementation of Manufacturing Constraints in Topology Optimisation

The increasing development of ALM has pushed several authors and research activities to include ALM-oriented constraints in TO algorithms, in order to bridge design and manufacturing requirements and to obtain an optimised, actually manufacturable component. The first manufacturing constraints considered in TO literature are referred as minimum/maximum length scale constraints. Albeit their nature is intrinsically geometric and not technological, the considered geometric quantities can be related to technological requirements. For instance, imposing a minimum length scale in TO in both SIMP [35] and LSM [45] frameworks aims at assuring a minimum size of structural elements and,

therefore, to eliminate in the design phase all those topological features that are not manufacturable because smaller than the technological minimum allowable size. In this bibliographic study, this kind of constraints are not considered; Chapter 5 will explicitly deal with the implementation of minimum/maximum length scale in TO, together with other geometric quantities taken into account in the framework of the proposed methodology.

The concept of “manufacturing tolerant” or “robust” design TO was introduced by Sigmund in [53]: here, the robustness must be interpreted as consistence of the length scale of the optimised configurations with respect to manufacturing imprecisions. To this aim, the idea of *eroded* and *dilated* designs (pseudo-density distributions) is introduced and the imposed constraints are checked on the three designs. Finally, this results in imposing a minimum length scale in TO. More ALM-oriented exigencies, as the volume of support structures, have been addressed in [54]: however, this work does not provide concrete implementation of an ALM constraint, but just advises deas for future developments. Although the ALM technology is subject to several constraints of different nature (as discussed in section 1.2), the volume of the support structure (and strategies for its minimisation) is the most studied ALM constraint in bibliography [55].

In [56], a post-processing method to produce self-supporting structures is presented. The idea is to act on the smoothed boundary of the optimised structure in order to identify the non-self-supporting zones and to insert a suitable support volume, selecting a suitable build direction for the ALM process. This method presents two main drawbacks: firstly, a certain increment of volume (eventually small) of material is necessary; secondly, the consistence of the imposed constraints is not assured after the proposed post-treatment. Other approaches in literature have integrated the overhang constraint directly in TO algorithms and not in the post-processing phase. Gaynor and Guest [57, 58] suggest to use an auxiliary variable field, defined at the mesh nodes, in order to achieve the element densities as an Heaviside projection: such a projection is performed by using a smart filter technique that allows material only if supported (a suitable building direction and a critical overhang angle must be specified before the optimisation task). However, the authors admit that this constraint is inefficient from a computational view point because it mimics the ALM building strategy and the sensitivity analysis is evaluated “layer-by-layer”. Moreover, the non-linear projection generates issues for the convergence and it is not simple to tune the optimisation parameters of the employed algorithm (namely, the MMA) to balance efficiency, solution convergence and solution quality. A similar technique has been proposed in [59]: elements densities are allowed to take a value close to 1 only if there is a sufficient amount of material in the support region. This method has been developed only for a regular mesh. An extension to non-mapped mesh with several industrial applications have been provided by authors in [60]. In spite of the overhang angle for support material is managed, such a scheme does not allow for a steep density distribution at the end of the optimisation and wide intermediate density regions

appear. This fact could lead to issues in the results interpretation/post-processing phase. Moreover, a common shortcoming of the strategies proposed in [57–60] is that the support volume is not considered explicitly and its mass/volume is not computed. Indeed, these works aim at developing support-free structures. However, this could be very limiting and the algorithm could propose configurations, corresponding to local optima solutions, which are really far from the ALM-unconstrained optimum solution.

A more challenging research topic would be providing a trade-off criterion among performances and support requirements [61]. In [62], for instance, the support structure sensitivity is studied. Here, the support structure is considered and an evaluation of its volume is provided. Consistently, an explicit constraint enhances the TO problem of minimum compliance with imposed volume fraction in the framework of a particular LSM, based on a Pareto optimal front. An alternative method aiming at providing a sort of measure of the support volume has been developed by Qian in the framework of density-based algorithms [63]: both the undercut control and the maximum overhang angle control, which are quantities intrinsically defined on the boundary of the material domain, are considered as integral volume on the computational domain. It turns out that the constraints formulated according to Qian’s approach implicitly impose a limit on the perimeter of the boundary subjected to undercut issues or to the perimeter of the boundary subjected to maximum overhang issues. In spite of this physical meaning, it is not easy for the designer to set the optimisation parameters, which are strongly problem dependent. Moreover, since the constraints definition induces a limitation on the density gradient, a further constraint is necessary in order to avoid trivial solutions with wide flat regions of the density function (anti-greyiness constraint).

Beyond the support volume, other ALM-related aspects and potentials are considered in TO formulation: for instance, in the work of Clausen [61], multi-material applications, multi-scale approaches and coated structures are investigated. Authors in [64] try to bridge a hybrid topological/lattice structure design with manufacturing requirements.

## 1.5 Conclusions on the Literature Review

This bibliographic study has highlighted some main features, which are listed here below.

- Each ALM technology has specific requirements. Nevertheless, general design methodologies should be developed in order to keep control of common features of ALM. The contrasting effects of the whole set of parameters that the designer can tune in producing a component by ALM clearly show that it is impossible to optimise production time, costs and mechanical properties at the same time. Thus, design choices have to be made referring to a specific requirement or to a compromise strategy.
- Two main algorithms families can be identified in literature for TO. The first one

is composed of density-based algorithms, the second one is constituted of LSM algorithms. Both of them exhibit advantages and drawbacks. It is not possible to state which family provides better results and the choice should be made by considering the problem at hand.

- It has been shown that considering ALM constraints (like the support volume) in a post-processing phase of a TO algorithm is not a sound procedure because important inconsistencies between imposed design constraints and actual produced component can be observed. However, the implementation of realistic and meaningful constraints in TO algorithms is not simple and suitable numerical strategies must be developed in order to overcome the strong non-linearity related to these requirements. Research is ongoing in order to provide suitable TO algorithms capable of managing ALM constraints and, therefore, to consistently bridge ALM, optimisation and technological specifications.



# Chapitre 1

## Revue de la littérature

### 1.1 Introduction de la revue de la littérature

Ce premier chapitre vise à fournir un état de l'art sur trois sujets principaux liés à cette thèse. Premièrement, la technologie ALM est décrite du point de vue du processus de fabrication. L'objectif est de mettre en évidence la complexité des phénomènes physiques présents dans le processus ALM et de souligner la nécessité de la prise en compte de plusieurs aspects, dès les premières étapes du processus de conception. En second lieu, les deux méthodes les plus répandues pour réaliser l'Optimisation Topologique sont décrites : les avantages et les inconvénients de chacune des deux sont présentés. Enfin, un état de l'art est fourni, concernant certaines techniques récentes de prise en compte de contraintes orientées ALM dans les algorithmes d'Optimisation Topologique.

### 1.2 La technologie de fabrication additive par couche (Additive Layer Manufacturing)

L'ALM a connu un développement important au cours des dernières décennies et plusieurs technologies sont apparues. Au-delà des particularités caractérisant les différentes technologies ALM, une procédure commune permettant d'obtenir des pièces a été proposée par un auteur [14] de la littérature comme expliqué ci-dessous en déroulant les étapes suivantes :

1. **Modèle CAO.** Le modèle CAO de la pièce à produire est réalisé. Ce modèle est le résultat soit d'un processus d'optimisation soit simplement le meilleur choix du concepteur.
2. **Conversion au format de fichier standard.** Les surfaces limites du modèle



CAO doivent être converties en un format de fichier standard contenant une forme discrétisée. Généralement, un fichier STL est choisi.

3. **Transfert vers la machine ALM et manipulation des fichiers.** Le fichier standard est transféré à l'ordinateur contrôlant la machine ALM afin de gérer les incohérences géométriques, et pour finalement mettre en place les structures du support.
4. **Configuration de la machine.** Cette tâche consiste à affecter les bonnes valeurs aux paramètres caractéristiques de la technologie ALM.
5. **Réalisation.** La phase de construction proprement dite est une tâche automatique pour la plupart des machines ALM de nos jours.
6. **Retrait et post-traitement.** Une fois retirée de la machine, la pièce doit être soigneusement nettoyée.

Le schéma précédent doit être adapté pour chaque technologie ALM spécifique. En effet, il est possible à partir des nombreuses références de la bibliographie d'identifier des spécificités liées aux différents processus de fabrication, comme discuté dans ce qui suit.

### 1.2.1 Classification des processus ALM

Dans cette revue bibliographique sur les technologies ALM, la classification de [14] est considérée.

Les procédés de photo polymérisation exploitent la propriété de réactivité des photopolymères aux rayonnements dans la gamme des longueurs d'onde ultraviolettes (UV).

De nombreux procédés appartiennent à la catégorie des procédés de fusion par lit de poudre, tels que le frittage sélectif par laser (Selective Laser Sintering - SLS), la fusion sélective par laser (Selective Laser Melting - SLM) et la fusion par faisceau d'électrons (Electron Beam Melting - EBM). Classiquement, une source d'énergie est utilisée pour générer la fusion partielle ou totale des particules de poudre. Des mécanismes adaptés supplémentaires sont nécessaires pour un étalement uniforme de la poudre et l'apport de nouvelles couches de poudre.

Des systèmes à extrusion sont également couramment utilisés en raison du coût d'achat relativement bas de la machine et de la simplicité de la configuration de cette machine. Le principe consiste à faire fondre le matériau dans une chambre appropriée et à pousser le matériau à travers une buse.

Les procédés d'impression 3D par encollage de feuilles, (Laminated Object Manufacturing - LOM), impliquent une stratification couche par couche de feuilles de matériau et un mécanisme de collage adapté (adhésifs, liaison thermique, serrage, soudage par ultrasons). Cette méthode est très sensible à la variation des propriétés des interactions et en particulier dans la direction normale aux couches.

Des processus prometteurs appartenant à la classe de Material Jetting et de Binder Jetting font également leur apparition. Dans le premier cas, le matériau est directement déposé à partir d'une tête d'impression, tandis que dans le second cas, le matériau est un substrat de poudre et un liant approprié est imprimé sur celui-ci.

Les machines DED (Directed Energy Deposition) permettent de réaliser des pièces en fondant des matériaux lors de leur dépôt : à cet effet, la tête d'impression est constituée d'une source de chaleur (par exemple un faisceau laser ou un faisceau d'électrons) et d'un système d'alimentation de la poudre. Par rapport aux procédés de fusion par lit de poudre, la précision des machines DED n'est pas très élevée et la rugosité de surface n'est pas très satisfaisante pour les applications d'ingénierie.

Les technologies Direct Write (DW) sont-elles dédiées aux composants de petite échelle. Les applications typiques de ce processus de fabrication additive sont la production de capteurs thermiques ou de déformation et la fabrication d'antennes.

Dans ce manuscrit, l'attention sera portée sur la technologie par lit de poudre métallique et, en particulier, sur la technologie SLM. Le SLM étant le plus adapté aux applications automobile et aérospatiale, il a été choisi comme technologie ALM de référence dans le cadre du projet FUTURPROD.

### 1.2.2 La technologie de fusion sélective par laser (SLM)

Dans le cadre de la SLM, les chercheurs ont étudié les propriétés mécaniques des pièces obtenues en utilisant différents matériaux : aciers inoxydables [16–18], plusieurs alliages d'aluminium (souvent des alliages à base de silicium) [19,20], alliages de titane [21], hastelloys [22] et Inconel [23]. Les paramètres contrôlant le processus ont été mis en évidence par Spears et Gold [24] : 50 paramètres ont été identifiés et seulement 12 pouvant être contrôlés par l'utilisateur. Les défauts les plus importants apparaissant dans les structures SLM sont liés à certains de ces paramètres comme suit.

- **Porosités.** Les porosités ne peuvent pas être totalement éliminées en SLM et elles constituent des défauts internes, sujets à l'apparition de fissures. La porosité est principalement affectée par la vitesse de balayage, la puissance du laser, la stratégie de lasage et les paramètres de la poudre (température de préchauffage, épaisseur de la couche). La qualité de la poudre elle-même est un paramètre important, ainsi que le taux d'absorption de l'énergie laser.
- **Défauts de fissuration à chaud.** Ce phénomène dépend de la sévérité du gradient thermique subi par la zone affectée par la chaleur produite par le laser. La limitation du gradient de température est prépondérante pour résoudre ce problème, ainsi que l'ajout d'éléments d'alliage à basse température de fusion dans la composition chimique de la pièce. Les déformations ou contraintes résiduelles sont également liées au gradient thermique [25]. Afin de parvenir à un compromis entre les conditions

de changement de phase du matériau et des gradients de température admissibles, Kruth et al. [16] ont démontré que des parties moins déformées peuvent être obtenues en lasant une plaque d'acier en la divisant en petits secteurs rectangulaires plutôt qu'en suivant des chemins parallèles.

- **Anisotropie.** Il n'est pas difficile de deviner que les structures ALM, produites selon une direction de construction préférentielle, présentent un comportement anisotrope global. Cette propriété n'est pas cruciale dans le cas de structures réalisées par SLM. Dans tous les cas, la réduction des gradients thermiques et l'orientation appropriée de la pièce dans la chambre de la machine sont des bonnes pratiques rendant le comportement presque isotrope.
- **Rugosité de surface.** La qualité de surface obtenue pour les structures produites par SLM est loin d'être appropriée et satisfaisante face aux exigences de durabilité. Des phases de post-traitement de finition et de polissage s'avèrent nécessaires. La porosité est l'une des principales causes de la mauvaise qualité des surfaces. De plus, l'orientation de la pièce dans la machine a aussi un fort impact sur la qualité de la surface [26,27]. Enfin, la rugosité de surface est également affectée par le phénomène de *balling*, empêchant un dépôt de couche lisse et diminuant la densité de la pièce produite.

D'autres paramètres sont d'une importance primordiale et dépendent de l'expérience du concepteur. L'orientation de la pièce par rapport au raclor et à la position dans la machine a un fort impact sur plusieurs caractéristiques du processus et la qualité de la pièce finale. La pièce doit être inclinée par rapport à la direction de balayage du raclor afin d'éviter les chocs (le raclor heurte la pièce) et les mauvaises conditions de recouvrement de la poudre. En outre, étant donné que le temps de fabrication dépend de la puissance du laser et du nombre de couches à réaliser, il convient de disposer la pièce de manière à ce que sa dimension la plus élevée se situe sur la base de la machine. Toutefois il faut noter que la quantité de matière lasée doit également être prise en compte pour l'évaluation du temps de fabrication. Les supports sont intimement liés à l'orientation des pièces dans la machine. Ils sont nécessaires lorsqu'une zone du composant surmonte d'un angle d'inclinaison critique la partie basse de la pièce (voir Fig. 1.1) afin d'éviter un défaut matière ou une instabilité [28].

Ces supports sont vraiment utiles pour améliorer la qualité de la pièce finale :

- les supports peuvent être utilisés comme interfaces entre la plaque de base et la pièce ; de cette façon, les efforts sont mieux répartis à l'intérieur de la structure et les premières couches irrégulières ne sont pas incluses dans la pièce elle-même ;
- un support est essentiel pour stabiliser la pièce ;

- un placement approprié des supports pendant la fabrication peut rigidifier certaines régions critiques afin de respecter les contraintes géométriques après la phase de refroidissement.

L’optimisation de la structure du support (en termes de volume de support minimum, meilleure conductivité thermique) est un sujet de recherche actif [28,30,31] et est souvent réalisée dans le cadre de structures lattice [29].

## 1.3 Méthodes d’Optimisation Topologique

Au cours des trois dernières décennies, l’optimisation topologique (OT) a suscité un intérêt croissant dans les domaines académiques et industriels. Le but de l’Optimisation Topologique pour les applications structurelles est de répartir une ou plusieurs phases de matériaux dans un domaine prédéfini afin de satisfaire aux exigences du problème en question. Habituellement, le problème de conception est formulé comme un problème de programmation non linéaire convexe (CNLPP), dans lequel une fonction coût (ou objectif) donnée doit être minimisée, en satisfaisant complètement l’ensemble des contraintes d’optimisation.

### 1.3.1 Méthodes basées sur la densité

Classiquement, les premières méthodes d’Optimisation Topologique étaient basées sur une description aux éléments finis (FE) du domaine de conception [34]. La discussion qui suit portera sur la méthode dite du matériau solide isotrope avec pénalisation (SIMP) [5]. Plus précisément, la formulation mathématique est ici limitée, par souci de clarté, au problème de la minimisation de la compliance d’une structure, soumise à une contrainte d’égalité sur le volume.

Soit  $D \subset \mathbb{R}^3$  un sous-ensemble compact défini comme dans l’éq. (1.1) dans l’espace euclidien 3D :  $O(x_1, x_2, x_3)$  est un repère orthogonal cartésien, tandis que  $a_1$ ,  $a_2$  et  $a_3$  sont trois longueurs de référence du domaine (liées au problème en question), définies le long des axes  $x_1$ ,  $x_2$  et  $x_3$ , respectivement. La distribution d’un “matériau hétérogène” isotrope donné (i.e. la définition de zones vides et avec présence de matière) dans le domaine de conception  $D$  est recherchée afin de minimiser le travail virtuel des charges externes appliquées à la structure et répondant à une contrainte d’égalité de volume appropriée. Soit  $\Omega \subseteq D$  le domaine *matériel*. Dans l’approche SIMP,  $\Omega$  est déterminé au moyen d’une fonction de densité fictive  $\rho(x_1, x_2, x_3) \in [0, 1]$  définie sur l’ensemble du domaine de conception  $D$ . Un tel champ de densité est lié à la distribution du matériau :  $\rho(x_1, x_2, x_3) = 0$  signifie l’absence de matériau, alors que  $\rho(x_1, x_2, x_3) = 1$  implique la présence de matériau comme montré à la fig. 1.2.

Le champ de densité affecte le tenseur de rigidité  $E_{ijkl}(x_1, x_2, x_3)$  selon l’éq. (1.2), où  $E_{ijkl}^0$  est le tenseur de rigidité du matériau isotrope et  $\alpha > 1$  est un paramètre approprié

qui vise à pénaliser toutes les densités [5]. En considérant la formulation éléments finis du problème dans le cas d'un équilibre statique et élastique dans le référentiel global, soit  $\{d\}$  le vecteur des degrés de liberté (degrees of freedom - DOFs) et  $\{f\}$  le vecteur des forces nodales généralisées appliquées. La relation entre  $\{d\}$  et  $\{f\}$  est fournie par l'éq. (1.3), où  $[K]$  est la matrice de rigidité globale de la structure. En conséquence, la compliance de la structure est calculée comme détaillé dans l'éq. (1.4). En prenant en compte l'éq. (1.2),  $[K]$  peut être exprimé par l'éq. (1.5), où  $\rho_e$  est la densité fictive calculée au centroïde de l'élément de maillage  $e$ ,  $N_e$  le nombre total d'éléments, tandis que  $[K_e]$  est la matrice de rigidité de l'élément non pénalisé étendue sur l'ensemble des DOFs de la structure. Le problème de minimisation de la compliance d'une structure 3D soumise à une contrainte sur le volume global est énoncé dans l'éq. (1.6).

Dans l'éq. (1.6),  $V_{ref}$  est un volume de référence,  $V(\rho_e)$  est le volume du domaine matériel  $\Omega$ , alors que  $\gamma$  est la fraction volumique fixe;  $V_e$  est le volume de l'élément  $e$  et  $\rho_{min}$  représente la limite inférieure, imposée au champ de densité afin d'éviter toute singularité pour la solution du problème d'équilibre statique. Les variables de conception du problème d'OT dans le cadre de la méthode SIMP classique sont les densités fictives définies au centroïde de chaque élément : donc le nombre global de variables de conception est égal à  $N_e$ .

Il est bien établi que le problème classique de la méthode SIMP (1.6) est mal posé et plusieurs techniques peuvent être adoptées pour résoudre ce problème [5]. En outre, il convient de souligner que les problèmes d'OT sont non convexes en général. Par conséquent, si un algorithme basé sur le gradient est retenu pour la mise à jour des variables de conception à chaque itération (à savoir pour fournir une nouvelle distribution de densité), le résultat récupéré sera probablement un optimum local et non un optimum global. Néanmoins, la stratégie globale permet une meilleure exploration des variables de conception mais échoue régulièrement et est fortement déconseillé en raison du grand nombre de variables de conception, caractérisant généralement les problèmes d'OT [36]. L'avantage de l'utilisation de la programmation numérique (par exemple, Méthode des *Moving Asymptotes* - MMA [39, 40]) par opposition aux algorithmes d'ordre zéro (par exemple méta-heuristiques) est la possibilité d'exploiter les informations fournies par les dérivées des fonctions objectifs/contraintes par rapport à l'ensemble des variables de conception pour la recherche de solutions. Dans le cas spécifique du problème (1.6), les dérivées de la compliance et du volume sont calculées comme suit (voir [34] pour plus de détails). La dérivée partielle de la compliance est écrite selon l'éq. (1.7). Si la compliance d'un élément utilisé pour le maillage est introduite (voir l'éq. (1.8)), alors l'éq. (1.7) peut être simplifié en éq. (1.9). La dérivée partielle du volume est fournie par l'éq. (1.10). Le calcul ses dérivées est généralement appelé *analyse de sensibilité* dans l'OT. Une vue d'ensemble de l'algorithme SIMP est détaillée sur la fig. 1.3.

Une alternative intéressante à la stratégie de programmation numérique plus rigoureuse (et plus longue) est la méthode basée sur la condition d'optimalité [5]. L'idée est

d'exploiter les conditions nécessaires d'optimalité (comme discuté dans le chapitre 3) afin de développer un schéma de mise à jour efficace (bien qu'heuristique) pour la densité des éléments. Ces approches ont été largement testées dans la littérature [37, 38]. Bien sûr, le défaut évident des méthodes basées sur la condition d'optimalité est qu'elle n'est pas générale et qu'une règle *ad hoc* doit être fournie pour toute contrainte ou fonction objectif.

Un résumé des avantages et des inconvénients de la méthode SIMP (que l'on peut considérer comme la méthode de densité de référence) est fourni ci-dessous.

### Avantages

- La méthode SIMP est relativement facile à comprendre et peut être implémentée dans des scripts très compacts [41].
- La quantité considérable de références bibliographiques fournies dans la suite de ce manuscrit prouve que la méthode SIMP est extrêmement efficace et polyvalente pour plusieurs types de fonctions objectifs/contraintes.
- La méthode SIMP est intégrée dans des progiciels bien connus (Altair OptiStruct cite Optistruct, TOSCA cite Tosca), constituant actuellement la référence de l'Optimisation Topologique dans le domaine industriel.

### Désavantages

- Depuis les premiers travaux sur l'OT, différentes stratégies ont été proposées au cours des dernières années afin de surmonter les inconvénients typiques de la méthode SIMP, tels que l'effet damier et la dépendance de la dimension du maillage [10, 42, 43]. Cependant, une bonne formulation mathématique du problème est sujette à des choix artificiels en termes de paramètres d'optimisation.
- En dépit de sa relative simplicité, la méthode SIMP fournit une description de la géométrie finale à base d'éléments finis et un post-traitement approprié doit être prévu afin d'obtenir une conception compatible avec la CAO.
- Il n'y a pas la possibilité de garder le contrôle sur la frontière de la topologie en cours d'optimisation, en raison de l'absence d'une entité purement géométrique décrivant la topologie.

### 1.3.2 La Méthode *Level-Set*

Plus récemment, une méthode alternative, connue sous le nom de Level-Set Method (LSM), a été développée et appliquée avec succès pour résoudre les problèmes d'OT [9,

44, 45]. En effet, la LSM trouve son origine dans l'exigence de résolution de problèmes mathématiques et physiques impliquant des évolutions de la frontière (évolution d'interfaces dans des écoulements multiphasiques, segmentation d'images, etc.) [1]. Plusieurs algorithmes basés sur la LSM ont été étudiés dans le cadre de l'OT [46]. L'idée de base est de décrire la topologie du composant via une *Level-Set Function* (LSF) appropriée. Le signe de la LSF peut être classiquement associé à des zones matérielles ou vides, tandis que la valeur zéro représente la limite de la structure optimisée (voir éq. (1.11)). Dans l'éq. (1.11),  $\psi$  est la LSF définie sur le domaine de calcul  $D$ ,  $\Omega$  est le domaine matériel, tandis que  $\partial\Omega$  est la limite du domaine d'étude. Un exemple de LSF pour les structures 2D, est représenté au moyen d'une surface 3D appropriée (fig. 1.4). De manière analogue, la LSF est une hyper-surface 4D pour l'optimisation des pièces 3D et son ensemble de 0-level ( $\partial\Omega$  selon cette notation) est constitué de surfaces 3D. Par conséquent, la LSM offre une définition implicite de la limite de topologie, puisque la LSF est paramétrée sur le domaine de calcul. Bien que plusieurs versions de la LSM aient été proposées dans la littérature [9, 46, 47], elles partagent toutes la même procédure logique générale, dont l'organigramme est représenté sur la fig. 1.5. La caractéristique la plus importante est que la description de la topologie est liée à la LSF et non plus au maillage sous-jacent : un modèle éléments finis (et le maillage associé) est nécessaire pour effectuer l'évaluation des réponses physiques. Chaque étape de la LSM est brièvement décrite dans ce qui suit.

## 1- Paramétrage de la LSF

Le choix de la LSF affecte directement la nature du problème d'OT en termes de non-linéarité et de convexité. La LSF peut être choisie parmi différents ensembles de fonctions [46]. La taille du support de la LSF est un premier critère de classification : ce domaine se caractérise par la région du domaine où la fonction de base n'est pas nulle. En ce qui concerne le type de fonctions de base, les fonctions de type *radial basis function* (RBF) sont souvent utilisées en raison de leur polyvalence et de leur simplicité [47–49]. D'autres choix possibles sont les fonctions de base classiques utilisées en éléments finis ou les fonctions de la série de Fourier.

## 2- *Geometry Mapping*

Geometry Mapping est l'opération qui permet de transférer les informations fournies par la LSF au modèle éléments finis associé pour l'évaluation des réponses physiques. Cette tâche peut être effectuée au moyen de deux méthodes. La première est appelée *discrétisation conforme* et peut être interprétée comme une description lagrangienne de la frontière : seule la phase matérielle  $\Omega$  est considérée à travers un maillage non structuré. Cette méthode est bien adaptée à l'évaluation des quantités locales, mais l'apparition d'incohérences, telles que des zones matérielles entourées par une phase de vide, ou la dégénérescence des éléments finis, est très problématique. La deuxième méthode est la

*cartographie* basée sur la densité. C'est une approche eulérienne : un maillage régulier est mis en place sur l'ensemble du domaine de conception  $D$  et les propriétés mécaniques de chaque élément peuvent être instanciées en définissant une fonction de pseudo-densité  $\rho$ . Cette deuxième approche est plus robuste et converge généralement en moins d'itérations mais elle souffre des mêmes inconvénients que la méthode SIMP.

### 3- Modèle Structurel

Une fois que les informations concernant la topologie ont été transférées au modèle mécanique, les réponses physiques peuvent être évaluées. L'analyse éléments finis peut être effectuée.

### 4- Sensibilité

La *sensibilité* dans la LSM doit être interprétée comme la sensibilité des réponses à une variation élémentaire de la frontière dans la direction normale. La méthode la plus générale est appelée *analyse de sensibilité variationnelle* [8] et elle est fortement liée à une procédure de mise à jour particulière, à savoir l'équation différentielle aux dérivées partielles de Hamilton-Jacobi (HJ) [1] décrite à l'éq. (1.15). Dans l'éq. (1.15),  $t$  est un paramètre temporel fictif et  $V_n(\mathbf{x}, t)$  est la composante normale (à la frontière) d'un champ de vitesse. Comme indiqué dans [9], on peut montrer qu'une manière efficace de gérer l'équation HJ est d'identifier le champ de vitesse  $V_n(\mathbf{x}, t)$  comme la dérivée de forme de la fonction lagrangienne du problème (prise en compte de la fonction objectif et des contraintes d'optimisation). Il convient de noter que la dérivée de forme ne permet pas la création de nouveaux trous dans la structure (les trous préexistants ne peuvent que fusionner) : ce problème peut être résolu en évaluant le dérivée topologique [8, 50]. D'autres méthodes d'évaluation de la sensibilité ont été développées principalement pour des procédures de mise à jour basées sur la programmation numérique (dérivées de paramètres de forme, dérivées de paramètres matériels).

### 5- Procédure de mise à jour

Comme précédemment évoqué, les deux méthodes fondamentales pour mettre à jour la topologie sont la résolution de l'équation d'HJ et l'évaluation de la direction de descente (à travers des dérivées de fonction appropriées) dans le cadre d'algorithmes de programmation numérique. La résolution de l'équation d'HJ n'est pas triviale : plusieurs spécificités doivent être étudiées (extension appropriée du champ de vitesse, adjonction de termes pour régulariser la frontière, termes éventuels puits/source pour activer le mécanisme de nucléation des trous, etc). De plus, tous les paramètres liés à ces spécificités affectent le résultat final. Dans un schéma de résolution explicite, l'intervalle de temps  $\Delta t$  et l'espacement de grille  $h$  devraient être liés au champ de vitesse normal  $V_n$  par la condition de Courant-Friedrichs-Lewy (CFL), soit  $\Delta t \max(V_n) \leq h$ . Par conséquent, le pas de temps est



donc limité et une quantité importante d'itérations est nécessaire pour observer certains changements topologiques. En ce qui concerne la programmation numérique, les auteurs ont résolu des problèmes d'OT dans le cadre de la LSM en utilisant la programmation quadratique séquentielle, la MMA, et l'approximation quadratique. Dans la plupart de ces cas, la programmation numérique est couplée à des LSF simples, comme par exemple RBFs [48]. L'avantage de l'utilisation de la programmation numérique est essentiellement de fournir une méthode rigoureuse et générale pour la gestion des contraintes.

## 6- Régularisation

La LSM est une méthode numériquement difficile, donc plusieurs techniques de régularisation doivent être prises en compte. Les schémas de régularisation *ad hoc* agissent aux différentes étapes de l'exécution de l'algorithme d'OT lié à la LSM. Il a été observé que la LSF associée au gradient peut fortement varier le long de la frontière, ce qui gêne la convergence. Afin de surmonter ce problème, la LSF est réinitialisée grâce à une fonction distance signée, dont la norme du gradient est constante et égale à 1 [1]. D'autres schémas de régularisation consistent à pénaliser le gradient de la méthode LSF [46] en utilisant, par exemple, la régularisation de Tichonov [51] : le gradient de la méthode LSF peut être lié à un terme énergétique et peut être sommé à la fonction objectif. Cependant, il a été montré que les résultats dépendent du coefficient de diffusion introduit [50]. Certaines techniques de régularisation peuvent également être adoptées dans la phase de cartographie de la géométrie. A savoir, l'exécution de la projection de la LSF sur le champ de pseudo-densité à travers une fonction Heaviside approximée peut augmenter la région d'influence de la frontière et, par conséquent, la sensibilité et la convergence s'améliorent.

A la vue des aspects susmentionnés concernant la LSM, les avantages et inconvénients suivants peuvent être identifiés.

### Avantages

- La LSF donne une représentation géométrique implicite claire de la frontière du domaine matériel. La description de la topologie est donc découplée du maillage sous-jacent associé au modèle mécanique.
- L'effet d'échelle de gris (éléments du maillage avec des valeurs intermédiaires de pseudo-densité) caractérisant la méthode SIMP est dépassé ici.
- Les problèmes très complexes, voire non linéaires, les contacts ou les chargements dépendants des formes de la pièce peuvent être résolus en utilisant une approche de cartographie conforme.

### Désavantages

- La topologie dépend uniquement du contour lié au niveau zéro de la LSF et sa définition n'est, donc, pas unique. De plus, puisque la solution finale dépend fortement de la topologie initiale de la LSF (lorsque le mécanisme de nucléation des trous n'est pas considéré), la convexité du problème n'est pas garantie même pour un simple problème de minimisation de la compliance.
- Lorsque la procédure de mise à jour repose sur la solution de l'équation d'HJ, le champ de vitesse doit être lié à la dérivée de la forme lagrangienne du problème, en considérant la fonction objectif avec des contraintes d'optimisation associées à leur multiplicateur de Lagrange respectif. Souvent, le traitement des multiplicateurs de Lagrange n'est mathématiquement pas rigoureux et ils sont simplement mis à des valeurs constantes [9], ou ils sont mis à jour par des règles heuristiques [52]. Le choix du multiplicateur de Lagrange fixe affecte les résultats.
- La LSM nécessite une grande quantité de techniques de régularisation. Souvent, ces techniques de régularisation ont un caractère intrinsèquement numérique et mathématique qui n'est pas lié à des exigences physiques.

## 1.4 Implémentation des contraintes de l'ALM dans l'OT

Le développement croissant de l'ALM a poussé plusieurs auteurs et travaux de recherche à inclure des contraintes propres à l'ALM dans les algorithmes d'OT. Ceci est réalisé afin de satisfaire les exigences de conception et de fabrication et d'obtenir une pièce optimisée.

Le concept de conception "tolérante à la fabrication" ou "robuste" a été introduit par Sigmund dans [53]. Bien que la technologie de type ALM soit soumise à plusieurs contraintes de nature différente (comme discuté dans la section 1.2), le volume des supports (et les stratégies de minimisation associées) est la contrainte ALM la plus étudiée dans la bibliographie [55]. Le volume des supports est le principal problème abordé dans [54] : cependant, ce travail ne fournit pas une solution d'implémentation concrète et ne fournit que des idées pour de futurs développements. Dans [56], une méthode de post-traitement pour produire des structures autoportantes est présentée. L'idée est d'agir sur la frontière lissée de la structure optimisée afin d'identifier les zones non autoportantes et d'insérer un volume de support approprié, en sélectionnant une direction de fabrication opportune pour le processus ALM. Cette méthode ne garantit pas la cohérence des contraintes imposées. D'autres approches dans la littérature ont intégré une contrainte de surplomb directement dans les algorithmes d'OT et non pas dans la phase de post-traitement. Gaynor et Guest [57, 58] suggèrent d'utiliser un champ variable auxiliaire, défini au niveau des nœuds du maillage, afin de déterminer des densités d'éléments par une projection de Heaviside : une telle projection est effectuée en utilisant une technique

intelligente de filtre qui ne permet la présence de matériau que s'il est supporté. Cependant, les auteurs admettent que cette contrainte est inefficace du point de vue d'une approche numérique et il n'est pas simple d'ajuster les paramètres d'optimisation de l'algorithme utilisé (à savoir le MMA) pour assurer l'efficacité, la convergence de la solution et la qualité de la solution. Une technique similaire a été proposée dans [59] : les densités des éléments sont autorisées à prendre des valeurs proches de 1 seulement s'il y a suffisamment de matière dans la région voisine du support. Cette méthode a été développée uniquement pour un maillage régulier. Une extension à un maillage non structuré avec plusieurs applications industrielles a été fournie par les auteurs dans [60]. Malgré le fait que l'angle de surplomb pour le matériau de support est géré, un tel schéma ne permet pas une distribution de densité efficace à la fin de l'optimisation et de larges régions de densité intermédiaire apparaissent. Ce phénomène peut entraîner des problèmes dans la phase d'interprétation et post-traitement des résultats. De plus, un défaut commun des stratégies proposées dans [57–60] est que le volume du support n'est pas considéré explicitement et que sa masse et son volume ne sont pas calculés. En effet, ces travaux visent à développer des structures sans support. Cependant, cela peut être très limitatif et l'algorithme peut être amené à proposer des configurations correspondant à des solutions optimales locales, qui sont vraiment éloignées de la solution optimale obtenue sans contrainte liée à l'ALM.

Un sujet de recherche plus complexe est de fournir un critère de compromis entre les performances et les exigences liées au support [61]. Dans [62], par exemple, la sensibilité à la structure des supports est étudiée. Ici, la structure du support est considérée et une évaluation de son volume est fournie. De manière cohérente, une contrainte explicite enrichit le problème de la minimisation de la compliance avec la fraction volumique imposée dans le cadre d'une LSM particulière, basée sur un front de Pareto optimal. Une méthode alternative vise à fournir une sorte de mesure du volume du support a été développée par Qian dans le cadre d'algorithmes de densité [63] : le contrôle de la contre-dépouille et le contrôle de l'angle de surplomb maximal (définies sur les frontières du domaine matériel) sont considérés comme un volume intégral sur le domaine de calcul. Il s'avère que les contraintes formulées selon l'approche de Qian imposent implicitement une limite au périmètre de la frontière soumise à des contre-dépouilles ou au périmètre de la frontière soumise à des problèmes de surplomb maximal. En dépit de cette signification physique, il n'est pas facile pour le concepteur de définir les paramètres d'optimisation car ils dépendent fortement du problème. De plus, comme la définition des contraintes induit une limitation du gradient de densité, une contrainte supplémentaire est nécessaire afin d'éviter des solutions triviales avec de larges régions planes au niveau de la fonction de densité (contrainte anti-gris). Au-delà du volume du support, d'autres aspects et potentiels liés à l'ALM sont pris en compte dans la formulation de l'OT : par exemple, dans le travail de Clausen [61], des applications multi-matériaux, des approches multi-échelles et des structures revêtues sont envisagées. Les auteurs de [64] tentent de créer une conception

de structures hybride topologique/treillis avec des exigences de fabrication.

## 1.5 Conclusions de la revue de la littérature

Cette étude bibliographique a mis en évidence certaines caractéristiques principales énumérées ci-dessous.

- Chaque technologie ALM a des exigences spécifiques. Néanmoins, des méthodologies générales de conception doivent être développées afin de garder le contrôle des caractéristiques communes de l'ALM. Les effets contradictoires de l'ensemble des paramètres que le concepteur peut ajuster lors de la production d'une pièce par ALM montrent clairement qu'il est impossible d'optimiser en même temps le temps de production, les coûts et les propriétés mécaniques. Ainsi, les choix de conception doivent être faits en référence à une exigence spécifique ou à une stratégie de compromis.
- Deux familles d'algorithmes principaux peuvent être identifiées dans la littérature pour l'OT. Le premier est composé d'algorithmes basés sur la densité, le second est constitué d'algorithmes basés sur la LSM. Les deux présentent des avantages et des inconvénients. Il n'est pas possible d'indiquer quelle famille fournit de meilleurs résultats et le choix doit être fait en considérant le problème en question.
- Il a été montré que considérer les contraintes de l'ALM (comme le volume du support) dans une phase de post-traitement d'un algorithme d'OT n'est pas une procédure robuste car des incohérences importantes entre les contraintes de conception imposées et les pièces réellement produites peuvent être observées. Cependant, la mise en œuvre de contraintes réalistes et représentatives de la physique dans les algorithmes d'OT n'est pas simple et des stratégies numériques appropriées doivent être développées afin de surmonter la forte non-linéarité liée à ces exigences. Des recherches sont en cours afin de fournir des algorithmes d'OT appropriés capables de gérer les contraintes d'ALM, et par conséquent, de faire des rapprochements constants entre ALM, optimisation et spécifications technologiques.



# Chapter 2

## Fundamentals of Geometrical Modelling

### 2.1 Introduction to the Fundamentals of Geometrical Modelling

This Chapter focuses on the fundamentals of B-Spline and NURBS geometrical entities. B-Spline and NURBS entities constitute a further development of the well-known Bézier's curves and surfaces [65–68]. The discussion starts with curves, pursues with surfaces and finally ends with the most general concept of NURBS/B-Spline hyper-surfaces. Among the valuable bibliographic references on the topic, it has been decided to adopt the notation of [68].

### 2.2 The NURBS curves theory

The parametric explicit form of a NURBS curve is:

$$\mathbf{C}(u_1) = \sum_{i_1=0}^{n_1} R_{i_1,p_1}(u_1) \mathbf{P}_{i_1}, \quad (2.1)$$

where  $\mathbf{C}(u_1) = \{x_1(u_1), x_2(u_1), x_3(u_1)\}$  are the Cartesian coordinates of a point belonging to the curve, whilst  $R_{i_1,p_1}(u_1)$  is the generic rational basis function having the form

$$R_{i_1,p_1}(u_1) = \frac{N_{i_1,p_1}(u_1)w_{i_1}}{\sum_{j=0}^{n_1} N_{j,p_1}(u_1)w_j}. \quad (2.2)$$

In Eqs. (2.1) and (2.2),  $u_1$  is a dimensionless parameter defined in the range  $[0, 1]$ ,  $N_{i_1,p_1}(u_1)$  are the basis functions, recursively defined according to Bernstein polynomials

[67, 68],  $p_1$  is the maximum degree,  $w_{i_1}$  are the weights and  $\mathbf{P}_{i_1} = \{X_{i_1}^{(1)}, X_{i_1}^{(2)}, X_{i_1}^{(3)}\}$  the Cartesian coordinates of the control points. The set of the  $(n_1 + 1)$  control points form the so-called *control polygon*. The blending functions  $N_{i_1, p_1}(u_1)$  are defined as

$$N_{i_1, 0}(u_1) = \begin{cases} 1, & \text{if } U_{i_1}^{(1)} \leq u_1 < U_{i_1+1}^{(1)}, \\ 0, & \text{otherwise,} \end{cases} \quad (2.3)$$

$$N_{i_1, q}(u_1) = \frac{u_1 - U_{i_1}^{(1)}}{U_{i_1+q}^{(1)} - U_{i_1}^{(1)}} N_{i_1, q-1}(u_1) + \frac{U_{i_1+q+1}^{(1)} - u_1}{U_{i_1+q+1}^{(1)} - U_{i_1+1}^{(1)}} N_{i_1+1, q-1}(u_1), \quad q = 1, \dots, p_1, \quad (2.4)$$

where  $U_{i_1}^{(1)}$  is the  $i_1$ -th component of the following non-periodic non-uniform *knot vector*:

$$\mathbf{U}^{(1)} = \underbrace{\{0, \dots, 0\}}_{p_1+1}, U_{p_1+1}^{(1)}, \dots, U_{m_1-p_1-1}^{(1)}, \underbrace{\{1, \dots, 1\}}_{p_1+1}. \quad (2.5)$$

It is noteworthy that the size of the knot vector is  $m_1 + 1$ , with

$$m_1 = n_1 + p_1 + 1. \quad (2.6)$$

The knot vector is a non-decreasing sequence of real numbers that can be interpreted as a discrete collection of values of the dimensionless parameter  $u_1$ , splitting the curve in arcs. The components of  $\mathbf{U}^{(1)}$  are called *knots* and each knot can have a multiplicity  $\lambda$ .

Among the properties characterising blending functions, one of the most important is the *partition of unit property*, i.e.

$$\sum_{i_1=0}^{n_1} N_{i_1, p_1}(u_1) = 1, \quad \forall u_1 \in [0, 1]. \quad (2.7)$$

This property allows for defining the simpler B-Spline curve starting from the more general definition of NURBS curve of Eqs. (2.1)-(2.2). Indeed, B-Spline curves constitute a subset of more general NURBS curves because the weights  $w_{i_1}$  take the same value. Considering the partition of unit property, Eq. (2.2) simplifies into  $R_{i_1, p_1}(u_1) = N_{i_1, p_1}(u_1)$  and the equation characterising a B-Spline curve reads

$$\mathbf{C}(u_1) = \sum_{i_1=0}^{n_1} N_{i_1, p_1}(u_1) \mathbf{P}_{i_1}. \quad (2.8)$$

Historically, the interest behind the utilisation of NURBS curves is related to their ability to exactly represent conic sections (parabola, hyperbola, ellipse), unlike B-Spline and Bézier's curves, which can only approximate conics.

In order to clarify the previous concepts, a meaningful example is proposed here below. In this example, two plane B-Spline and NURBS curves are represented: both curves are

## 2.2. The NURBS curves theory

characterised by blending functions of degree  $p_1 = 3$  and by means of seven control points ( $n_1 = 6$ ). For each curve, two knot vectors have been considered: the first one is a uniform knot vector,

$$\mathbf{U}_{\mathbf{A}}^{(1)} = \{0, 0, 0, 0, 0.25, 0.50, 0.75, 1, 1, 1, 1\}. \quad (2.9)$$

whilst the second one has a component with multiplicity  $\lambda = 2$ ,

$$\mathbf{U}_{\mathbf{B}}^{(1)} = \{0, 0, 0, 0, 0.25, 0.60, 0.60, 1, 1, 1, 1\}. \quad (2.10)$$

The corresponding blending functions are represented in Fig. 2.1. The B-Spline and NURBS curves are shown in Fig. 2.2: the NURBS curve has been obtained by assigning a stronger weight to the fifth control point. As it can be inferred from Fig. 2.2, when the weight related to a given control point increases, the curve is attracted towards such a control point.

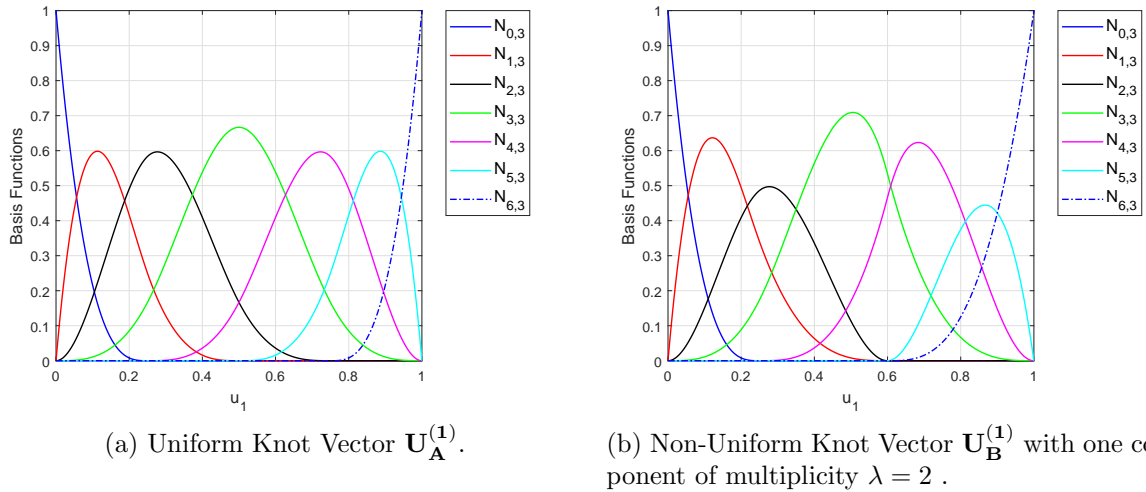


Figure 2.1 – Example of Blending Functions for B-Spline/NURBS curves of degree  $p_1 = 3$  and number of control points  $n_1 = 6$ .

Some of the most important properties characterising B-Spline and NURBS curves can be deduced by Figs. 2.1-2.2.

- a) *Local support property.* It can be observed that each blending function is defined within a given local support (i.e. the blending function is identically null outside of an assigned interval of values of the parameter  $u_1$ ). The dimension of the local support depends on the knot vector distribution. Accordingly, the local support property can be stated as

$$N_{i_1, p_1}(u_1) \neq 0 \text{ if } u_1 \in \left[ U_{i_1}^{(1)}, U_{i_1 + p_1 + 1}^{(1)} \right]. \quad (2.11)$$



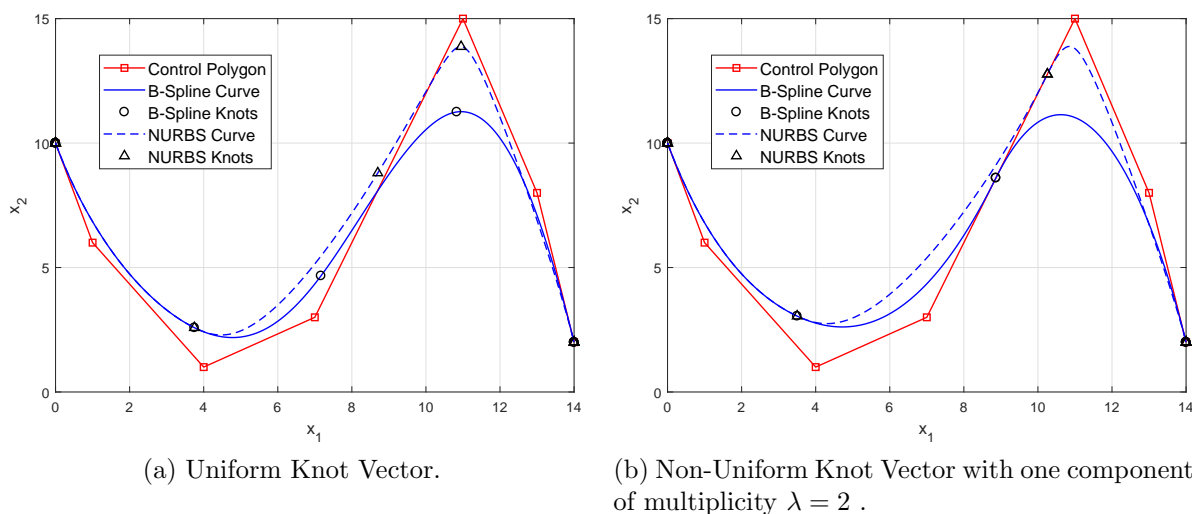


Figure 2.2 – Example of B-Spline and NURBS curves of degree  $p_1 = 3$  and number of control points  $n_1 = 6$ .

Therefore, moving a control point or changing the value of its related weight has a *local effect* on the curve shape: it affects only those curve points whose respective parameter  $u_1$  values fall within the local support of the blending function related to the considered control point, as it can be observed in Fig. 2.2. Since the local support of the blending function  $N_{4,3}$  does not include the first part of the parametric space, the first arcs of the NURBS and of the BSpline curves are the same, and the influence of the weight related to control point  $P_4$  can be observed starting from the second knot.

- b) *Strong convex hull property.* Since the blending functions are non-negative and since the unit partition property holds, it can be shown that the NURBS/B-Spline curve  $\mathbf{C}(u_1)$  is always in the convex hull of the control polygon [68].
- c) *Continuity and differentiability property.* The basis function  $N_{i_1, p_1}(u_1)$  is  $p_1 - \lambda$  times continuously differentiable at a given knot. Thus, increasing the degree increases the continuity, whilst increasing the knot multiplicity decreases the continuity. It is evident that the knot vector strongly affects the basis functions and, accordingly, the shape of a NURBS curve. The curves of Fig. 2.2a are continuous, with first and second derivative continuous. The curves of Fig. 2.2b undergo a jump of curvature at the knot with multiplicity  $\lambda = 2$ .

It is noteworthy that only those properties which are useful for this thesis have been recalled here. However, this is not an exhaustive discussion about the B-Spline/NURBS curves properties. For a deeper insight in the matter, the reader is addressed to [66–68].

## 2.3 The NURBS surfaces theory

NURBS surfaces formulae can be deduced by those on NURBS curves. It will be pointed out that the previously stated properties can be straightforwardly reformulated also for NURBS surfaces.

A point belonging to a NURBS surface is defined through

$$\mathbf{S}(u_1, u_2) = \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} R_{i_1, i_2}(u_1, u_2) \mathbf{P}_{i_1, i_2}, \quad (2.12)$$

where  $R_{i_1, i_2}(u_1, u_2)$  are the piecewise rational basis functions, which are related to the NURBS blending functions  $N_{i_1, p_1}(u_1)$  and  $N_{i_2, p_2}(u_2)$  by means of the relationship

$$R_{i_1, i_2}(u_1, u_2) = \frac{w_{i_1, i_2} N_{i_1, p_1}(u_1) N_{i_2, p_2}(u_2)}{\sum_{j_1=0}^{n_1} \sum_{j_2=0}^{n_2} w_{j_1, j_2} N_{j_1, p_1}(u_1) N_{j_2, p_2}(u_2)}. \quad (2.13)$$

In Eqs. (2.12) and (2.13),  $\mathbf{S}(u_1, u_2) \in \mathbb{R}^3$  is a bivariate vector-valued piecewise rational function,  $(u_1, u_2)$  are scalar dimensionless parameters both defined in the interval  $[0, 1]$ , while  $p_1$  and  $p_2$  are the NURBS degrees along  $u_1$  and  $u_2$  directions, respectively;  $w_{i_1, i_2}$  are the weights and  $\mathbf{P}_{i_1, i_2} = \{X_{i_1, i_2}^{(1)}, X_{i_1, i_2}^{(2)}, X_{i_1, i_2}^{(3)}\}$  the Cartesian coordinates of the generic control point, with  $i_1 = 0, \dots, n_1$ ,  $i_2 = 0, \dots, n_2$  and  $\mathbf{X}^{(j)} \in \mathbb{R}^{(n_1+1) \times (n_2+1)}$ ,  $j = 1, 2, 3$ . The net of  $(n_1 + 1) \times (n_2 + 1)$  control points constitutes the so-called *control net*. Each parametric direction  $u_1$  and  $u_2$  requires a family of blending functions to correctly define the NURBS surface. The blending functions are recursively defined by means of the Bernstein's polynomials and Eqs. (2.3)-(2.4) still apply for the two families  $N_{i_1, p_1}(u_1)$  and  $N_{i_2, p_2}(u_2)$ . Consistently, two knot vectors are needed, i.e.

$$\mathbf{U}^{(1)} = \{\underbrace{0, \dots, 0}_{p_1+1}, U_{p_1+1}^{(1)}, \dots, U_{m_1-p_1-1}^{(1)}, \underbrace{1, \dots, 1}_{p_1+1}\}, \quad (2.14)$$

$$\mathbf{U}^{(2)} = \{\underbrace{0, \dots, 0}_{p_2+1}, U_{p_2+1}^{(2)}, \dots, U_{m_2-p_2-1}^{(2)}, \underbrace{1, \dots, 1}_{p_2+1}\}, \quad (2.15)$$

whose sizes are  $m_1 + 1$  and  $m_2 + 1$ , respectively. The two integer quantities  $m_1$  and  $m_2$  are related to the respective degree and the respective number of control points according to Eq. (2.6). In the framework of NURBS surfaces, the knots are identified as combinations of knot vectors components. Particularly, a surface point corresponding to a knot can be identified by setting  $u_1$  equal to whatever  $\mathbf{U}^{(1)}$  component,  $u_2$  equal to whatever  $\mathbf{U}^{(2)}$  component and by injecting these dimensionless parameters values into Eq. (2.12).

As in the case of NURBS curves, also for NURBS surfaces the *partition of unit property* holds and it reads

$$\sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} N_{i_1,p_1}(u_1)N_{i_2,p_2}(u_2) = 1, \forall (u_1, u_2) \in [0, 1] \times [0, 1]. \quad (2.16)$$

Of course, the general definition of a NURBS surface includes that of a B-Spline surface as well: if all the weights  $w_{i_1,i_2}$  take the same value, the rational basis function of Eq. (2.13) simplifies in  $R_{i_1,i_2}(u_1, u_2) = N_{i_1,p_1}(u_1)N_{i_2,p_2}(u_2)$  and, consequently, Eq. (2.12) changes into

$$\mathbf{S}(u_1, u_2) = \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} N_{i_1,p_1}(u_1)N_{i_2,p_2}(u_2)\mathbf{P}_{i_1,i_2}. \quad (2.17)$$

A simple example is provided here below to give an intuitive idea of the difference among NURBS and B-Spline surfaces (the interested reader is addressed to [68] for a deeper insight into the matter). For instance, the two surfaces of Fig. 2.3 have been obtained by considering the same knot vectors, degrees and control points coordinates (the control net, in red, is the same for the two surfaces). However, the surface of Fig. 2.3a is a B-Spline surface, whilst the one of Fig. 2.3b is a NURBS surface. As it is confirmed by a visual comparison between Fig. 2.3a and Fig. 2.3b, the weight corresponding to the highest peak has been decreased, while the weight associated to the control point of the smallest peak has been increased: therefore, the higher is the value of the weight  $w_{i_1,i_2}$ , the more the surface is attracted to the corresponding control point  $\mathbf{P}_{i_1,i_2}$  and *vice versa*.

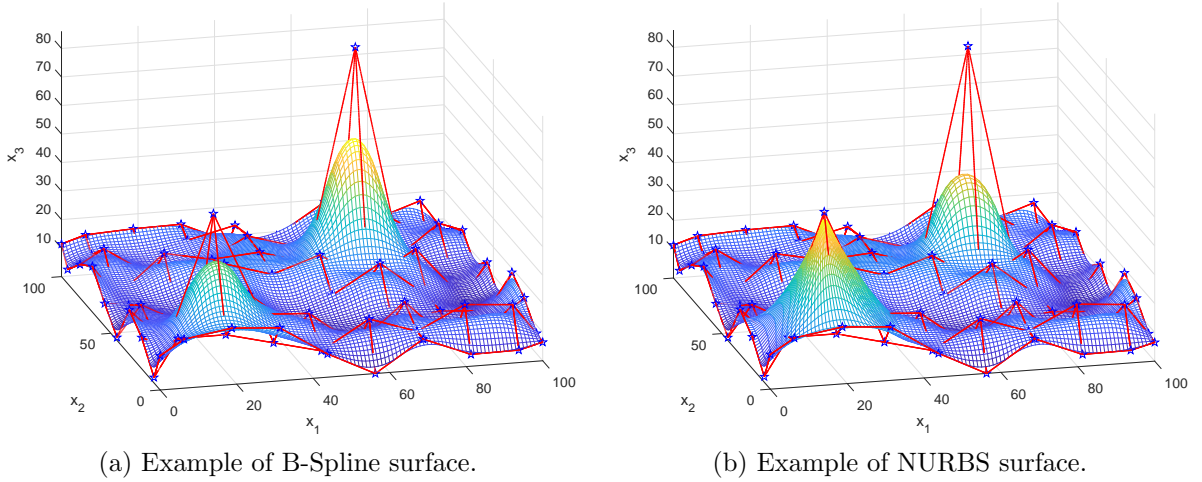


Figure 2.3 – B-Spline and NURBS surfaces with same degrees, control points and knot vectors.

The main properties already evoked for NURBS curves can be transposed to the case of NURBS surfaces as well:

- a) *Local support property.*  $N_{i_1, p_1}(u_1) = 0$  if  $u_1$  is outside the interval  $[U_{i_1}^{(1)}, U_{i_1+p_1+1}^{(1)}]$ . Therefore, it is evident that

$$R_{i_1, i_2}(u_1, u_2) \neq 0 \text{ if } (u_1, u_2) \in [U_{i_1}^{(1)}, U_{i_1+p_1+1}^{(1)}] \times [U_{i_2}^{(2)}, U_{i_2+p_2+1}^{(2)}]. \quad (2.18)$$

The rectangle  $[U_{i_1}^{(1)}, U_{i_1+p_1+1}^{(1)}] \times [U_{i_2}^{(2)}, U_{i_2+p_2+1}^{(2)}]$  is the local support associated to the control point  $\mathbf{P}_{i_1, i_2}$ . This property is evident by observing that the effect of the weights in Fig. 2.3b is restrained to a limited portion of the global domain wherein the NURBS is defined: roughly speaking, the NURBS is unchanged with respect to the B-Spline of Fig. 2.3a within those zones far away from the two highest peaks.

- b) *Strong convex hull property.* Since the blending functions are non-negative and since the unit partition property holds, it can be shown that the NURBS/B-Spline surface  $\mathbf{S}(u_1, u_2)$  is always in the convex hull of the control net [68].
- c) *Continuity and differentiability property.* The same criteria described in the case of curves blending functions apply to each blending function family  $N_{i_1, p_1}(u_1)$  and  $N_{i_2, p_2}(u_2)$  in case of NURBS surfaces. In general, it can be stated that increasing the knot multiplicity in whatever parametric direction decreases the differentiability of the surface along that parametric direction.

## 2.4 The NURBS hyper-surfaces theory

The concept of *NURBS hyper-surfaces* is presented in this section. In general, a NURBS hyper-surface is defined as a function  $\mathbf{H} : \mathbb{R}^N \rightarrow \mathbb{R}^M$ , where  $N$  is the dimension of the *parametric space* (domain) and  $M$  the dimension of the *hyper-surface space* (co-domain). For instance, if  $N = 1$  and  $M = 2$ , the NURBS entity is a plane curve, whilst a NURBS curve in the 3D space is characterised by  $N = 1$  and  $M = 3$ . Of course,  $N = 2$  and  $M = 3$  in the case of a NURBS surface. Generally speaking, a NURBS hyper-surface is characterised by  $M > 3$ . In the following of this manuscript, particular interest will be dedicated to the special case of  $N = 3$  and  $M = 4$ : as discussed in Chapter 4, this peculiar geometric entity will be used in the formulation of the topology optimisation problem in the most general 3D case.

The NURBS hyper-surfaces theory is a natural generalisation of the NURBS surfaces theory. Contrary to the NURBS surfaces case, wherein two scalar parameters  $u_1$  and  $u_2$  allow for defining three spatial coordinates, a NURBS hyper-surface cannot be represented in the standard 3D space. So, the generic formula of a NURBS hyper-surface  $\mathbf{H} : \mathbb{R}^N \rightarrow \mathbb{R}^M$  is

$$\mathbf{H}(u_1, \dots, u_N) = \sum_{i_1=0}^{n_1} \dots \sum_{i_N=0}^{n_N} R_{i_1, \dots, i_N}(u_1, \dots, u_N) \mathbf{P}_{i_1, \dots, i_N}, \quad (2.19)$$

where the expression of  $R_{i_1, \dots, i_N}(u_1, \dots, u_N)$  writes

$$R_{i_1, \dots, i_N}(u_1, \dots, u_N) = \frac{w_{i_1, \dots, i_N} \prod_{k=1}^N N_{i_k, p_k}(u_k)}{\sum_{j_1=0}^{n_1} \dots \sum_{j_N=0}^{n_N} \left[ w_{j_1, \dots, j_N} \prod_{k=1}^N N_{j_k, p_k}(u_k) \right]}. \quad (2.20)$$

In Eq. (2.19),  $\mathbf{H}(u_1, \dots, u_N)$  is a  $M$ -dimensional vector-valued rational function. The  $j$ -th hyper-surface control point coordinate ( $X_{i_1, \dots, i_N}^{(j)}$ ) is stored in the array  $\mathbf{X}^{(j)}$ , whose size is  $(n_1 + 1) \times \dots \times (n_N + 1)$ . In this way, control points constituting the so-called *control hyper-net* have the following coordinates in  $\mathbb{R}^M$ :

$$\begin{aligned} \mathbf{P}_{i_1, \dots, i_N} &= \{X_{i_1, \dots, i_N}^{(1)}, \dots, X_{i_1, \dots, i_N}^{(M)}\}, \\ \mathbf{X}^{(j)} &\in \mathbb{R}^{(n_1+1) \times \dots \times (n_N+1)}, \quad j = 1, \dots, M. \end{aligned} \quad (2.21)$$

A suitable weight  $w_{i_1, \dots, i_N}$  is related to the respective control point  $\mathbf{P}_{i_1, \dots, i_N}$ . In perfect analogy with NURBS surfaces,  $(u_1, \dots, u_N)$  are the scalar dimensionless parameters, defined in the interval  $[0, 1]$ . For each parametric direction, described through the generic  $u_k$ , the NURBS blending functions are characterised by the degree  $p_k$ ; the generic expression of the blending function related to the parametric direction  $u_k$  is

$$N_{i_k, 0}(u_k) = \begin{cases} 1 & \text{if } U_{i_k}^{(k)} \leq u_k < U_{i_k+1}^{(k)}, \\ 0 & \text{otherwise,} \end{cases} \quad (2.22)$$

$$N_{i_k, q}(u_k) = \frac{u_k - U_{i_k}^{(k)}}{U_{i_k+q}^{(k)} - U_{i_k}^{(k)}} N_{i_k, q-1}(u_k) + \frac{U_{i_k+q+1}^{(k)} - u_k}{U_{i_k+q+1}^{(k)} - U_{i_k+1}^{(k)}} N_{i_k+1, q-1}(u_k), \quad (2.23)$$

$$q = 1, \dots, p_k,$$

where each constitutive function is defined on the related knot vector

$$\mathbf{U}^{(k)} = \underbrace{\{0, \dots, 0\}}_{p_k+1}, U_{p_k+1}^{(k)}, \dots, U_{m_k-p_k-1}^{(k)}, \underbrace{\{1, \dots, 1\}}_{p_k+1}, \quad (2.24)$$

whose dimension is  $m_k + 1$ , with

$$m_k = n_k + p_k + 1. \quad (2.25)$$

The concept of B-Spline hyper-surface is inferred from the NURBS hyper-surface definition. Even in this case, it can be stated that the product of the  $N$  blending functions satisfies the partition of unit property, i.e.,

$$\sum_{j_1=0}^{n_1} \dots \sum_{j_N=0}^{n_N} \left( \prod_{k=1}^N N_{j_k, p_k}(u_k) \right) = 1. \quad (2.26)$$

Hence, eliminating the dependence on the NURBS weights, the rational basis functions simplify into  $R_{i_1, \dots, i_N}(u_1, \dots, u_N) = \prod_{k=1}^N N_{i_k, p_k}(u_k)$  and, accordingly, the general B-Spline hyper-surfaces formula reads

$$\mathbf{H}(u_1, \dots, u_N) = \sum_{i_1=0}^{n_1} \dots \sum_{i_N=0}^{n_N} \left( \prod_{k=1}^N N_{i_k, p_k}(u_k) \right) \mathbf{P}_{i_1, \dots, i_N}. \quad (2.27)$$

Of course, representing a NURBS/B-Spline hyper-surface is, in general, not an easy task. An example of 4D B-Spline and NURBS hyper-surfaces having four coordinates and depending on three parameters is provided in Fig. 2.4 ( $N = 3$ ,  $M = 4$ ). In this particular case, the representation is possible since the first three components of the array  $\mathbf{H}$  are associated to the physical spatial coordinates in the reference frame  $O(x_1, x_2, x_3)$ . The fourth coordinate of  $\mathbf{H}$  is plotted by means of a colour legend. All control points related to the the fourth coordinate of the B-Spline hyper-surface of Fig. 2.4a take the same value, apart eight of them. Those eight special control points coordinates, which are placed near the eight corners of the cubic domain, take higher values. This results in warmer colours close to the eight domain corners. The NURBS of Fig. 2.5b is obtained as follows: as far as the eight previously described special control points are concerned, increased weights are assigned to the four control points on the top of the domain, whilst decreased weights are assigned to those special control points of the bottom part. The effects can be observed by comparing the colour scale in Fig. 2.4. In order to provide a clearer idea about the behaviour of the NURBS hyper-surface with respect to the B-Spline one, the iso-level contour plot, corresponding to the threshold value of 10, has been represented in Fig. 2.5a. Of course, only in the case  $N = 3$ ,  $M = 4$  the iso-level contour plot of the hyper-surface is a 3D surface.

As far as the NURBS hyper-surfaces properties are concerned, they still apply even in the most general case. The *strong convex hull property* and the *differentiability* deepened for the NURBS curves and surfaces can be easily transposed to the most general hyper-surface case. Furthermore, considering the hyper-net constituted by the components of the  $N$  knot vectors, the *local support property* holds for NURBS hyper-surfaces as well:

$$R_{i_1, \dots, i_N}(u_1, \dots, u_N) \neq 0 \text{ if } (u_1, \dots, u_N) \in \left[ U_{i_1}^{(1)}, U_{i_1+p_1+1}^{(1)} \right] \times \dots \times \left[ U_{i_N}^{(N)}, U_{i_N+p_N+1}^{(N)} \right], \quad (2.28)$$

where the hyper-rectangle  $\left[ U_{i_1}^{(1)}, U_{i_1+p_1+1}^{(1)} \right] \times \dots \times \left[ U_{i_N}^{(N)}, U_{i_N+p_N+1}^{(N)} \right]$  is the influence zone affected by the control point  $\mathbf{P}_{i_1, \dots, i_N}$ .

## 2.5 Conclusions on the NURBS entities theory

In this chapter, the fundamentals of NURBS and B-Spline curves, surfaces and hyper-surfaces have been provided. Although these entities are conceived for an immediate

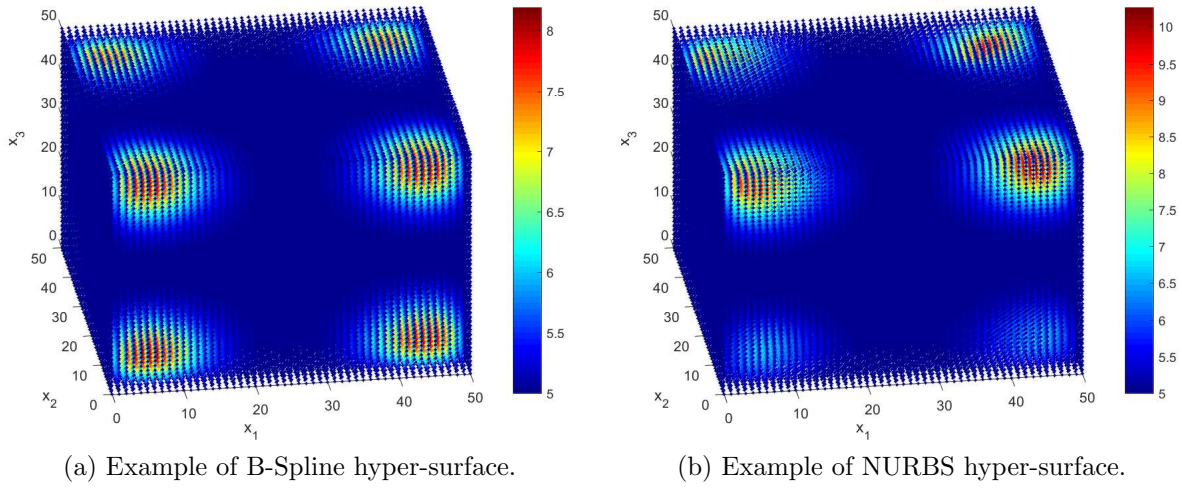


Figure 2.4 – B-Spline and NURBS hyper-surfaces with same degrees, control points and knot vectors.

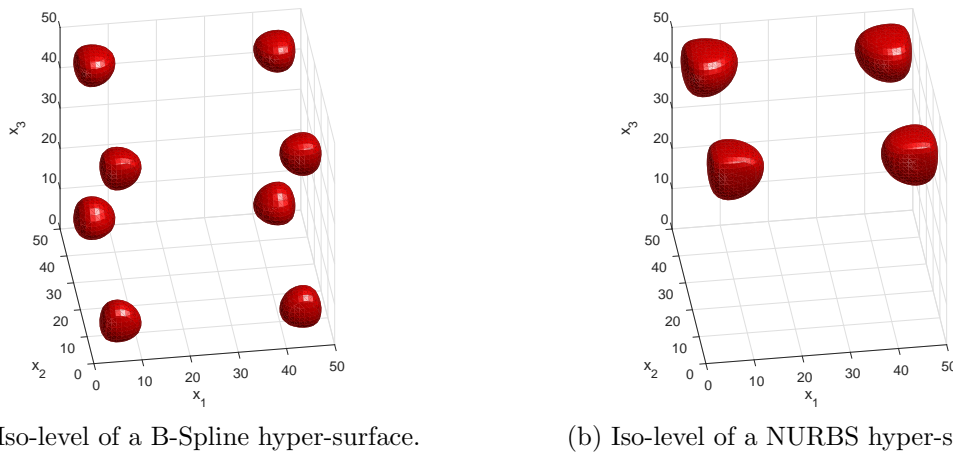


Figure 2.5 – Iso-level corresponding to 10 of B-Spline and NURBS hyper-surfaces.

application in the CAD environment, the mathematical background is very general and different applications can be envisaged. B-Spline and NURBS geometric entities have very general and interesting properties, which can be exploited in other applications, different from those proposed in [68]. Indeed, several authors have taken advantage of the wide range of possibilities provided by the NURBS in solving many problems of different nature, as it will be remarked in the next Chapters.

# Chapitre 2

## Principes fondamentaux de la modélisation géométrique

### 2.1 Introduction aux principes fondamentaux de la modélisation géométrique

Ce chapitre se concentre sur les principes fondamentaux des entités géométriques B-Spline et NURBS. Les entités B-Spline et NURBS constituent un développement complémentaire des courbes et des surfaces de Bézier bien connues [65–68]. Ce chapitre commence par une discussion sur les courbes, puis se poursuit avec une présentation des surfaces et se finit avec le concept plus général des hyper-surfaces NURBS / B-Spline. Parmi les nombreuses références bibliographiques sur le sujet, il a été décidé d'adopter la notation de [68].

### 2.2 La théorie des courbes NURBS

La forme explicite paramétrique d'une courbe NURBS est présentée à l'éq. (2.1). Elle est définie par la fonction de base rationnelle générique ayant la forme de l'éq. (2.2). Dans les éqs. (2.1) et (2.2),  $u_1$  est un paramètre sans dimension défini dans l'intervalle  $[0, 1]$ ,  $N_{i_1, p_1}(u_1)$  sont les fonctions de base, définies récursivement selon les polynômes de Bernstein [67, 68],  $p_1$  est le degré maximum,  $w_{i_1}$  sont les poids et  $\mathbf{P}_{i_1} = \{X_{i_1}^{(1)}, X_{i_1}^{(2)}, X_{i_1}^{(3)}\}$  les coordonnées cartésiennes des points de contrôle. L'ensemble des points de contrôle  $(n_1 + 1)$  forme le polygone de contrôle.

Les fonctions de base  $N_{i_1, p_1}(u_1)$  sont définies selon les éqs. (2.3) et (2.4), où  $U_{i_1}^{(1)}$  est la composante  $i_1$ -th du *knot vector* non périodique, non uniforme, dont l'expression est énoncée dans l'éq. (2.5). Il convient de noter que la taille du *knot vector* est  $m_1 + 1$ , et  $m_1$  est lié à  $p_1$  et à  $n_1$  par l'éq. (2.6). Le *knot vector* est une suite de nombres réels non décroissante qui peut être interprétée comme une liste discrète de valeurs du paramètre sans dimension  $u_1$ , divisant la courbe en arcs. Les composantes de  $\mathbf{U}^{(1)}$  sont appelées noeuds et chaque valeur correspondante à un noeud peut se retrouver  $\lambda$  fois dans la liste, on a alors à faire



à une multiplicité  $\lambda$ .

Parmi les propriétés caractérisant les fonctions de base, l'une des plus importantes est la propriété de partition de l'unité, définie par la relation de l'éq. (2.7). Cette propriété permet de définir la courbe B-Spline plus simplement à partir de la définition plus générale de la courbe NURBS des éqs. (2.1)-(2.2). En effet, les courbes B-Spline constituent un sous-ensemble de courbes NURBS car les poids  $w_{i_1}$  prennent la même valeur. Considérant la partition de la propriété de l'unité, l'éq. (2.2) se simplifie et l'équation caractérisant une courbe B-Spline est fournie par l'éq. (2.8). Historiquement, l'intérêt de l'utilisation des courbes NURBS est lié à leur capacité à représenter exactement les coniques (parabole, hyperbole, ellipse), contrairement aux courbes B-Spline et de Bézier, qui ne peuvent qu'approcher les coniques.

Afin de clarifier les concepts précédents, un exemple est proposé. Une courbe plane de type B-Spline et une courbe NURBS plane sont représentées : les deux courbes sont caractérisées par des fonctions de base de degré  $p_1 = 3$  et au moyen de sept points de contrôle ( $n_1 = 6$ ). Pour chaque courbe, deux *knot vectors* ont été considérés : le premier est un *knot vector* uniforme (voir l'éq. (2.9)), tandis que le second a une composante de multiplicité  $\lambda = 2$  (voir l'éq. (2.10)). Les fonctions de base correspondantes sont représentées sur la fig. 2.1. Les courbes B-Spline et NURBS sont représentées sur la fig. 2.2 : la courbe NURBS a été obtenue en assignant un poids plus fort au cinquième point de contrôle. Comme on peut le déduire de la fig. 2.2, lorsque le poids lié à un point de contrôle donné augmente, la courbe est attirée vers un tel point de contrôle.

Certaines des propriétés les plus importantes caractérisant les courbes B-Spline et NURBS peuvent être déduites par les figs. 2.1-2.2. Chaque fonction de base est identiquement nulle en dehors d'un intervalle de valeurs assigné au paramètre  $u_1$  (propriété du support local). La dimension du support local dépend de la distribution du *knot vector* selon l'éq. (2.11). Une application de cette propriété est que le support local de la fonction de base  $N_{4,3}$  de l'exemple n'inclut pas la première partie de l'espace paramétrique : en conséquence, les premiers arcs des courbes NURBS et B-Spline sont les mêmes, et l'influence du poids lié au point de contrôle  $\mathbf{P}_4$  peut être observée à partir du second nœud. De plus, comme les fonctions de base sont non négatives et que la propriété de partition de l'unité est valide, on peut montrer que la courbe NURBS / B-Spline est toujours dans l'enveloppe convexe du polygone de contrôle [68]. Enfin, la fonction de base  $N_{i_1, p_1}(u_1)$  est  $p_1 - \lambda$  est continuellement différentiable à un nœud donné. Ainsi, augmenter le degré augmente la continuité, tandis que l'augmentation de la multiplicité des nœuds diminue la continuité. Pour un aperçu plus important de la question, le lecteur pourra se référer aux références [66–68].

## 2.3 La théorie des surfaces NURBS

Les formules des surfaces NURBS peuvent être déduites de celles des courbes NURBS. Les propriétés précédemment indiquées peuvent être étendues directement aux surfaces

NURBS.

Un point appartenant à une surface NURBS est défini par l'éq. (2.12), tandis que la relation entre les fonctions de base rationnelles par morceaux et les fonctions de base NURBS est exprimée par l'éq. (2.13). Dans les éqs. (2.12) et (2.13),  $\mathbf{S}(u_1, u_2) \in \mathbb{R}^3$  est une fonction rationnelle par morceaux à deux variables,  $(u_1, u_2)$  sont des paramètres scalaires sans dimension définis dans l'intervalle  $[0, 1]$ , tandis que  $p_1$  et  $p_2$  sont les degrés NURBS selon les directions  $u_1$  et  $u_2$ , respectivement ;  $w_{i_1, i_2}$  sont les poids et  $\mathbf{P}_{i_1, i_2} = \{X_{i_1, i_2}^{(1)}, X_{i_1, i_2}^{(2)}, X_{i_1, i_2}^{(3)}\}$  les coordonnées cartésiennes du point de contrôle générique, avec  $i_1 = 0, \dots, n_1, i_2 = 0, \dots, n_2$  et  $\mathbf{X}^{(j)} \in \mathbb{R}^{(n_1+1) \times (n_2+1)}$ ,  $j = 1, 2, 3$ . Le réseau de points de contrôle  $(n_1 + 1) \times (n_2 + 1)$  constitue ce que l'on appelle le réseau de contrôle. Chaque direction paramétrique  $u_1$  et  $u_2$  nécessite une famille de fonctions de base pour définir correctement la surface NURBS. Les fonctions de base sont définies récursivement au moyen des polynômes de Bernstein et les éqs. (2.3)-(2.4) s'appliquent toujours aux deux familles  $N_{i_1, p_1}(u_1)$  et  $N_{i_2, p_2}(u_2)$ . Constamment, les deux *knot vectors* des éqs. (2.14)-(2.15) sont nécessaires. La propriété de partition de l'unité dans le cas des surfaces NURBS est fournie par l'éq. (2.16). Bien entendu, la définition générale d'une surface NURBS inclut également celle d'une surface B-Spline : si tous les poids  $w_{i_1, i_2}$  prennent la même valeur, l'éq. (2.12) change en éq. (2.17).

Par exemple, les deux surfaces de la fig. 2.3 ont été obtenues en considérant les mêmes *knot vectors*, les mêmes degrés et les mêmes coordonnées des points de contrôle (le réseau de contrôle, en rouge, est le même pour les deux surfaces). Cependant, la surface de la fig. 2.3a est une surface B-Spline, tandis que celle de la fig. 2.3b est une surface NURBS. Comme cela est confirmé par une comparaison visuelle entre la fig. 2.3a et la fig. 2.3b, le poids correspondant au pic le plus élevé a été diminué, tandis que le poids associé au point de contrôle du plus petit pic a été augmenté : par conséquent, plus le poids  $w_{i_1, i_2}$  est élevé, plus la surface est attirée par le point de contrôle correspondant  $\mathbf{P}_{i_1, i_2}$  et vice versa.

Les propriétés principales déjà évoquées pour les courbes NURBS peuvent également être transposées dans le cas des surfaces NURBS. La propriété de support local pour les surfaces NURBS est exprimée par l'éq. (2.18). Cette propriété est évidente en observant l'effet des poids sur la fig. 2.3b qui est restreint à une partie limitée du domaine global dans lequel la NURBS est définie : la NURBS est inchangée par rapport à la B-Spline de la fig. 2.3a dans les zones éloignées des deux plus hauts sommets. De manière analogue au cas des courbes NURBS, on peut montrer que la surface NURBS/B-Spline est toujours dans l'enveloppe convexe du réseau de contrôle [68]. En ce qui concerne la propriété de continuité et de différentiabilité, on peut affirmer que l'augmentation de la multiplicité des nœuds dans n'importe quelle direction paramétrique diminue la différentiabilité de la surface le long de cette direction paramétrique.

## 2.4 La théorie des hyper-surfaces NURBS

Le concept des hyper-surfaces NURBS est présenté dans ce paragraphe. En général, une hyper-surface NURBS est définie comme une fonction  $\mathbf{H} : \mathbb{R}^N \rightarrow \mathbb{R}^M$ , où  $N$  est la dimension de l'espace paramétrique (domaine) et  $M$  la dimension de l'espace hyper-surface (co-domaine). Par exemple, si  $N = 1$  et  $M = 2$ , l'entité NURBS est une courbe plane, tandis qu'une courbe NURBS dans l'espace 3D est caractérisée par  $N = 1$  et  $M = 3$ . Bien sûr,  $N = 2$  et  $M = 3$  dans le cas d'une surface NURBS. D'une manière générale, une hyper-surface NURBS est caractérisée par  $M > 3$ . La théorie des hyper-surfaces NURBS est une généralisation naturelle de la théorie des surfaces NURBS. Contrairement au cas des surfaces NURBS, où deux paramètres scalaires  $u_1$  et  $u_2$  permettent de définir trois coordonnées spatiales, une hyper-surface NURBS ne peut pas être représentée dans l'espace 3D standard. Ainsi, la formule générique d'une hyper-surface NURBS  $\mathbf{H} : \mathbb{R}^N \rightarrow \mathbb{R}^M$  est donnée par les éqs. (2.19)-(2.20).

Dans l'éq. (2.19),  $\mathbf{H}(u_1, \dots, u_N)$  est une fonction rationnelle vectorielle de dimension  $M$ . La coordonnée du point de contrôle de l'hyper-surface  $j$ -th ( $X_{i_1, \dots, i_N}^{(j)}$ ) est stockée dans la matrice  $\mathbf{X}^{(j)}$ , dont la taille est  $(n_1 + 1) \times \dots \times (n_N + 1)$ . De cette manière, les coordonnées des points de contrôle sont données par l'éq. (2.21) et peut être exprimé dans  $\mathbb{R}^M$ . Un poids approprié  $w_{i_1, \dots, i_N}$  est lié au point de contrôle respectif  $\mathbf{P}_{i_1, \dots, i_N}$ . En parfaite analogie avec les surfaces NURBS,  $(u_1, \dots, u_N)$  sont les paramètres scalaires sans dimension, définis dans l'intervalle  $[0, 1]$ . Pour chaque direction paramétrique, décrite par le  $u_k$  générique, les fonctions de base NURBS sont caractérisées par le degré  $p_k$ ; l'expression générique de la fonction de base liée à la direction paramétrique  $u_k$  peut être trouvée dans les éqs. (2.22)-(2.23), où chaque fonction constitutive est définie sur le *knot vector* associé (éq. (2.24)).

Le concept d'hyper-surface B-Spline est déduit de la définition des hyper-surfaces NURBS. Même dans ce cas, on peut affirmer que le produit des fonctions de base  $N$  satisfait la propriété de partition de l'unité (voir l'éq. (2.26)). Par conséquent, en éliminant la dépendance sur les poids des NURBS, les fonctions de base rationnelles se simplifient et, en conséquence, la formule générale des hyper-surfaces B-Spline s'écrit comme celle de l'éq. (2.27).

La représentation d'une hyper-surface NURBS / B-Spline n'est en général pas une tâche facile. Un exemple d'hyper-surfaces 4D B-Spline et NURBS ayant quatre coordonnées et dépendant de trois paramètres est fourni dans la fig. 2.4 ( $N = 3$ ,  $M = 4$ ). Dans ce cas particulier, la représentation est possible puisque les trois premières composantes de la matrice  $\mathbf{H}$  sont associées aux coordonnées spatiales physiques dans le repère  $O(x_1, x_2, x_3)$ . La quatrième coordonnée de  $\mathbf{H}$  est tracée au moyen d'une légende de couleur. Tous les points de contrôle liés à la quatrième coordonnée de l'hyper-surface B-Spline de la fig. 2.4a prennent la même valeur, à part huit d'entre eux. Ces huit points de contrôle spéciaux, placés près des huit coins du domaine cubique, prennent des valeurs plus élevées.

Il en résulte des couleurs plus chaudes proches des huit coins du domaine. La NURBS de la fig. 2.5b est obtenu avec les huit points de contrôle spéciaux précédemment décrits, des poids accrus sont attribués aux quatre points de contrôle situés en haut du domaine, tandis que des poids réduits sont affectés aux points de contrôle spéciaux de la partie inférieure. Les effets peuvent être observés en comparant l'échelle de couleur de la fig. 2.4. La courbe d'iso-niveau, correspondant à la valeur seuil de 10, a été représentée sur la fig. 2.5. Bien sûr, seulement dans le cas  $N = 3$ ,  $M = 4$  le tracé de contour iso-niveau de l'hyper-surface est une surface 3D.

Les propriétés des hyper-surfaces NURBS s'appliquent même dans le cas le plus général. La propriété de l'enveloppe fortement convexe et la différentiabilité approfondie pour les courbes et les surfaces NURBS peuvent être facilement transposées au cas des hyper-surfaces les plus générales. De plus, compte tenu de l'hyper-réseau constitué par les composantes des vecteurs  $N$  knot, la propriété de support local s'applique aussi aux hyper-surfaces NURBS selon l'éq. (2.28).

## 2.5 Conclusions sur la théorie des entités NURBS

Dans ce chapitre, les principes fondamentaux des courbes, surfaces et hyper-surfaces NURBS et B-Spline ont été fournis. Bien que ces entités soient conçues pour une application immédiate dans l'environnement CAO, le contexte mathématique est très général et différentes applications peuvent être envisagées. Les entités géométriques B-Spline et NURBS ont des propriétés très générales et intéressantes, qui peuvent être exploitées dans d'autres applications, différentes de celles proposées dans [68]. En effet, plusieurs auteurs ont profité du large éventail de possibilités offertes par le NURBS pour résoudre de nombreux problèmes de nature différente, comme cela sera remarqué dans les prochains chapitres.



# Chapter 3

## Optimisation Methods and Algorithms

### 3.1 Introduction to Optimisation Methods

Optimisation methods and algorithms have been rising the interest of researchers and companies from several decades. Often, the word “optimisation” is excessively and improperly used in different contexts. Optimisation has been exploited in almost all the significant developments related to this Ph.D. thesis. In this Chapter, the main features of optimisation are discussed. Of course, the word “optimisation” is intended in the sense of mathematical programming throughout this manuscript. Roughly speaking, *optimisation* can be defined as the selection of a best element (with regard to some criterion) from a set of available alternatives [69]. Before proceeding with the discussion, some main concepts are introduced.

The aim of optimisation is to minimise an assigned *objective function*  $f$  depending on several (sometime a huge amount of) parameters. Some of these parameters can be set to a constant. Instead, other parameters have a significant impact on the objective function. These parameters are called *optimisation* or *design variables* and are collected in the array  $\mathbf{x} \in \mathbb{R}^n$ . The design variables constitute the unknowns of the optimisation problem, so the dependence of  $f$  on  $\mathbf{x}$  is made clearer by the notation  $f(\mathbf{x})$ . In practical engineering problems, it is very rare that an optimisation problem is characterised by only an objective function and the related design variables. Usually, the problem formulation is enhanced with  $m_e$  *equality constraints*, in the form of  $h_i(\mathbf{x}) = 0$ ,  $i = 1, \dots, m_e$ , and/or  $m_i$  *inequality constraints*, in the form  $g_j(\mathbf{x}) \leq 0$ ,  $j = 1, \dots, m_i$ . The optimisation constraints formalise some physical or technical requirements, which depend upon the specificity of the problem at hand. A point  $\mathbf{x}^*$  meeting all equality and inequality constraints is called *feasible*. The choice of the mathematical form of the objective/constraints functions and of the design variables is called *modeling*.

From a mathematical viewpoint, the optimisation problem is classically stated in the form of a constrained minimisation problem as follows:

$$\begin{aligned} & \min_{\mathbf{x}} f(\mathbf{x}), \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m_i, \\ h_j(\mathbf{x}) = 0, \quad j = 1, \dots, m_e, \\ \mathbf{x}_{\text{LB}} \leq \mathbf{x} \leq \mathbf{x}_{\text{UB}}. \end{array} \right. \end{aligned} \quad (3.1)$$

In Eq. (3.1),  $\mathbf{x}_{\text{LB}}$  and  $\mathbf{x}_{\text{UB}}$  are the vectors of lower and upper bounds, respectively. Problem (3.1) is conventionally referred as a *Constrained Non-Linear Programming Problem* (CNLPP).

Usually, the solution of problem (3.1) cannot be derived in a closed form (apart some very special cases) and a suitable optimisation algorithm must be used in order to carry out the solution search. The choice of a pertinent optimisation algorithm is influenced by several factors:

- the nature of design variables;
- the presence of constraint functions;
- the nature of both objective/constraint functions (i.e. continuity, convexity, linearity, etc).

Accordingly, optimisation algorithms can be classified taking into account the previous criteria: a possible classification, inspired by [70], is the one proposed in Fig. 3.1.

The first classification criterion is the presence of optimisation constraints. An example of unconstrained minimisation problems could be the least square problem in the framework of curve fitting. Although, usually, engineering problems involve several constraints, unconstrained optimisation theory is basic because constrained problems are typically solved by reducing them to equivalent unconstrained ones. Secondly, the linearity is considered: if objective and constraints functions are linear functions of the design variables  $\mathbf{x}$ , then the problem is considered linear. Linear Programming constitutes an important class of optimisation problems. Their peculiarity is that the only information that can be exploited is related, at most, to the gradient of objective/constraint functions (in fact, the Hessian matrix is identically null). Then, another basic property to take into account is the convexity. Convexity is extremely important in the optimisation framework because it brings information about the nature of the optimum solution that the algorithm is searching for. The notion of convexity is known in the case of sets and functions. An optimisation problem is *convex* if all the following conditions are met:

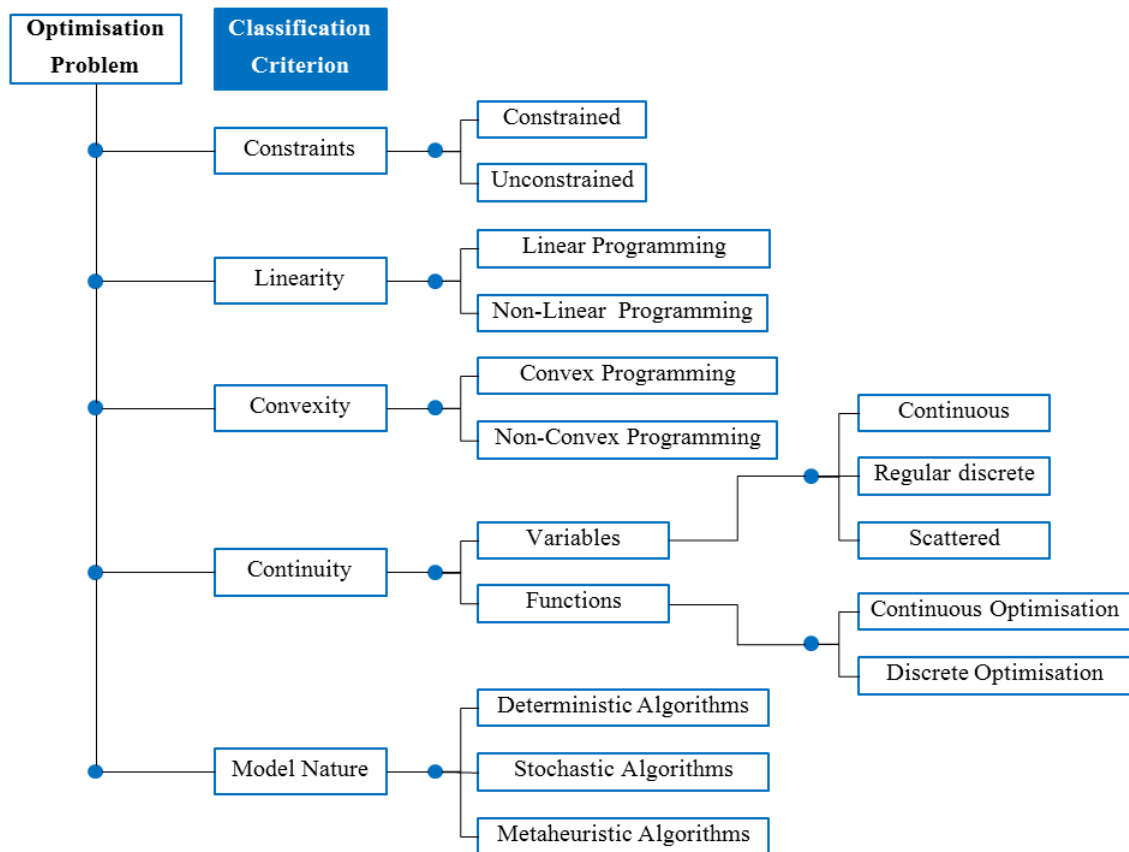


Figure 3.1 – Classification of optimisation algorithm.

- the objective function is convex;
- the equality constraint functions are linear;
- the inequality constraint functions are convex (i.e.  $g_j(\mathbf{x})$  is convex  $\forall j = 1, \dots, m_i$ ).

A point  $\mathbf{x}^*$  is a *global minimiser* if  $f(\mathbf{x}^*) \leq f(\mathbf{x})$  for all  $\mathbf{x}$  in the feasible domain. Accordingly,  $f(\mathbf{x}^*)$  is the global minimum or global optimum or global solution. Alternatively, if  $f(\mathbf{x}^*) \leq f(\mathbf{x})$  only in a neighbourhood of  $\mathbf{x}^*$ ,  $\mathbf{x}^*$  is called *local minimiser* and  $f(\mathbf{x}^*)$  is the local minimum (local optimum, local solution). It can be shown that when  $f$  is convex, any local minimiser is a global minimiser and, in addition, if  $f$  is differentiable, then any stationary point  $\mathbf{x}^*$  is a global minimiser [70]. On the other hand, it is highlighted that the most of real-world engineering problems are intrinsically non-convex. Therefore, it is not surprising that searching the global solution is prohibitive for several problems from a computational viewpoint. Another property affecting the nature of the optimisation problem is the continuity. An optimisation can be defined continuous only when



objective/constraint functions are continuous and, meanwhile, the set of design variables is continuous in  $\mathbb{R}^n$  (or, at least, in a subset of  $\mathbb{R}^n$ ). The discontinuous nature of a problem can come from the functions or from the variables set. If the variables are discrete (regularly or scattered, i.e. without a fixed step length), suitable algorithms should be employed to solve the related optimisation problem.

Finally, the information available about the model are important in order to determine which algorithm is the most suited for solving the optimisation problem. When the model is completely known and the optimisation variables as well as objective and constraint functions are continuous, deterministic algorithms are the best solution. However, it could happen that some quantities are characterised by uncertainty. In this case, stochastic algorithms are the most suited. A special class of algorithms, deserving a particular attention, is referred as meta-heuristics [71]. The main feature of these algorithms is the insertion of a random component to perform the solution search for an optimisation problem. The randomness allows for an efficient exploration of the computational domain and, therefore, meta-heuristics are particularly recommended in case of non-convex, multiple minima optimisation problems. Moreover, meta-heuristics are the only possible solution when the optimisation problem is characterised by discontinuous design variables and/or objective and constraint functions. Nevertheless, an exhaustive overview on all the variants of optimisation algorithms is out of the scopes of this Chapter.

Unfortunately, problems faced in this manuscript are mostly non-convex. To deal with these problems, both deterministic and meta-heuristic algorithms have been used. In the following sections, some basic information about the adopted deterministic and meta-heuristics algorithms is given. The main deterministic algorithms for solving CNLPP will be briefly recalled with a particular attention on the related advantages and drawbacks. As far as meta-heuristics are concerned, the discussion will focus on a particular Genetic Algorithm (GA), whose capabilities have been already shown in literature (refer to [72]).

## 3.2 Deterministic Methods for CNLPP

### 3.2.1 Generalities on Deterministic Methods

Deterministic methods exploit the available information about the model to update the design variables array from  $\mathbf{x}_k$  to  $\mathbf{x}_{k+1}$ , where  $k$  and  $k + 1$  indicate the current iteration. They constitute a really well-established domain in literature (refer, for instance, to [73] and [74]). In the following, the expression “deterministic” and “gradient-based” can be confused. This fact means that the basic information for updating the design variables relies on the gradient operator.

Before discussing the main features of constrained optimisation, a brief note on unconstrained optimisation is needed. When dealing with unconstrained optimisation, there are essentially two methods in literature for generating the iterates  $\mathbf{x}_k$ . They are *line search*

methods and *trust region* methods [70].

- *Line search methods.* Firstly a suitable direction  $\mathbf{p}_k$  is chosen as a *descent direction* for the objective function  $f$ . Secondly, the following mono-dimensional simpler optimisation problem is solved

$$\min_{\alpha > 0} f(\mathbf{x}_k + \alpha \mathbf{p}_k). \quad (3.2)$$

The different algorithms available in literature mainly differ because of the different choice of the descent direction.

The steepest descent direction  $\mathbf{p}_k = -\nabla f(\mathbf{x}_k)$  seems to be the most natural choice because it is the direction that maximise, at each iteration, the reduction of the objective function. In spite of only information about the gradient is required, performing the line search through a steepest descent direction could be extremely slow for complicated problems.

Albeit it is implicit, the minimiser of the objective function can be attained by choosing any direction making an angle strictly less than  $\pi/2$  radians with  $-\nabla f(\mathbf{x}_k)$ . Therefore, another possibility is the Newton direction, that is particularly effective when the objective function is not too far from its quadratic approximation in a neighbourhood of  $\mathbf{x}_k$ . It can be shown that, for the Newton direction,  $\mathbf{p}_k = -\mathbf{H}_k^{-1}\nabla f(\mathbf{x}_k)$ , where  $\mathbf{H}_k = \nabla^2 f(\mathbf{x}_k)$  is the Hessian matrix at iteration  $k$ . Choosing the Newton direction guarantees for a quadratic convergence rate but the knowledge of the Hessian matrix is required at each iteration.

An interesting alternative to the Newton direction is the quasi-Newton direction: here, the Hessian matrix is not explicitly computed and a suitable approximation is exploited:  $\mathbf{p}_k = -\mathbf{B}_k^{-1}\nabla f(\mathbf{x}_k)$ , with  $\mathbf{B}_k$  the approximation of the Hessian matrix, generally given by the BFGS formula (Broyden-Fletcher-Goldfarb-Shanno). It can be shown that the convergence rate of a quasi-Newton algorithm is not quadratic but superlinear.

The last searching direction derives from the *non-linear conjugate gradient method*. The current search direction  $\mathbf{p}_k$  can be easily computed by knowing the searching direction at the previous step  $\mathbf{p}_{k-1}$  and by exploiting the idea of conjugate direction. Let consider a set of directions  $\{\mathbf{p}_1, \dots, \mathbf{p}_l\}$  and a symmetric positive definite matrix  $\mathbf{A}$ . Then, the directions are conjugate if  $\mathbf{p}_i^T \mathbf{A} \mathbf{p}_j = 0, \forall i \neq j$ . In such a framework, it can be shown that conjugate gradient-methods can converge in at most  $n$  steps, where  $n$  is the dimension of the design variables array. The rate of convergence is not as fast as Newton or quasi-Newton methods but it does not require storing big matrices.

- *Trust-region methods.* In trust-region methods, firstly the step length is chosen, then the searching direction is calculated. Since the search direction is unknown at

the beginning of the iteration, it is preferable to speak about ball radius (or trust-region radius)  $\Delta_k$  instead of step length. A quadratic approximation of the original problem is solved within the trust region, in order to give the following iterate. Provided a suitable technique for updating  $\Delta_k$  during iterations, the possibilities for the searching direction are the same of the aforementioned line search algorithms (excepted for the conjugate gradient method, that has not an equivalent in the trust region framework). An important remark concerning trust-region algorithms is their sensitivity to poorly scaled models. A model is poorly scaled when the sensitivity of the objective function with respect to the design variables can significantly vary (even about orders of magnitude). Line search algorithms are, in general much more robust and they can intrinsically guarantee for *invariance*, i.e. non-sensitivity to poorly scaled models.

Although unconstrained problems seldom constitute a real interest in science and applications, algorithms developed for unconstrained optimisation are basic because their improved and enhanced version can be used to solve CNLPP, which are often reduced to an equivalent unconstrained problem [70]. This claim holds for both line search and trust-region algorithms.

### 3.2.2 Optimality Conditions for CNLPP

In the following, the two basic theorems for constrained optimisation theory are discussed. Given a CNLPP in the form of Eq. (3.1), the functional  $L$  can be defined as the Lagrangian of problem (3.1):

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{g}(\mathbf{x}) + \boldsymbol{\mu}^T \mathbf{h}(\mathbf{x}), \quad (3.3)$$

where  $\boldsymbol{\lambda}$  and  $\boldsymbol{\mu}$  are the arrays of Lagrange multipliers for inequality constraints ( $\lambda_i \geq 0, \forall i = 1, \dots, m_i$ ) and equality constraints ( $\mu_j, \forall j = 1, \dots, m_e$ ), respectively.

#### Theorem 3.2.1 *First-order necessary conditions.*

*Assume that*

1.  $\mathbf{x}^*$  is a local solution of problem (3.1);
2. the functions  $f, g_i$  and  $h_j$  are continuously differentiable;
3. the LICQ (Linear Independence Constraint Qualification) condition holds at  $\mathbf{x}^*$ , i.e. the gradients of each equality and inequality constraint function must be linearly independent.

Then, two Lagrange multipliers arrays  $\boldsymbol{\lambda}^* \geq \mathbf{0}$  and  $\boldsymbol{\mu}^*$  exist and the following conditions, known as Karush-Kuhn-Tucker (KKT) conditions, are met:

$$\begin{cases} \nabla_x L(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \mathbf{0}, \\ \lambda_i^* g_i(\mathbf{x}^*) = 0, \quad \forall i = 1, \dots, m_i, \\ h_j(\mathbf{x}^*) = 0, \quad \forall j = 1, \dots, m_e. \end{cases} \quad (3.4)$$

In Eq. (3.4),  $\nabla_x$  is the gradient operator with respect to the design variables  $\mathbf{x}$ . The point  $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$  is named KKT point.

**Theorem 3.2.2 *Second-order sufficient conditions.***

*Let the functions  $f$ ,  $g_i$  and  $h_j$  twice continuously differentiable. Suppose  $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$  is a KKT point and suppose the Hessian of the Lagrangian  $\nabla^2 L(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$  is positive definite. Then  $\mathbf{x}^*$  is a strict local solution of problem (3.1).*

For the proof of the aforementioned theorems, the reader is addressed to [70, 73–75].

### 3.2.3 Deterministic Algorithms for CNLPP

The deterministic optimisation strategies proposed in this manuscript rely on the well-known *fmincon* optimisation toolbox implemented into the MATLAB package [76]. In this framework, it is remarked that the *fmincon* trust-region algorithm is really effective but, meanwhile, subject to important limitations: it can be used only when the objective function gradient is known from analytical formulae and in case of either bounds or linear equality constraints (but not both). Because of these important shortcomings, the trust-region algorithm is not considered in this discussion. The other *fmincon* algorithms are line search algorithms:

- SQP - Sequential Quadratic Programming
- AS - Active Set
- IP - Interior-Point

#### SQP and AS Algorithm

The Sequential Quadratic Programming and the Active Set algorithms can be treated together. Indeed, AS is just a particular SQP method wherein constraints are handled in a more effective manner. Therefore, the main steps described here below hold for both SQP and AS. Some remarkable differences will be discussed later on. The main idea behind an SQP method consists of approximating the CNLPP at hand in a sequence of Quadratic Programming (QP) problems. The conditions to be met for a QP problem are

- the objective function  $f$  is a *quadratic function* of the design variables;
- equality ( $h_j$ ) and inequality ( $g_i$ ) constraints are *linear functions* of the design variables.

A solution can always be provided by using QP techniques or, at least, it can be proven that the solution does not exist. The disadvantage of such a method is that the computational burden depends on the objective function and on the number of optimisation constraints. An extensive amount of books and articles can be found in literature about QP [70, 77, 78].

A general overview of the SQP algorithm is provided in Fig. 3.2.

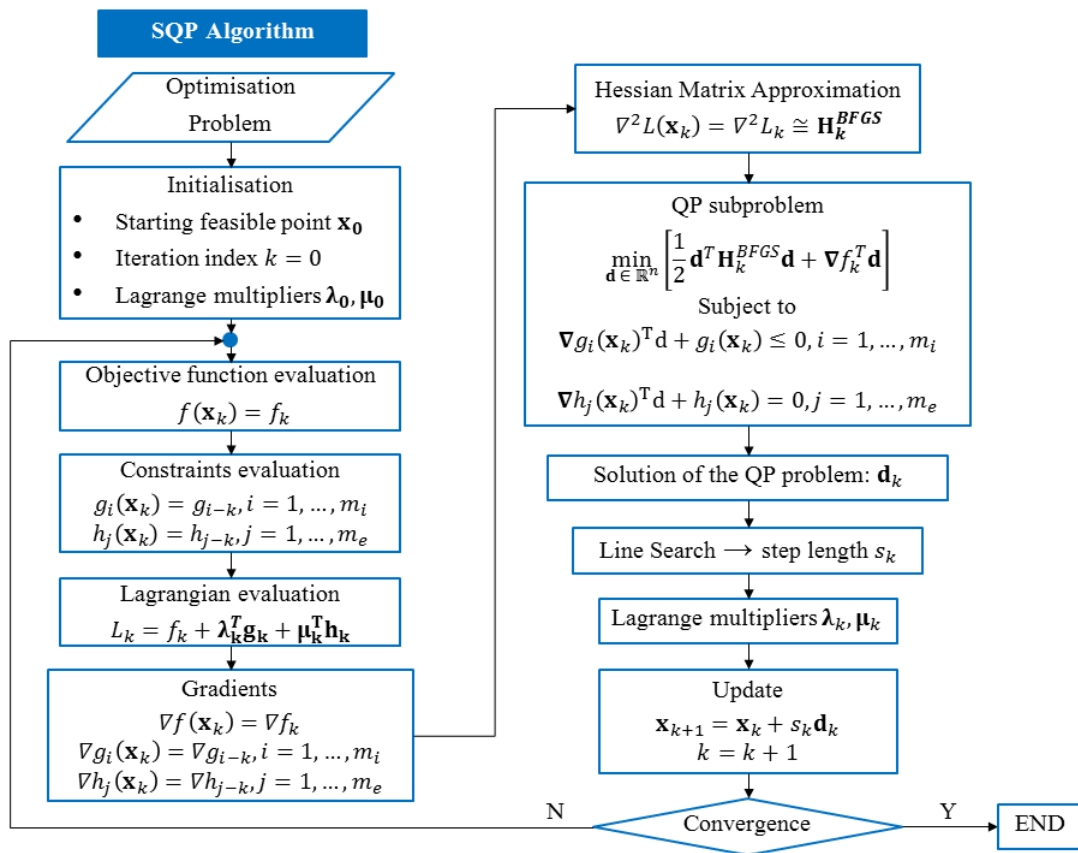


Figure 3.2 – Overview of the SQP algorithm.

Once the CNLPP has been stated in the form of Eq. (3.1), a suitable initial guess for the design variables array should be provided. Meanwhile, the Lagrange multipliers are initialised and the iteration index  $k$  is set to 0. Then, the Lagrangian form is evaluated through Eq. (3.3). The gradients of objective and constraint functions are needed for

the following steps: they can be analytically provided or numerically evaluated (e.g. through a finite-difference derivative approximation scheme). Hence, the Hessian matrix of the Lagrangian is suitably approximated and it is used, together with the previously computed gradients, to set up the local QP subproblem:

$$\begin{aligned} \min_{\mathbf{d}} Q_k(\mathbf{d}) &= \min_{\mathbf{d}} \frac{1}{2} \mathbf{d}^T \mathbf{H}^{\text{BFGS}}_k \mathbf{d} + \nabla f_k^T \mathbf{d}, \\ &\text{subject to:} \\ &\overline{\mathbf{A}}_k \mathbf{d} \leq \mathbf{b}_k. \end{aligned} \tag{3.5}$$

In Eq. (3.5), it is assumed that the approximation of the Hessian matrix has been carried out thanks to the *BFGS* formula. The array  $\mathbf{d}$  constitutes the array of design variables for the subproblem and it is interpreted as the search direction for the  $k$ -th iteration of the SQP algorithm. It is noteworthy that, being subproblem (3.5) a QP problem, optimisation constraints should be in linearised form. Therefore, the optimisation constraints of the original problem (3.1) are linearised, as shown in Fig. 3.2, and their gradients are collected in the matrix  $\overline{\mathbf{A}}_k$ . Without providing details, it is highlighted that the assembly of the matrix  $\overline{\mathbf{A}}_k$  distinguishes the SQP method and the AS method. Particularly, in an AS context, only those inequality constraints that are violated give a contribution to the matrix  $\overline{\mathbf{A}}_k$  (equality constraints are always included). Analogously, the coefficients of the constraints first-order approximation are gathered in the array  $\mathbf{b}_k$ . The QP subproblem (3.5) is solved thanks to standard techniques [70]. In an AS framework, some internal iterations could be needed to check if the active-set of optimisation constraints has been correctly evaluated and, eventually, the matrix  $\overline{\mathbf{A}}_k$  is updated. As previously remarked, the solution is the line search direction  $\mathbf{d}_k$  and now it is possible to look for the correct step  $s_k$  along this descent direction. Lagrange multipliers can be updated now, as well as the design variables array  $\mathbf{x}_{k+1}$ . Several convergence criteria are considered in the MATLAB implementation of the SQP/AS algorithms [76]:

- *Maximum number of iterations:*  $k + 1 = K_{max}$ ;
- *Small Objective function improvement:*  $|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)| < \sigma_f$ , with  $0 < \sigma_f \ll 1$ ;
- *Negligible change of variables values:*  $|\mathbf{x}_{k+1} - \mathbf{x}_k| < \sigma_x$ , with  $0 < \sigma_x \ll 1$ ;
- *Norm of the Lagrange function gradient close to 0:*  $|\nabla L(\mathbf{x}_{k+1})| < \sigma_{\nabla}$ , with  $0 < \sigma_{\nabla} \ll 1$ .

If one among the convergence criteria is met, the algorithm stops. Of course, the latter set of criteria makes sense only if the objective function and the imposed constraints are dimensionless. For a deeper insight into SQP algorithms, the reader is addressed to [70, 76].

When compared to SQP methods, the AS algorithm can tolerate some iterative steps out of the feasible region in solving constrained optimisation problems. This fact allows for an efficient exploration of the feasible domain (especially its boundary) in CNLPPs.

### IP Algorithm

IP algorithms are the most efficient algorithms for solving CNLPP together with SQP methods. Generally speaking, it can be asserted that in some conditions IP algorithms could be more efficient than SQP ones and converge in less iterations. However, IP algorithms are slightly less robust than SQP methods, especially in exploring the boundary of the feasible domain.

The main idea consists of changing the CNLPP (3.1) into a sequence of simpler optimisation problems, wherein only equality constraints are taken into account (as illustrated in Fig. 3.3).

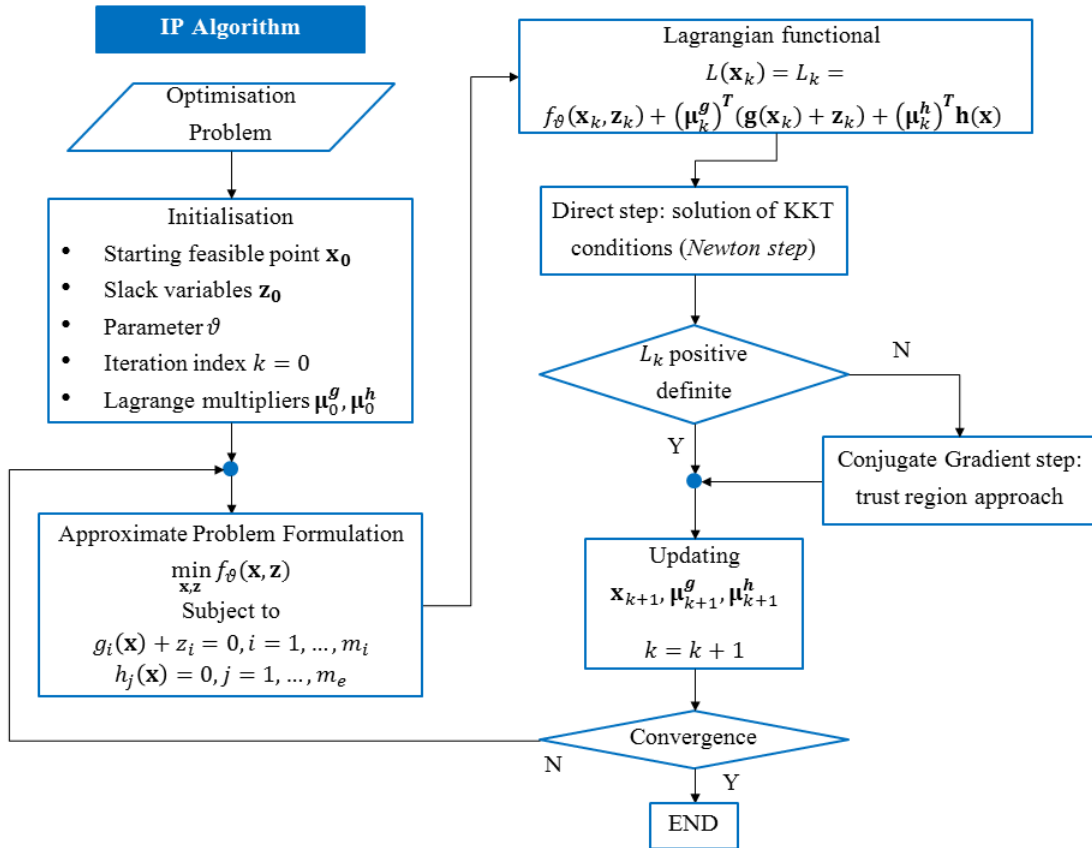


Figure 3.3 – Overview of the IP algorithm.

Considering a certain iterate index  $k$ , and a scalar positive quantity  $\vartheta$  the correspond-

ing optimisation problem is

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{z}} f_{\vartheta}(\mathbf{x}, \mathbf{z}) &= \min_{\mathbf{x}, \mathbf{z}} f(\mathbf{x}) - \vartheta \sum_{i=1}^{m_i} \ln(z_i) \\ &\text{subject to:} \\ h_j(\mathbf{x}) &= 0, \quad j = 1, \dots, m_e, \\ g_i(\mathbf{x}) + z_i &= 0, \quad i = 1, \dots, m_i. \end{aligned} \tag{3.6}$$

Often the IP method is referred as “barrier method” because of the logarithmic form appearing in problem (3.6). The positive quantities  $z_i$  are usually labelled as *slack* variables: there are as many slack variables as inequality constraints of the original problem. The solution of the approximated problem is provided as follows. As a first attempt, the KKT conditions are imposed on a linearised approximation of the problem at hand (this phase is referred as *Newton step*). When the related matrix is factorised, it is possible to check if the Hessian of the Lagrangian functional is positive defined. If it is positive defined, the Newton step can be carried out and the design variables together with the Lagrange multipliers vectors can be updated. Otherwise, in order to produce a new iterate for the algorithm, a Conjugate Gradient step is necessary: here, a quadratic approximation of the approximate problem is solved in the trust region framework. In any case, the new iterate  $\mathbf{x}_{k+1}$  is given and the convergence criteria can be checked.

The IP method tends to keep the iterations far from the boundaries and suitable strategies must be put into practice to explore the boundary defined by the inequality constraints.

## 3.3 Meta-heuristic Methods for CNLPP

### 3.3.1 Generalities on Meta-heuristics

Meta-heuristics can be defined as “global” optimisation methods for non-convex CNLPP making use of several empirical rules, which are inspired by a precise natural phenomenon. The word “global” must be interpreted in the sense that a meta-heuristic allows for a better exploration of the domain if compared to deterministic methods, because it acts on a population of points within the design domain rather than on a single point. However, a meta-heuristic *can* find the global optimum but there is no guarantee to actually find it.

Many works concerning meta-heuristics have been inspired by the concept of *evolution* in bibliography [79–81]. The common principle consists in establishing the optimisation variables of the problem at hand, providing a *population* of candidate solutions and making this population evolve towards an improvement of the objective function. Particular interest has been risen by Holland’s Genetic Algorithm (GA) [82, 83]. GAs are part of a special class of algorithms referred as *artificial intelligence*: initially, the aleatory



component is strong in order to explore the design domain, then progressively the stored information is exploited to improve performances. Other widespread meta-heuristics, known as *swarm intelligence*, do not reproduce the evolution mechanisms, rather they mimic the behaviour of groups of individuals in nature (as birds or ants). For instance, in the Ant Colony Optimisation method [84], many ants (or agents) are used to pass through the solution space and to find out locally productive areas; in Particle Swarm Optimisation [85], a population (swarm) of candidate solutions (particles) moves in the search space and the movement of the particles is affected both by their own best-known position and by the swarm global best-known position.

It is evident that the domain of meta-heuristic methods for mathematical programming is a domain of outstanding vastness. Here, the discussion is restrained on GAs and, more particularly, on a special GA, firstly proposed in [86, 87] and further developed in [72]. A brief overview on such an algorithm, with particular emphasis on its enhanced capabilities when compared to standard GAs, is provided in section 3.3.2. In this Ph.D. thesis, the algorithm has been modified to properly deal with problems proposed in Chapter 7. A description of these modifications can be found in section 3.3.3.

### 3.3.2 The Genetic Algorithm BIANCA

Before proceeding, it is opportune to introduce the standard terminology of GAs. As previously stated, the basic idea of GAs is making a *population* evolve according to the survival capability of its *individuals*. An individual is a collection of design variables values, i.e. a candidate solution for the problem at hand. The population evolution is obtained by simulating the reproduction among individuals and the features transmission mechanisms, which are typical of genetics. The genetic heritage of two individuals is combined by means of suitable operations involving *chromosomes* and *genes*. An individual is composed of chromosomes, and each chromosome is composed, on its turn, of genes. It is pointed out that the genetic operations are carried out on a binary encoded version of the design variables, usually referred as *genotype*. The corresponding set of decoded values of design variables is called *phenotype*. The encoding is performed by discretising the interval of definition of each variable. Consider the generic variable  $x \in [x_{\text{LB}}, x_{\text{UB}}] \subset \mathbb{R}$ : if a discretisation step  $\Delta x$  is assigned, the possible values that the variable  $x$  can take are

$$x_I^d = x_{\text{LB}} + (I - 1)\Delta x, \quad (3.7)$$

where  $I$  can be whatever component of the array  $\mathbf{I} = \{1, 2, \dots, n_{\text{var}}\}$ , with

$$n_{\text{var}} = 1 + \frac{x_{\text{UB}} - x_{\text{LB}}}{\Delta x}. \quad (3.8)$$

In this framework, there is a bijective relationship between the components of the array  $\mathbf{x}^d$  and those of the component of the array  $\mathbf{I}$ . The binary encoding is therefore performed

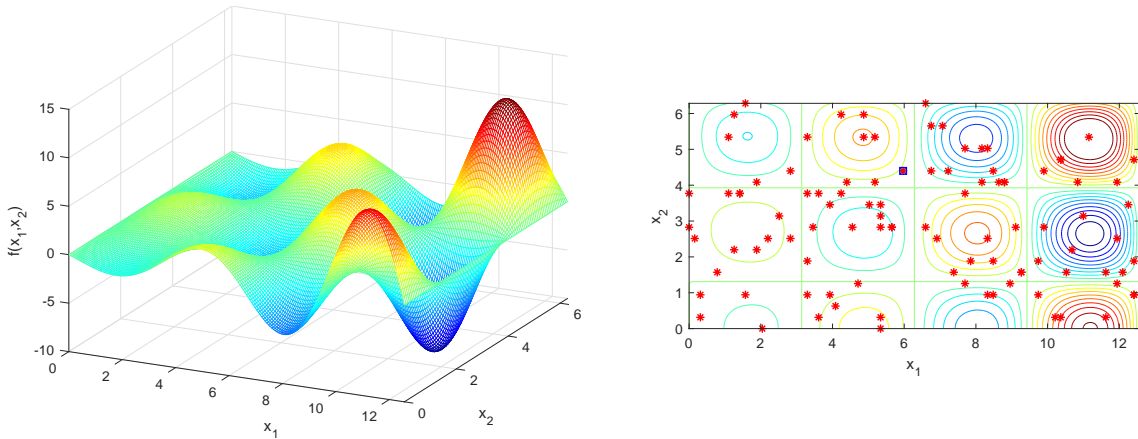
### 3.3. Meta-heuristic Methods for CNLPP

on the components of the array  $\mathbf{I}$ .

Let consider a simple example to make clearer these points. The following CNLPP has to be solved:

$$\begin{aligned} \min_{x_1, x_2} f(x_1, x_2) &= -e^{ka\sqrt{x_1^2+x_2^2}} \sin(ax_1) \cos(2bx_2), \\ &\text{subject to:} \\ &\begin{cases} 0 \leq x_1 \leq 4\pi, \\ 0 \leq x_2 \leq 2\pi, \end{cases} \end{aligned} \quad (3.9)$$

with  $k = 0.2$ ,  $a = 1$  and  $b = 0.6$ . The function  $f$  has been represented in Fig. 3.4a. Fig. 3.4b represents some iso-lines of the function  $f$  and all the individuals (the red stars) constituting the first generation. For this problem, a number of individuals equal to 100 ( $N_{ind} = 100$ ) has been chosen, whilst the discretisation steps are  $\Delta x_1 = \pi/20$  and  $\Delta x_2 = \pi/10$ . It is evident that one individual is just a point in the design domain. In this particular case, each individual has one chromosome and two genes; each gene is dedicated to the encoding of one variable.



(a) Representation of the function  $f$  of problem (3.9).

(b) Population at generation 0.

Figure 3.4 – GAs: an example.

Let consider the individual represented through a square in Fig. 3.4b: its phenotype is constituted by the values taken by the design variables, i.e.  $x_1 = 5.9690$  and  $x_2 = 4.3982$ . These values uniquely correspond to the indices  $I_1 = 39$  and  $I_2 = 15$ . Thus, the genotype is obtained by translating the indices values in binary code, i.e. 0100111 and 01111 (as shown in Fig. 3.5). It is noteworthy that the number of bits should be always equal to the minimum number of digits needed to translate the maximum index, that is 81 for  $x_1$  and 21 for  $x_2$ : hence, 7 and 5 digits are needed to encode the  $x_1$  values and the  $x_2$  values, respectively.

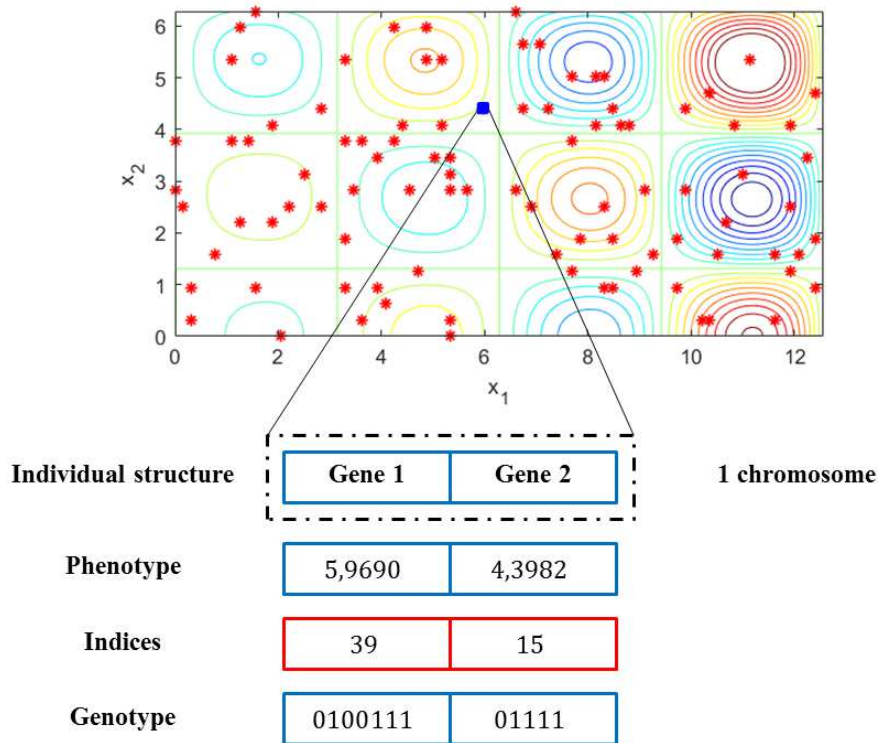


Figure 3.5 – Individual Structure for problem (3.9).

Once the phenotype has been evaluated and translated into genotype for all the individuals, the reproduction phase can start: two *parents* are replaced by two *children* and the population can evolve.

In this framework, the attention is here focused on the GA called *BIANCA* (Biologically Inspired ANalysis of Composite Assemblages). The most important feature of *BIANCA* is the capability of dealing with *engineering modular systems*. A modular system in structural engineering is whatever application constituted of several elementary and repetitive unit entities (*the modules*) which have certain intrinsic parameters. A typical example of a modular system is a composite laminate, in which the module is represented by the single ply that has some intrinsic parameters such as the fibres orientation angle, thickness, material properties, etc. It is noteworthy that a complete optimisation of modular systems has been considered as a prohibitive problem for decades. The most important challenge is related to the fact that, when optimising a modular system, there is no criterion to *a priori* define the optimum number of modules. Therefore, the number of modules should be included among the optimisation variables together with the intrinsic parameters of each module. The striking impediment is the definition of an optimisation problem defined over a space with non-constant dimension. Roughly speaking, the

problem is characterised by a “variable number of optimisation variables”.

For a modular system, the optimisation problem can be stated as

$$\begin{aligned}
 & \min_{\mathbf{x}} f(\mathbf{x}), \\
 & \text{subject to:} \\
 & \left\{ \begin{array}{l} g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m_i(n_c), \\ h_j(\mathbf{x}) = 0, \quad j = 1, \dots, m_e(n_c), \\ \mathbf{x}_{\text{LB}} \leq \mathbf{x} \leq \mathbf{x}_{\text{UB}}, \\ \mathbf{x}_{\text{LB}}, \mathbf{x}, \mathbf{x}_{\text{UB}} \in \mathbb{R}^{n(n_c)}, \end{array} \right. \quad (3.10)
 \end{aligned}$$

where the dependence of the number of design variables and of the number of constraints on the number of modules  $n_c$  has been made explicit. The subscript  $c$  is used because it seems natural in optimising modular systems to identify each module with a chromosome, whilst the intrinsic parameters are associated to the genes of each module. For more details on this topic the interested reader is addressed to [72].

Albeit problem (3.9) is very simple, it is representative of a wide class of problems addressed in literature, i.e. mono-chromosome problems. An example of more challenging problem in the form of Eq. (3.10) is the optimisation of a laminated plate, as shown in Fig. 3.6: in this problem, the material properties of the fibre phase and of the matrix are established and the optimisation variables are the orientation angle of the fibres and the thickness related to each ply. Of course, the number of plies is an optimisation variable as well. In the framework of application of GAs to modular systems, the concept of *species* comes out in a very natural way: individuals with the same number of chromosomes (modules) belong to the same specie. The latest version of BIANCA has been explicitly conceived to deal with this kind of problems [72], including the reproduction among individuals belonging to different species. This means that suitable enhancements have been necessary to cope with intrinsic difficulties characterising the optimisation of modular systems. A brief description of the structure of the algorithm BIANCA with a particular attention to its distinguishing features with respect to other standard GAs is provided in the here below. A summarising scheme of the algorithm BIANCA is shown in Fig. 3.7.

**Starting Population.** The starting population of individuals is generated by randomly selecting the integer indices associated to the variables, as it has been shown in the previous example (3.9). Once the encoding has been performed, it will be used in next steps for genetic operations. Before proceeding, the evaluation of both objective and constraint functions is required. Of course, they depend on the mathematical statement of the problem at hand. It is noteworthy that a general constraints handling is not a trivial task. In this algorithm a smart *Automatic Dynamic Penalisation* (ADP) strategy is used [7, 72, 88]. As in other penalisation methods, the CNLPP (3.1) is changed into an

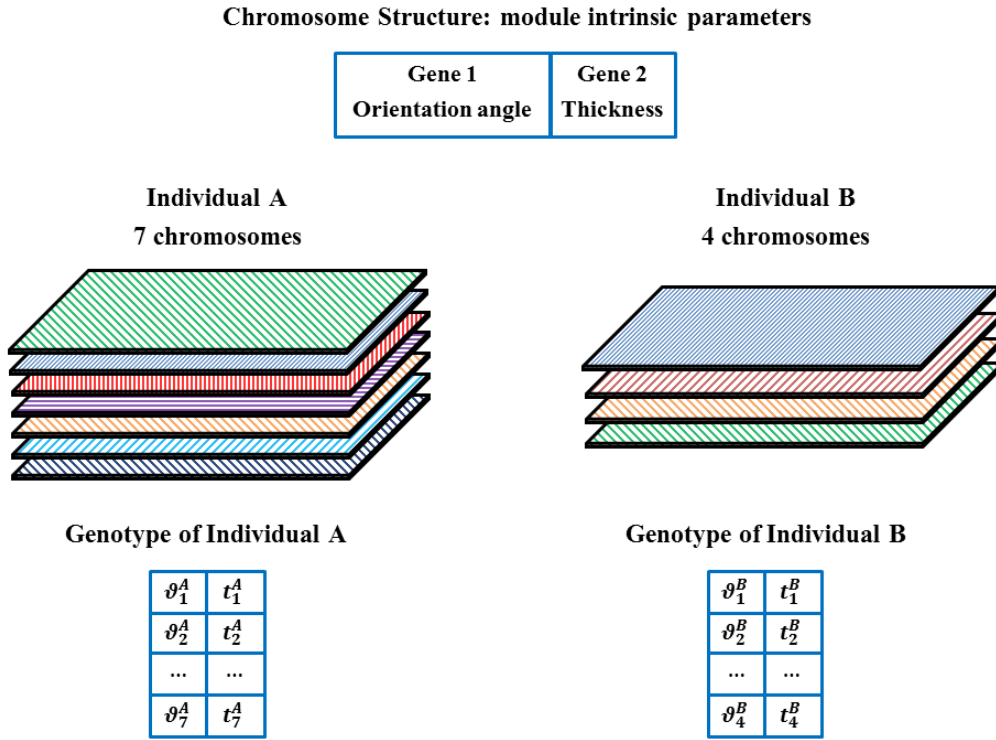


Figure 3.6 – Example of optimisation of modular system.

unconstrained one:

$$\begin{aligned}
 & \min_{\mathbf{x}} f_P(\mathbf{x}) , \\
 & \text{where :} \\
 f_P(\mathbf{x}) = & \begin{cases} f(\mathbf{x}) & \text{if } g_i(\mathbf{x}) \leq 0 \text{ and } h_j(\mathbf{x}) = 0, \\ f(\mathbf{x}) + \sum_{i=1}^{m_i} c_i G_i(\mathbf{x}) & \text{if } g_i(\mathbf{x}) > 0 \text{ and } h_j(\mathbf{x}) = 0, \\ f(\mathbf{x}) + \sum_{j=1}^{m_e} q_j H_j(\mathbf{x}) & \text{if } g_i(\mathbf{x}) \leq 0 \text{ and } h_j(\mathbf{x}) \neq 0, \\ f(\mathbf{x}) + \sum_{i=1}^{m_i} c_i G_i(\mathbf{x}) + \sum_{j=1}^{m_e} q_j H_j(\mathbf{x}) & \text{if } g_i(\mathbf{x}) > 0 \text{ and } h_j(\mathbf{x}) \neq 0, \end{cases} \quad (3.11) \\
 & i = 1, \dots, m_i, \quad j = 1, \dots, m_e,
 \end{aligned}$$

In Eq. (3.11),  $G_i(\mathbf{x}) = \max[0, g_i(\mathbf{x})]$ ,  $H_j(\mathbf{x}) = \max[0, |h_j(\mathbf{x})| - \varepsilon]$ , with  $\varepsilon \ll 1$ , and  $f_P(\mathbf{x})$  is the penalised objective function. The power of the ADP strategy implemented

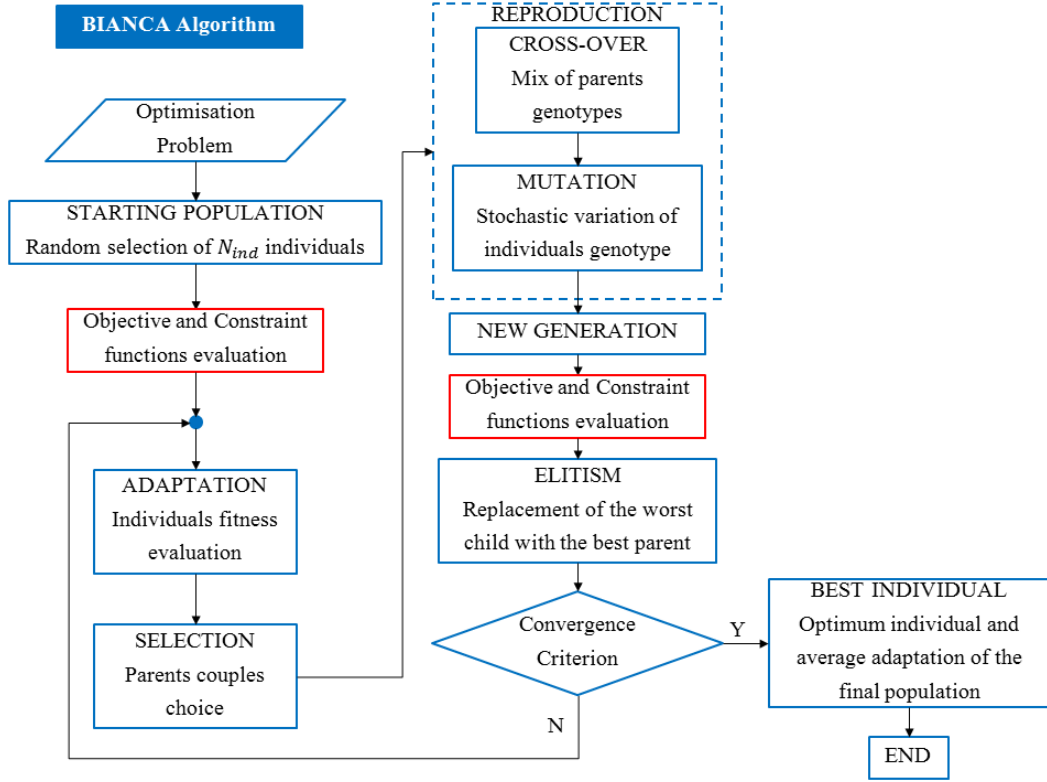


Figure 3.7 – The GA BIANCA.

in BIANCA consists of automatically and adaptively updating the coefficients  $c_i$  and  $q_j$  in such a way that the penalisation is not too low or too high, assuring a good trade-off between preventing infeasible solutions and an efficient exploration of the feasible domain boundary. The ADP formulae are applied at each iteration without the user intervention and considering the individuals distribution (feasible and infeasible):

$$\begin{cases} c_i = \frac{|f_{best}^F - f_{best}^{NF}|}{(G_i)_{best}^{NF}}, & i = 0, \dots, m_i \\ q_j = \frac{|f_{best}^F - f_{best}^{NF}|}{(H_j)_{best}^{NF}}, & j = 0, \dots, m_e. \end{cases} \quad (3.12)$$

In Eq. (3.12),  $f_{best}^F$  and  $f_{best}^{NF}$  are the objective function evaluation of the best individual of the feasible domain and of the infeasible one, respectively. Considering now best individual within the non-feasible domain,  $(G_i)_{best}^{NF}$  and  $(H_j)_{best}^{NF}$  represent its violated inequality and equality constraints. The basic idea is to not penalize high-performance solutions, although they are infeasible, in order to make them act as point of attraction

for the solution. In this way, the exploration of the boundary of the feasible region is really improved. For a deeper insight into the ADP, the reader is addressed to [7, 72].

**Adaptation.** For each individual, a unique measure of its adaptation needs to be provided. In this phase, a suitable fitness function is defined: this is a scalar function that accepts the values of the objective function as input and returns a value between 0 and 1 (the fitness function takes the value 0 for the worst individual of the generation and 1 for the best one). Several choices are possible for the fitness function, see [72].

**Selection.** Individuals are selected in this phase in order to constitute the  $N_{ind}/2$  couples of parents for the reproduction phase. The basic concept is that the fitter individuals (i.e. those individuals corresponding to high values of the fitness function) have a high probability to be chosen for the reproduction phase. The selection criterion aims at mimicking this simple natural phenomenon: most adapted individuals (with respect to the surrounding environment) will live longer than less adapted ones, therefore they have more time and an increased probability to reproduce. In practice, the previously defined fitness function is employed to assign to each individual a probability of selection and, then, the effective selection is done through an *ad hoc* criterion (usually the roulette-wheel).

**Cross-over.** The cross-over aims at combining the genetic heritage of the two parents and it is the first step of the *Reproduction*, i.e. the phase providing new individuals who will constitute the new generation. The cross-over operator acts on homologous genes, which are randomly cut with a probability  $p_{cross}$ . Then the genes are combined as shown in Fig. 3.8, where two individuals of the first generation of problem (3.9) are considered.

When the optimisation of modular systems is faced, the standard cross-over of Fig. 3.8 is no longer possible and suitable techniques should be developed. Without providing details about the implementation, a synthetic scheme giving an intuitive idea of the cross-over among different species is given in Fig. 3.9.

In the example illustrated in Fig. 3.9, two individuals belonging to different species are considered: the first one is composed of three chromosomes and the second one has five chromosomes. Each chromosome is made of two genes coding the design variable related to the generic module. Firstly, the genes of the individual with the lowest number of modules are shifted. The shifting occurs with a probability  $p_{shift}$ . Once the genes are shifted, those genes which have a counterpart in the other individual undergo the standard cross-over operations described in Fig. 3.8. The other genes do not change. At the end of the cross-over, the genes of the individual having the least number of modules are shifted again in order to respect the initial formalism.

**Mutation.** Mutations can be interpreted as a *second-order adaptation mechanism*, aiming at improving the exploration capabilities of the algorithm and to avoid that the individuals are flawed by a local optimum solution at the beginning of the optimisation. Accordingly, the role of the mutation operator is to create biodiversity within the population at the generic iteration in order to avoid a premature convergence towards pseudo-optimal solutions. There two kinds of mutations in BIANCA. The first mutation

### 3.3. Meta-heuristic Methods for CNLPP

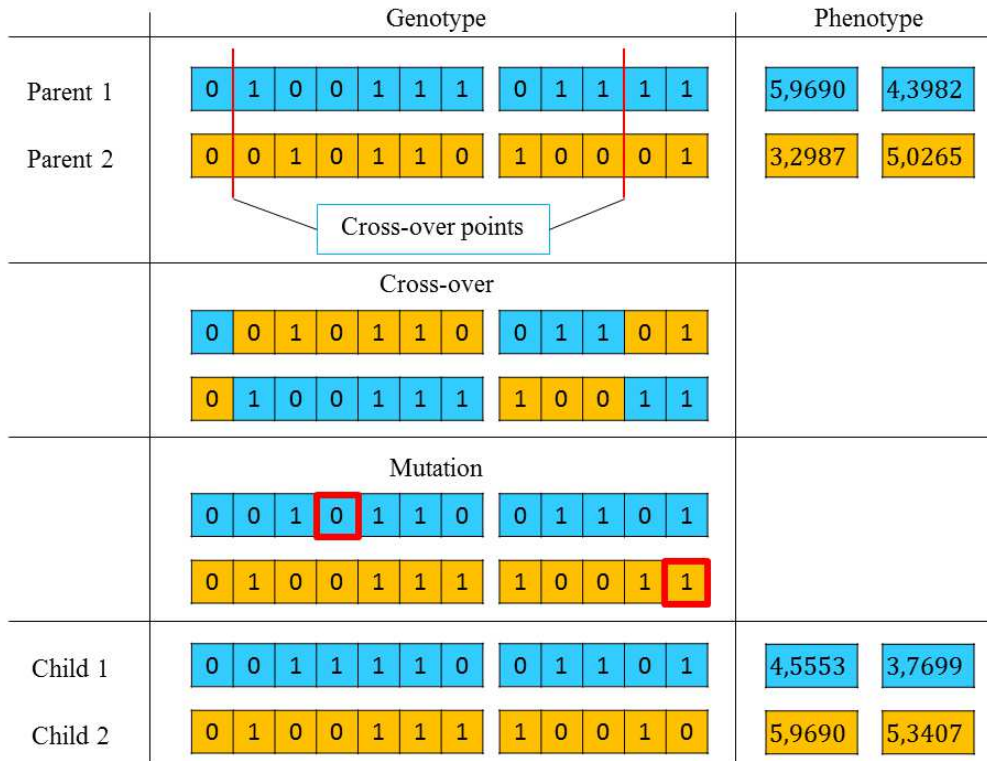


Figure 3.8 – Reproduction phase in a standard GA.

acts at the end of the standard cross-over phase. One location for the binary digit (also referred as *locus*) is randomly chosen and the digit can be modified with a probability  $p_m$ . The second mutation intervenes in the case of reproduction among individuals belonging to different species. At the end of the previously described reordering phase, one module (chromosome) can be deleted/added with a probability  $p_{mc}$ .

**New Generation and Elitism.** When all the new individuals are generated, the population can be completely updated (performances of each new individual are evaluated). However, in order to protect the good genetic heritage, the elitism operator guarantees that the worst individual of the offspring is replaced by the best individual of the ancestors. It is highlighted here that the ADP strategy for constraints handling always penalises non-feasible solutions more severely with respect to feasible ones (even if the latter have high values of the objective function). Thus, when the ADP is combined with the elitism and a feasible solution is found, then individuals of the next generation will necessarily provide an improved (decreased) value of the objective function or, at least, the same; in fact, if no individuals can improve the objective function, the worst of the current generation is replaced by the best of the previous one.

**Convergence Criterion and Best Individual.** The convergence criterion is typically



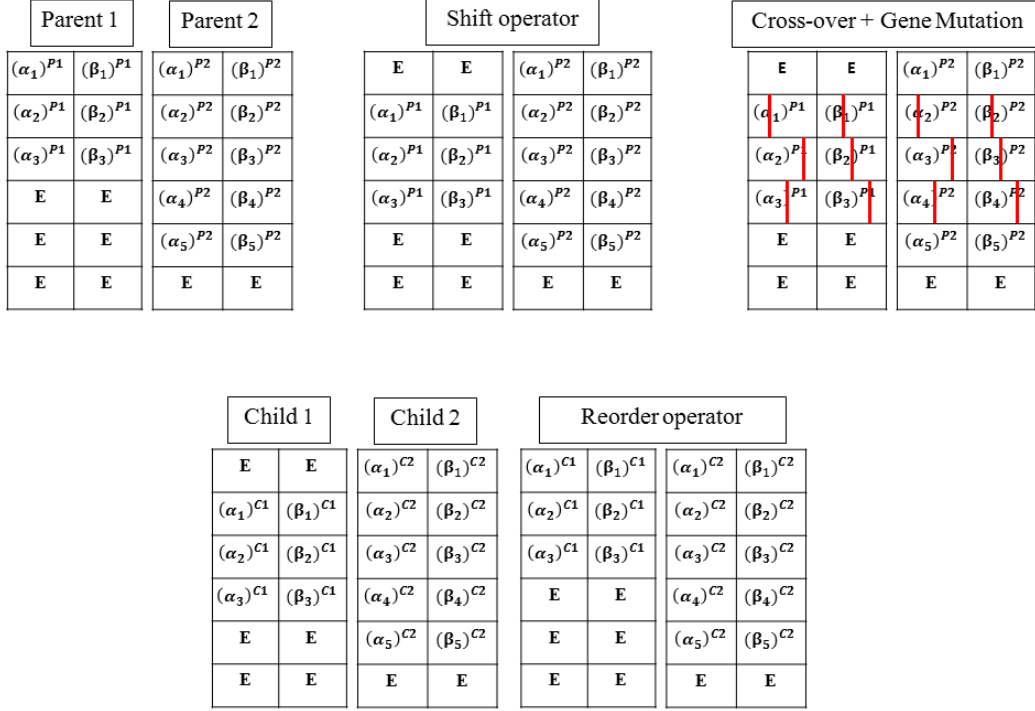


Figure 3.9 – Modified reproduction phase between different species for optimisation of modular systems.

a threshold number of generations or a no-improvement condition on the objective function (i.e. the algorithm is stopped if the objective function is not decreased for a certain number of consecutive iterations). Eventually, another possible stopping criterion is the achievement of an assigned value for the objective function. It is pointed out that, as a practical rule, the fixed number of generations  $N_{gen}$  can be set by assuring that the product  $N_{gen} \times N_{ind}$  is at least 80000.

A summary of the advantages and drawbacks of GAs in general and of the new possible opportunities of BIANCA are listed here below.

### Advantages of GAs

- GAs can deal with non-convex problem because of their intrinsic capability of exploring the design domain. Moreover, they work on the improvement of a population of individuals and not on the improvement of a single point in the solutions space.
- GAs are *zero-order methods*, i.e. they only need the evaluation of objective and

constraint functions, without any further information. This fact allows for dealing with discontinuous problems, wherein the discontinuity could occur in terms of non-regular function or in terms of variables nature (discrete/scattered variables can be easily handled as well as continuous ones).

- The use of probability-based rules instead of deterministic ones does not mean that GAs act completely randomly. The information about the behaviour of the objective function and constraints is suitably stored and exploited all along the iterations.

#### **Drawbacks of GAs**

- Real-world engineering problems, especially those of the structural domain, need expensive computational efforts. One single objective function evaluations could require a significant amount of time. The solution to this issue is to promote systems providing a reliable, but still computationally cheap, approximation of the objective function.
- GAs are not effective when decisions problems are faced.
- GAs are sensitive to the setting of its intrinsic parameters, namely the cross-over and mutation probabilities, the size of the population, the choice of the selection operator, etc..

#### **Improved Capabilities of BIANCA**

- It has been proven that BIANCA can effectively handle modular systems: see the successful applications shown in [89–94].
- BIANCA has no limitations in terms of constraints handling thanks to the implemented ADP strategy [7].
- Other operators improving the capabilities of standard GAs (multiple population, migration) have been implemented [72]. They are not discussed here for the sake of brevity.

### **3.3.3 The MATLAB version of BIANCA**

BIANCA has been originally coded in FORTRAN environment in order to foster computational speed. In spite of the rigid formalism imposed by the FORTRAN programming language, it has been possible to successfully implement a suitable individual structure to deal with modular systems, as stated in the previous section. In particular, the individual organisation in terms of chromosomes and genes was handled through opportune multidimensional arrays. In this study, an improved version of the GA BIANCA has been

recoded into the MATLAB environment. Even though this choice penalises the computational time, the utilisation of the MATLAB version of the GA is easier when compared to the ancient FORTRAN version. In addition, the ancient FORTRAN version of BIANCA was capable to deal with an only kind of module; now, thanks to the MATLAB structured variables, the architecture of the individual's genotype has been enriched and generalised, as illustrated in Fig. 3.10, in order to consider at the same time different kinds of modules. Without loss of generality, let  $N_m$  be the number of different types of

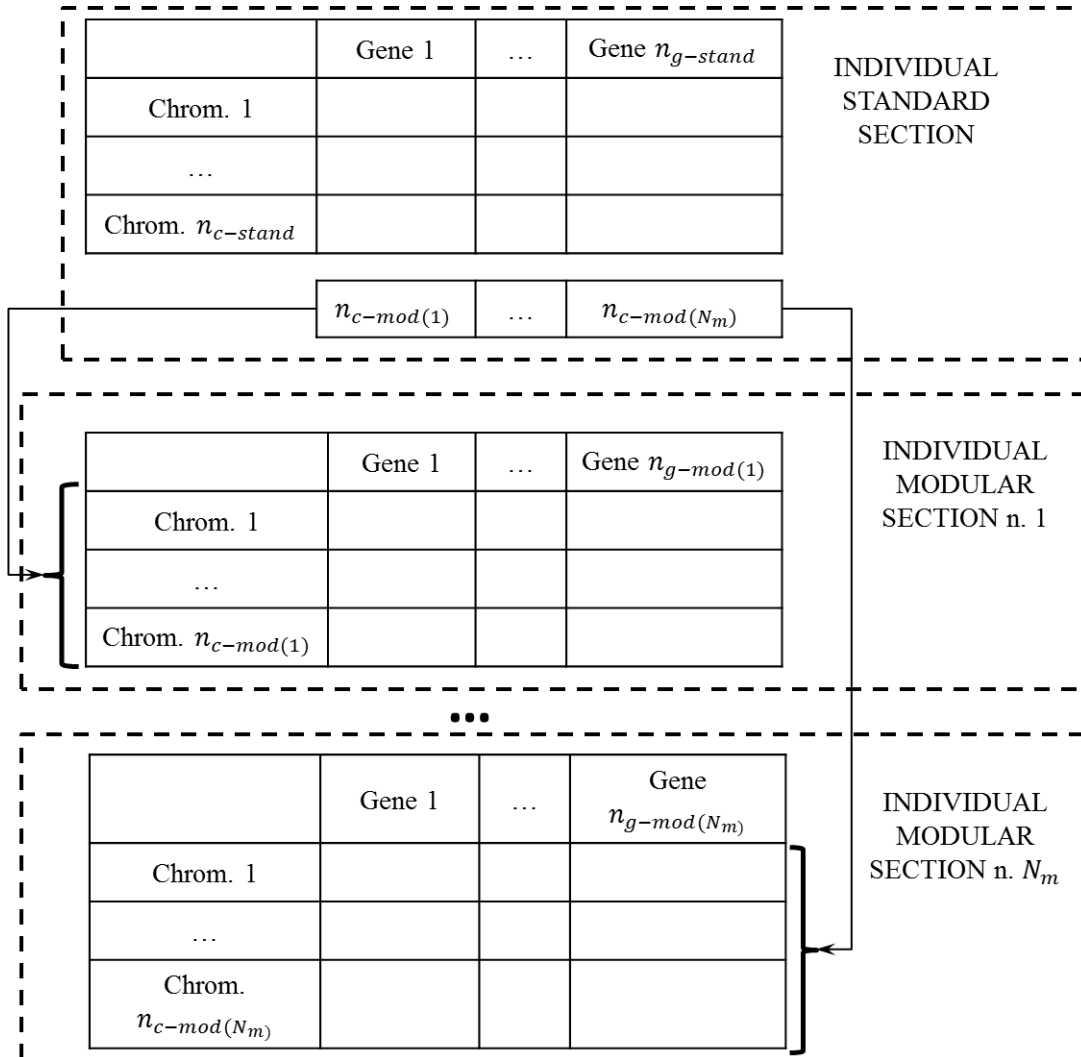


Figure 3.10 – The general individual's structure for the MATLAB version of BIANCA

modules for the problem at hand. Each individual (i.e. a point in the design space) is characterised by a genome composed of  $N_m + 1$  sections having a precise hierarchy. The first section (i.e. the *standard section*) is linked to the non-modular part of the prob-

lem and its genotype is split in two parts: the first one is composed of a fixed number ( $n_{c-stand}$ ) of chromosomes and each chromosome is made of  $n_{g-stand}$  genes. The second part is composed of only one chromosome having  $N_m$  genes which can be related (or not) to the values of some genes of the first part. This first section undergoes the action of the standard GA operators (refer to section 3.3.2). As shown in Fig. 3.10, each gene belonging to the mono-chromosome structure of the *standard section* is related to the number of modules  $n_{c-mod(k)}$  of the generic  $k$ -th *modular section*, ( $k = 1, \dots, N_m$ ). Accordingly, each one of the remaining  $N_m$  modular sections is characterised by a genotype composed of  $n_{c-mod(k)}$  chromosomes and  $n_{g-mod(k)}$  genes. Of course, the reproduction between species is allowed only on the modular sections. This new and improved version of BIANCA has been used in Chapter 7 to solve the curve fitting problem in a very general framework.

## 3.4 Conclusions on Optimisation Methods and Algorithms

Some optimisation tools have been discussed. As the domain of optimisation is incredibly wide, the purpose of this chapter is just to introduce the main features of some optimisation algorithms and to explain why they are important. Classification criteria for optimisation problems have been provided as well and the attention has been focused on deterministic and meta-heuristic algorithms for the solution of CNLPP. The most remarkable aspect of this discussion is that the “best algorithm” does not exist and the choice of the optimisation algorithm should be carefully addressed according to the problem at hand. Particularly, it will be shown in the following Chapters that many TO problems are still solved through deterministic algorithms, although they are non-convex. In the light of the topics related to meta-heuristics, one may wonder about the worthiness of this choice: a meta-heuristic algorithm could better explore the design domain and reduce the possibility to fall on a pseudo-optimal solution. The evident objection is that TO problems are characterised by an enormous amount of variables and a complete solution by means of a purely meta-heuristic approach could be prohibitive even for a simple 2D problem. Thus, when the optimisation problem is non-convex, the use of meta-heuristics is not the automatic right answer to solve it. For instance, providing a suitable starting point and to use a deterministic method could be a good choice: in this case, the driving criteria for selecting a meaningful initial guess for the design variables is provided by the knowledge of the phenomena related to the problem at hand. Another trade-off choice is the use of hybrid meta-heuristic - deterministic tools: initially, the domain is investigated through a meta-heuristic method and, when a profitable zone is identified, the optimum solution of the meta-heuristic algorithm is used as initial guess for the deterministic one. An example in this sense is given in Chapter 7.



# Chapitre 3

## Méthodes et algorithmes d'optimisation

### 3.1 Introduction aux méthodes d'optimisation

Les méthodes et algorithmes d'optimisation ont suscité l'intérêt de chercheurs et d'entreprises depuis plusieurs décennies. L'optimisation a été exploitée dans presque tous les développements liés à cette thèse. Dans ce chapitre, les principales caractéristiques de l'optimisation sont discutées. Avant de poursuivre la discussion, certains concepts principaux sont introduits.

Le but de l'optimisation est de minimiser une fonction objectif assignée  $f$  en fonction de plusieurs paramètres (parfois le nombre est très important). Certains de ces paramètres peuvent être définis comme constants. Au lieu de cela, d'autres paramètres ont un impact significatif sur la fonction objectif. Ces paramètres sont appelés variables d'optimisation ou de conception et sont collectés dans le tableau  $\mathbf{x} \in \mathbb{R}^n$ . Les variables de conception constituent les inconnues du problème d'optimisation, de sorte que la dépendance de  $f$  de  $\mathbf{x}$  est clarifiée par la notation  $f(\mathbf{x})$ . Dans les problèmes d'ingénierie, il est très rare qu'un problème d'optimisation ne soit caractérisé que par une fonction objectif et les variables de conception associées. Habituellement, la formulation du problème est enrichie avec  $m_e$  *contraintes d'égalité*, sous la forme de  $h_i(\mathbf{x}) = 0$ ,  $i = 1, \dots, m_e$  et / ou  $m_i$  *contraintes d'inégalité*, sous la forme  $g_j(\mathbf{x}) \leq 0$ ,  $j = 1, \dots, m_i$ . Les contraintes d'optimisation formalisent certaines exigences physiques ou techniques, qui dépendent de la spécificité du problème. Un point  $\mathbf{x}^*$  répondant à toutes les contraintes d'égalité et d'inégalité est appelé *faisable*. Le choix de la forme mathématique des fonctions objectif/contraintes et des variables de conception est appelé *modélisation*.

D'un point de vue mathématique, le problème d'optimisation est classiquement présenté sous la forme d'un problème de minimisation sous contrainte, comme indiqué dans l'éq. (3.1), où  $\mathbf{x}_{LB}$  et  $\mathbf{x}_{UB}$  sont les vecteurs des limites inférieure et supérieure, respectivement. Le problème (3.1) est classiquement appelé *problème de programmation non linéaire sous contraintes* (CNLPP).

Habituellement, la solution du problème (3.1) ne peut pas être déduite sous une forme explicite (à part quelques cas très particuliers) et un algorithme d'optimisation approprié doit être utilisé pour effectuer la recherche de solution. Le choix d'un algorithme d'optimisation pertinent est influencé par plusieurs facteurs, qui constituent autant de critères pour classer les algorithmes d'optimisation dans la bibliographie [70]. Ces critères sont résumés dans la fig. 3.1 et discutés ci-dessous.

- Présence de contraintes d'optimisation.
- Linéarité. Si les fonctions objectif et contraintes sont des fonctions linéaires des variables de conception  $\mathbf{x}$ , alors le problème est considéré comme linéaire (programmation linéaire).
- Convexité. Un problème d'optimisation est *convexe* si la fonction objectif est convexe, les fonctions de contrainte d'égalité sont linéaires et les fonctions de contrainte d'inégalité sont convexes. En cas de problème convexe, le minimum locale est le minimum globale et, si  $f$  est différentiable, un point stationnaire  $\mathbf{x}^*$  est un point de minimum global [70]. Il est souligné que la plupart des problèmes d'ingénierie du monde réel sont intrinsèquement non-convexes et que la recherche de la solution globale est prohibitive du point de vue informatique.
- Continuité. Une optimisation est définie continue lorsque les fonctions objectif / contraintes sont continues et l'ensemble des variables de conception est continu dans  $\mathbb{R}^n$  (ou au moins dans un sous-ensemble de  $\mathbb{R}^n$ ). Si les variables sont discrètes (régulièrement ou dispersées, c'est-à-dire sans longueur de pas fixe), des algorithmes appropriés doivent être utilisés pour résoudre le problème d'optimisation associé.
- Modèle mathématique. Lorsque le modèle est complètement connu et les variables d'optimisation ainsi que les fonctions objectif et contraintes sont continues, les algorithmes déterministes constituent la meilleure solution. Cependant, il peut arriver que certaines quantités soient caractérisées par de l'incertitude. Dans ce cas, les algorithmes stochastiques sont les plus adaptés. Une classe spéciale d'algorithmes, qui mérite une attention particulière, est appelée méta-heuristique [71]. La principale caractéristique de ces algorithmes est l'insertion d'un composant aléatoire pour effectuer la recherche de solution pour un problème d'optimisation.

Malheureusement, les problèmes rencontrés dans ce manuscrit sont pour la plupart non-convexes. Pour traiter ces problèmes, des algorithmes déterministes et méta-heuristiques ont été utilisés.

## 3.2 Méthodes déterministes pour les problèmes de type CNLPP

### 3.2.1 Généralités sur les méthodes déterministes

Les méthodes déterministes exploitent les informations disponibles sur le modèle pour mettre à jour le vecteur de variables de conception de  $\mathbf{x}_k$  à  $\mathbf{x}_{k+1}$ , où  $k$  et  $k + 1$  indiquent l'itération en cours. Ils constituent un domaine bien établi en littérature (voir, par exemple, [73] et [74]). Dans ce qui suit, les expressions «déterministique» et «basé sur le gradient» peuvent être confondues. Cela signifie que les informations de base pour la mise à jour des variables de conception reposent sur l'opérateur de gradient.

En ce qui concerne l'optimisation sans contraintes, deux méthodes principales existent pour générer les itérations  $\mathbf{x}_k$ . Ce sont les méthodes de *recherche linéaire* et les méthodes *région de confiance* [70].

Dans les méthodes de recherche linéaire, tout d'abord, une direction appropriée  $\mathbf{p}_k$  est choisie comme *direction de descente* pour la fonction objectif  $f$  et ensuite, le problème d'optimisation monodimensionnel plus simple de l'éq. (3.2) est résolu. Les différents algorithmes disponibles dans la littérature diffèrent principalement en raison du choix différent de la direction de descente. Bien que d'autres choix soient disponibles dans la littérature (tels que la direction de descente la plus rapide ou la direction de Newton), les deux plus efficaces et utilisés dans la bibliographie sont la direction quasi-Newton et la direction fournie par la méthode du gradient conjugué. En particulier, dans les méthodes quasi-Newton, la matrice Hessienne n'est pas explicitement calculée et une approximation appropriée est exploitée :  $\mathbf{p}_k = -\mathbf{B}_k^{-1}\nabla f_k$ , avec  $\mathbf{B}_k$  l'approximation de la matrice Hessienne (donnée, par exemple, par la formule de Broyden-Fletcher-Goldfarb-Shanno). En revanche, le taux de convergence des méthodes à gradient conjugué n'est pas aussi rapide que celui des méthodes de Newton ou de quasi-Newton, mais elles ne nécessitent pas le stockage de grandes matrices.

Dans les méthodes région de confiance, tout d'abord la longueur du pas est choisie, puis la direction de recherche est calculée. Une approximation quadratique du problème d'origine est résolue dans la région de confiance, afin de donner l'itération suivante. Avec une technique appropriée pour mettre à jour  $\Delta_k$  pendant les itérations, les possibilités de recherche sont les mêmes que celles des algorithmes de recherche linéaire susmentionnés (à l'exception de la méthode du gradient conjugué, qui n'a pas d'équivalent dans la méthode à région de confiance). Une remarque importante concernant les algorithmes à région de confiance est leur sensibilité à l'échelle des variables de conception.

Bien que les problèmes sans contraintes constituent rarement un réel intérêt pour la science et les applications, les algorithmes développés pour l'optimisation sans contrainte sont fondamentaux car leurs version améliorée peut être utilisée pour résoudre des problèmes de type CNLPP, souvent réduite à un problème sans contraintes équivalent.



### 3.2.2 Conditions d'optimalité pour les CNLPP

Etant donné un CNLPP sous la forme de l'éq. (3.1), la fonction lagrangienne  $L$  peut être définie comme indiqué dans l'éq. (3.3), où  $\boldsymbol{\lambda}$  et  $\boldsymbol{\mu}$  sont les vecteurs des multiplicateurs de Lagrange pour les contraintes d'inégalité ( $\lambda_i \geq 0, \forall i = 1, \dots, m_i$ ) et les contraintes d'égalité ( $\mu_j, \forall j = 1, \dots, m_e$ ), respectivement.

**Théorème 3.2.1 : Conditions nécessaires du premier ordre.**

Supposons que

1.  $\mathbf{x}^*$  est une solution locale du problème (3.1) ;
2. les fonctions  $f, g_i$  et  $h_j$  sont continuellement différentiables ;
3. la condition LICQ (Qualification de contrainte d'indépendance linéaire) est maintenue à  $\mathbf{x}^*$ , c'est-à-dire que les gradients de chaque fonction de contrainte d'égalité et d'inégalité doivent être linéairement indépendants.

Alors, deux vecteurs de multiplicateurs de Lagrange  $\boldsymbol{\lambda}^* \geq \mathbf{0}$  et  $\boldsymbol{\mu}^*$  existent et les conditions de l'éq. (3.4), connues sous le nom de conditions Karush-Kuhn-Tucker (KKT), sont remplies.

Dans l'éq. (3.4),  $\nabla_x$  est l'opérateur de gradient par rapport aux variables de conception  $\mathbf{x}$ . Le point  $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$  est nommé point KKT.

**Théorème 3.2.2 : Conditions suffisantes du second ordre.**

Supposons que les fonctions  $f, g_i$  et  $h_j$  soient deux fois continuellement différentiables. Supposons que  $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$  soit un point KKT et supposons l'Hessien du Lagrangien  $\nabla^2 L(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$  est défini positif. Alors  $\mathbf{x}^*$  est une solution locale stricte du problème (3.1).

Pour une revue des théorèmes susmentionnés, le lecteur est invité à se référer à [70, 73–75].

### 3.2.3 Algorithmes déterministiques pour les CNLPP

Les stratégies d'optimisation déterministiques proposées dans ce manuscrit s'appuient sur le package d'optimisation bien connue *fmincon* implémentée dans MATLAB [76]. Dans ce cadre, on remarque que l'algorithme à région de confiance de *fmincon* est vraiment efficace mais, soumis à des limitations importantes : il ne peut être utilisé que lorsque le gradient de la fonction objectif est connu sous forme analytique et dans le cas de contraintes d'égalité ou de contraintes linéaires (mais pas les deux). En raison de ces manques importantes, l'algorithme à région de confiance n'est pas pris en compte dans cette discussion. Les autres algorithmes de *fmincon* sont des algorithmes de recherche linéaire : SQP (Sequential Quadratic Programming), AS (Active Set), IP (Interior Point).

#### **SQP et AS**

AS n'est qu'une méthode SQP particulière dans laquelle les contraintes sont traitées de manière plus efficace. Par conséquent, les principales étapes décrites ci-dessous sont

variables pour les algorithmes SQP et AS. Certaines différences remarquables seront discutées plus tard. L'idée principale derrière une méthode SQP consiste à aborder le CNLPP dans une séquence de problèmes de programmation quadratique (QP), qui peut toujours être résolue ou, au moins, il peut être prouvé que la solution n'existe pas [70, 77, 78]. Un aperçu général de l'algorithme SQP est fourni à la fig. 3.2.

Une fois que le CNLPP a été défini sous la forme de l'éq. (3.1), une estimation initiale appropriée des variables de conception doit être fournie. Pendant ce temps, les multiplicateurs de Lagrange sont initialisés et l'index d'itération  $k$  est défini sur 0. Ensuite, la forme lagrangienne est évaluée par l'éq. (3.3). Les gradients de la fonction objectif et des contraintes sont nécessaires pour les étapes suivantes. Par conséquent, la matrice Hessienne de la Lagrangien est convenablement approximée et elle est utilisée, avec les gradients calculés précédemment, pour définir le sous-problème QP local de l'éq. (3.5). On suppose que l'approximation de la matrice Hessienne a été réalisée grâce à la formule *BFGS*. SQP et AS diffèrent alors dans la méthode de résolution du problème (3.5). En particulier, AS tolère une itération non réalisable et dispose de capacités améliorées pour explorer les limites définies par les contraintes actives.

La solution du problème (3.5) est la direction de recherche  $\mathbf{d}_k$  : les prochaines phases sont consacrées à l'étape  $s_k$  le long de cette direction de descente et à mettre à jour les deux multiplicateurs de Lagrange ainsi que le vecteur des variables de conception  $\mathbf{x}_{k+1}$ . Plusieurs critères de convergence sont pris en compte dans l'implémentation de MATLAB, tels que le nombre maximal d'itérations, l'amélioration de la fonction objectif, le changement négligeable des valeurs des variables ou la norme du gradient de la fonction de Lagrange proche de 0. Pour une compréhension plus approfondie des algorithmes SQP, le lecteur est invité à lire [70, 76].

### Algorithme IP

Les algorithmes IP sont les algorithmes les plus efficaces pour résoudre les CNLPP ensemble aux méthodes SQP. De manière générale, on peut affirmer que dans certaines conditions, les algorithmes IP pourraient être plus efficaces que les algorithmes SQP et converger en moins d'itérations. Cependant, les algorithmes IP sont légèrement moins robustes que les méthodes SQP, en particulier en explorant les limites du domaine faisable. L'idée principale consiste à transformer le CNLPP (3.1) en une suite de problèmes d'optimisation plus simples, dans lesquels seules les contraintes d'égalité sont prises en compte (comme illustré à la fig. 3.3). Considérant un certain index itératif  $k$  et une quantité scalaire positive  $\vartheta$ , le problème d'optimisation correspondant est donné par l'éq. (3.6). La solution d'un tel problème est fournie en utilisant les faux variables  $z_i$  (une faux variable est requise pour chaque contrainte d'inégalité du problème d'origine). La méthode IP tend à garder les itérations éloignées des frontières imposées par les contraintes et des stratégies appropriées doivent être mises en œuvre pour explorer ces frontières.

## 3.3 Méthodes méta-heuristiques pour les CNLPP

### 3.3.1 Généralités sur les méta-heuristiques

Les méta-heuristiques peuvent être définies comme des méthodes d'optimisation «globales» pour les CNLPP non convexes en utilisant plusieurs règles empiriques, inspirées par un phénomène naturel précis. Les méta-heuristiques permettent une meilleure exploration du domaine par rapport aux méthodes déterministes, car elles agissent sur une population de points dans le domaine de la conception plutôt que sur un seul point. Cependant, une méta-heuristique peut trouver l'optimum global, mais il n'y a aucune garantie de le trouver réellement.

De nombreux travaux concernant les méta-heuristiques ont été inspirés par le concept d'évolution dans la bibliographie [79–81]. Le principe commun consiste à établir les variables d'optimisation du problème considéré, à fournir une population de solutions candidates et à faire évoluer cette population vers une amélioration de la fonction objectif. L'algorithme génétique de Holland (GA) [82, 83] a suscité un intérêt particulier. D'autres méta-heuristiques ne reproduisent pas les mécanismes d'évolution, mais imitent plutôt le comportement de groupes d'individus dans la nature [84, 85]. Il est évident que le domaine des méthodes méta-heuristiques pour la programmation mathématique est un domaine d'une exceptionnelle immensité. Ici, la discussion est restreinte aux GA et, plus particulièrement, à un GA spécial, d'abord proposé dans [86, 87] et développée dans [72]. Dans cette thèse, l'algorithme a été modifié pour traiter correctement les problèmes proposés dans le chapitre 7.

### 3.3.2 L'algorithme génétique BIANCA

L'idée de base des GA est de faire évoluer une population en fonction de la capacité de survie de ses individus. Un individu est un ensemble de valeurs des variables de conception, c'est-à-dire une solution candidate au problème en question. L'évolution de la population est obtenue en simulant la reproduction parmi les individus et en simulant les caractéristiques des mécanismes de transmission, qui sont caractéristiques de la génétique. Le patrimoine génétique de deux individus est combiné par des opérations appropriées impliquant des chromosomes et des gènes. Un individu est composé de chromosomes et chaque chromosome est composé, à son tour, de gènes. Il est à noter que les opérations génétiques sont effectuées sur une version codée en binaire des variables de conception, généralement appelée génotype. L'ensemble correspondant de valeurs décodées des variables de conception est appelé phénotype. Le codage binaire est effectué sur les indices  $I$ , de l'éq. (3.7) après une étape de discrétisation appropriée. Un exemple est discuté à l'éq. (3.9) et aux figs. 3.4-3.5 afin de clarifier ces points.

L'attention est ici concentrée sur le GA appelé *BIANCA* (Biologically Inspired ANalysis of Composite Assemblages). La caractéristique la plus importante de *BIANCA* est

la capacité de traiter des systèmes modulaires d'ingénierie. Un système modulaire en ingénierie des structures est toute application constituée de plusieurs entités unitaires élémentaires et répétitives (les modules) qui présentent certains paramètres intrinsèques. Un exemple typique de système modulaire est un stratifié composite, dans lequel le module est représenté par un seul pli qui a certains paramètres intrinsèques tels que l'angle d'orientation des fibres, l'épaisseur, les propriétés du matériau, etc. Lorsqu'on optimise un système modulaire il n'y a pas de critère à priori pour définir le nombre optimal de modules. Par conséquent, le nombre de modules doit être inclus parmi les variables d'optimisation avec les paramètres intrinsèques de chaque module. En conclusion, le problème est caractérisé par un «nombre variable de variables d'optimisation». Ce fait a été mis en exergue dans l'éq. (3.10), où la dépendance du nombre de variables de conception et du nombre de contraintes avec le nombre de modules  $n_c$  a été rendu explicite. Pour plus de détails sur ce sujet, le lecteur intéressé est invité à lire [72].

Bien que le problème (3.9) soit très simple, il est représentatif d'une vaste classe de problèmes abordés dans la littérature, à savoir les problèmes mono-chromosomiques. Un exemple de problème plus difficile sous la forme de l'éq. (3.10) est l'optimisation d'une plaque stratifiée, comme le montre la fig. 3.6.

Dans le cadre de l'application des GA aux systèmes modulaires, le concept d' *espèces* se présente de manière très naturelle : les individus ayant le même nombre de chromosomes (modules) appartiennent à la même espèce. La dernière version de BIANCA a été explicitement conçue pour traiter ce genre de problèmes [72], y compris la reproduction parmi les individus appartenant à différentes espèces. Un schéma récapitulatif de l'algorithme BIANCA est montré à la fig. 3.7 et une description synthétique est rapportée ci-dessous.

**Population de départ.** La population de départ des individus est générée en sélectionnant aléatoirement les indices entiers associés aux variables, comme cela a été montré dans l'exemple de l'éq. (3.9).

**Adaptation.** Pour chaque individu, une unique mesure de son adaptation doit être fournie. Dans cette phase, une fonction de fitting est définie : il s'agit d'une fonction scalaire qui accepte les valeurs de la fonction objectif en entrée et renvoie une valeur comprise entre 0 et 1 (la fonction de fitting prend la valeur 0 pour le pire individu de la génération et 1 pour le meilleur).

**Sélection.** Les individus sont sélectionnés dans cette phase afin de constituer les couples de parents  $N_{ind}/2$  pour la phase de reproduction. Le concept de base est que les meilleurs individus ont une forte probabilité d'être choisis pour la phase de reproduction. En pratique, la fonction de fitting définie précédemment est utilisée pour attribuer à chaque individu une probabilité de sélection et, ensuite, la sélection effective est effectuée au moyen d'un critère ad hoc (généralement la roulette).

**Croisement.** Le croisement vise à combiner le patrimoine génétique des deux parents et constitue la première étape de la reproduction, c'est-à-dire la phase fournissant de nouveaux individus qui constitueront la nouvelle génération. L'opérateur de croisement

agit sur des gènes homologues, qui sont coupés aléatoirement avec une probabilité  $p_{cross}$ . Ensuite, les gènes sont combinés, comme le montre la fig. 3.8, où deux individus de la première génération du problème (3.9) sont pris en compte. Lorsque l'optimisation des systèmes modulaires est réalisée, le croisement standard de la fig. 3.8 n'est plus possible et des techniques appropriées doivent être développées. Sans fournir de détails sur la mise en œuvre, un schéma synthétique donnant une idée intuitive du croisement entre différentes espèces est donné à la fig. 3.9.

**Mutation.** Les mutations peuvent être interprétées comme un mécanisme d'adaptation de second ordre, visant à améliorer les capacités d'exploration de l'algorithme et à éviter que les individus ne soient entachés d'une solution locale optimale au début de l'optimisation. La première mutation consiste à choisir aléatoirement (avec une probabilité  $p_m$ ) un chiffre du codage binaire des variables et à le modifier à la fin de la phase de croisement. La seconde mutation intervient dans le cas de la reproduction sur des individus appartenant à des espèces différentes. Un module (chromosome) peut être supprimé / ajouté avec une probabilité  $p_{mc}$ .

**Nouvelle génération et élitisme.** Lorsque tous les nouveaux individus sont générés, la population peut être complètement mise à jour (les performances de chaque nouvel individu sont évaluées). Cependant, afin de protéger le bon patrimoine génétique, l'opérateur d'élitisme garantit que le plus mauvais individu de la progéniture est remplacé par le meilleur individu des ancêtres. Lorsque l'ADP est combiné avec l'élitisme et qu'une solution réalisable est trouvée, les individus de la génération suivante fourniront nécessairement une valeur améliorée (diminuée) de la fonction objectif ou, au moins, la même.

**Critère de convergence et meilleur individu.** Le critère de convergence est généralement un nombre seuil de générations, une condition d'absence d'amélioration sur la fonction objectif ou l'atteinte d'une valeur assignée à la fonction objectif.

Un résumé des avantages et des inconvénients des AG en général et des nouvelles opportunités possibles de BIANCA est présenté ci-dessous.

#### **Avantages des GA**

- Les GA peuvent traiter des problèmes non convexes en raison de leur capacité intrinsèque à explorer le domaine de la conception.
- Les GA sont des *méthodes d'ordre zéro*, c'est-à-dire qu'elles nécessitent uniquement l'évaluation des fonctions d'objectif et de contrainte, sans aucune information supplémentaire. Cette particularité permet également de traiter des problèmes discontinus.
- L'utilisation de règles basées sur les probabilités au lieu de règles déterministes ne signifie pas que les AG agissent de manière totalement aléatoire. Les informations sur le comportement de la fonction objectif et des contraintes sont stockées et exploitées de manière appropriée tout au long des itérations.

## Inconvénients des GA

- Les problèmes d'ingénierie du monde réel, en particulier ceux du domaine de l'étude des structures, nécessitent des efforts de calcul coûteux.
- Les GA ne sont pas efficaces lorsque des problèmes de décision sont rencontrés.
- Les GA sont sensibles à la définition de ses paramètres intrinsèques, à savoir les probabilités de croisement et de mutation, la taille de la population, le choix de l'opérateur de sélection, etc.

## Amélioration des capacités des GA avec BIANCA

- Il a été prouvé que BIANCA peut gérer efficacement les systèmes modulaires : voir des applications réussies présentées dans [89–94].
- BIANCA n'a pas de limites en termes de gestion des contraintes grâce à la stratégie ADP [7]. La puissance de la stratégie ADP implémentée dans BIANCA consiste à mettre à jour automatiquement et de manière adaptative certains coefficients, assurant un bon compromis entre prévention des solutions irréalisables et exploration efficace de la frontière du domaine réalisable.
- D'autres opérateurs améliorant les capacités des GA standard (population multiple, migration) ont été mis en place [72].

### 3.3.3 La version MATLAB de BIANCA

BIANCA a été initialement codé dans l'environnement FORTRAN afin de favoriser la vitesse de calcul. En dépit du formalisme rigide imposé par le langage de programmation FORTRAN, il a été possible de mettre en œuvre avec succès une structure élémentaire adaptée aux systèmes modulaires, comme indiqué dans le paragraphe précédent. Dans cette étude, une version améliorée de GA BIANCA a été recodée dans l'environnement MATLAB. Même si ce choix pénalise le temps de calcul, l'utilisation de la version MATLAB du GA est plus facile que celle de l'ancienne version FORTRAN. En outre, l'ancienne version FORTRAN de BIANCA était capable de traiter un seul type de module ; maintenant, grâce aux variables structurées de MATLAB, l'architecture du génotype de l'individu a été enrichie et généralisée, comme illustré à la fig. 3.10, afin de considérer simultanément différents types de modules. Sans perte de généralité,  $N_m$  représente le nombre de types de modules différents pour le problème en question. Chaque individu (c'est-à-dire un point dans l'espace de conception) est caractérisé par un génome composé de sections  $N_m + 1$  ayant une hiérarchie précise. La première section (la section standard) est liée à la partie non modulaire du problème et son génotype est divisé en deux parties : la première est composée d'un nombre fixe ( $n_{c-stand}$ ) de chromosomes et chaque

chromosome est constitué de gènes  $n_{g-stand}$ . La deuxième partie est composée d'un seul chromosome ayant des gènes  $N_m$  qui peuvent être liés (ou non) aux valeurs de certains gènes de la première partie. Cette première section subit l'action des opérateurs du GA standard (voir la section 3.3.2). Comme le montre la fig. 3.10, chaque gène appartenant à la structure mono-chromosomique de la section standard est lié au nombre de modules  $n_{c-mod(k)}$  de la  $k$ -ème section modulaire, ( $k = 1, \dots, N_m$ ). En conséquence, chacune des sections modulaires  $N_m$  restantes est caractérisée par un génotype composé de  $n_{c-mod(k)}$  chromosomes et  $n_{g-mod(k)}$  de gènes. Bien entendu, la reproduction entre espèces n'est autorisée que sur les sections modulaires. Cette nouvelle version améliorée de BIANCA a été utilisée dans le chapitre 7 pour résoudre le problème d'approximation de courbe dans un cadre très général.

### 3.4 Conclusions sur les méthodes et algorithmes d'optimisation

Certains outils d'optimisation ont été discutés. Le domaine de l'optimisation étant incroyablement vaste, l'objectif de ce chapitre est simplement de présenter les principales caractéristiques de certains algorithmes d'optimisation et d'expliquer pourquoi elles sont importantes. Des critères de classification pour les problèmes d'optimisation ont également été fournis et l'attention s'est concentrée sur les algorithmes déterministes et méta-heuristiques pour la solution d'un CNLPP. L'aspect le plus remarquable de cette discussion est que le "meilleur algorithme" n'existe pas et que le choix de l'algorithme d'optimisation doit être soigneusement pris en compte en fonction du problème rencontré. En particulier, il sera montré dans les chapitres suivants que de nombreux problèmes d'OT sont encore résolus par des algorithmes déterministes, bien qu'ils soient non convexes. À la lumière des sujets liés aux méta-heuristiques, on peut s'interroger sur la pertinence de ce choix : un algorithme méta-heuristique pourrait mieux explorer le domaine de la conception et réduire la possibilité de tomber sur une solution pseudo-optimale. L'objection évidente est que les problèmes d'OT sont caractérisés par une quantité énorme de variables et qu'une solution complète au moyen d'une approche purement méta-heuristique pourrait être impossible, même pour un problème 2D simple. Ainsi, lorsque le problème d'optimisation est non convexe, l'utilisation de méta-heuristiques n'est pas systématiquement la bonne réponse pour le résoudre. Par exemple, fournir un point de départ approprié et utiliser une méthode déterministe pourraient être des bon choix : dans ce cas, le critère déterminant pour sélectionner une estimation initiale significative des variables de conception est fourni par la connaissance des phénomènes liés au problème lui-même. Un autre choix est l'utilisation d'outils méta-heuristiques - déterministes hybrides : dans un premier temps, le domaine est étudié par une méthode méta-heuristique et, lorsqu'une zone intéressante est identifiée, la solution optimale de l'algorithme méta-heuristique est uti-

lisée comme solution initiale pour l'algorithme déterministe. Un exemple en ce sens est donné au chapitre 7.





# Chapter 4

## A NURBS-based Topology Optimisation Algorithm

### 4.1 Introduction

Non-Uniform Rational Basis Splines (NURBS) are extremely versatile mathematical entities, which are used to solve several engineering problems. As stated in Chapter 2, they constitute the foundations of Computer Aided Design (CAD) for surfaces and curves modelling [66–68]. The utilisation of NURBS has shown a considerably growth over the years: nowadays, they are employed in problems of different nature, such as optimisation strategies for curve/surface fitting [95–97] or meta-modelling [98, 99]. Moreover, an increasing number of work is carried out, wherein NURBS entities are associated to the structural analysis of mechanical components. A smart application of NURBS surfaces has been shown in [93, 94, 100] for optimising mechanical properties of variable angle tow (VAT) composites. NURBS also constitute the basis of the relatively new concept of *isogeometric analysis* [101], which represents a challenging integration of CAD entities into the Finite Element (FE) method. When looking at the consistent amount of contributions dealing with NURBS curves and surfaces (of which the aforementioned works constitute just an example), one can imagine to exploit their potential as well as their interesting properties (refer to Chapter 2) in the domain of TO.

Recent research efforts, finalised to fill in some gaps of current TO methods, have directed their attention towards NURBS entities. As known, the main source of the drawbacks related to density-based TO methods is the lack of a geometric entity dedicated to the description of the topology: the mesh of the part to be optimised and the pseudo-density function provide information about both the performance of the structure (according to the problem at hand) and the topology (refer to Chapter 1). The LSM represents a first attempt to overcome this difficulty by introducing geometric entities to properly describe the topology. In this context, some authors tried to integrate the advantages of NURBS.

B-Spline curves constitute the shape function of the special class of FE employed in [102]. In the procedure discussed in [103], the LSM is coupled to the isogeometric analysis by means of the NURBS formalism for both LSF parametrisation and objective function computation. However, the intrinsic difficulties related to the LSM and the sensitivity of the solution to the starting guess of the LSF (see Chapter 1) still constitute a major issue that circumvents the wide spread of LSM-based TO strategies. Therefore, it is not surprising that pioneering TO density-based strategies [5, 34, 41] are still widely studied research topics. An interesting and promising enhancement of the SIMP method, based on the application of B-Spline entities, has been provided in [104] and [105] for 2D and 3D applications, respectively. In these studies, the fictitious density field has been related to a B-Spline surface/hyper-surface. The B-Spline formalism permits to take advantage of an implicitly defined filter zone, whose size depends on the B-Spline parameters. Therefore, the well-known checker-board and mesh dependence effects can be overcome without any dedicated strategy, e.g. distance based filters [5] or projection methods [10], [42].

This Chapter introduces an innovative TO method and the related algorithm, based on a further development and a generalisation of the method proposed in [104, 105]. Here, the pseudo-density field characterising the SIMP method is related to a NURBS surface/hyper-surface for 2D and 3D applications, respectively. For 2D problems, each point of the NURBS control net is then characterised by three coordinates of which two are Cartesian coordinates and the third one is the pseudo-density. For 3D problems, a 4D hyper-surface is used to represent the fictitious density field: each point constituting the control *hyper-net* has three Cartesian coordinates and the fourth coordinate is the pseudo-density. The proposed approach is called NURBS-based SIMP method.

The impact of other parameters, such as the NURBS weights, is investigated and suitable comparisons are carried out between solutions of TO problems obtained through B-Spline and NURBS.

Furthermore, the discussion is not restrained only to the beneficial implicit filter zone provided by the NURBS formalism. It is well-known that one of the main shortcomings of density-based methods is the time consuming postprocessing phase, necessary to rebuild the boundary of the optimum topology of the structure starting from a FE “pixelised”/“voxelised” domain (providing the required smoothness). A careful description of the geometry is crucial in TO not only to save time in postprocessing but, mostly, to ensure that the optimum shape of the component (rebuilt at the end of TO) could meet the design constraints. It will be shown that the NURBS-based approach can easily provide fully CAD-compatible optimised geometries and that optimisation constraints are met on the actual reassembled geometry in 2D.

As far as 3D applications are concerned, although the NURBS hyper-surfaces are geometric, potentially CAD-compatible entities, the geometry reconstruction/assembly phase of the structure boundary after optimisation in terms of CAD surfaces still remains a challenge for 3D problems. Currently, the 3D topology is obtained after an intersection

operation between the 4D NURBS hyper-surface and a suitable hyper-plane. Hence, the resulting geometry is described by a “well-defined” Standard Tessellation Language (STL) CAD native format (i.e. an STL file without missing or degenerated triangles) representing the boundary of the optimum topology. The difficulties related to the full CAD-compatibility of 3D structures will be discussed as well.

The Chapter follows this outline: firstly, the mathematical statement of the NURBS-based TO method is provided for both 2D and 3D applications. The classic problem of compliance minimisation with an equality constraint on the volume is considered in this discussion. Secondly, the algorithm related to the NURBS-based approach is presented and its capabilities are discussed, together with its limitations. Then, meaningful results are shown on standard benchmarks. The contribution of the weights appearing in the NURBS formulation is investigated and results are compared to those obtained using B-Spline entities. Moreover, a sensitivity analysis to the NURBS discrete parameters (i.e. the degrees and the number of control points) is carried out. Results provided by the proposed algorithm are validated by means of the well-established TO software Altair OptiStruct<sup>®</sup>. Finally, conclusions and perspectives on the contributions of the NURBS-based SIMP method are given.

The contents of this Chapter refer to the articles [106–109].

## 4.2 Mathematical Formulation of the NURBS-based Topology Optimisation Method

The notation used in this Chapter is consistent to that introduced in Chapters 1 and 2. In particular, the reference domain, wherein the TO takes place, is always defined as a rectangular compact of size  $a_1 \times a_2$  in 2D and a parallelepiped compact  $a_1 \times a_2 \times a_3$  in 3D. In the NURBS-based SIMP method, the pseudo-density field (i.e. the topological descriptor) is represented through a suitable NURBS entity. Therefore, a NURBS surface is used for 2D problems:

$$\rho(u_1, u_2) = \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} R_{i_1, i_2}(u_1, u_2) \widehat{\rho}_{i_1, i_2}. \quad (4.1)$$

Analogously, a NURBS hyper-surface is necessary for 3D problems:

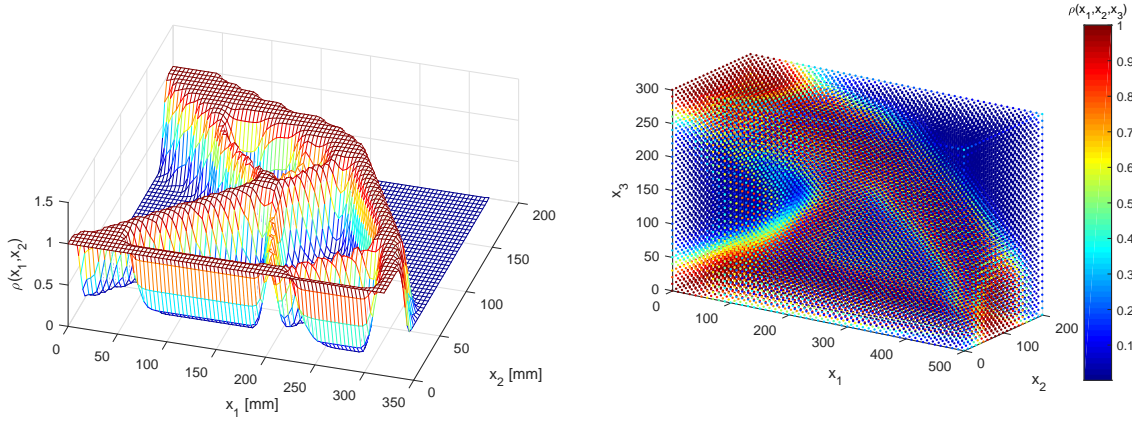
$$\rho(u_1, u_2, u_3) = \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \sum_{i_3=0}^{n_3} R_{i_1, i_2, i_3}(u_1, u_2, u_3) \widehat{\rho}_{i_1, i_2, i_3}. \quad (4.2)$$

In Eqs. (4.1) and (4.2),  $R_{i_1, i_2}(u_1, u_2)$  and  $R_{i_1, i_2, i_3}(u_1, u_2, u_3)$  are the NURBS rational basis functions, defined according to Eqs. (2.13) and (2.20), respectively. Of course,  $\rho(u_1, u_2)$  of Eq. (4.1) represents only the third coordinate of the array  $\mathbf{S}(u_1, u_2)$  of Eq. (2.12): the

three coordinates in the *NURBS space* are the two spatial coordinates and the pseudo-density. Similarly,  $\rho(u_1, u_2, u_3)$  of Eq. (4.2) constitutes the fourth coordinate of the array  $\mathbf{H}(u_1, u_2, u_3)$  in the special case of Eq. (2.19) specified for  $N = 3$  and  $M = 4$ . Hence, the dimensionless parameters  $u_1$ ,  $u_2$ , and  $u_3$  are directly related to the physical coordinates as follows:

$$u_j = \frac{x_j}{a_j}, j = 1, 2, 3. \quad (4.3)$$

Control points  $\hat{\rho}_{i_1, i_2}$  are organised in a two-dimensional array (two indices), whilst  $\hat{\rho}_{i_1, i_2, i_3}$  are normally set in a three-dimensional array, according to the NURBS surfaces and hyper-surfaces algorithms. It is remarked here that a 3D geometric entity is needed to correctly describe the topology of a 2D domain (Fig. 4.1a). By generalising this concept, the fictitious density field can be associated to a 4D NURBS hyper-surface to describe the topology of a structure in the 3D Euclidean space. In this case, a typical representation could be provided in terms of colours to plot 4D graphs, as shown in Fig. 4.1b.



(a) NURBS surface for 2D structures.

(b) NURBS hyper-surface for 3D structures.

Figure 4.1 – SIMP pseudo-density representation through NURBS entities.

As stated above, there are many parameters affecting the shape of NURBS entities intervening in Eqs. (4.1) and (4.2). Among them, the NURBS control points and the related weights are identified as *design variables*. They are arranged in the arrays  $\xi_1^{2D} \in \mathbb{R}^{[(n_1+1)(n_2+1)] \times 1}$  and  $\xi_2^{2D} \in \mathbb{R}^{[(n_1+1)(n_2+1)] \times 1}$  for 2D problems:

$$\begin{aligned} \xi_1^{2D} &= \{\hat{\rho}_{0,0}, \dots, \hat{\rho}_{n_1,0}, \hat{\rho}_{0,1}, \dots, \hat{\rho}_{n_1,1}, \dots, \hat{\rho}_{n_1,n_2}\}, \\ \hat{\rho}_{i_1, i_2} &\in [\hat{\rho}_{min}, \hat{\rho}_{max}], \quad \forall i_1 = 0, \dots, n_1, \quad \forall i_2 = 0, \dots, n_2, \end{aligned} \quad (4.4)$$

$$\begin{aligned} \xi_2^{2D} &= \{w_{0,0}, \dots, w_{n_1,0}, w_{0,1}, \dots, w_{n_1,1}, \dots, w_{n_1,n_2}\}, \\ w_{i_1, i_2} &\in [w_{min}, w_{max}], \quad \forall i_1 = 0, \dots, n_1, \quad \forall i_2 = 0, \dots, n_2. \end{aligned} \quad (4.5)$$

## 4.2. Mathematical Formulation of the NURBS-based Topology Optimisation Method

Analogously, control points and weights are collected in the arrays  $\xi_1^{3D} \in \mathbb{R}^{[(n_1+1)(n_2+1)(n_3+1)] \times 1}$  and  $\xi_2^{3D} \in \mathbb{R}^{[(n_1+1)(n_2+1)(n_3+1)] \times 1}$  for 3D applications:

$$\begin{aligned} \xi_1^{3D} &= \{\hat{\rho}_{0,0,0}, \dots, \hat{\rho}_{n_1,0,0}, \hat{\rho}_{0,1,0}, \hat{\rho}_{n_1,1,0}, \dots, \hat{\rho}_{n_1,n_2,0}, \dots, \hat{\rho}_{0,0,n_3}, \dots, \hat{\rho}_{n_1,n_2,n_3}\}, \\ \hat{\rho}_{i_1,i_2,i_3} &\in [\hat{\rho}_{min}, \hat{\rho}_{max}], \quad \forall i_1 = 0, \dots, n_1, \quad \forall i_2 = 0, \dots, n_2, \quad \forall i_3 = 0, \dots, n_3, \end{aligned} \quad (4.6)$$

$$\begin{aligned} \xi_2^{3D} &= \{w_{0,0,0}, \dots, w_{n_1,0,0}, w_{0,1,0}, w_{n_1,1,0}, \dots, w_{n_1,n_2,0}, \dots, w_{0,0,n_3}, \dots, w_{n_1,n_2,n_3}\}, \\ w_{i_1,i_2,i_3} &\in [w_{min}, w_{max}], \quad \forall i_1 = 0, \dots, n_1, \quad \forall i_2 = 0, \dots, n_2, \quad \forall i_3 = 0, \dots, n_3. \end{aligned} \quad (4.7)$$

The other NURBS parameters can be identified as *design parameters*, i.e. their value is *a priori* set at the beginning of the TO analysis and it is not optimised. Here below a concise discussion on the attended effect of these parameters on the final optimum topology is given.

- *The degrees*: increasing the degree implies broadening the local support size (refer to Chapter 2) and the effects of this operation need to be investigated.
- *The control points number*: increasing the control points number implies enhancing the description of the topology and, hence, achieving better performances in terms of objective function. Of course, this fact involves a higher number of design variables and, consequently, an increased computational burden is expected.
- *The knot vector*: the non-trivial knot vectors components appearing in Eqs. (2.14)-(2.15) and in Eq. (2.24) have been uniformly distributed on the interval  $[0, 1]$  for both 2D and 3D problems.
- *Spatial coordinates of control points*: they are useless in terms of optimisation because only the third coordinate of  $\mathbf{S}$  or the fourth coordinate of  $\mathbf{H}$  are related to the topology description. However, they can become meaningful for postprocessing operations and for other applications described in Chapter 5 of this manuscript. The idea is to distribute control points in the Euclidean space in such a way that the NURBS evaluation along the  $x_j$  coordinate coincides directly with  $x_j$ ,  $j = 1, 2, 3$ . When B-Spline entities are employed, this requirement can be exactly formulated from a mathematical viewpoint and it results in a simple and useful formula, providing the so called *Greville's abscissae* for control points spatial coordinates, i.e.

$$\begin{cases} X_{I_1,*,*}^{(1)} = \frac{a_1}{p_1} \sum_{k=0}^{p_1-1} U_{I_1+k+1}^{(1)}, & I_1 = 0, \dots, n_1, \\ X_{*,I_2,*}^{(2)} = \frac{a_2}{p_2} \sum_{k=0}^{p_2-1} U_{I_2+k+1}^{(2)}, & I_2 = 0, \dots, n_2, \\ X_{*,*,I_3}^{(3)} = \frac{a_3}{p_3} \sum_{k=0}^{p_3-1} U_{I_3+k+1}^{(3)}, & I_3 = 0, \dots, n_3. \end{cases} \quad (4.8)$$

wherein the symbol  $*$ , replacing two of the three indices, aims at pointing out that the considered Greville's abscissa depends only upon the corresponding knot vector. Eq. (4.8) holds for 3D problems, whilst only the first two equations must be considered and the index  $I_3$  must be disregarded for 2D problems. The *Greville's abscissae* guarantee that the spatial coordinates of the surface/hyper-surface control points are distributed in such a way that the B-Spline evaluation at the  $x_j$  coordinate coincides directly with  $x_j$ , with  $j = 1, 2, 3$ .

For the sake of synthesis, the two following arrays can be defined:

$$\Xi^{(l)} = \begin{cases} \xi_l^{2D} & \text{if } N = 2, \\ \xi_l^{3D} & \text{if } N = 3, \end{cases} \quad l = 1, 2. \quad (4.9)$$

Of course, the total number of control points is trivially  $n_{tot} = (n_1 + 1)(n_2 + 1)$  in 2D and  $n_{tot} = (n_1 + 1)(n_2 + 1)(n_3 + 1)$  in 3D.

In this framework, the formulation of the classic TO problem of compliance minimisation subject to an equality constraint on the volume is revised. The problem statement of Eq. (1.6) changes into

$$\begin{aligned} & \min_{\Xi^{(1)}, \Xi^{(2)}} \frac{c(\rho(\Xi^{(1)}, \Xi^{(2)}))}{c_{ref}}, \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} (\sum_{e=1}^{N_e} \rho_e^\alpha [K_e]) \{d\} = [K] \{d\} = \{f\}, \\ \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref}} = \frac{\sum_{e=1}^{N_e} \rho_e V_e}{V_{ref}} = \gamma, \\ \Xi_k^{(1)} \in [\hat{\rho}_{min}, \hat{\rho}_{max}], \\ \Xi_k^{(2)} \in [w_{min}, w_{max}], \\ \forall k = 1, \dots, n_{tot}. \end{array} \right. \quad (4.10) \end{aligned}$$

In Eq. (4.10),  $\rho_e$  is the generic element pseudo-density, i.e.

$$\rho_e = \begin{cases} \rho(u_1^e, u_2^e) = \rho\left(\frac{x_1^e}{a_1}, \frac{x_2^e}{a_2}\right), & \text{if } N = 2, \\ \rho(u_1^e, u_2^e, u_3^e) = \rho\left(\frac{x_1^e}{a_1}, \frac{x_2^e}{a_2}, \frac{x_3^e}{a_3}\right), & \text{if } N = 3, \end{cases} \quad (4.11)$$

where  $x_k^e$  is the  $k$ -th Cartesian coordinate of the element centroid. The objective function is divided by a reference compliance ( $c_{ref}$ ), to obtain a dimensionless value. Similarly, the volume is divided by a reference volume ( $V_{ref}$ ) in the constraint function. Of course, the SIMP penalisation scheme in the form of Eq. (1.5) is still active. It is highlighted that

the new design variables of problem (4.10) are the NURBS control points and weights, no more the element densities like in the standard problem (1.6).

Problem (4.10) can be solved through a suitable gradient-based strategy (refer to Chapter 3). As usual for TO problems, the number of design variables is generally high, thus a numerical evaluation of the derivatives of both objective and constraint functions is strongly discouraged. Therefore, the computation of the derivatives of both objective and constraint functions with respect to the design variables in closed form is needed. This task is achieved by exploiting the NURBS *local support property* (refer to Chapter 2) and the simple chain rule for derivatives calculation. Particularly, the local support related to a control point  $\hat{\rho}_{I_1, I_2}$  in 2D or  $\hat{\rho}_{I_1, I_2, I_3}$  in 3D can be defined as

$$S_\tau = \begin{cases} S_{I_1, I_2} = \left[ U_{I_1}^{(1)}, U_{I_1+p_1+1}^{(1)} \right] \times \left[ U_{I_2}^{(2)}, U_{I_2+p_2+1}^{(2)} \right], & \text{if } N = 2, \\ S_{I_1, I_2, I_3} = \left[ U_{I_1}^{(1)}, U_{I_1+p_1+1}^{(1)} \right] \times \left[ U_{I_2}^{(2)}, U_{I_2+p_2+1}^{(2)} \right] \times \left[ U_{I_3}^{(3)}, U_{I_3+p_3+1}^{(3)} \right], & \text{if } N = 3. \end{cases} \quad (4.12)$$

In Eq. (4.12), the triplet of capital indices  $(I_1, I_2, I_3)$  identifies a specific control point or weight with linear index  $\tau$  through the following relationships:

$$\tau = \begin{cases} I_1 + (I_2 - 1)(n_1 + 1), & \text{if } N = 2, \\ I_1 + (I_2 - 1)(n_1 + 1) + (I_3 - 1)(n_1 + 1)(n_2 + 1), & \text{if } N = 3. \end{cases} \quad (4.13)$$

Let  $Q$  be a generic scalar function to be considered in a TO problem whose gradient with respect to the generic element pseudo-density, i.e.  $\frac{\partial Q}{\partial \rho_e}$ , is known. In the framework of the NURBS-based SIMP approach, it is required to determine the explicit expression of  $\frac{\partial Q}{\partial \Xi_\tau^{(1)}}$  and  $\frac{\partial Q}{\partial \Xi_\tau^{(2)}}$  that symbolically read

$$\frac{\partial Q}{\partial \Xi_\tau^{(1)}} = \sum_{e \in S_\tau} \frac{\partial Q}{\partial \rho_e} \frac{\partial \rho_e}{\partial \Xi_\tau^{(1)}}, \quad (4.14)$$

$$\frac{\partial Q}{\partial \Xi_\tau^{(2)}} = \sum_{e \in S_\tau} \frac{\partial Q}{\partial \rho_e} \frac{\partial \rho_e}{\partial \Xi_\tau^{(2)}}. \quad (4.15)$$

In Eqs. (4.14)-(4.15), only those elements falling within the local support of control point  $\Xi_\tau^{(1)}$  give a non-null contribution to the derivatives. It can be shown (see Appendix A) that the derivatives of the NURBS entity with respect to an assigned control point and the related weight take the form

$$\frac{\partial \rho_e}{\partial \Xi_\tau^{(1)}} = R_\tau^e, \quad (4.16)$$

and

$$\frac{\partial \rho_e}{\partial \Xi_\tau^{(2)}} = \frac{R_\tau^e}{\Xi_\tau^{(2)}} (\Xi_\tau^{(1)} - \rho_e). \quad (4.17)$$



The scalar quantity  $R_\tau^e$  appearing in Eqs. (4.16)-(4.17) is simply the suitable NURBS rational basis function, related to control point  $\Xi_\tau^{(1)}$ :

$$R_\tau^e = \begin{cases} R_{I_1, I_2}(u_1^e, u_2^e), & \text{if } N = 2, \\ R_{I_1, I_2, I_3}(u_1^e, u_2^e, u_3^e), & \text{if } N = 3. \end{cases} \quad (4.18)$$

If the quantity  $Q$  is the total compliance of the structure  $c$ , then the derivative  $\frac{\partial c}{\partial \rho_e}$  is given by Eq. (1.9) and, accordingly, the sought derivatives are

$$\frac{\partial c}{\partial \Xi_\tau^{(1)}} = -\alpha \sum_{e \in S_\tau} \frac{c_e}{\rho_e} R_\tau^e, \quad (4.19)$$

$$\frac{\partial c}{\partial \Xi_\tau^{(2)}} = -\frac{\alpha}{\Xi_\tau^{(2)}} \sum_{e \in S_\tau} c_e \frac{\Xi_\tau^{(1)} - \rho_e}{\rho_e} R_\tau^e. \quad (4.20)$$

Analogously, assumed that  $Q$  is equal to the total volume  $V$  of the structure and taken into account the expression of the derivative  $\frac{\partial V}{\partial \rho_e}$  provided by Eq. (1.10), the derivatives of the volume read

$$\frac{\partial V}{\partial \Xi_\tau^{(1)}} = \sum_{e \in S_\tau} V_e R_\tau^e, \quad (4.21)$$

$$\frac{\partial V}{\partial \Xi_\tau^{(2)}} = \frac{1}{\Xi_\tau^{(2)}} \sum_{e \in S_\tau} (\Xi_\tau^{(1)} - \rho_e) V_e R_\tau^e. \quad (4.22)$$

Some consequences of outstanding importance result from the NURBS-based approach and they can be summarised as follows:

1. The number of design variables is unrelated to the number of elements. In the classic SIMP approach, each element introduces a new design variable. In the NURBS framework, the accuracy of the topology description is characterised solely by the number of points of the control net in 2D and of the hyper-net in 3D, i.e.  $(n_1 + 1) \times (n_2 + 1)$  and  $(n_1 + 1) \times (n_2 + 1) \times (n_3 + 1)$ , respectively;
2. The *local support property* of the NURBS blending functions defines an implicit filter zone. The size of such a filter zone is related to the dimension of the local support of the blending functions, i.e. to the components of the knot vectors, to the number of control points as well as to the degrees of the basis functions. It should be remarked that TO filters create a mutual dependency among the elements densities, i.e. the design variables in the standard SIMP formulation. In the case of NURBS, the inter-dependence is automatically provided due to the NURBS local support, without the need of defining a filter on the mesh elements densities [104, 105].

3. Thanks to the NURBS formalism, a mathematically well-defined description of the geometric boundary of the optimum topology is always available during the iterations of the optimisation process. Moreover, local information, such as the local normal and tangent vectors, can be easily deduced from standard NURBS formulae. Therefore, on the one hand new *ad hoc* constraints can be conceived and, on the other hand, it is possible to improve the handling of optimised topologies at the end of the TO phase.

### 4.3 The algorithm *SANTO* (SIMP And NURBS for Topology Optimisation)

The solution search for problem (4.10) is carried out by means of an in-house code developed at I2M laboratory. The algorithm is called *SANTO* (*SIMP And NURBS for Topology Optimisation*) and it exhibits a modular structure that can be easily interfaced with any FE code (both commercial and in-house software). The architecture of the algorithm is very general and suitable for both 2D and 3D FE models. In the following, the algorithm capabilities are discussed. A flowchart summarising the architecture of *SANTO* is shown in Fig. 4.2. It is noteworthy that the algorithm is developed in MATLAB environment; instead of using an in-house FE code for the computation of the physical responses involved in the TO problem, it has been preferred to prove the potential as well as the versatility of the proposed numerical strategy by interfacing the NURBS-based SIMP algorithm with a commercial FE code. In the context of the numerical examples illustrated in this Chapter, the FE commercial code ANSYS is utilised to calculate the mechanical responses of the structure. A synthetic description of the logical steps is given here below.

1. *Problem Setting.* In this step, the external user has to initialise/modify all the parameters and settings characterising the problem at hand. These quantities are of different nature. Firstly, the user can choose the dimension of the problem (2D or 3D) and the computational domain composition by properly setting the location of both Design Regions (DRs) and Non-Design Regions (NDRs). DRs represent those zones of the FE model wherein the topology will be optimised, whereas NDRs are those parts whose geometry remains unchanged during the optimisation process. Then, the domain can be parametrised through NURBS or B-Spline entities (surfaces or hyper-surfaces for 2D or 3D problems, respectively). The objective and constraints functions must be defined and lower and/or upper bounds for constraints have to be entered. The dimensions of the computation domain ( $a_1$ ,  $a_2$  and eventually  $a_3$ ) are also defined during this step. Finally, different kinds of symmetry (mirror, rotational, translational, etc) can be set by the external user, as illustrated in Fig. 4.3.

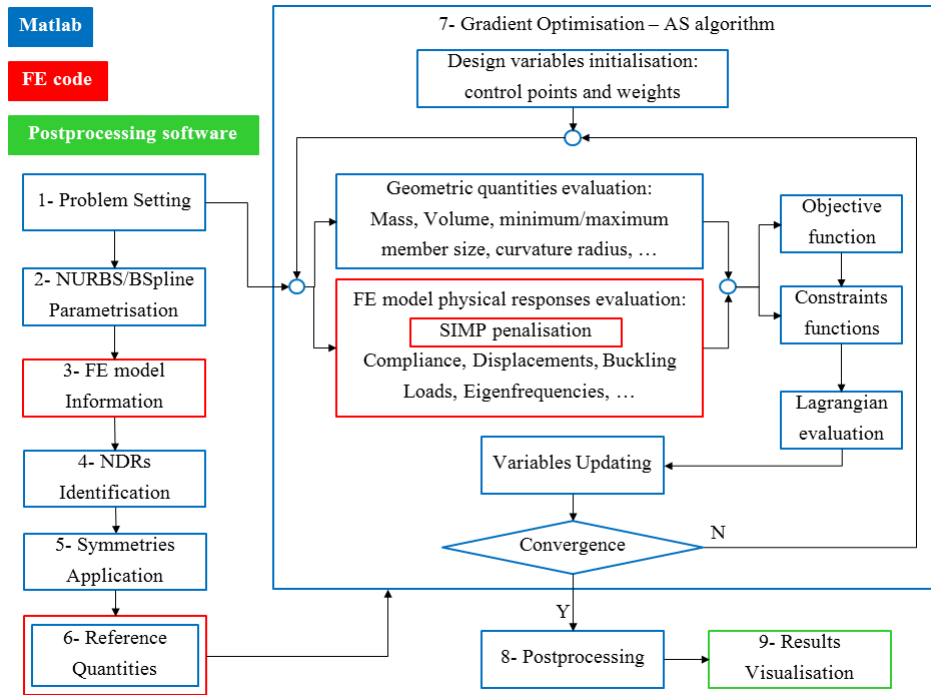


Figure 4.2 – SANTO flowchart

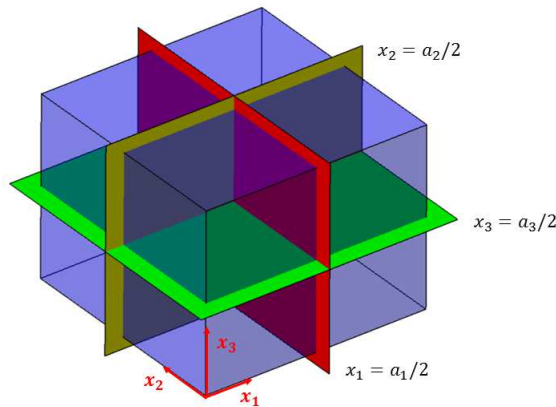


Figure 4.3 – Examples of possible mirror symmetries for 3D TO problems

2. *NURBS/B-Spline parametrisation.* Whatever 2D or 3D structure can be embedded

in a suitable rectangular or prismatic hold-all, respectively, as shown in Fig. 4.4, where the NURBS/B-Spline geometric entity control net has been plotted as well. Moreover, a local reference frame can be set by the user in order to define the NURBS/B-Spline entity dimensionless parameters according to Eq. (4.3). The definition of the NURBS/B-Spline geometric entity can be completed by setting the discrete parameters (degrees and number of control points) as well as the values of the non-trivial components of each knot-vector (see Eq. (2.24)). The previous quantities are just geometric parameters of the NURBS blending functions which are not included in the vector of design variables. If the user does not provide the knot-vector components, they are automatically determined in order to uniformly split the interval  $[0, 1]$ .

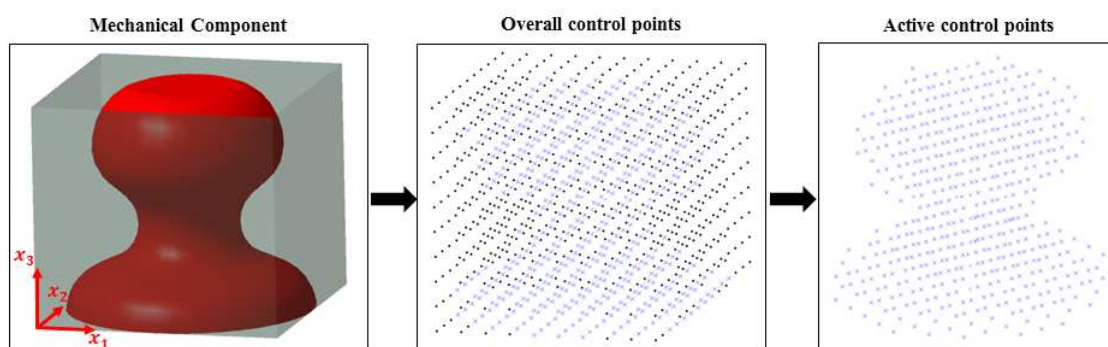


Figure 4.4 – Example of embedding domain, reference system and control points for a 3D component.

3. *FE model Information.* The FE model of the problem at hand must be properly prepared in the external FE code in terms of both geometry and mesh of DRs and NDRs. Once the mesh has been finalised, the elements belonging to DRs are selected and their data are passed to the MATLAB environment. For each element, the relevant pieces of information are its identifier, the physical coordinates of its centre of gravity as well as its volume. The local coordinate system defined for the NURBS parametrisation ( $RF = \{O : x_1, x_2, x_3\}$ ) does not necessarily correspond to the reference system used in the FE software. Therefore, a coordinates transformation is automatically performed before associating the physical coordinates of the elements centres of gravity to the NURBS dimensionless parameters  $u_j$ , according to Eq. (4.3).
4. *NDRs Identification.* As previously remarked, the proposed procedure is very general due to the possibility of embedding any structure in a regular compact subspace of the Euclidean space. Moreover, not all the zones of the structure have to be optimised and this requirement can be fulfilled by forecasting proper NDRs within the

FE model at specific locations. Therefore, including all the control points densities and the corresponding weights of the NURBS/B-Spline surface/hyper-surface (even those control points whose local support falls within NDRs) among the design variables is totally meaningless and useless. To this purpose, before launching the optimisation calculation, a dedicated function checks all control points local supports. Let the local support of the generic control point be considered, i.e.  $S_\tau$ , defined in Eq. (4.12): if the dimensionless coordinates ( $(u_1^e, u_2^e)$  in 2D and  $(u_1^e, u_2^e, u_3^e)$  in 3D) of at least one element centre of gravity belong to  $S_\tau$ , then the corresponding control point and the related weight are inserted in the design variables array; otherwise, they are deleted from this array. The discarded control points values are set to  $\hat{\rho}_{max}$  or to  $\hat{\rho}_{min}$  for solid phase and void phase, respectively. They are not modified any more all along the optimisation process. The result of this selection operation can be observed in Fig. 4.4, where the control points tuning the NURBS/B-Spline entity values have been highlighted in blue.

5. *Symmetries Application.* As the previous step, symmetries can be interpreted as variables saving from a computational viewpoint. Only independent control points and weights are effectively collected in the variables array: in fact, geometrical symmetries correspond to peculiar symmetric conditions on the control points coordinates matrices.
6. *Reference Quantities.* Both geometrical and physical responses used to obtain dimensionless objective and constraint functions are set here. Their definition is not unique and the algorithm allows the external user to define the reference quantities according to the problem at hand. However, pre-set strategies can be selected for the most common TO problems. For example, when the volume is considered as objective or constraint function, the reference volume is the total volume of the different DRs. If the classic problem of compliance minimisation with an imposed volume fraction is faced, as in the case of Eq. (4.10), the reference compliance ( $c_{ref}$ ) is the compliance of a fictitious structure wherein all the DRs elements have the same density. Since the TO problem will be solved through one of the gradient-based algorithms described in Chapter 3, the convergence is achieved only when one suitable criterion is met. Excepted for the maximum number of iterations, the other criteria (no improvement of the objective function, no change of the optimisation variables, almost-null gradient norm) are meaningful only if the objective/constraints functions are dimensionless (see Chapter 3).
7. *Gradient Optimisation.* By default, the optimisation is carried out through the AS algorithm, which belongs to the MATLAB *fmincon* algorithms family. The theoretical basis of this algorithm (a generalised quasi-Newton algorithm for constrained minimisation problems) can be found in [70, 76] and it has been briefly discussed in Chapter 3. For the sake of completeness, some details are here recalled about the

application of *fmincon* algorithms in the framework of SANTO. As far as the initialisation is concerned, suitable values for the design variables must be provided in such a way that the starting solution is feasible (the full set of optimisation constraints is met). Indeed, the initialisation is closely related to the choice of the *Reference Quantities*, as discussed in the previous step. Of course, the initialisation is a crucial point and it could become a real challenge when a huge number of constraints is considered for the problem at hand: in fact, the higher is the constraints number, the more complicated is finding out a feasible starting point. Once the AS algorithm is run, the objective and constraint functions declared in the *Problem Setting* block are called from two dedicated libraries. The first library contains functions related to specific mechanical quantities (e.g. compliance, eigenfrequencies, buckling, etc.): their evaluation requires to call the user-defined external FE model. In order to properly evaluate mechanical quantities, an *ad hoc* penalisation law is used: the SIMP power law of Eq. (1.5) could be a good choice in the simple case of the compliance problem. Other possible penalisation laws are discussed in Chapter 6. The second library contains only geometric quantities, as volume/mass, minimum and maximum member size or the curvature radius (see Chapter 5). In each function, the calculation of derivatives is included. Then, the Lagrangian functional evaluation and the approximation of Hessian matrix are performed. The design variables array can be updated by means of standard AS methodologies and a convergence check is performed (see Chapter 3). The aforementioned procedure is repeated until a convergence criterion is attained.

8. *Postprocessing.* As usual for density-based TO methods, a threshold operation is necessary at the end of the optimisation to retrieve the optimised boundary. When the TO is performed in the NURBS framework, this task is trivial for 2D applications, as illustrated in Fig. 4.5. Firstly, the 3D NURBS surface representing the pseudo-density field is transferred to a CAD software through a standard IGS file. Secondly, a rectangular domain (size  $a_1 \times a_2$ ) is drafted at an altitude corresponding to the threshold value for the density. This threshold value  $\rho_{th}$  is chosen in such a way that the optimisation constraints are met. Finally, the rectangular domain and the NURBS surface are intersected. The result of this operation is the 2D optimum topology shown in Fig. 4.5. For 2D applications, the final result is fully CAD-compatible and it requires neither further geometric manipulation nor reconstruction operations. Eventually, the consistency of the results can be easily checked by transferring the 2D optimised geometry to the FE solver via a new IGS file. A similar procedure should be followed in the case of 3D structures. However, the complexity of the intersection operation increases. The 4D NURBS/B-Spline hyper-surface must be intersected with a suitable hyper-plane corresponding to the threshold value for the density. While for 2D structures the intersection takes place in a CAD environment and the results are directly available in terms of geometric

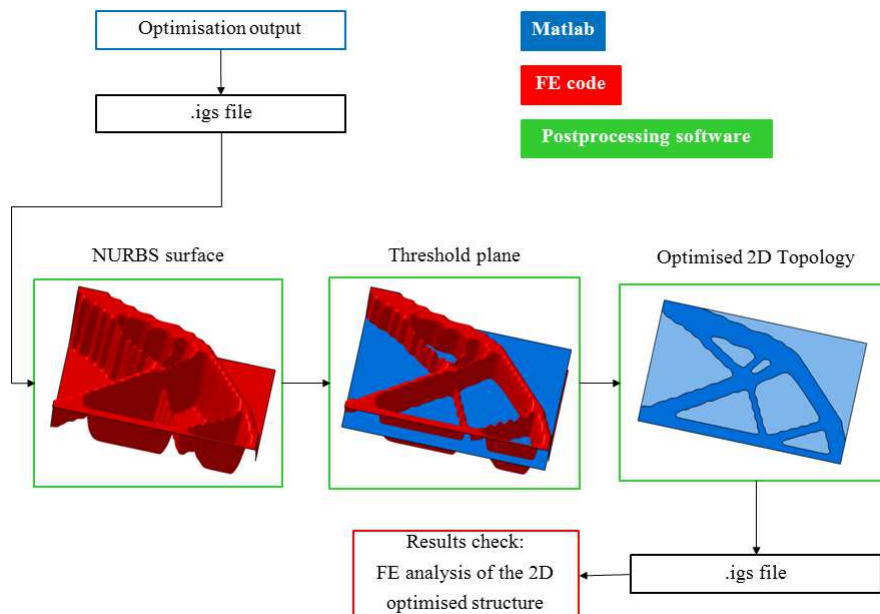


Figure 4.5 – Postprocessing procedure for 2D structures.

entities (the boundary lines), for 3D components such an intersection can be easily performed only in a dedicated environment (e.g. MATLAB in this study) and the result is a *Delaunay triangulation*, which is saved in a STL file (see Fig. 4.6). On the other hand, since a NURBS-based description of the pseudo-density is available, one can imagine to extract a 3D representation of the topology boundary in a better format than a simple STL file. In this sense, one possibility could be developing the 4D NURBS entity for CAD environment. This should allow for mimicking, in the 3D case, the postprocessing step proposed for 2D structures. However, this task is anything but trivial. Another possibility to retrieve the 3D structure boundary in terms of CAD entities (i.e. suitable surfaces and not in the form of a *Delaunay triangulation*) could be the use of a very general and automated surface approximation technique for surfaces of *genus n* [110]. In this context, the problem of retrieving a CAD compatible representation of the boundary can be stated as a classical surface fitting problem generalised to the case of surfaces of *genus n*. The goal of such a problem is the minimisation of the distance between a set of sampling points belonging to the 3D topology boundary and those belonging to a set of pertinent NURBS/B-Spline surfaces opportunely linked (in order to correctly describe the boundary). Nevertheless, the most complicated aspect when dealing with such a problem is to determine of a proper *surface parametrisation*. Surface paramet-

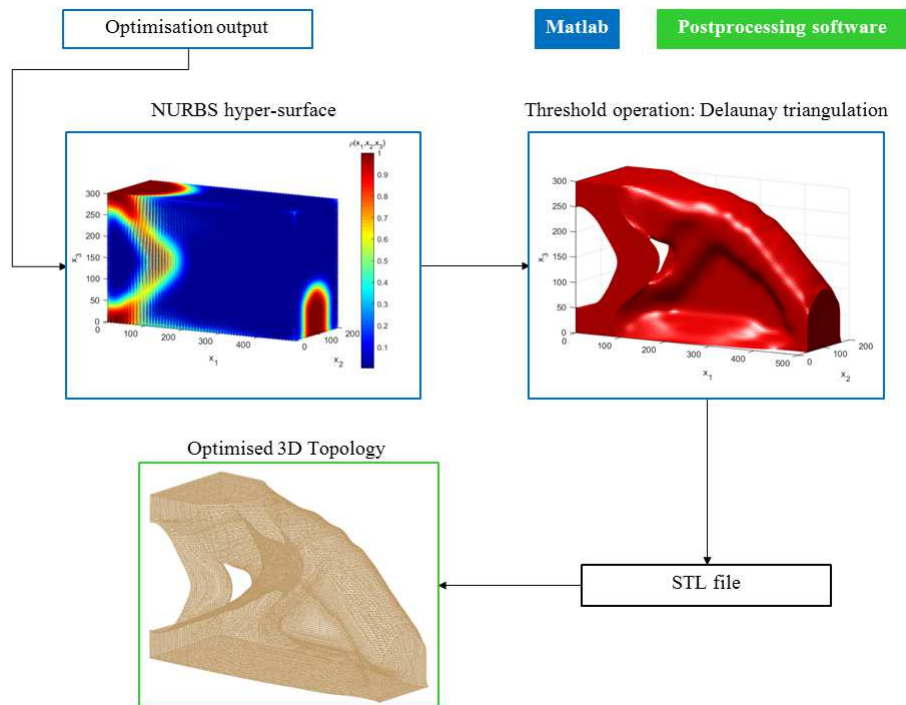


Figure 4.6 – Postprocessing procedure for 3D structures.

risation for a general surface of *genus*  $n$  is a very challenging problem which does not present a unique solution. In particular, good results can be achieved for open surfaces [111], but parametrising whatever closed surface having a general shape is a really hard problem [110]. A deep discussion on these topics is out of the scopes of this Chapter. Some related aspects will be discussed in Chapter 7.

9. *Results Visualisation.* As previously stated, the 2D optimised geometry can be easily stored in a IGS file and it is naturally representable in a CAD environment. The *Delaunay triangulation* obtained for 3D problems can be converted in a standard STL format and can be imported in a CAD for further operations. It must be pointed out that the result of the TO in the NURBS framework for 3D structures is not fully CAD compatible, as the STL native format is composed of a set of triangles defining the topology boundary (which are surfaces) with the related normal vector. However, since the triangulation derives from a geometric entity, i.e. the intersection between a hyper-plane and a NURBS hyper-surface, it exhibits fair features: no degenerated or missing triangles are present within the STL format.



## 4.4 Results Discussion

The effectiveness of the proposed strategy is proven through some meaningful 2D and 3D benchmarks. For each considered case, the optimum topology is shown after the cutting operation through the threshold plane or hyper-plane described in section 4.3: therefore, it is represented through 2D rebuilt boundary or in the form of a Delaunay triangulation.

Lower and upper bounds for both the sets of variables involved in the NURBS-based SIMP method are fixed as follows:

- $\Xi_k^{(1)} \in [\hat{\rho}_{min}, \hat{\rho}_{max}]$ ,  $\hat{\rho}_{min} = 10^{-3}$ ,  $\hat{\rho}_{max} = 1$ ,
- $\Xi_k^{(2)} \in [w_{min}, w_{max}]$ ,  $w_{min} = 1/2$ ,  $w_{max} = 10$ ,

$\forall k = 1, \dots, n_{tot}$ .

These bounds do not change and they will not be specified in the problems statements related to each benchmark for the sake of compactness.

Different aspects of the NURBS-based SIMP method are investigated:

- the influence of the NURBS discrete parameters (degrees, number of control points) on the final optimum topology is studied;
- a comparison between the results provided by a NURBS entity and a B-Spline one is carried out on both 2D and 3D benchmarks (in this way, the effects of the NURBS weights on the final optimum topology can be understood);
- the TO problem performed on the considered benchmarks is solved by means of the commercial software Altair OptiStruct<sup>®</sup> and results are compared to those obtained through the proposed NURBS-based SIMP approach;
- some particular cases including NDRs and symmetries are investigated as well.

In the followings, the problem formulation for each benchmark will be stated in the most general framework, wherein NURBS entities are considered (therefore both the arrays  $\Xi^{(1)}$  and  $\Xi^{(2)}$  are involved in the TO). When just B-Spline are employed, the array  $\Xi^{(2)}$  is implicitly removed from the design variables and all weights are set to 1.

### 4.4.1 The 2D benchmark

The problem of the compliance minimisation with an imposed volume fraction is considered here for an aluminium cantilever plate. All geometric and material data are provided in the caption of Fig. 4.7.

The mathematical statement of the related TO problem is given by

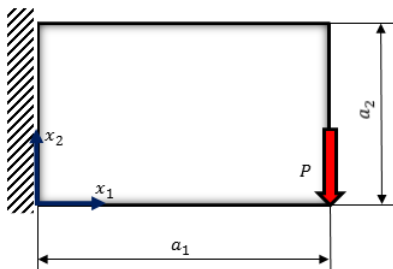


Figure 4.7 – Cantilever plate problem -  $a_1 = 320$  mm,  $a_2 = 200$  mm, Thickness  $t = 2$  mm, Young's Modulus  $E = 72000$  MPa, Poisson's ratio  $\nu = 0.33$ , Load  $P = 1000$  N.

$$\begin{aligned} & \min_{\Xi^{(1)}, \Xi^{(2)}} \frac{c(\rho(\Xi^{(1)}, \Xi^{(2)}))}{C_{ref}}, \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} [K]\{U_{FEM}\} = \{F\}, \\ \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref}} = 0.4. \end{array} \right. \end{aligned} \quad (4.23)$$

In Eq. (4.23),  $V_{ref} = a_1 a_2 t = 1.28 \times 10^5$  mm<sup>3</sup> is the total volume of the plate, whilst  $c_{ref} = 3520.43$  Nmm is the compliance of the structure when a uniform pseudo-density field  $\rho = 0.4$  affects the elements stiffness matrix. The FE model of the rectangular domain is discretised by means of Ansys SHELL181 elements, i.e. shell elements with 4 nodes and 6 DOFs per node. After a preliminary check on the convergence of the results, the size of the *mapped* mesh of the rectangular domain has been chosen equal to  $80 \times 50$  elements. The equality constraint is split in two inequality constraints by considering a tolerance of 0.005 on the value of  $f$ . Then, the volume fraction constraint will be met if  $0.395 < \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref}} < 0.405$ .

#### 4.4.2 Results for 2D problems: sensitivity to the NURBS surface degrees, number of control points and weights

Problem (4.23) has been solved through the algorithm described in section 4.3 for three values of surface degrees and three different values of the overall number of control points:

- degrees:  $p_1, p_2 = 2, 3, 4$ ;
- number of control points:  $(n_1 + 1) \times (n_2 + 1) = 16 \times 10, 32 \times 20, 48 \times 30$ .

Results are provided in terms of compliance  $c$  and volume fraction  $V/V_{ref}$  in captions of Figs. 4.8-4.13. The CAD compatibility of NURBS surfaces is fully exploited here, so useless elements have been easily cut off through the threshold operation: the objective function  $c$  and the volume constraint are evaluated on the “true structure” (optimum topology) instead of on the meshed reference domain (wherein “void” elements still hold on together with the “material” elements), that is meaningless from an engineering viewpoint.

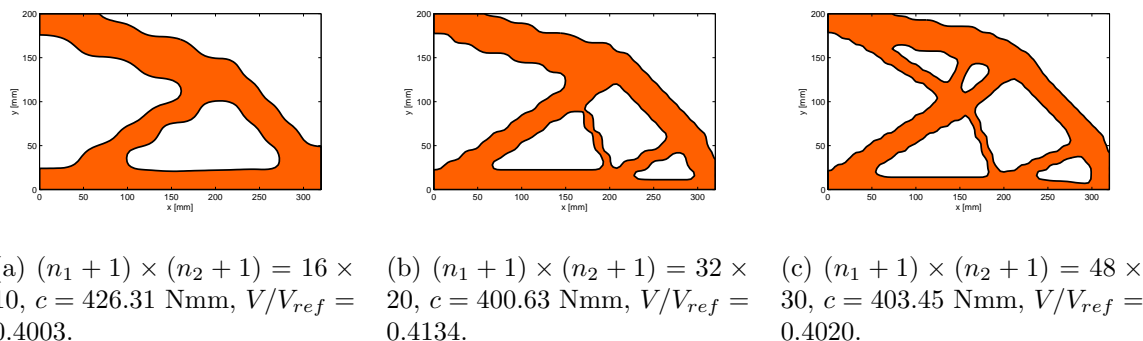


Figure 4.8 – B-Spline results for  $p_1, p_2 = 2$

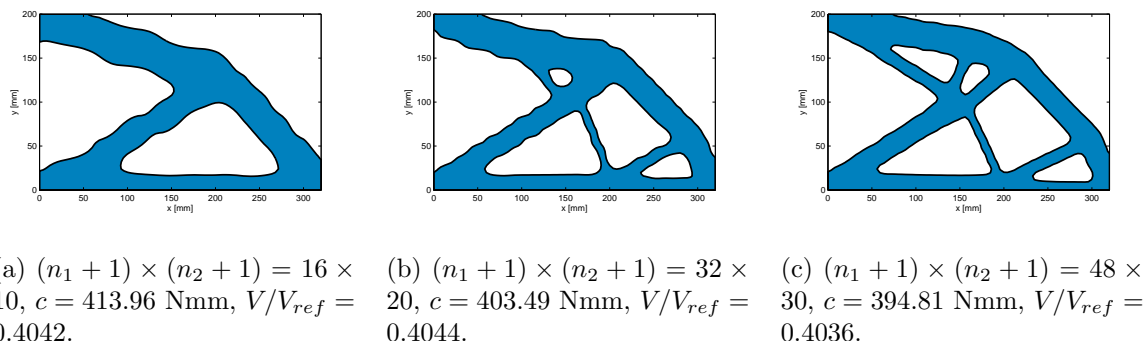
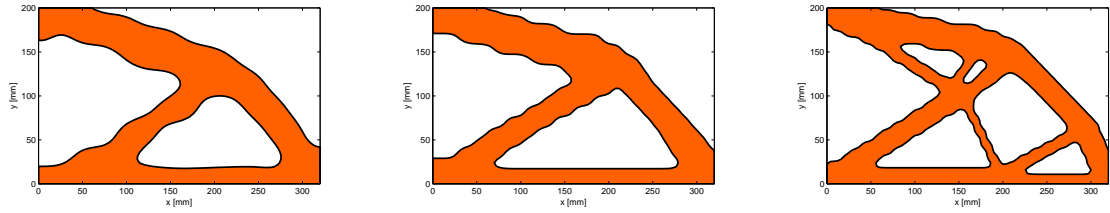


Figure 4.9 – NURBS results for  $p_1, p_2 = 2$

Numerical results concerning the true compliance as a function of the number of control points are synthetically plotted in Fig. 4.14. It is interesting to compare this compliance with the compliance calculated on the reference domain at the end of the solution phase before the cutting operation, i.e. the compliance of the whole rectangular domain constituted by all elements with the respective pseudo-density value (it is referred as “projected compliance”, see Fig. 4.15).

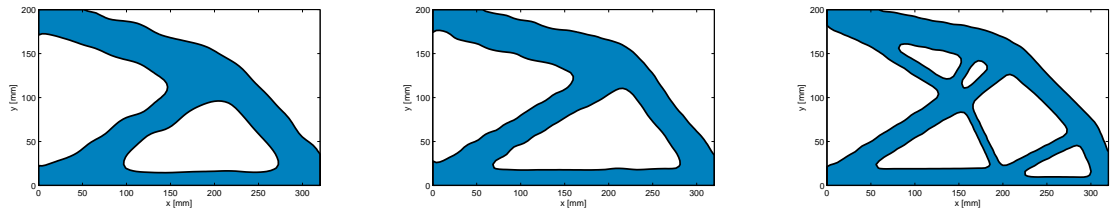
The following remarks arise from the analysis of the numerical results:

#### 4.4. Results Discussion



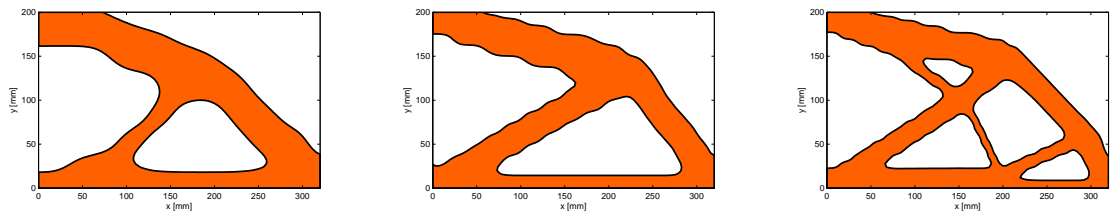
(a)  $(n_1 + 1) \times (n_2 + 1) = 16 \times 10$ ,  $c = 432.29$  Nmm,  $V/V_{ref} = 0.4011$ .  
 (b)  $(n_1 + 1) \times (n_2 + 1) = 32 \times 20$ ,  $c = 408.37$  Nmm,  $V/V_{ref} = 0.3994$ .  
 (c)  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$ ,  $c = 402.39$  Nmm,  $V/V_{ref} = 0.4025$ .

Figure 4.10 – B-Spline results for  $p_1, p_2 = 3$



(a)  $(n_1 + 1) \times (n_2 + 1) = 16 \times 10$ ,  $c = 416.96$  Nmm,  $V/V_{ref} = 0.4048$ .  
 (b)  $(n_1 + 1) \times (n_2 + 1) = 32 \times 20$ ,  $c = 404.07$  Nmm,  $V/V_{ref} = 0.4050$ .  
 (c)  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$ ,  $c = 394.45$  Nmm,  $V/V_{ref} = 0.4047$ .

Figure 4.11 – NURBS results for  $p_1, p_2 = 3$



(a)  $(n_1 + 1) \times (n_2 + 1) = 16 \times 10$ ,  $c = 446.71$  Nmm,  $V/V_{ref} = 0.4020$ .  
 (b)  $(n_1 + 1) \times (n_2 + 1) = 32 \times 20$ ,  $c = 407.31$  Nmm,  $V/V_{ref} = 0.4032$ .  
 (c)  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$ ,  $c = 401.69$  Nmm,  $V/V_{ref} = 0.4009$ .

Figure 4.12 – B-Spline results for  $p_1, p_2 = 4$

- a) The boundary of the topologies obtained through a NURBS-based representation of the fictitious density function are smoother than those obtained by means of B-Spline-based description, see Figs. 4.8-4.13. Moreover, as clearly shown in Fig. 4.15,

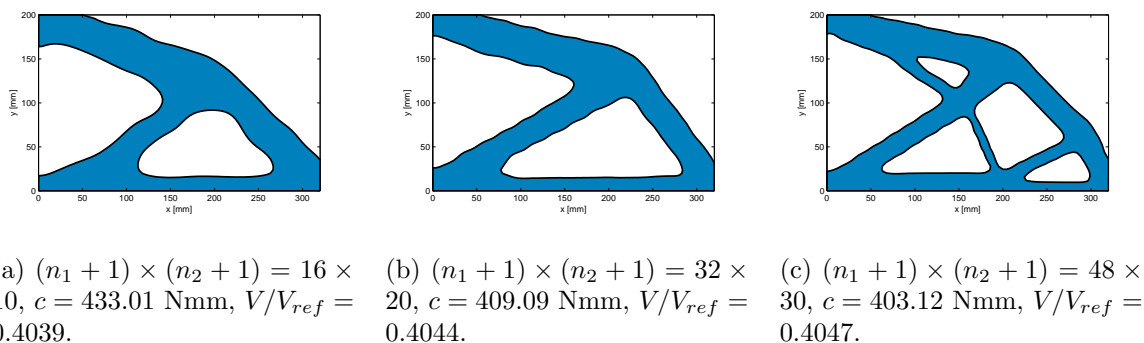


Figure 4.13 – NURBS results for  $p_1, p_2 = 4$

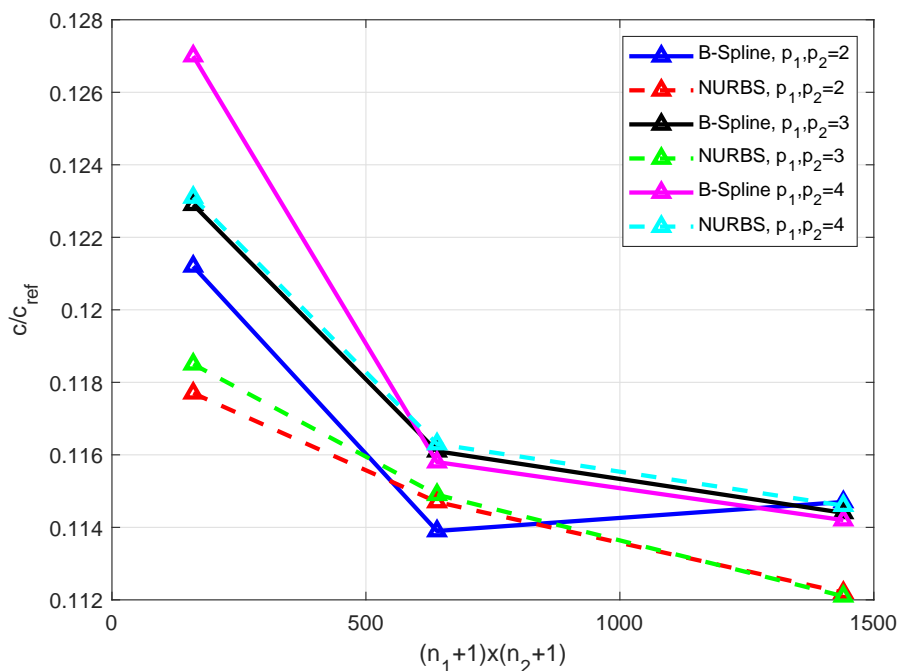


Figure 4.14 – Cantilever plate problem - True compliance trends.

the optimum topology obtained using NURBS surfaces takes lower values of the projected compliance when compared to those resulting from a B-Spline-based representation. This fact justifies the utilisation of the more general NURBS surfaces formalism. However, when looking at the true compliance (Fig. 4.14), NURBS have still significantly better performances than the respective B-Splines only when the number of control points is kept small. If the number of control points increases, even if NURBS topologies are still smoother than B-Spline topologies, the decrease of the objective function disappears and, sometimes, a B-Spline solution could be

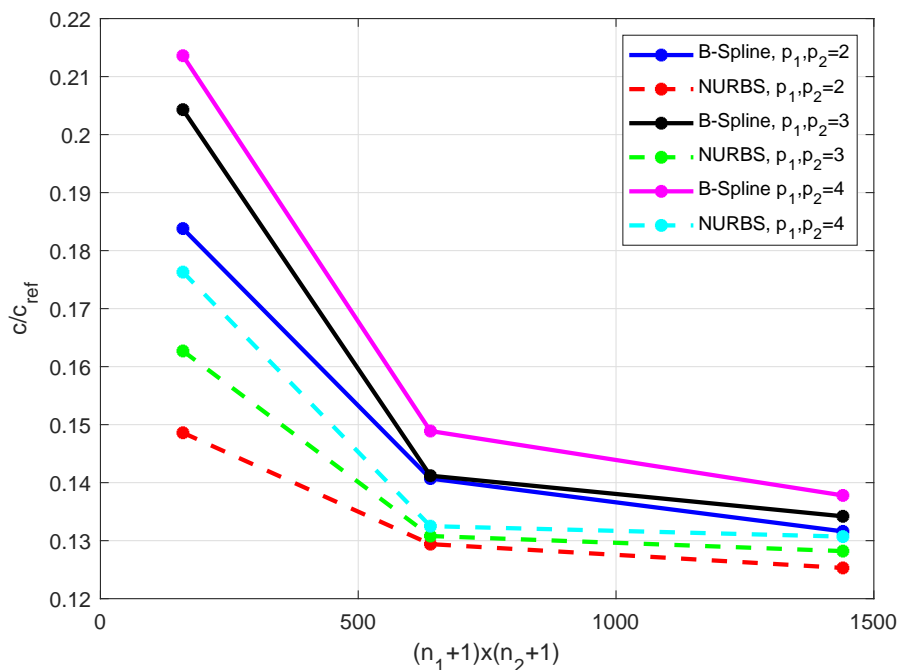


Figure 4.15 – Cantilever plate problem - Projected compliance trends.

better than a NURBS solution (refer to the cases of Fig. 4.12b and Fig. 4.13b). Thus, the utilisation of NURBS instead of B-Spline surfaces in TO must be carefully assessed.

- b) Taking inspiration from [104], the behaviour of the solutions has been investigated by varying both the number of control points  $(n_1 + 1) \times (n_2 + 1)$  and the degrees of the surface  $(p_1, p_2)$ . These parameters affect the dimension of the local support of the blending functions. The local support, in the context of TO, behaves as a *filter zone*, i.e. a region of the reference domain wherein the densities of “neighbour elements” are interdependent. Such a filter zone is sought in standard density-based algorithms to prevent the checker-board effect [5]. Therefore, the NURBS-based SIMP method naturally ensures an implicitly-defined filter zone without the need of introducing artificial distance-based filters as in [5]. This aspect is strictly related to the *local support property* of NURBS blending functions [68]: the filter dimension increases if the degrees increase or if the number of control points decreases. Conversely, as the degrees decrease and the number of control points increases, the filter gets smaller. Taking into account these considerations, it seems natural that thinner topology elements (branches) are allowed when the size of the filter zone decreases. This trend is evident in B-Spline and NURBS benchmarks, see Figs. 4.8-4.13: B-Spline and NURBS surfaces sharing the same degrees, knot-vectors components and same

control point coordinates have the same filter zone. Furthermore, the smaller is the filter size, the lower is the true compliance value.

- c) Several remarks arise from a deeper investigation of results shown in Figs. 4.14-4.15. Firstly, the projected compliance trend is smoother than the respective true compliance trend. This fact is a consequence of the cut operation in the postprocessing phase, which constitutes a sort of “discontinuity” from a mathematical viewpoint. Indeed, this cutting operation can lead to a pseudo-optimum solution: the objective function decreases but constraints are not met (see for example the solution of Fig. 4.8b). Secondly, the projected compliance exhibits an early phase of a plateau (Fig. 4.15), that disappears in the graph of the true compliance of Fig. 4.14. Actually, considering the decrease of the objective function as long as the number of control points increases, it can be stated that the true compliance does not depend any more on the number of control points beyond a certain threshold value. In other words, increasing the number of design variables does not necessarily imply better performances, even if the topology could appear different (see sub-figures b and c of Figs. 4.8-4.13). This result allows for introducing a sort of “design rule”: the number of control points (design variables) can be chosen and tuned as a compromise between accuracy in the topology description and time saving in running the algorithm.
- d) It is noteworthy that the projected compliance is always greater than the respective true compliance: this point is of paramount importance because it means that the NURBS-based SIMP approach is conservative, unlike the classical SIMP method.

### 4.4.3 Results for 2D problems: comparison between classical and NURBS-based SIMP approaches

The previous results have been validated through a comparison with those obtained when solving problem (4.23) (Fig. 4.7) *via* the commercial software Altair OptiStruct<sup>®</sup> [11]. Of course, the same conditions have been set, namely the same reference domain has been meshed through  $80 \times 50$  PSHELL Elements [11]. Being the software founded on a classic density-based method, the design variables are the element densities. Moreover, the final rebuilt optimum topology is obtained by means of a smoothing phase (OSSmooth module of Altair Hyperworks package). It is pointed out that OptiStruct<sup>®</sup> needs a minimum member size constraint to properly work (see Chapter 1 and Chapter 5): in fact, the minimum member size acts as a filter for TO. In this case, a minimum member size  $d_{min} = 12$  mm has been chosen (i.e. 3 times the mesh size, as suggested by OptiStruct<sup>®</sup>'s reference guide [11]).

Let us consider the B-Spline-based solution of Fig. 4.10c and the NURBS-based solution of Fig. 4.11c with  $p_1 = p_2 = 3$  and  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$ . Both

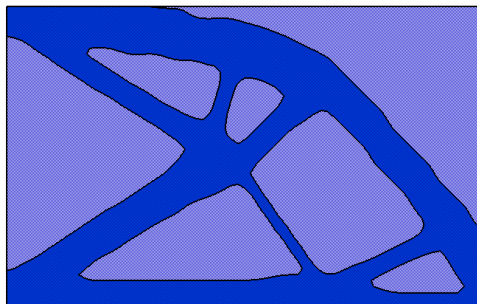


Figure 4.16 – Hyperworks-OptiStruct<sup>®</sup> solution of the cantilever plate problem:  $c = 398.66 \text{ Nmm}$ ,  $V/V_{ref} = 0.3992$ .

the topologies, as well as the OptiStruct<sup>®</sup> solution, meet the volume constraint. As far as the true compliance is concerned, for the OptiStruct<sup>®</sup> solution it is obtained  $c = 398.66 \text{ Nmm}$  (4000 design variables), for the B-Spline-based solution  $c = 402.39 \text{ Nmm}$  (1440 design variables) and for the NURBS-based solution  $c = 394.45 \text{ Nmm}$  (2880 design variables). Consequently, it can be stated that the NURBS-based algorithm and the software OptiStruct<sup>®</sup> provide consistent results. Finally, it can be asserted that in the framework of the NURBS-based SIMP approach the optimum topology (showing equal or superior performances when compared to those provided by the classical SIMP approach) is obtained with a considerable reduction in the number of design variables.

#### 4.4.4 Results for 2D problems: influence of Non-Design Regions

In order to show the versatility of the proposed method, the effects of two prescribed NDRs are investigated in this section. The problem of Fig. 4.7 has been slightly changed, as it is shown in Fig. 4.17: a circular sector (red zone in which  $\rho = 1$ , centred at  $x_1 = a_1/2$ ,  $x_2 = a_2/2$  with  $R_{int} = 40 \text{ mm}$ ,  $R_{ext} = 50 \text{ mm}$ ) surrounding a “void” circle (yellow zone wherein  $\rho = 0$ ) has been defined over the rectangular plate. The same mathematical statement of problem (4.23) still holds in this case.

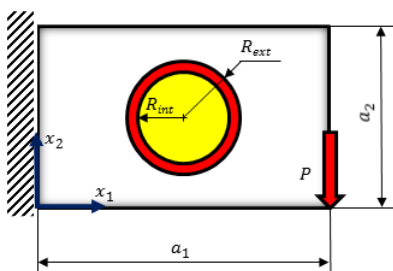
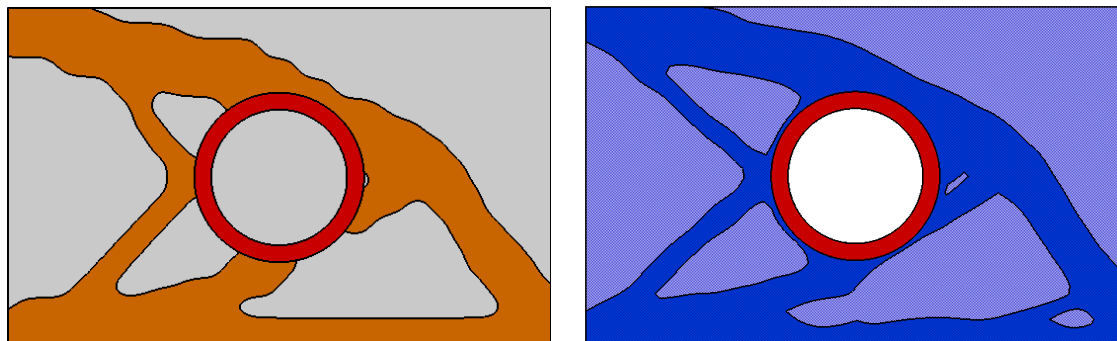


Figure 4.17 – Cantilever plate problem with NDRs: design domain in white, prescribed material NDR in red, prescribed void NDR in yellow.



In this case, a B-Spline surface is used to get the solution, its parameters are  $p_1 = p_2 = 3$  and  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$ .



(a) NURBS-based SIMP solution,  $c = 425.95$  Nmm,  $V/V_{ref} = 0.3997$ .  
 (b) Hyperworks-OptiStruct<sup>®</sup> solution,  $c = 406.37$  Nmm,  $V/V_{ref} = 0.4085$ .

Figure 4.18 – Cantilever plate problem with NDRs.

The optimum solutions provided by both the NURBS-based SIMP approach and the classic SIMP method are illustrated in Fig. 4.18; the values of the true compliance and of the constraint on the volume fraction are reported in the figure captions. As it can be easily noticed, the percentage difference in terms of the objective function is lower than 5%, while the overall volume fraction provided by the NURBS-based SIMP approach is lower than that resulting from the classic SIMP approach: in the latter case, the constraint on the volume fraction is not met. Therefore, these solutions (which are slightly different in terms of topological branches) are “equivalent” and consistent from an engineering viewpoint with a considerable difference: in this case the fictitious density field represented through a B-Spline surface is characterised “only” by 1440 design variables while the OptiStruct<sup>®</sup> model is characterised by 2392 design variables.

#### 4.4.5 Results for 2D problems: influence of a symmetry constraint

In this section, the problem shown in Fig. 4.7 and described by Eq. (4.23) is enhanced with a further geometric constraint: the topology is forced to be symmetric with respect to the plane  $x_2 = a_2/2$ . In this case the B-Spline and NURBS main parameters are set as follows:  $p_1 = p_2 = 3$  and  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$ . The numerical values of compliance and volume fraction are collected in captions of Fig. 4.19a and Fig. 4.19b. The same problem has been solved in OptiStruct<sup>®</sup> environment and results are depicted in Fig. 4.19c. The symmetry constraint allows for halving the number of variables, so the B-Spline solution is characterised by a compliance  $c = 454.80$  Nmm with 720 design variables, the NURBS one by a compliance  $c = 456.47$  Nmm with 1440 design variables,

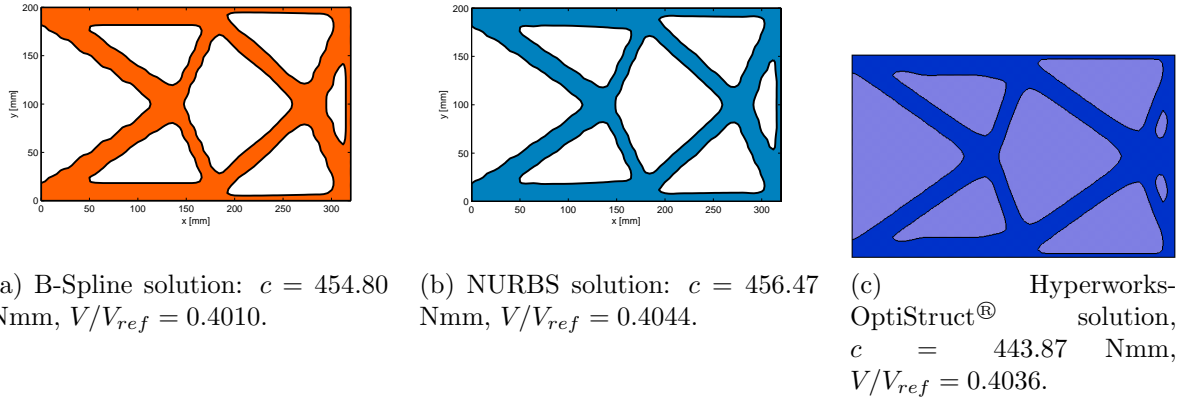


Figure 4.19 – Cantilever plate problem with symmetry constraint.

whilst for the OptiStruct<sup>®</sup> solution it is obtained  $c = 443.87$  Nmm with 2000 design variables.

#### 4.4.6 The 3D benchmarks

The following two proposed benchmarks have been chosen in order to provide an exhaustive idea of the behaviour of the presented algorithm on 3D problems involving domains with different aspect ratios. In the first case, one dimension is dominant with respect to the other two; in the second case, the three dimensions are the same. For both problems, the reference volume  $V_{ref}$  is the volume of the DR and the reference compliance  $c_{ref}$  is the compliance of the structure with an evenly distributed fictitious density equal to the imposed volume fraction  $\gamma$ .

The first structure to be optimised is a solid prism, whose sizes are  $a_1 = 500$  mm,  $a_2 = 100$  mm and  $a_3 = 160$  mm. The base material is Ti6Al4V ( $E = 110000$  MPa and  $\nu = 0.34$ ). The solid is oriented in the local reference system as shown in Fig. 4.20. A thin layer of material corresponding to  $x_3 \in [0, 10]$  mm is constrained to be a NDR and an uniformly distributed load  $p_s = 0.4$  MPa is applied at  $x_3 = 0$  mm. The yellow walls at  $x_1 = 0$  mm and  $x_1 = 500$  mm of the NDR are clamped (refer to Fig. 4.20). Static FE analyses have been carried out through  $50 \times 10 \times 16$  SOLID185 elements (linear formulation, 8 nodes, 3 DOFs per node). The TO problem can be stated as a classical

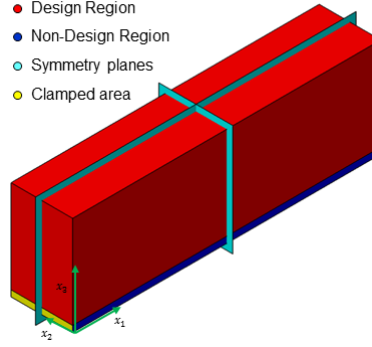


Figure 4.20 – Problem settings of the first benchmark

constrained non-linear programming problem (CNLPP) as follows:

$$\begin{aligned}
 & \min_{\Xi^{(1)}, \Xi^{(2)}} \frac{c(\rho(\Xi^{(1)}, \Xi^{(2)}))}{c_{ref-1}}, \\
 & \text{subject to:} \\
 & \left\{ \begin{array}{l}
 (\sum_{e=1}^{N_e} \rho_e^\alpha [K_e]) \{d\} = [K] \{d\} = \{f\}, \\
 \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref-1}} = \gamma_1 = 0.2133, \\
 \text{symmetry with respect to } x_1 = a_1/2, \\
 \text{symmetry with respect to } x_2 = a_2/2,
 \end{array} \right. \quad (4.24)
 \end{aligned}$$

where  $c_{ref-1} = 4957.13 \text{ Nmm}$  and  $V_{ref-1} = 7.5 * 10^6 \text{ mm}^3$ .

The second problem is the TO of a cube ( $a_1 = a_2 = a_3 = 300 \text{ mm}$ ) clamped at one face and undergoing a downward surface load of  $0.4 \text{ MPa}$  distributed on the patch highlighted in blue in Fig. 4.21. For this problem, only one symmetry constraint is imposed on the NURBS/B-Spline hyper-surface with respect to the plane  $x_2 = a_2/2$ . The bulk material, in this case, is aluminium ( $E = 72000 \text{ MPa}$  and  $\nu = 0.33$ ). Those elements which are attached to the load patch are considered as a NDR. The FE analysis is carried out with  $40 \times 40 \times 40$  SOLID185 Ansys elements.

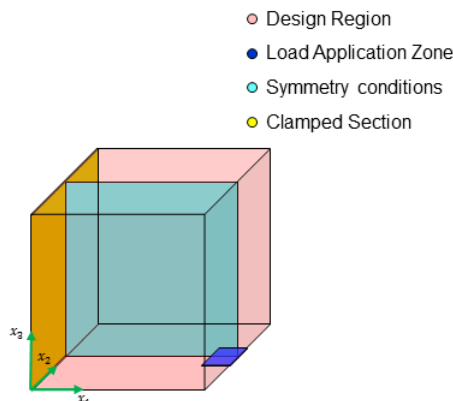


Figure 4.21 – Problem settings of the second benchmark

Analogously to the first benchmark, the optimisation problem is

$$\begin{aligned} & \min_{\Xi^{(1)}, \Xi^{(2)}} \frac{c(\rho(\Xi^{(1)}, \Xi^{(2)}))}{c_{ref-2}}, \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} (\sum_{e=1}^{N_e} \rho_e^\alpha [K_e]) \{d\} = [K] \{d\} = \{f\}, \\ \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref-2}} = \gamma_2 = 0.25, \\ \text{symmetry with respect to } x_2 = a_2/2, \end{array} \right. \end{aligned} \quad (4.25)$$

with  $c_{ref-2} = 672.33$  Nmm and  $V_{ref-2} = 2.7 \times 10^7$  mm<sup>3</sup>. A sensitivity study to the NURBS degrees and number of control points has been carried out for both benchmarks but the results of the sensitivity analysis have been reported only for benchmark related to problem (4.24) for sake of brevity.

#### 4.4.7 Results for 3D problems: sensitivity to the NURBS hyper-surface degrees, number of control points and weights

Problem (4.24) is solved for several values of blending function degrees and number of control points. All the tested combinations of basis functions degrees and number of control points are chosen among the following values

- degrees:  $p_1, p_2, p_3 = 2, 3, 4$ ;
- number of control points:  $(n_1 + 1) \times (n_2 + 1) \times (n_3 + 1) = 36 \times 6 \times 10, 42 \times 8 \times 12, 46 \times 10 \times 14$ .

Taken into account the double symmetry, the number of design variables ( $n_{var}$ ) is equal to a quarter or to half of the overall number of control points if a B-Spline or a NURBS hyper-surface is used, respectively.

Results are shown in Figs. 4.22-4.30. Only the ratio between the resulting compliance and the reference compliance is given in figures captions, whilst the optimisation constraint on the volume is met for all the configurations with a percentage error inferior to 0.5%.

For the sake of brevity, results are synthetically reported in the graphs of Fig. 4.31, where the values of  $c(\rho)/c_{ref}$  are plotted as a function of iterations.

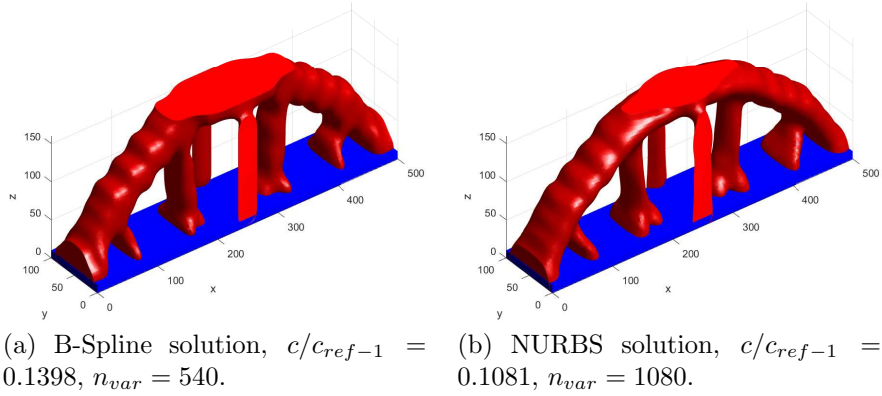


Figure 4.22 – Benchmark 1,  $p_1 = p_2 = p_3 = 2$ ,  $n_{tot} = 36 \times 6 \times 10$

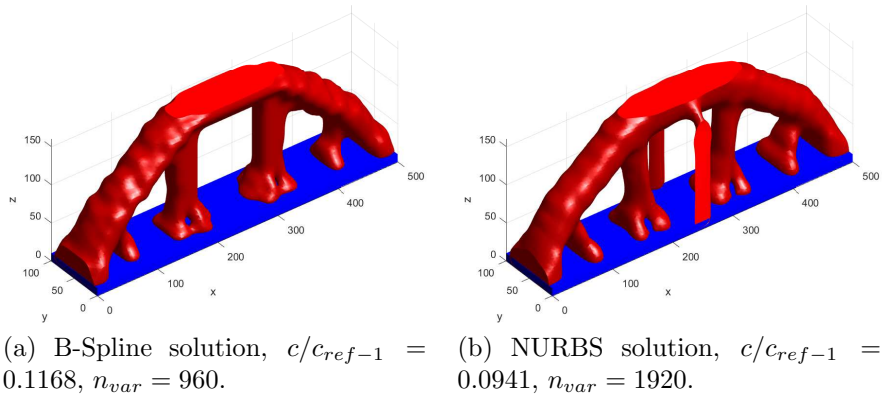


Figure 4.23 – Benchmark 1,  $p_1 = p_2 = p_3 = 2$ ,  $n_{tot} = 40 \times 8 \times 12$

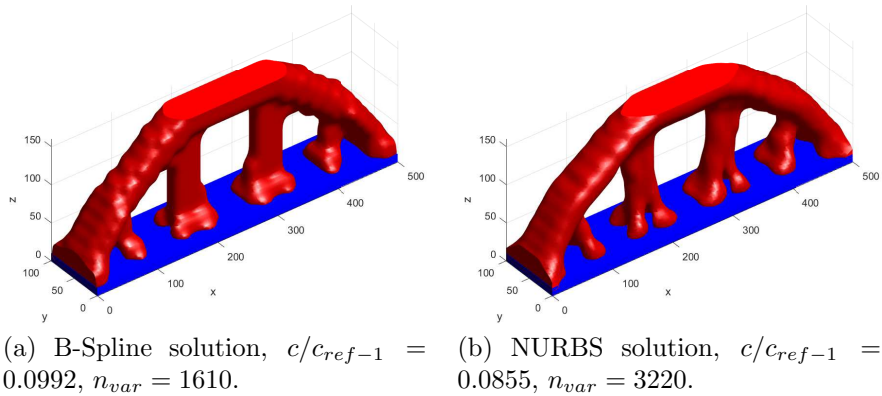


Figure 4.24 – Benchmark 1,  $p_1 = p_2 = p_3 = 2$ ,  $n_{tot} = 46 \times 10 \times 14$

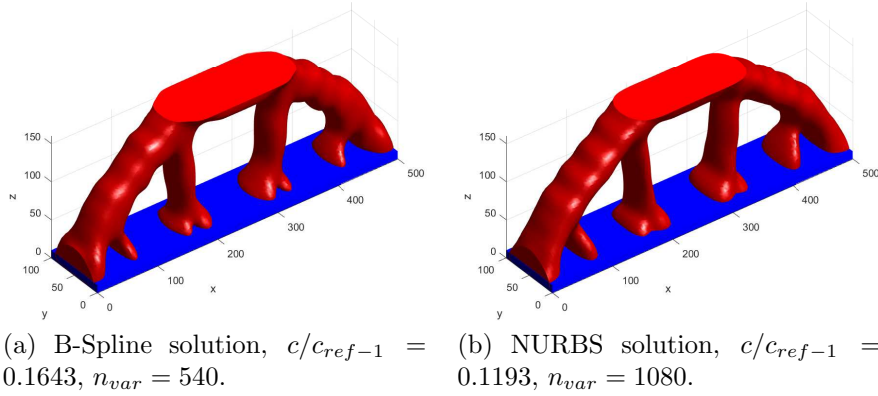


Figure 4.25 – Benchmark 1,  $p_1 = p_2 = p_3 = 3$ ,  $n_{tot} = 36 \times 6 \times 10$

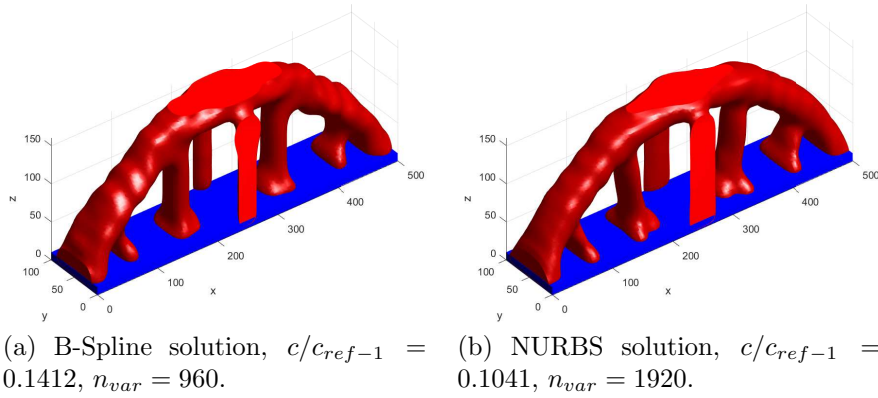


Figure 4.26 – Benchmark 1,  $p_1 = p_2 = p_3 = 3$ ,  $n_{tot} = 40 \times 8 \times 12$

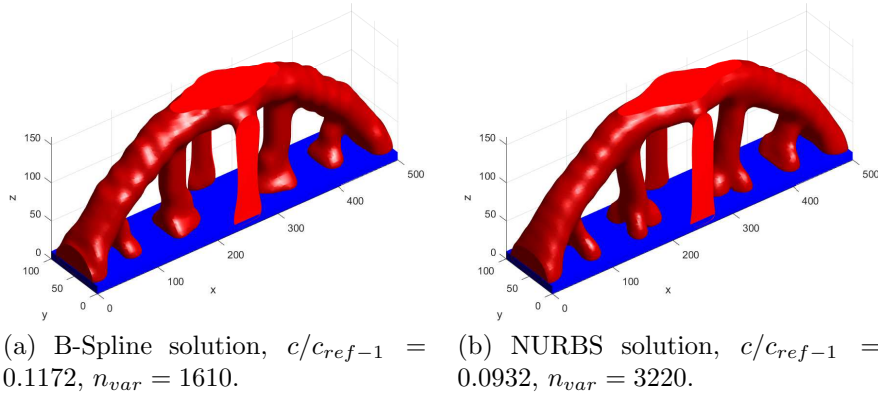


Figure 4.27 – Benchmark 1,  $p_1 = p_2 = p_3 = 3$ ,  $n_{tot} = 46 \times 10 \times 14$

Some comments can be retained by analysing the results.

- Increasing the number of control points implies reducing the objective function but

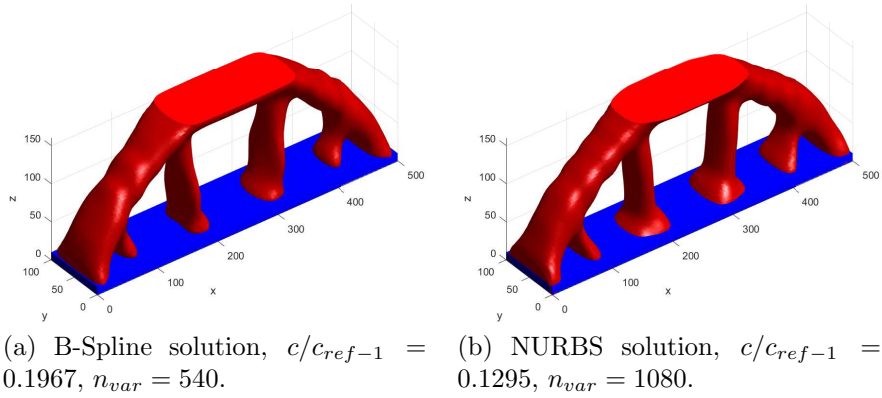


Figure 4.28 – Benchmark 1,  $p_1 = p_2 = p_3 = 4$ ,  $n_{tot} = 36 \times 6 \times 10$

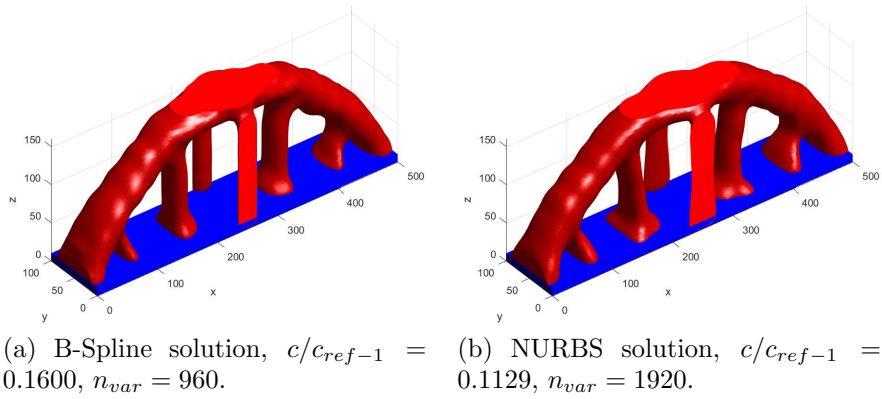


Figure 4.29 – Benchmark 1,  $p_1 = p_2 = p_3 = 4$ ,  $n_{tot} = 40 \times 8 \times 12$

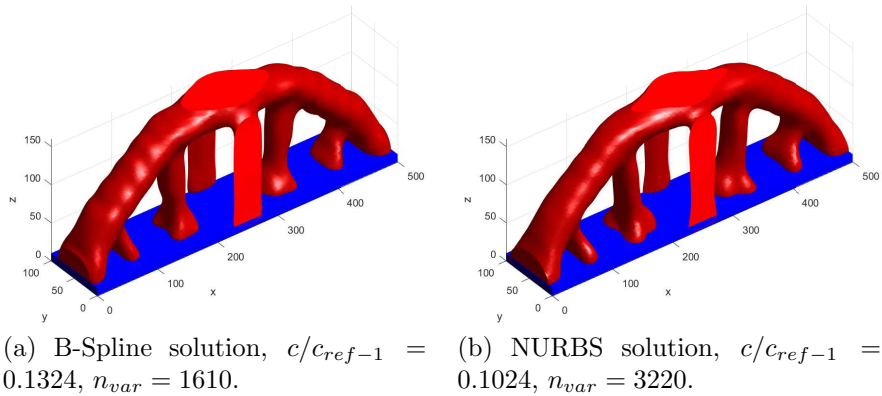


Figure 4.30 – Benchmark 1,  $p_1 = p_2 = p_3 = 4$ ,  $n_{tot} = 46 \times 10 \times 14$

the TO analysis becomes more cumbersome: this is analogous to refining the mesh in standard SIMP algorithms and it confirms the results obtained for 2D problems.



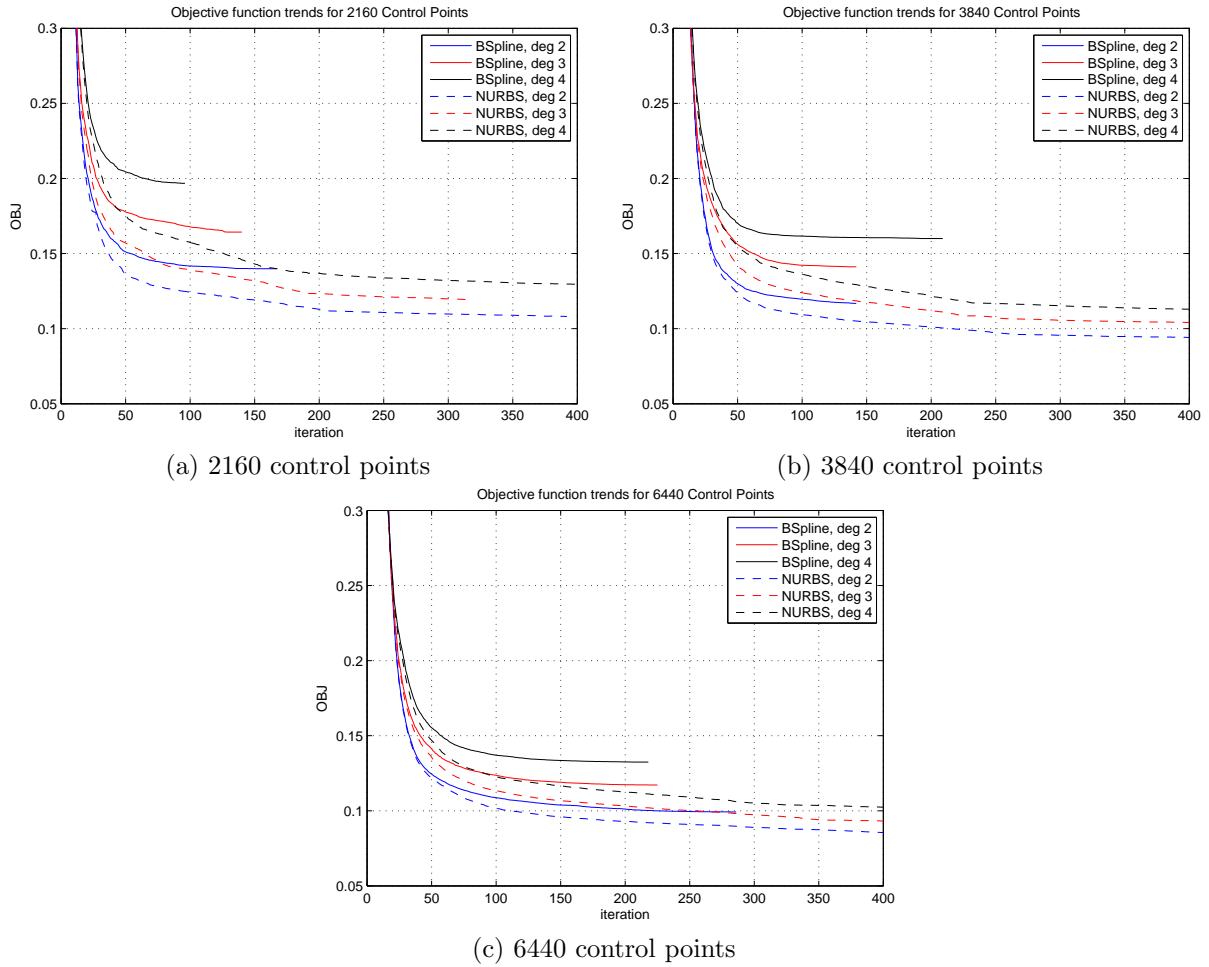


Figure 4.31 – Synthetic results: Objective function vs Iterations

- Like for 2D problems (refer to section 4.4.2), the higher the degree the greater the local support is, thus a single control point affects a wider region of the computation domain, as suggested by the definition of the local support of Eq. (4.12). Consequently, it is more complicated to balance the design variables values during the optimisation process for the algorithm in order to obtain an efficient solution (in terms of objective function). This is confirmed by the fact that the most slender topological branches disappear from the final optimum design when the degree increases. The higher the degree, the worse the solution will be in terms of objective function. However, since the designer aims at minimising the postprocessing operations, it should be highlighted that solutions with higher degrees exhibit a smoother boundary (e.g. refer to Fig. 4.22a, Fig. 4.25a and Fig. 4.28a).
- The effects of introducing the NURBS weights among the design variables in 3D

test cases confirm the results achieved for 2D problems (refer to section 4.4.2): on the one hand, weights contribute to improve the final design performances (the objective function of a NURBS solution is always lower than the corresponding one of a B-Spline solution); on the other hand, a smoother boundary can be obtained thanks to the introduction of weights among the design variables.

- If the ratio between the final objective function of a NURBS solution and a B-Spline solution is considered ( $\Lambda = c_{NURBS}/c_{B-SPLINE}$ ), the graph of Fig. 4.32 can be plotted. If the number of control points is constant, as the degree takes higher

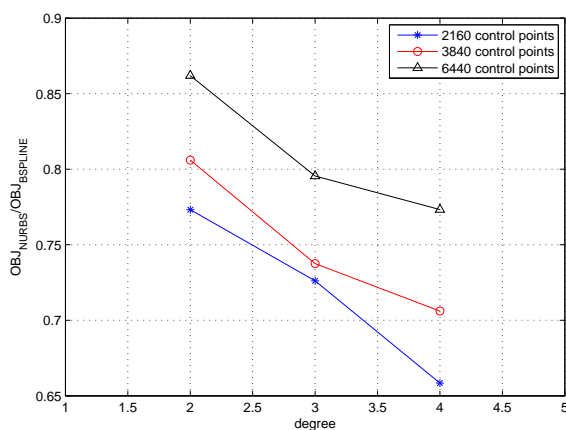


Figure 4.32 – Objective function ratio vs hyper-surfaces degrees

values,  $\Lambda$  will tend to decrease, i.e. the NURBS solution will be significantly more efficient than its B-Spline counterpart. Conversely, if the degree is fixed and the number of control points increases,  $\Lambda$  gets closer to 1: NURBS and B-Spline seem to provide almost equivalent designs in the presence of a large amount of design variables. This fact can be interpreted as a sort of saturation of the improvements given by the NURBS weights. It should be remarked that, about this aspect, there is a substantial difference between 2D and 3D TO problems. In section 4.4.2), it has been shown that NURBS solutions and B-Spline solutions provide similar results in terms of performances, thus a B-Spline surface suffices to describe the optimum 2D topology. In the most general 3D case, the use of NURBS hyper-surfaces instead of B-Spline ones is recommended: when the degree increases, the value of  $\Lambda$  can significantly decrease (as illustrated in Fig. 4.32). Since higher degrees are desired in order to obtain smoother solutions, the only way to avoid an excessive penalisation of the solution field efficiency is to consider NURBS hyper-surfaces for the description of the density field.

#### 4.4.8 Results for 3D problems: comparison between classical and NURBS-based SIMP approaches

Here, a comparison is presented between the optimal topologies found by means of the NURBS-based SIMP strategy and the results provided by the commercial software Altair OptiStruct<sup>®</sup>. The aim of this comparison is providing a more precise idea about the reliability of the proposed NURBS-based TO algorithm.

##### 3D Benchmark 1

The benchmark problem of Eq. (4.24) has been solved under identical conditions (in terms of boundary conditions, loads, NDRs). The FE model realised in OptiStruct<sup>®</sup> is composed of PSOLID elements (solid elements with 8 nodes and 3 DOFs per node). As far as the mesh size is concerned, two sizes have been tested: the first one is 10 mm, i.e. the same mesh (8000 finite elements) that is used to provide the NURBS-based solutions; the second one is 5 mm (64000 elements, i.e. a finer mesh). The corresponding number of design variables is 1875 and 15000, respectively, after applying the double symmetry condition as illustrated in Fig. 4.20.

The qualitative results provided by OptiStruct<sup>®</sup> are given in Fig. 4.33. At a first glance,

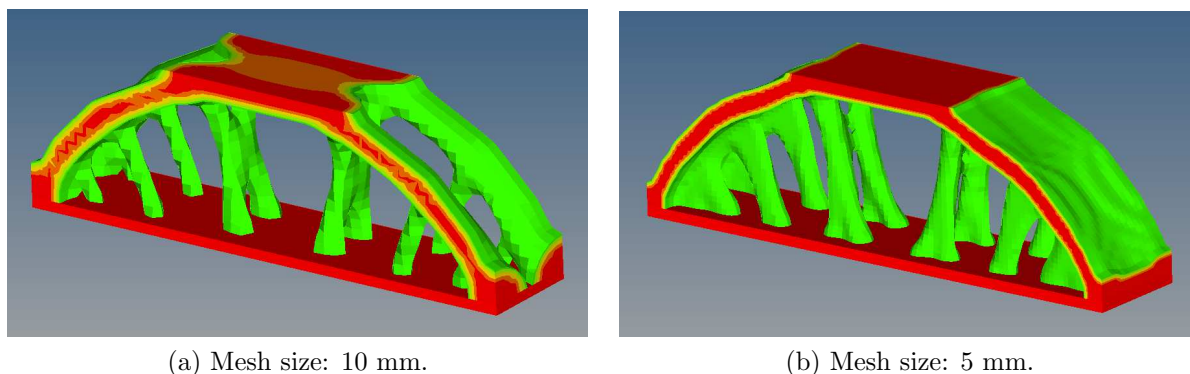


Figure 4.33 – Solutions of the first benchmark provided by the commercial software OptiStruct<sup>®</sup>

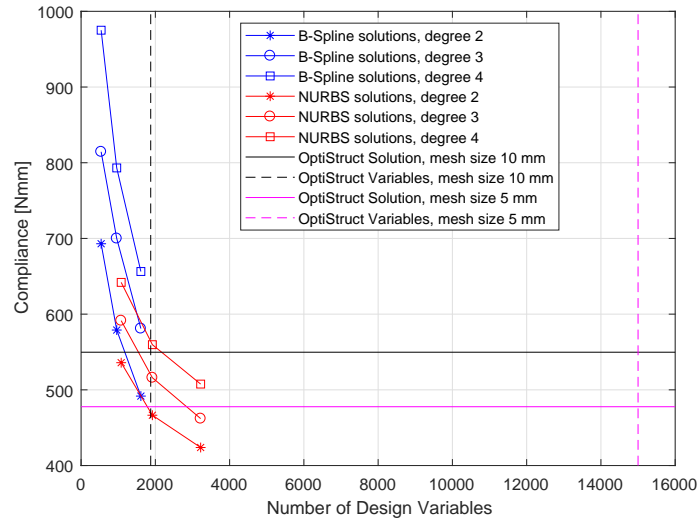
the material distribution of the solution proposed by OptiStruct<sup>®</sup> is different from that of NURBS-based configurations when the same mesh size of 10 mm is employed: instead of a single arch with thick rods, the OptiStruct<sup>®</sup> solution exhibits separated structural elements and the resulting configuration presents cavities in the middle. On the other hand, for the finer mesh, the classic SIMP method and the NURBS-based TO algorithm seem to provide consistent results (from a qualitative viewpoint).

A physical quantity must be used as a numerical measure of the efficiency of both models to establish a quantitative confrontation. The natural choice is to consider the compliance

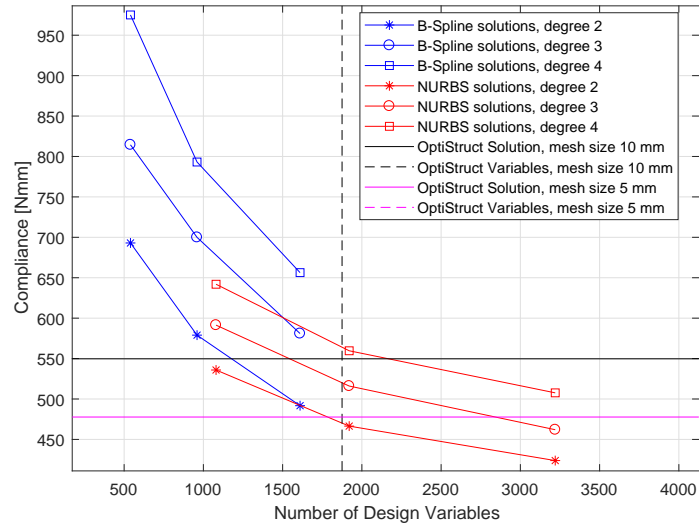
of the structure, i.e. the objective function at the end of the optimisation process. However, it should be remarked that the compliance is a derived quantity in a FE software, i.e. it is affected by the element type (and the related formulation in terms of shape functions, reduced integration, etc.) and solver options. In order to perform a correct and meaningful comparison, it has been decided to retrieve the OptiStruct<sup>®</sup> results in terms of the pseudo-density field and to transfer such a field to the FE model developed within the ANSYS environment. It is noteworthy that, in performing the previous phase, no postprocessing operation on the pseudo-density threshold value is carried out: all the model finite elements are preserved and they are simply affected by the correct value of density provided by OptiStruct<sup>®</sup> through suitable files (i.e. the “.sh” file and the “.fem” file). The density field provided by OptiStruct<sup>®</sup> at the end of the optimisation meets the equality constraint on the volume. Therefore, after transferring the OptiStruct<sup>®</sup> density field to the DR of the ANSYS FE model, a linear static analysis is performed on this configuration. This procedure allows for discussing results by taking into account only the difference between the two density fields: the first one is retrieved from the OptiStruct<sup>®</sup> solutions, while the second one is provided by the NURBS-based configurations. In this background, the resulting compliance value for the OptiStruct<sup>®</sup> solution is 549.71 Nmm for the coarser mesh and 477.697 Nmm for the finer one. The graph of Fig. 4.34 shows the compliance of all the configurations as a function of the number of design variables ( $n_{var}$ ), that is evaluated by considering the overall number of control points (and also that of the weights in case of NURBS formulation) after applying the double symmetry, as specified in section 4.4.6.

In Fig. 4.34a, the horizontal black and magenta lines represent the compliance of the configuration of Fig. 4.33a and Fig. 4.33b, respectively. The corresponding number of design variables is highlighted through two vertical dashed lines. It is evident that changing the mesh size from 10 mm to 5 mm implies a reduction of the compliance but the number of design variables and, consequently, the computational burden increases. If the solution of Fig. 4.33a is considered as a reference, the bottom-left part of the domain (with respect to the black lines) constitutes the “optimal” region, where more efficient B-Spline and NURBS-based solutions characterised by a lower number of design variables than that used by OptiStruct<sup>®</sup> can be found (Fig. 4.34b).

Analogue considerations can be inferred when considering the solution of Fig. 4.33b as a reference, corresponding to the magenta lines in Fig. 4.34a. In this case, more efficient NURBS-based solutions characterised by one seventh of design variables (with respect to the related SIMP reference solution) have been found. In general, it can be concluded that it is possible to obtain, through the NURBS-based algorithm, performances which are comparable with those provided by the SIMP method *via* the well-established TO commercial software Altair OptiStruct<sup>®</sup>.



(a) Compliance vs Number of Design Variables - General Overview.



(b) Compliance vs Number of Design Variables - Details.

Figure 4.34 – Comparison of the proposed methodology and the commercial software OptiStruct<sup>®</sup>

### 3D Benchmark 2

In order to further prove the effectiveness of the proposed approach, also problem (4.25) has been solved for the benchmark represented in Fig. 4.21 in the NURBS framework. Then, results are compared to those provided by OptiStruct<sup>®</sup> by following the aforementioned procedure (refer to Fig. 4.35). For sake of brevity, only the results related to a B-Spline hyper-surface are reported for this example. Consequently, the array  $\Xi^{(2)}$  of

weights is not considered. The same mesh of  $40 \times 40 \times 40$  solid elements has been used for both the B-Spline and the OptiStruct<sup>®</sup> configurations. As far as the B-Spline solution is concerned, the parametrisation of the design domain has been done by making use of a degree 2 hyper-surface with  $20 \times 20 \times 20 = 8000$  control points.

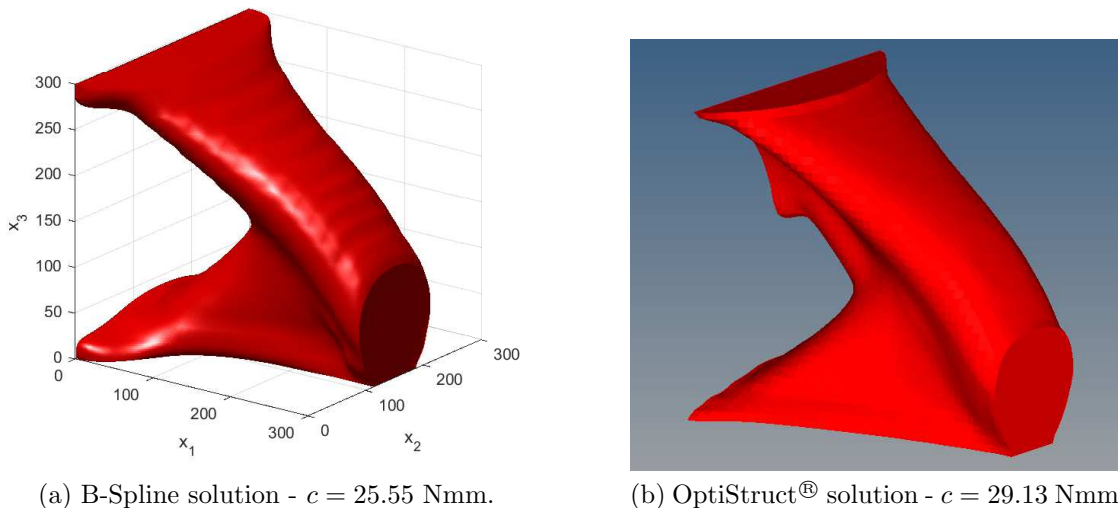


Figure 4.35 – Comparison for the second benchmark case.

Of course, the volume equality constraint is correctly met for both configurations (in the sense that  $V(\rho(\Xi^{(1)}))/V_{ref-2}$  is actually equal to  $\gamma_2$ , provided a suitable small tolerance). As it can be qualitatively remarked from Fig.4.35 and from the objective function values, the NURBS solution allows for obtaining a topology which is very close to that obtained through OptiStruct<sup>®</sup> with less design variables (4000 instead of 32000) and with a compliance value decreased by 12.3% .

#### A remark on the consistency of the volume constraint

As final observation, the consistency of solutions has been verified with respect to the volume constraint for the first benchmark when the threshold operation is carried out. To this purpose, the STL files of the optimised solutions proposed by the NURBS-based algorithm and OptiStruct<sup>®</sup> have been retrieved at the end of the optimisation process and the respective enclosed volume is computed by means of the divergence theorem:

$$\int_V \text{div}(\mathbf{r}) dV = \int_{\partial V} \mathbf{r} \cdot \mathbf{n} dS. \quad (4.26)$$

In Eq. (4.26),  $V \in \mathbb{R}^3$  is the volume of the Euclidean compact, bounded by a surface  $\partial V$  with outward normal  $\mathbf{n} = [n_{x_1}, n_{x_2}, n_{x_3}]^t$ , while  $\mathbf{r}$  is a whatever  $C^1$  vector field. If

$\mathbf{r} = \frac{1}{3} [x_1, x_2, x_3]^t$ , it can be inferred that  $div(\mathbf{r}) = 1$ ; therefore, the left-hand side of Eq. (4.26) is directly the volume enclosed by the surface  $\partial V$ . Consequently, the volume of the component represented through the STL file can be computed as

$$V = \int_{\partial V} \frac{1}{3} (x_1 n_{x_1} + x_2 n_{x_2} + x_3 n_{x_3}) dS. \quad (4.27)$$

Eq. (4.27) can be directly exploited by means of the information provided by the STL file. Since the external surface of the optimised configuration is described in terms of triangles vertices coordinates and respective normals, it is straightforward to deduce the discrete form of Eq. (4.27), i.e.

$$V = \sum_{i=1}^{N_t} \frac{A_i}{3} (x_{1G-i} n_{x_1-i} + x_{2G-i} n_{x_2-i} + x_{3G-i} n_{x_3-i}), \quad (4.28)$$

where  $N_t$  is the overall number of triangles,  $[x_{1G-i}, x_{2G-i}, x_{3G-i}]^t$  identifies the centroid of the  $i$ -th triangle, whilst  $[n_{x_1-i}, n_{x_2-i}, n_{x_3-i}]^t$  the respective normal vector. Let consider now the STL files provided by the OSSmooth software (the Altair OptiStruct<sup>®</sup> post-processor) related to the first benchmark. For both configurations represented in Fig. 4.33, the threshold value for the density has been set to that proposed by the software. The result of the application of Eq. (4.28) gives  $1.494836 \times 10^6 \text{ mm}^3$  and  $1.600126 \times 10^6 \text{ mm}^3$  for the volume of the solution shown in Fig. 4.33a and in Fig. 4.33b, respectively. If this value is compared to the imposed volume (or nominal volume), i.e.  $\gamma V_{ref} = 1.6 \times 10^6 \text{ mm}^3$ , it can be inferred that the percentage difference between the volume enclosed in the STL file provided by OptiStruct<sup>®</sup> and the imposed volume is  $-6.57\%$  in the case of the coarser mesh, whilst it is almost exact when a very finer mesh is used. When the analogue computation is repeated for the configurations resulting from the NURBS-based approach, wherein the coarser mesh is set, this percentage difference is systematically lower than  $0.5\%$  (absolute value). Therefore, it can be stated that the proposed NURBS-based SIMP method is capable of ensuring consistency between the imposed volume and the actual volume in the postprocessing phase. The discrepancy highlighted in OptiStruct<sup>®</sup> environment is related to the choice of the pseudo-density threshold value, that is automatically set by the software. A “try & error” phase is necessary in order to consistently provide the correct threshold density value in OptiStruct<sup>®</sup>. If the the finer mesh is used, this phase could be less tedious, but in general it is a time consuming step, especially in 3D.

## 4.5 Conclusions and Perspectives on the NURBS-based TO Algorithm

This Chapter has introduced and discussed a new formulation of the popular SIMP TO method in the NURBS mathematical framework: all the contents are related to references

[106–109]. In the proposed TO algorithm, the fictitious density field is represented through a NURBS surface/hyper-surface for 2D or 3D applications, respectively. The work related to this Chapter significantly enhances the contributions provided by authors in [104, 105] because of the following reasons:

- *Variables Saving and Implicitly Defined Filter Zone.* Thanks to the *local support propriety* of NURBS blending functions, a single control point (and the respective weight) affects the fictitious density field only in a well-defined portion of the computational domain. Unlike to the classical SIMP approach, there is no need to define a further filter zone, as the NURBS local support establishes an implicit relationship among contiguous mesh elements. Regardless the problem dimension (2D or 3D), the size of this filter zone depends on the NURBS discrete parameters (i.e. the degrees and the number of control points), which can be properly tuned in order to obtain a good compromise among performances, variables saving and smoothness of bounds. The higher the number of control points, the smaller the filter size is and improved performances can be achieved. Of course, computational burden is more significant in this case. Moreover, when the overall number of control points is kept constant (i.e. the number of design variables is fixed), increasing the degree means enlarging the filter zone size, which implies smoother (but less efficient) topologies. As a side effect of the local support property, the implicitly defined filter zone imposes a length scale on the design domain. This implicitly defined length scale makes mesh refinement useless if a more detailed topology description is sought. Only a refinement of the number of control points (for a given combination of blending functions degrees) could lead to an enhanced (i.e. a more detailed) topology in the case of the NURBS-based SIMP algorithm.
- *Importance of the NURBS Weights.* In the previous scenario, the influence of the NURBS weights on the final optimum topology is investigated. Including the NURBS weights among the design variables implies, on the one hand, improved quality of the solution (in terms of objective function) and, on the other hand, a smoother boundary. However, when a NURBS entity is used instead of a B-Spline one, the number of design variables increases and a longer computational time is required.
- *Performances and Robustness.* The presented algorithms systematically provides solutions, which exhibit equivalent or better performances, if compared to those obtained through a well-established commercial software as Altair OptiStruct<sup>®</sup>. The consistence of the NURBS-based SIMP method has been tested in the case of NDRs or symmetry constraints as well.
- *Results provision and consistency.* The advantages of NURBS surfaces are fully exploited in terms of their CAD compatibility: a suitable postprocessing phase can



be implemented and utilised in order to straightforwardly obtain the final optimised geometry 2D problems. Such a geometry can be stored in a standard IGS file. As far as 3D problems are concerned, results are provided in terms of a high quality STL file (no missing or degenerated triangles have been remarked in several tests). Even if this is not an outstanding improvement of the state of the art, it has been shown that the proposed approach can provide consistent results with respect to the volume constraint after the threshold operation, without any need of a “try & error” phase for both 2D and 3D problems (often necessary for optimum topology provided by commercial software).

- *Actual reassembled 2D topologies.* As previously stated, the TO result is fully CAD-compatible in the 2D case. This fact allows for evaluating the performances of the structure on a true geometry, i.e. the material phase retrieved from the wider compact domain, wherein all mesh element are present with the respective density values. When looking at the resulting topologies and at the trends of the true compliance, it can be inferred that increasing the number of design variable beyond a certain threshold value does not imply an improvement of the objective function. Thus the number of control points should be chosen as compromise between geometric accuracy and computational burden; as a further fine point of the 2D NURBS-based algorithm, the true compliance, evaluated on the rebuilt structure after the postprocessing phase, is always lower than the projected compliance (i.e. the one of the reference domain with all mesh elements). In this sense, the proposed method is conservative.

As usual for numerical methods, even the NURBS-based SIMP approach is not free from shortcomings. Actually these drawbacks constitute just as many perspectives or possible improvements for future research, as discussed in the following.

- a) The importance of the NURBS-based SIMP method goes beyond aspects considered in this Chapter: the topology description relies on a purely geometric entity (i.e. the NURBS surface/hyper-surface) defined over the computational domain and it is unrelated to the underlying mesh. Therefore, it seems really promising using the further information provided by the geometric description to formulate consistent geometric/technological constraints. For instance, recent progresses in metal ALM make this technology extremely interesting for manufacturing the topologies provided by the optimisation process (see Chapter 1). Integrating ALM constraints in the NURBS-based SIMP approach (through a dedicated formulation), as support material limitation or other kinds of customisable cost functions, constitutes an important challenge for the following of this study. First developments that exploit the NURBS parametrisation in the TO framework for suitably imposing purely geometric constraints are provided in Chapter 5 of this manuscript.

- b) The library of possible optimisation responses (objective/constraint functions) should be extended: in order to effectively design/optimize real-world engineering structures under operative service conditions, suitable constraints should be implemented. These constraints could include purely mechanical features (e.g. plasticity and failure criteria, buckling, eigenfrequencies, etc.) or specific requirements (e.g. imposed displacements/rotations in some prescribed regions). Nevertheless, multi-physics studies are possible in the context of the NURBS-based TO algorithm, so other kind of physical quantities, like temperature or heat flow, can be taken into account. Some mechanical responses, different from the structure compliance, are considered in Chapter 6 and the difficulties related to their implementation are discussed as well.
- c) The proposed NURBS-based TO algorithm aims at moving the first step towards a TO tool allowing for a direct exploitation of results by concretely simplifying the CAD reconstruction phase for both 2D and 3D problems. There is a strong potential in the proposed NURBS-based SIMP approach to achieve a better description of 3D structures boundary than an STL file: ideally, results should be obtained through the least number of patches approximating the structure boundary and, of course, with consistent continuity conditions among them. Nowadays, tools like OSSmooth provide, if required, the boundary in terms of NURBS patches via an IGS file but the external user has not a concrete control neither on the number of entities or on the connectivity conditions among adjacent surfaces (interface discontinuities). Since the CAD-compatibility is fully exploited for 2D structures, research is ongoing on these topics in order to exploit a similar procedure also in 3D problems: this task implies overcoming non-trivial difficulties as the surface parametrisation. A more specific insight into the typical issues encountered in handling these topics is given in Chapter 7.
- d) A further topic that should be faced in future works is providing a driving criterion to the user in order to choose a suitable value for the number of control points and the NURBS blending functions degrees to perform the TO via the NURBS-based algorithm. Actually, as it will be shown in Chapter 5, this point is strongly related to the implementation in the NURBS framework of geometric/mechanical constraints (as minimum and maximum length scale), which can give such NURBS parameters a more precise geometrical/physical meaning.
- e) The development of a suitable tool to manage the solutions provided by the NURBS-based algorithm is forecast. Currently, the postprocessing phase relies on the utilisation of CAD commercial software (e.g. CATIA), so a more specific and dedicated tool is necessary in order to facilitate the control points displacement or weights arrangement by a user who could not be familiar with TO concepts. Some smoothing method can be integrated in this tool in order to smooth the B-Spline “wave effect”.

Indeed, this topic is closely related to the development of the previously discussed techniques for surface approximation.

- f) Since numerical results are encouraging, the multi-scale TO of structures is perfectly conceivable. In this background, an interesting real-world engineering application could deal with the problem of designing *lattice* structures. This class of structures has gained an increasing attention since lattices can be easily manufactured by means of ALM processes. Currently, they are utilised in several fields: scaffolds for prosthesis (biomedical field), crash-worthiness parts (automotive and aerospace fields), etc.. Of course, in this context, a suitable homogenisation technique should be coupled to the present NURBS-based TO algorithm.

# Chapitre 4

## Un algorithme d'Optimisation Topologique basé sur les entités NURBS

### 4.1 Introduction

Les NURBS sont des entités mathématiques extrêmement polyvalentes qui ne sont pas exclusivement utilisées pour la CAO (modélisation des surfaces et des courbes) [66–68]. Aujourd'hui les NURBS sont employées dans des problèmes de natures différentes, tels que les stratégies d'approximation de courbes / surfaces [95–97], méta-modélisation [98, 99], optimisation des propriétés mécaniques de composites à angle variable (VAT) [93, 94, 100] et l'analyse isogéométrique [101]. On peut donc imaginer exploiter le potentiel et les propriétés intéressantes des NURBS dans le domaine de l'OT.

La méthode Level-Set représente une première tentative pour découpler la description de la topologie (à travers des entités géométriques) et l'évaluation des performances via les EF. Dans ce contexte, certains auteurs ont tenté d'intégrer les avantages des NURBS [102]. Dans la procédure de [103], la LSM est couplée à l'analyse iso-géométrique au moyen d'un formalisme NURBS à la fois pour le paramétrage de la LSF et le calcul de la fonction objectif. Concernant les stratégies fondées sur la densité pour l'OT [5, 34, 41], une amélioration intéressante et prometteuse de la méthode SIMP, basée sur l'application des entités B-Spline, a été fournie dans [104] et [105] respectivement pour les applications 2D et 3D. Dans ces études, le champ de densité fictif a été associé à une surface/hyper-surface B-Spline.

Ce chapitre présente une méthode innovante d'OT (et l'algorithme associé), basée sur un développement ultérieur et une généralisation de la méthode proposée dans [104, 105]. Ici, le champ de pseudo-densité caractérisant la méthode SIMP est lié à une surface / hyper-surface NURBS pour les applications 2D et 3D, respectivement. Pour les problèmes 2D, chaque point du réseau de contrôle NURBS est alors caractérisé par trois coordonnées dont deux sont des coordonnées cartésiennes et la troisième est la pseudo-densité. Pour les

problèmes 3D, une hyper-surface 4D est utilisée pour représenter le champ de densité fictif : chaque point constituant le contrôle *hyper-réseau* a trois coordonnées cartésiennes et la quatrième coordonnée est la pseudo-densité. L'approche proposée s'appelle la méthode SIMP basée sur les NURBS. L'impact des autres paramètres, tels que les poids des NURBS, est étudié et des comparaisons appropriées sont effectuées entre les solutions des problèmes d'OT obtenues par B-Spline et par NURBS.

En outre, l'approche basée sur les NURBS peut facilement fournir des géométries optimisées entièrement compatibles avec la CAO et les contraintes d'optimisation sont remplies sur la géométrie réassemblée en 2D. En ce qui concerne les applications 3D, actuellement, la géométrie résultante est décrite par un format natif STL « bien défini » (i.e. un fichier STL sans triangles manquants ou dégénérés) représentant les surfaces limites de la topologie optimale.

Le chapitre est présenté selon ce schéma : tout d'abord, l'énoncé mathématique de la méthode d'OT basée sur les NURBS est fourni pour les applications 2D et 3D. Le problème classique de la minimisation de la compliance avec une contrainte d'égalité sur le volume est considéré dans cette discussion. Deuxièmement, l'algorithme lié à l'approche basée sur les NURBS est présenté et ses capacités ainsi que ses limites sont discutées. Ensuite, on montre les résultats les plus significatifs sur les *benchmarks* standard. La contribution des poids apparaissant dans la formulation NURBS est étudiée et les résultats sont comparés à ceux obtenus avec les entités B-Spline. De plus, une analyse de sensibilité aux paramètres discrets (les degrés de la NURBS, le nombre de points de contrôle) est effectuée. Les résultats fournis par l'algorithme proposé sont validés au moyen du logiciel Altair OptiStruct®. Enfin, des conclusions et perspectives sur les contributions de la méthode SIMP basée sur les NURBS sont données.

Le contenu de ce chapitre fait référence aux articles [106–109].

## 4.2 Formulation Mathématique de la Méthode d'Optimisation Topologique basée sur les NURBS

La notation utilisée dans ce chapitre est cohérente avec celle introduite dans les chapitres 1 et 2. Notamment, le domaine de référence, dans lequel l'OT a lieu, est toujours défini comme un rectangle compact de taille  $a_1 \times a_2$  en 2D et un parallélépipède compact  $a_1 \times a_2 \times a_3$  en 3D. Dans le procédé SIMP basé sur les NURBS, le champ de pseudo-densité est représenté par une entité NURBS appropriée. Par conséquent, une surface NURBS est utilisée pour les problèmes 2D et une hyper-surface NURBS en 3D (voir respectivement éqs. (4.1) et (4.2)).

Bien sûr,  $\rho(u_1, u_2)$  de l'éq. (4.1) ne représente que la troisième coordonnée du vecteur  $\mathbf{S}(u_1, u_2)$  de l'éq. (2.12) : les trois coordonnées du *espace NURBS* sont les deux coordonnées spatiales et la pseudo-densité. De même,  $\rho(u_1, u_2, u_3)$  de l'éq. (4.2) constitue la quatrième

coordonnée du vecteur  $\mathbf{H}(u_1, u_2, u_3)$  dans le cas spécial de l'éq. (2.19) spécifiée pour  $N = 3$  et  $M = 4$ . Par conséquent, les paramètres sans dimension  $u_1$ ,  $u_2$ , et  $u_3$  sont directement liés aux coordonnées physiques, comme indiqué dans l'éq. (4.3).

Les points de contrôle  $\hat{\rho}_{i_1, i_2}$  sont organisés dans une matrice à deux dimensions (deux index), tandis que  $\hat{\rho}_{i_1, i_2, i_3}$  sont normalement définis dans une matrice en trois dimensions. On remarque ici qu'une entité géométrique 3D est nécessaire pour décrire correctement la topologie d'un domaine 2D (fig. 4.1a). En généralisant ce concept, le champ de densité fictif peut être associé à une hyper-surface 4D NURBS pour décrire la topologie d'une structure dans l'espace euclidien 3D (fig. 4.1b).

Parmi les nombreux paramètres affectant la forme des entités NURBS, les points de contrôle NURBS et les poids associés sont identifiés comme *variables de conception*. Ils sont disposés dans les vecteurs  $\xi_1^{2D} \in \mathbb{R}^{[(n_1+1)(n_2+1)] \times 1}$  et  $\xi_2^{2D} \in \mathbb{R}^{[(n_1+1)(n_2+1)] \times 1}$  pour des problèmes 2D et dans les vecteurs  $\xi_1^{3D} \in \mathbb{R}^{[(n_1+1)(n_2+1)(n_3+1)] \times 1}$  et  $\xi_2^{3D} \in \mathbb{R}^{[(n_1+1)(n_2+1)(n_3+1)] \times 1}$  pour les applications 3D (voir les éqs. (4.4) et (4.7)).

Les autres paramètres NURBS peuvent être identifiés comme *paramètres de conception*, c'est-à-dire que leur valeur est définie *a priori* au début de l'analyse d'OT et n'est pas optimisée.

- *Les degrés* : L'augmentation du degré implique l'élargissement de la taille du support local et les effets de cette opération doivent être étudiés.
- *Le nombre de points de contrôle* : augmenter le nombre de points de contrôle implique d'améliorer la description de la topologie et, par conséquent, d'obtenir de meilleures performances en termes de fonction objectif. Bien entendu, ce fait implique un plus grand nombre de variables de conception et des temps de calcul accrus sont attendus.
- *Le knot vector* : les composants des *knot vectors* non triviaux apparaissant dans les éqs. (2.14)-(2.15) et dans l'éq. (2.24) ont été uniformément distribués sur l'intervalle  $[0, 1]$  pour les problèmes 2D et 3D.
- *Coordonnées spatiales des points de contrôle* : elles sont inutiles en termes d'optimisation car seulement la troisième coordonnée de  $\mathbf{S}$  ou la quatrième coordonnée de  $\mathbf{H}$  sont liées à la description de la topologie. Cependant, elles peuvent devenir significatives pour les opérations de post-traitement et pour les autres applications décrites dans le chapitre 5 de ce manuscrit. L'idée est de distribuer des points de contrôle dans l'espace euclidien de telle manière que l'évaluation NURBS selon la coordonnée  $x_j$  coïncide directement avec  $x_j$ ,  $j = 1, 2, 3$ . Lorsque des entités B-Spline sont employées, cette exigence peut être formulée exactement d'un point de vue mathématique et se traduit par une formule simple et utile donnée par l'éq. (4.8), fournissant les abscisses de *Greville*.

Dans un souci de synthèse, les deux vecteurs  $\Xi^{(1)}$  et  $\Xi^{(2)}$  peuvent être définis comme à l'éq. (4.9). Le nombre total de points de contrôle est  $n_{tot} = (n_1 + 1)(n_2 + 1)$  en 2D et  $n_{tot} = (n_1 + 1)(n_2 + 1)(n_3 + 1)$  en 3D.

Dans ce contexte, le problème classique d'OT de l'éq. (1.6) devient celui de l'éq. (4.10), où  $\rho_e$  est la pseudo-densité élémentaire définie par l'éq. (4.11). La fonction objectif est divisée par une compliance de référence ( $c_{ref}$ ), pour obtenir une valeur adimensionnelle. De même, le volume est divisé par un volume de référence ( $V_{ref}$ ) dans la fonction contrainte. Les nouvelles variables de conception du problème (4.10) sont les points de contrôle et les poids de l'entité NURBS, non plus les densités des EF comme dans le problème standard (1.6).

Le calcul des dérivées est réalisé en exploitant la propriété de support locale des NURBS (voir le chapitre 2). Le support local lié à un point de contrôle  $\hat{\rho}_{I_1, I_2}$  en 2D ou  $\hat{\rho}_{I_1, I_2, I_3}$  en 3D peut être défini comme à l'éq. (4.12), où le triplet des index majuscules ( $I_1, I_2, I_3$ ) identifie un point de contrôle (ou un poids) spécifique avec index linéaire  $\tau$  via les formules de l'éq. (4.13).

Soit  $Q$  une fonction scalaire générique à considérer dans un problème d'OT dont le gradient par rapport à la pseudo-densité de l'élément générique, i.e.  $\frac{\partial Q}{\partial \rho_e}$ , est connue. Dans le cadre de l'approche SIMP basée sur les NURBS, il est nécessaire de déterminer l'expression explicite de  $\frac{\partial Q}{\partial \Xi_\tau^{(1)}}$  et  $\frac{\partial Q}{\partial \Xi_\tau^{(2)}}$ . Dans les éqs. (4.14)-(4.15), seulement les éléments figurant dans le support local du point de contrôle  $\Xi_\tau^{(1)}$  apportent une contribution non nulle aux dérivées. On peut montrer (voir Annexe A) que les dérivés de l'entité NURBS par rapport à un point de contrôle attribué et au poids associé prennent la forme des éqs. (4.16) et (4.17). La quantité scalaire  $R_\tau^e$ , apparaissant dans les éqs. (4.16) et (4.17) est simplement la fonction de base appropriée, liée au point de contrôle  $\Xi_\tau^{(1)}$ . Les dérivées pour la compliance sont données par les éqs. (4.19) et (4.20) tandis que celles pour le volume par les éqs. (4.21) et (4.22).

### 4.3 L'algorithme *SANTO* (SIMP And NURBS for Topology Optimisation)

La recherche des solutions du problème (4.10) est effectuée au moyen d'un code développé au laboratoire I2M. L'algorithme s'appelle *SANTO* (SIMP And NURBS for Topology Optimisation). L'architecture de l'algorithme est très générale et convient à la fois aux modèles 2D et 3D. Un organigramme résumant l'architecture de *SANTO* est montré à la fig. 4.2. Le logiciel ANSYS est utilisé pour calculer les réponses mécaniques de la structure dans le contexte des exemples numériques. Une description synthétique des étapes logiques est donnée ci-dessous.

1. *Problem Setting*. L'utilisateur doit initialiser tous les paramètres du problème en question : la dimension du problème (2D ou 3D), les régions de conception (DR)

et les régions de non-conception (NDR), le paramétrage du domaine (via les entités NURBS ou B-Spline). La fonction objectif et les contraintes doivent également être établies. Enfin, différents types de symétrie (miroir, rotation, translation, etc.) peuvent être définis par l'utilisateur externe, comme illustré à la fig. 4.3.

2. *NURBS/B-Spline parametrisation.* Quelle que soit la structure 2D ou 3D, il est possible de l'insérer dans un domaine rectangulaire ou prismatique appropriée, comme la fig. 4.4 le montre. Les valeurs des *knot vectors* sont uniformément distribuées dans l'intervalle  $[0, 1]$ .
3. *FE model Information.* Une fois que le maillage du modèle EF est finalisé, les éléments appartenant aux DRs sont sélectionnés et leurs données sont transmises à l'environnement MATLAB.
4. *NDRs Identification.* Toutes les zones de la structure ne doivent pas être optimisées et cette exigence peut être satisfaite en prévoyant des NDRs appropriés dans le modèle EF à des emplacements spécifiques. Les valeurs des points de contrôle associés aux NDRs sont définies  $\hat{\rho}_{max}$  ou  $\hat{\rho}_{min}$  respectivement pour la phase solide et la phase vide. Ils ne sont plus modifiés tout au long du processus d'optimisation.
5. *Symmetries Application.* Seuls les points de contrôle et les poids indépendants sont effectivement collectés dans le vecteur des variables : en fait, les symétries géométriques correspondent à des conditions particulières sur les matrices de coordonnées des points de contrôle.
6. *Reference Quantities.* Les réponses géométriques et physiques utilisées pour obtenir des fonctions objectif et de contrainte adimensionnelles sont définies ici. Leur valeur n'est pas unique et l'algorithme permet à l'utilisateur externe de définir les quantités de référence en fonction du problème rencontré.
7. *Gradient Optimisation.* Par défaut, l'optimisation est effectuée via l'algorithme AS de la famille *fmincon* de MATLAB. La base théorique a été brièvement discuté dans le chapitre 3. Une fois que l'algorithme AS est exécuté, les fonctions objectif et contrainte déclarées dans le bloc *Problem Setting* sont appelées à partir de deux bibliothèques dédiées. La première bibliothèque contient des fonctions liées à des grandeurs mécaniques spécifiques (compliance, fréquences propres, flambage, etc.) : leur évaluation nécessite d'appeler le logiciel EF externe. La seconde bibliothèque contient uniquement des quantités géométriques, telles que volume / masse, taille minimale et maximale ou rayon de courbure (voir le chapitre 5).
8. *Postprocessing.* Comme pour toutes les méthodes d'OT basées sur la densité, une opération de seuil est nécessaire à la fin de l'optimisation pour récupérer la frontière optimisée. Lorsque l'OT est réalisée dans le contexte NURBS, cette tâche est triviale



pour les applications 2D, comme illustré dans la fig. 4.5. À terme, la cohérence des résultats peut être facilement vérifiée en transférant la géométrie 2D optimisée au solveur EF via un nouveau fichier IGS. Une procédure similaire doit être suivie dans le cas de structures 3D. Cependant, alors que pour les structures 2D l'intersection se déroule dans un environnement CAO et que les résultats sont directement disponibles en termes d'entités géométriques (lignes), une telle intersection peut être facilement réalisée seulement dans un environnement dédié (MATLAB dans cette étude) pour le 3D et le résultat est une *triangulation Delaunay*, enregistrée dans un fichier STL (voir la fig. 4.6). Les aspects mathématiques qui empêchent d'obtenir un format de fichiers plus CAO-compatible par rapport au STL et les difficultés associées sont discutées au chapitre 7.

9. *Results Visualisation.* La géométrie optimisée peut être facilement stockée dans un fichier IGS pour le 2D et dans un format STL standard pour le 3D. Dans ce dernier cas, la triangulation ne présente aucun triangle dégénéré ou manquant.

## 4.4 Discussion des Résultats

L'efficacité de la stratégie proposée est prouvée grâce à des benchmarks 2D et 3D. Pour chaque cas considéré, la topologie optimale est affichée après l'opération de découpe à travers le plan ou l'hyper-plan de seuil décrits dans la section 4.3.

Les limites inférieure et supérieure pour les deux ensembles de variables impliquées dans la méthode SIMP basée sur les NURBS sont choisies comme suit :

- $\Xi_k^{(1)} \in [\hat{\rho}_{min}, \hat{\rho}_{max}]$ ,  $\hat{\rho}_{min} = 10^{-3}$ ,  $\hat{\rho}_{max} = 1$ ,
- $\Xi_k^{(2)} \in [w_{min}, w_{max}]$ ,  $w_{min} = 1/2$ ,  $w_{max} = 10$ ,

$\forall k = 1, \dots, n_{tot}$ .

Différents aspects de la méthode sont abordés :

- l'influence des paramètres discrets NURBS (degrés, nombre de points de contrôle) sur la topologie optimale finale est étudiée ;
- une comparaison entre les résultats fournis par une entité NURBS et une entité B-Spline est effectuée à la fois sur des benchmarks 2D et 3D ;
- le problème d'OT associé aux benchmarks considérés est résolu à l'aide du logiciel commercial Altair OptiStruct<sup>®</sup> et les résultats sont comparés à ceux obtenus avec l'approche SIMP basée sur les NURBS ;
- certains cas particuliers, incluant des NDRs et des symétries, sont également étudiés.

### 4.4.1 Le benchmark 2D

Le problème de la minimisation de la compliance avec une fraction de volume imposée est considéré ici pour une plaque encastrée en aluminium. Toutes les données géométriques et matérielles sont fournies à la fig. 4.7. L'énoncé mathématique du problème associé est donné par l'éq. (4.23), où  $V_{ref} = a_1 a_2 t = 1,28 \times 10^5 \text{ mm}^3$  est le volume total de la plaque, tandis que  $c_{ref} = 3520.43 \text{ Nmm}$  est la compliance de la structure lorsqu'un champ de pseudo-densité uniforme  $\rho = 0.4$  affecte la matrice de rigidité des éléments. Le modèle EF du domaine rectangulaire est discrétisé au moyen de  $80 \times 50$  éléments SHELL181 d'Ansys.

### 4.4.2 Résultats pour les problèmes 2D : sensibilité aux paramètres de la surface NURBS

Le problème (4.23) a été résolu grâce à l'algorithme décrit dans la section 4.3 pour  $p_1, p_2 = 2, 3, 4$  et  $(n_1 + 1) \times (n_2 + 1) = 16 \times 10, 32 \times 20, 48 \times 30$ . Les résultats sont fournis en termes de compliance et de fraction volumique  $V/V_{ref}$  aux figs. 4.8-4.13. La compatibilité CAO des surfaces NURBS est pleinement exploitée ici : la fonction objectif  $c$  et la contrainte de volume sont évaluées sur la « structure vraie » (topologie optimale) plutôt que sur le domaine de référence maillé. Les résultats numériques concernant la compliance réelle en fonction du nombre de points de contrôle sont représentés de manière synthétique sur la fig. 4.14. Il est intéressant de comparer cette compliance avec la compliance calculée sur le domaine de référence à la fin de l'optimisation avant l'opération de découpe (« compliance projetée », voir fig. 4.15). Les remarques suivantes découlent de l'analyse des résultats numériques :

- a) Les frontières des topologies obtenues via une surface NURBS sont plus lisses que celles obtenues avec une B-Spline, comme le montrent les figs. 4.8-4.13. Lorsque l'on regarde la compliance réelle (fig. 4.14), les performances obtenues par des NURBS sont significativement meilleures que celles obtenues par des B-Splines lorsque le nombre de points de contrôle reste faible. Si le nombre de points de contrôle augmente, une solution B-Spline pourrait être meilleure qu'une solution NURBS (voir les cas de fig. 4.12b and fig. 4.13b). Ainsi, l'utilisation de NURBS au lieu des surfaces B-Spline dans l'OT doit être soigneusement évaluée.
- b) Les paramètres  $(n_1 + 1) \times (n_2 + 1)$  et  $(p_1, p_2)$  affectent la dimension du support local des fonctions de base. Le support local, dans le contexte de l'OT, se comporte comme une *zone de filtrage*, c'est-à-dire une région du domaine de référence dans laquelle les densités des "éléments voisins" sont interdépendantes. Ses dimensions augmentent si le nombre de points de contrôle diminue ou le degré augmente et vice-versa.
- c) En observant les figs. 4.14-4.15, l'augmentation du nombre de variables de conception au-delà d'une certaine valeur seuil n'implique pas de meilleures performances, même

si la topologie peut sembler différente (voir les sous-figures b et c des figs. 4.8-4.13).

- d) Il est à noter que la compliance projetée est toujours supérieure à la compliance réelle respective : l'approche SIMP basée sur les NURBS est donc conservatrice, contrairement à la méthode SIMP classique.

### 4.4.3 Résultats pour les problèmes 2D : comparaison entre la méthode SIMP classique et l'algorithme basé sur les NURBS

Les résultats précédents ont été validés par une comparaison avec ceux obtenus via le logiciel commercial Altair OptiStruct<sup>®</sup> [11]. Bien entendu, les mêmes conditions ont été définies, à savoir le domaine de référence a été maillé avec à  $80 \times 50$  PSHELL Elements [11]. La topologie optimale reconstruite est obtenue au moyen du module OSSmooth du package Altair Hyperworks. Dans ce cas, une taille minimale  $d_{min} = 12$  mm a été choisie pour les éléments topologiques [11]. Considérons la solution B-Spline de la fig. 4.10c et la solution NURBS de la fig. 4.11c avec  $p_1 = p_2 = 3$  et  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$ . En ce qui concerne la compliance réelle, pour la solution OptiStruct<sup>®</sup>, on obtient  $c = 398.66$  Nmm (4000 variables de conception), tandis que pour la solution B-Spline  $c = 402.39$  Nmm (1440 variables de conception) et pour la solution NURBS  $c = 394.45$  Nmm (2880 variables de conception). On peut donc affirmer que l'algorithme basé sur les NURBS donne des résultats cohérents à OptiStruct<sup>®</sup> avec une réduction considérable des variables de conception.

### 4.4.4 Résultats pour les problèmes 2D : influence de Non-Design Regions

Les effets de deux NDRs sont étudiés pour montrer la polyvalence de la méthode proposée. Le problème de la fig. 4.7 a été légèrement modifié, comme le montre la fig. 4.17, où  $R_{int} = 40$  mm et  $R_{ext} = 50$  mm). Le même énoncé mathématique du problème (4.23) est toujours valable. Dans ce cas, une surface B-Spline est utilisée pour obtenir la solution et ses paramètres sont  $p_1 = p_2 = 3$  et  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$ . Les solutions optimales fournies par l'approche SIMP basée sur les NURBS et la méthode SIMP classique sont illustrées dans la fig. 4.18. La différence en pourcentage de la fonction objectif est inférieure à 5%, tandis que la fraction volumique globale fournie par l'approche SIMP basée sur les NURBS est inférieure à celle résultant de l'approche SIMP classique : dans ce dernier cas, la contrainte sur la fraction volumique n'est pas strictement respectée. Le champ de densité fictif représenté par une surface B-Spline est caractérisé « seulement » par 1440 variables de conception, tandis que le modèle OptiStruct<sup>®</sup> est caractérisé par 2392 variables de conception.

#### 4.4.5 Résultats pour les problèmes 2D : influence d'une contrainte de symétrie

Le problème montré à la fig. 4.7 et décrit par l'éq. (4.23) est enrichi par une autre contrainte géométrique : la topologie est forcée à être symétrique par rapport à au plan  $x_2 = a_2/2$ . Les paramètres principaux des entités B-Spline et NURBS sont définis comme suit :  $p_1 = p_2 = 3$  et  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$ . Les valeurs numériques de la compliance et de la fraction volumique sont rassemblées à la fig. 4.19a et à la fig. 4.19b. Les résultats d'OptiStruct<sup>®</sup> sont représentés à la fig. 4.19c. La contrainte de symétrie permet de diviser par deux le nombre de variables, donc la solution B-Spline est caractérisée par une compliance  $c = 454.80$  Nmm avec 720 variables de conception et la solution NURBS par une compliance  $c = 456.47$  Nmm (1440 variables), tandis que, pour la solution OptiStruct<sup>®</sup>, il est obtenu  $c = 443.87$  Nmm avec 2000 variables de conception.

#### 4.4.6 Les benchmarks 3D

Pour les deux problèmes présentés ci-dessous, le volume de référence  $V_{ref}$  est le volume de la DR et la compliance de référence  $c_{ref}$  est la compliance de la structure avec une densité fictive régulièrement répartie et égale à la fraction volumique imposée  $\gamma$ .

La première structure à optimiser est un prisme plein, dont les tailles sont  $a_1 = 500$  mm,  $a_2 = 100$  mm et  $a_3 = 160$  mm. Le matériau de base est Ti6Al4V ( $E = 110000$  MPa et  $\nu = 0.34$ ). Le solide est orienté dans le repère local, comme le montre la fig. 4.20. Une charge uniformément distribuée  $p_s = 0,4$  MPa est appliquée à  $x_3 = 0$  mm. Les analyses EF ont été effectuées à travers  $50 \times 10 \times 16$  éléments SOLID185. Le problème d'OT correspondant peut être énoncé comme indiqué dans l'éq. (4.24), où  $c_{ref-1} = 4957.13$  Nmm et  $V_{ref-1} = 7.5 \times 10^6$  mm<sup>3</sup>.

Le second problème est l'OT d'un cube ( $a_1 = a_2 = a_3 = 300$  mm) encastré sur une face et subissant une charge surfacique vers le bas de  $0,4$  MPa répartie sur la zone coloré en bleu sur la fig. 4.21. Pour ce problème, une seule contrainte de symétrie est imposée sur l'hyper-surface NURBS/B-Spline par rapport au plan  $x_2 = a_2/2$ . Le matériau est de l'aluminium ( $E = 72000$  MPa et  $\nu = 0,33$ ). Les éléments attachés à la zone de chargement sont considérés comme une NDR. L'analyse EF est effectuée avec  $40 \times 40 \times 40$  éléments SOLID185. Le problème d'optimisation est donné par l'éq. (4.25) avec  $c_{ref-2} = 672.33$  Nmm et  $V_{ref-2} = 2.7 \times 10^7$  mm<sup>3</sup>.

#### 4.4.7 Résultats pour les problèmes 3D : sensibilité aux paramètres de l'hyper-surface NURBS

Le problème (4.24) est résolu avec  $p_1, p_2, p_3 = 2, 3, 4$  et  $(n_1 + 1) \times (n_2 + 1) \times (n_3 + 1) = 36 \times 6 \times 10, 42 \times 8 \times 12, 46 \times 10 \times 14$ . Compte tenu de la double symétrie, le nombre de variables de conception ( $n_{var}$ ) est opportunément diminué. Les résultats sont présentés

dans les figs. 4.22-4.30. Par souci de concision, les résultats sont synthétisés dans les graphiques de la fig. 4.31, où les valeurs de  $c(\rho)/c_{ref}$  sont représentées en fonction des itérations.

Les mêmes conclusions que celles en 2D peuvent être déduites en 3D en analysant les résultats qualitatifs et numériques. Si l'on considère le rapport entre la fonction objectif finale d'une solution NURBS et une solution B-Spline ( $\Lambda = c_{NURBS}/c_{B-SPLINE}$ ), le graphique de la fig. 4.32 peut être tracé. Si le degré est fixe et que le nombre de points de contrôle augmente,  $\Lambda$  se rapproche de 1 : les NURBS et B-Spline semblent fournir des configurations presque équivalentes en présence d'un grand nombre de variables de conception. Il convient de noter qu'à propos de cet aspect, il existe une différence substantielle entre les problèmes 2D et 3D en OT. Au contraire du 2D, en 3D, l'utilisation d'hyper-surfaces NURBS au lieu de B-Spline est recommandée car, lorsque le degré augmente, la valeur de  $\Lambda$  peut diminuer de manière significative.

#### 4.4.8 Résultats pour les problèmes 3D : comparaison entre la méthode SIMP classique et l'algorithme basé sur les NURBS

##### 3D Benchmark 1

Le problème (4.24) a été résolu dans des conditions identiques (en termes de conditions limites, de charges, de NDRs). Le modèle EF réalisé dans OptiStruct<sup>®</sup> est composé d'éléments PSOLID. En ce qui concerne la taille du maillage, deux tailles ont été testées : la première est de 10 mm (8000 EF) ; la seconde est de 5 mm (64000 EF). Le nombre des variables de conception est respectivement 1875 et 15000 après application de la condition de double symétrie illustrée à la fig. 4.20. Les résultats qualitatifs fournis par OptiStruct<sup>®</sup> sont donnés dans la fig. 4.33.

Afin d'effectuer une comparaison correcte et significative, il a été décidé de récupérer le champ de pseudo-densité d'OptiStruct<sup>®</sup> et de le transférer au modèle EF développé dans l'environnement ANSYS. Tous les EF du modèle sont conservés et ils sont simplement affectés par la valeur correcte de la densité fournie par OptiStruct<sup>®</sup> via les fichiers « .sh » et « .fem ». Ensuite, une analyse statique linéaire est effectuée sur cette configuration. Dans ce contexte, la valeur de compliance obtenue pour la solution OptiStruct<sup>®</sup> est de 549,71 Nmm pour le maillage le plus grossier et de 477,697 Nmm pour le maillage le plus fin. Le graphique de la fig. 4.34 montre la compliance de toutes les configurations en fonction du nombre de variables de conception ( $n_{var}$ ).

Dans la fig. 4.34a, les lignes horizontales noires et magenta représentent respectivement la compliance de la configuration de la fig. 4.33a et de la fig. 4.33b. Le nombre de variables de conception est mis en évidence par deux lignes verticales en pointillés. Il est évident que changer le maillage de 10 à 5 mm implique une réduction de la compliance mais aussi du nombre de variables de conception et, par conséquent, la charge de calcul augmente. En considérant la solution de la fig. 4.33b comme référence, des solutions plus efficaces

basées sur les NURBS ont été trouvées : elles sont caractérisées par sept fois moins de variables de conception par rapport à la solution de référence donnée par la méthode SIMP. En général, on peut conclure qu'il est possible d'obtenir, grâce à l'algorithme basé sur les NURBS, des performances comparables à celles fournies par la méthode SIMP via un logiciel commercial bien testé comme Altair OptiStruct®.

### 3D Benchmark 2

Le problème (4.25) a également été résolu pour le benchmark représenté sur la fig. 4.21 dans le contexte des NURBS. Ensuite, les résultats sont comparés à ceux fournis par OptiStruct® (voir fig. 4.35). Par souci de synthèse, seuls les résultats liés à une hypersurface B-Spline sont rapportés pour cet exemple. Le même maillage de  $40 \times 40 \times 40$  éléments solides a été utilisé pour les configurations B-Spline et OptiStruct®. En ce qui concerne la solution B-Spline, le paramétrage du domaine de conception a été effectué en utilisant une hyper-surface de degré 2 avec  $20 \times 20 \times 20 = 8000$  points de contrôle. Comme on peut le remarquer qualitativement à la fig. 4.35 sur les valeurs de la fonction objectif, la solution NURBS permet d'obtenir une topologie très proche de celle obtenue par OptiStruct® avec moins de variables de conception (4000 au lieu de 32000) et avec une valeur de compliance diminuée du 12,3%.

#### Une remarque sur la cohérence de la contrainte sur le volume

La cohérence des solutions a été vérifiée par rapport à la contrainte de volume pour le premier benchmark lorsque l'opération de seuil est effectuée. A cette fin, les fichiers STL des solutions optimisées (à la fois ceux proposés par l'algorithme basé sur les NURBS et par OptiStruct®) ont été récupérés à la fin du processus d'optimisation et le volume inclus est calculé à l'aide de l'éq.(4.27) et sa forme discrétisée donnée par l'éq. (4.28). Dans l'éq. (4.28),  $N_i$  est le nombre total de triangles,  $[x_{1G-i}, x_{2G-i}, x_{3G-i}]^t$  identifie le barycentre du triangle  $i$ , tandis que  $[n_{x_1-i}, n_{x_2-i}, n_{x_3-i}]^t$  le vecteur normal associé.

Le résultat de l'application de l'éq. (4.28) donne  $1,494836 \times 10^6 \text{ mm}^3$  et  $1.600126 \times 10^6 \text{ mm}^3$  pour le volume de la solution montrée à la fig. 4.33a et à la fig. 4.33b, respectivement. En comparant cette valeur au volume imposé (ou volume nominal), i.e.  $\gamma V_{ref} = 1,6 \times 10^6 \text{ mm}^3$ , la différence en pourcentage entre le volume contenu dans le fichier STL fourni par OptiStruct® et le volume imposé est  $-6,57\%$  dans le cas du maillage le plus grossier, alors qu'il est presque exact si le maillage plus fin est utilisé. Lorsque le même calcul est répété pour les configurations résultant de l'approche basée sur les NURBS avec un maillage plus grossier, cette différence en pourcentage est systématiquement inférieure à 0.5%. L'écart remarqué dans l'environnement OptiStruct® est lié au choix de la valeur de seuil de pseudo-densité, qui est automatiquement définie par le logiciel. Une phase « try & error » est nécessaire pour fournir systématiquement la valeur de densité de seuil correcte dans OptiStruct®.

## 4.5 Conclusions et Perspectives concernant l’Algorithme d’OT basé sur les NURBS

Ce chapitre a présenté et discuté une nouvelle formulation de la populaire méthode SIMP en OT dans le cadre mathématique des entités NURBS : tout le contenu est lié aux références [106–109]. Ce travail fournit essentiellement ces contributions

- *Economie de Variables et Zone de Filtrage implicite.* Grâce à la *propriété du support local* des fonctions de base des NURBS, contrairement à l’approche SIMP classique, il n’est pas nécessaire de définir une autre zone de filtrage, car le support local établit une relation implicite entre les éléments de maillage contigus. Seulement un raffinement du nombre de points de contrôle (pour une combinaison donnée de degrés de fonctions de base) pourrait conduire à une topologie plus détaillée dans le cas de l’algorithme SIMP basé sur les NURBS.
- *Importance des poids des NURBS.* L’inclusion des poids des NURBS parmi les variables de conception implique, d’une part, une amélioration de la qualité de la solution (en termes de fonction objectif) et, d’autre part, une frontière plus lisse. Cependant, lorsqu’une entité NURBS est utilisée à la place d’une entité B-Spline, le nombre de variables de conception augmente.
- *Performances and Robustesse.* Les algorithmes présentés fournissent systématiquement des solutions avec des performances équivalentes ou meilleures que celles obtenues avec Altair OptiStruct®. La cohérence de la méthode SIMP basée sur les NURBS a également été testée dans le cas des NDRs ou des contraintes de symétrie.
- *Cohérence des Résultats.* Les avantages des surfaces NURBS sont pleinement exploités en termes de compatibilité CAO : une phase de post-traitement appropriée peut être implémentée et utilisée afin d’obtenir directement les géométries 2D optimisées. En ce qui concerne les problèmes 3D, les résultats sont fournis en termes de fichier STL de haute qualité (aucun triangle manquant ou dégénéré n’a été remarqué dans plusieurs tests).
- *Topologies 2D réassemblées.* La CAO-compatibilité dans le cas 2D permet d’évaluer les performances de la structure sur une vraie géométrie, c’est-à-dire la phase matérielle extraite du domaine plus large, où tous les éléments du maillage avec les valeurs de densité respectives sont pris en compte. En examinant les topologies résultantes et les allures de la compliance réelle, on peut en déduire que l’augmentation du nombre de variables de conception au-delà d’une certaine valeur de seuil n’implique pas une amélioration de la fonction objectif.

Comme les autres méthodes numériques, l'approche SIMP basée sur les NURBS n'est pas exempte d'inconvénients, qui constituent autant de perspectives ou d'améliorations possibles pour les recherches futures, comme indiqué ci-après.

- a) La description de la topologie repose sur une entité purement géométrique (i.e. surface / hyper-surface NURBS) définie sur le domaine de calcul dans la méthode proposée. Par conséquent, il semble très prometteur d'utiliser les informations supplémentaires fournies par la description géométrique pour formuler des contraintes géométriques / technologiques cohérentes et spécifiques. Par exemple, l'intégration des contraintes de l'ALM dans l'approche SIMP basée sur les NURBS (à travers une formulation dédiée) constitue un défi important pour la suite de cette étude.
- b) La bibliothèque de réponses d'optimisation possibles (fonctions objectif / contraintes) devrait être étendue : pour concevoir / optimiser efficacement des structures d'ingénierie réelles dans des conditions de service opérationnelles, des contraintes appropriées doivent être implémentées. Des problèmes mécaniques ou multi-physiques sont possibles dans le contexte de l'algorithme d'OT basé sur les NURBS, et donc autres types de grandeurs physiques, telles que la température ou le flux de chaleur, peuvent être pris en compte.
- c) L'algorithme présenté constitue le premier pas d'un outil d'OT permettant une exploitation directe des résultats en simplifiant concrètement la phase de reconstruction CAO pour les problèmes 2D et 3D. Il existe un fort potentiel dans l'approche SIMP basée sur les NURBS pour obtenir une meilleure description des frontières des structures 3D qu'un fichier STL. Des recherches sont en cours sur ces sujets afin d'exploiter une procédure similaire dans les problèmes 3D : cette tâche implique de surmonter des difficultés non négligeables comme le paramétrage de la surface.
- d) Un autre sujet qui devrait être abordé dans les travaux futurs est de fournir un critère à l'utilisateur afin de choisir une valeur appropriée pour le nombre de points de contrôle et le degré des fonctions de base NURBS pour effectuer l'OT. Ce point est fortement lié à l'implémentation dans le cadre NURBS de contraintes géométriques / mécaniques (par exemple celle de l'épaisseur minimale et maximale d'éléments constitutifs de la topologie), pouvant donner à tels paramètres NURBS une signification géométrique / physique plus précise.
- e) Le développement d'un outil approprié pour gérer les solutions fournies par l'algorithme basé sur les NURBS est prévu. Actuellement, la phase de post-traitement repose sur l'utilisation de logiciels CAO commerciaux (par exemple, CATIA). Un outil plus spécifique et dédié est donc nécessaire pour faciliter le déplacement ou la répartition des poids des points de contrôle.



- f) Les résultats numériques étant encourageants, l'OT multi-échelle des structures est parfaitement envisageable. Dans ce contexte, une application d'ingénierie réelle intéressante pourrait traiter le problème de la conception de structures *lattice*. Bien entendu, dans ce contexte, une technique d'homogénéisation appropriée devrait être couplée à l'algorithme d'OT actuel basé sur les NURBS.

# Chapter 5

## Geometrical Constraints Implementation in the Framework of the NURBS-based Topology Optimisation Algorithm

### 5.1 Introduction to Geometrical Constraints in Topology Optimisation

In this Chapter, a general enhancement of problem (4.10) is considered

$$\begin{aligned} & \min_{\Xi^{(1)}, \Xi^{(2)}} \frac{c(\rho(\Xi^{(1)}, \Xi^{(2)}))}{c_{ref}}, \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} (\sum_{e=1}^{N_e} \rho_e^\alpha [K_e]) \{d\} = [K] \{d\} = \{f\}, \\ \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref}} = \frac{\sum_{e=1}^{N_e} \rho_e V_e}{V_{ref}} = \gamma, \\ \mathbf{g}(\Xi^{(1)}, \Xi^{(2)}) \leq \mathbf{0}, \\ \Xi_K^{(1)} \in [\hat{\rho}_{min}, \hat{\rho}_{max}], \\ \Xi_K^{(2)} \in [w_{min}, w_{max}], \\ \forall k = 1, \dots, n_{tot}. \end{array} \right. \end{aligned} \quad (5.1)$$

In Eq. (5.1), the constraints array  $\mathbf{g}$  has been introduced. Its components will be specified all along the discussion, when required.

As stated in Chapters 1 and 4, TO software have become well-established tools in the preliminary design phase of engineering products. However, one of the most basic needs for engineers is the integration of manufacturing requirements into the TO problem formulation in order to achieve optimised as well as manufacturable solutions [42]. Manufacturing requirements are in general strongly dependent on the chosen technology: nevertheless, the minimum and maximum admissible size of structural elements constitutes a fundamental aspect, regardless the considered process.

More specifically, each technology has an intrinsic minimum achievable size. Therefore, controlling the minimum length scale of topological features in the structure to be optimised is of outstanding importance, in order to avoid obtaining an extremely performing but absolutely non-manufacturable component. One of the first methods to take into account the minimum length scale in a standard density-based strategy is described in [112] and it is implemented in the commercial software Altair OptiStruct<sup>®</sup> [11]: the minimum length scale is imposed through a control on the slope of the pseudo-density function on the whole design domain. Such a method turns out to be computationally efficient and can replace the perimeter penalisation or the density filtering operation for mesh independence [5]. However, an important lack of consistence between the imposed minimum member size and the actually measured minimum member size at the end of the optimisation process can be remarked.

A different approach has been proposed by Poulsen [35] and it relies on the monotonicity control of the pseudo-density function along  $n_d$  preferential directions depending on the problem dimension:  $n_d = 4$  or  $n_d = 13$  for 2D and 3D problems, respectively. Although this method is sound, it provides solutions with jagged boundaries and, moreover, its efficiency is strongly restricted to regular mapped meshes (the extension of the method to whatever free mesh is anything but trivial). As projection methods have been included in TO approaches, also a filter-based minimum member size control has been implemented [10]. Projection methods have been further developed by making use of the concept of “eroded”, “intermediate” and “dilated” design [42,53]. This kind of techniques guarantees a strict control on the minimum member size but they are computationally burdensome because they need three FE analyses (one for each density phase). The technique discussed in [42,53] is referred as “robust formulation” but the robustness must be interpreted as a consistence of the length scale of the optimised configurations with respect to manufacturing imprecisions. Anyway, the minimum length scale must be *a posteriori* checked on the final (CAD reassembled) geometry. Other strategies have been developed in the framework of the very general and versatile LSM (see Chapter 1) applied to structural TO problems. Authors in [113] have developed a smart strategy to get an explicit and local control of the minimum length scale in the context of LSM, based on the mathematical concept of “structural skeleton”. Furthermore, they have extended this method to the SIMP framework [114]. However, these works neglect the possible change of the skeleton in the sensitivity analysis (i.e. the computation of the derivatives of the objective and

constraints functions with respect to the design variables). The drawbacks related to this aspect are discussed in [115] and a suitable solution is proposed in [45], where it is shown that a mathematically exact definition of member size does not exist. The difficulty related to the sensitivity analysis appearing in the method described in [114] has been recently overcome by authors in [116]: they combine the skeleton method with the three-phases projection method and, finally, the minimum length scale control is performed thanks to two *structural indicator functions* and two *ad hoc* constraints. An alternative method for including the minimum length scale requirement in a B-Spline density-based TO algorithm has been suggested in [117]: the proposed strategy is based on the combination of the approach presented in [104] and the three-phases projection methods [42]. However, from a theoretical viewpoint, it seems inconsistent to choose a method requiring three FE analyses in order to control a purely geometric feature.

The minimum member size has been largely investigated in TO literature, as it can be understood by the previous state of the art. Contrariwise, the maximum member size has been tackled by less authors. However, controlling the maximum length scale in TO algorithms has a significant interest, which goes beyond the technological requirements and originates from different needs. As remarked in [118, 119], imposing a maximum length scale in TO results in final optimised configurations meeting fail-safe engineering requirements, since they exhibit multiple load paths. This fact has been confirmed in [120], wherein a simplified damage model is included in TO analyses. One of the first approaches to set up a maximum length scale control in TO has been suggested by Guest in [43], where explicit constraints are imposed on the neighbourhood of each mesh element by making use of projection methods, which allow for better stability of solutions. The evident shortcoming of such a formulation is the high number of optimisation constraints and suitable aggregation techniques should be forecast. In order to provide a more efficient tool, two alternatives have been proposed in [119]. Firstly, the technique used in [121] has been generalised and the maximum member size is controlled through a low-pass filter in the frequency domain and a Fast Fourier Transform (FFT). This method requires to control the size of the topological features by suitably setting the design parameters in the frequency domain and it does not allow the designer to have a direct sensitivity of the geometry. Secondly, the concept of mathematical morphology is used to formulate a suitable constraint for the maximum member size but the provided solutions can exhibit disjoint zones of material phase, which are meaningless from a physical viewpoint. As in the case of the minimum length scale requirement, a lack of consistence is observed for the maximum member size constraint as well: the imposed maximum size (during TO) and the maximum size measured on the final topology, i.e. after CAD reassembly operations, do not match.

As discussed above, the minimum and the maximum length scale requirements are not controlled in satisfactory way. To overcome the restrictions related to the previous techniques, the implementation of the minimum and maximum member size control in

the NURBS-based SIMP approach is discussed in this Chapter. As a first attempt, the mathematically sound minimum length scale formulation proposed by Poulsen [35] has been revisited and tailored to the NURBS-based approach: its effectiveness is illustrated on a 2D example. Furthermore, in order to fully exploit the useful geometrical features of NURBS blending functions, a more general method has been developed for handling the minimum length scale requirement in a very smart and effective way. The main idea is to propose a method (based on a set of precise rules and relationships) able to relate the minimum member size to the characteristic parameters of the NURBS geometric entity, such as the control point number and the degrees of the NURBS blending function along each parametric direction, for both 2D and 3D TO problems. Moreover, the proposed method aims at simplifying the TO problem formulation: the minimum length scale requirement is not introduced as an explicit optimisation constraint. Instead, it is implicitly tuned through the NURBS parameters.

On the other hand, the maximum length scale requirement is integrated as an explicit constraint: to this aim, Guest's maximum member size constraint [43] has been opportunely reformulated in the NURBS framework for both 2D and 3D TO problems. Finally, it is noteworthy that the length scale control, albeit basic, is not the only need in the framework of a design process. In order to show the capabilities of the NURBS-framed TO algorithm, an innovative geometric constraint has been implemented thanks to the NURBS formalism. It is focused on the minimum local curvature radius on the boundary of the structure being optimised.

This Chapter is outlined as follows. Firstly, Poulsen's minimum length scale constraint is reformulated in the framework of the NURBS-based SIMP algorithm and results are discussed for 2D problems. Secondly, a purely geometric method based on the NURBS parameters is presented in order to control the minimum length scale without formulating an explicit optimisation constraint. Then, details about the implementation of Guest's maximum length scale constraint are provided. Results about the minimum and maximum length scale control are given for both 2D and 3D applications. Finally the minimum curvature radius constraint is mathematically stated and its effects are investigated on a simple 2D benchmark.

The contents of this Chapter refer to part of [107] and to [122].

## 5.2 Poulsen's Formulation of the Minimum Length Scale Constraint

### 5.2.1 Mathematical Statement of Poulsen's Minimum Length Scale constraint

The intuitive idea of the Poulsen's method for the minimum length scale constraint [35] specified for 2D structures consists of imposing the monotonicity of the fictitious density function in a circular area having a diameter equal to the minimum member size ( $d_{min}$ ). The circular area is sketched around each mesh element and the monotonicity is checked along each of the four directions ( $0^\circ, 90^\circ, \pm 45^\circ$ ). Mathematically speaking, the monotonicity of a function on an interval  $I$  along a direction  $\mathbf{b}$  can be checked by means of the following integral:

$$M_{\mathbf{b}}(f) = \int_I |\nabla f \cdot \mathbf{b}| dl - \left| \int_I \nabla f \cdot \mathbf{b} dl \right|. \quad (5.2)$$

$M_{\mathbf{b}}(f)$  is strictly equal to 0 if  $f$  is monotone and greater than 0 otherwise. Therefore, the constraint on the minimum member size is formulated as follows:

$$g_{d_{min}} = \sum_{e=1}^{N_e} \left( \sum_{\mathbf{b}_i} M_{\mathbf{b}_i}(\rho) \right)^\theta - \sigma \leq 0, \quad (5.3)$$

where  $N_e$  is the number of mesh elements,  $\mathbf{b}_i$  the checking direction ( $i = 1, \dots, 4$ ),  $\theta$  a penalising exponent, whilst  $\sigma$  is used to relax the constraint and to provide numerical stability. Of course,  $M_{\mathbf{b}_i}(\rho)$  is the monotonicity integral and its evaluation domain is the circular zone having diameter  $d_{min}$  and centred at the centroid of each element. The explicit expression of  $M_{\mathbf{b}_i}(\rho)$  is

$$M_{\mathbf{b}_i}(\rho) = \int_{-d_{min}/2}^{d_{min}/2} |\nabla \rho \cdot \mathbf{b}_i| ds - \left| \int_{-d_{min}/2}^{d_{min}/2} \nabla \rho \cdot \mathbf{b}_i ds \right|. \quad (5.4)$$

In Eq. (5.4),  $s$  is a suitable abscissa along the current checking direction  $\mathbf{b}_i$ . In particular,  $\mathbf{b}_1 = [1, 0]^t$ ,  $\mathbf{b}_2 = [0, 1]^t$ ,  $\mathbf{b}_3 = [\sqrt{2}/2, \sqrt{2}/2]^t$  and  $\mathbf{b}_4 = [\sqrt{2}/2, -\sqrt{2}/2]^t$ .

In order to formulate a discrete version of Eq. (5.4), let us consider a regular *mapped* mesh of square elements. Then,  $N_{\mathbf{b}_i}$  is the number of mesh elements spanning the diameter  $d_{min}$  along the  $\mathbf{b}_i$  direction. It is straightforward to verify (see [35]) that Eq. (5.4) changes into

$$M_{\mathbf{b}_i}(\rho) = \sum_{j=1}^{N_{\mathbf{b}_i}-1} |\rho_{j+1} - \rho_j| - \left| \rho_{N_{\mathbf{b}_i}} - \rho_1 \right|. \quad (5.5)$$

In Eq. (5.5),  $j$  is just a mute index that sweeps the mesh elements on the interval  $[0, d_{min}]$ , like the abscissa  $s$  in Eq. (5.4). Furthermore, a smooth approximation of the absolute function has been employed to avoid the derivative discontinuity in the gradient-based algorithm, namely

$$|z| \approx \sqrt{z^2 + \epsilon^2} - \epsilon, \quad (5.6)$$

with  $\epsilon = 0.01$ . The final expression of  $M_{\mathbf{b}_i}(\rho)$  to be implemented is

$$M_{\mathbf{b}_i}(\rho) = \sum_{j=1}^{N_{\mathbf{b}_i}-1} \left( \sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2} - \epsilon \right) - \sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2} + \epsilon. \quad (5.7)$$

In Eqs. (5.5) and (5.7)  $\rho_1$  and  $\rho_{N_{\mathbf{b}_i}}$  are the pseudo-densities and related to the first element and to the last element spanned along the  $\mathbf{b}_i$  direction, respectively. As far as the sensitivity analysis is concerned, the same notation introduced in Chapter 4 is used. Accordingly, the derivative of the minimum member size constraint with respect to the NURBS control points reads

$$\frac{\partial g_{d_{min}}}{\partial \Xi_{\tau}^{(1)}} = \theta \sum_{e=1}^{N_e} \left( \sum_{\mathbf{b}_i} M_{\mathbf{b}_i}(\rho) \right)^{\theta-1} \left( \sum_{\mathbf{b}_i} \frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_{\tau}^{(1)}} \right). \quad (5.8)$$

Analogously, the derivative with respect to the NURBS weights is

$$\frac{\partial g_{d_{min}}}{\partial \Xi_{\tau}^{(2)}} = \theta \sum_{e=1}^{N_e} \left( \sum_{\mathbf{b}_i} M_{\mathbf{b}_i}(\rho) \right)^{\theta-1} \left( \sum_{\mathbf{b}_i} \frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_{\tau}^{(2)}} \right). \quad (5.9)$$

The only difficult consists in evaluating the terms  $\frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_{\tau}^{(1)}}$  and  $\frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_{\tau}^{(2)}}$ . The detailed computation is carried out in Appendix B and the final result is provided here:

$$\frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_{\tau}^{(1)}} = \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{(\rho_{j+1} - \rho_j) (R_{\tau}^{j+1} - R_{\tau}^j)}{\sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2}} - \frac{(\rho_{N_{\mathbf{b}_i}} - \rho_1) (R_{\tau}^{N_{\mathbf{b}_i}} - R_{\tau}^1)}{\sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2}}, \quad (5.10)$$

$$\begin{aligned} \frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_{\tau}^{(2)}} &= \frac{\Xi_{\tau}^{(1)}}{\Xi_{\tau}^{(2)}} \frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_{\tau}^{(1)}} + \\ &+ \frac{1}{\Xi_{\tau}^{(2)}} \left[ \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{(\rho_{j+1} - \rho_j) (\rho_j R_{\tau}^j - \rho_{j+1} R_{\tau}^{j+1})}{\sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2}} - \frac{(\rho_{N_{\mathbf{b}_i}} - \rho_1) (\rho_1 R_{\tau}^1 - \rho_{N_{\mathbf{b}_i}} R_{\tau}^{N_{\mathbf{b}_i}})}{\sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2}} \right]. \end{aligned} \quad (5.11)$$

In Eqs. (5.10) and (5.11), it is assumed  $\rho_j = \rho(u_j, v_j)$ , whilst  $R_{\tau}^j$  is defined according to Eq. (4.18).

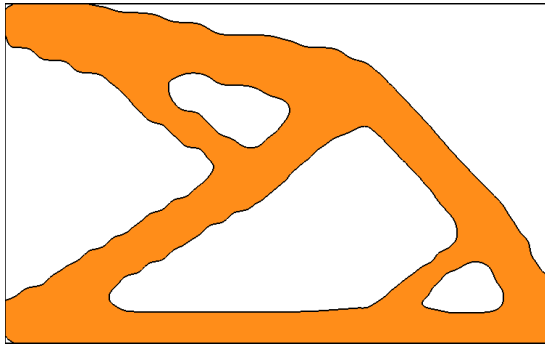
### 5.2.2 Poulsen's Minimum Length Scale constraint: numerical results

The revisited version of the Poulsen's minimum length scale constraint in the NURBS framework is tested here on the benchmark illustrated in Fig. 4.7. Thus, the constraint of Eq. (5.3) is added to problem (4.23). Particularly, a B-Spline surface with  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$  control points and  $p_1 = p_2 = 3$  is used to carry out the solution. The constraint on the minimum member size is imposed by considering three values of  $d_{min}$ , i.e. 16, 20, and 25 mm. Results are collected in Figs. 5.1-5.3 for each analysis. In each figure, the first image is the rebuilt geometry provided by the NURBS-based SIMP method after the postprocessing phase, whilst the second image is the final rebuilt geometry provided by the commercial tool OptiStruct<sup>®</sup>. The FE model of the reference domain is the same, as illustrated in section 4.4.1.

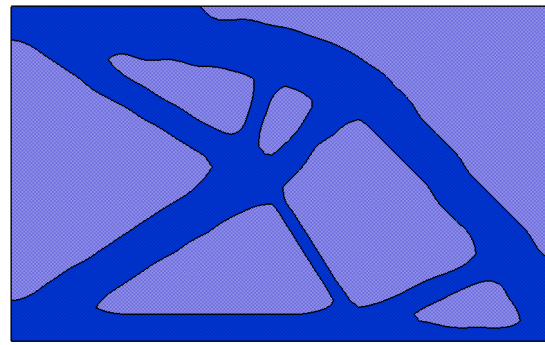
Two remarks of paramount importance can be inferred from the analysis of the results depicted in Figs. 5.1-5.3:

- Some of the advantages related to the NURBS geometrical properties are exploited in this case. Thanks to the combined action of the local support property and of the Poulsen's minimum member size constraint, all the meaningless "grey" zones are filtered and the final topology exactly meets the minimum member size constraint. Conversely, even if the optimum topologies resulting from OptiStruct<sup>®</sup> are characterised by better performances in terms of the compliance value, they systematically do not meet the minimum member size constraint due to the presence of thin topological branches. Quantitatively, the minimum size provided by OptiStruct<sup>®</sup> is 8 mm instead of 16 mm for the first case, 8 mm instead of 20 mm for the second one and 7 mm instead of 25 mm for the last one.
- The optimum solutions provided by the NURBS-based SIMP approach show non-smooth boundaries. Indeed this aspect is related to the formulation of the minimum member size constraint according to the Poulsen's formula, see Eq. (5.3). As discussed in [35], the minimum member size is evaluated, for each element, only along four directions in order to reduce the computational effort. Therefore, a smoother boundary could be achieved by increasing the number of checking direction in the Poulsen's equation.



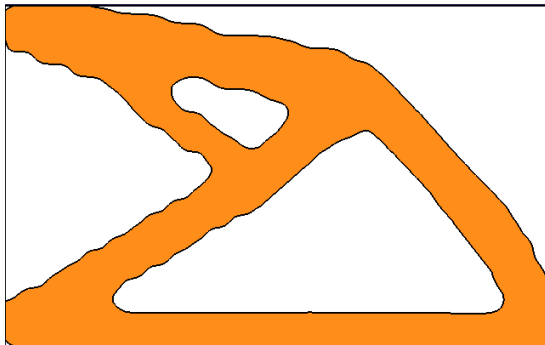


(a) B-Spline solution:  $c = 441.98$  Nmm,  $V/V_{tot} = 0.3996$ .

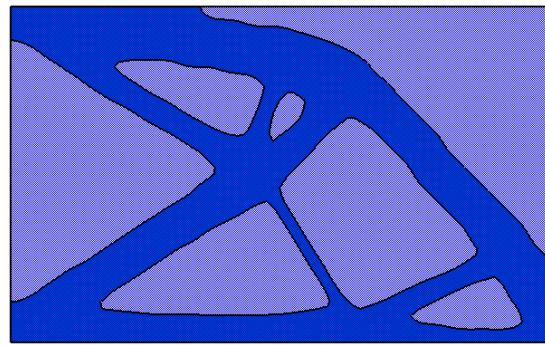


(b) OptiStruct<sup>®</sup> solution:  $c = 401.02$  Nmm,  $V/V_{tot} = 0.3994$ .

Figure 5.1 – Cantilever plate problem,  $d_{min} = 16$  mm.

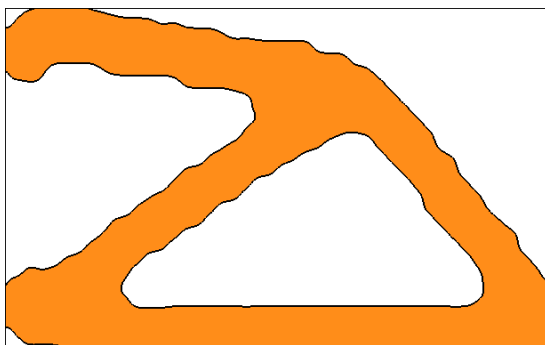


(a) B-Spline solution:  $c = 438.78$  Nmm,  $V/V_{tot} = 0.4027$ .

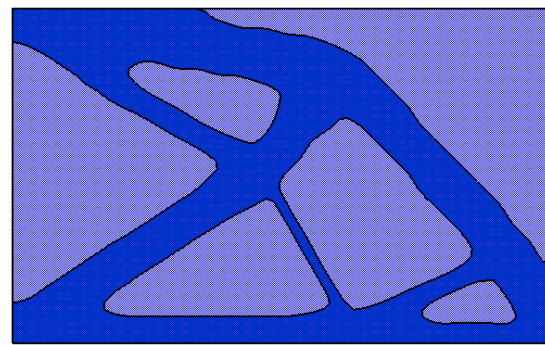


(b) OptiStruct<sup>®</sup> solution:  $c = 400.22$  Nmm,  $V/V_{tot} = 0.4006$ .

Figure 5.2 – Cantilever plate problem,  $d_{min} = 20$  mm.



(a) B-Spline solution:  $c = 488.57$  Nmm,  $V/V_{tot} = 0.4036$ .



(b) OptiStruct<sup>®</sup> solution:  $c = 400.73$  Nmm,  $V/V_{tot} = 0.4000$ .

Figure 5.3 – Cantilever plate problem,  $d_{min} = 25$  mm.

## 5.3 Minimum Length Scale control in the NURBS-based TO Algorithm

### 5.3.1 Minimum length scale resulting from B-Spline entities

The NURBS formalism discussed in Chapter 4 advises to develop a more effective criterion to impose a minimum length scale in TO. Although Poulsen’s method has shown encouraging and consistent results when it is formulated in the NURBS framework, it is subject to important shortcomings in practice. Namely, it properly works only on mapped mesh (its extension to whatever free mesh is not trivial) and it induces a jagged boundary. In order to overcome these issues, a purely geometric strategy is proposed here to satisfy the minimum length scale requirement on the final topology, for both 2D and 3D structures. Of course, the NURBS parameters listed in Chapters 2 and 4 are supposed to have a strong impact on this topological feature. For the sake of clarity, only B-Spline entities are considered in a first time. In this background, some peculiar requirements need to be introduced in order to implicitly ensure a given minimum length scale, without introducing an explicit constraint into the problem formulation.

*Requirement 1:* The spatial coordinates of control points defining the B-Spline surfaces and hyper-surfaces are distributed according the Greville’s abscissae, provided through Eq. (4.8).

*Requirement 2:* the condition of minimum length scale must be simulated. As far as 2D problems are concerned, the minimum member size condition is reproduced by assigning the value  $\hat{\rho}_{min} = 10^{-3}$  to each control point coordinate  $\hat{\rho}_{i_1, i_2}$ , apart from either a column or a row of control points, which are set to  $\hat{\rho}_{max} = 1$ , as shown in Fig. 5.4a and Fig. 5.4c. Thanks to the *strong convex-hull property* of NURBS entities (see Chapter 2), the pseudo-density function takes values in the interval  $[10^{-3}, 1]$ . The result of the intersection between the B-Spline surface and a suitable plane (representing the pseudo-density threshold value) is a strip of material phase (Fig. 5.4b and Fig. 5.4d). The thickness of this strip represents the minimum length scale, that can be obtained along  $x_2$  or  $x_1$  axes if control points coordinates are set to 1 column-wise or row-wise, respectively.

Since control points coordinates are arranged in 3D arrays for B-Spline hyper-surfaces, a similar procedure can be repeated in 3D by setting  $\hat{\rho}_{min}$  for all control point coordinates  $\hat{\rho}_{i_1, i_2, i_3}$ , except for those points belonging to a suitable “page” of the 3D array: e.g. the minimum member size that can be identified orthogonally to the  $x_1$  direction is detected by setting  $\hat{\rho}_{I_1, i_2, i_3} = 1, \forall i_2 = 0, \dots, n_2, \forall i_3 = 0, \dots, n_3$ , with an assigned  $I_1$ . Similarly,  $\hat{\rho}_{i_1, I_2, i_3} = 1, \forall i_1 = 0, \dots, n_1, \forall i_3 = 0, \dots, n_3$  and  $\hat{\rho}_{i_1, i_2, I_3} = 1, \forall i_1 = 0, \dots, n_1, \forall i_2 = 0, \dots, n_2$  define the minimum length scale orthogonally to  $x_2$  and  $x_3$  axes, respectively (see Fig. 5.5).

*Requirement 3:* the threshold value used for the pseudo-density field ( $\rho_{th}$ ) has an impact on the minimum length scale for the problem at hand, as it can be easily inferred

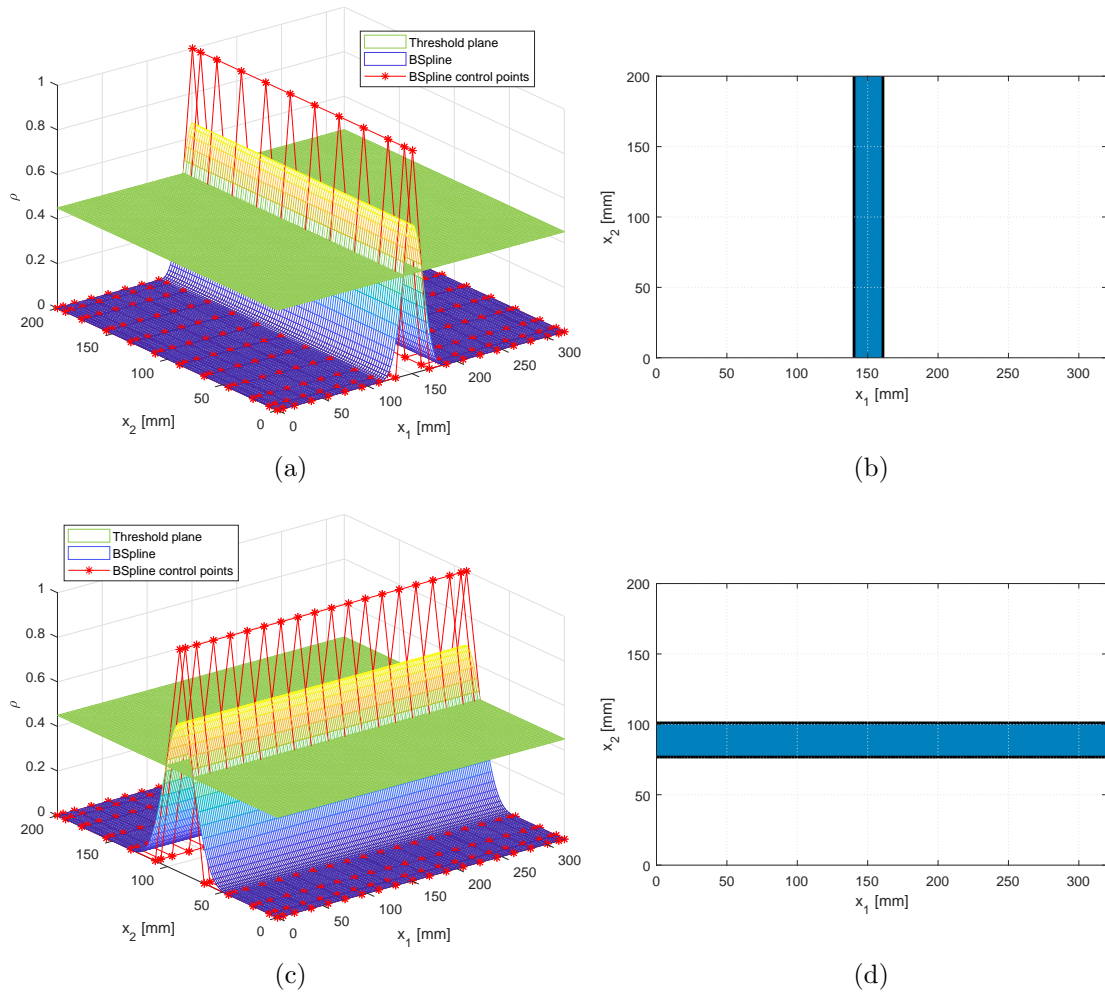


Figure 5.4 – Simulation of 2D minimum length scale

from Fig. 5.4. As observed in all the simulations performed here and in Chapter 4, the threshold density can be set  $\rho_{th} \in [\rho_{th-LB}, \rho_{th-UB}]$ , with  $\rho_{th-LB} = 0.35$  and  $\rho_{th-UB} = 0.6$ .

*Requirement 4:* the NURBS-based SIMP method has been outlined in Chapter 4 in its most general form. Particularly, the same methodology can be applied regardless the knot vectors components. Moreover, the Greville’s abscissae formula applies for whatever knot vector distribution. In the context of the minimum member size constraint, knot vectors components have been chosen uniformly spanned in the interval  $[0, 1]$ , unless otherwise stated, i.e.

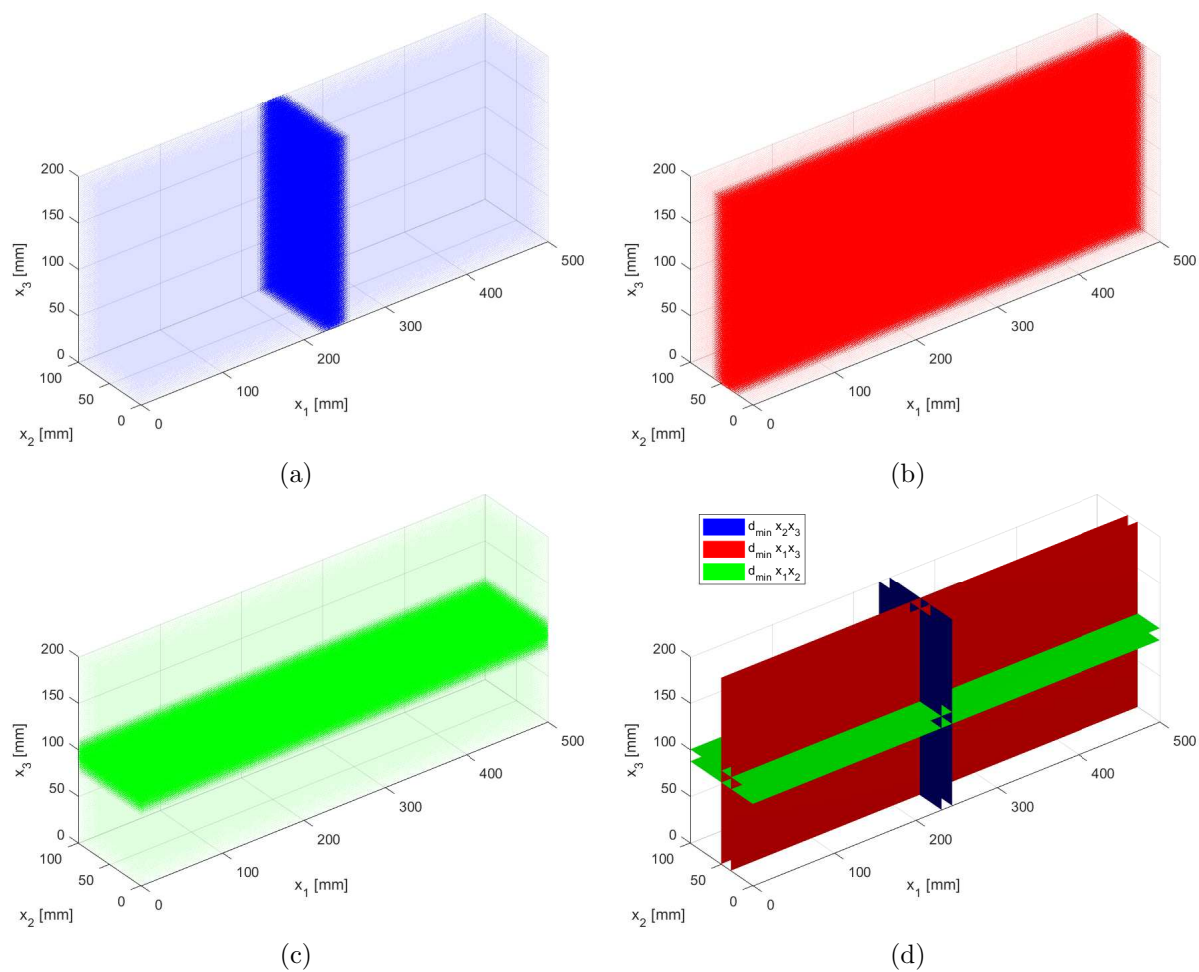


Figure 5.5 – Simulation of 3D minimum length scale.

$$\mathbf{U}^{(j)} = \left\{ \underbrace{0, \dots, 0}_{p_j+1}, \frac{1}{n_j - p_j + 1}, \dots, \frac{k_j}{n_j - p_j + 1}, \dots, \frac{n_j - p_j}{n_j - p_j + 1}, \underbrace{1, \dots, 1}_{p_j+1} \right\}, \quad (5.12)$$

$$k_j = 1, \dots, n_j - p_j, \quad j = 1, 2, 3.$$

Thus, the distance between two non-trivial components of the knot-vector is

$$\Delta U^{(j)} = \frac{1}{n_j - p_j + 1}, \quad j = 1, 2, 3. \quad (5.13)$$

Furthermore, it is easy to verify the following relationship among control points coordinates and the respective knot vectors components:

$$\Delta X^{(j)} = \frac{a_j}{n_j - p_j + 1} = a_j \Delta U^{(j)}, \quad j = 1, 2, 3. \quad (5.14)$$

Considering the previous aspects, a sensitivity analysis of the minimum length scale, referred as  $d_{min}$ , is presented in the following. In particular, the influence of the number of control points on  $d_{min}$  is investigated. For these preliminary analyses, the blending functions degrees along each parametric direction, i.e.  $p_j$ ,  $j = 1, 2, 3$ , are kept constant and the knot-vectors components are equispaced in the range  $[0, 1]$  according to Eq. (5.12). The logical steps of Algorithm 1, devoted to the derivation of the minimum length scale curves, are reported here below.

---

**Algorithm 1** Minimum length scale requirement.

---

- 1: Set  $a_j$  and  $p_j$  ( $j = 1, 2, 3$ ). Set the initial value of the number of control points along each direction, i.e.  $n_{0j}$ . Initialise the slack variable  $k = 1$  and set its upper bound  $k_{max}$ .
- 2: Update the control points number in each parametric direction according to

$$n_j = n_{0j} + \Delta n_j(k), \quad j = 1, 2, 3. \quad (5.15)$$

- 3: Evaluate the uniform knot-vector according to Eq. (5.12).
  - 4: Calculate Greville's abscissae according to Eq. (4.8).
  - 5: Determine  $\Delta X^{(j)}(k)$ ,  $j = 1, 2, 3$ , according to Eq. (5.14).
  - 6: For 2D applications, two B-Spline surfaces denoted as  $\rho_1(k)$  and  $\rho_2(k)$  are created, while, for 3D problems, three hyper-surfaces  $\rho_1(k)$ ,  $\rho_2(k)$  and  $\rho_3(k)$  are defined as described in *Requirement 2*.
  - 7: The minimum member size  $d_{min}^j(k)$  is evaluated by performing a suitable intersection between the two B-Spline surfaces and the threshold plane in 2D or between the three B-Spline hyper-surfaces and the threshold hypereplane in 3D. Different threshold density values have been considered, i.e.  $[0.35, 0.40, 0.45, 0.50, 0.55, 0.60]$ .
  - 8:  $k = k + 1$ . If  $k < k_{max}$ , go to point 2, otherwise go to point 9.
  - 9: The trend of  $d_{min}^j$  vs.  $\Delta X^{(j)}$  is plotted.
- 

The trend of  $d_{min}^j$  vs.  $\Delta X^{(j)}$  is illustrated in Figs. 5.6-5.10 for  $p_j = 3$  and for both 2D and 3D problems.

If the graphs of Fig. 5.6, related to 2D problems, are represented on the same plane  $d_{min}$  vs.  $\Delta X$ , the result is an almost perfect superposition, as shown in Fig. 5.8a. The same procedure is carried out for the graphs of Fig. 5.7 related to the 3D case, and the result is completely analogue (see Fig. 5.8b).

As it can be deduced by the complete superposition of the curves  $d_{min}^{(j)}$  vs.  $\Delta X^{(j)}$  for each density threshold value, there is no need to use a specific graph for each parametric direction / physical coordinate: a single set of curves is dedicated to the 2D case, whilst a different set to the 3D case, as illustrated in Fig. 5.8. These graphs depend also upon the degree of the blending functions: the cases  $p_j = 2$  and  $p_j = 4$  are given in Figs. 5.9 and 5.10, respectively, for both 2D and 3D problems.

### 5.3. Minimum Length Scale control in the NURBS-based TO Algorithm

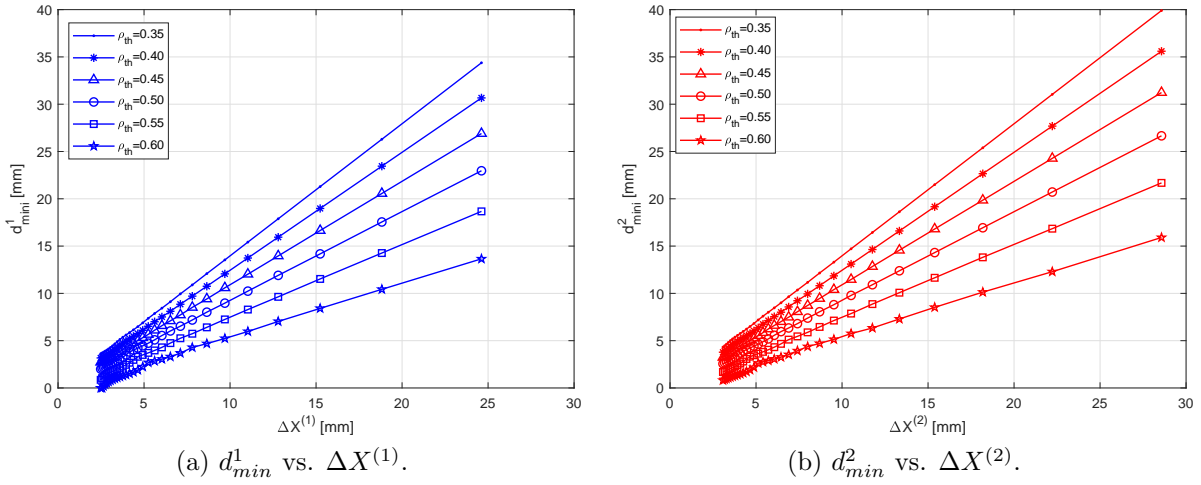


Figure 5.6 –  $d_{min}$  trends in 2D,  $p_j = 3$ .

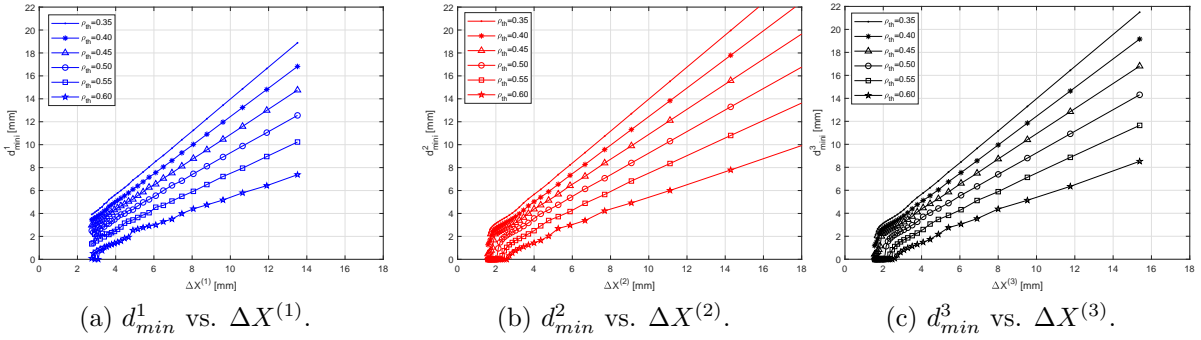


Figure 5.7 –  $d_{min}$  trends in 3D,  $p_j = 3$ .

The curves  $d_{min}$  vs.  $\Delta X$  can be used as a design tool to forecast the minimum length scale for different combinations of  $n_j$  and  $p_j$ . In particular, given a certain value of  $d_{min}$  (e.g. imposed by technological requirements), the designer can choose the B-Spline degree and, through the corresponding *abaci* of Figs. 5.8-5.10, can select the related  $\Delta X$  to be used along each physical direction  $x_j$ ,  $j = 1, 2, 3$  (this fact asks the designer to forecast a suitable value for the final threshold density). Through the knowledge of  $\Delta X$  from Eq. (5.14), the designer can easily determine a suitable number of control points along each parametric direction, i.e.  $n_j$  ( $j = 1, 2, 3$ ). Furthermore, since the knot vectors are uniform, the previous graphs can be converted in  $d_{min}$  vs.  $n/a$  curves, that is the minimum member size as a function of the control points density, as shown in Figs. 5.11-5.13. Therefore, the TO analysis is performed by setting the right number of control points: once convergence is achieved, the minimum length scale is measured on the reassembled geometry (after the cutting operation through the threshold density plane/hyperplane) and it should be

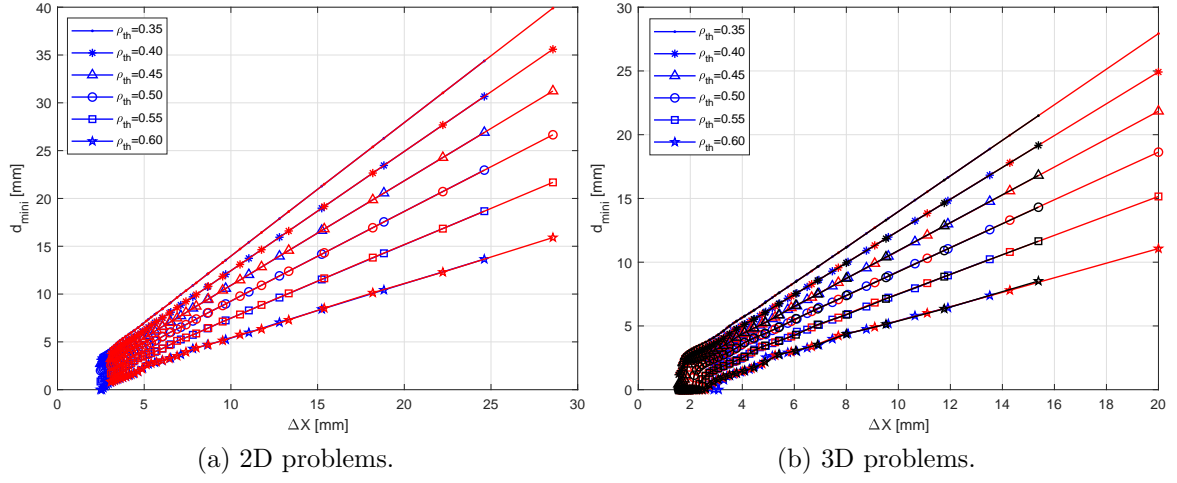


Figure 5.8 –  $d_{min}$  vs.  $\Delta X$ ,  $p_j = 3$ .

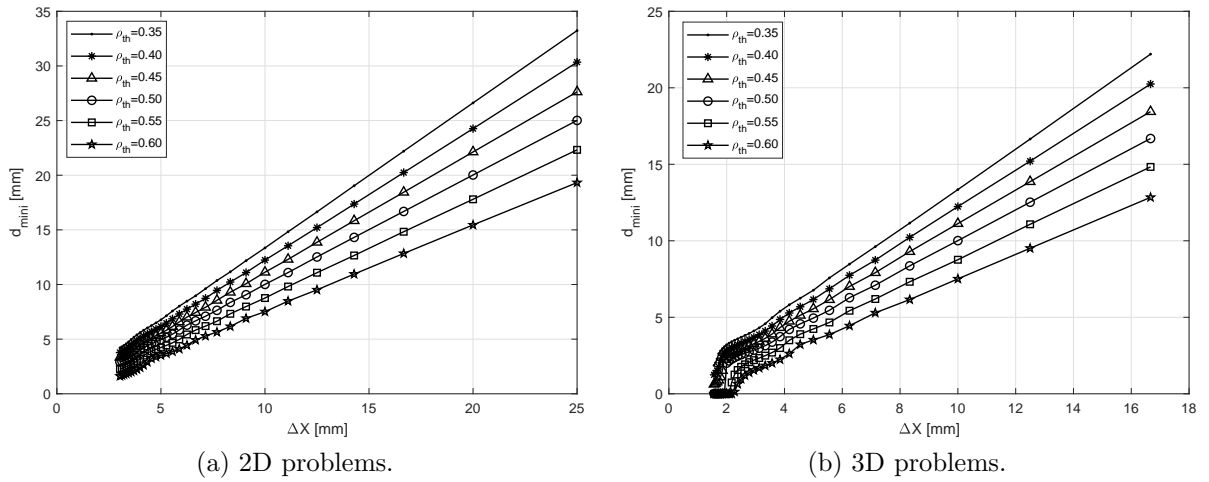


Figure 5.9 –  $d_{min}$  vs.  $\Delta X$ ,  $p_j = 2$ .

verified that the minimum member size measured on the CAD model is always greater than or equal to the minimum length scale forecast by means of Figs. 5.8-5.10.

The procedure for determining suitable *abaci* to take into account for the minimum length scale must be slightly modified for those regions which are close to the boundary of the computational domain. This fact is perfectly logic, since the previously discussed algorithm holds for a constant  $\Delta X^{(j)}$ . Indeed, as control points are distributed on the reference domain by means of the Greville's abscissae formula of Eq. (4.8),  $\Delta X^{(j)}$  is not constant and strongly varies within the regions adjacent to the boundary of the

### 5.3. Minimum Length Scale control in the NURBS-based TO Algorithm

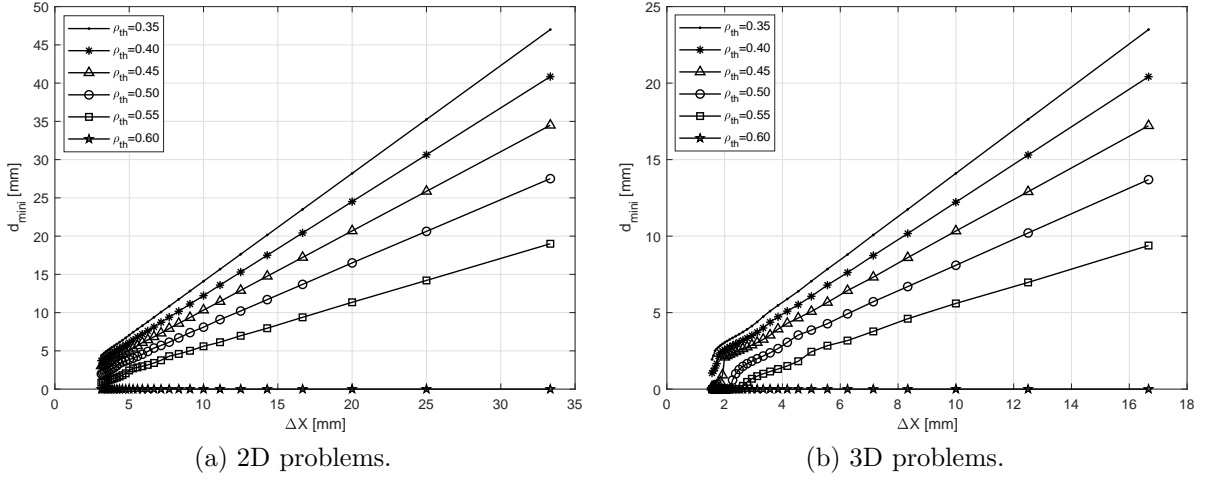


Figure 5.10 –  $d_{min}$  vs.  $\Delta X$ ,  $p_j = 4$ .

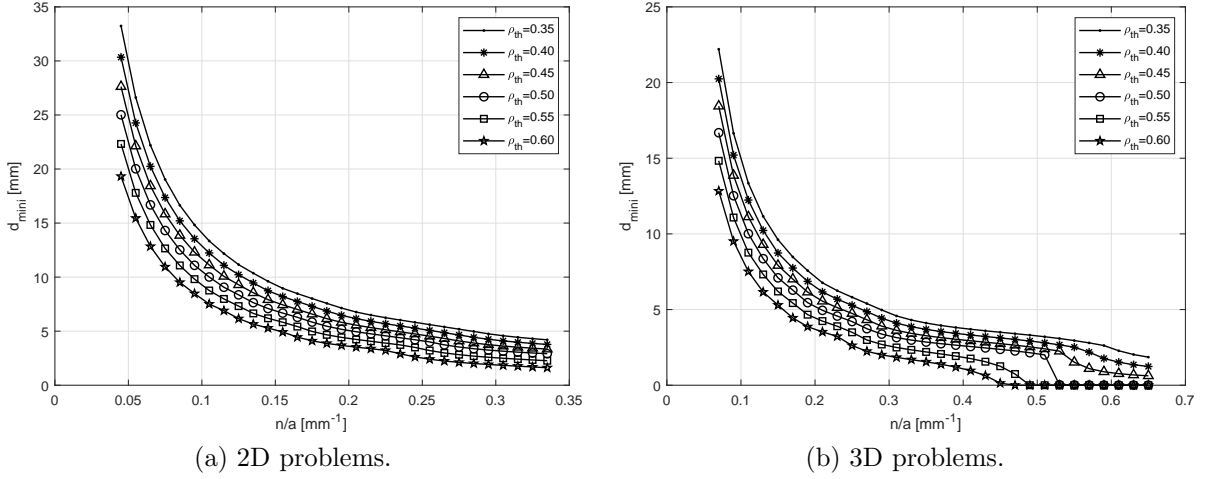


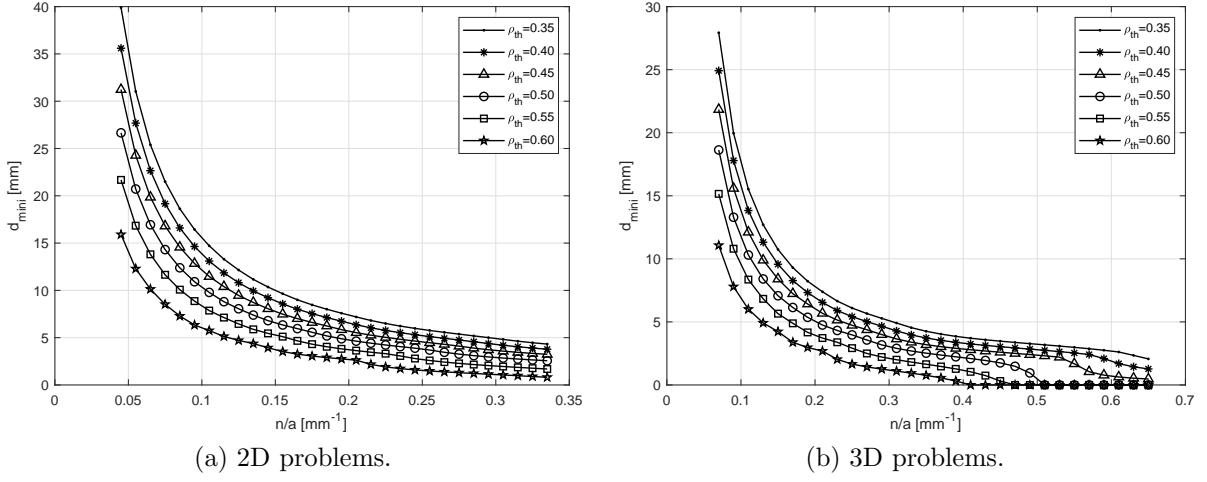
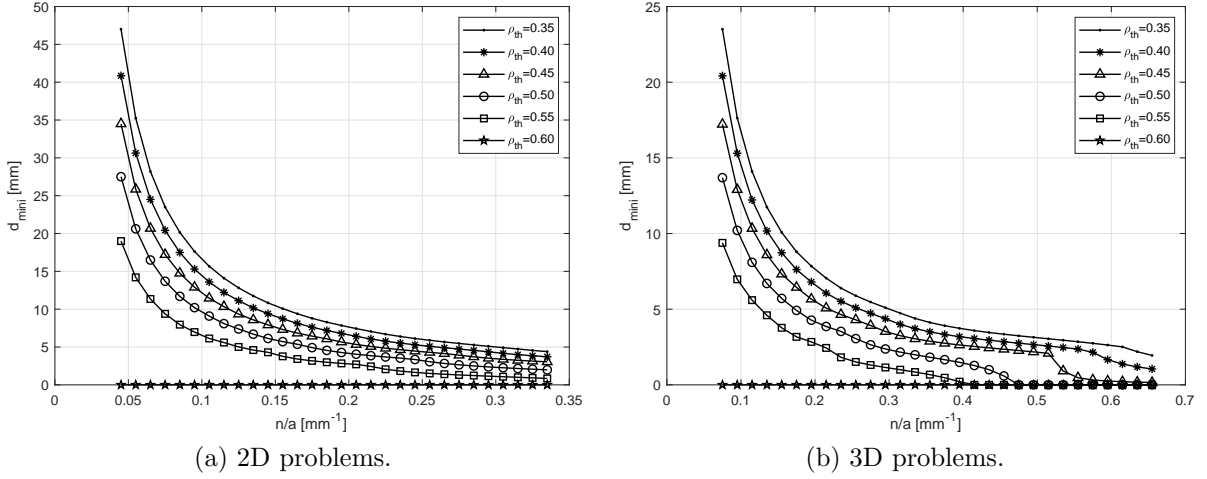
Figure 5.11 –  $d_{min}$  vs.  $n/a$ ,  $p_j = 2$ .

computation domain, wherein Eq. (5.14) should be replaced by

$$\Delta X_0^{(j)} = \frac{a_j}{p_j} \Delta U^{(j)}, j = 1, 2, 3. \quad (5.16)$$

In Eq. (5.16),  $\Delta X_0^{(j)}$  indicates the value of  $\Delta X^{(j)}$  at the boundary. Therefore, the idea is to provide design *abaci* similar to those of Figs. 5.8-5.10 by using Eq. (5.16) at point 5 of the aforementioned algorithm. The *abaci* related to the minimum member size near the boundary of the computation domain, which is denoted  $d_{min}^B$ , are illustrated in Figs. 5.14-5.16. These *abaci* show the same trend of the previous ones and they can be used as a design tool in order to forecast the minimum member size at the boundary of the design




 Figure 5.12 –  $d_{min}$  vs.  $n/a$ ,  $p_j = 3$ .

 Figure 5.13 –  $d_{min}$  vs.  $n/a$ ,  $p_j = 4$ .

domain.

It is noteworthy that the minimum member size next to the computational domain boundary constitutes a special condition that deserves a particular attention. To understand this point, let consider a 2D TO problem: the B-Spline degrees are set to  $p_1 = p_2 = 3$  and the domain characteristic dimensions are  $a_1 = 320$  mm and  $a_2 = 200$  mm. Suppose that the demanded minimum member size is  $d_{min} = 5$  mm. The corresponding graph of Fig. 5.8 is examined and it can be retrieved that  $\Delta X = 5.5$  mm for an estimated threshold value  $\rho_{th} = 0.5$ . Accordingly, the graph of Fig. 5.12a provides a control points density  $n/a = 0.191$ . Therefore,  $n_1 + 1 = 62$  and  $n_2 + 1 = 39$  are enough to obtain a final design meeting the imposed minimum member size constraint. However, this control

### 5.3. Minimum Length Scale control in the NURBS-based TO Algorithm

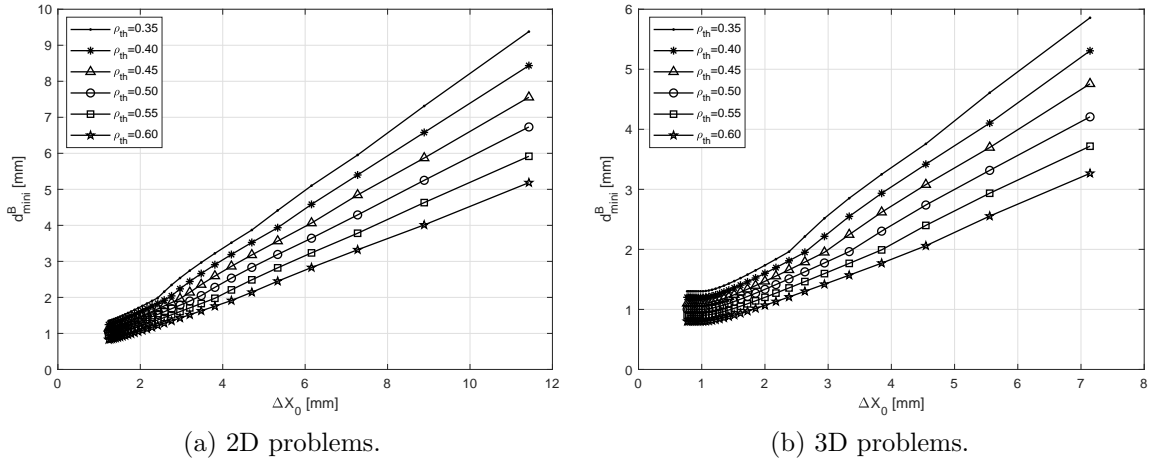


Figure 5.14 –  $d_{min}^B$  vs.  $\Delta X_0$ ,  $p_j = 2$ .

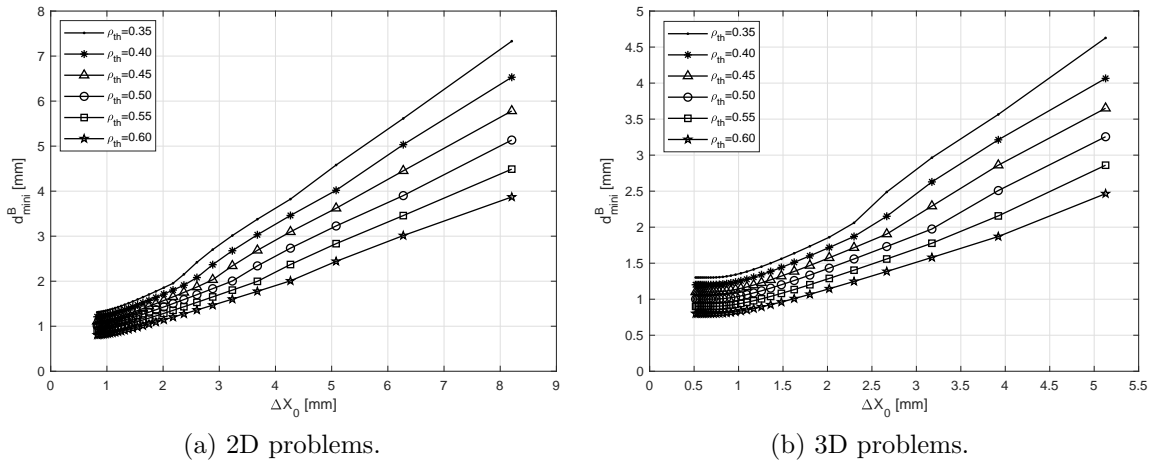
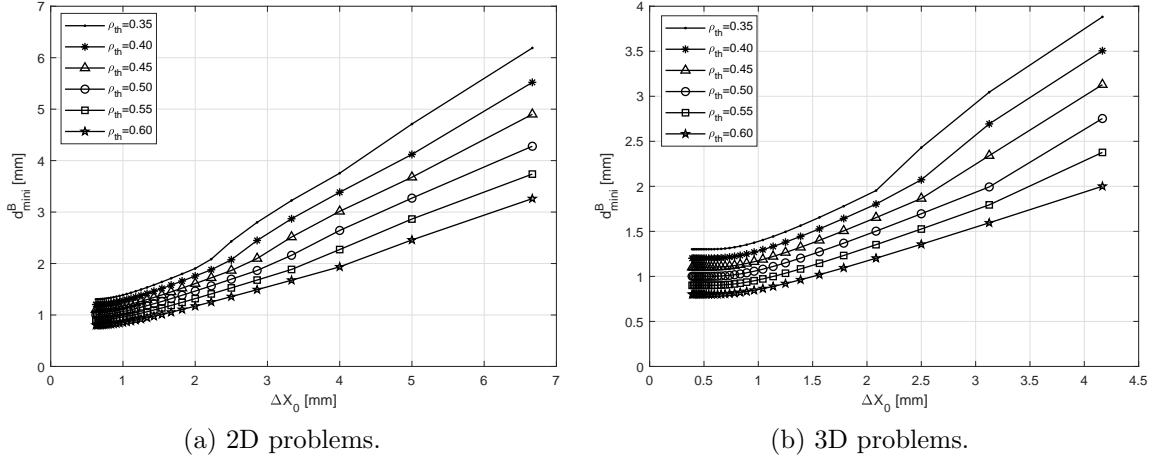


Figure 5.15 –  $d_{min}^B$  vs.  $\Delta X_0$ ,  $p_j = 3$ .

points distribution defines  $\Delta X_0 = \Delta X/3 = 1.83$  mm according to Eq. (5.16), that implies an expected minimum member size  $d_{min}^B = 1.35$  mm, as it can be deduced from the graph of Fig. 5.15a. This result does not mean that the design will surely provide a minimum member size of 1.35 mm, but it rather warns the designer that the minimum member size could decrease up to 1.35 mm near the boundary. Under these circumstances, the designer has two choices: on the one hand, the demanded minimum member size condition can be forced on the design domain boundary, i.e. a greater  $\Delta X_0$  is chosen, and, accordingly, a smaller number of control points; on the other hand, the designer can try to run the TO computation and then analyse the resulting configuration. In any case, the method allows the designer to be aware about the effects of his choices: reducing too much the number of control points could lead to a poor description of the final topology and, consequently,


 Figure 5.16 –  $d_{min}^B$  vs.  $\Delta X_0$ ,  $p_j = 4$ .

to inefficient configurations. Contrariwise, launching the TO analysis with  $n_1 + 1 = 62$  and  $n_2 + 1 = 39$  could produce too thin features in the neighbourhood of the domain boundary, that could be unsatisfactory from a technological/mechanical viewpoint.

A smart way to overcome this dichotomy is the utilisation of a non-uniform knot vector, i.e. a knot vector which is not characterised by equally spaced components. In this case, design *abaci* similar to those of Figs. 5.8-5.10 can be provided. However these graphs are not reported here for sake of brevity. Rather the beneficial effect of the use of a non-uniform knot vector on the final optimised topology and on the related minimum length scale is directly illustrated on a meaningful benchmark in section 5.3.6.

### 5.3.2 Some remarks about the proposed approach

The approach making use of the previous *abaci* presents different aspects of outstanding importance.

The method is simple and intuitive for the designer and does not need the introduction of a further constraint in the TO problem formulation. This fact is of paramount importance because constraints involving the minimum member size could become burdensome from a computational viewpoint. Furthermore, these constraints are often not met on the reassembled geometry at the end of the optimisation process. Contrariwise, the proposed approach allows for setting a pertinent combination of number of control points and degrees of the NURBS blending functions, which automatically satisfy the imposed minimum member size. The provided design curves aim at forecasting  $d_{min}$  on the reassembled geometry, in order to obtain consistent results.

Even if the advantages of the NURBS-based approach have been already shown in Chapter 4, the choice of the NURBS discrete parameters (i.e. the number of control points and/or the degree) was not unique and it was left to the designer experience. By

means of the *abaci* of Figs. 5.8-5.10, it is possible to choose a suitable number of control points by setting the desired minimum member size.

Nevertheless, the proposed approach is not free of drawbacks. There are two most evident issues to be discussed. Firstly, the designer must arbitrarily set the degree of the blending functions and, *a priori*, he should try several degrees in order to understand the influence of this parameter on the final optimum topology. Secondly, the designer is obliged to set a threshold density. However, the actual value of this quantity is provided at the end of the optimisation in order to meet the imposed constraints (see Chapters 1 and 4); this means that  $\rho_{th}$  is unknown before performing the TO analysis. Anyway, these are only minor issues that can be easily overcome.

The range of degrees to be used for the TO analysis is limited because, as widely explained in Chapter 4, high degrees hamper the correct convergence of the algorithm towards an efficient solution, especially when the number of control points is small. Therefore, there is no interest in using high degrees and the designer can try just the values  $p_j = 2, 3, 4$ .

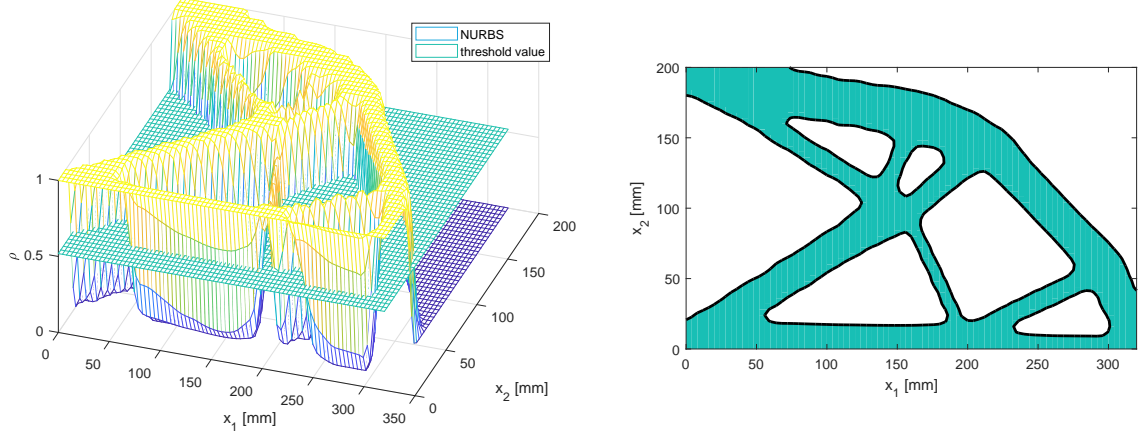
The first choice for the threshold density value should be 0.5 and 0.45 for 2D and 3D applications, respectively. Then, if the problem at hand presents particular constraints, which could lead to a poor convergence behaviour, the previous values can be reduced. If the designer is uncertain among two values for  $\rho_{th}$ , the highest value should be set in order to make the most conservative choice.

#### 5.3.3 The effects of the NURBS weights on the minimum length scale

Including the NURBS weights among the design variables in a NURBS-based TO strategy leads to some advantages. As discussed in Chapter 4, better performances can be achieved when weights and control points are optimised at the same time and final geometries exhibit a smoother boundary. However, their use should be carefully assessed because choosing a NURBS rather than a B-Spline implies doubling the number of design variables if the same number of control points is used.

The minimum member size should be forecast in the case of NURBS as well. The main difficulty arising by considering weights as design variables consists of a lack of a simple relationship between physical coordinates and the respective knot vector. In other words, the Greville's abscissae formula in the form of Eq. (4.8) does not hold any more, so it is not possible to easily calculate the physical coordinates  $x_j$ ,  $j = 1, 2, 3$  of control points on the design domain. Moreover, it should be remarked that weights are now design variables and that their value is not *a priori* known. Since the previously depicted method seemed to be sound enough for B-Splines entities, it is sought to exploit the methodology also in the case of NURBS. To this purpose, the simple benchmark of Fig. 4.7 related to problem (4.23) has been considered again. Particularly, the discussion

is focused on the NURBS solution of Fig. 4.11c ( $n_1 = 47$ ,  $n_2 = 29$  and  $p_1 = p_2 = 3$ ), that is reported in Fig. 5.17 for convenience. As it can be observed from Fig. 5.17b, a



(a) NURBS surface and related threshold value. (b) The final optimised topology of the 2D structure.

Figure 5.17 – The 2D optimised topology provided by the NURBS-based SIMP method.

thin topological element appears in the middle-bottom zone of the domain. Therefore, the control points and weights affecting this part of the design domain are observed. Control points coordinates and weights are ranged in two matrices with  $n_1 + 1$  rows and  $n_2 + 1$  columns, as explained in Chapter 2. If the column index is set and the trend of both control points and weights is observed “column-wise”, this will be equivalent to look at the behaviour of the aforementioned quantities along an horizontal line drawn on the design domain. Particularly, it is sought to identify some situations reproducing the minimum length scale condition, as those of Fig. 5.4. Some of the most representative trends are shown in Fig. 5.18.

In each graph, the  $d_{min}$  condition is reproduced by means of a single peak in the control points density trend, i.e. a single control point density is set to 1 and that of its neighbours is set to 0. In this particular condition, the trend of the corresponding weights is observed: the weight corresponding to the peak control point is always greater than 1, whilst the weights related to the two closest control points at the two sides of the peak are lower than 1. This fact means that, when the minimum length scale condition is reproduced, a TO pseudo-density field represented through a NURBS surface is more pushed towards the material phase rather than the equivalent field represented by means of a B-Spline surface. Thus, the most critical condition (i.e. the lowest value of member size) is obtained with B-Spline surfaces and not with NURBS surfaces. The natural conclusion of this study is that the design graphs of section 5.3.1 can be used for B-Spline and NURBS entities as well, since in any case they will constitute a conservative estimation of the minimum length scale.

### 5.3. Minimum Length Scale control in the NURBS-based TO Algorithm

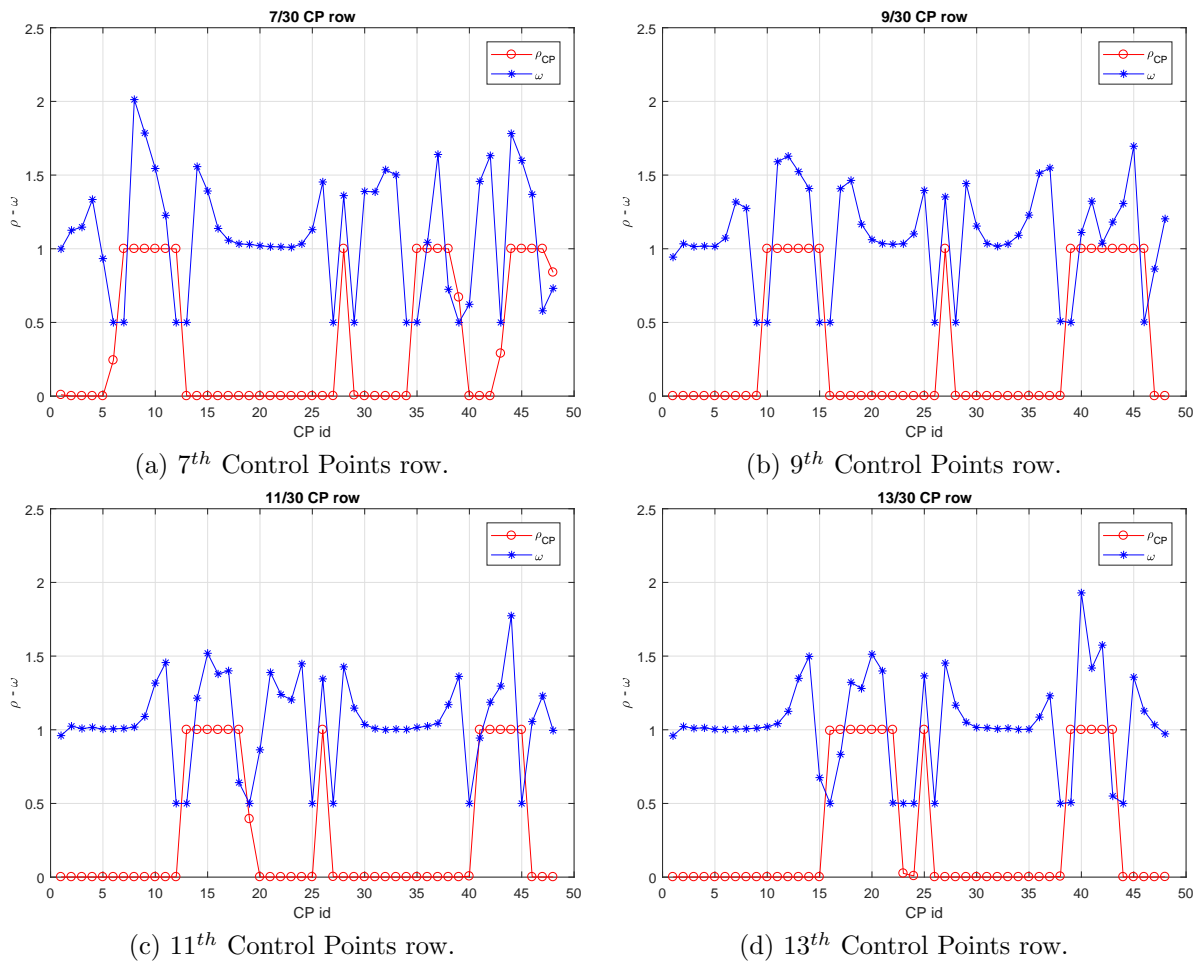


Figure 5.18 – Weights and Control Points trends for a NURBS solution.

Of course, the minimum length scale cannot be forecast in a rigorous manner when NURBS entities are used. The use of the design graphs, that is sound for B-Spline entities, can be only partially justified for NURBS entities.

Indeed, the graphs of section 5.3.1 are referred to  $\Delta X$  and  $n/a$ . In the case of NURBS, these quantities must be interpreted as average quantities, since it is not possible to locate control points according to the Greville's abscissae formula.

Finally, even if the validity of the proposed method cannot be rigorously justified in the case of NURBS entities, its effectiveness is empirically shown on some meaningful benchmarks in sections 5.3.4 and 5.3.5.

### 5.3.4 Results: Minimum length scale in 2D

The effectiveness of the *abaci* used to forecast the minimum length scale is proven in this paragraph for 2D problems through the simple benchmark of Fig. 5.19 (refer to related caption for details about the geometry and the FE analysis). The TO problem is formu-

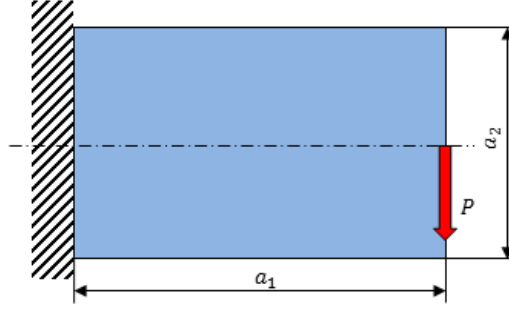


Figure 5.19 – 2D benchmark - Geometric parameters:  $a_1 = 320$  mm,  $a_2 = 200$  mm, thickness  $t = 2$  mm - Material parameters:  $E = 72000$  MPa,  $\nu = 0.33$  - Mesh:  $96 \times 60$  SHELL181 elements - Load:  $P = 1000$  N.

lated according to Eq. (4.10), where  $V_{ref} = a_1 a_2 t$  and  $\gamma = 0.4$ . Furthermore, a symmetry condition is imposed with respect to the plane  $x_2 = a_2/2$ . When the convergence is achieved, the result is exported in IGS format and the 2D solution is retrieved by means of the threshold operations described in Chapter 4. This operation is performed in the CAD environment of CATIA<sup>®</sup> (Fig. 5.20a). Once the 2D structure is obtained, the actual minimum thickness of structural elements is identified and measured. The idea behind the minimum length scale is pretty clear but a mathematically exact definition does not exist: therefore, it has been conventionally established that the measured minimum length scale is the diameter of the smallest circle that can be inscribed within the structure (refer to Fig. 5.20b).

The method presented in section 5.3.1 is tested by comparing the minimum length scale predicted by graphs of Figs. 5.11-5.13 and Figs. 5.14-5.16 with the measured minimum member size on the actual reassembled structure. In particular, the benchmark problem has been solved by making use of both B-Spline and NURBS surfaces (polynomials degrees  $p_j = 3$ ,  $j = 1, 2, 3$ ) and the TO analysis has been repeated several times in order to change the number of control points. To be remarked that this campaign of results has been obtained with uniform knot vectors.

Solutions are shown in Fig. 5.21 and numerical results are collected in Tables 5.1 and 5.2, wherein the compliance is divided by a reference quantity  $c_{ref} = 2625$  Nmm, that is the compliance of the structure with a uniform initial density  $\rho(u_1, u_2) = \gamma$ . The measured minimum length scale is referred as  $d_{min}^M$ , whilst the forecast value of the minimum length scale is referred as  $d_{min}^B$  on the internal zones and  $d_{min}^B$  on the boundary

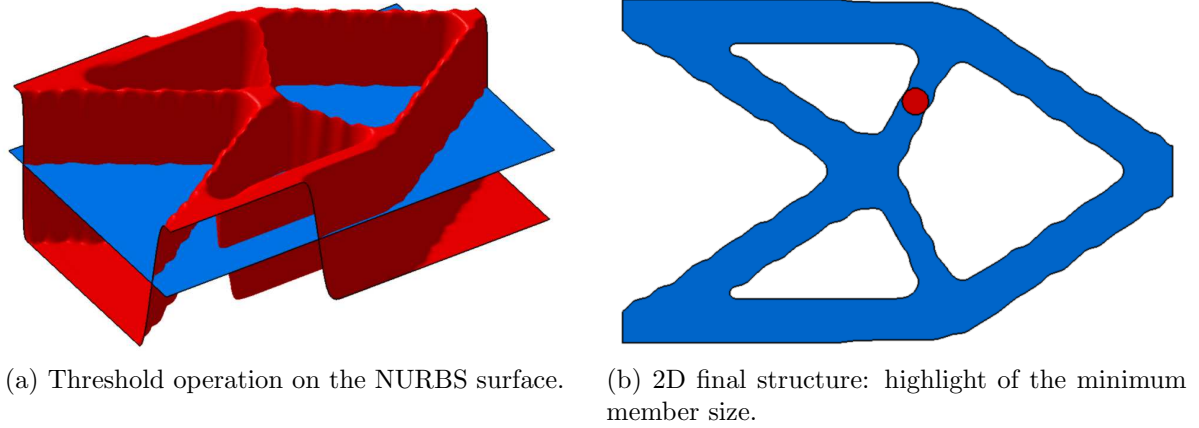


Figure 5.20 – Procedure to measure the actual minimum member size.

of the design domain, respectively. The two values  $d_{min}$  and  $d_{min}^B$  are evaluated for an estimated threshold value of the density  $\rho_{th} = 0.5$ .

In the following Tables, the ( $B$ ) symbol appears next to the values of  $d_{min}^M$  when the minimum member size is measured in the neighbourhood of the boundary; otherwise, the critical zone wherein  $d_{min}^M$  is measured occurs within the design domain.

$n_{tot}$	$c/c_{ref}$ [Nmm]	$d_{min}^M$ [mm]	$d_{min}$ [mm]	$d_{min}^B$ [mm]
$16 \times 10$	0.263	19.705 ( $B$ )	26.650	5.14
$32 \times 20$	0.178	16.340	12.010	2.34
$48 \times 30$	0.151	16.170	6.765	1.59
$64 \times 40$	0.146	5.173	4.937	1.32
$80 \times 50$	0.144	6.470	3.575	1.16
$96 \times 60$	0.145	4.960	2.910	1.08

Table 5.1 – Minimum length scale for 2D B-Spline solutions

As it can be retained from Tables 5.1 and 5.2, the minimum length scale is correctly forecast for both B-Spline and NURBS solutions. The previous claim must be interpreted in this sense: the minimum length scale that is forecast through the proposed methodology is always lower than the actual minimum length scale that is measured on the CAD-reassembled geometry. In this sense the method is conservative.



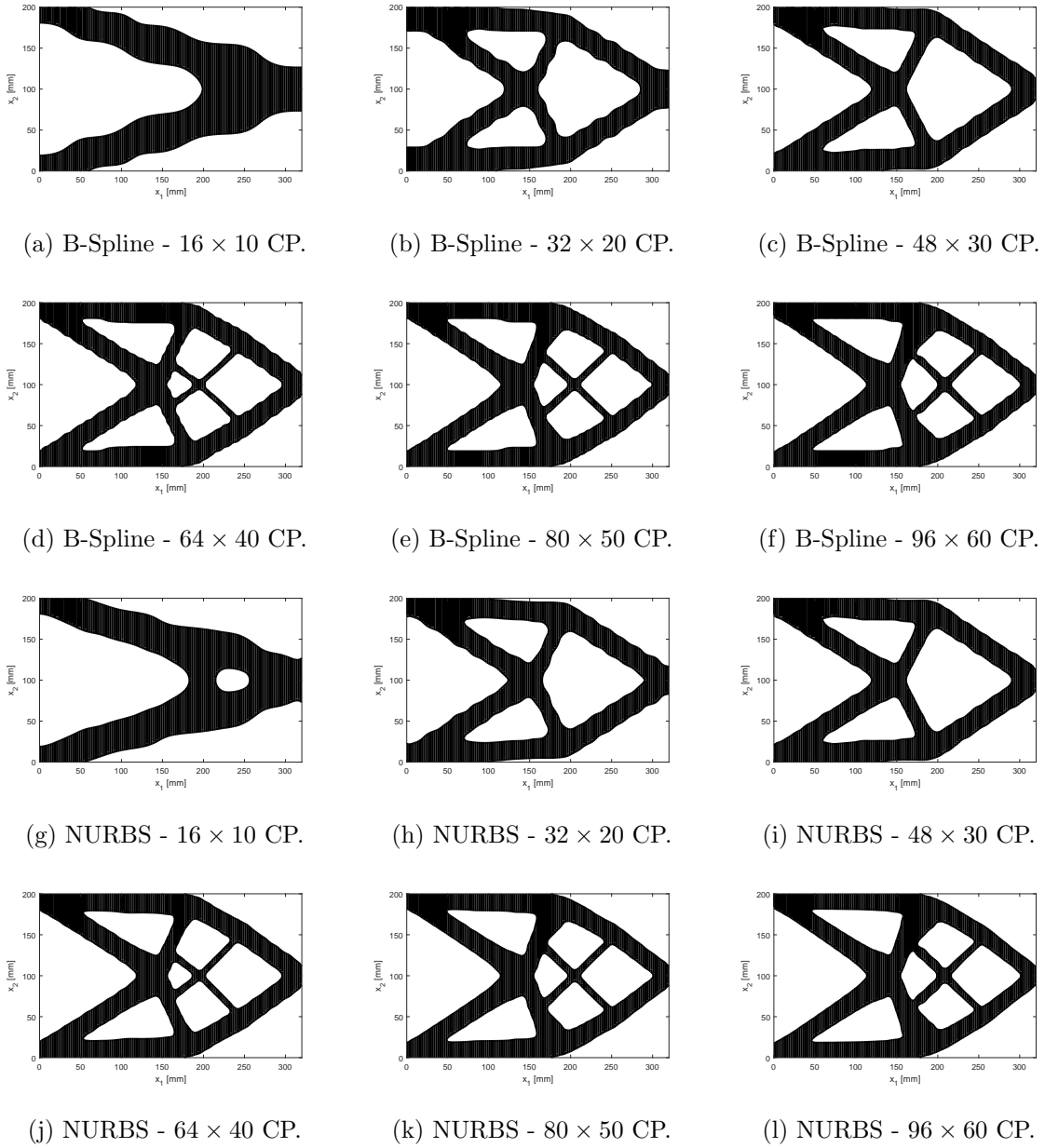


Figure 5.21 – 2D solutions - minimum length scale,  $p_j = 3$ .

### 5.3.5 Results: Minimum length scale in 3D

Because of the complexity of 3D topologies, the minimum length scale condition is checked in a different way with respect to the case of 2D structures. The problem of Eq. (4.10) is solved for the benchmark of Fig. 5.22. Two symmetry conditions, with respect to the

$n_{tot}$	$c/c_{ref}$ [Nmm]	$d_{min}^M$ [mm]	$d_{min}$ [mm]	$d_{min}^B$ [mm]
$16 \times 10$	0.210	18.905 ( $B$ )	26.650	5.14
$32 \times 20$	0.154	14.470	12.010	2.34
$48 \times 30$	0.142	17.159	6.765	1.59
$64 \times 40$	0.142	6.586	4.937	1.32
$80 \times 50$	0.143	6.946	3.575	1.16
$96 \times 60$	0.145	7.301	2.910	1.08

Table 5.2 – Minimum length scale for 2D NURBS solutions

planes  $x_2 = a_2/2$  and  $x_3 = a_3/2$ , are added. Solutions are provided in the following for different control points numbers and by setting the B-Spline/NURBS hyper-surfaces degrees equal to 3.

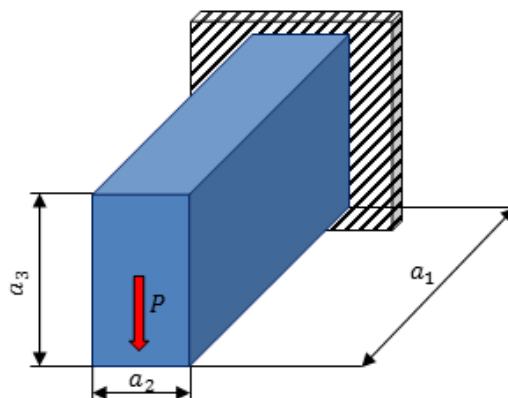


Figure 5.22 – 3D benchmark - Geometric parameters:  $a_1 = 400$  mm,  $a_2 = 100$  mm,  $a_3 = 200$  mm - Material parameters:  $E = 72000$  MPa,  $\nu = 0.33$  - Mesh:  $64 \times 16 \times 32$  SOLID185 elements - Load:  $P = 2000$  N.

For 3D problems, the optimised configurations can be reassembled, at the end of the optimisation process, in the form of STL files. The graphs of section 5.3.1 can be used in order to forecast the minimum member size within the domain and on its boundary. The STL file collects the  $n_{TR}$  triangles composing the boundary of the optimised volume, thus the local outward normal vector  $\mathbf{n}$  can be identified on each boundary surface. This information can be exploited to measure the actual minimum member size at the end of the optimisation, in order to check the effectiveness and the robustness of the approach based on the aforementioned *abaci* to forecast this quantity.

It is noteworthy that the boundary of the 3D solid is retrieved from the knowledge of the iso-surface  $\rho = \rho_{th}$  of the fictitious density field. Considering the normal vector, the

fourth coordinate of the NURBS hyper-surface (which describes the pseudo-density field) takes values  $\rho < \rho_{th}$  (no material phase) along the outward direction and  $\rho > \rho_{th}$  along the inward direction (material phase). Therefore, the idea is to move from the iso-surface towards the material phase at least for a distance equal to the minimum member size and to check if the opposite side of the boundary is still in the material phase. The main steps realising such operations are described in Algorithm 2.

---

**Algorithm 2** Minimum length scale check for 3D problems.

---

- 1: Two counters are set  $n_F = 0$  and  $n_{UF} = 0$ . Set  $j = 0$ .
  - 2:  $j = j + 1$ .
  - 3: Determine the Cartesian coordinates of the center of gravity of the  $j$ -th triangle, i.e.  $\mathbf{OG}_j$ .
  - 4: If the topology boundary is inside the computation domain, move along the local inward direction according to  $\mathbf{OG}'_j = \mathbf{OG}_j - d_{min}\mathbf{n}$ ; if the topology boundary is located on the boundary of the computation domain move along the local inward direction according to  $\mathbf{OG}'_j = \mathbf{OG}_j - d_{min}^B \mathbf{n}$ .
  - 5: The B-Spline/NURBS is evaluated in  $\mathbf{OG}'_j$ : if  $\rho_{\mathbf{OG}'_j} > \rho_{th}$ ,  $n_F = n_F + 1$ ; otherwise, if  $\rho_{\mathbf{OG}'_j} < \rho_{th}$ ,  $n_{UF} = n_{UF} + 1$ .
  - 6: If  $j < n_{TR}$  go to point 2, else go to point 7.
  - 7: Evaluate the fraction of triangles satisfying the minimum length scale, i.e.  $f_F = n_F/n_{TR}$ . The fraction of triangles violating such a constraints is denoted as  $f_{UF} = 1 - f_F$ .
- 

Numerical results are collected in Tables 5.3 and 5.4: for each solution, the objective function value is highlighted together with the forecast minimum length scale within the domain  $d_{min}$  and on the boundary  $d_{min}^B$ . The fractions  $f_F$  and  $f_F^B$  of “feasible” triangles are reported as well.

$n_{tot}$	$c/c_{ref}$ [Nmm]	$d_{min}$ [mm]	$d_{min}^B$ [mm]	$f_F$	$f_F^B$
$24 \times 6 \times 12$	0.0873	15.000	5.000	0.978	0.958
$30 \times 12 \times 18$	0.0700	10.295	2.356	0.936	0.974
$36 \times 18 \times 24$	0.0620	6.136	1.522	0.982	0.999

Table 5.3 – Minimum length scale for 3D B-Spline solutions

The topologies corresponding to the configurations appearing in Tables 5.3 and 5.4 are shown in Figs. 5.23-5.25 for B-Spline solutions and in Figs. 5.26-5.28 for NURBS solutions. In particular, boundary in red highlights the domain regions where the minimum length scale is correctly forecast, whilst boundary in blue is constituted of “infeasible” triangles, i.e. those zones characterised by a thickness smaller than that forecast by means of the *abaci*.

### 5.3. Minimum Length Scale control in the NURBS-based TO Algorithm

$n_{tot}$	$c/c_{ref}$ [Nmm]	$d_{min}$ [mm]	$d_{min}^B$ [mm]	$f_F$	$f_F^B$
$24 \times 6 \times 12$	0.0706	15.000	5.000	1.000	0.880
$30 \times 12 \times 18$	0.0653	10.295	2.356	0.924	0.971
$36 \times 18 \times 24$	0.0608	6.136	1.522	1.000	0.998

Table 5.4 – Minimum length scale for 3D NURBS solutions

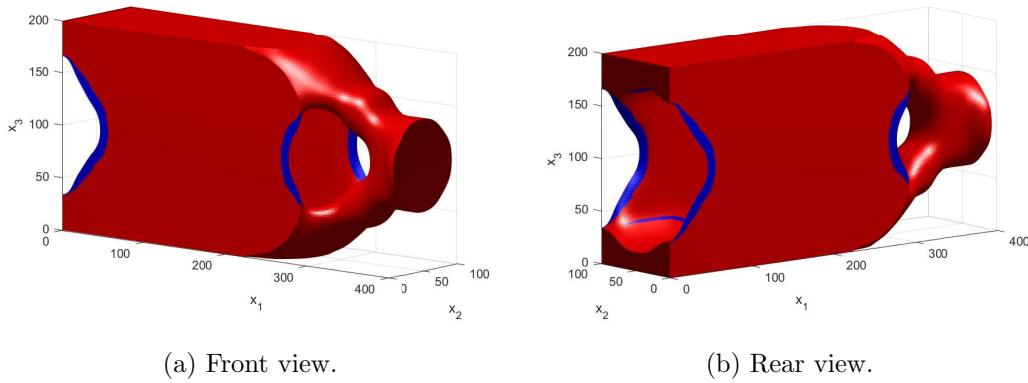


Figure 5.23 – Highlight of the minimum length scale in 3D, B-Spline solution,  $p_j = 3$ ,  $24 \times 6 \times 12$  control points.

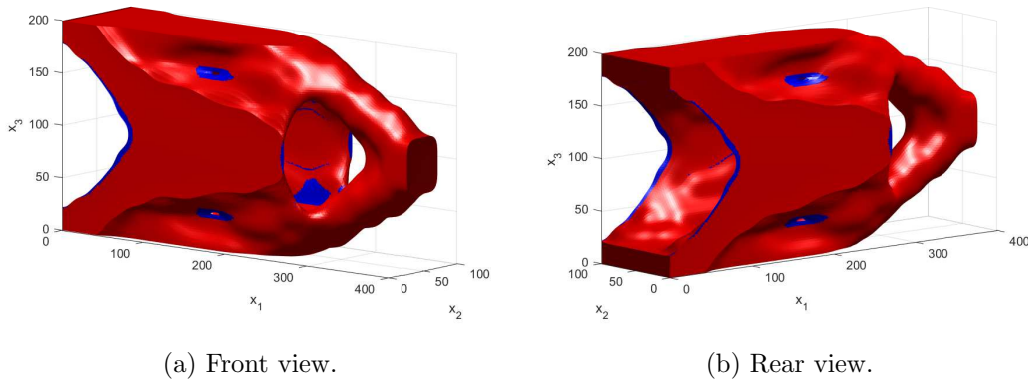


Figure 5.24 – Highlight of the minimum length scale in 3D, B-Spline solution,  $p_j = 3$ ,  $30 \times 12 \times 18$  control points.

Results clearly show that the minimum length scale is correctly forecast in a wide zone of the domain. The minimum member size is smaller than the predicted value only in very circumscribed regions. The fractions of Tables 5.3 and 5.4 are not exactly 1 because

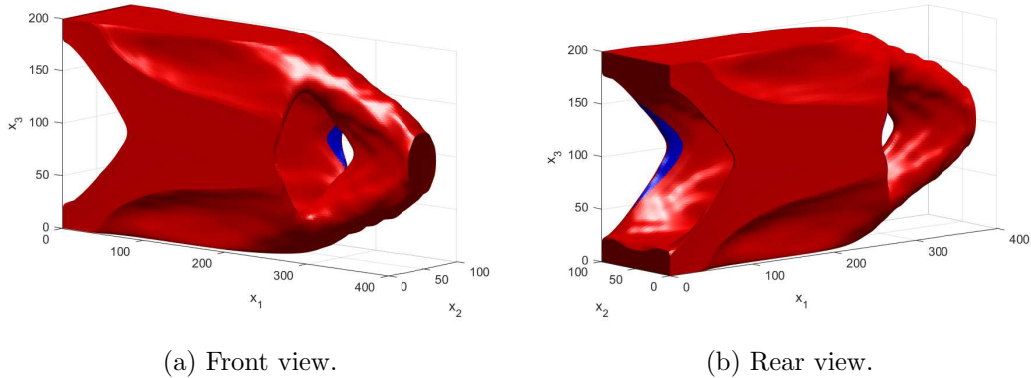


Figure 5.25 – Highlight of the minimum length scale in 3D, B-Spline solution,  $p_j = 3$ ,  $36 \times 18 \times 24$  control points.

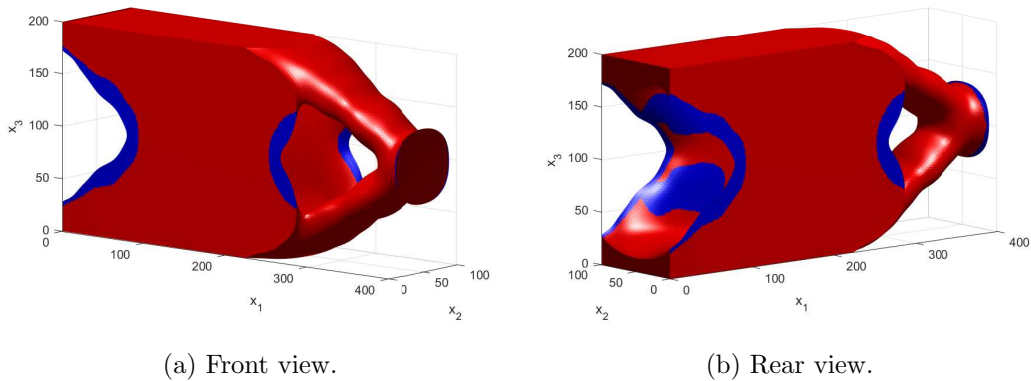


Figure 5.26 – Highlight of the minimum length scale in 3D, NURBS solution,  $p_j = 3$ ,  $24 \times 6 \times 12$  control points.

of several reasons. Firstly, the estimation of the minimum member size is based on the assumptions described in section 5.3.1, which can be not met in whatever circumstances. Secondly, while the method seems quite intuitive in 2D, its extrapolation in 3D is not immediate. In particular, for 3D problems the evaluation of the pseudo-density threshold value is not unique and some trial-and-error approach is required (at the end of the optimisation process) before achieving a good compromise in terms of performances and minimum length scale requirement.

Anyway, it is noteworthy that the proposed strategy is both simple and reliable enough to be used in the NURBS-based SIMP algorithm.

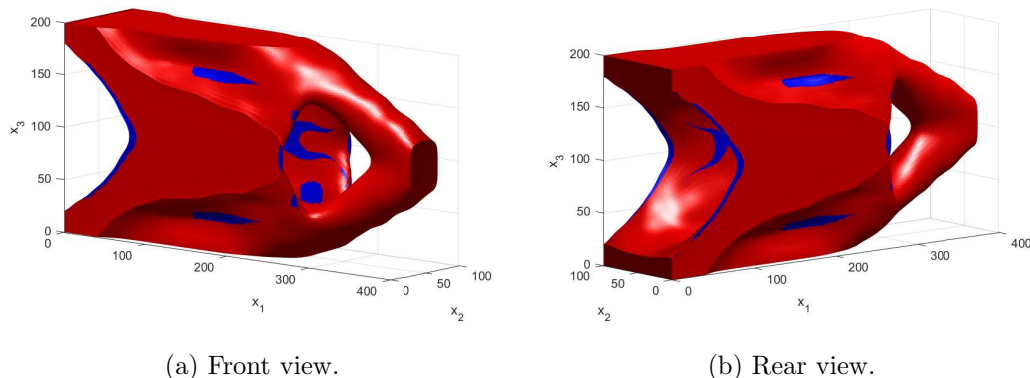


Figure 5.27 – Highlight of the minimum length scale in 3D, NURBS solution,  $p_j = 3$ ,  $30 \times 12 \times 18$  control points.

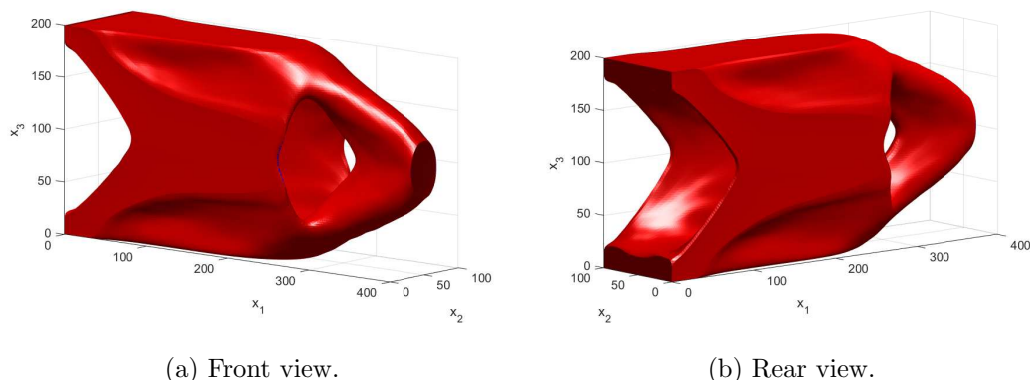
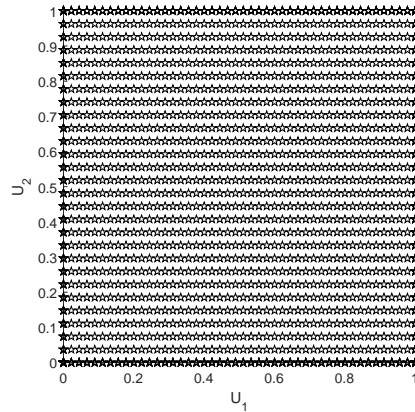


Figure 5.28 – Highlight of the minimum length scale in 3D, NURBS solution,  $p_j = 3$ ,  $36 \times 18 \times 24$  control points.

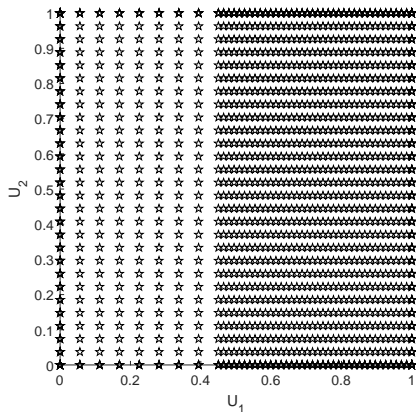
### 5.3.6 The effects of a non-uniform knot vector on the minimum length scale

Since only uniform knot vectors have been considered until now, it seems interesting to investigate the influence of a non-uniform knot vector on the optimised topology. In parallel, the robustness of the method for predicting the minimum length scale is tested. Here, only 2D TO problems are considered, but the extension of results to the 3D case is straightforward. Let consider the 2D benchmark of section 5.3.4, and, in particular, the solution corresponding to  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$  (Figs. 5.21c and 5.21i illustrate the B-Spline and the NURBS solutions, respectively). Let now focus on the knot vectors distributions: in cases of Figs. 5.21c and 5.21i, the net constituted by the knot vectors

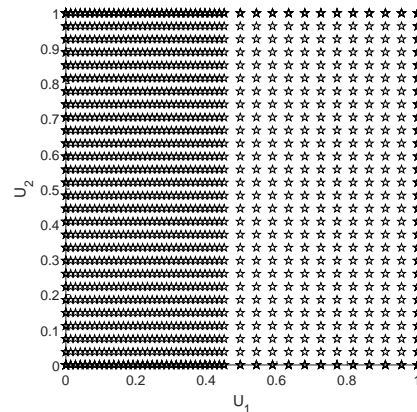
components is represented in Fig. 5.29a. Of course, there is a strict analogy between the knot vectors components distribution in the parametric domain and the control points distribution in the physical space. Two further non-uniform distribution of the knot-vector components are considered for this example: they are shown in in Figs. 5.29b and 5.29c.



(a) Uniform distribution.



(b) Clumping 1.



(c) Clumping 2.

Figure 5.29 – Knot vectors distributions.

The same optimisation problem is run for the 2D cantilever plate by applying these two different knot vectors and results are shown in Fig. 5.30 for B-Spline surfaces and in Fig. 5.31 for NURBS ones.

The results of Figs. 5.30 and 5.31 highlight one strong potential of the NURBS-based TO algorithm: the same mesh and the same number of control points and polynomials degrees can produce different topologies thanks to the knot vectors distributions. It is interesting to check that the minimum member size forecast by means of the graphs of

### 5.3. Minimum Length Scale control in the NURBS-based TO Algorithm

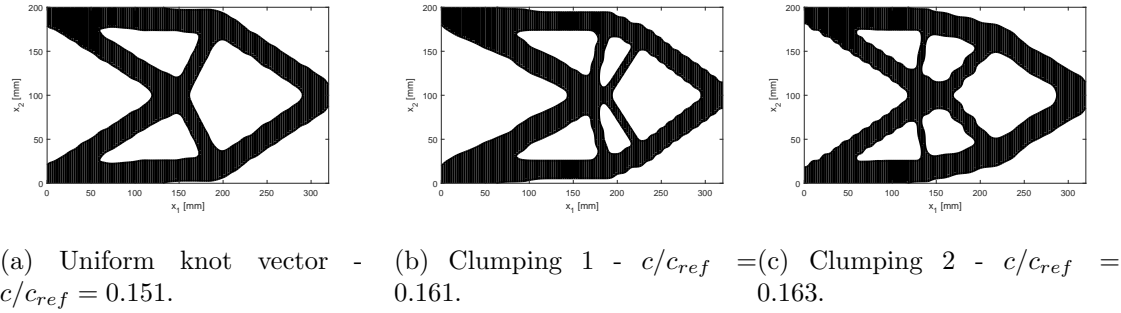


Figure 5.30 – B-Spline solutions with different knot vector distributions.

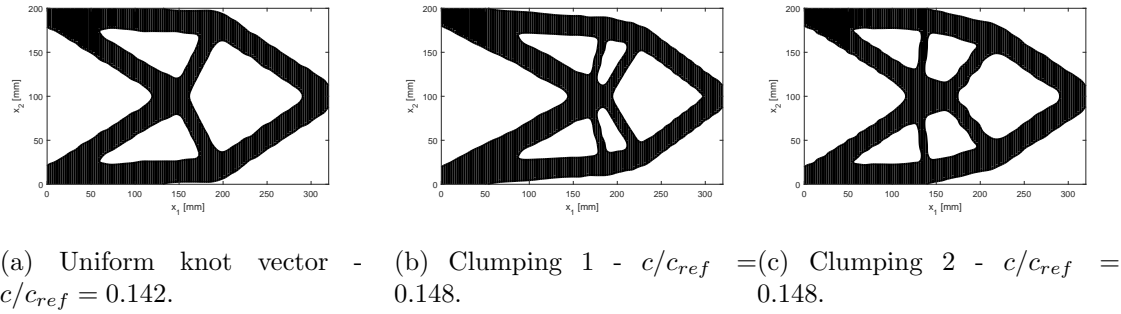


Figure 5.31 – NURBS solutions with different knot vector distributions.

section 5.3.1 is always conservative (the estimated threshold value is  $\rho_{th} = 0.5$ ). In fact, if the first knot vector clumping is considered, the most critical value  $\Delta X = 4.756$  mm is obtained along the  $x_1$  axis on the right side of the domain: the corresponding minimum member size, forecast by using the graph of Fig. 5.8, is  $d_{min} = 4.314$  mm. The actual (measured) minimum member size is  $d_{min}^M = 7.7$  mm for the B-Spline solution (Fig. 5.30b) and  $d_{min}^M = 7.6$  mm for the NURBS solution (Fig. 5.30c). As far as the second knot vector clumping is considered, the critical value  $\Delta X = 4.367$  mm is obtained along the  $x_1$  axis on the left side of the domain. In this case,  $d_{min} = 3.986$  mm and  $d_{min}^M = 5.1$  mm for the B-Spline solution (Fig. 5.31b) while  $d_{min}^M = 7.0$  mm for the NURBS solution (Fig. 5.31c). Therefore, these results put in evidence that, when the number of control points is kept constant, the designer can distribute the knot vectors components in order to have different dimensions of topological features on specified domain zones. Moreover, the generality of the method is proven, since the  $\Delta X$  must be interpreted as a local information, regardless if it results from a uniform or non-uniform knot vector components distribution.



## 5.4 The Maximum Length Scale

### 5.4.1 Mathematical Statement of the Maximum Length Scale constraint

The maximum length scale constraint is used in TO analyses in order to limit the maximum thickness of topological features appearing during optimisation. Using the maximum and the minimum member size simultaneously is a smart choice to obtain structures with uniform dimensions. Unlike the minimum length scale constraint, that was implicitly defined thanks to the NURBS formalism and properties, the maximum length scale cannot be forecast and controlled by simply setting the NURBS discrete parameters. Therefore, an explicit constraint should be implemented. Furthermore, this constraint should be able to check all the design domain and to penalise too thick material zones. Taking inspiration from Guest's work [43], a general formulation is provided here for the maximum length scale constraint in the framework of the NURBS-based SIMP approach for both 2D and 3D problems.

The intuitive idea is straightforward: accounted for an assigned structure, a circular or spherical region is drawn around each mesh element centroid for 2D and 3D problems, respectively. The circle/sphere diameter is equal to the imposed maximum length scale (referred as  $d_{max}$  hereafter). Let  $\Omega_e$  be the area/volume of the circular/spherical region centred at element  $e$ ; its measure can be computed as:

$$\Omega_e = \int_{\Omega_e} d\Omega = \begin{cases} \pi \left( \frac{d_{max}}{2} \right)^2, & \text{if } N = 2, \\ \frac{4}{3} \pi \left( \frac{d_{max}}{2} \right)^3, & \text{if } N = 3. \end{cases} \quad (5.17)$$

Thus, the following condition must be met for each element:

$$\sum_{i \in \Omega_e} \bar{\rho}_i \mu_i \leq \Omega_e (1 - \psi), \forall e. \quad (5.18)$$

In Eq. (5.18),  $i$  is a mute index to indicate the mesh elements whose centroid falls into  $\Omega_e$  (this set can be referred as the ‘‘neighbourhood’’ of element  $e$ ),  $\psi$  is a relaxing parameter ( $0 < \psi < 1$ ),  $\mu_i$  is the measure of the element  $i$  (the element volume  $V_i$  in 3D and the element surface  $A_i$  in 2D), i.e.

$$\mu_i = \begin{cases} A_i, & \text{if } N = 2, \\ V_i, & \text{if } N = 3, \end{cases} \quad (5.19)$$

and  $\bar{\rho}_i$  is the projected fictitious density function. In this work, such a projection is performed through the relation

$$\bar{\rho}_e = \rho_e^\beta, \quad (5.20)$$

where  $\beta \geq 1$  is a penalisation parameter (its effects are similar to those of the parameter  $\alpha$ , typical of the SIMP method), refer to Eq. (1.2). In other words, it is explicitly asked that the maximum material phase thickness is inferior to an assigned dimension, i.e.  $d_{max}$ , on the whole design domain. However, this formulation implies a number of optimisation constraints equal to  $N_e$ , that would be impossible to be handled. Hence, a suitable aggregation strategy should be considered. The natural choice, that would avoid compensatory side effects, consists of choosing the maximum value of the left-hand side of Eq. (5.18) among the mesh elements. Let  $\delta_e$  be defined as

$$\delta_e = \sum_{i \in \Omega_e} \bar{\rho}_i \mu_i. \quad (5.21)$$

In order to insert the maximum operator in a gradient-based algorithm, a suitable smooth approximation should be given. The  $\chi$ -norm is used here below, that is defined as

$$\delta_{max} = \left( \sum_{e=1}^{N_e} \delta_e^\chi \right)^{\frac{1}{\chi}}, \quad (5.22)$$

wherein  $\chi$  is a tuning parameter whose value should be high enough. Therefore, the constraint is formulated by combining Eq. (5.18) with Eq. (5.22):

$$\delta_{max} = \left( \sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi \right)^{\frac{1}{\chi}} \leq \Omega_e (1 - \psi). \quad (5.23)$$

Then, Eq. (5.23) is arranged in order to be dimensionless and put in the form of a standard inequality constraint for the problem (4.10) as follows

$$g_{d_{max}} = \frac{\left( \sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi \right)^{\frac{1}{\chi}}}{\Omega_e (1 - \psi)} - 1 \leq 0. \quad (5.24)$$

Since the maximum length scale is imposed through an explicit constraint, its gradient with respect to the optimisation variables (i.e. the NURBS control points and weights) must be computed. Here below, the analytical expression of the constraint derivatives is reported. For more details about the related mathematical passages to get these results, the reader is addressed to Appendix C.

$$\frac{\partial g_{d_{max}}}{\partial \Xi_{\tau}^{(1)}} = \beta(g_{d_{max}} + 1) \frac{\sum_{e=1}^{N_e} \left( \left( \sum_{i \in \Omega_e} \rho_i^{\beta} \mu_i \right)^{\chi-1} \left( \sum_{i \in \Omega_e} \rho_i^{\beta-1} R_{\tau}^i \mu_i \right) \right)}{\sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \bar{\rho}_i \mu_i \right)^{\chi}}, \quad (5.25)$$

$$\begin{aligned} \frac{\partial g_{d_{max}}}{\partial \Xi_{\tau}^{(2)}} &= \frac{\Xi_{\tau}^{(1)}}{\Xi_{\tau}^{(2)}} \frac{\partial g_{d_{max}}}{\partial \Xi_{\tau}^{(1)}} + \\ &= \frac{\beta(g_{d_{max}} + 1)}{\Xi_{\tau}^{(2)}} \frac{\sum_{e=1}^{N_e} \left( \left( \sum_{i \in \Omega_e} \rho_i^{\beta} \mu_i \right)^{\chi-1} \left( \sum_{i \in \Omega_e} \rho_i^{\beta} R_{\tau}^i \mu_i \right) \right)}{\sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \bar{\rho}_i \mu_i \right)^{\chi}}. \end{aligned} \quad (5.26)$$

The maximum length scale constraint in the form of Eq. (5.24) presents several advantages. Firstly, both the intuitive idea and its mathematical formulation are simple. Secondly, the constraint statement is general and holds for 2D and 3D applications. Finally, the  $d_{max}$  constraint can be imposed in particularly hard TO problems as well, which are characterised by a poor convergence rate: this is possible thanks to the “free” parameters  $\psi$  and  $\chi$ . Usually they are set as  $\psi = 0.05$  and  $\chi = 15$  and they do not change during iterations. However, it is possible to start the optimisation with lower values and to increase them by a continuation method in order to improve the convergence quality.

### 5.4.2 Results: Maximum Length Scale in 2D

The effects of the maximum member size constraint in the framework of the NURBS-based SIMP approach are investigated on the 2D benchmark illustrated in Fig. 5.32: a rectangular domain subject to a traction load is considered. All the material and geometrical data are provided in the caption of Fig. 5.32. After a preliminary check on the convergence of the results, the rectangular domain is discretised by means of  $100 \times 50$  shell elements.

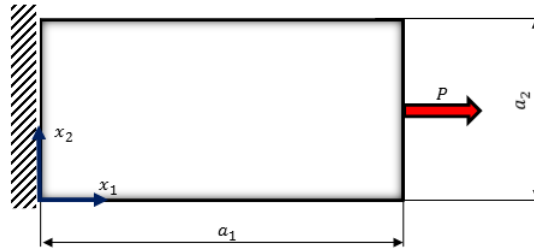


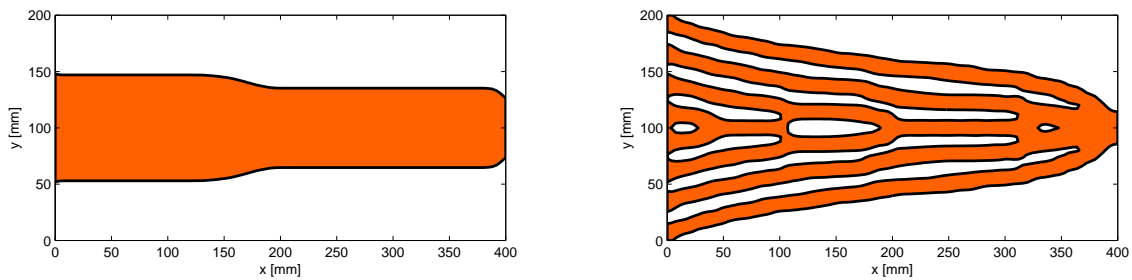
Figure 5.32 – Traction plate problem -  $a_1 = 400$  mm,  $a_2 = 200$  mm, Thickness  $t = 2$  mm, Young’s Modulus  $E = 72000$  MPa, Poisson’s ratio  $\nu = 0.33$ , Load  $P = 1000$  N.

A B-Spline surface ( $p_1 = p_2 = 3$ ,  $(n_1 + 1) \times (n_2 + 1) = 40 \times 20$ ) is chosen to perform the TO analysis. Firstly, the standard TO problem (4.10) is solved for  $\gamma = 0.4$  and

#### 5.4. The Maximum Length Scale

$V_{ref} = a_1 a_2 t$ . Then, the introduction of a maximum member size constraint according to Eq. (5.24) is investigated: in particular, problem (5.1) is now solved by considering the constraint  $g_{d_{max}}$ , with the maximum allowable dimension of topological elements fixed to  $d_{max} = 25$  mm.

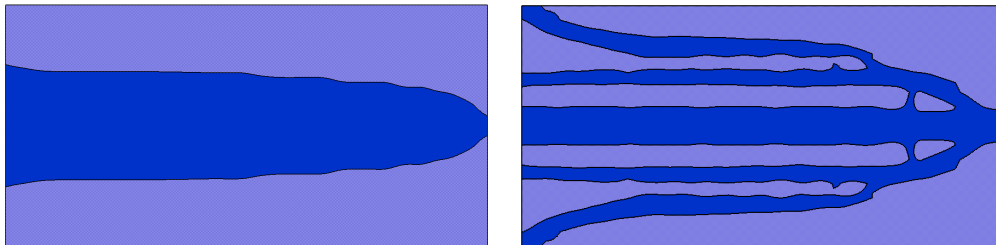
Solutions of the two problems, provided by both the proposed approach and OptiStruct<sup>®</sup> are shown in Figs. 5.33 and 5.34, respectively.



(a)  $c = 55.85$  Nmm,  $V/V_{ref} = 0.4000$ .

(b)  $d_{max} = 25$  mm,  $c = 74.47$  Nmm,  $V/V_{ref} = 0.3984$ .

Figure 5.33 – NURBS-based SIMP solutions for the traction plate problem (a) without and (b) with maximum member size constraint.



(a)  $c = 58.02$  Nmm,  $V/V_{ref} = 0.3938$ .

(b)  $d_{max} = 25$  mm,  $c = 72.95$  Nmm,  $V/V_{ref} = 0.3832$ .

Figure 5.34 – OptiStruct<sup>®</sup> solution of the traction plate problem (a) without and (b) with maximum member size constraint.

Considering the simplest case without constraints shown in Fig. 5.33a and in Fig. 5.34a, it can be stated that the NURBS-based SIMP method provides consistent results with those obtained by means of the commercial software OptiStruct<sup>®</sup> from a numerical point of view: the percentage difference is 3.7%, but the number of design variables for the

NURBS-based SIMP approach is significantly smaller (800) than that characterising the OptiStruct<sup>®</sup> solution (5000). When considering the maximum length scale constraint, the percentage difference among the NURBS-based solution and the OptiStruct<sup>®</sup> solution reduces to 2.1%. However, in the second case, significant topology changes can be observed, see Fig. 5.33b and Fig. 5.34b. Moreover, it should be pointed out that the maximum member size constraint, as well as the Poulsen's minimum member size constraint, has been formulated in *global sense* and not in *local sense*: this means that, even if the constraint of Eq. (5.24) is globally met during the iterations (on the meshed reference domain), it will not be necessarily satisfied locally after the postprocessing phase (i.e. when the geometry is rebuilt in order to be CAD-compatible). In other words, if the size of topological elements is measured on the rebuilt geometry, the maximum member size of 25 mm is not necessarily met in the proximity of the region where the load is applied, see Figs. 5.33b and 5.34b. Nevertheless, this circumstance is more critical as far as concerns the solution provided by OptiStruct<sup>®</sup>. In particular, the OptiStruct<sup>®</sup> solution shows a central branch of approximately 30 mm > 25 mm (see Fig. 5.34b), thus the constraint is violated on a larger portion of the definition domain when compared to the NURBS-based SIMP algorithm solution.

### 5.4.3 Results: Maximum Length Scale in 3D

The effectiveness of the maximum length scale constraint in 3D is proven in this section through the benchmark of Fig. 5.35. The domain is parametrised with a B-Spline hyper-surface, whose parameters are set as  $p_1 = p_2 = p_3 = 2$  and  $(n_1 + 1) \times (n_2 + 1) \times (n_3 + 1) = 32 \times 8 \times 12$ .

The classic problem of compliance minimisation with an imposed volume fraction is firstly solved according to Eq. (4.10), wherein  $c_{ref} = 241.42$  Nmm,  $V_{ref} = 10^7$  mm<sup>3</sup> and  $\gamma = 0.4$ , without considering the maximum member size constraint. The result of this optimisation is shown in Fig. 5.36. The dimensionless compliance of the final structure is  $c/c_{ref} = 0.1422$ .

Then, problem (4.10) is enhanced with the maximum length scale constraint in the form of Eq. (5.24), wherein the maximum member size is set  $d_{max} = 30$  mm. The solution of problem (4.10) with the maximum length scale constraint is shown in Fig. 5.37. Due to the active constraint on the maximum length scale, the compliance fraction is  $c/c_{ref} = 0.2994$ . The constraint value is  $g_{d_{max}} = 1.601 \times 10^{-6}$ : thus, the maximum length scale constraint is practically met on all the design domain.

It is interesting to remark that the proposed formulation of the maximum length scale properly works also with a relatively coarse mesh.

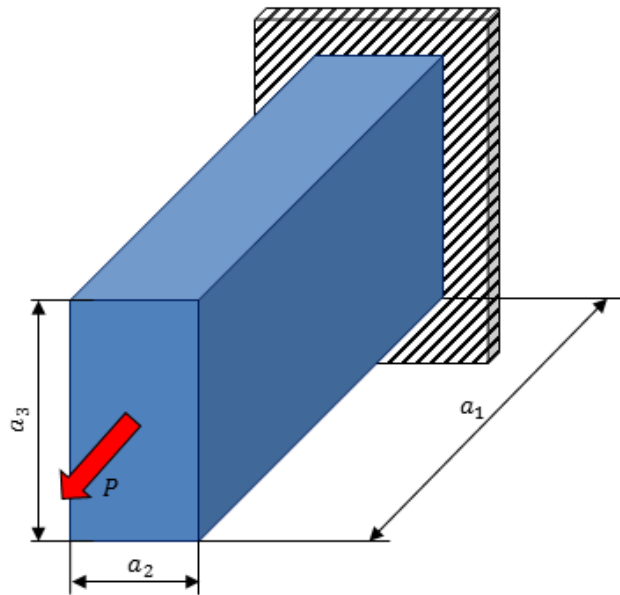


Figure 5.35 – 3D benchmark - Geometric parameters:  $a_1 = 500$  mm,  $a_2 = 100$  mm,  $a_3 = 200$  mm - Material parameters:  $E = 72000$  MPa,  $\nu = 0.33$  - Mesh:  $60 \times 16 \times 24$  SOLID185 elements - Load:  $P = 5000$  N.

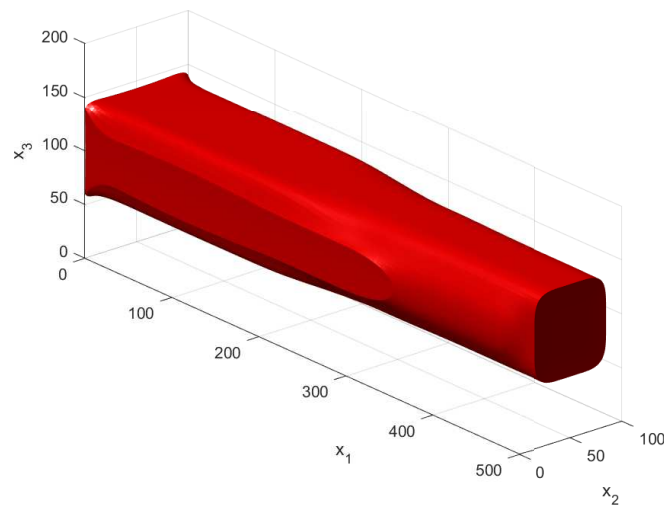


Figure 5.36 – 3D TO result - Compliance minimisation with equality constraint on the volume.

## 5.5 The Minimum Curvature Radius

### 5.5.1 Mathematical Statement of the Minimum Curvature Radius constraint

This kind of constraint has an interest for functional and manufacturing requirements. Ideally, if the boundary of the structure is mathematically defined, the local radius of

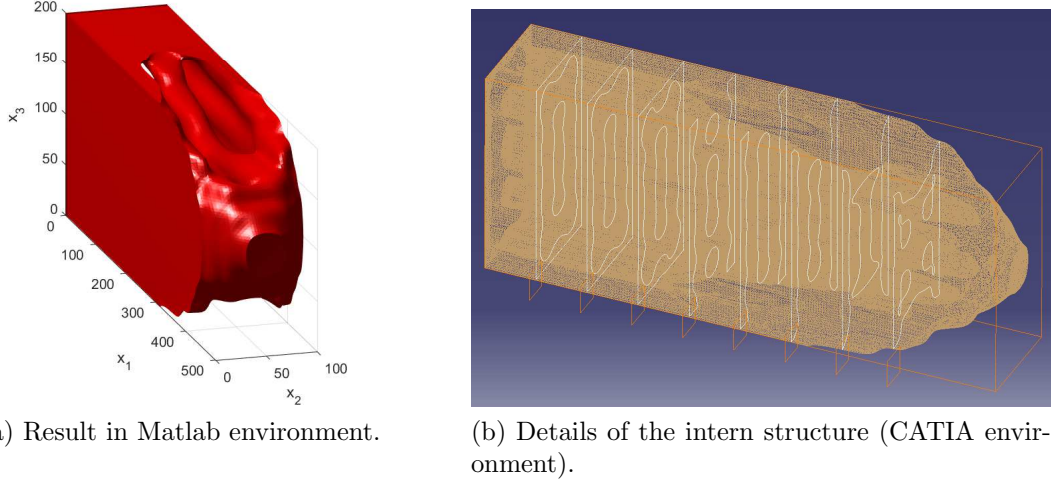


Figure 5.37 – 3D TO result - Compliance minimisation with equality volume constraint and maximum member size constraint.

curvature can be evaluated and its minimum value can be identified. Then, the minimum curvature radius can be constrained to be superior to an admissible reference value. In the framework of classical SIMP approach, it is not possible to formulate this kind of constraints, since the boundary of the structure is not defined (nor in implicit neither in explicit way). Conversely, in the context of the NURBS formulation, a description of the boundary is available at each iteration by establishing a cutting plane for the NURBS surface representing the fictitious density function. The discussion is restrained to 2D TO problems in the following.

Let  $\Omega \subseteq D$  be the material domain and  $\rho_{cut} \in [\hat{\rho}_{min}, \hat{\rho}_{max}]$  the cutting value for the density field. In order to have a precise description of the contour, it can be assumed that

$$\begin{cases} (x_1, x_2) \in \Omega, & \text{if } \rho(x_1, x_2) > \rho_{cut}, \\ (x_1, x_2) \in \partial\Omega, & \text{if } \rho(x_1, x_2) = \rho_{cut}, \\ (x_1, x_2) \in D \setminus \Omega, & \text{if } \rho(x_1, x_2) < \rho_{cut}. \end{cases} \quad (5.27)$$

For an implicit 2D curve, the expression of the curvature reads [123]

$$\kappa = - \frac{\left\{ \frac{\partial \rho}{\partial x_2} \quad - \frac{\partial \rho}{\partial x_1} \right\} \begin{bmatrix} \frac{\partial^2 \rho}{\partial x_1^2} & \frac{\partial^2 \rho}{\partial x_1 \partial x_2} \\ \frac{\partial^2 \rho}{\partial x_1 \partial x_2} & \frac{\partial^2 \rho}{\partial x_2^2} \end{bmatrix} \begin{Bmatrix} \frac{\partial \rho}{\partial x_2} \\ - \frac{\partial \rho}{\partial x_1} \end{Bmatrix}}{\left( \left( \frac{\partial \rho}{\partial x_1} \right)^2 + \left( \frac{\partial \rho}{\partial x_2} \right)^2 \right)^{\frac{3}{2}}}. \quad (5.28)$$

In Eq. (5.28), the dependence of  $\rho$  on the spatial coordinates  $(x_1, x_2)$  is omitted for sake of compactness. Using the relations of Eq. (4.3) and by rearranging Eq. (5.28), the curvature radius can be achieved:

$$r = -\frac{1}{a_1 a_2} \frac{\left( a_2^2 \left( \frac{\partial \rho}{\partial u_1} \right)^2 + a_1^2 \left( \frac{\partial \rho}{\partial u_2} \right)^2 \right)^{\frac{3}{2}}}{\left( \frac{\partial \rho}{\partial u_1} \right)^2 \frac{\partial^2 \rho}{\partial u_2^2} - 2 \frac{\partial \rho}{\partial u_1} \frac{\partial \rho}{\partial u_2} \frac{\partial^2 \rho}{\partial u_1 \partial u_2} + \left( \frac{\partial \rho}{\partial u_2} \right)^2 \frac{\partial^2 \rho}{\partial u_1^2}}. \quad (5.29)$$

Hence, the constraint can be formulated as

$$\min_{\partial \Omega} |r(x_1, x_2)| \geq \bar{r}. \quad (5.30)$$

The absolute value is approximated by means of Eq. (5.6), whilst the minimum operator has been estimated through the Kreisselmeier-Steinhauser function [124]. Let  $N_r$  be the number of radius evaluations on the contour of the structure. Hence, Eq. (5.30) changes into the following relation:

$$g_r = 1 + \frac{1}{\bar{r} \Upsilon} \ln \left( \sum_{k=1}^{N_r} \exp \left( -\Upsilon (\sqrt{r_k^2 + \epsilon^2} - \epsilon) \right) \right) \leq 0, \quad (5.31)$$

where  $\Upsilon$  should be big enough.

The derivatives of the constraint on the local curvature radius with respect to design variables (control points and weights) can be expressed as follows:

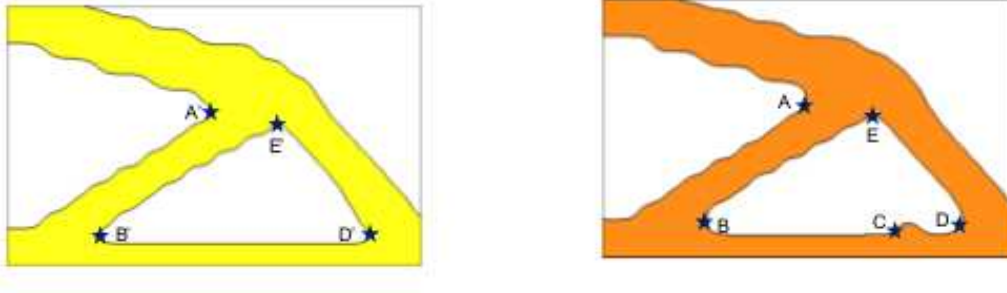
$$\frac{\partial g_r}{\partial \Xi_\tau^{(l)}} = -\frac{1}{\bar{r}} \frac{\sum_{k=1}^{N_r} \frac{\exp \left( -\Upsilon (\sqrt{r_k^2 + \epsilon^2} - \epsilon) \right) r_k \frac{\partial r_k}{\partial \Xi_\tau^{(l)}}}{\sqrt{r_k^2 + \epsilon^2}}}{\sum_{k=1}^{N_r} \exp \left( -\Upsilon (\sqrt{r_k^2 + \epsilon^2} - \epsilon) \right)}, \quad l = 1, 2, \quad (5.32)$$

where the gradient of the generic curvature radius can be evaluated thanks to Eq. (5.29). Details are provided in Appendix D.

### 5.5.2 Results on the application of the Minimum Curvature Radius constraint

The constraint on the local curvature radius is tested here. An enhanced version of problem (4.23) is solved by taking into account the constraint of Eq. (5.31). In particular, a minimum curvature radius  $\bar{r} = 7.5$  mm is imposed. In this section, the solution of Fig. 4.10b is considered a reference solution (the constraint on the local radius of curvature is





(a)  $c = 408.37$  Nmm,  $V/V_{tot} = 0.3994$ .

(b)  $c = 412.00$  Nmm,  $V/V_{tot} = 0.3997$ ,  $r_{min} = 10$  mm.

Figure 5.38 – Solution of the cantilever plate problem (a) without and (b) with the minimum local curvature radius constraint

not imposed here). For sake of completeness, it is recalled that the considered topology has been obtained through a B-Spline surface ( $p_1 = p_2 = 3$ ,  $(n_1 + 1) \times (n_2 + 1) = 32 \times 20$ ). Critical curvature points are highlighted in Fig. 5.38a and the respective curvature radii are:  $r_{A'} = 4.4$  mm,  $r_{B'} = 2.7$  mm,  $r_{D'} = 4.4$  mm,  $r_{E'} = 4.7$  mm. The solution of the same problem enhanced with the minimum curvature radius constraint is illustrated in Fig. 5.38b. The critical curvature radii are:  $r_A = 9.3$  mm,  $r_B = 8.9$  mm,  $r_C = 8.7$  mm,  $r_D = 8.9$  mm,  $r_E = 7.8$  mm.

Even if restrained to 2D problems, this last example, allows for understanding the true potential hidden behind the NURBS-based SIMP approach. The NURBS formulation permits to have a precise and well-defined geometric description of the boundary of the topology at each iteration during the solution process. Thus, local quantities (like the curvature radius) can be easily computed by means of the NURBS formalism. In this last case, a comparison with the results provided by OptiStruct<sup>®</sup> is no longer possible simply because this feature cannot be realised in the framework of the classical SIMP approach.

## 5.6 Conclusions and Perspectives on Geometrical Constraints in the NURBS-based TO algorithm

In this Chapter, it has been shown that geometric constraints can be effectively handled into the NURBS-based SIMP approach.

As a first attempt, the minimum length scale has been considered by reformulating in the NURBS framework the well-known Poulsen's monotonicity constraint. Results show topologies that are consistent with the imposed minimum length scale but some

important drawbacks prevent a widespread use of this method in TO. Beyond the already discussed dependence on a regular (mapped) mesh and the jagged boundary, Poulsen's method could also become cumbersome when dealing with its extension to the 3D case (wherein 13 directions are checked instead of 4).

Therefore, the most relevant contribution of this Chapter is presenting an innovative method for imposing the minimum length scale, which is capable of overcoming the difficulties related to Poulsen's formulation. Thanks to the NURBS entity geometrical features, it is possible to forecast the minimum length scale by simply choosing the NURBS discrete parameters (number of control points and degrees) and some of their continuous parameters, like knot vectors components, starting from very general design *abaci*. No further optimisation constraints need to be added to the problem formulation. Three main consequences immediately follow: firstly, the proposed method for the minimum length scale is completely geometry-based and does not depend upon the underlying mesh. In this sense, the definition of the minimum length scale becomes totally independent from the mesh size. Secondly, the minimum length scale can be controlled not only on the FE model of the structure but also on the reassembled geometry at the end of the optimisation process. Thirdly, it has been shown that the designer can decide to set knot vector according an *ad hoc* criterion and, consequently, to perform TO with a different minimum length scale in different regions of the domain. All these aspects can be controlled by the designer by properly tuning the NURBS parameters.

Conversely, the maximum length scale needs to be controlled through a suitable constraint: the Guest's formulation has been revisited in the framework of the NURBS-based SIMP method. Even if the maximum length scale is not completely free from the FE mesh, its effectiveness has been shown in the most general context of 4D NURBS hyper-surfaces (3D problems) as well as for 2D structures. The most important shortcoming of the proposed formulation of the maximum length scale constraint is that it is a global constraint, i.e. there is no guarantee that the constraint is locally met on all the material phase after the postprocessing. However, it has been shown that those zones wherein the maximum member size constraint is not met are actually circumscribed to small portions of the design domain.

Controlling minimum and maximum length scale is of paramount importance and their interdependence should be remarked. Before choosing a value for the  $d_{max}$  when launching the TO analysis, the implicitly defined length scale deriving from the NURBS formalism must be taken into account. Using the NURBS parameters and the presented technique for forecasting the minimum length scale, it can be verified which is the value of the expected minimum member size. This checking task is necessary because, if neglected, inconsistent constraints could be imposed: for instance, if the imposed maximum member size is too close to the minimum one, convergence could be seriously prevented because the design variables domain is indeed too narrow and there are not enough variables for the TO algorithm to provide a feasible, meaningful design.

Furthermore, a new geometric constraint on the local curvature radius has been implemented. In the framework of the NURBS formalism, this is a relatively straightforward task because a well-defined geometric description of the boundary of the current topology is always available during the iterations. Since the topology is described *via* FE in the standard SIMP algorithm, local information such as tangent and normal to the boundary vectors are not defined and this kind of constraints cannot be implemented. To be highlighted that the minimum curvature radius constraint should be conscientiously used: it is well known that more efficient solutions in TO are provided when the number of design variables is increased, promoting a higher number of thinner topological elements rather than a lower number of thick branches. However, the former solution is prone to small curvature radii. If a too big curvature radius is demanded, than the TO algorithm will provide a solution that is really far from the optimum configuration that would have been obtained without the curvature radius constraint.

Perspectives of the work presented in this Chapter are manifold:

- a) As far as the minimum length scale requirement is concerned, the derived *abaci* can be used, in a conservative sense, in the framework of NURBS entities but their utilisation is rigorously justified only in the case of B-Spline entities. Indeed, for NURBS entities, the current formulation of the minimum length scale could be considered valuable only from a practical viewpoint, as it provides a sort of rule of thumb for forecasting the minimum member size in TO problems. In this background, closed-form solution for Greville's abscissae for NURBS entities must be derived and integrated within the algorithm proposed in this work in order to obtain pertinent *abaci*.
- b) This Chapter has shown some promising consequences of varying the knot vector components in the NURBS-based TO method. Namely, *ad hoc* knot vector distributions could be proposed to accomplish extremely specific tasks. These aspects go beyond the minimum length scale requirement. Indeed, the knot vectors components could be integrated among the optimisation variables in order to improve the quality of results and to meet the constraint on the minimum length scale on the overall computation domain. Furthermore, it is expected that an optimised knot vector distribution could fix the "wave effect" sometime exhibited by B-Spline solutions (see Chapter 4). Research is ongoing on this topic.
- c) Imposing a minimum and a maximum length scale allows for issuing topologies with branches of uniform size. This aspect could be particularly advantageous in ALM production in order to avoid residual stresses in the final structure: in fact, a relevant difference in structural members size implies a difference in the amount of exposed surface (thinner elements will chill faster than massive parts). Thus, a non-uniform heat exchange occurs. The consequent temperature gradient will constitute one of the most important causes of residual stresses (see Chapter 1).

## 5.6. Conclusions and Perspectives on Geometrical Constraints in the NURBS-based TO algorithm

---

Relating geometric information and thermo-mechanical behaviour could provide a deeper insight in bridging TO and ALM.

- d) The adopted method to impose the minimum length scale has an important beneficial side effect: it is capable to give some of the NURBS parameters a precise physical/geometric meaning. Therefore, those parameters, which normally have just a mathematical meaning and whose impact on the final topology is not immediate to the designer, gain a precise role and the designer is guided in their choice. To this purpose, implementing further geometric/mechanical constraints into the NURBS-based SIMP algorithm is of paramount importance. One possibility could be formulating a completely mesh-free maximum length scale constraint and to relate this quantity only to the NURBS parameters, similarly to the minimum member size.



# Chapitre 5

## Implémentation de Contraintes Géométriques dans l'Algorithme d'OT basé sur les NURBS

### 5.1 Introduction aux Contraintes Géométriques en OT

Dans ce chapitre, un enrichissement du problème (4.10) est considérée l'éq. (5.1), où le vecteur de contraintes  $\mathbf{g}$  a été introduit. Ses composantes seront spécifiées tout au long de la discussion, si nécessaire.

L'un des besoins fondamentaux des ingénieurs est l'intégration des exigences de fabrication dans la formulation des problèmes d'OT afin d'obtenir des solutions optimisées ainsi que des solutions fabricables [42]. Les contraintes de fabrication dépendent fortement, en général, de la technologie choisie : néanmoins, les épaisseurs minimale et maximale admissibles des éléments constitutifs de la structure constituent un aspect fondamental, quel que soit le procédé considéré.

Dans les méthodes basées sur la densité pour prendre en compte l'épaisseur minimale [5, 112], on peut remarquer un manque de cohérence important entre l'épaisseur minimale imposée et celle réellement mesurée à la fin du processus d'optimisation. La méthode de Poulsen est basée sur la monotonie de la pseudo-densité selon  $n_d$  directions préférentielles en fonction de la dimension du problème :  $n_d = 4$  ou  $n_d = 13$  respectivement pour les problèmes 2D et 3D. Bien que cette méthode soit robuste, elle fournit des solutions irrégulières aux frontières et, de plus, son efficacité est fortement limitée aux maillages réguliers. Les méthodes de projection [10, 42, 53] ne sont pas non plus efficaces du point de vue du calcul et garantissent un contrôle strict de l'épaisseur minimale des éléments topologiques uniquement sur le modèle EF et non pas sur la géométrie réelle. D'autres stratégies ont été développées dans le cadre de la méthode LSM (voir le chapitre 1) appliquée aux problèmes structurels d'OT [113, 114, 116]. Les inconvénients liés à cette technique sont discutés dans [45, 115], où il est montré qu'une définition mathématiquement exacte de l'épaisseur minimale des éléments topologiques n'existe pas.

Une méthode alternative pour inclure la contrainte d'épaisseur minimale dans un algorithme d'OT basé sur les B-Spline a été suggérée dans [117] : la stratégie proposée est basée sur la combinaison de l'approche présentée dans [104] et des méthodes de projection en trois phases [42]. Cependant, d'un point de vue théorique, il semble incohérent de choisir une méthode nécessitant trois analyses EF afin de contrôler une caractéristique purement géométrique.

En ce qui concerne l'épaisseur maximale, comme noté dans [118–120], l'imposition d'une épaisseur maximale en OT donne lieu à des configurations optimisées finales répondant aux exigences techniques en matière de sécurité. Une des premières approches pour établir un contrôle de l'épaisseur maximale dans l'OT a été suggérée par Guest dans [43], où des contraintes explicites sont imposées au voisinage de chaque élément du maillage en utilisant des méthodes de projection.

Comme discuté ci-dessus, les exigences d'épaisseur minimale et maximale ne sont pas contrôlées de manière satisfaisante. Pour surmonter les restrictions liées aux techniques précédentes, l'implémentation du contrôle de l'épaisseur minimale et maximale dans l'approche SIMP basée sur les NURBS est abordée dans ce chapitre. De plus, afin de montrer les capacités de l'algorithme, une contrainte géométrique innovante (le rayon de courbure) a été implémentée grâce au formalisme NURBS. Il est axé sur la valeur du rayon de courbure local minimal de la frontière de la structure en cours d'optimisation.

Ce chapitre est décrit comme suit. Tout d'abord, la contrainte de l'épaisseur minimale de Poulsen est reformulée dans le cadre de l'algorithme SIMP basé sur les NURBS et les résultats sont discutés pour des problèmes 2D. Deuxièmement, une méthode purement géométrique basée sur les paramètres NURBS est présentée afin de contrôler l'épaisseur minimale sans formuler des contraintes d'optimisation explicites. Des détails sur l'implémentation de la contrainte d'épaisseur maximale sont ensuite fournis. Les résultats relatifs au contrôle de l'épaisseur minimale et maximale sont donnés pour les applications 2D et 3D. Enfin, la contrainte du rayon de courbure minimale est mathématiquement définie et ses effets sont étudiés sur un simple benchmark 2D.

Le contenu de ce chapitre fait référence à une partie de [107] et à [122].

## 5.2 La formulation de Poulsen pour la contrainte de l'épaisseur minimale

### 5.2.1 L'énoncé mathématique de la contrainte de l'épaisseur minimale selon Poulsen

L'idée intuitive de la méthode de Poulsen pour la contrainte d'épaisseur minimale [35] en 2D consiste à imposer la monotonie de la fonction de densité fictive dans une zone circulaire ayant un diamètre égal à l'épaisseur minimale ( $d_{min}$ ). La zone circulaire est

esquissée autour de chaque élément du maillage et la monotonie est vérifiée le long de chacune des quatre directions ( $0^\circ$ ,  $90^\circ$ ,  $\pm 45^\circ$ ). Mathématiquement parlant, la monotonie d'une fonction sur un intervalle  $I$  sur une direction  $\mathbf{b}$  peut être vérifiée au moyen de l'intégrale de l'éq. (5.2).  $M_{\mathbf{b}}(f)$  est strictement égal à 0 si  $f$  est monotone et supérieur à 0 autrement. Par conséquent, la contrainte sur l'épaisseur minimale est formulée comme indiqué dans l'éq. (5.3), où  $N_e$  est le nombre d'éléments du maillage,  $\mathbf{b}_i$  la direction de vérification ( $i = 1, \dots, 4$ ),  $\theta$  un exposant de pénalisation et  $\sigma$  est utilisé pour relâcher la contrainte et assurer la stabilité numérique. L'expression explicite de  $M_{\mathbf{b}_i}(\rho)$  est donnée par l'éq. (5.4), où  $s$  est une abscisse définie sur la direction de vérification  $\mathbf{b}_i$ . En particulier,  $\mathbf{b}_1 = [1, 0]^t$ ,  $\mathbf{b}_2 = [0, 1]^t$ ,  $\mathbf{b}_3 = [\sqrt{2}/2, \sqrt{2}/2]^t$  et  $\mathbf{b}_4 = [\sqrt{2}/2, -\sqrt{2}/2]^t$ .

Afin de formuler une version discrète de l'éq. (5.4), considérons un maillage régulier d'éléments carrés. Alors,  $N_{\mathbf{b}_i}$  est le nombre d'éléments du maillage couvrant le diamètre  $d_{min}$  le long de la direction  $\mathbf{b}_i$ . Il est simple à vérifier (voir [35]) que l'éq. (5.4) se transforme en l'éq. (5.5), où  $j$  est juste un index muet qui balaye les éléments du maillage sur l'intervalle  $[0, d_{min}]$ , comme l'abscisse  $s$  dans l'éq. (5.4). De plus, l'approximation donnée par l'éq. (5.6) pour la fonction absolue a été employée pour éviter la discontinuité de la dérivée. L'expression finale de  $M_{\mathbf{b}_i}(\rho)$  à implémenter est fournie par l'éq. (5.7).

Dans les eqs. (5.5) et (5.7)  $\rho_1$  et  $\rho_{N_{\mathbf{b}_i}}$  sont les pseudo-densités liées au premier élément et au dernier, respectivement dans la direction  $\mathbf{b}_i$ . En ce qui concerne l'analyse de sensibilité, la même notation introduite au chapitre 4 est utilisée. En conséquence, les dérivées de la contrainte d'épaisseur minimale par rapport aux points de contrôle et aux poids de la NURBS sont calculées comme dans les eqs. (5.8) et (5.9).

Les termes  $\frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_\tau^{(1)}}$  et  $\frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_\tau^{(2)}}$  peuvent être évalués comme montré dans l'Annexe B.

## 5.2.2 La contrainte de l'épaisseur minimale selon Poulsen : résultats

La version revisitée de la contrainte d'épaisseur minimale de Poulsen dans le cadre NURBS est testée sur le benchmark illustré à la fig. 4.7. Ainsi, la contrainte de l'éq. (5.3) est ajoutée au problème (4.23). En particulier, une surface B-Spline avec  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$  control points et  $p_1 = p_2 = 3$  est utilisée pour trouver la solution. La contrainte sur l'épaisseur minimale est imposée en considérant trois valeurs de  $d_{min}$  : 16, 20 et 25 mm. Les résultats sont rassemblés dans les figs. 5.1-5.3 pour chaque analyse. Dans chaque figure, la première image est la géométrie reconstruite fournie par la méthode SIMP basée sur les NURBS après la phase de post-traitement, tandis que la deuxième image est la géométrie reconstruite finale fournie par l'outil commercial OptiStruct®.

Deux remarques très importantes peuvent être déduites

- Grâce à l'action combinée de la propriété du support local et de la contrainte d'épaisseur minimale selon Poulsen, toutes les zones « grises » sans signification sont filtrées et la topologie finale répond exactement à la contrainte imposée. Inversement,



les topologies optimales résultant d’OptiStruct® ne répondent pas systématiquement à la contrainte.

- Les solutions optimales fournies par l’approche SIMP basée sur les NURBS font apparaître des frontières non lisses. En effet cet aspect est directement lié à la formulation de la contrainte d’épaisseur minimale selon la formule de Poulsen, voir éq. (5.3).

## 5.3 Contrôle de l’épaisseur minimale dans l’algorithme d’OT basé sur les NURBS

### 5.3.1 L’épaisseur minimale résultant des entités B-Spline

Une stratégie purement géométrique est proposée ici afin de satisfaire aux exigences d’épaisseur minimale sur la topologie assemblée finale, à la fois pour les structures 2D et 3D. Bien entendu, les paramètres NURBS listés dans les chapitres 2 et 4 sont censés avoir un impact fort sur cette entité topologique. Par souci de clarté, seules les entités B-Spline sont considérées dans un premier temps. Dans ce contexte, certaines exigences particulières doivent être remplies.

*Condition 1* : les coordonnées spatiales des points de contrôle définissant les surfaces et les hyper-surfaces B-Spline sont distribuées selon les abscisses de Greville, fournies par l’éq. (4.8).

*Condition 2* : la condition de l’épaisseur minimale doit être simulée. En ce qui concerne les problèmes 2D, telle condition est reproduite en attribuant la valeur  $\hat{\rho}_{min} = 10^{-3}$  à chaque point de contrôle  $\hat{\rho}_{i_1, i_2}$ , à l’exception d’une colonne ou d’une ligne de points de contrôle, qui ont pour valeur  $\hat{\rho}_{max} = 1$ , comme illustré aux figs. 5.4a et 5.4c. Le résultat de l’intersection entre la surface B-Spline et un plan approprié (représentant la valeur seuil de pseudo-densité) est une bande de phase de matériau (figs. 5.4b et 5.4d). Bien entendu, l’épaisseur de cette bande représente l’épaisseur minimale. Comme les coordonnées des points de contrôle sont organisées en matrices 3D pour les hyper-surfaces B-Spline, une procédure similaire peut être répétée en 3D en définissant  $\hat{\rho}_{min}$  pour toutes les coordonnées  $\hat{\rho}_{i_1, i_2, i_3}$ , à l’exception des points appartenant à une « page » appropriée de la matrice 3D (voir fig. 5.5).

*Condition 3* : la densité de seuil peut être définie comme  $\rho_{th} \in [\rho_{th-LB}, \rho_{th-UB}]$ , avec  $\rho_{th-LB} = 0.35$  et  $\rho_{th-UB} = 0.6$ .

*Condition 4* : la méthode SIMP basée sur les NURBS a été décrite dans le chapitre 4 dans sa forme la plus générale. En particulier, la même méthodologie peut être appliquée quels que soient les composantes des *knot vectors*. Dans le contexte de la contrainte d’épaisseur minimale, les composantes des *knot vectors* ont été choisies uniformément dans l’intervalle  $[0, 1]$ , comme indiqué dans éq. (5.12). Ainsi, la distance entre deux com-

posantes non triviales du *knot vector*, à savoir  $\Delta U^{(j)}$ , est donnée par éq. (5.13). De plus, l'éq. (5.14) met en relation les coordonnées des points de contrôle et les composantes des *knot vectors*.

Compte tenu des aspects précédents, une analyse de sensibilité de l'épaisseur minimale, appelée  $d_{min}$ , est présentée ci-après. En particulier, l'influence du nombre de points de contrôle sur  $d_{min}$  est étudiée. Pour cette analyse préliminaire, les degrés des fonctions de base, à savoir  $p_j$ ,  $j = 1, 2, 3$ , sont maintenus constants et les composantes des *knot vectors* sont équidistantes dans la plage  $[0, 1]$  selon l'éq. (5.12). L'allure de  $d_{min}^j$  par rapport à  $\Delta X^{(j)}$  est illustrée dans les figs. 5.6-5.10 ( $p_j = 3$ ) pour les problèmes 2D et 3D. Les étapes logiques pour construire ces graphiques, qui dépendent aussi des degrés de l'entité NURBS, sont rapportées dans l'Algorithme 1.

Les courbes  $d_{min}$  vs.  $\Delta X$  peuvent être utilisées comme outil de conception pour prévoir l'épaisseur minimale pour différentes combinaisons de  $n_j$  et  $p_j$ . De plus, comme les *knot vectors* sont uniformes, les graphiques précédents peuvent être convertis en courbes  $d_{min}$  vs.  $n/a$ , i.e. l'épaisseur minimale en fonction de la densité des points de contrôle, comme montré aux figs. 5.11-5.13.

La procédure de détermination des abaques pour prendre en compte l'épaisseur longueur minimale doit être légèrement modifiée pour les régions proches de la frontière du domaine de calcul, où l'éq. (5.14) devrait être remplacé par l'éq. (5.16). Dans l'éq. (5.16),  $\Delta X_0^{(j)}$  indique la valeur de  $\Delta X^{(j)}$  à la frontière. Les abaques liés à l'épaisseur minimale près de la frontière du domaine de calcul, qui est noté  $d_{min}^B$ , sont illustrés aux figs. 5.14-5.16.

### 5.3.2 Quelques remarques sur l'approche proposée

L'approche utilisant les précédents abaques est simple et intuitive pour le concepteur : il n'y a pas besoin d'introduire une contrainte supplémentaire dans la formulation du problème d'OT. Ce point est d'une importance primordiale car les contraintes impliquant l'épaisseur minimale peuvent devenir lourdes d'un point de vue informatique. De plus, les courbes pour la conception fournies visent à prévoir  $d_{min}$  sur la géométrie réassemblée, afin d'obtenir des résultats cohérents. En plus, il est possible de choisir un nombre approprié de points de contrôle en définissant l'épaisseur minimale souhaitée via les abaques des figs. 5.8-5.10.

Concernant les paramètres libres de la méthode, la plage des possibles degrés des fonctions à utiliser pour l'analyse d'OT est limitée car il n'y a pas d'intérêt à utiliser des degrés élevés et le concepteur ne pourra qu'essayer les valeurs  $p_j = 2, 3, 4$ . Le premier choix pour la valeur de densité de seuil doit être de 0,5 et 0,45 pour les applications 2D et 3D, respectivement. Si le concepteur est incertain parmi deux valeurs pour  $\rho_{th}$ , la valeur la plus élevée doit être définie afin de faire le choix le plus prudent.

### 5.3.3 Les effets des poids des NURBS sur l'épaisseur minimale

L'épaisseur minimale des éléments devrait également être prévue dans le cas des NURBS, où la formule des abscisses de Greville sous la forme de l'éq. (4.8) n'est plus valable. Il n'est donc pas possible de calculer facilement les coordonnées physiques  $x_j$ ,  $j = 1, 2, 3$  des points de contrôle sur le domaine de conception.

Pour avoir une idée du comportement des poids des NURBS lorsque la condition d'épaisseur minimale est rencontrée, la discussion se concentre sur la solution NURBS de la fig. 4.11c. Supposons de figer l'index des colonnes des points de contrôle et d'observer les valeurs des points de contrôle et des poids sont observées selon une colonne. Cela équivaudra à regarder les valeurs des quantités susmentionnées sur une ligne horizontale dessinée sur le domaine de conception. Certains des graphiques les plus représentatifs sont présentés à la fig. 5.18.

De manière empirique, il a été observé que la condition la plus critique (c'est-à-dire la valeur la plus faible d'épaisseur des éléments topologiques) est obtenue avec les surfaces B-Spline et non avec les surfaces NURBS. La conclusion naturelle de cette étude est que les courbes de conception de la section 5.3.1 peuvent également être utilisés pour les entités B-Spline et NURBS, car elles constitueront de toute façon une estimation prudente de l'épaisseur minimale. Bien entendu, l'épaisseur minimale ne peut pas être prévue de manière rigoureuse lorsque des entités NURBS sont utilisées. L'utilisation des abaques, qui est robuste pour les entités B-Spline, ne peut être que partiellement justifiée pour les entités NURBS.

### 5.3.4 Résultats : l'épaisseur minimale en 2D

L'efficacité des abaques utilisés pour prévoir l'épaisseur minimale est prouvée dans ce paragraphe pour les problèmes 2D par le simple benchmark de la fig. 5.19 (se référer à la légende associée pour plus de détails sur la géométrie et l'analyse EF) .

Le problème d'OT est formulé selon l'éq. (4.10), où  $V_{ref} = a_1 a_2 t$  et  $\gamma = 0.4$ . De plus, une condition de symétrie est imposée par rapport au plan  $x_2 = a_2/2$ . Lorsque la convergence est atteinte, le résultat est exporté au format IGS et la solution 2D est récupérée au moyen des opérations de seuil décrites au chapitre 4 (fig. 5.20a). Une fois la structure 2D obtenue, l'épaisseur minimale réelle des éléments structurels est identifiée et mesurée. Il a été établi de manière conventionnelle que l'épaisseur minimale mesurée correspond au diamètre du plus petit cercle pouvant être inscrit dans la structure (voir la fig. 5.20b).

La méthode présentée dans la section 5.3.1 est testée en comparant l'épaisseur minimale prédite par les graphiques des figs. 5.11-5.13 et figs. 5.14-5.16 avec l'épaisseur minimale mesurée sur la structure réassemblée. En particulier, le benchmark a été résolu en utilisant à la fois les surfaces B-Spline et NURBS (degrés  $p_j = 3$ ,  $j = 1, 2, 3$ ) et l'analyse d'OT a été répétée plusieurs fois pour changer le nombre de points de contrôle.

Les solutions sont présentées dans la fig. 5.21 et les résultats numériques sont rassemblés dans les Tableaux 5.1 et 5.2, étant la compliance divisée par une quantité de référence  $c_{ref} = 2625$  Nmm, soit la compliance de la structure avec une densité initiale uniforme  $\rho(u_1, u_2) = \gamma$ . L'épaisseur minimale mesurée est appelée  $d_{min}^M$ . Les deux valeurs  $d_{min}$  et  $d_{min}^B$  sont évaluées pour une valeur seuil estimée de la densité  $\rho_{th} = 0,5$ . Dans les Tableaux 5.1 et 5.2, le symbole ( $B$ ) apparaît à côté des valeurs de  $d_{min}^M$  lorsque l'épaisseur minimale est mesurée dans le voisinage de la frontière ; sinon, la zone critique dans laquelle  $d_{min}^M$  est mesurée se trouve dans le domaine de conception.

Comme il peut être observé à partir des Tableaux 5.1 et 5.2, l'épaisseur minimale est correctement prévue pour les solutions B-Spline et NURBS : cette-ci est toujours inférieure à l'épaisseur minimale réelle mesurée sur la géométrie réassemblée en CAO. En ce sens, la méthode est conservatrice.

### 5.3.5 Résultats : l'épaisseur minimale en 3D

En raison de la complexité des topologies 3D, la condition d'épaisseur minimale est vérifiée de manière différente par rapport au 2D. Le problème de l'éq. (4.10) est résolu pour le benchmark de la fig. 5.22. Deux conditions de symétrie, par rapport aux plans  $x_2 = a_2/2$  et  $x_3 = a_3/2$ , sont ajoutées. Des solutions sont fournies ci-dessous pour différents nombres de points de contrôle et en définissant les degrés d'hyper-surfaces B-Spline / NURBS égaux à 3.

Pour les problèmes 3D, les configurations optimisées peuvent être rassemblées, à la fin du processus d'optimisation, sous la forme de fichiers STL. Les graphiques de la section 5.3.1 peuvent être utilisés afin de prévoir l'épaisseur minimale dans le domaine et sur ses limites. Le fichier STL collecte les triangles  $n_{TR}$  qui composent la frontière du volume optimisé. Ainsi, le vecteur normal sortant  $\mathbf{n}$  peut être identifié sur chaque surface limite. Ces informations peuvent être exploitées pour mesurer l'épaisseur minimale réelle à la fin de l'optimisation, afin de vérifier l'efficacité et la robustesse de l'approche basée sur les abaques. Les principales étapes pour mesurer l'épaisseur minimale en 3D sont décrites dans l'Algorithme 2.

Les résultats numériques sont collectés dans les Tableaux 5.3 et 5.4. Les fractions  $f_F$  et  $f_F^B$  des triangles « faisables » sont également indiquées.

Les topologies correspondant aux configurations apparaissant dans les Tableaux 5.3 et 5.4 sont montrées sur les figs. 5.23-5.25 pour des solutions B-spline et sur les figs. 5.26-5.28 pour les solutions NURBS. En particulier, la frontière en rouge met en évidence les régions du domaine où l'épaisseur minimale est correctement prévue, tandis que la frontière en bleu est constituée de triangles « non faisables », c'est-à-dire les zones caractérisées par une épaisseur inférieure à celle prévue.

Les résultats montrent clairement que l'épaisseur minimale est correctement prévue dans une large zone du domaine. L'épaisseur minimale est inférieure à la valeur prédite uniquement dans des régions très circonscrites. Quoi qu'il en soit, il convient de noter

que la stratégie proposée est à la fois simple et fiable pour être utilisée dans l’algorithme SIMP basé sur les NURBS.

### 5.3.6 Les effets d’un *knot vector* non uniforme sur l’épaisseur minimale

Étant donné que uniquement des *knot vectors* uniformes ont été considérés jusqu’à présent, il semble intéressant d’étudier l’influence d’un *knot vector* non uniforme sur la topologie optimisée. En parallèle, la robustesse de la méthode de prédiction de l’épaisseur minimale est testée. Considérons le benchmark 2D de la section 5.3.4, et en particulier la solution correspondant à  $(n_1 + 1) \times (n_2 + 1) = 48 \times 30$  (les figs. 5.21c et 5.21i illustrent respectivement les solutions B-Spline et NURBS). Dans le cas des figs. 5.21c et 5.21i, le réseau constitué par les composantes des *knot vectors* est représenté à la fig. 5.29a. Deux autres distributions non uniformes des composantes sont considérées pour cet exemple : elles sont indiquées sur les figs. 5.29b et 5.29c.

Le même problème d’optimisation est posé pour la plaque encadrée 2D en appliquant ces deux *knot vectors* différents et les résultats sont présentés sur la fig. 5.30 pour les surfaces B-Spline et sur la fig. 5.31 pour les NURBS.

Les résultats des figs. 5.30 et 5.31 mettent en évidence un fort potentiel de l’algorithme d’OT basé sur les NURBS : le même maillage et le même nombre de points de contrôle et degré de polynômes peuvent produire différentes topologies grâce aux distributions des *knot vectors*. Il est intéressant de vérifier que la prévision de l’épaisseur minimale au moyen des graphiques de la section 5.3.1 est toujours conservatrice (la valeur de seuil estimée est  $\rho_{th} = 0,5$ ). Ces résultats mettent en évidence que, lorsque le nombre de points de contrôle est maintenu constant, le concepteur peut distribuer les composantes des *knot vectors* afin d’avoir différentes dimensions des entités topologiques sur des zones du domaine spécifiées.

## 5.4 L’épaisseur maximale dans l’algorithme d’OT basé sur les NURBS

### 5.4.1 L’énoncé mathématique de la contrainte de l’épaisseur maximale

La contrainte d’épaisseur maximale est utilisée dans les analyses d’OT afin de limiter l’épaisseur maximale des entités topologiques apparaissant lors de l’optimisation. Contrairement à la contrainte d’épaisseur minimale, une contrainte explicite doit être implémentée. S’inspirant du travail de Guest [43], une formulation générale est fournie ici pour la contrainte d’épaisseur maximale dans le cadre de l’approche SIMP basée sur les NURBS pour les problèmes 2D et 3D.

L'idée est simple : pour une structure choisie, une région circulaire ou sphérique est dessinée autour de chaque barycentre des éléments du maillage pour les problèmes 2D et 3D, respectivement. Le diamètre est égal à l'épaisseur maximale imposée (appelée  $d_{max}$  ci-après). Soit  $\Omega_e$  l'aire/volume de la région circulaire/sphérique centrée sur l'élément  $e$ .

Ainsi, la condition de l'éq. (5.18) doit être remplie pour chaque élément. Dans l'éq. (5.18),  $i$  est un index muet pour indiquer les éléments de maillage dont le barycentre se trouve dans  $\Omega_e$  (cet ensemble peut être appelé le « voisinage » de l'élément  $e$ ),  $\psi$  est un paramètre relaxant ( $0 < \psi < 1$ ),  $\mu_i$  est la mesure de l'élément  $i$  (le volume  $V_i$  en 3D et la surface  $A_i$  en 2D), et  $\bar{\rho}_i$  est la fonction de densité fictive projetée. Dans ce travail, une telle projection est réalisée par l'éq. (5.20), où  $\beta \geq 1$  est un paramètre de pénalisation. Une stratégie d'agrégation appropriée doit être envisagée. Le choix naturel, qui éviterait les effets secondaires compensatoires, consiste à choisir la valeur maximale du côté gauche de l'éq. (5.18) parmi les éléments du maillage. Soit  $\delta_e$  défini comme indiqué dans l'éq. (5.21). Afin d'insérer l'opérateur maximum dans un algorithme basé sur un gradient, une approximation appropriée doit être donnée. La  $\chi$ -norme de l'éq. (5.22) est utilisé ci-dessous, où  $\chi$  est un paramètre de réglage dont la valeur doit être suffisamment élevée. Par conséquent, la contrainte est formulée selon l'éq. (5.23) et se présente sous la forme de la contrainte d'inégalité standard de l'éq. (5.24). De plus, les paramètres libres  $\psi$  et  $\chi$  sont définis comme  $\psi = 0.05$  et  $\chi = 15$  et ils ne changent pas pendant les itérations.

L'expression analytique des dérivées de la contrainte est rapportée dans les eqs. (5.25) et (5.26) (voir Appendix C).

### 5.4.2 Résultats : l'épaisseur maximale en 2D

Les effets de la contrainte d'épaisseur maximale dans le cadre de l'approche SIMP basée sur les NURBS sont étudiés ici sur le benchmark 2D illustré à la fig. 5.32. Le domaine rectangulaire est discrétisé au moyen de  $100 \times 50$  éléments coque.

Une surface B-Spline ( $p_1 = p_2 = 3$ ,  $(n_1 + 1) \times (n_2 + 1) = 40 \times 20$ ) est choisie pour effectuer l'analyse d'OT. Premièrement, le problème standard (4.10) est résolu avec  $\gamma = 0.4$  et  $V_{ref} = a_1 a_2 t$ . Ensuite, l'introduction d'une contrainte d'épaisseur maximale selon l'éq. (5.24) est considérée avec  $d_{max} = 25$  mm.

Les solutions des deux problèmes, fournies à la fois par l'approche proposée et par OptiStruct<sup>®</sup>, sont présentées aux figs. 5.33a et 5.33b, respectivement.

La méthode SIMP basée sur les NURBS fournit des résultats cohérents avec ceux obtenus par le logiciel commercial OptiStruct<sup>®</sup> du point de vue numérique : la différence en pourcentage est de 3,7%, mais le nombre de variables de conception pour l'approche SIMP basée sur les NURBS est beaucoup plus petite (800) que celle qui caractérise la solution OptiStruct<sup>®</sup> (5000). Lorsque l'on considère la contrainte d'épaisseur maximale, la différence en pourcentage entre la solution basée sur les NURBS et la solution OptiStruct<sup>®</sup> est réduite à 2,1%. Cependant, dans le second cas, des modifications significatives de la topologie peuvent être observées, voir les figs. 5.33b et 5.34b.

Si l'épaisseur des éléments topologiques est mesurée sur la géométrie reconstruite, la valeur maximale de 25 mm n'est pas nécessairement satisfaite à proximité de la région où la charge est appliquée, voir figs. 5.33b et 5.34b. Néanmoins, cette circonstance est plus critique en ce qui concerne la solution fournie par OptiStruct®.

### 5.4.3 Résultats : l'épaisseur maximale en 3D

L'efficacité de la contrainte d'épaisseur maximale en 3D est prouvée dans ce paragraphe à travers le benchmark de la fig. 5.35. Le domaine est paramétré avec une hyper-surface B-Spline, dont les paramètres sont définis comme  $p_1 = p_2 = p_3 = 2$  et  $(n_1 + 1) \times (n_2 + 1) \times (n_3 + 1) = 32 \times 8 \times 12$ .

Le problème classique de la minimisation de la compliance avec une fraction de volume imposée est d'abord résolu selon l'éq. (4.10), où  $c_{ref} = 241.42$  Nmm,  $V_{ref} = 10^7$  mm<sup>3</sup> et  $\gamma = 0.4$  et sans la contrainte d'épaisseur maximale. Le résultat de cette OT est représenté à la fig. 5.36. La compliance adimensionnelle de la structure finale est  $c/c_{ref} = 0,1422$ .

Alors, le problème (4.10) est enrichie avec la contrainte d'épaisseur maximale sous la forme de l'éq. (5.24), où l'épaisseur maximale est définie  $d_{max} = 30$  mm. La solution du problème (4.10) avec la contrainte d'épaisseur maximale est illustrée à la fig. 5.37. En raison de cette contrainte active, le rapport de compliance est  $c/c_{ref} = 0,2994$ . La valeur de la contrainte est  $g_{d_{max}} = 1.601 \times 10^{-6}$ .

Il est intéressant de noter que la formulation proposée pour l'épaisseur maximale fonctionne également avec un maillage relativement grossier.

## 5.5 Le rayon minimal de courbure

### 5.5.1 Enoncé mathématique de la contrainte sur le rayon minimal de courbure

Ce type de contrainte a un intérêt pour les exigences fonctionnelles (du point de vue des concepteurs) et de fabrication. La valeur minimale du rayon de courbure peut être contrainte à être supérieure à une valeur de référence admissible.

Soit  $\Omega \subseteq D$  le domaine matériel et  $\rho_{cut} \in [\hat{\rho}_{min}, \hat{\rho}_{max}]$  une valeur de coupe pour le champ de densité. Pour avoir une description précise du contour, l'éq. (5.27) peut être considérée. Pour une courbe 2D implicite, l'expression de la courbure est donnée par l'éq. (5.28) [123].

En utilisant les relations de l'éq. (4.3) et en réarrangeant l'éq. (5.28), on peut obtenir le rayon de courbure comme dans l'éq. (5.29). Par conséquent, la contrainte peut être formulée comme indiqué dans l'éq. (5.30). La valeur absolue est approximée au moyen de l'éq. (5.6), tandis que l'opérateur minimum a été estimé via la fonction Kreisselmeier-Steinhauser [124]. Soit  $N_r$  le nombre d'évaluations du rayon sur le contour de la structure.

L'éq. (5.30) se transforme en éq. (5.31), où  $\Upsilon$  doit être assez grand.

Les dérivées de la contrainte sur le rayon de courbure local par rapport aux variables de conception (points de contrôle et poids) peuvent être exprimées par l'éq. (5.32). Des détails sont fournis à l'Annexe D.

### 5.5.2 Résultats : application de la contrainte sur le rayon minimal de courbure

La contrainte sur le rayon de courbure local est testée ici. Une version améliorée du problème (4.23) est résolue en tenant compte de la contrainte de l'éq. (5.31). En particulier, un rayon de courbure minimum  $\bar{r} = 7,5$  mm est imposé. Dans cette section, la solution de la fig. 4.10b est considérée comme une solution de référence. Les points avec une courbure critique sont mis en évidence à la fig. 5.38a et les rayons de courbure respectifs sont :  $r_{A'} = 4.4$  mm,  $r_{B'} = 2.7$  mm,  $r_{D'} = 4.4$  mm,  $r_{E'} = 4.7$  mm. La solution du même problème, enrichi par la contrainte de rayon de courbure minimale, est illustrée à la fig. 5.38b. Les rayons de courbure critiques sont :  $r_A = 9.3$  mm,  $r_B = 8.9$  mm,  $r_C = 8.7$  mm,  $r_D = 8.9$  mm,  $r_E = 7.8$  mm.

La formulation NURBS permet d'avoir une description géométrique précise et bien définie de la frontière de la topologie à chaque itération pendant le processus de résolution. Ainsi, les grandeurs locales (comme le rayon de courbure) peuvent être facilement calculées au moyen du formalisme NURBS.

## 5.6 Conclusions et Perspectives concernant les contraintes géométriques dans l'Algorithme d'OT basé sur les NURBS

Dans ce chapitre, il a été démontré que l'approche SIMP basée sur les NURBS permet de gérer efficacement les contraintes géométriques.

Comme premier aspect, il a été prouvé que l'épaisseur minimale selon la contrainte de Poulsen souffre de certains inconvénients empêchant une utilisation généralisée de cette méthode dans l'OT. Au-delà de la dépendance à un maillage régulier et de la frontière irrégulière, la méthode de Poulsen pourrait également devenir lourde, surtout en 3D.

Grâce aux caractéristiques géométriques de l'entité NURBS, il a été possible de surmonter les difficultés de la méthode de Poulsen et de contrôler l'épaisseur minimale sur la géométrie réassemblée en choisissant simplement les paramètres discrets des NURBS (nombre de points de contrôle et degrés) et certains de leurs paramètres continus, à travers des abaques très généraux. Aucune autre contrainte d'optimisation ne doit être ajoutée à la formulation du problème. Cette méthode est entièrement basée sur la géométrie et ne dépend pas du maillage sous-jacent. Il a été démontré que le concepteur peut décider de définir un *knot vector* selon un critère *ad hoc* et, par conséquent, de réaliser une OT avec



une épaisseur minimale différente dans certaines régions du domaine.

Inversement, l'épaisseur maximale doit être contrôlée par une contrainte appropriée : la formulation de Guest a été reformulée dans le cadre de la méthode SIMP basée sur les NURBS. Même si l'épaisseur maximale n'est pas complètement indépendante du maillage EF, son efficacité a été démontrée dans le contexte le plus général des hyper-surfaces 4D NURBS (problèmes 3D) ainsi que des structures 2D. Malheureusement, rien ne garantit que la contrainte soit localement satisfaite sur toute la phase matérielle après le post-traitement.

Une nouvelle contrainte géométrique sur le rayon de courbure local a été implémentée. Dans le cadre du formalisme NURBS, il s'agit d'une tâche relativement simple car une description géométrique bien définie de la frontière de la topologie est toujours disponible pendant les itérations.

Les perspectives du travail présenté dans ce chapitre sont multiples :

- a) En ce qui concerne l'épaisseur minimale requise, les abaques dérivés peuvent être utilisés, avec une vision conservatrice du problème, dans le cadre des entités NURBS, mais leur utilisation n'est rigoureusement justifiée que dans le cas des entités B-Spline. Dans ce contexte, une solution en forme fermée pour les abscisses de Greville pour les entités NURBS doit être dérivée et intégrée dans l'algorithme proposé dans ce travail afin d'obtenir des abaques pertinents.
- b) Ce chapitre a mis en évidence certaines conséquences prometteuses dans la variation des composantes du *knot vector* dans la méthode OT basée sur les NURBS. En effet, les composantes des *knot vectors* pourraient être intégrées parmi les variables d'optimisation afin d'améliorer la qualité des résultats et de répondre à la contrainte sur l'épaisseur minimale sur le domaine de calcul global. En outre, on s'attend à ce qu'une distribution du *knot vector* optimisée puisse corriger l'effet «vaguelettes» parfois présenté par les solutions B-Spline (voir le chapitre 4). Des recherches sont en cours sur ce sujet.
- c) L'imposition d'une épaisseur minimale et maximale permet de créer des topologies avec des branches de taille uniforme. Cet aspect pourrait être particulièrement avantageux dans la production d'ALM afin d'éviter les contraintes résiduelles dans la structure finale. La mise en relation des informations géométriques et du comportement thermomécanique pourrait permettre de mieux comprendre les liens entre OT et ALM.
- d) La méthode adoptée pour imposer l'épaisseur minimale a un effet secondaire bénéfique important : elle est capable de donner à certains paramètres NURBS une signification physique / géométrique précise. Par conséquent, l'implémentation de contraintes géométriques / mécaniques supplémentaires dans l'algorithme SIMP basé sur les NURBS est d'une importance primordiale.

# Chapter 6

## Eigenvalue Problems in the Framework of the NURBS-Based Topology Optimisation Algorithm

### 6.1 Introduction

This Chapter focuses on the formulation of eigenvalue problems in the framework of the NURBS-based SIMP method. The related implementation in the algorithm *SANTO*, presented in Chapter 4, is discussed as well. Therefore, the library of physical responses provided in *SANTO* is enhanced by introducing buckling loads and natural frequencies, which are often part of technical requirements in components design. TO problems involving these quantities, in their simplest (linearised) formulation, exhibit several similarities, since both are stated as eigenvalue problems in the FE context. Although the non-linear behaviour is neglected, the integration of buckling loads or natural frequencies in TO presents some important and well-known numerical complications: ill-conditioning aspects result from density-based methods and from the mathematical nature of the optimisation problem.

Some issues related to the formulation of buckling criteria have been firstly remarked in the framework of truss topology optimisation [125]. For example, in [126], it is specified that buckling or stress constraints could lead to misleading results, wherein the actual stable structure is far from that proposed by the algorithm. Some improvements have been suggested in [127], where system stability constraints and structure imperfections are taken into account. A basic issue that is associated to truss TO is the presence of “singular optima” [128]: due to the non-convex nature of the optimisation problem and to the implicit non-linearity, the domain of feasible solutions is not mathematically connected. Particularly, there are isolated, potentially optimum solutions, which cannot be reached if a standard gradient-based strategy is used. Authors in [128] propose to modify the

optimisation problem statement through the so-called  $\epsilon$ -relaxed formulation. Although the resulting optimum of the relaxed problem constitute a pseudo-optimum for the original problem, it has been verified that the pseudo-optimum solution is a good approximation of the real optimum. The “singular optima” phenomenon is typical of eigenvalue problems in both discrete and continuous formulations, preventing a good convergence rate and the achievement of the global optimum. An exhaustive discussion on these aspects can be found in [129]. In density-based TO algorithms, a strong ill-conditioning of both stiffness and geometric stiffness matrices arises in low density regions; the consequence is the occurrence of spurious modes associated to very low buckling loads [130]. In order to avoid this phenomenon, preventing a good convergence behaviour, it is advised to neglect those elements that exhibit both a low pseudo-density value and low normalised stress when assembling the geometric stiffness matrix. However, this numerical artifact could lead to abrupt changes of both objective and constraints functions. Therefore, it results in convergence hindrance [5]. To solve the problem of local spurious modes, authors in [5] propose to use different penalisation schemes for the stiffness matrix and for the geometric stiffness matrix, when buckling problems are solved. Instead of using the simple power law formula of the SIMP method, two *ad hoc* material interpolation schemes should be chosen in order to obtain a stronger penalisation on the geometric stiffness matrix within the low density regions of the design domain. The effectiveness of this technique has been proven also in multi-scale optimisation problems including buckling constraints [131].

Another issue affecting eigenvalue formulations is the “mode switching”, that occurs when, for instance, the first buckling load (or the first eigen-frequency) is maximised [5]. In this case, if special precautions are not taken, the other buckling loads (eigen-frequencies) and the associated eigenvectors (i.e. the modes) can become the most critical and the objective function undergoes a jump, with detrimental effects on convergence. To overcome this problem, a suitable bound-formulation is proposed in [5].

A valuable study on the implementation of buckling in density-based TO algorithms has been conducted in [132], where it is underlined that even the method proposed in [5] should be carefully handled, since the involved parameters have a significant impact on the final solution. Moreover, the typical complications related to buckling loads disappear when a “base thickness” formulation is adopted. Imposing a base thickness corresponds to increase the lower bound for the pseudo-density field ( $10^{-1}$  instead of  $10^{-3}$ ). Therefore, the TO problem changes into a *reinforcement problem*: in fact, in this case, the low density zones cannot strictly be interpreted as a cavity in the ground structure. An important aspect discussed in [132] is the importance of continuous load path in buckling problems. Actually, the quality of the convergence can be significantly improved if preferential load paths can be identified: they can be directly established *a priori* by setting suitable NDRs [132], or by an *ad hoc* method for assuring load path continuity, like that proposed in [133].

The extension of a buckling criterion to a non-linear TO algorithm is a significant

complication of the topic and it has been studied in [134], where structures exhibiting snap-through behaviour are investigated. One of the main conclusions of [134] is that standard strategies commonly employed to overcome the issues related to low-density regions are ineffective in the non-linear formulation.

In the last five years, some variants of classic buckling problems have been proposed. In [135], the problem of volume minimisation is formulated in the framework of binary programming methods and it is enhanced with two constraints on stiffness and on stability, respectively. In [136], the buckling problem in TO is replaced by a locally formulated TO problem based on compliance minimisation. An interesting study has been carried out by authors in [137], who apply an energetic method for the buckling modes identification. In particular, instead of an *ad hoc* penalisation scheme, which relies on problem-dependent parameters, it has been preferred to exploit some spurious modes properties to identify them: firstly, they are characterised by extremely low eigenvalues and, secondly, the contribution to the strain energy mainly originates from low density elements for these modes. Unfortunately, even if the energetic method seems to be sound, the modes identification is always dependent on threshold parameters. Furthermore, the authors insist on the fact that coupling compliance minimisation with a stability constraint could lead to an ill-conditioned problem because of the conflictual nature of stiffness and stability in TO. Thus, a smart two-step strategy is proposed, wherein the buckling constraint is activated only after a certain number of iterations and the complete problem of compliance minimisation with volume and buckling load constraint is solved by taking as initial guess for the design variables the results of the classic compliance minimisation with volume constraint problem. This approach has been deepened in [138]: in order to obtain better convergence, the compliance and the buckling constraints should be barely active during all the optimisation process. Indeed, if one among the two constraints is slack, the topology evolves by ignoring it and, at next iterations, a non-feasible solution could appear. Hence, supplementary computational resources and time must be spent to redistribute material and meet both constraints. The solution to this problem is including the SIMP penalisation parameter among the design variables and to compute its initial guess in such a way that both constraints are barely met. This artifact can be interpreted as an improvement of classic continuation methods [139].

The natural frequencies problem exhibits many common features with the buckling load problem in TO, as singular optima, local spurious modes and mode switching. The latter issue is one of the most sensitive in the context of eigen-frequencies problems. In order to simultaneously handle several natural frequencies, many different techniques are available in literature. For example, in [140], some optimisation problems with natural frequencies are shown, like maximisation of an eigen-frequency or maximisation of the distance of two eigen-frequencies from a prescribed value. The idea of weighted mean eigen-frequency is discussed and used in [141].

Like in buckling problems [132], increasing the lower bound of the pseudo-density is still

an effective practice to avoid spurious localised modes [142], but this fact invalidates the original TO problem statement. Instead of solving reinforcement problems, Pedersen uses a continuous but not differentiable penalisation scheme for the stiffness matrix, assuring a finite ratio between structural mass and structural stiffness in low density elements [143]. The mutual penalisation scheme of stiffness and mass matrices plays a fundamental role in any dynamic problem in TO, including the maximisation of the steady state response of a structure subject to a time-harmonic load [144]. A further application, focusing on the maximisation of the gap between two adjacent eigen-frequencies for two-material structures, has been discussed in [145]. In order to improve the algorithm robustness with respect to the mode switching phenomenon, authors in [146] propose a sensitivity analysis based on the mathematical perturbations analysis, whilst a suitable mass matrix penalisation allows for handling the spurious modes. Eigen-frequencies have been considered in the context of multi-scale problems as well [147]: the first eigen-frequency of the macro structure is maximised by acting on the topology of the micro structure. Meanwhile, a polynomial penalisation scheme is suggested in order to obtain a stiffness-to-mass ratio different from zero in low density regions. A smart way to track eigenvalue modes (and consequently to prevent mode switching) is the Modal Assurance Criterion (MAC) discussed in [148]: the problem formulation is enhanced by adding a constraint on a special parameter, measuring the correlation among two modes. An interesting generalisation of the eigen-frequencies problem to the framework of the couple-stress continuum has been provided by [149].

Although several numerical issues have been considered in bibliography about eigenvalue problems in TO, the consistency between the optimised configuration and the actual reassembled geometry in CAD environment is still an open problem. Buckling loads and eigen-frequencies constitute requirements of paramount importance, especially in specific industrial domains: for instance, buckling loads are among the most important failure criteria for aeronautic structures. Therefore, an effective design of structures undergoing buckling phenomena or involving vibrations necessarily needs suitable tools. To this aim, buckling loads and eigen-frequencies are included in the TO algorithm *SANTO*, in order to exploit the advantages of the NURBS formulation also when dealing with these mechanical quantities in the framework of 2D structures.

The Chapter follows this outline. Sections 6.2 and 6.3 deal with buckling problems and eigen-frequencies problems, respectively. Both sections 6.2 and 6.3 are organised by following the same structure: firstly, the TO problem is stated in the classic SIMP framework and, secondly, the corresponding formulation for the NURBS-based TO algorithm is provided. A particular attention is dedicated to the calculation of the derivatives for the gradient-based algorithm when a commercial FE software is used (as stated in previous Chapters, no in-house routines have been used for the structural analysis). Then, numerical issues are discussed and results are shown. Finally, section 6.4 ends the Chapter with conclusions and perspectives.

## 6.2 Eigenvalue Buckling Problems

### 6.2.1 First buckling load maximisation in the classic SIMP framework

The governing equation of a linear eigenvalue buckling problem in FE framework is

$$([\mathbf{K}_\lambda] + \lambda_i [\mathbf{K}_\sigma]) \{\phi_i\} = \{0\}, \quad (6.1)$$

where  $[\mathbf{K}_\lambda]$  is the global stiffness matrix of the structure when the buckling problem is faced,  $\lambda_i$  is the  $i$ -th buckling load (eigenvalue),  $[\mathbf{K}_\sigma]$  is the so-called geometric stiffness matrix (depending on the load case at hand) and  $\{\phi_i\}$  is the  $i$ -th buckling mode (the eigenvector of DOFs associated to the  $i$ -th eigenvalue). The stiffness and the geometric stiffness matrices are related to the pseudo-density field through two *ad hoc* penalisation schemes, i.e.

$$[\mathbf{K}_\lambda] = \sum_{e=1}^{N_e} f_\lambda(\rho_e) [\mathbf{K}_e^0] = \sum_{e=1}^{N_e} [\mathbf{K}_{\lambda-e}], \quad (6.2)$$

$$[\mathbf{K}_\sigma] = \sum_{e=1}^{N_e} f_\sigma(\rho_e) [\mathbf{K}_{\sigma-e}^0] = \sum_{e=1}^{N_e} [\mathbf{K}_{\sigma-e}]. \quad (6.3)$$

Eqs. (6.2)-(6.3) are consistent with the notation of Chapter 1, therefore  $\rho_e$  is the fictitious density computed at the centroid of the generic element  $e$ ,  $N_e$  is the total number of elements;  $[\mathbf{K}_e^0]$  and  $[\mathbf{K}_{\sigma-e}^0]$  are the non-penalised element stiffness and element geometric stiffness matrices, respectively, both expanded over the whole set of DOFs of the structure.  $f_\lambda(\rho_e)$  and  $f_\sigma(\rho_e)$  represent the penalisation functions for the respective matrix and they are, *a priori*, different. As detailed in section 6.1, the classic SIMP power law  $\rho_e^\alpha$  is not the recommended penalisation scheme in the case of buckling loads. Hence, the functions  $f_\lambda(\rho_e)$  and  $f_\sigma(\rho_e)$  should be carefully selected in order to prevent the spurious modes phenomenon. Their choice is discussed in section 6.2.3.

A typical problem in TO [5] is the maximisation of the first buckling load with an upper bound on the structure volume. In the SIMP context, this problem reads

$$\begin{aligned} & \max_{\rho_e} \lambda_1(\rho_e), \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} ([\mathbf{K}_\lambda] + \lambda_1 [\mathbf{K}_\sigma]) \{\phi_1\} = \{0\}, \\ \frac{V(\rho_e)}{V_{ref}} = \frac{\sum_{e=1}^{N_e} \rho_e V_e}{V_{ref}} \leq \gamma, \\ \rho_{min} \leq \rho_e \leq 1, \quad e = 1, \dots, N_e. \end{array} \right. \quad (6.4) \end{aligned}$$

Similarly to the compliance problem (1.6), the buckling problem (6.4) can be solved through a suitable gradient-based algorithm: to this purpose, the derivatives of both the objective and the constraint functions with respect to the elements fictitious densities must be computed.

The derivatives of the volume are given by Eq. (1.10).

Computing the gradient of the buckling loads is not a simple task. Considering Eq. (6.1), the following relation can be inferred:

$$\{\phi_i\}^T ([K_\lambda] + \lambda_i [K_\sigma]) \{\phi_i\} = 0. \quad (6.5)$$

If the derivative with respect to the element pseudo-density  $\partial/\partial\rho_e$  is performed, it is obtained

$$\begin{aligned} \frac{\partial \{\phi_i\}^T}{\partial\rho_e} ([K_\lambda] + \lambda_i [K_\sigma]) \{\phi_i\} + \{\phi_i\}^T \frac{\partial ([K_\lambda] + \lambda_i [K_\sigma])}{\partial\rho_e} \{\phi_i\} + \\ + \{\phi_i\}^T ([K_\lambda] + \lambda_i [K_\sigma]) \frac{\partial \{\phi_i\}}{\partial\rho_e} = 0. \end{aligned} \quad (6.6)$$

Taking into account Eq. (6.1) and the symmetry of the matrix  $([K_\lambda] + \lambda_i [K_\sigma])$ ,

$$\frac{\partial \{\phi_i\}^T}{\partial\rho_e} ([K_\lambda] + \lambda_i [K_\sigma]) \{\phi_i\} = \{\phi_i\}^T ([K_\lambda] + \lambda_i [K_\sigma]) \frac{\partial \{\phi_i\}}{\partial\rho_e} = 0, \quad (6.7)$$

and consequently

$$\{\phi_i\}^T \frac{\partial ([K_\lambda] + \lambda_i [K_\sigma])}{\partial\rho_e} \{\phi_i\} = 0. \quad (6.8)$$

The explicit expression of Eq. (6.8) is

$$\{\phi_i\}^T \frac{\partial [K_\lambda]}{\partial\rho_e} \{\phi_i\} + \frac{\partial\lambda_i}{\partial\rho_e} \{\phi_i\}^T [K_\sigma] \{\phi_i\} + \lambda_i \{\phi_i\}^T \frac{\partial [K_\sigma]}{\partial\rho_e} \{\phi_i\} = 0. \quad (6.9)$$

The derivative  $\partial [K_\lambda]/\partial\rho_e$  can be easily computed thanks to Eq. (6.2). Nevertheless, before proceeding, it is necessary to clarify the physical meaning of the quantities appearing in Eq. (6.9). As well-known, the geometric stiffening matrix  $[K_\sigma]$  is assembled in the “pre-stress” analysis, that is usually performed before solving the eigenvalue problem. This matrix is a numerical artifact, necessary in order to simulate the equilibrium condition in a configuration which is slightly different from the starting one (i.e. a small perturbation of the non-deformed shape of the structure is considered). Assembling this matrix requires the access to a huge amount of information concerning the FE formulation (the element shape functions, integration points, etc.). In this work, the algorithm *SANTO* is interfaced to a commercial FE software and the previously listed information cannot be easily retrieved. In the context of 2D plates instability problems, an energetic approach

## 6.2. Eigenvalue Buckling Problems

---

can be exploited in order to perform the sensitivity analysis. Accordingly, the matrix  $[K_\sigma]$  does not need to be explicitly computed. Firstly, two suitable quantities can be defined:

$$c_{tot}^i = \{\phi_i\}^T [K_\lambda] \{\phi_i\}, \quad (6.10)$$

that is the total compliance associated to the  $i$ -th buckling mode and

$$c_\sigma^i = \{\phi_i\}^T [K_\sigma] \{\phi_i\}, \quad (6.11)$$

i.e. the virtual work done by the pre-stress loads on the non-linear component of displacements corresponding to the  $i$ -th mode. In the light of these two definitions, Eq. (6.5) can be written as

$$c_{tot}^i + \lambda_i c_\sigma^i = 0. \quad (6.12)$$

It is straightforward to verify that Eq. (6.9) can be rearranged by exploiting Eq. (6.2) and Eq. (6.3):

$$\frac{\partial \lambda_i}{\partial \rho_e} c_\sigma^i = - \left[ \frac{\partial f_\lambda}{\partial \rho_e} \frac{1}{f_\lambda} \{\phi_i\}^T [K_{\lambda-e}] \{\phi_i\} + \lambda_i \frac{\partial f_\sigma}{\partial \rho_e} \frac{1}{f_\sigma} \{\phi_i\}^T [K_{\sigma-e}] \{\phi_i\} \right]. \quad (6.13)$$

In Eq. (6.13),  $c_e^i = \{\phi_i\}^T [K_{\lambda-e}] \{\phi_i\}$  is the contribution of element  $e$  to the total compliance related to the  $i$ -th buckling mode, whilst  $c_{\sigma-e}^i = \{\phi_i\}^T [K_{\sigma-e}^0] \{\phi_i\}$  is the contribution of element  $e$  to the total virtual work of pre-stress loads on the non-linear component of displacements of the  $i$ -th buckling mode. The latter contribution deserves particular attention to be evaluated and, particularly, it needs to evaluate the stress-stiffness matrix for each structure element. If the simple Kirchhoff's plate theory is considered, the kinematics is completely determined by the displacement field  $(v_1, v_2, v_3)$ , wherein  $v_1$  and  $v_2$  represent the in-plane components of displacement, whilst  $v_3$  is the out-of-plane component. Let  $N_{x_1}^0, N_{x_2}^0$  and  $N_{x_1 x_2}^0$  be the linear efforts, whose values are obtained as the integral on the plate thickness of the stress components  $\sigma_{x_1}^0, \sigma_{x_2}^0$  and  $\sigma_{x_1 x_2}^0$ , due to pre-stress loads. Under these hypotheses,  $c_\sigma^i$  is easy to compute:

$$c_\sigma^i = \int_D \left( N_{x_1}^0 \left( \frac{\partial v_3}{\partial x_1} \right)^2 + N_{x_2}^0 \left( \frac{\partial v_3}{\partial x_2} \right)^2 - 2N_{x_1 x_2}^0 \frac{\partial v_3}{\partial x_1} \frac{\partial v_3}{\partial x_2} \right) dS. \quad (6.14)$$

Therefore,  $c_{\sigma-e}^i$  is evaluated by limiting the previous integral to the  $e$ -th element surface.

Thus, the final form of the sensitivity analysis of the  $i$ -th buckling load is

$$\frac{\partial \lambda_i}{\partial \rho_e} = \frac{\lambda_i}{c_{tot}^i} \left( \frac{\partial f_\lambda}{\partial \rho_e} \frac{1}{f_\lambda} c_e^i + \lambda_i \frac{\partial f_\sigma}{\partial \rho_e} \frac{1}{f_\sigma} c_{\sigma-e}^i \right), \quad e = 1, \dots, N_e. \quad (6.15)$$

The provided sensitivity analysis of buckling loads can be exploited for solving problem (6.4).



## 6.2.2 Mathematical formulation of buckling problems in the NURBS-based TO algorithm

As it has been done for the compliance problem in Chapter 4, the buckling problem can be reformulated in the framework of the NURBS-based SIMP method. The new design variables are the NURBS control points and weights, collected in the arrays  $\Xi_1$  and  $\Xi_2$ , respectively.

The mathematical statement of problem (6.4) in the NURBS framework is

$$\begin{aligned} & \min_{\Xi^{(1)}, \Xi^{(2)}} - \frac{\lambda_1(\rho(\Xi^{(1)}, \Xi^{(2)}))}{\lambda_{ref}}, \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} ([K_\lambda] + \lambda_i [K_\sigma]) \{\phi_i\} = \{0\}, \\ \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref}} = \frac{\sum_{e=1}^{N_e} \rho_e V_e}{V_{ref}} \leq \gamma, \\ \Xi_k^{(1)} \in [\hat{\rho}_{min}, \hat{\rho}_{max}], \\ \Xi_k^{(2)} \in [w_{min}, w_{max}], \\ \forall k = 1, \dots, n_{tot}. \end{array} \right. \end{array} \quad (6.16)$$

In order to solve problem (6.16) through a suitable gradient-based algorithm, the derivatives of the buckling load with respect to the new set of design variables must be computed. The gradient of a specific functional with respect to the control point  $\hat{\rho}_{I_1, I_2} = \Xi_\tau^{(1)}$  or the weight  $w_{I_1, I_2} = \Xi_\tau^{(2)}$  can be easily calculated by exploiting the local support property and the chain rule for derivatives (as explained in Chapter 4). The relationship between the doublet  $I_1, I_2$  and the linear index  $\tau$  is still given by Eq. (4.13).

The derivatives of whatever buckling load  $\lambda_i$  with respect to the NURBS control points and weights can be deduced by combining Eqs. (4.16)-(4.17) with Eq. (6.15) and read

$$\frac{\partial \lambda_i}{\partial \Xi_\tau^{(1)}} = \frac{\lambda_i}{c_{tot}^i} \sum_{e \in S_\tau} R_\tau^e \left( \frac{\partial f_\lambda}{\partial \rho_e} \frac{1}{f_\lambda} c_e^i + \lambda_i \frac{\partial f_\sigma}{\partial \rho_e} \frac{1}{f_\sigma} c_{\sigma-e}^i \right), \quad (6.17)$$

$$\frac{\partial \lambda_i}{\partial \Xi_\tau^{(2)}} = \frac{\lambda_i}{\Xi_\tau^{(2)} c_{tot}^i} \sum_{e \in S_\tau} R_\tau^e (\Xi_\tau^{(1)} - \rho_e) \left( \frac{\partial f_\lambda}{\partial \rho_e} \frac{1}{f_\lambda} c_e^i + \lambda_i \frac{\partial f_\sigma}{\partial \rho_e} \frac{1}{f_\sigma} c_{\sigma-e}^i \right). \quad (6.18)$$

## 6.2.3 Discussion on numerical aspects

The numerical tool used to solve problem (6.16) is the algorithm *SANTO* described in Chapter 4.

In this section, the attention is focused on numerical strategies which have been adopted to solve the TO eigenvalue buckling problem. Particularly, the spurious modes numerical issue is solved through two techniques, which are detailed here below:

- the double penalisation scheme, firstly suggested in [5];
- the so-called “base-thickness” technique, discussed in [132].

Being impossible to determine *a priori* which approach exhibits better performances when the TO eigenvalue buckling problem is formulated in the NURBS framework, both techniques have been investigated in this Chapter.

The double penalisation technique consists of assigning a different penalisation scheme to the stiffness matrix and to the geometric stiffness matrix. In this background, the stiffness matrix is penalised as

$$f_{\lambda}(\rho_e) = \rho_{low} + (1 - \rho_{low})\rho_e^{\alpha}, \quad (6.19)$$

whilst the geometric stiffness matrix undergoes the classic SIMP penalisation scheme, i.e.

$$f_{\sigma}(\rho_e) = \rho_e^{\alpha}. \quad (6.20)$$

In Eqs. (6.19)-(6.20),  $\alpha \geq 1$  is the classic SIMP parameter (refer to Chapter 1). In the proposed approach,  $\alpha = 3$ . The parameter  $\rho_{low}$  affects the behaviour of the stiffness matrix with respect to the geometric stiffness matrix when the element density takes low values (close to  $10^{-3}$ ). As remarked by several authors, its choice can affect the result: as shown in Fig. 6.1, the higher the  $\rho_{low}$  value, the stronger the penalisation of the geometric stiffness matrix is with respect to the stiffness matrix in the low density regions, whilst their ratio is very close to 1 in high density regions. A high  $\rho_{low}$  value guarantees to avoid spurious modes but, meanwhile, it gives a fictitious stiffness to the structure and it could result in solving a different problem with respect to the original one. Conversely, a too weak value for  $\rho_{low}$  could lead to spurious modes effect and, again, the TO algorithm would provide misleading results.

The second methodology to prevent the spurious modes issue is to use a “base-thickness” for the structure. This purpose can be achieved by increasing the lower boundary of the pseudo-density in classic SIMP-based algorithms: for instance, it is advised to set  $\rho_{min} = 10^{-1}$  instead of  $\rho_{min} = 10^{-3}$ . In this case, the same penalisation scheme for both the stiffness and geometric stiffness matrices can be used: in fact, the occurrence of spurious modes is prevented by the fact that too “weak” regions never appear during iterations. In this work, instead of acting on the lower boundary of the density, it has been preferred to leave  $\rho_{min} = 10^{-3}$  as usual and to exploit a suitable NDR. Since 2D thin-walled structures are mostly interested by buckling phenomena, an additional stiffness is provided by imposing a thin layer of bulk material inside the plate to be optimised, whose thickness is  $t$ . This inner layer shares the same mean plane of the plate to be optimised

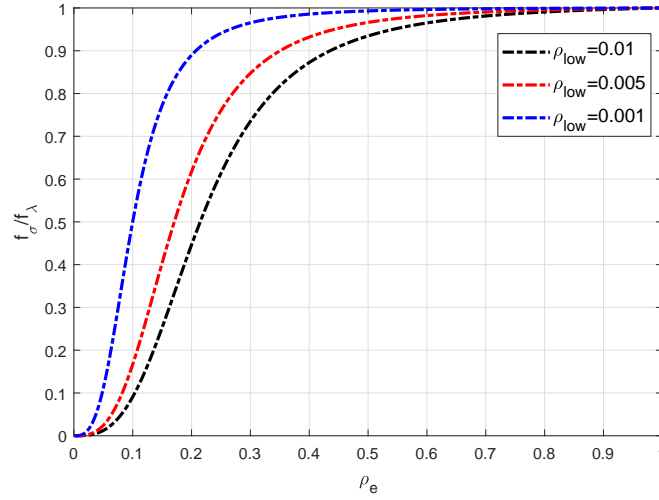


Figure 6.1 – The  $f_\sigma/f_\lambda$  ratio.

and its thickness is referred as  $t_{base}$  (refer to Fig 6.2). When the base-thickness technique is used, the same penalisation scheme is used for both stiffness and geometric stiffness matrices, i.e.  $f_\lambda = f_\sigma = \rho_e^\alpha$ .

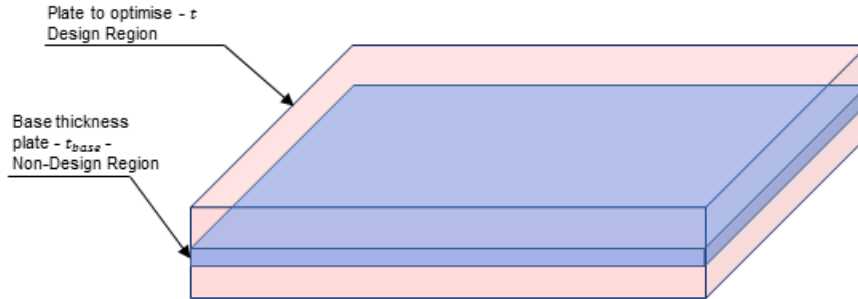


Figure 6.2 – The base-thickness as Non-Design Region.

As the parameter  $\rho_{low}$  influences the on the final solutions for the double penalisation strategy, the thickness of the Non-Design layer ( $t_{base}$ ) affects the quality of the result in the framework of the base-thickness method.

## 6.2.4 Results

### Benchmark description

A meaningful benchmark is considered in this section to prove the effectiveness of the above proposed techniques when formulated in the NURBS-based SIMP approach framework. The benchmark is shown in Fig. 6.3 (all the necessary parameters are included in the figure caption). It consists of an outer frame to be reinforced in order to increase its buckling critical load. This benchmark has been taken from [132] but, contrary to the choice made in [132], the structure allows for natural out-of-plane buckling modes.

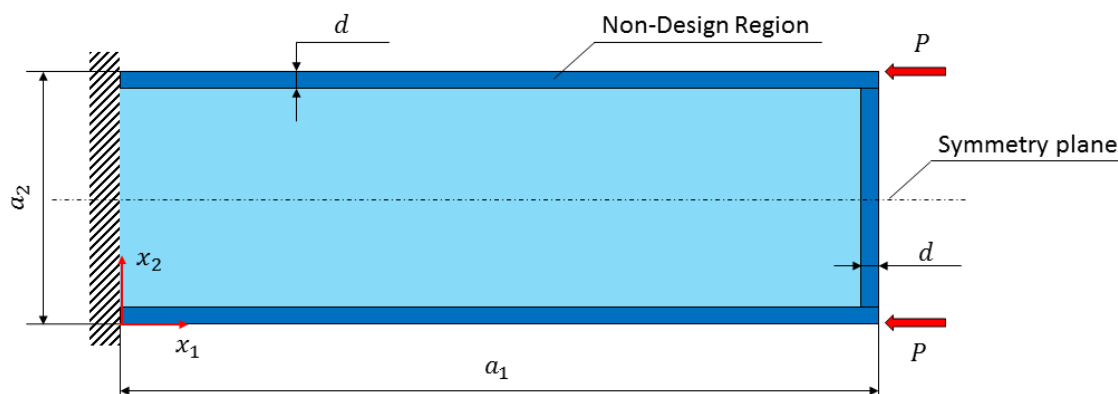


Figure 6.3 – Geometric parameters:  $a_1 = 900$  mm,  $a_2 = 300$  mm,  $d = 20$  mm, thickness  $t = 2$  mm, symmetry plane  $x_2 = a_2/2$  - Material Parameters: Steel, Young's modulus  $E = 200000$  MPa, Poisson's ratio  $\nu = 0.3$  - Loads: Unit vectors

The mathematical formulation of the TO to be solved is stated as

$$\min_{\Xi^{(1)}, \Xi^{(2)}} - \frac{\lambda_1(\rho(\Xi^{(1)}, \Xi^{(2)}))}{\lambda_{ref}},$$

subject to:

$$\left\{ \begin{array}{l} ([K_\lambda] + \lambda_i [K_\sigma]) \{\phi_i\} = \{0\}, \\ \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref}} = \frac{\sum_{e=1}^{N_e} \rho_e V_e}{V_{ref}} \leq \gamma, \\ \text{symmetry with respect to the plane } x_2 = a_2/2, \\ \Xi_k^{(1)} \in [10^{-3}, 1], \\ \Xi_k^{(2)} \in [1/2, 10], \\ \forall k = 1, \dots, n_{tot}. \end{array} \right. \quad (6.21)$$

In problem (6.21),  $\lambda_{ref}$  is the buckling load of the outer frame reinforced by the inner region with  $\rho_e = \gamma$ ,  $\forall e = 1, \dots, N_e$ . This corresponds to the feasible solution

provided for the algorithm initialisation. The reference volume is the volume of the DR, i.e.  $V_{ref} = 457600 \text{ mm}^3$ . For this benchmark,  $\gamma = 0.4$ . Preliminary buckling analyses have been carried out on the frame without the internal region (the DR) and convergence on the critical load is achieved with  $10 \text{ mm} \times 10 \text{ mm}$  SHELL181 Ansys elements. In this conditions, the eigenvalue is  $\lambda_0 = 8.137 \text{ N}$ . Therefore, it has been decided to use  $90 \times 30$  SHELL181 Ansys elements for all the TO analyses on the benchmark of Fig. 6.3.

Numerical tests have been carried out by varying the NURBS parameters. Firstly, the double penalisation scheme is used. B-Spline and NURBS solutions are compared by varying the number of control points and the blending functions degrees. Analogous computations are performed by using the base-thickness technique.

The following values of NURBS/BSpline parameters have been tested:

- $(n_1 + 1) \times (n_2 + 1) = 42 \times 14, 60 \times 20, 78 \times 26;$
- $p_1, p_2 = 2, 3, 4.$

Since the purpose of this Chapter is restrained to investigate the effects of the NURBS parameters on the final solutions, the low density parameter  $\rho_{low}$  for the double penalisation scheme and the base-thickness  $t_{base}$  are both initially set and kept constant for all the TO analyses. They have been chosen as  $\rho_{low} = 10^{-3}$  and  $t_{base} = t/30$ : using these values implies a reference buckling load  $\lambda_{ref} = 11.5 \text{ N}$  for both the double penalisation scheme and for the base-thickness technique. Eventual comparisons between results are, therefore, consistent.

Results will be presented in their final form, i.e. after the postprocessing phase. Thanks to the NURBS formalism, the CAD reconstruction/assembly step after the TO analysis is a very straightforward task. Therefore, the numerical results concerning the buckling loads have been obtained by performing an eigenvalue buckling analysis on the actual, reassembled geometry (and not on the starting meshed plate with all mesh elements affected by the related pseudo-density values).

### **B-Spline and NURBS solutions with the double penalisation scheme**

All the geometries shown here below meet the volume constraint. The corresponding buckling load is directly reported in captions of Figs. 6.4-6.6 for B-Spline solutions and of Figs. 6.7-6.9 for NURBS solutions.

Numerical results are summarised in the graph of Fig. 6.10.

Some remarks follows from the analysis of the obtained topologies and from numerical values of the critical buckling loads.

- The double penalisation scheme seems to be effective and avoids the occurrence of spurious modes. This technique is general and it could be easily extended to the case of buckling TO for 3D applications.

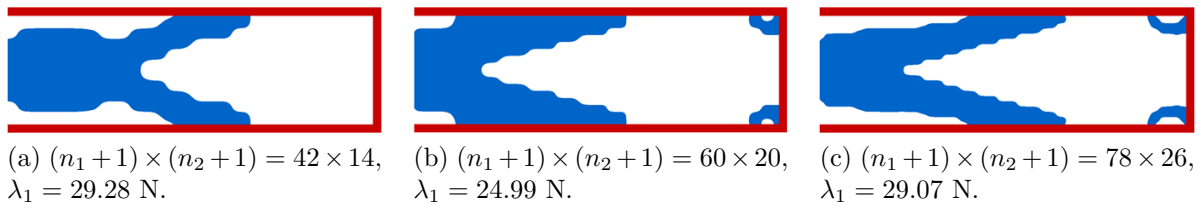


Figure 6.4 – Double penalisation scheme: B-Spline results,  $p_1, p_2 = 2$ .

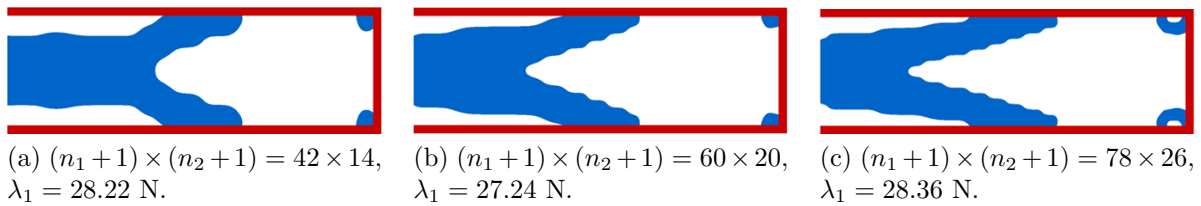


Figure 6.5 – Double penalisation scheme: B-Spline results,  $p_1, p_2 = 3$ .

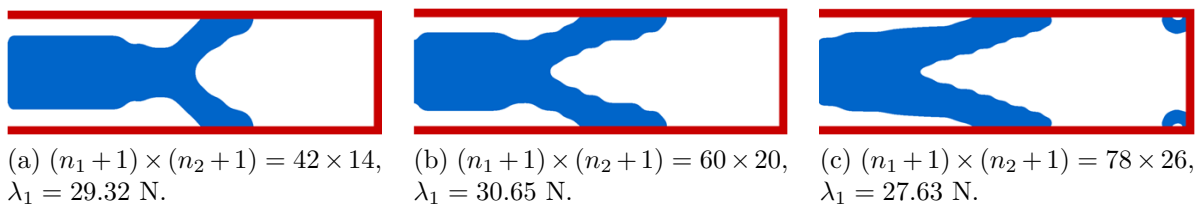


Figure 6.6 – Double penalisation scheme: B-Spline results,  $p_1, p_2 = 4$ .

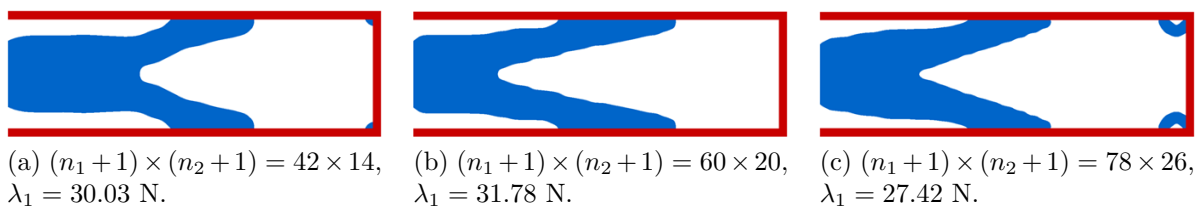


Figure 6.7 – Double penalisation scheme: NURBS results,  $p_1, p_2 = 2$ .

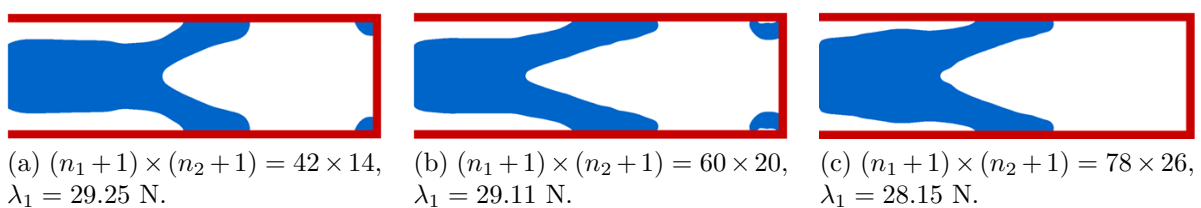


Figure 6.8 – Double penalisation scheme: NURBS results,  $p_1, p_2 = 3$ .

- NURBS solutions exhibit better performances than B-Spline solutions. In the case of

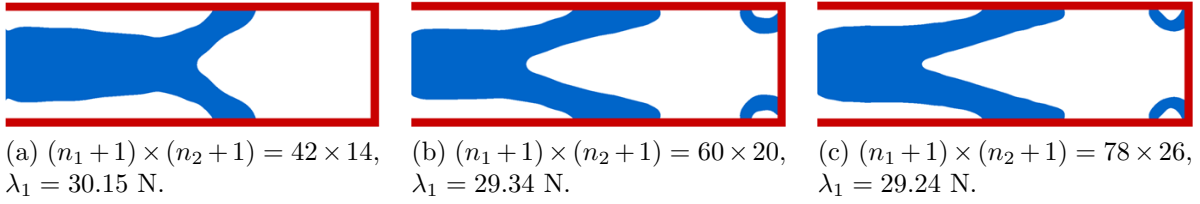


Figure 6.9 – Double penalisation scheme: NURBS results,  $p_1, p_2 = 4$ .

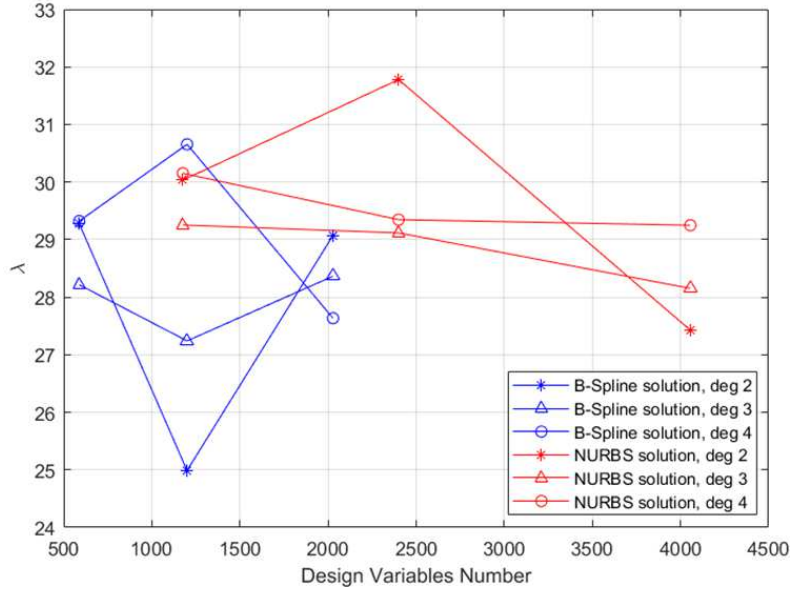


Figure 6.10 – Double penalisation technique: synthesis of results.

buckling, the advantage of including the NURBS weights among the design variables is more evident: a smoother 2D structure boundary implies a significant increase of the buckling load.

- Increasing the number of control point is not, in general, the best choice to obtain more efficient solutions, as it can be easily remarked by Fig. 6.10.

### B-Spline and NURBS solutions with the base-thickness strategy

The same numerical campaign has been repeated in the case of the base-thickness strategy. Results are directly provided in Figs. 6.11-6.16.

A synthesis of numerical results are shown in the graph of Fig. 6.17.

When using the base-thickness technique, the same behaviour of Fig. 6.10 can be observed again: increasing the number of control points does not necessarily imply better performances. Contrariwise, the use of NURBS is strongly recommended to obtain

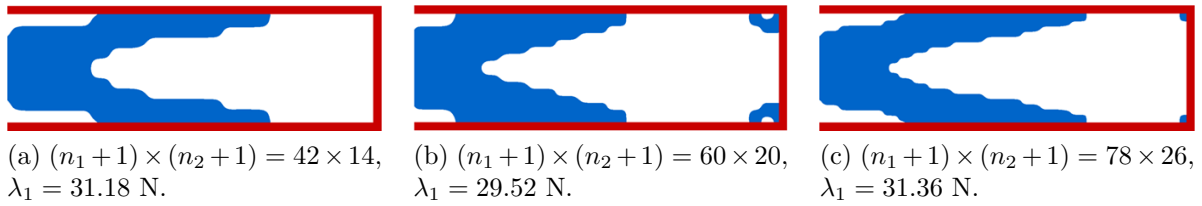


Figure 6.11 – Base-thickness strategy: B-Spline results for  $p_1, p_2 = 2$ .

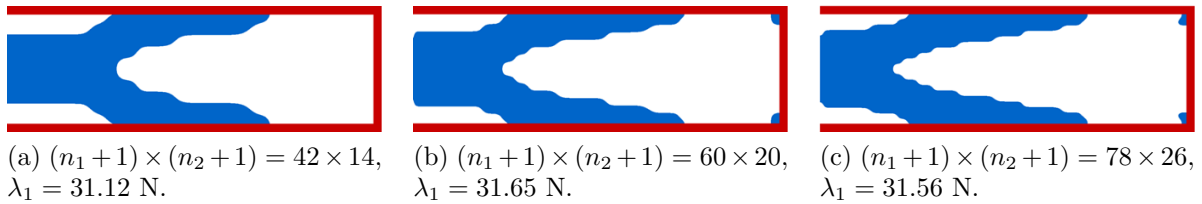


Figure 6.12 – Base-thickness strategy: B-Spline results for  $p_1, p_2 = 3$ .

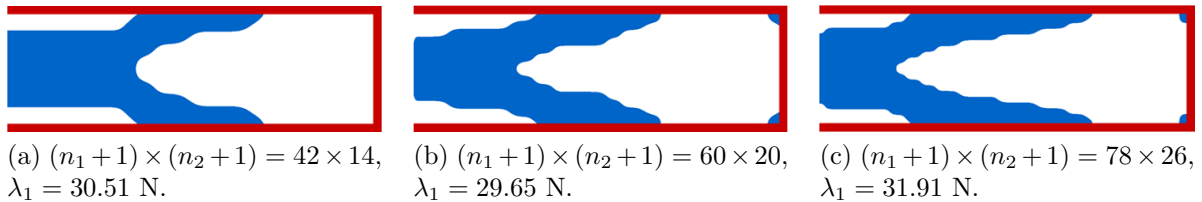


Figure 6.13 – Base-thickness strategy: B-Spline results for  $p_1, p_2 = 4$ .

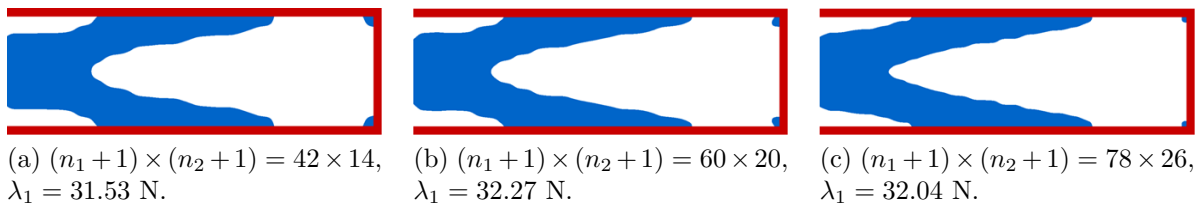


Figure 6.14 – Base-thickness strategy: NURBS results for  $p_1, p_2 = 2$ .

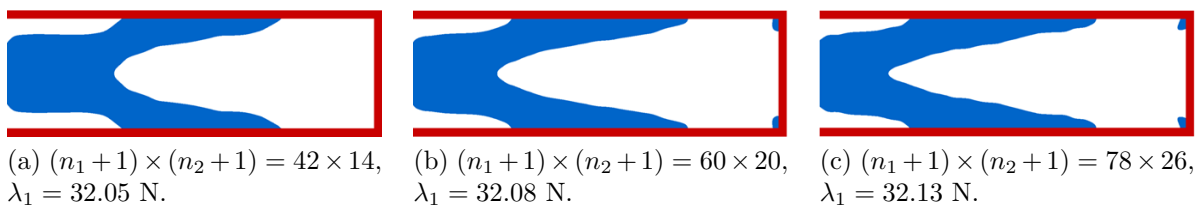


Figure 6.15 – Base-thickness strategy: NURBS results for  $p_1, p_2 = 3$ .

smoother geometries and more efficient configurations.



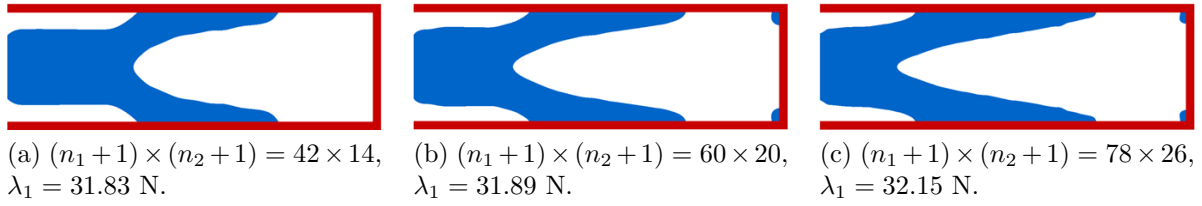


Figure 6.16 – Base-thickness strategy: NURBS results for  $p_1, p_2 = 4$ .

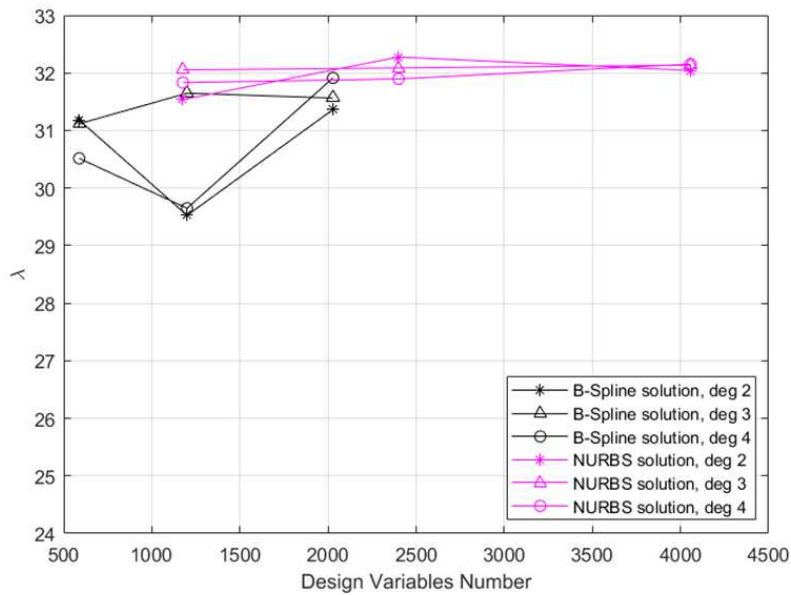


Figure 6.17 – Base-thickness strategy: synthesis of results.

If numerical results are compared (see Fig. 6.10 and Fig. 6.17), the base-thickness technique seems, in general, to provide more efficient solutions when the same number of control points and the same degrees are used. This fact occurs for B-Spline as well as for NURBS surfaces. However, it should be remarked that the double penalisation scheme is general and could be applied, eventually, to whatever structure. The base thickness strategy holds only in the case of plates and shells and it is not general.

## 6.3 Eigen-frequencies Problems

### 6.3.1 First Eigen-frequency maximisation in the classic SIMP framework

If the eigen-frequencies problem is considered, the state equation is

$$([\mathbf{K}_\omega] - \omega_i^2 [\mathbf{M}]) \{\psi_i\} = \{0\}. \quad (6.22)$$

In Eq. (6.22),  $[\mathbf{K}_\omega]$  is the global stiffness matrix of the structure when the eigen-frequencies problem is faced,  $\omega_i$  is the  $i$ -th eigen-frequency,  $[\mathbf{M}]$  is the global mass matrix, whilst  $\{\psi_i\}$  is the  $i$ -th mode (eigenvector of DOFs) related to the corresponding eigen-frequency. In this particular framework, the global stiffness matrix is related to the non-penalised element stiffness matrix  $[\mathbf{K}_e^0]$  through

$$[\mathbf{K}_\omega] = \sum_{e=1}^{N_e} f_\omega(\rho_e) [\mathbf{K}_e^0] = \sum_{e=1}^{N_e} [\mathbf{K}_{\omega-e}], \quad (6.23)$$

where  $f_\omega(\rho_e)$  is a dedicated penalisation scheme and it will be discussed in the following section. Although other methods have been developed in literature (see section 6.1), the penalisation scheme of the mass matrix has been assumed naturally linear in this work, i.e.

$$[\mathbf{M}] = \sum_{e=1}^{N_e} \rho_e [\mathbf{M}_e^0] + [\mathbf{M}_l] = \sum_{e=1}^{N_e} [\mathbf{M}_e] + [\mathbf{M}_l]. \quad (6.24)$$

In Eq. (6.24),  $[\mathbf{M}_e^0]$  is the non-penalised element mass matrix, whilst  $[\mathbf{M}_l]$  is matrix accounting for lumped non-structural masses, i.e. those masses which do not contribute to the global stiffness matrix.

The eigen-frequencies TO problem considered in this work is the maximisation of the first eigen-frequency with a volume inequality constraint: of course, in order to deal with a well-posed problem, it is assumed that suitable NDRs or a non-structural mass are present.

$$\begin{aligned} & \max_{\rho_e} \omega_1(\rho_e), \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} ([\mathbf{K}_\omega] - \omega_1^2 [\mathbf{M}]) \{\psi_1\} = \{0\}, \\ \frac{V(\rho_e)}{V_{ref}} = \frac{\sum_{e=1}^{N_e} \rho_e V_e}{V_{ref}} \leq \gamma, \\ \rho_{min} \leq \rho_e \leq 1, \quad e = 1, \dots, N_e. \end{array} \right. \end{array} \quad (6.25)$$

As far as the sensitivity analysis of the eigen-frequencies is concerned, analogous steps to the buckling load case can be repeated. Particularly, starting from Eq. (6.22), the following relation can be inferred:

$$\{\psi_i\}^T ([\mathbf{K}_\omega] - \omega_i^2 [\mathbf{M}]) \{\psi_i\} = 0. \quad (6.26)$$

Eq. (6.26) is derived with respect to the generic element pseudo-density and, after standard algebraic passages, one can obtain

$$\{\psi_i\}^T \frac{\partial ([\mathbf{K}_\omega] - \omega_i^2 [\mathbf{M}])}{\partial \rho_e} \{\psi_i\} = 0, \quad (6.27)$$

that can be rewritten as

$$\{\psi_i\}^T \frac{\partial [\mathbf{K}_\omega]}{\partial \rho_e} \{\psi_i\} - 2\omega_i \frac{\partial \omega_i}{\partial \rho_e} \{\psi_i\}^T [\mathbf{M}] \{\psi_i\} - \omega_i^2 \{\psi_i\}^T \frac{\partial [\mathbf{M}]}{\partial \rho_e} \{\psi_i\} = 0. \quad (6.28)$$

Moreover, the virtual work of internal forces (the double of the strain energy) and the kinetic energy when the structure is vibrating at  $\omega_i$  can be defined as

$$c_{tot}^i = 2\epsilon^i = \{\psi_i\}^T [\mathbf{K}_\omega] \{\psi_i\}, \quad (6.29)$$

and

$$2\kappa_{tot}^i = \omega_i^2 \{\psi_i\}^T [\mathbf{M}] \{\psi_i\}, \quad (6.30)$$

respectively. In the light of the previous relationships, it is straightforward to arrange Eq. (6.28) to obtain

$$\frac{4\kappa_{tot}^i}{\omega_i} \frac{\partial \omega_i}{\partial \rho_e} = \frac{\partial f_\omega}{\partial \rho_e} \frac{1}{f_\omega} \{\psi_i\}^T [\mathbf{K}_{\omega-e}] \{\psi_i\} - \frac{1}{\rho_e} \omega_i^2 \{\psi_i\}^T [\mathbf{M}_e] \{\psi_i\}. \quad (6.31)$$

As it can be easily deduced,  $2\epsilon_e^i = \{\psi_i\}^T [\mathbf{K}_{\omega-e}] \{\psi_i\}$  represents the contribution to the strain energy given by element  $e$ , whilst  $2\kappa_e^i = \omega_i^2 \{\psi_i\}^T [\mathbf{M}_e] \{\psi_i\}$  is the portion of kinetic energy related to the element  $e$ . Finally, the gradient of the  $i$ -th eigen-frequency can be expressed as:

$$\frac{\partial \omega_i}{\partial \rho_e} = \frac{\omega_i}{2\kappa_{tot}^i} \left( \frac{\partial f_\omega}{\partial \rho_e} \frac{1}{f_\omega} \epsilon_e^i - \frac{1}{\rho_e} \kappa_e^i \right), \quad e = 1, \dots, N_e. \quad (6.32)$$

It is noteworthy that the form of the derivative in Eq. (6.32), contrary to the buckling load case, is general and does not require any further hypothesis to be applied. Particularly, the sensitivity analysis of the  $i$ -th eigen-frequency does not require to access to the FE software internal information, such as shape functions or Gauss points, and it can be evaluated by means of some physical quantities, commonly available in the postprocessing phase of FE analyses.

### 6.3.2 Mathematical formulation of eigen-frequencies problems in the NURBS-based TO algorithm

In this section, problem (6.25) is reformulated in the NURBS-based SIMP approach framework. In particular, the dependence of the mechanical/geometric quantities on the NURBS variables (i.e. pseudo-density at control points and related weights) is emphasised.

The mathematical statement of problem (6.25) in the NURBS framework is

$$\begin{aligned} & \min_{\Xi^{(1)}, \Xi^{(2)}} - \frac{\omega_1(\rho(\Xi^{(1)}, \Xi^{(2)}))}{\omega_{ref}}, \\ & \text{subject to:} \\ & \left\{ \begin{array}{l} ([K_\omega] + \omega_i^2 [M]) \{\psi_i\} = \{0\}, \\ \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref}} = \frac{\sum_{e=1}^{N_e} \rho_e V_e}{V_{ref}} \leq \gamma, \\ \Xi_k^{(1)} \in [\hat{\rho}_{min}, \hat{\rho}_{max}], \\ \Xi_k^{(2)} \in [w_{min}, w_{max}], \\ \forall k = 1, \dots, n_{tot}. \end{array} \right. \end{array} \quad (6.33)$$

In Eq. (6.33),  $\omega_{ref}$  is a reference frequency, defined according to the problem at hand.

The derivatives of the eigen-frequencies with respect to the NURBS control points and weights are easy to compute. Combining Eqs. (4.16)-(4.17) with Eq. (6.32), the following relationships can be easily derived

$$\frac{\partial \omega_i}{\partial \Xi_\tau^{(1)}} = \frac{\omega_i}{2\kappa_{tot}^i} \sum_{e \in S_\tau} R_\tau^e \left( \frac{\partial f_\omega}{\partial \rho_e} \frac{1}{f_\omega} \epsilon_e^i - \frac{1}{\rho_e} \kappa_e^i \right), \quad (6.34)$$

$$\frac{\partial \omega_i}{\partial \Xi_\tau^{(2)}} = \frac{\omega_i}{2\kappa_{tot}^i \Xi_\tau^{(2)}} \sum_{e \in S_\tau} R_\tau^e (\Xi_\tau^{(1)} - \rho_e) \left( \frac{\partial f_\omega}{\partial \rho_e} \frac{1}{f_\omega} \epsilon_e^i - \frac{1}{\rho_e} \kappa_e^i \right). \quad (6.35)$$

### 6.3.3 Discussion on numerical aspects

As stated in section 6.1, the occurrence of spurious modes in low-density regions can be observed also in problems involving free vibrations in TO. The main cause of this phenomenon relies on the penalisation scheme. The global mass matrix is assembled by using a naturally linear penalisation scheme on the element mass matrices. Since changing the penalisation scheme on the mass matrix introduces inconsistencies from a physical viewpoint, the only chance is to set a suitable penalisation formula on the stiffness matrix. If the standard SIMP power law is chosen, the spurious modes issue is not prevented because the stiffness-to-mass ratio goes to zero as  $\rho_e^{\alpha-1}$ : as usually  $\alpha = 3$ ,

the local stiffness goes to zero faster than the local mass and, consequently, the result is a spurious mode involving low density regions of the design domain. Furthermore, this mode is related to a very low natural frequency. Without considering an adequate numerical strategy to prevent this phenomenon, the spurious mode is identified as the first eigen-mode and, therefore, the algorithm assumes the related eigen-frequency as  $\omega_1$ . Of course, this fact flaws and invalidates the TO.

In order to overcome these difficulties, the stiffness matrix is penalised according to the formula suggested by authors in [147]:

$$f_\omega(\rho_e) = 1.1\rho_e^3 - 0.2\rho_e^2 + 0.1\rho_e. \quad (6.36)$$

As it can be observed from Fig. 6.18, the ratio  $f_\omega/\rho_e$  goes to a finite quantity when  $\rho_e$  goes to zero. This fact prevents the occurrence of spurious modes and it allows the topology to correctly evolve during the iterations.

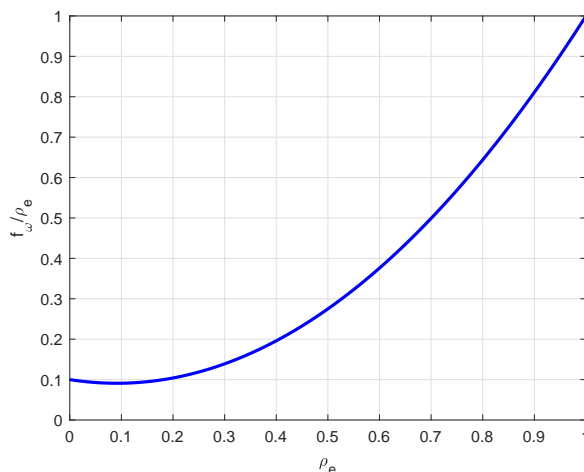


Figure 6.18 – The  $f_\omega/\rho_e$  ratio.

### 6.3.4 Results

#### Benchmark description

The 2D benchmark illustrated in Fig. 6.19 consists of a square plate with a non structural mass  $m_{NS}$  placed at its centre. All the geometric and material parameters are provided in the caption of Fig. 6.19. The natural double symmetry condition with respect to the middle planes is exploited and only a quarter of the plate is modelled in the FE environment with suitable boundary conditions. Moreover, a NDR is set: two material strips forming a cross are identified in the neighbourhood of the symmetry planes in order to avoid isolated portion of material phases and to guarantee for material continuity.

### 6.3. Eigen-frequencies Problems

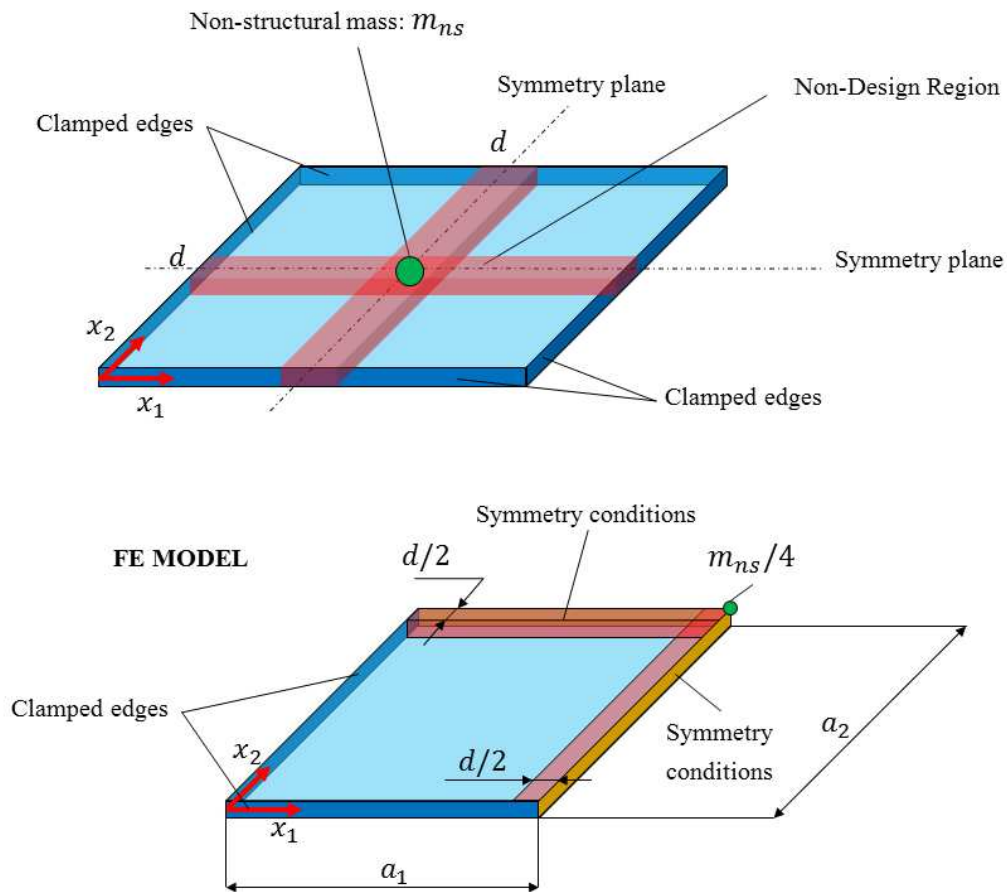


Figure 6.19 – Geometric parameters:  $a_1 = a_2 = 160$  mm,  $d = 20$  mm, thickness  $t = 2$  mm  
 - Material Parameters: Aluminium Alloy, Young's modulus  $E = 72000$  MPa, Poisson's ratio  $\nu = 0.33$  - Non-Structural Mass:  $m_{NS} = 40$  Kg.

The TO problem related to the structure shown in Fig. 6.19 is

$$\begin{aligned}
 & \min_{\Xi^{(1)}, \Xi^{(2)}} - \frac{\omega_1(\rho(\Xi^{(1)}, \Xi^{(2)}))}{\omega_{ref}}, \\
 & \text{subject to:} \\
 & \left\{ \begin{array}{l} ([K_\omega] + \omega_i^2 [M]) \{\psi_i\} = \{0\}, \\ \frac{V(\rho(\Xi^{(1)}, \Xi^{(2)}))}{V_{ref}} = \frac{\sum_{e=1}^{N_e} \rho_e V_e}{V_{ref}} \leq \gamma, \\ \Xi_k^{(1)} \in [10^{-3}, 1], \\ \Xi_k^{(2)} \in [1/2, 10], \\ \forall k = 1, \dots, n_{tot}. \end{array} \right. \quad (6.37)
 \end{aligned}$$

Problem (6.37) aims at maximising the first eigen-frequency of the structure with a constraint on the volume.  $\omega_{ref} = 6.39$  Hz is the natural frequency of the initial configuration, i.e. the inner cross (material phase) with the pseudo-density of the surrounding material  $\rho_e = \gamma$ ,  $\forall e = 1, \dots, N_e$ . For this particular problem it has been chosen  $V_{ref} = 45000$  mm<sup>3</sup> (volume of the Design Region) and  $\gamma = 0.45$ . The mesh is constituted of  $32 \times 32$  SHELL181 Ansys elements.

Similarly to the buckling problem, the aim of this study is to deduce the effect of the B-Spline/NURBS discrete parameters on the optimised solution. TO analyses have been run for all the combinations of the following number of control points and blending functions degrees

- $(n_1 + 1) \times (n_2 + 1) = 10 \times 10, 16 \times 16, 20 \times 20$ ;
- $p_1, p_2 = 2, 3, 4$ .

Once again, the intrinsic CAD compatibility of NURBS entities allows for easily retrieving the 2D boundary of the optimised structure. Therefore, results will be presented on the actual geometry after the postprocessing phase.

### B-Spline and NURBS solutions

Results are shown in Figs. 6.20-6.22 for B-Spline solutions and in Figs. 6.23-6.25 for NURBS solutions.

Numerical results are synthetically reported in the graph of Fig. 6.26.

At a first glance, no relevant trends can be retrieved. It seems that increasing the number of variables by adding control points does not necessarily improve the performances of the structure. On the other hand, as in the case of buckling, increasing the number of variables by including the NURBS weights generally contributes to increase the first eigen-frequency. Unfortunately, this claim does not hold for all the configurations (see, for instance, the B-Spline solution of Fig. 6.21b and the NURBS one of Fig. 6.24b).

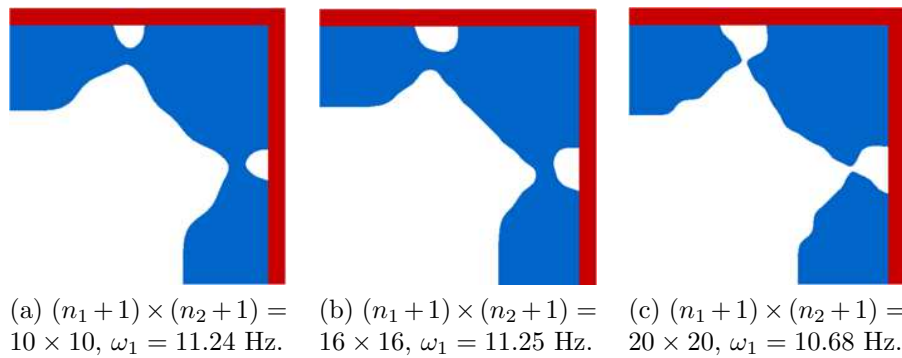


Figure 6.20 – Eigen-frequencies problem: B-Spline results,  $p_1, p_2 = 2$ .

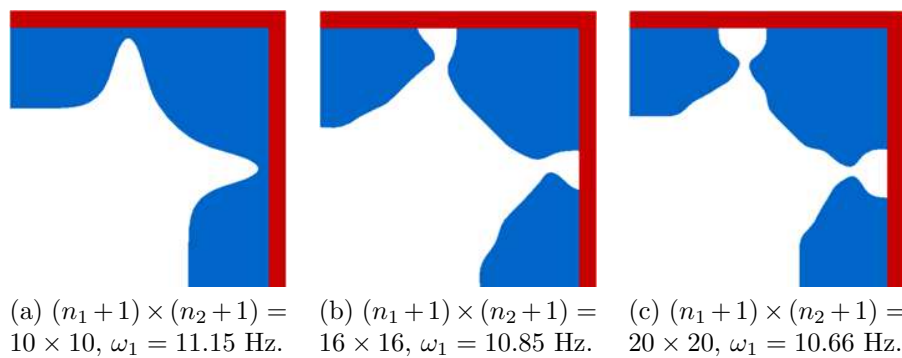


Figure 6.21 – Eigen-frequencies problem: B-Spline results,  $p_1, p_2 = 3$ .

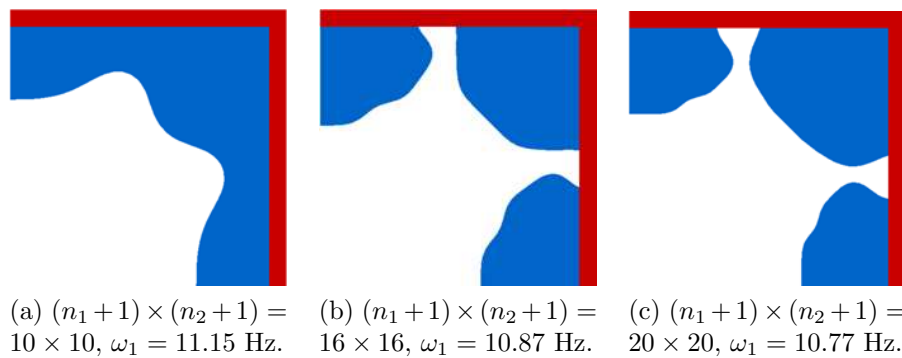


Figure 6.22 – Eigen-frequencies problem: B-Spline results,  $p_1, p_2 = 4$ .

However, it can be noticed that better solutions exhibit a common feature: the optimised geometry has, for each configuration, three main regions of material phase (the biggest one on the top-right corner of the plate and two on the top-left corner and on the bottom-right corner, respectively). The three regions are always bridged, at least, by the NDR. When these zones are linked also by a thin portion of material phase in the DR, the



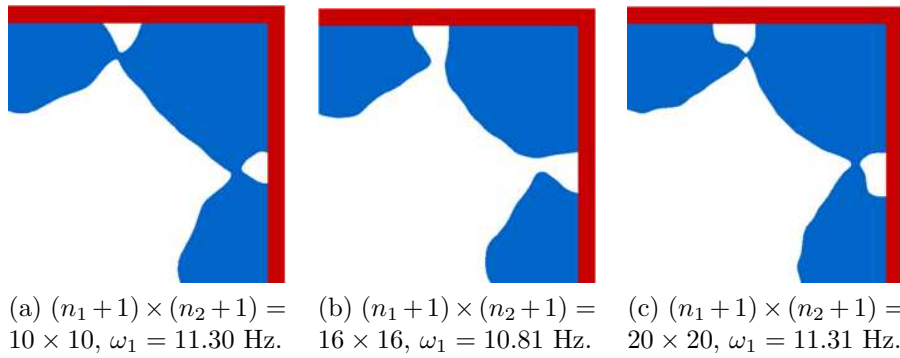


Figure 6.23 – Eigen-frequencies problem: NURBS results,  $p_1, p_2 = 2$ .

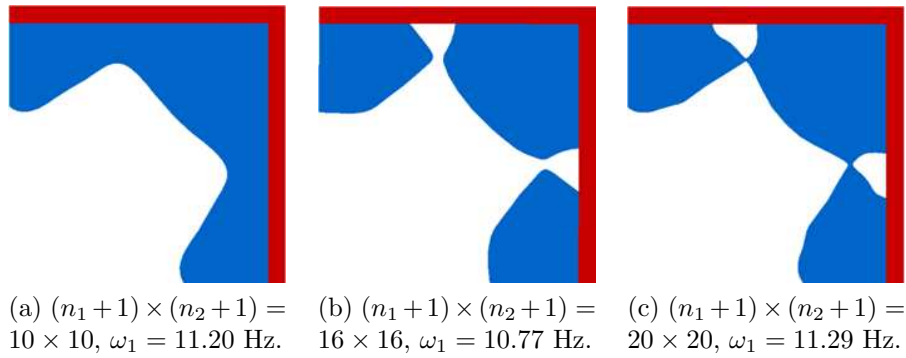


Figure 6.24 – Eigen-frequencies problem: NURBS results,  $p_1, p_2 = 3$ .

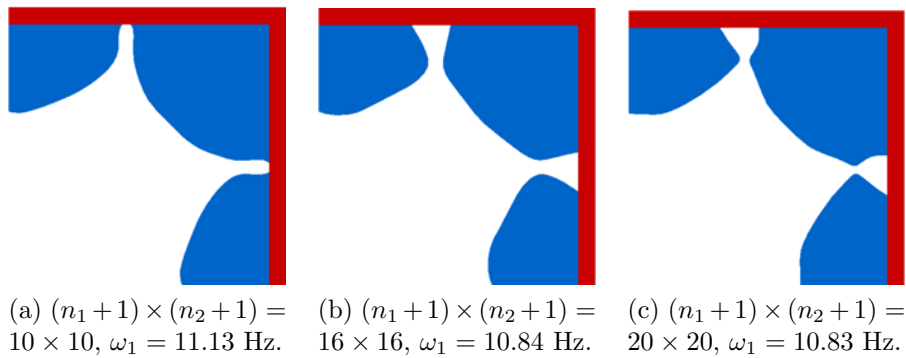


Figure 6.25 – Eigen-frequencies problem: NURBS results,  $p_1, p_2 = 4$ .

stiffness increases and  $\omega_1$  increases accordingly.

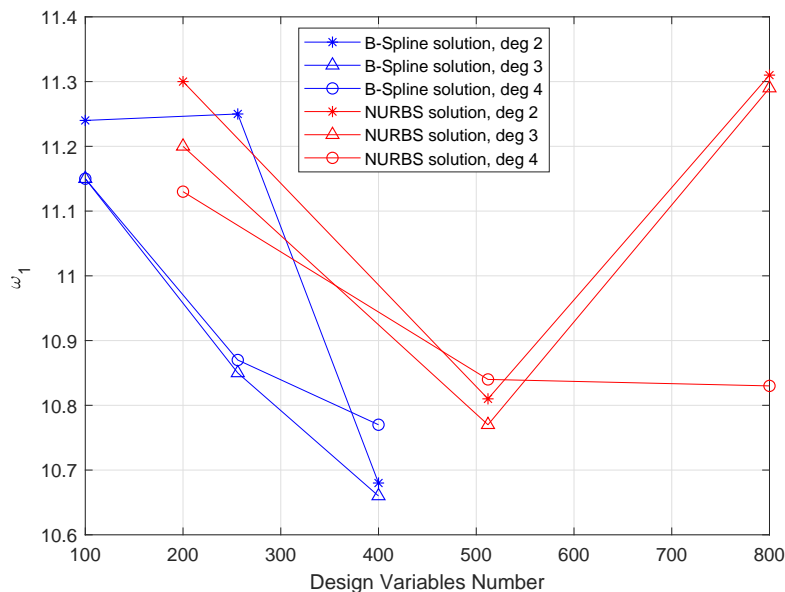


Figure 6.26 – Eigen-frequencies problem: synthesis of results.

## 6.4 Conclusions and Perspectives

This Chapter has shown that eigenvalue problems, dealing with buckling loads and eigen-frequencies, can be easily integrated into the NURBS-based approach introduced in Chapter 4. These two problems present several analogies in both the state equation (eigenvalue problem) and in the numerical issues (spurious modes and mode switching). In both cases, the attention has been focused on the spurious mode occurrence, since the mode switching can be easily handled by setting *ad-hoc* explicit constraints, requiring, for instance, that the second eigenvalue is higher than the first one. For the 2D benchmarks presented in this Chapter, the mode switching was not critical: it has been verified that the second eigen-value is higher enough than the first one in the reassembled structures after the postprocessing phase.

As far as the buckling loads are concerned, two techniques have been presented in order to overcome the spurious modes occurrence when running TO analyses in the NURBS framework. Their effects on the solutions are summarised in the following.

- The double penalisation scheme is a general methodology (eventually, it can be easily generalised to the 3D case) that implies a different penalisation law for the stiffness matrix and for the geometric stiffness matrix.
- The base-thickness strategy is based on the introduction of a thin NDR in the shell structure. Of course, this technique is not general and an equivalent version does not exist in 3D.

It has been shown that, for 2D problems, both B-Spline and NURBS configurations are more efficient when the base-thickness strategy is used.

Moreover, the quality of the solution improves by including the NURBS weights among the design variables, whilst increasing the number of control points does not necessarily increase the first buckling load.

The spurious modes issue has been solved by means of a double penalisation scheme in the case of eigen-frequencies. A suitable penalisation law has been chosen from literature for the stiffness matrix because it assures, in low density regions, a finite stiffness-to-mass ratio. Results have been presented on a simple benchmark. From this single application scheme, it is not possible to deduce a precise behaviour. It can be stated that slight changes in the final topology can affect the results because the distribution of mass and stiffness changes. Moreover, final results of the eigen-frequencies problem seem to be more sensitive to the local pseudo-density gradient.

Some interesting perspectives concerning eigenvalue problems in the NURBS-based SIMP algorithm are issued from this work.

- a) One of the most important numerical issues in the proposed approach is providing the gradient of the considered mechanical quantities. The difficulty does not arise from the NURBS formalism because the chain rule that relates the derivatives with respect to the NURBS parameters to the derivatives with respect to the element densities generally holds. Rather, the true problem to overcome is to provide the derivatives in such a form that they can be computed by using standard quantities, usually available after a FE analysis. Providing the gradient in such a form means making some assumptions, as in the case of the buckling loads, which make the procedure not general enough. Therefore, a more general form of the derivative of the buckling load with respect to the NURBS control points and weights must be developed: the aim is to easily extend the buckling eigenvalue TO to the 3D case. The extension is trivial for the eigen-frequencies problem because the sensitivity analysis takes, in this case, a very general form. Research is ongoing on this topic.
- b) Eigenvalue problems should be extended to more challenging application cases. Firstly, the buckling loads and eigen-frequencies must be considered as constraints and not only as objective function (for instance, in order to overcome the mode switching numerical issue). Secondly, mechanical requirements can involve both stiffness and stability/vibrational requirements in the framework of the TO problem. These coupled problems are very difficult to solve and their solution constitutes an active research axis.
- c) The non-linear formulation in the NURBS framework of the buckling TO problems constitutes an important axis for future researches. Geometric non-linearities and post-buckling behaviour are basic in order to solve more complex problems as

those related to aeronautics, wherein strongly non-linearities can occur (e.g. snap through).

- d) A further development could deal with harmonic analyses. Some quantities related to the steady state response of a structure to a harmonic load can be minimised/maximised or considered in constraints formulation. Eventually, developing suitable tools for taking into account transient analysis in the NURBS-based SIMP approach can be envisaged. In this case, attention should be paid to balance the computational burden of the FE analysis and the penalisation strategy.



# Chapitre 6

## Problèmes aux valeurs propres dans le cadre de l’algorithme d’OT basé sur les NURBS

### 6.1 Introduction

Ce chapitre se concentre sur la formulation de problèmes de valeurs propres dans le cadre de la méthode SIMP basée sur les NURBS. L’implémentation dans l’algorithme *SANTO* est également discutée.

Certains problèmes liés à la formulation des critères de flambage ont été tout d’abord remarqués dans le cadre de l’OT en treillis [125,126]. Des améliorations ont été suggérées dans [127], où les contraintes de stabilité du système et les imperfections de la structure sont prises en compte. La présence de « singular optima » [128] est un problème associé à l’OT en treillis. Le phénomène des optimums singuliers est typique des problèmes de valeurs propres dans les formulations discrètes et continues, empêchant un bon taux de convergence vers l’optimum global. Une discussion exhaustive sur ces aspects peut être trouvée dans [129].

Des problèmes numériques analogues se produisent lorsqu’on traite des algorithmes d’OT basés sur la densité [130]. Pour éviter ce phénomène, il est conseillé de négliger les éléments présentant à la fois une faible valeur de pseudo-densité et une faible contrainte normalisée lors de l’assemblage de la matrice de rigidité géométrique. Cependant, cet artefact numérique pourrait entraîner des changements brusques des fonctions objectif et contraintes. Par conséquent, il en résulte une convergence malaisée [5]. Pour résoudre le problème des faux-modes locaux, les auteurs dans [5] proposent d’utiliser différents schémas de pénalisation pour la matrice de rigidité et pour la matrice de rigidité géométrique, lorsque les problèmes de flambage sont résolus. Un autre problème qui affecte les formulations aux valeurs propres est le « mode switching », qui se produit lorsque, par exemple, la première charge de flambage (ou la première fréquence propre) est maximisée [5]. Dans ce cas, si des précautions spéciales ne sont pas prises, les autres charges de flambage (ou

fréquences propres) et les vecteurs propres associés (les modes) peuvent devenir les plus critiques et la fonction objectif subit un saut, avec des effets néfastes sur la convergence.

Une étude intéressante sur la mise en œuvre du flambage dans un algorithme d'OT basé sur la densité a été réalisée dans [132], où il est souligné que même la méthode proposée dans [5] doit être manipulée avec attention, car les paramètres intervenants ont un impact significatif sur la solution finale. De plus, les complications typiques liées aux charges de flambage disparaissent lorsqu'une formulation «d'épaisseur de base» est adoptée. Imposer une épaisseur de base correspond à augmenter la limite inférieure du champ de pseudo-densité ( $10^{-1}$  au lieu de  $10^{-3}$ ).

Au cours des cinq dernières années, des progrès ont été réalisés sur ce sujet. Dans [135], le problème de la minimisation de volume est formulé dans le cadre des méthodes de programmation binaire et est renforcé par deux contraintes sur la rigidité et la stabilité, respectivement. Dans [136], le problème de flambage en OT est remplacé par un problème d'OT formulé localement en fonction de la minimisation de la compliance. Une étude intéressante a été réalisée par les auteurs de [137], qui appliquent une méthode énergétique pour l'identification des modes de flambage. Cette approche a été approfondie dans [138] : pour obtenir une meilleure convergence, les contraintes de compliance et de flambage devraient être à peine actives pendant tout le processus d'optimisation.

Le problème des fréquences naturelles présente de nombreuses caractéristiques communes avec le problème de la charge de flambage dans l'OT, comme les optimums singuliers, faux-modes locaux et commutation de modes. Ce dernier problème est l'un des plus sensibles dans le contexte des problèmes de fréquences propres. Afin de gérer simultanément plusieurs fréquences propres, de nombreuses techniques différentes sont disponibles dans la littérature. Par exemple, dans [140], certains problèmes d'optimisation avec les fréquences propres sont présentés, comme la maximisation d'une fréquence propre ou la maximisation de la distance de deux fréquences propres à partir d'une valeur prescrite. L'idée de fréquence propre moyenne pondérée est discutée et utilisée dans [141].

Comme dans les problèmes de flambage [132], augmenter la limite inférieure de la pseudo-densité est toujours une pratique efficace pour éviter les faux-modes localisés [142], mais cela invalide l'énoncé du problème d'OT d'origine. Le schéma de pénalisation mutuelle des matrices de rigidité et de masse joue un rôle fondamental dans tout problème dynamique dans l'OT [143,144]. Une autre application, axée sur la maximisation de l'écart entre deux fréquences propres adjacentes pour les structures à deux matériaux, a été discutée dans [145]. Afin d'éviter le phénomène de « mode switching », les auteurs de [146] proposent une analyse de sensibilité basée sur l'analyse de perturbations mathématiques, tandis qu'une pénalisation de matrice de masse appropriée permet de gérer les faux-modes. Les fréquences propres ont également été considérées dans le contexte de problèmes multi-échelle [147]. Une façon intelligente de suivre les modes des valeurs propres (et par conséquent d'empêcher leur commutation) est le Modal Assurance Criterion (MAC) décrit dans [148].

Bien que plusieurs problèmes numériques aient été pris en compte dans la bibliographie sur les problèmes de valeurs propres en OT, la cohérence entre la configuration optimisée et la géométrie réassemblée dans un environnement CAO reste un problème ouvert. Les charges de flambage et les fréquences propres constituent des exigences primordiales, en particulier dans des domaines industriels spécifiques : les charges de flambage figurent parmi les critères de défaillance les plus importants pour les structures aéronautiques. Par conséquent, une conception efficace de structures subissant un phénomène de flambage ou impliquant des vibrations nécessite nécessairement des outils appropriés. Dans ce but, comme cela a été fait précédemment pour la rigidité, les charges de flambage et les fréquences propres sont incluses dans l'algorithme d'OT *SANTO*, afin d'exploiter les avantages de la formulation NURBS.

Le chapitre suit ce schéma. Les paragraphes 6.2 et 6.3 traitent respectivement des problèmes de flambage et les problèmes de fréquences propres. La discussion se limite aux structures 2D dans ce chapitre. Les deux paragraphes 6.2 et 6.3 sont organisés en suivant la même structure : premièrement, le problème d'OT est énoncé dans le cadre SIMP classique et, deuxièmement, la formulation correspondante pour l'OT dans le cadre des NURBS. Une attention particulière est consacrée au calcul des dérivées pour l'algorithme basé sur le gradient lorsqu'un logiciel EF du commerce est utilisé. Ensuite, les problèmes numériques sont discutés et les résultats sont discutés. Enfin, le paragraphe 6.4 termine le chapitre avec des conclusions et des perspectives.

## 6.2 Problèmes de flambage aux valeurs propres

### 6.2.1 Maximisation de la première charge critique de flambage dans la méthode SIMP classique

La relation typique d'un problème de flambage aux valeurs propres dans le cadre des EF est donnée par l'éq. (6.1), où  $[K_\lambda]$  est la matrice de rigidité globale de la structure lorsque le problème de flambage est rencontré,  $\lambda_i$  est la charge de flambage liée à la  $i$ -ème valeur propre,  $[K_\sigma]$  est la soi-disant matrice de rigidité géométrique et  $\{\phi_i\}$  est le  $i$ -em mode de flambage (le vecteur propre des DOFs associés à la  $i$ -ème valeur propre). Les matrices de rigidité et de rigidité géométrique sont liées au champ de pseudo-densité par deux schémas de pénalisation *ad hoc* des éqs. (6.2)-(6.3) :  $[K_e^0]$  et  $[K_{\sigma-e}^0]$  sont les matrices de rigidité et de rigidité géométrique des éléments non pénalisés, toutes deux développées sur l'ensemble de DOFs de la structure.  $f_\lambda(\rho_e)$  et  $f_\sigma(\rho_e)$  représentent les fonctions de pénalisation pour les matrices respectives et elles sont *a priori* différentes.

Un problème typique dans l'OT [5] est la maximisation de la première charge de flambage avec une limite supérieure sur le volume de la structure. Dans le contexte de la méthode SIMP, ce problème se lit comme indiqué dans l'éq. (6.4). Le problème (6.4) peut être résolu par un algorithme approprié basé sur le gradient : les dérivées de la fonction



objectif et des contraintes par rapport aux densités fictives doivent être calculées.

La dérivée du volume est donnée par l'éq. (1.10). Le calcul du gradient des charges de flambage n'est pas une tâche simple. Considérant l'éq. (6.1), on peut déduire l'éq. (6.9).

Avant de continuer, il est nécessaire de clarifier la signification physique des quantités apparaissant dans l'éq. (6.9). Comme on le sait, la matrice de raideur géométrique  $[K_\sigma]$  est assemblée dans l'analyse de "pre-stress", généralement effectuée avant de résoudre le problème des valeurs propres. Dans ce travail, l'algorithme *SANTO* est interfacé avec un logiciel EF commercial et les informations nécessaires à l'assemblage de  $[K_\sigma]$  ne peuvent pas être facilement récupérées. Dans le contexte des problèmes d'instabilité des plaques 2D, une approche énergétique peut être exploitée pour effectuer l'analyse de sensibilité. En conséquence, la matrice  $[K_\sigma]$  n'a pas besoin d'être explicitement calculée. La compliance totale associée au mode de flambage  $i$  est définie par l'éq. (6.10) et le travail virtuel effectué par les charges de précontrainte sur la composante non linéaire des déplacements correspondant à l' $i$ -iem mode est donnée au moyen de l'éq. (6.11). Par conséquent, l'éq. (6.12) peut être déduite.

Il est simple de vérifier que l'éq. (6.9) peut être réorganisé en exploitant les eqs. (6.2) et (6.3). L'éq. (6.13) est donc obtenue. La contribution  $c_{\sigma-e}^i$  mérite une attention particulière et, en particulier, elle doit évaluer la matrice de rigidité des contraintes pour chaque élément de structure. Si l'on considère la simple théorie des plaques de Kirchhoff, la cinématique est complètement déterminée par le champ de déplacement  $(v_1, v_2, v_3)$ , où  $v_1$  et  $v_2$  représentent les composantes du déplacement dans le plan, tandis que  $v_3$  est le composant hors plan. Soient  $N_{x_1}^0, N_{x_2}^0$  et  $N_{x_1x_2}^0$  les efforts linéaires, dont les valeurs sont obtenues comme intégrale sur l'épaisseur de la plaque des composantes de la contrainte  $\sigma_{x_1}^0, \sigma_{x_2}^0$  et  $\sigma_{x_1x_2}^0$ , en raison des charges de précontrainte. Dans ces hypothèses,  $c_\sigma^i$  est calculé par l'éq. (6.14). Par conséquent,  $c_{\sigma-e}^i$  est évalué en limitant l'intégrale précédente à la surface de l'élément  $e$ .

Ainsi, la forme finale de l'analyse de sensibilité de la charge de flambage  $i$  est donnée par l'éq. (6.15).

### 6.2.2 Énoncé mathématique des problèmes de flambage dans l'algorithme d'OT basé sur les NURBS

Le problème de flambage peut être reformulé dans le cadre de la méthode SIMP basée sur les NURBS. Les nouvelles variables de conception sont les points de contrôle et les poids de la NURBS, collectés respectivement dans les vecteurs  $\Xi_1$  et  $\Xi_2$ . L'énoncé mathématique du problème (6.4) dans le cadre des NURBS est donné par l'éq. (6.16).

Afin de résoudre le problème (6.16) par un algorithme basé sur le gradient, les dérivées de la charge de flambage par rapport au nouvel ensemble de variables de conception doivent être calculées. Le gradient d'une fonction spécifique par rapport au point de contrôle  $\hat{\rho}_{I_1, I_2} = \Xi_\tau^{(1)}$  ou le poids  $w_{I_1, I_2} = \Xi_\tau^{(2)}$  peut être facilement calculé en exploitant la

propriété de support local et la règle de chaîne pour les dérivés.

Les dérivées de la charge de flambage  $\lambda_i$  par rapport aux points de contrôle et aux poids de la NURBS peuvent être déduites et exprimées par les éqs. (6.17) et (6.18).

### 6.2.3 Discussion sur les aspects numériques

L'outil numérique utilisé pour résoudre le problème (6.16) est l'algorithme *SANTO* décrit au chapitre 4.

L'attention dans ce paragraphe est concentrée sur les stratégies numériques qui ont été adoptées pour résoudre le problème de flambage des valeurs propres en OT. En particulier, le problème numérique des faux-modes parasites est résolu par deux techniques, à savoir le schéma de double pénalisation [5] et la technique de l'épaisseur de base [132].

La technique de double pénalisation consiste à affecter un schéma de pénalisation différent à la matrice de rigidité et à la matrice de rigidité géométrique. Dans le cadre d'un schéma de double pénalisation, la matrice de rigidité est pénalisée comme montré dans l'éq. (6.19), tandis que la matrice de rigidité géométrique subit le schéma classique de pénalisation SIMP. Le paramètre  $\rho_{low}$  affecte le comportement de la matrice de rigidité par rapport à la matrice de rigidité géométrique lorsque la densité de l'élément prend des valeurs faibles (proches de  $10^{-3}$ ). Une valeur élevée de  $\rho_{low}$  garantit d'éviter les faux-modes mais elle donne une rigidité fictive à la structure. À l'inverse, une valeur trop faible pour  $\rho_{low}$  pourrait amener des résultats trompeurs.

La deuxième méthode pour éviter le problème des faux modes consiste à utiliser une épaisseur de base pour la structure. Dans ce cas, le même schéma de pénalisation pour les matrices de rigidité et de rigidité géométrique peut être utilisé. Une rigidité supplémentaire est obtenue en imposant une mince couche de matériau en vrac à l'intérieur de la plaque à optimiser, dont l'épaisseur est  $t$ . Cette couche interne partage le même plan moyen de la plaque à optimiser et son épaisseur est appelée  $t_{base}$  (voir la fig. 6.2).

### 6.2.4 Résultats

#### Description du benchmark

Dans ce paragraphe, un benchmark significatif est considéré pour comparer l'efficacité des techniques proposées ci-dessus lorsqu'elles sont formulées dans le cadre de la méthode SIMP basée sur les NURBS. Le benchmark est illustré à la fig. 6.3. Cette structure consiste en un cadre extérieur à renforcer afin d'augmenter sa charge critique au flambage. Ce benchmark a été pris de [132] mais, contrairement au choix fait dans [132], la structure permet des modes de flambement naturels hors plan.

La formulation mathématique du problème à résoudre est indiquée dans l'éq. (6.21). Dans le problème (6.21),  $\lambda_{ref}$  est la charge de flambage du cadre externe renforcée par la région interne avec  $\rho_e = \gamma$ ,  $\forall e = 1, \dots, N_e$ . Cela correspond à la solution faisable fournie

pour l'initialisation de l'algorithme. Le volume de référence est le volume de la design region, c'est-à-dire  $V_{ref} = 457600 \text{ mm}^3$ . Pour ce benchmark,  $\gamma = 0.4$ . Le maillage est composé de  $90 \times 30$  éléments SHELL181 pour toutes les analyses.

Les valeurs suivantes des paramètres NURBS / BSpline ont été utilisées pour effectuer des tests numériques :

- $(n_1 + 1) \times (n_2 + 1) = 42 \times 14, 60 \times 20, 78 \times 26$  ;
- $p_1, p_2 = 2, 3, 4$ .

Concernant les paramètres de pénalisation, ils ont été choisis comme  $\rho_{low} = 10^{-3}$  et  $t_{base} = t/30$  : utiliser ces valeurs implique une charge de flambage de référence  $\lambda_{ref} = 11,5 \text{ N}$  pour le schéma de double pénalisation et pour la technique d'épaisseur de base.

Les résultats seront présentés sous leur forme définitive, c'est-à-dire après la phase de post-traitement. Par conséquent, les résultats numériques concernant les charges de flambage ont été obtenus en effectuant une analyse aux valeurs propres sur la géométrie réassemblée réelle.

### Solutions B-Spline et NURBS obtenues par le schéma de double pénalisation

Toutes les géométries présentées ci-dessous répondent à la contrainte de volume. La charge de flambage correspondante est directement rapportée dans les légendes des figs. 6.4-6.6 pour les solutions B-Spline et des figs. 6.7-6.9 pour les solutions NURBS.

Les résultats numériques sont résumés dans le graphique de la fig. 6.10.

Certaines remarques découlent de l'analyse des topologies obtenues et des valeurs numériques des charges de flambage critiques.

- Le système de double pénalisation semble être efficace et évite l'apparition de modes parasites. Cette technique est générale et pourrait être facilement étendue au cas de flambage pour les applications d'OT 3D.
- Les solutions NURBS présentent de meilleures performances que les solutions B-Spline. Dans le cas du flambage, l'avantage d'inclure les poids NURBS parmi les variables de conception est plus évident : une frontière plus lisse implique une augmentation significative de la charge de flambage.
- Augmenter le nombre de points de contrôle n'est pas, en général, le meilleur choix pour obtenir des solutions plus efficaces, comme la fig. 6.10 le montre.

### Solutions B-Spline et NURBS obtenues par la stratégie de l'épaisseur de base

La même campagne numérique a été répétée dans le cas de la stratégie d'épaisseur de base. Les résultats sont directement fournis aux figs. 6.11-6.16.

Une synthèse des résultats numériques est présentée dans le graphique de la fig. 6.17.

En utilisant la technique d'épaisseur de base, le même comportement de la fig. 6.10 peut être observé à nouveau : augmenter le nombre de points de contrôle n'implique pas nécessairement de meilleures performances. Au contraire, l'utilisation de NURBS est fortement recommandée pour obtenir des géométries plus lisses et des configurations plus efficaces.

Si les résultats numériques sont comparés (voir les figs. 6.10 et 6.17), la technique d'épaisseur de base semble, en général, fournir des solutions plus efficaces lorsque le même nombre de points de contrôle et les mêmes degrés sont utilisés. Ce fait se produit aussi bien pour les surfaces B-Spline que pour les NURBS. Cependant, il convient de noter que le système de double pénalisation est général et pourrait éventuellement être appliqué à n'importe quelle structure. La stratégie d'épaisseur de base ne s'applique que dans le cas de plaques et de coques et n'est pas générale.

## 6.3 Problèmes de fréquences propres

### 6.3.1 Maximisation de la première fréquence propre dans la méthode SIMP classique

Si l'on considère le problème des fréquences propres, sa formulation en OT est analogue à celle du problème (6.4). Dans ce cas, l'équation d'état est donnée par l'éq. (6.22), où  $[K_\omega]$  est la matrice de rigidité globale de la structure lorsque le problème des fréquences propres est abordé,  $\omega_i$  est la  $i$ -ème fréquence propre,  $[M]$  est la matrice de masse globale, tandis que  $\{\psi_i\}$  est le  $i$ -ème mode (vecteur propre des DOF) lié à la fréquence propre correspondante. Dans ce cadre particulier, la matrice de rigidité globale est liée à la matrice de rigidité des éléments non pénalisés  $[K_e^0]$  par l'éq. (6.23), où  $f_\omega(\rho_e)$  est un schéma de pénalisation dédié. Bien que d'autres méthodes aient été développées dans la littérature, le schéma de pénalisation de la matrice de masse a été supposé naturellement linéaire dans ce travail (voir l'éq. (6.24)). Il est à noter que  $[M_e^0]$  est la matrice de masse des éléments non pénalisés, tandis que  $[M_l]$  est la matrice qui prend en compte les masses non structurelles localisées, c.-à-d. les masses qui ne contribuent pas à la matrice de rigidité globale.

Le problème des fréquences propres considéré dans ce travail est la maximisation de la première fréquence propre avec une contrainte d'inégalité de volume, comme montré dans l'éq. (6.25) : bien sûr, pour traiter un problème bien posé, il est supposé que des NDRs appropriées ou une masse non structurelle sont présents.

En ce qui concerne l'analyse de sensibilité des fréquences propres, des étapes analogues au cas du flambage peuvent être répétées. Particulièrement, à partir de l'éq. (6.22), on peut déduire l'éq. (6.32). Dans l'éq. (6.32),  $c_{tot}^i$  est le travail virtuel des forces internes (le double de l'énergie de déformation) et  $\kappa_{tot}^i$  l'énergie cinétique lorsque la structure vibre à  $\omega_i$ .

Il est à noter que la forme de la dérivée dans l'éq. (6.32), contrairement au cas du flambage, est générale et ne nécessite aucune autre hypothèse à appliquer.

### 6.3.2 Enoncé mathématique des problèmes de fréquences propres dans l'algorithme d'OT basé sur les NURBS

Dans ce paragraphe, le problème (6.25) est reformulé dans le cadre d'approche SIMP basée sur les NURBS. En particulier, la dépendance des quantités mécaniques / géométriques des variables des NURBS (c'est-à-dire la pseudo-densité aux points de contrôle et les poids associés) est soulignée.

L'énoncé mathématique du problème (6.25) dans le cadre NURBS est donné par l'éq. (6.33)

Dans l'éq. (6.33),  $\omega_{ref}$  est une fréquence de référence, définie en fonction du problème rencontré.

Les dérivées des fréquences propres par rapport aux points de contrôle et aux poids NURBS sont faciles à calculer. Les eqs. (6.34)-(6.35) peuvent être facilement dérivées.

### 6.3.3 Discussion sur les aspects numériques

L'occurrence de faux modes dans les régions à faible densité peut être observée également dans les problèmes impliquant des vibrations libres dans l'OT. La principale cause de ce phénomène repose sur le schéma de pénalisation. Puisque l'utilisation d'un schéma de pénalisation non linéaire sur la matrice de masse introduit des incohérences d'un point de vue physique, la seule possibilité est de définir une formule de pénalisation adaptée à la matrice de rigidité. Si la loi de puissance pour la méthode SIMP est choisie, le problème des faux modes n'est pas résolu, car le rapport rigidité / masse approche zéro comme  $\rho_e^{\alpha-1}$  : donc, localement, la rigidité s'annule plus rapidement que la masse et, par conséquent, le résultat est un faux mode impliquant des régions de faible densité pour le domaine de conception.

Afin de surmonter ces difficultés, la matrice de rigidité est pénalisée selon l'éq. (6.36) [147].

Comme on peut l'observer de la fig. 6.18, le rapport  $f_\omega/\rho_e$  devient fini lorsque  $\rho_e$  s'approche à zéro. Ce phénomène empêche l'apparition de faux modes et permet à la topologie d'évoluer correctement pendant les itérations.

### 6.3.4 Résultats

#### Description du benchmark

Le benchmark 2D illustré à la fig. 6.19 consiste en une plaque carrée avec une masse non structurelle  $m_{NS}$  placée en son centre. La condition de double symétrie naturelle par rapport aux plans médians est exploitée et seul un quart de la plaque est modélisé dans

l'environnement EF avec des conditions aux limites appropriées. De plus, une NDR est établie : deux bandes de matériau formant une croix sont identifiées au voisinage des plans de symétrie afin d'éviter une partie isolée des phases matérielles et de garantir la continuité des matériaux.

Le problème d'OT lié à la structure représentée à la fig. 6.19 est donné par l'éq. (6.37).

Le problème (6.37) vise à maximiser la première fréquence propre de la structure avec une contrainte sur le volume.  $\omega_{ref} = 6.39$  Hz est la fréquence propre de la configuration initiale, c'est-à-dire la croix intérieure (phase matérielle) avec la pseudo-densité du matériau environnant  $\rho_e = \gamma$ ,  $\forall e = 1, \dots, N_e$ . Pour ce problème particulier, il a été choisi  $V_{ref} = 45000 \text{ mm}^3$  (volume de la DR) et  $\gamma = 0.45$ . Le maillage est constitué de  $32 \times 32$  éléments SHELL181.

Les analyses d'OT ont été exécutées pour toutes les combinaisons du nombre de points de contrôle et de degrés des fonctions de base NURBS suivants :

- $(n_1 + 1) \times (n_2 + 1) = 10 \times 10, 16 \times 16, 20 \times 20$  ;
- $p_1, p_2 = 2, 3, 4$ .

Encore une fois, la compatibilité CAO intrinsèque des entités NURBS permet de récupérer facilement la frontière 2D de la structure optimisée. Par conséquent, les résultats seront présentés sur la géométrie réelle après la phase de post-traitement.

## Solutions B-Spline et NURBS

Les résultats sont présentés aux figs. 6.20-6.22 pour les solutions B-Spline et aux figs. 6.23-6.25 pour les solutions NURBS .

Les résultats numériques sont synthétisés dans le graphique de la fig. 6.26.

À première vue, aucune tendance pertinente ne peut être récupérée. Il semble que l'augmentation du nombre de variables en ajoutant des points de contrôle n'améliore pas nécessairement les performances de la structure. En revanche, comme dans le cas du flambage, l'augmentation du nombre de variables en incluant les poids NURBS contribue généralement à augmenter la première fréquence propre. Malheureusement, cette affirmation n'est pas valable pour toutes les configurations (voir, par exemple, la solution B-Spline de la fig. 6.21b et celle NURBS de la fig. 6.24b). Cependant, on peut noter que les meilleures solutions présentent une caractéristique commune : la géométrie optimisée a, pour chaque configuration, trois régions principales de la phase matérielle. Les trois régions sont toujours reliées, au moins, par la NDR. Lorsque ces zones sont également liées par une faible partie de la phase matérielle dans la DR, la rigidité augmente et  $\omega_1$  augmente en conséquence.

## 6.4 Conclusions et Perspectives

Ce chapitre a montré que les problèmes de valeurs propres, tels que les analyses de flambage et de fréquences naturelles, peuvent être facilement intégrés dans l'approche basée sur les NURBS introduite dans le chapitre 4. Dans les deux cas, l'attention s'est focalisée sur l'occurrence de faux modes, car la commutation de mode peut être facilement gérée en définissant des contraintes explicites.

En ce qui concerne les charges de flambage, deux techniques ont été présentées afin de surmonter les faux modes lors de l'exécution d'analyses d'OT basée sur les NURBS. Le schéma de double pénalisation est une méthodologie générale qui implique une loi de pénalisation différente pour la matrice de rigidité et pour la matrice de rigidité géométrique. La stratégie d'épaisseur de base repose sur l'introduction d'une NDR mince dans la structure de la coque. Bien entendu, cette technique n'est pas générale et une version équivalente n'existe pas en 3D. Il a été démontré que, pour les problèmes 2D, les configurations B-Spline et NURBS sont plus efficaces lorsque la stratégie d'épaisseur de base est utilisée. De plus, la qualité de la solution s'améliore en incluant les poids NURBS parmi les variables de conception, tandis que l'augmentation du nombre de points de contrôle n'augmente pas nécessairement la première charge de flambage.

La question des modes parasites a été résolue au moyen d'un système de double pénalisation dans le cas des fréquences propres. Une loi de pénalisation appropriée a été choisie parmi la littérature pour la matrice de rigidité, car elle assure, dans les régions de faible densité, un rapport rigidité-masse fini. Les résultats ont été présentés sur un benchmark simple. À partir de seulement ce cas d'étude, il n'est pas possible de déduire un comportement précis. On peut dire que de légères modifications de la topologie finale peuvent affecter les résultats car la distribution de la masse et de la rigidité change. De plus, les résultats finaux du problème des fréquences propres semblent être plus sensibles au gradient de pseudo-densité local.

Des perspectives intéressantes concernant les problèmes aux valeurs propres dans l'algorithme SIMP basé sur les NURBS sont issues de ce travail.

- a) L'un des problèmes numériques les plus importants dans l'approche proposée est de fournir le gradient des grandeurs mécaniques considérées. Le véritable problème à résoudre consiste à fournir les dérivées sous une forme telle qu'elles puissent être calculées en utilisant des quantités standard, généralement disponibles après une analyse EF. Fournir le gradient sous une telle forme signifie faire des hypothèses, comme dans le cas des charges de flambage, qui rendent la procédure pas assez générale. Par conséquent, une forme plus générale de la dérivée de la charge de flambage par rapport aux points de contrôle et aux poids de la NURBS doit être développée. Des recherches sont en cours sur ce sujet.
- b) Les problèmes de valeurs propres devraient être étendus aux cas d'application plus complexes. Les applications mécaniques peuvent impliquer des exigences de rigidité

et de stabilité / vibration dans le cadre des problèmes d'OT. Ces problèmes couplés sont très difficiles à résoudre et leur solution constitue un axe de recherche actif.

- c) La formulation non linéaire dans le cadre NURBS des problèmes de flambage en OT constitue un axe important pour les recherches futures. Les non-linéarités géométriques et le comportement post-flambage sont fondamentaux pour résoudre des problèmes plus complexes liés à l'aéronautique, dans lesquels des non-linéarités importantes peuvent se produire (par exemple, le snap-through).
- d) Un développement ultérieur pourrait porter sur les analyses harmoniques. Certaines quantités liées à la réponse en régime permanent d'une structure à une charge harmonique peuvent être minimisées / maximisées ou prises en compte dans la formulation des contraintes.





# Chapter 7

## General Fitting Techniques for Curves and Surfaces Reconstruction

### 7.1 Introduction to Curve and Surface Reconstruction

Curve and surface reconstruction is a widely studied topic in informatics, geometric modelling and reverse engineering. In the case of curve reconstruction, the goal is to find all the parameters which uniquely identify a parametric curve approximating a set of data points, i.e. the *target points* (TPs). The curve fitting problem can be stated as a classical least squares problem, wherein the Euclidean distance between TPs and a set of suitable points belonging to the curve is minimised. Similarly, the problem of surface reconstruction consists of identifying the parameters that synthetically define a surface approximating the TPs. Usually, the starting data for curve reconstruction are just the set of TPs coordinates. On the other hand, the available data for surface reconstruction are not necessarily restrained to the TPs coordinates, but they can be provided in the form of a *Delaunay triangulation* or in “mixed form”: in the former case, TPs constitute the vertices of triangles and the information about the local normal vector can be exploited; in the latter case, some curves can be assigned as sections of the surface, which is supposed to approximate the cloud of TPs.

In general, the curve/surface reconstruction is composed of two main steps: *parametrisation* and *fitting*. A proper parametrisation is established on the TPs clouds when the relationship between neighbour points in the parametric domain reflects the shape of the curve or the topology of the surface in the physical 3D space. Instead, the fitting task identifies all the mathematical-programming operations needed in order to solve the least square optimisation problem, related to the specific curve/surface reconstruction.

In the case of curve/surface fitting (or approximation) problems, a significant amount of TPs is available. These TPs do not necessarily satisfy the equation of the fitting

geometric entity, which is available in a suitable parametric form at the end of the optimisation (implicit or explicit). On the other hand, *interpolation* problems aim at finding a parametric geometric entity passing through the set of TPs (i.e. TPs exactly satisfy the parametric entity equation). This last class of problems is not considered in this Chapter.

Parametrisation is a straightforward task in the case of curves, but it constitutes a really hard problem in the case of surfaces. Several techniques can be found in literature for parametrisation. One of the most difficult aspects of parametrisation (or mapping) is that the solution is not unique: a given geometric entity can be represented through several parametrisations. When the approximating surface is an open surface, Floater's shape preserving strategy can be employed [111]. As far as *genus* zero surfaces (i.e. simple closed surfaces without holes) are concerned, the method described in [150] relates each TP of the physical space to a point belonging to a unit radius sphere in the parametric space. An extensive discussion on more complicated topologies (of genus greater than or equal to one) can be found in [110, 151], where global and conformal parametrisations are provided for non-trivial surfaces: in these works, some mathematical properties of gradient fields of conformal mapping (harmony, conjugacy, duality and symmetry) are exploited. The parametrisation can be established on the initial data independently from the fitting problem.

Standard gradient-based optimisation methods (refer to Chapter 3) have been broadly employed in order to solve the curve fitting problem in its classical least squares formulation [152–154]. In particular, in [152] and [154], the formulation of the objective function was modified by introducing the *tangent distance minimisation method* and the *square distance minimisation method*. The most relevant contributions of these techniques are the improvement of the convergence rate and the stability of the solution. Ueng *et al.* [153] enhance the objective function by inserting information about tangent and curvature of the approximating curve as weighted quantities. However, weight parameters must be carefully tuned *a priori* by the designer in [153]: accordingly, their definition is problem-dependent. Analogue strategies have been employed in the surface fitting problem [68, 155]. Some approaches have been proposed in order to regularise sets of TPs in the 3D Euclidean space by defining a curvature-based functional to be minimised [156]: these techniques are very useful to circumvent the effect of noise on data points and provide a smoothing effect, but they suffer from the same shortcomings of the methods discussed in [152–154] for curve fitting.

Several methodologies deal with the curve/surface fitting problem in the framework of NURBS geometric entities (refer to Chapter 2). Performing a fitting by means of a NURBS entity is particularly advantageous because it is completely CAD-compatible, i.e. its parameters can be transferred through standard format files to CAD software. However, the considerable quantity of parameters defining a NURBS entity also constitutes the main drawback: it is very hard to properly tune all parameters defining its shape. In the last three decades, the massive development of meta-heuristic procedures has brought

engineers to apply such strategies in the framework of the curve/surface fitting problem. Meta-heuristics can intervene at different levels in solving the curve/surface fitting problem. For instance, Li *et al.* [157] present a preprocessing method, based on the discrete evaluation of the curvature, to provide a starting B-Spline knot vector, which reflects the shape of the curve to be approximated. In this case, the meta-heuristic is used to overcome a difficult task characterising gradient-based strategies, i.e. providing the initial guess (initialisation). Alternatively, meta-heuristics can be used directly to solve the curve/surface fitting problem stated as a least squares optimisation problem. Limaiem *et al.* [158] make use of a GA to find the optimum value of the parameters defining the approximating curve. Kang *et al.* [159] use a sparse optimisation to iteratively update the knot vector length and components of the approximating B-Spline curve. In [96], a particle swarm optimisation algorithm has been employed to approximate the TPs by means of B-Spline surfaces. The same authors propose a two-steps GA to choose data points parametrisation and knot vectors to perform surface fitting via B-Spline entities in [160]. Interesting suggestions are provided in [161], where some stability requirements are imposed on the final position of control points of the approximating NURBS surface. Recently, Garcia-Capulin *et al.* [162] employed a Hierarchical GA to optimise both the number and the value of the knots of a B-Spline curve. However, the approach presented in [162] is based on the resolution of a bi-objective unconstrained optimisation problem that needs the definition of a “fictitious” objective function to economise the number of knots, which is not related to any geometrical requirement. Moreover, the degree of the basis functions is kept constant in [162] and the problem is not stated in the more general framework of NURBS curves.

As it can be easily deduced from this (non-exhaustive) state of the art on curve/surface fitting in the mathematical framework of NURBS entities, the main limitations and drawbacks characterising the vast majority of the studies on this topic are essentially two:

- the lack of a proper problem formulation (without considering arbitrary penalisation coefficients, which must be defined by the user and that are problem-dependent);
- the lack of a very general numerical strategy, able to simultaneously optimise the *number* as well as the *value* of the constitutive parameters (i.e. the *design variables*) defining the shape of the NURBS curve/surface.

To overcome the previous restrictions, in this Chapter, an innovative approach to the curve fitting problem is proposed. A new formulation of the mathematical problem has been developed: the curve fitting problem is stated as a CNLPP by introducing a constraint on the maximum value of the curvature. In this study, the curve fitting problem is solved in the framework of NURBS curves. The main idea is to keep all the parameters defining the NURBS curve as design variables in order to state the curve fitting problem in the most general sense. Nevertheless, this fact implies some consequences of paramount importance, constituting just as many difficulties in solving the related CNLPP.

- When the curve fitting problem is formulated by including the number of control points and the degree of the basis functions among the unknowns, the *overall number* of design variables (i.e. the overall number of parameters defining the shape of the curve) for the problem at hand is not *a priori* set: hence, the resulting CNLPP is defined over a search space of variable dimension.
- The optimisation variables of the CNLPP are of different nature (continuous and discrete).
- The numerical strategy chosen to face such a problem must be able to handle design variables of different nature and to optimise, at the same time, the dimension of the design domain as well as the value of each constitutive parameter of the NURBS curve.

This kind of problems is referred as optimisation of “modular systems” in bibliography [72, 163, 164]. Here, the numerical strategy considered for the solution search of CNLPP of modular systems is based on the MATLAB version of the GA BIANCA, presented in Chapter 3.

Furthermore, the surface reconstruction problem has been faced as well. After a brief discussion on parametrisation strategies, a two-steps gradient-based algorithm for surface fitting is proposed. The number of control points and the degrees of the blending functions are set at the beginning of the optimisation and they are not considered design variables. In the first step, the least squares problem for surface fitting is solved by using a B-Spline surface and only its control points coordinates are considered design variables. In the second step, control points coordinates are retrieved from the first step and they do not change any more: now, a NURBS surface is considered and the new design variables for the second optimisation step are the weights. Therefore, the final solution is provided in the form of a NURBS surface approximating the TPs cloud.

This Chapter is outlined as follows. The innovative general strategy to solve the curve fitting problem is described in section 7.2: the mathematical formulation and the related numerical strategy are deeply discussed. Results are shown on benchmarks of different difficulty and on a more challenging real-world engineering problem. Section 7.3 focuses on the preliminary developments on the surface reconstruction: the issues related to the generalisation of the curve fitting problem to the case of surfaces are identified and possible solutions are discussed. Finally, section 7.4 ends the Chapter with conclusions and future perspectives.

## 7.2 A General Hybrid Optimisation Strategy for Curve Fitting in the NURBS Framework

### 7.2.1 Mathematical Formulation of the Curve Fitting Problem

In this section, the curve fitting problem is stated as a CNLPP and it is formulated in the most general case, i.e. by considering the full-set of design variables describing the shape of the parametric curve.

Let us consider the classical form of the curve fitting problem, namely

$$\min_{\xi} f(\xi), \quad f = \sum_{k=0}^{\mu} \|\mathbf{C}(u_1^k) - \mathbf{Q}_k\|^2. \quad (7.1)$$

In Eq. (7.1),  $(\mu+1)$  is the number of TPs,  $\mathbf{Q}_k$  the generic  $k$ -th point,  $\mathbf{Q}_k = \{\bar{x}_1^k, \bar{x}_2^k, \bar{x}_3^k\}$  are the Cartesian coordinates of the TPs, while  $\mathbf{C}(u_1^k) = \{C_{x_1}(u_1^k), C_{x_2}(u_1^k), C_{x_3}(u_1^k)\}$  are their counterpart belonging to the parametric curve when the dimensionless parameter  $u_1$  gets the value  $u_1^k$ . In the same equation, vector  $\xi$  collects all the optimisation variables, i.e. the full set of parameters (of different nature) defining the shape of the curve. In the most general case, when the parametric curve of Eq. (7.1) is represented in the mathematical framework of NURBS basis functions, its shape depends upon the following parameters:

- *discrete parameters*, i.e. the number of control points  $n_1 + 1$ , the number of knots  $m_1 + 1$  and the degree of the blending functions  $p_1$ ;
- *continuous parameters*, namely the non-decreasing sequence of components of the knot vector  $U_j^{(1)}$ ,  $j \in [p_1 + 1, m_1 - p_1 - 1]$ , the coordinates of the control points  $\mathbf{P}_{i_1} = \{X_{i_1}^1, X_{i_1}^2, X_{i_1}^3\}$ ,  $i_1 \in [0, n_1]$ , the weights values  $w_{i_1}$ ,  $i_1 \in [0, n_1]$  and the set of suitable values of the dimensionless parameter of the curve  $u_1^k$ ,  $k \in [0, \mu]$ .

Firstly, let us consider the integer parameters: Eq. (2.6) gives the relationship among  $m_1$ ,  $p_1$  and  $n_1$ . In standard approaches [152–154, 157], the maximum control point index  $n_1$  is fixed *a priori*, while the value of  $p_1$  is chosen as a compromise between accuracy and noise introduction. Accordingly, the maximum index of the knot vector components is deduced. Unlike standard approaches, no assumptions are made on the integer parameters of a NURBS curve in this work. In particular,  $m_1$  and  $p_1$  are included into the vector of design variables, whilst  $n_1$  will be calculated according to Eq. (2.6).

Secondly, let us consider the set of continuous parameters. The  $u_1^k$  values of the curve dimensionless parameter are calculated through the *chord length method* [68], so they are no longer design variables. In particular, the *chord length*  $L_{TP}$  of the curve can be defined in terms of Euclidean distance among consecutive TPs,

$$L_{TP} = \sum_{k=0}^{\mu-1} \|\mathbf{Q}_{k+1} - \mathbf{Q}_k\|. \quad (7.2)$$

Assumed that  $u_1^0 = 0$  and  $u_1^\mu = 1$ , the general parameter  $u_1^k$  can be computed through

$$u_1^{k+1} = u_1^k + \frac{\|\mathbf{Q}_{k+1} - \mathbf{Q}_k\|}{L_{TP}}, \quad k = 0, \dots, \mu - 2. \quad (7.3)$$

The chord length formula is a possible parametrisation for curve reconstruction problem. For more details, the interested reader is addressed to [68].

Moreover, the optimum value of the control points coordinates can be obtained through the analytical approach of Ueng *et al.* [153]. Let  $\mathbf{X}_P^1, \mathbf{X}_P^2, \mathbf{X}_P^3 \in \mathbb{R}^{n_1+1}$  be column vectors collecting the  $x_1, x_2$  and  $x_3$  coordinates of the control points and  $\bar{\mathbf{X}}_Q^1, \bar{\mathbf{X}}_Q^2, \bar{\mathbf{X}}_Q^3 \in \mathbb{R}^{\mu+1}$  the counterparts for TPs. Furthermore, matrix  $[\mathbf{A}] \in \mathbb{R}^{(\mu+1) \times (n_1+1)}$  can be defined as

$$A_{k,i_1} = R_{i_1,p_1}(u_1^k), \quad k = 0, \dots, \mu + 1, \quad i_1 = 0, \dots, n_1, \quad (7.4)$$

and matrix  $[\mathbf{B}] \in \mathbb{R}^{(n_1+1) \times (n_1+1)}$  as

$$[\mathbf{B}] = ([\mathbf{A}]^T[\mathbf{A}])^{-1}. \quad (7.5)$$

Therefore, the following proposition applies.

**Proposition 7.2.1** *For a NURBS curve of assigned degree  $p_1$ , number of control points  $(n_1 + 1)$ , knot vector  $\mathbf{U}^{(1)}$  and weights  $w_{i_1}$  ( $i_1 = 0, \dots, n_1$ ), the control point coordinates minimising the cost function  $f$  of problem (7.1) are*

$$\mathbf{X}_P^j = [\mathbf{B}][\mathbf{A}]^T \bar{\mathbf{X}}_Q^j, \quad j = 1, 2, 3. \quad (7.6)$$

*Proof.* The proof is provided here for the coordinate  $x_1$  and can be easily extended to other coordinates. Since the objective function  $f$  is convex (in terms of control points coordinates), a necessary and sufficient condition for getting the minimum is

$$\frac{\partial f}{\partial X_l^1} = 0, \quad \forall l = 0, \dots, n_1. \quad (7.7)$$

After few simple passages, the previous relationship can be written as

$$\begin{aligned} \sum_{k=0}^{\mu} \left[ 2 (C_{x_1}(u_1^k) - \bar{x}_1^k) \frac{\partial C_{x_1}(u_1^k)}{\partial X_l^1} \right] &= 0, \\ \sum_{k=0}^{\mu} \left[ \left( \sum_{i_1=0}^{n_1} R_{i_1,p_1}(u_1^k) X_{i_1}^1 - \bar{x}_1^k \right) R_{l,p_1}(u_1^k) \right] &= 0, \\ \sum_{k=0}^{\mu} \sum_{i_1=0}^{n_1} R_{l,p_1}(u_1^k) R_{i_1,p_1}(u_1^k) X_{i_1}^1 &= \sum_{k=0}^{\mu} R_{l,p_1}(u_1^k) \bar{x}_1^k, \quad \forall l = 0, \dots, n_1 \end{aligned} \quad (7.8)$$

The last relation of Eq. (7.8) must be satisfied for each control point and can be stated in a more compact form:

$$[\mathbf{A}]^T[\mathbf{A}]\mathbf{X}_P = [\mathbf{A}]^T\bar{\mathbf{X}}_Q. \quad (7.9)$$

Finally, the inversion of matrix  $([\mathbf{A}]^T[\mathbf{A}])$  allows for obtaining the vector  $\mathbf{X}_P$ . ■

It is noteworthy that matrix  $([\mathbf{A}]^T[\mathbf{A}])$  could have some almost null eigenvalue, so its inversion could be ill-conditioned. In this Chapter, the inversion has been performed by means of Moore-Penrose's pseudo-inverse matrix [153], in order to overcome this issue.

A quick glance to Eqs. (7.4)-(7.6) suffices to deduce that the Cartesian coordinates of the control points are affected by the other parameters of the NURBS curve, so they are no longer design variables but they can be interpreted as derived quantities. More precisely, matrix  $[\mathbf{A}]$  depends upon the NURBS blending functions, hence its terms depend on the value of both integer and continuous variables, i.e.  $m_1, p_1, U_j^{(1)}$  and  $w_{i_1}$ , as well as on the  $u_1^k$  values. As a consequence of the previous considerations, design variables can be ranged in two vectors  $\boldsymbol{\xi}_1$  and  $\boldsymbol{\xi}_2$ :

- $\boldsymbol{\xi}_1$  collects the *integer variables*, i.e. the knot vector maximum index  $m_1$  and the curve degree  $p_1$ ;
- $\boldsymbol{\xi}_2$  collects *continuous variables*, i.e. the knot vector non-trivial components  $U_j^{(1)}$  and the weights  $w_{i_1}$ .

Mathematically speaking, vectors  $\boldsymbol{\xi}_1$  and  $\boldsymbol{\xi}_2$  are represented as

$$\boldsymbol{\xi}_1 = \{m_1, p_1\} \in \mathbb{N}^2, \quad (7.10)$$

$$\boldsymbol{\xi}_2 = \{U_{p_1+1}^{(1)}, \dots, U_{m_1-p_1-1}^{(1)}, w_0, \dots, w_{m_1-p_1-1}\} \in \mathbb{R}^{N_v}, \quad (7.11)$$

where

$$N_v = 2m_1 - 3p_1 - 1. \quad (7.12)$$

$(N_v + 2)$  is the overall number of design variables.

As previously stated, in this work, the curve approximation problem is still framed as an optimisation problem, but a more general formulation is introduced. On the one hand, the objective function has been modified with respect to Eq. (7.1), namely:

$$\min_{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2} \Phi(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2) = \min_{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2} \left[ \frac{\sum_{k=0}^{\mu} \|\mathbf{C}(u_1^k) - \mathbf{Q}_k\|^2}{L_{TP}^2} \right]^{1/m_1}. \quad (7.13)$$

In Eq. (7.13), the parameter  $1/m_1$  appears as a power of the sum of squares of the Euclidean distance divided by the square of *chord length* of the curve  $L_{TP}$ , refer to Eq. (7.2). On the other hand, an optimisation constraint on the maximum radius of curvature of the NURBS curve is introduced: in real-world engineering problems, such a requirement is often imposed to improve the smoothness of the approximating curve. This constraint can be stated as:

$$g(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2) = \frac{\kappa_{max} - \kappa_{adm}}{\kappa_{adm}}, \quad (7.14)$$



with

$$\kappa_{max} = \max_{u_1} \kappa(u_1), \quad (7.15)$$

$$\kappa(u_1) = \frac{\|\mathbf{C}'(u_1) \wedge \mathbf{C}''(u_1)\|}{\|\mathbf{C}'(u_1)\|^3}. \quad (7.16)$$

In Eq. (7.14),  $\kappa_{adm}$  is the admissible value for the curvature which must be established according to the problem at hand. It should be noticed that the purpose of the constraint on the maximum curvature of the NURBS curve is twofold: on the one hand, it constitutes a precise technological requirement that affects the final shape of the curve; on the other hand, it allows for defining a well-posed mathematical problem, because it limits the growth of the degree  $p_1$  of the blending functions during optimisation.

**Remark** In order to understand the latter assertion, let us consider a very simple parametric curve  $\Gamma$  in the  $x_1 - x_2$  plane, namely,

$$x_1(t) = t, \quad x_2(t) = t^p, \quad (7.17)$$

For this case, the curvature  $\kappa_\Gamma(t)$  writes

$$\kappa_\Gamma(t) = \frac{|p(p-1)t^{p-2}|}{(1+p^2t^{2(p-1)})^{3/2}}. \quad (7.18)$$

Of course,  $\kappa_\Gamma(t)$  depends upon the local abscissa  $t$  as well as on the curve degree  $p$ . The maximum value of  $\kappa_\Gamma(t)$  can be calculated for different values of  $p$ . The result of such a computation is synthetically illustrated in Fig. 7.1. As it can be deduced from

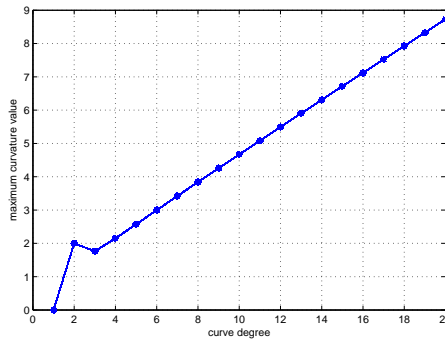


Figure 7.1 – Trend  $\kappa_{\Gamma max}$  vs  $p$  for the curve  $\Gamma$

Fig. 7.1, increasing the degree implies a higher value of the maximum curvature value for a simple polynomial curve as  $\Gamma$ . Being the NURBS curves defined through special polynomial blending functions, intuitively it can be stated that imposing a constraint on the maximum curvature value means also limiting the maximum curve degree.

Finally, the curve fitting problem can be stated in the standard form of a CNLPP of modular systems (refer to Chapter 3) as follows:

$$\begin{aligned} & \min_{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2} \Phi(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2), \\ & \text{subject to:} \\ & \begin{cases} g(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2) \leq 0, \\ \boldsymbol{\xi}_{1-lb} \leq \boldsymbol{\xi}_1 \leq \boldsymbol{\xi}_{1-ub}, \quad \boldsymbol{\xi}_1 \in \mathbb{N}^2, \\ \boldsymbol{\xi}_{2-lb} \leq \boldsymbol{\xi}_2 \leq \boldsymbol{\xi}_{2-ub}, \quad \boldsymbol{\xi}_2 \in \mathbb{R}^{N_v}. \end{cases} \end{aligned} \quad (7.19)$$

In Eq. (7.19),  $\boldsymbol{\xi}_{i-lb}$  and  $\boldsymbol{\xi}_{i-ub}$  ( $i = 1, 2$ ) represent the lower and upper bounds, respectively, of the vector  $\boldsymbol{\xi}_i$ .

**Remark** No analytical solutions are available in literature for problem (7.19). This is essentially due to the following difficulties.

- The problem aims at optimising both discrete and continuous variables: pure gradient-based methods are automatically discarded and hybrid strategy must be considered.
- Since the dimension of the continuous design variables vector  $\boldsymbol{\xi}_2$  depends on the discrete design variables collected in  $\boldsymbol{\xi}_1$ , problem (7.19) is stated on a domain having variable dimension, see Eqs. (7.10), (7.11) and (7.12). Pure gradient-based methods are not able to provide the solution in such cases.
- When considering the full set of design variables, both the objective and the curvature constraint functions become non-linear and non-convex.

Since the solution cannot be provided in a closed form, an approximate, i.e. pseudo-optimal, solution of problem (7.19) can be found by making use of a hybrid strategy combining classical gradient-based methods and the special GA, both discussed in Chapter 3. The problem formulation (7.19) together with the special features of the proposed algorithm (see section 7.2.2) allows for determining a pseudo-optimal feasible solution. Furthermore, the unusual form of objective function (7.13) allows the algorithm to automatically determine the best compromise between the number of knot vector components (and implicitly the number of design variables) and the precision of the solution. Let consider Eq. (7.13): assume  $\varphi = \frac{\sum_{k=0}^{\mu} \|\mathbf{C}(u_1^k) - \mathbf{Q}_k\|^2}{L_{TP}^2}$ . During the first iterations, it could happen either  $\varphi > 1$  or  $\varphi < 1$  if the least squares distance is greater or smaller than  $L_{TP}$ , respectively. If  $\varphi > 1$ , the number of knot vector components is encouraged to quickly grow in order to minimise the overall objective function. Consequently, in the next iterations, the algorithm will tend towards a solution with  $\varphi < 1$ . So, after a certain number of iterations, the case  $\varphi < 1$  will become predominant and, from that moment,

increasing the number of knot vector components will not necessarily imply better performances: in fact, increasing the parameter  $m_1$  means getting a lower value of  $\varphi < 1$  but, meanwhile, a decreasing exponent  $1/m_1$ . Therefore, the best value of  $m_1$  will be determined as a result of the compromise between these two contrasting effects.

### 7.2.2 Numerical Strategy

Considering the mathematical features of problem (7.19), a hybrid optimisation tool has been developed, see Fig. 7.2. It is composed of the MATLAB version of the GA BIANCA illustrated in Chapter 3, and the MATLAB *fmincon* algorithm [76].

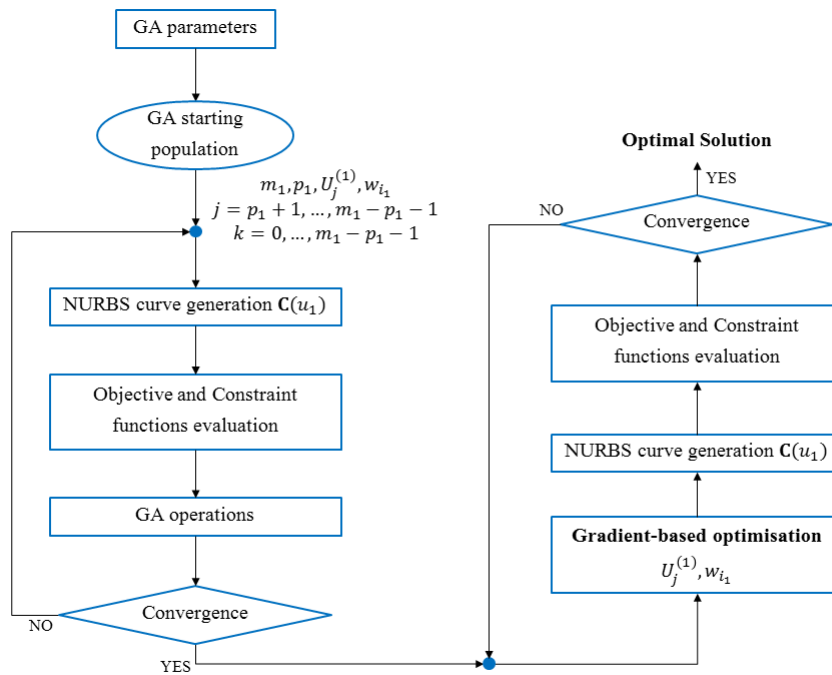


Figure 7.2 – Overview of the global numerical strategy for the curve fitting problem

As shown in Fig. 7.2, the optimisation procedure for problem (7.19) is split in two phases. During the first phase, solely the GA BIANCA is utilised to perform the solution search and the full set of design variables is taken into account.

The structure of the individual’s genotype for problem (7.19) is illustrated in Fig. 7.3. The first part of the *standard section* is characterised by one chromosome composed of two genes coding the design variables  $m_1$  and  $p_1$ , respectively. The second part of the *standard section* is constituted of a single chromosome with two genes coding the number of non-trivial components of the knot vector (the number of modules of the first type, i.e.  $n_{c-mod(1)} = m_1 - 2p_1 - 1$ ) and the number of weights (the number of modules of the

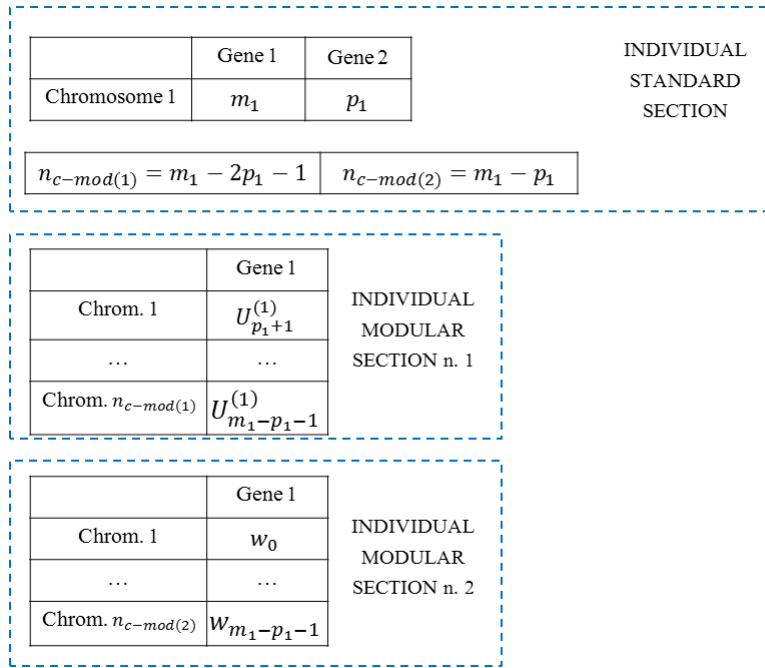


Figure 7.3 – The individual’s structure for the curve fitting problem

second type, i.e.  $n_{c-mod(2)} = m_1 - p_1$ ). Accordingly, the individual’s genome possesses two modular sections: the first one is composed of  $m_1 - 2p_1 - 1$  chromosomes with only one gene coding the value of the knot vector component  $U_j^{(1)}$ , while the second one is made of  $m_1 - p_1$  chromosomes with a single gene coding the value of the weight  $w_{i_1}$  in each control point.

Due to the strong non-linearity of problem (7.19), the aim of the genetic calculation is to provide a potential sub-optimal point in the design space, which constitutes the initial guess for the subsequent phase, i.e. the local optimisation, where the MATLAB *fmincon* gradient-based algorithm is employed to finalise the solution search. During this second phase only the components of the knot vector and the weights are considered as design variables, see Fig. 7.2.

### 7.2.3 Studied Cases and Results for curve fitting

In this section, some meaningful numerical examples are considered in order to prove the effectiveness of the proposed approach when dealing with the problem of the curve fitting. The set of genetic parameters tuning the behaviour of the GA (for each case) is listed in Table 7.1.

It is noteworthy that the number of both individuals and generations are chosen to

Parameter	Value
Number of populations ( $N_{pop}$ )	1
Number of individuals ( $N_{ind}$ )	250
Number of generations ( $N_{gen}$ )	320
Cross-over probability ( $p_{cross}$ )	0.85
Gene mutation probability ( $p_{mut}$ )	$1/N_{ind}$
Chromosome shift probability ( $p_{shift}$ )	0.5
Chromosome number mutation probability ( $p_{mut-chrom}$ )	$(n_{ch_{ub}} - n_{ch_{lb}})/N_{ind}$
Selection Operator	Roulette wheel
Elitism Operator	Active

Table 7.1 – Setting of genetic parameters

get  $N_{ind} \times N_{gen} = 80000$  function evaluations (as it is usual in literature [164]) for each considered problem. Furthermore, Table 7.2 summarises the design variables together with their bounds for problem (7.19).

Problem	$p_{1-lb}$	$p_{1-ub}$	$m_{1-lb}$	$m_{1-ub}$	$U_{j-lb}^{(1)}$	$U_{j-ub}^{(1)}$	$w_{i_1-lb}$	$w_{i_1-ub}$
The Descartes' folium	1	6	9	38	0	1	1	3
The four-leaf clover	1	8	8	67	0	1	1	3
The flame	1	8	100	130	0	1	1	3
The tennis ball stitching	1	8	8	67	0	1	1	3
The paddle curves	1	8	9	37	0	1	1	3

Table 7.2 – Setting of variables boundaries.

As far as concerns the *fmincon* optimisation tool employed for the local solution search at the end of the first step, the numerical algorithm chosen to carry out the calculations is the *active-set* method with non-linear constraints. The numerical results, for each case, are collected in Table 7.3 and Table 7.4. The curvature constraint is still active but the objective function has been slightly simplified.

The objective function of the gradient based algorithm is identified as  $\Phi_{grad}(\boldsymbol{\xi}_2)$  and it is simply the sum of the square distances. Since the objective function should be provided in dimensionless form (see Chapter 3),  $\Phi_{grad}(\boldsymbol{\xi}_2)$  is divided by the following reference quantity during the iterations:

$$\Phi_{grad-0} = L_{TP}^2 (\Phi^{opt})^{m_1}. \quad (7.20)$$

In Eq. (7.20),  $\Phi^{opt}$  is the value of the objective function  $\Phi$  at the end of the first optimisation step. It should be highlighted that the objective function  $\Phi_{grad}(\boldsymbol{\xi}_2)$  does not depend any more upon the discrete NURBS parameters: they have been optimised through the

Curve	$p_1$	$n_1$	$m_1$	$L_{TP}$	$\Phi^{opt}$	$g(\xi_1, \xi_2)$
Descartes'folium	4	15	20	3.01	0.4684	$-7.00 \times 10^{-2}$
Four-leaf clover	5	33	39	7.75	0.7572	$-6.00 \times 10^{-4}$
Flame	4	109	114	284.66	0.9232	$-1.42 \times 10^{-1}$
Tennis ball stitching	6	39	46	33.78	0.6235	$-1.76 \times 10^{-2}$
Paddle - c1	2	10	13	44.68	0.4522	$-8.20 \times 10^{-3}$
Paddle - c2	3	7	11	58.85	0.3979	$-1.00 \times 10^{-3}$
Paddle - c3	2	6	9	83.92	0.2981	$-7.00 \times 10^{-4}$
Paddle - c4	4	9	14	99.63	0.4455	$-8.91 \times 10^{-2}$
Paddle - c5	2	10	13	119.26	0.4059	$-2.28 \times 10^{-2}$
Paddle - c6	5	8	14	130.33	0.4775	$-4.70 \times 10^{-3}$
Paddle - c7	3	10	14	141.28	0.4229	$-4.70 \times 10^{-2}$
Paddle - c8	3	10	14	129.94	0.4697	$-5.95 \times 10^{-2}$
Paddle - c9	3	10	14	105.72	0.4285	$-9.23 \times 10^{-2}$
Paddle - c10	2	11	14	40.37	0.6360	$-4.68 \times 10^{-2}$
Paddle - t1	2	16	19	475.36	0.5552	$-1.00 \times 10^{-4}$
Paddle - t2	2	14	17	548.43	0.5051	$-1.00 \times 10^{-4}$

Table 7.3 – Genetic Algorithm: Numerical Results.

Curve	$\mu + 1$	$\Phi_{grad}(\xi_2)$	$d_{average}$
Descartes'folium	50	$1.60 \times 10^{-6}$	$2.53 \times 10^{-5}$
Four-leaf clover	211	$6.67 \times 10^{-4}$	$1.23 \times 10^{-4}$
Flame	315	$7.16 \times 10^{-1}$	$2.69 \times 10^{-3}$
Tennis ball stitching	201	$3.98 \times 10^{-7}$	$3.14 \times 10^{-6}$
Paddle - c1	86	$5.94 \times 10^{-2}$	$2.83 \times 10^{-3}$
Paddle - c2	97	$1.19 \times 10^{-1}$	$3.56 \times 10^{-3}$
Paddle - c3	93	$1.26 \times 10^{-1}$	$3.82 \times 10^{-3}$
Paddle - c4	89	$1.10 \times 10^{-1}$	$3.72 \times 10^{-3}$
Paddle - c5	86	$1.12 \times 10^{-1}$	$3.89 \times 10^{-3}$
Paddle - c6	93	$5.21 \times 10^{-1}$	$7.76 \times 10^{-3}$
Paddle - c7	90	$1.13 \times 10^{-1}$	$3.73 \times 10^{-3}$
Paddle - c8	89	$3.94 \times 10^{-1}$	$7.05 \times 10^{-3}$
Paddle - c9	83	$7.43 \times 10^{-2}$	$3.28 \times 10^{-3}$
Paddle - c10	78	$4.21 \times 10^0$	$2.63 \times 10^{-2}$
Paddle - t1	89	$5.40 \times 10^0$	$2.61 \times 10^{-2}$
Paddle - t2	88	$5.00 \times 10^0$	$2.54 \times 10^{-2}$

Table 7.4 – Gradient Algorithm: Numerical Results.

genetic step and they are kept constant in the gradient step. Finally, in Table 7.4, the quantity  $d_{average}$  is defined as

$$d_{average} = \frac{\Phi_{grad}(\xi_2)^{1/2}}{\mu + 1}, \quad (7.21)$$

which is an average distance between the TPs and the fitting curve, so  $d_{average}$  gives an idea of the fairness of the method.

### The Descartes' Folium

The Descartes' Folium is an open plane curve, whose parametric representation is

$$x_1(t) = at(t - 1), \quad x_2(t) = at(t - 1)(2t - 1). \quad (7.22)$$

The set of  $\mu + 1 = 50$  TPs is extracted from Eq. (7.22) by setting  $a = 2$  and it is shown in Fig. 7.4a. As it can be seen from the graphic results (Fig. 7.4b), the presence of the loop does not affect the final quality of the approximating curve. From Table 7.3, it can

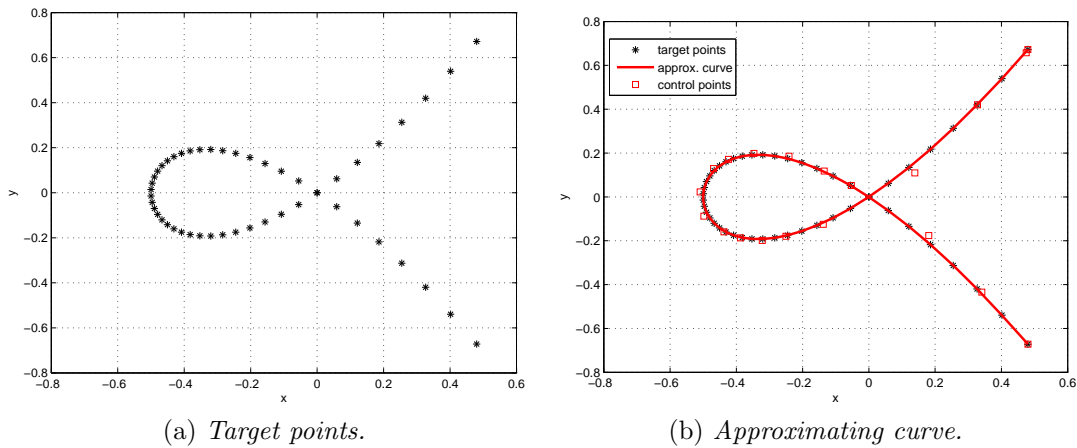


Figure 7.4 – The Descartes' Folium

be noticed that, due to the new form of the objective function and to the presence of the constraint on the maximum curvature, the optimum values of  $p_1$  and  $m_1$  are automatically determined by the GA because Eqs. (7.13) and (7.14) constitute implicit restrictions on both the degree of the basis functions and on the number of components of the knot vector.

### The Four-Leaf Clover

The Four-Leaf Clover is a plane closed curve described by the parametric equation

$$x_1(t) = \cos(t)\sin(2t), \quad x_2(t) = \sin(t)\sin(2t). \quad (7.23)$$

In this case,  $\mu + 1 = 211$  TPs have been extracted from the previous equation. The optimum fitting curve is illustrated in Fig. 7.5b, while the related numerical results are listed in Table 7.3 and Table 7.4. Regarding the optimum value of  $p_1$  and  $m_1$ , the same considerations as those of example 7.2.3 can be repeated here.

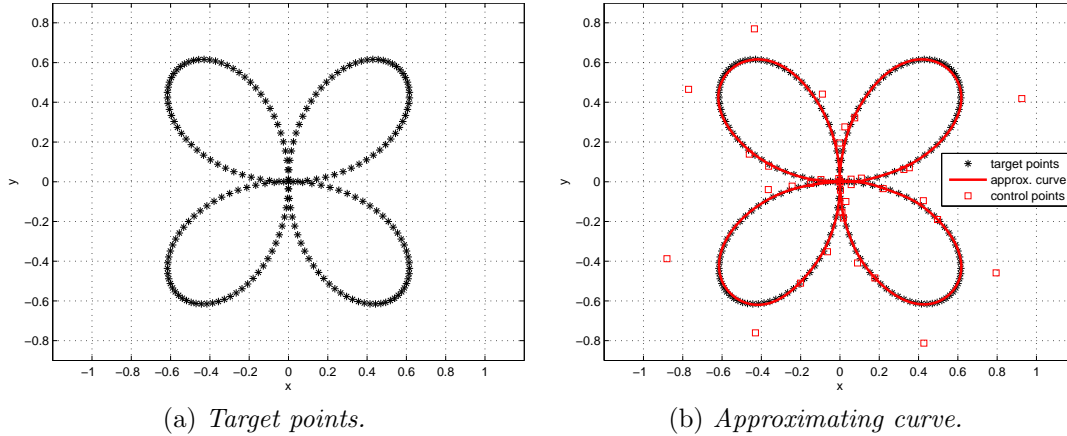


Figure 7.5 – The four-leaf clover

### The Flame

The third test case is a non-parametrised plane closed curve. 315 TPs have been sampled from the image of a flame, see Fig. 7.6a. This is a very challenging test case because of the complicated shape and the derivatives discontinuity. Indeed, the boundaries of the first two design variables have been broadened, in order to allow the curve to correctly evolve (see Table 7.2).

It must be pointed out that the constraint on the curvature is weaker than the previous cases, see Table 7.3: this is due, of course, to the presence of the cusps. Only for this example, the resulting knot vector and weights are provided in Appendix E to highlight the efficiency of the adopted strategy: some components are marked in bold font because they are very close, even the same. This fact reflects a well known NURBS property: if a knot has a multiplicity equal to  $\lambda$ , then the curve is  $p_1 - \lambda$  times continuously differentiable at the knot. As listed in Appendix E, the NURBS fitting curve is characterised by weights of different value: in particular, such weights get higher values for the control points located in the neighborhood of the cusps of the flame, see Fig. 7.7. However, all the weights values are close to the unity, which means that the cusps can be properly described through a smart choice of the knot vector components.



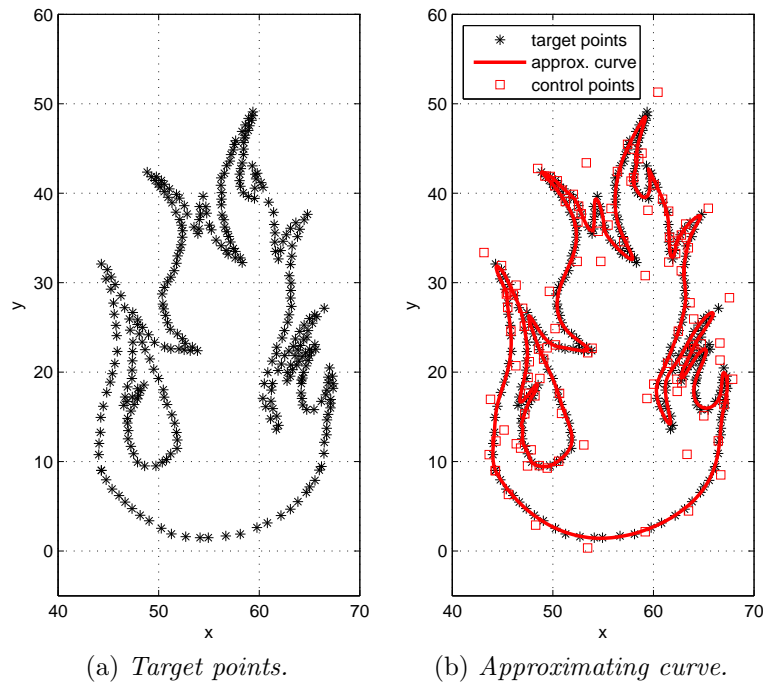


Figure 7.6 – The flame

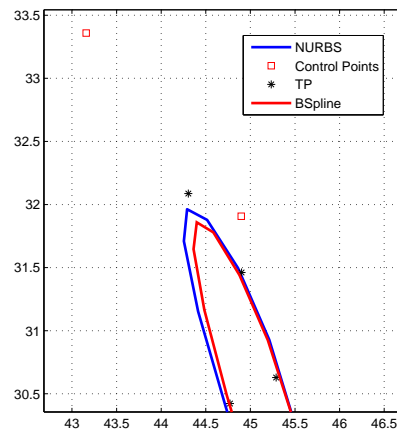


Figure 7.7 – Detail on the NURBS approximating the flame

### The Tennis Ball Stitching

The tennis ball stitching is a three-dimensional parametric curve. It has been chosen in order to provide a 3D test case for the curve fitting problem. The parametric form is

$$x_1(t) = a\cos(t) + b\cos(3t), \quad x_2(t) = a\sin(t) - b\sin(3t), \quad x_3(t) = c\sin(2t). \quad (7.24)$$

The  $\mu + 1 = 201$  TPs are extracted from Eq. (7.24) by setting  $a = 2$ ,  $b = 1$  and  $c = 2\sqrt{2}$ . The TPs as well as the optimum fitting curve are illustrated respectively in Fig. 7.8a and Fig. 7.8b.

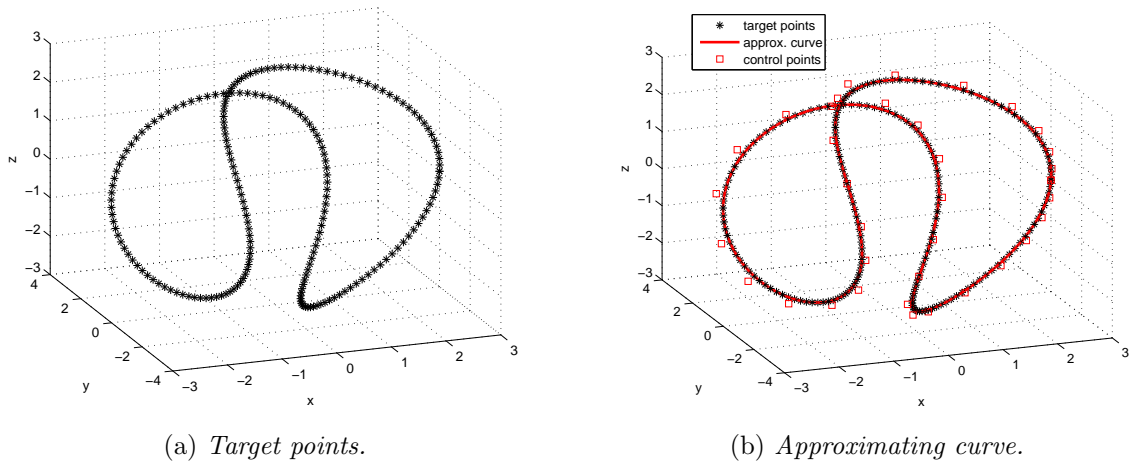


Figure 7.8 – The tennis ball stitching

### The Paddle

In this subsection, a real-world engineering problem is faced. A paddle has been scanned and all points representing its external surface are shown in Fig. 7.9. Hence, twelve subsets of TPs have been extracted (see Fig. 7.10a): each set is supposed to constitute a primitive three-dimensional curve that will be employed during the CAD reconstruction of the paddle. For each curve, a technological constraint on the curvature has been considered, as shown in Table 7.5.

Here, the effectiveness of the presented method is remarked through this real-world engineering application, since a complicated set of scanned points can be easily treated and the resulting NURBS curves restrain all the necessary information in order to rebuild the paddle, by adding a technological constraint. Fig. 7.10b gives a global overview of the shape of the paddle primitive curves provided by the proposed optimisation procedure.

### Discussion on the Presented Methodology

In this section, some remarks inherent to the parameters tuning the behaviour of the GA (to be set by the user) are discussed. A particular attention is dedicated to the definition of the bounds for the design variables, which have been established according to the following considerations.

*Continuous parameters* bounds are simple to set.

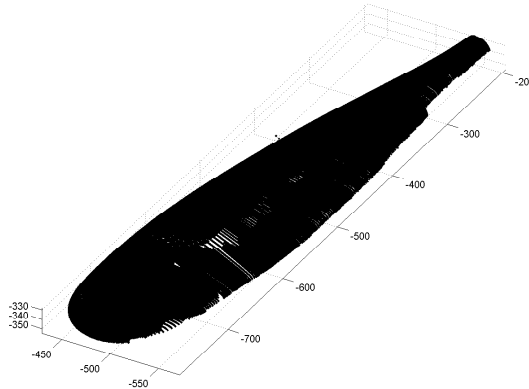


Figure 7.9 – Starting data set for the paddle problem

Curve	$\kappa_{adm}$
Descartes' folium	7.0000
Four-leaf clover	6.0000
Flame	70.0000
Tennis ball stitching	0.5500
Paddle - c1	0.2000
Paddle - c2	0.2000
Paddle - c3	0.1700
Paddle - c4	0.9000
Paddle - c5	0.9000
Paddle - c6	1.0000
Paddle - c7	0.2500
Paddle - c8	0.9000
Paddle - c9	0.8000
Paddle - c10	0.5500
Paddle - t1	0.0150
Paddle - t2	0.0120

Table 7.5 – Maximum allowed curvature values

- The knot vector components are defined between 0 and 1, so  $U_{j-lb}^{(1)} = 0$  and  $U_{j-ub}^{(1)} = 1$ .
- The weights of the NURBS curve can get, *a priori*, any real value in the range  $]0, \infty[$ . After a preliminary check on the first three proposed benchmarks (the Descartes's folium, the Four-Leaf Clover and the Flame problems), it was observed that the

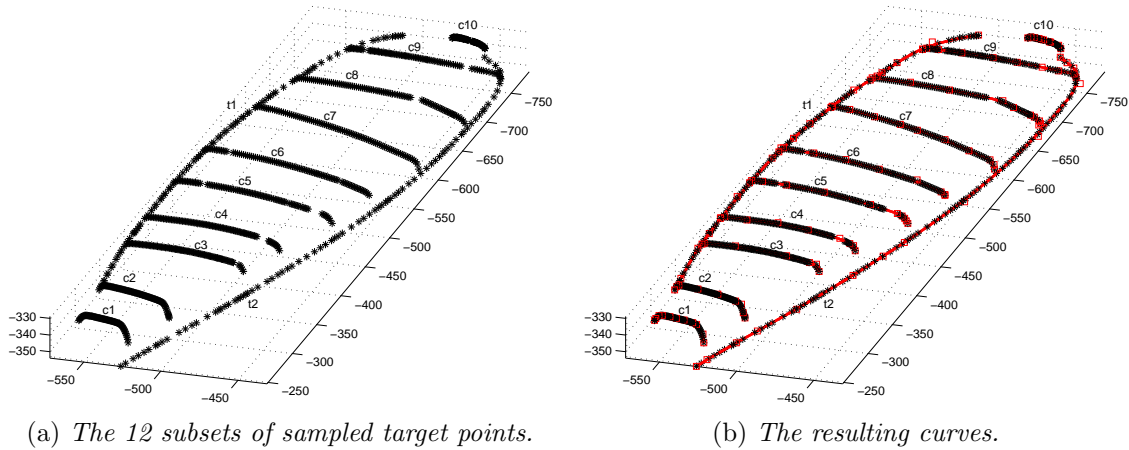


Figure 7.10 – The paddle

curve shape is affected by the ratio  $w_{ub}/w_{lb}$  rather than by the single value of the weight related to each control point. Moreover, as clearly shown in Appendix E, the weights are responsible of minor adjustments, which become significant only in presence of singularities (as in the case of the Flame problem). Taking into account these considerations, it has been set  $w_{lb} = 1$  and  $w_{ub} = 3$ .

Unlike weights, the *discrete parameters* have a major influence on the shape of the NURBS curve and their bounds must be carefully set.

- The minimum degree is, of course,  $p_{1-lb} = 1$ . The maximum degree has been fixed in order to avoid the introduction of noise that can become important when the upper bound is not properly set. Accordingly, the maximum degree has been set to  $p_{1-ub} = 8$  for all the examples with the exception of the first test case (the simplest one), where  $p_{1-ub} = 6$ .
- In order to establish lower and upper bounds for the number of the knot vector components ( $m_1 + 1$ ), the user should think about an ideal number of control points tuning the shape of the approximating NURBS curve. Indeed, this problem applies also in case of standard curve fitting methods (which are not capable of automatically optimise discrete parameters), where the user does not dispose of any criterion to choose a suitable number of control points. In the framework of the proposed method, the special GA utilised to perform the solution search for the curve fitting problem (refer to Eq. (7.19)) is able to automatically determine the optimum number of both knot vector components and degree of the basis functions, thus the related optimum number of control points, i.e.  $n_{opt} = m_{opt} - 1 - p_{opt}$ . Of course, the bounds on the variable  $m_1$  can be inferred according to empirical rules (taken from practice), utilised to define a criterion for setting the minimum and maximum

number of control points. In particular, the bounds on  $n_1$  can be set according to the following rules:

- 1) usually, the number of target points ( $\mu + 1$ ) should be, at least, three times the number of control points (in order to ensure redundancy). Therefore, the average number of control points can be assumed equal to  $(\mu + 1)/3$ ;
- 2) a suitable interval can be defined around this average value. In particular, the maximum number of control points must be lower than the number of target points, whilst the minimum one should be always greater or equal to 2. Anyway, regardless the definition of the interval for the variable  $m_1$ , an internal check (in the GA environment) is always performed to satisfy the condition  $n_1 \geq 1$ , thus meaningless situations, e.g.  $m_1 = 8$  and  $p_1 = 7$ , are always discarded.

Since the proposed hybrid algorithm is very efficient, it can be asserted that it is not important to choose the “right” narrow interval. When the shape of the curve is particularly complex and does not let the user guess the size of the interval, a wider range can be set, being the GA able to determine automatically the optimum value of the discrete parameters. Finally, it can be stated that the external user has a lower impact in the context of the proposed approach when compared to classical ones.

The previous discussion on the choice of the bounds for the number of knot vector components suggests investigating the sensitivity of the solution to the quantity of TPs. This is an interesting task that allows for discussing about the robustness and the efficiency of the methodology. Since the amount of parameters to be optimised is high, it is natural to wonder what happens when the number of data points (TPs) is reduced, i.e. when the algorithm benefits from less information. However, some remarks need a special attention.

Decreasing the number of TPs has a significant impact on the mathematical nature of the curve fitting problem in the form of the CNLPP (7.19). If the number of TPs (i.e. data points) is less than the number of design variables, the related system of equations becomes underdetermined and the solution is not unique. Conversely, solving the curve fitting problem can be interpreted as finding an approximate solution for an overdetermined system of equations. Therefore, talking about curve fitting when the number of TPs is lower than the number of design variables is meaningless. Indeed, in this case, there is not enough information to get a unique solution for the curve fitting problem.

Usually, in reverse engineering applications, the number of data points (retrieved, for instance, by means of a 3D scanner) is very high. In practice, the size of points clouds can be properly reduced (in order to save memory in data exchange), but a certain redundancy must be guaranteed in order to approximate the data points with a single (or multiple) CAD entity like a NURBS curve, tuned by a suitable number of parameters.

Taking into accounts these aspects, a sensitivity analysis to the number of TPs is provided in the following for the Four-Leaf Clover example (Fig. 7.11). Solutions depicted in Figs. 7.11b-7.11d have been obtained with a decreased number of TPs with respect to the reference solution of Fig. 7.11a and by using the same value of maximum allowable curvature ( $\kappa_{adm} = 6.0000$ ). Two cases have been considered:

- a) the bounds of design variables have been chosen according to the aforementioned empiric criteria (refer to Fig. 7.11b and Fig. 7.11c);
- b) the bounds of design variables do not change with respect to the reference case (Fig. 7.11d).

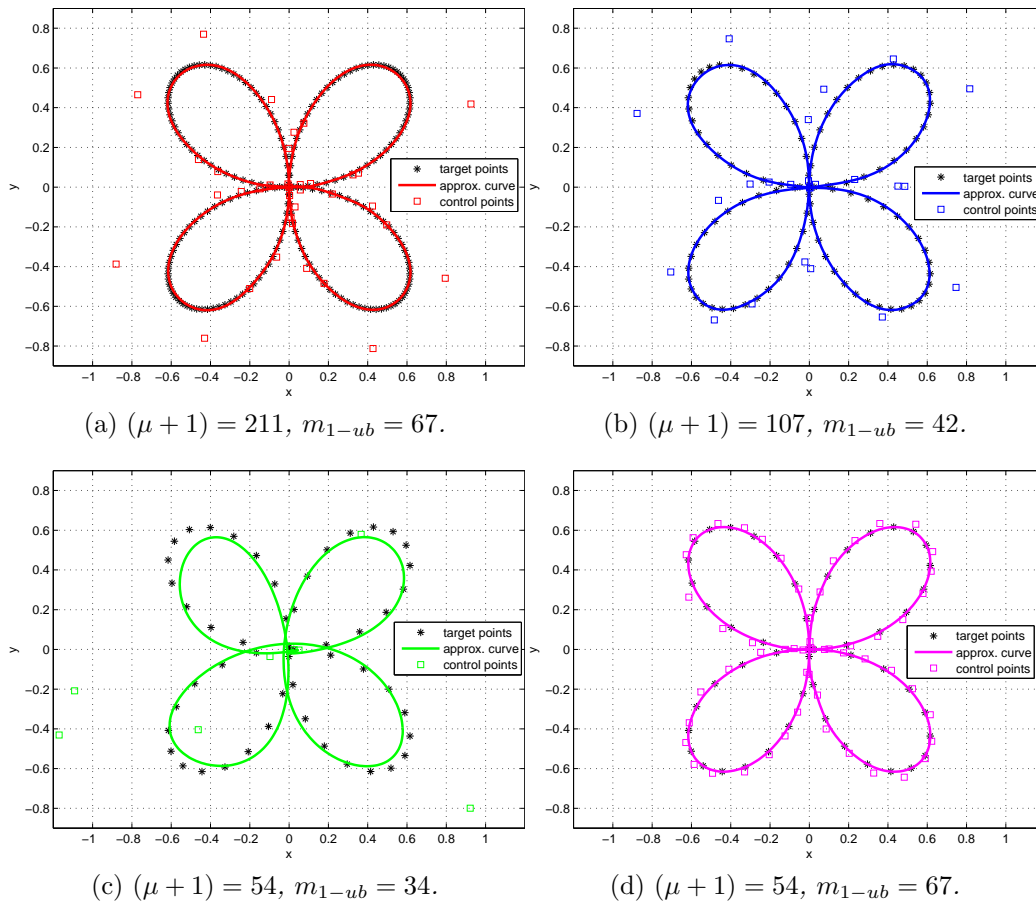


Figure 7.11 – Sensitivity to the number of TPs: approximating curves

As it can be inferred from both Fig. 7.11 and numerical results of Table 7.6, if the upper bound of  $m_1$  is set according to the proposed criterion, the lower is  $(\mu + 1)$ , the lower is

the quality of the solution: the quantity  $d_{average}$  increases and the approximating NURBS curve is not satisfactory, in particular when the number of TPs is reduced to 54 (Fig. 7.11c).

Four-Leaf Clover	$\mu + 1$	$m_{1-ub}$	$m_1$	$\Phi(\xi_1, \xi_2)$	$L_{TP}$	$\Phi_{grad}(\xi_2)$	$d_{average}$
1	211	67	39	0.7572	7.7516	$6.67 \times 10^{-4}$	$1.23 \times 10^{-4}$
2	107	42	28	0.7064	7.7408	$1.74 \times 10^{-3}$	$3.91 \times 10^{-4}$
3	54	34	16	0.6875	7.7105	$1.07 \times 10^{-1}$	$6.07 \times 10^{-3}$
4	54	67	63	0.3286	7.7105	$2.45 \times 10^{-31}$	$9.18 \times 10^{-18}$

Table 7.6 – Sensitivity to the number of TPs - Numerical results

This fact occurs because the criterion for choosing the bounds aims at balancing the number of design variables with the number of TPs, which makes sense in the context of the curve fitting problem. Nevertheless, when  $(\mu + 1) = 54$ , the system becomes undetermined. Actually, since the solution is not unique, when the upper bound of  $m$  is increased (Fig. 7.11d) without considering the proposed empirical rule, the algorithm provides an excellent solution, which can be seen as the solution of the related interpolation problem.

Finally, handling data points is an operation that should be carefully assessed: some crucial information could be removed and this operation could strongly impact on the problem definition (e.g. removing the peaks of singularity in the Flame example can lead to misleading results).

### 7.3 Surface Reconstruction in the NURBS Framework

As stated above, the surface reconstruction problem involves both the surface parametrisation and the surface fitting (see Fig. 7.12).

Similarly to the curve fitting problem, the surface fitting problem can be stated as

$$\min_{\xi} \sum_{k=0}^{\mu} \|\mathbf{S}(u_1^k, u_2^k) - \mathbf{Q}_k\|^2. \quad (7.25)$$

In Eq. (7.25), the same formalism of Eq. (7.1) is used:  $\mathbf{S}(u_1^k, u_2^k) = \{S_{x_1}(u_1^k, u_2^k), S_{x_2}(u_1^k, u_2^k), S_{x_3}(u_1^k, u_2^k)\}$  indicates the point belonging to the unknown surface that is related to the TP  $\mathbf{Q}_k$ . In the case of Eq. (7.25), the array  $\xi$  collects all the optimisation variables, i.e. all or part of the parameters (of different nature) defining the shape of the NURBS surface. A brief description of the two steps characterising the surface reconstruction problem is provided here below.

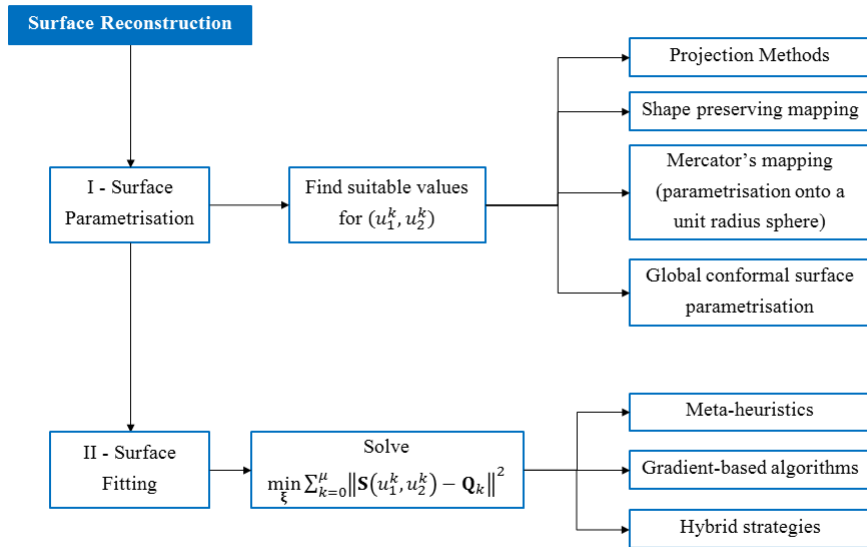


Figure 7.12 – Overview of the Surface Reconstruction scheme.

### 7.3.1 Surface Parametrisation

The parametrisation is the process of mapping a surface onto a region of the parametric plane. When the NURBS formalism of Eq. (7.25) is used, this claim means to determine suitable values for the dimensionless parameters  $u_1$  and  $u_2$  for a point belonging to the NURBS surface to be related to each TP  $\mathbf{Q}_k$ . The simplest parametrisation is the *projection* [68]. Projection is particularly advantageous when the TPs clouds can be approximated with a special open surface, where one of the coordinate is function of the other two (e.g.  $S_{x_3}(u_1^k, u_2^k) = S_{x_3}(S_{x_1}(u_1^k, u_2^k), S_{x_2}(u_1^k, u_2^k))$ ). In this peculiar conditions, projection is really efficient and the solution is unique.

However, most of practical applications of surface reconstruction do not deal with these special surfaces. If more complicated TPs clouds are considered, it could be necessary to make use of open but folded surfaces, wherein projection could easily fail because it is not unique. Alternatively, the algorithm developed by Floater in [111] can be used. This parametrisation technique can be used even for very complicated open surfaces under the following hypotheses: the boundary of the open surface can be identified and parametrised and the open surface does not present holes. The advantage of using the Floater's algorithm is that the resulting parametrisation mimics the physical topology of the surface used to approximate the TPs clouds (*shape preserving parametrisation*).



The importance of the shape preserving parametrisation goes beyond the simple case of an open surface: an important research topic is to find a proper division of closed TPs clouds and to approximate it by “patches” of open surfaces [165]. The real challenge of this approach is to make the procedure less dependent on the user (who has to choose the best way to cut the closed surface) and to impose suitable continuity conditions on the patches boundary (generally, the tangent continuity is required).

On the other hand, in the case of closed surfaces, it could be sought to approximate the TPs with a unique entity. When the surface is of genus zero, the approximation can be performed by using a particular Mercator projection, wherein each TPs is associated to specific latitude and longitude angles [150].

The problem becomes much more complex when the topology of the approximating surface presents holes. A general strategy can be the global conformal mapping suggested in [110]. The great benefit of this technique is that surfaces with arbitrary non-zero genus can be handled starting from a Delaunay triangulation. The properties of conformal maps are exploited here to establish parametrisation which are global, i.e. they hold everywhere with the exception of few points. In general, each surface which is homeomorphic to a torus can be easily parametrised. If the number of holes  $N_g$  is strictly superior to 1, then the number of special points where the parametrisation does not hold is  $2N_g - 2$ . As it is evident from this non-exhaustive presentation, parametrisation undoubtedly constitutes the most challenging step of surface reconstruction. However, an exhaustive state of the art on this topic is out of the scope of the present Thesis. Therefore, concerning the surface fitting problem, in the following it is assumed that a suitable parametrisation is provided by external routines.

### 7.3.2 Surface Fitting

Unlike the innovative curve fitting strategy described in section 7.2, the surface fitting proposed here below is carried out in the framework of a purely gradient-based algorithm, therefore the simultaneous optimisation of all the NURBS parameters (both continuous and discrete) is not considered here. Provided a TPs clouds, firstly a simple B-Spline surface is chosen: the only design variables are the control points coordinates. Once the control points coordinates are identified, a second optimisation step is performed in order to refine the quality of the solution. The complete algorithm is outlined here below: As far as the solution of problem (7.26) is concerned, it can be easily shown that a relationship similar to Eq. (7.9) can be obtained and it reads

$$[\mathbf{D}][\mathbf{D}]^T \mathbf{X}_P^j = [\mathbf{D}]\bar{\mathbf{X}}_Q^j, \quad j = 1, 2, 3. \quad (7.28)$$

In Eq. (7.28),  $\mathbf{X}_P^j$  is the array containing the  $x_j$  coordinates of all control points, whilst  $\bar{\mathbf{X}}_Q^j$  is the array of the TPs coordinates in the  $x_j$  directions. The matrix  $[\mathbf{D}] \in \mathbb{R}^{(n_1+1)(n_2+1) \times (\mu+1)}$

---

**Algorithm 1** Surface Fitting.

---

- 1: Set the B-Spline degrees  $p_1, p_2$  and number of control points  $(n_1 + 1) \times (n_2 + 1)$ . Set the components of the two knot vectors  $\mathbf{U}^{(1)}$  and  $\mathbf{U}^{(2)}$  as uniformly distributed in the interval  $[0, 1]$ .
- 2: Set the initial value of control points coordinates (initial guess of the approximating surface).
- 3: Solve the unconstrained optimisation problem

$$\begin{aligned} & \min_{\boldsymbol{\xi}_A} \sum_{k=0}^{\mu} \|\mathbf{S}(u_1^k, u_2^k) - \mathbf{Q}_k\|^2, \\ \boldsymbol{\xi}_A = & \left\{ X_{0,0}^{(1)}, \dots, X_{n_1,0}^{(1)}, \dots, X_{n_1,n_2}^{(1)}, X_{0,0}^{(2)}, \dots, X_{n_1,n_2}^{(2)}, X_{0,0}^{(3)}, \dots, X_{n_1,n_2}^{(3)} \right\}. \end{aligned} \quad (7.26)$$

- 4: Retrieve the solution of problem (7.26):  $\boldsymbol{\xi}_A^{opt}$ .
- 5: Set the control points coordinates of a new NURBS surface according to  $\boldsymbol{\xi}_A^{opt}$ . Set the same values for  $p_1, p_2, (n_1 + 1) \times (n_2 + 1), \mathbf{U}^{(1)}$  and  $\mathbf{U}^{(2)}$ .
- 6: Set the initial values of the NURBS weights equal to 1.
- 7: Solve the CNLPP

$$\begin{aligned} & \min_{\boldsymbol{\xi}_B} \sum_{k=0}^{\mu} \|\mathbf{S}(u_1^k, u_2^k) - \mathbf{Q}_k\|^2, \\ \boldsymbol{\xi}_B = & \{w_{0,0}, \dots, w_{n_1,0}, \dots, w_{n_1,n_2}\}, \\ & w_{i_1,i_2} \in [w_{lb}, w_{ub}], \quad \forall i_1 = 0, \dots, n_1, \quad \forall i_2 = 0, \dots, n_2. \end{aligned} \quad (7.27)$$

- 8: Retrieve the solution of problem (7.27):  $\boldsymbol{\xi}_B$ .
-

is defined as follows

$$D_{\tau,k} = R_{\tau}^k = R_{I_1,I_2}(u_1^k, u_2^k) = N_{I_1,p_1}(u_1^k)N_{I_2,p_2}(u_2^k), \quad (7.29)$$

where the linear index  $\tau$  is related to the indices  $I_1, I_2$  identifying a particular control point through the relationships of Eq. (4.13). Therefore, the solution of problem (7.26) is simply achieved by inverting the matrix  $([\mathbf{D}][\mathbf{D}]^T)$  and by applying

$$\mathbf{X}_P^j = ([\mathbf{D}][\mathbf{D}]^T)^{-1} [\mathbf{D}]\bar{\mathbf{X}}_Q^j, \quad j = 1, 2, 3. \quad (7.30)$$

The CNLPP (7.27) needs to be solved through one optimisation algorithm of those proposed in Chapter 3 because of the non-linear contribution of the NURBS weights, which are included among the design variables. Indeed, the gradient of the objective function with respect to the weights can be provided in closed form to speed up the computation:

$$\begin{aligned} & \frac{\partial \sum_{k=0}^{\mu} \|\mathbf{S}(u_1^k, u_2^k) - \mathbf{Q}_k\|^2}{\partial w_{I_1, I_2}} = \\ & = 2 \sum_{k=0}^{\mu} \left[ (S_{x_1}(u_1^k, u_2^k) - \bar{x}_1^k) \frac{\partial S_{x_1}(u_1^k, u_2^k)}{\partial w_{I_1, I_2}} + (S_{x_2}(u_1^k, u_2^k) - \bar{x}_2^k) \frac{\partial S_{x_2}(u_1^k, u_2^k)}{\partial w_{I_1, I_2}} + \right. \\ & \quad \left. + (S_{x_3}(u_1^k, u_2^k) - \bar{x}_3^k) \frac{\partial S_{x_3}(u_1^k, u_2^k)}{\partial w_{I_1, I_2}} \right] = \\ & = \frac{2}{w_{I_1, I_2}} \sum_{k \in S_{I_1, I_2}} R_{I_1, I_2}(u_1^k, u_2^k) \left[ (S_{x_1}(u_1^k, u_2^k) - \bar{x}_1^k) (X_{I_1, I_2}^1 - S_{x_1}(u_1^k, u_2^k)) + \right. \\ & \quad \left. + (S_{x_2}(u_1^k, u_2^k) - \bar{x}_2^k) (X_{I_1, I_2}^2 - S_{x_2}(u_1^k, u_2^k)) + (S_{x_3}(u_1^k, u_2^k) - \bar{x}_3^k) (X_{I_1, I_2}^3 - S_{x_3}(u_1^k, u_2^k)) \right]. \end{aligned} \quad (7.31)$$

### 7.3.3 Results on Surface Fitting

Results on surface fitting are presented here on some benchmarks taken from literature [160]. The data concerning each test case are reported in Table 7.7.

Results are shown in Figs. 7.13-7.16.

In order to point out the effects of the second step in the optimisation algorithm, numerical results have been collected in Table 7.8 and the objective function at the end of the first optimisation step ( $f_{obj-A}$ ) is compared to that resulting from the second one ( $f_{obj-B}$ ).

As it can be easily remarked by numerical results, the effect of the NURBS weights is more evident when the shape of the fitting surface becomes more intricate. In fact, the ratio of  $f_{obj-B}$  to  $f_{obj-A}$  takes the smallest value for the trefoil surface and the highest one for the simple paraboloid. Of course, if a higher number of control points is used, the impact of weights will be less important.

Type	$\mu + 1$	$n_1 + 1$	$n_2 + 1$	$p_1$	$p_2$
Paraboloid	400	6	6	2	2
Shell	2500	9	9	3	3
Klein	2500	15	15	3	3
Trefoil	4200	20	20	4	4

Table 7.7 – Surface Fitting - Data

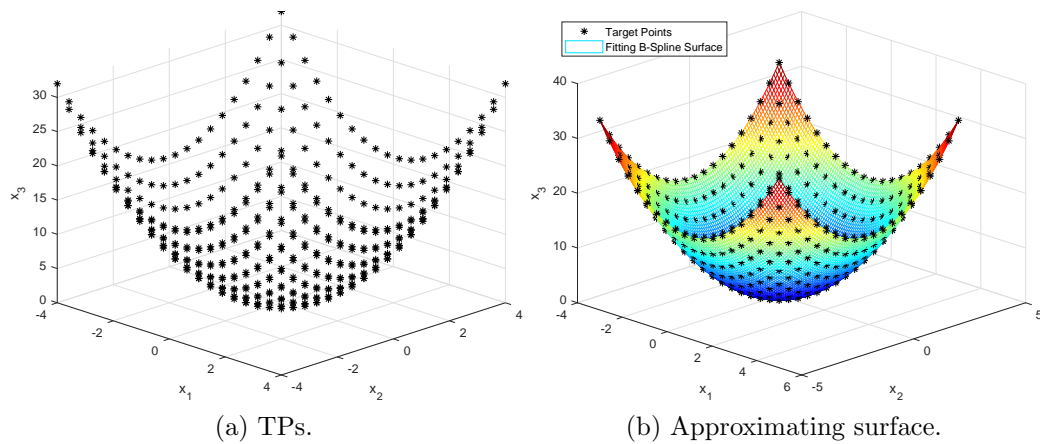


Figure 7.13 – Surface Fitting: Paraboloid.

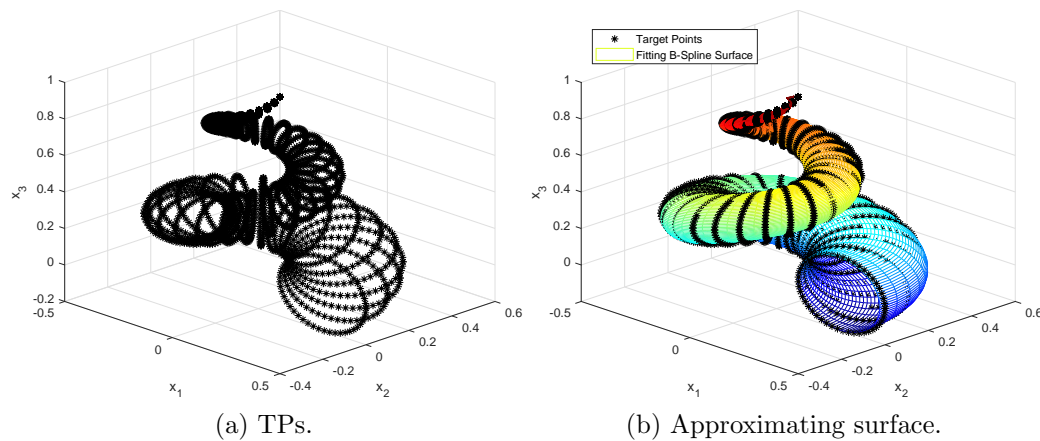


Figure 7.14 – Surface Fitting: Shell Surface.

## 7.4 Conclusions and Perspectives on Approximation Problems

In this Chapter, curve and surface fitting problems have been presented. Particularly, a general mathematical formalisation of the curve fitting problem together with an original

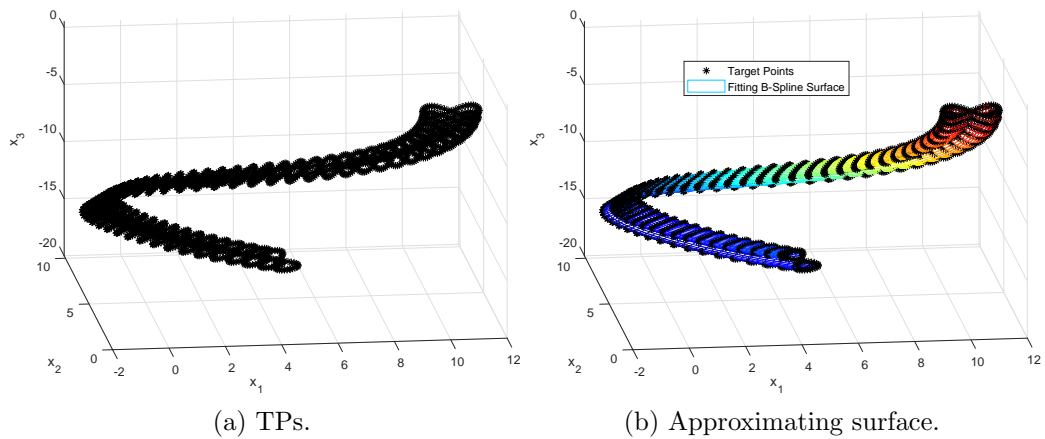


Figure 7.15 – Surface Fitting: Klein's Surface.

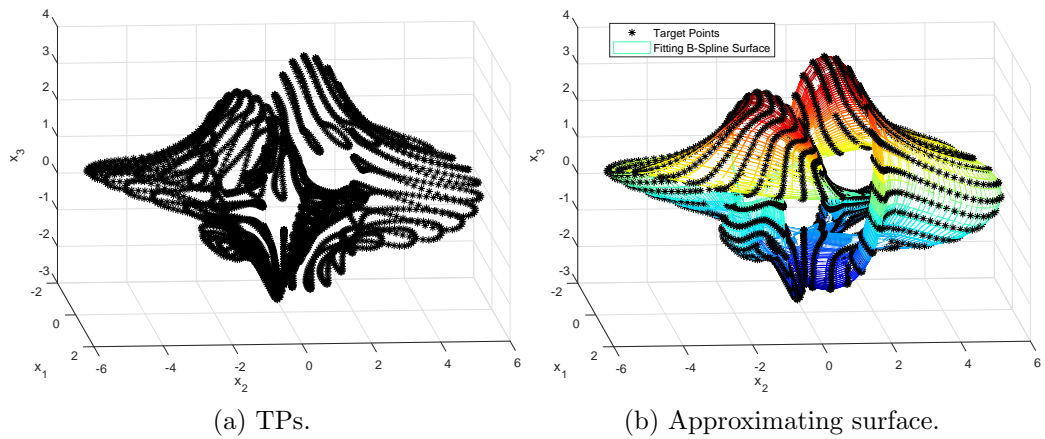


Figure 7.16 – Surface Fitting: Trefoil Surface.

Type	$f_{obj-A}$	$f_{obj-B}$	$f_{obj-B}/f_{obj-A}$
Paraboloid	$2.4957 \times 10^{-26}$	$2.3715 \times 10^{-26}$	0.9502
Shell	0.4186	0.3966	0.9476
Klein	1.6917	1.4271	0.8435
Trefoil	0.2861	0.2376	0.8305

Table 7.8 – Surface Fitting - Numerical results

optimisation procedure to perform the solution search in the framework of NURBS curves has been presented.

The proposed approach relies on the following features.

1. A new expression of the objective function, together with a suitable constraint on the maximum value of the curvature, has been introduced. These modifications imply a restriction on the integer design variables defining the shape of the NURBS curve. Moreover, the problem is stated as a CNLPP in which the number of unknowns is included among the design variables. Therefore, the problem of curve fitting is formulated in the most general case by considering as design variables both integer (the number of knots and the degree of the blending functions) and continuous (the components of the knot vector and the weights) parameters defining the NURBS curve. These aspects are of paramount importance, since the related CNLPP is defined over a domain of variable dimension, thus requiring a special optimisation procedure to find a feasible solution.
2. The non-convexity of the problem, together with a definition domain of variable dimension, justifies the use of non-standard numerical methods. To this purpose, the solution search for the curve fitting problem is performed by means of a hybrid optimisation tool (a GA coupled to a gradient-based method), of which the kernel is represented by the special GA BIANCA described in Chapter 3.

The effectiveness of the proposed approach is proven through some numerical examples focusing on 2D and 3D parametric as well as real-world engineering problems. The presented method can adapt the approximating curve to the desired level of smoothness, set through the curvature constraint: in fact, the algorithm is capable of successfully approximate smooth curves and those showing a discontinuity in the derivatives as well. The robustness of the method has been discussed with respect to the sensitivity to both the boundaries of the design variables and the number of initial target points. The number of knot vector components, i.e. the parameter that mainly affects the final quality of the approximating curve, needs suitable bounds which can be properly set by considering some practical guidelines provided in this study.

On the other hand, the surface fitting problem has been faced as well. Surface parametrisation constitutes the most challenging aspect when the surface reconstruction problem is addressed in real-world engineering applications (e.g. the surfaces reconstruction of a mechanical component whose boundary is provided in terms of an STL file, like in the case of the results of TO algorithms). However, from the preliminary results shown in this Chapter about the surface fitting in the NURBS framework, some interesting points deserve particular attention and constitute the main needs for future research.

- A formulation of the surface fitting problem in the most general case, i.e. by considering all parameters tuning the shape of a NURBS surface as design variables, does not exist. The choice of the number of control points along the parametric directions

and the degrees is, up to now, arbitrarily done by the user and not automatically provided. Analogously, the knot vector components are chosen evenly distributed in the interval  $[0, 1]$ .

- The importance of weights in surface fitting needs to be further investigated. In particular, a fundamental question naturally arises from the analysis of results achieved in this study: are weights really important for surface fitting or is it preferred to increase the number of control points?

Of course, the development of a more general strategy for surface fitting, similar to that proposed for curve fitting, is one of the most important perspective of this study. Nevertheless, the interest behind such a surface reconstruction tool goes beyond the more general mathematical reformulation of the surface fitting problem. From an engineering view-point, it would constitute a really helpful tool, which would be capable of reducing the number of arbitrary decisions usually taken by the designer when dealing with the surface reconstruction problem. Of course, this tool would be completed only by further investigating the following points.

- a) A suitable constraint should be formulated on the surface maximum allowable curvature, as it has been done for curves. This constraint is supposed to play an important role in avoiding too high blending functions degrees. However, the definition of curvature is not unique in the case of surfaces (mean curvature, Gauss curvature, etc.) and the choice should be addressed according to the specifications of the problem at hand.
- b) Research is ongoing to include an automatic or semi-automatic strategy for surface parametrisation. The proposed surface fitting algorithm can be coupled to the Mercator's mapping method for closed surfaces or to the conformal mapping method for genus one surfaces (i.e. those surfaces which are homomorphic to a torus). A natural extension to surfaces of genus  $N$  is possible but research should be oriented towards the study of topology, differential geometry (by putting the accent on the Riemannian geometry) and complex analysis.
- c) Eventually, the "patches" technique can be used to approximate mechanical components (e.g. the part resulting from a TO algorithm). In this case, the Floater shape preserving mapping method for open surfaces can be used for the parametrisation of each surface composing the boundary of the part. The kernel of such a strategy is the development of an automatic technique to choose the "optimum" number of patches to be retrieved through suitable cutting entities. Of course, the method must be able to automatically impose the desired continuity conditions among patches (simple continuity, tangent continuity, curvature continuity). This method is mathematically less elegant and rigorous if compared to the conformal

#### 7.4. *Conclusions and Perspectives on Approximation Problems*

---

map method. However, it would be extremely intuitive to the designer from a practical viewpoint.





# Chapitre 7

## Techniques de Fitting générales pour la Reconstruction CAO de Courbes et de Surfaces

### 7.1 Introduction à la Reconstruction de Courbes et Surfaces

La reconstruction de courbes et de surfaces est un sujet largement étudié en informatique, en modélisation géométrique et en ingénierie inverse. Dans le cas de la reconstruction de courbes, le but est de trouver tous les paramètres qui identifient de manière unique une courbe paramétrique se rapprochant d'un ensemble de *points cibles* (TPs). Le problème du fitting de courbes peut être défini comme un problème classique des moindres carrés, dans lequel la distance euclidienne entre les TPs et un ensemble de points appropriés appartenant à la courbe est minimisée. De même, le problème de la reconstruction de surface consiste à identifier les paramètres définissant synthétiquement une surface proche des TP.

En général, la reconstruction de courbes/surfaces se compose de deux étapes principales : *paramétrage* et *fitting*. Un paramétrage correct est établi sur les nuages de TPs lorsque la relation entre les points voisins dans le domaine paramétrique reflète la forme de la courbe ou la topologie de la surface dans l'espace 3D physique. Ainsi, la tâche de fitting identifie toutes les opérations de programmation mathématique nécessaires pour résoudre le problème d'optimisation des moindres carrés, lié à la reconstruction spécifique.

Dans le cas de problèmes de fitting (ou d'approximation) de courbes/surfaces, une quantité importante de TPs est disponible. Ces TPs ne satisfont pas nécessairement l'équation de l'entité géométrique qui convient, qui est disponible sous une forme paramétrique appropriée à des fins d'optimisation (implicite ou explicite). D'autre part, les problèmes d'*interpolation* visent à trouver une entité géométrique paramétrique passant par l'ensemble des TPs (c'est-à-dire que les TPs satisfont exactement à l'équation d'entité

paramétrique). Cette dernière classe de problèmes n'est pas considérée dans ce chapitre.

Les méthodes standard d'optimisation basées sur le gradient (voir le chapitre 3) ont été largement utilisées pour résoudre le problème de fitting de courbes dans sa formulation classique des moindres carrés [152–154]. Des stratégies analogues ont été utilisées dans le problème du fitting de surfaces [68, 155, 156].

Plusieurs méthodologies traitent le problème du fitting de courbes/surfaces dans le cadre des entités géométriques NURBS. Le développement massif de procédures méta-heuristiques a amené les ingénieurs à appliquer de telles stratégies au problème du fitting. Les méta-heuristiques peuvent être utilisées directement pour résoudre le problème de fitting formulé comme la minimisation des moindres carrés. Limaiem *et al.* [158] utilisent un GA pour trouver la valeur optimale des paramètres définissant la courbe d'approximation. Kang *et al.* [159] utilisent une "sparse optimisation" pour mettre à jour de manière itérative la longueur du *knot vector* et les composants de la courbe B-Spline approximante. Dans [96], un algorithme "particle swarm" a été utilisé pour approximer les TPs au moyen de surfaces B-Spline. Les mêmes auteurs proposent un GA à deux étapes pour choisir le paramétrage des points cible et les *knot vectors* pour effectuer le fitting de surfaces via des entités B-Spline dans [160]. Récemment, Garcia-Capulin *et al.* [162] a utilisé un GA hiérarchique pour optimiser à la fois le nombre et la valeur des knots d'une courbe B-Spline.

Les principales limitations et inconvénients caractérisant la grande majorité des études sur l'approximation par courbes / surfaces sont essentiellement deux :

- l'absence de formulation appropriée du problème (sans considérer des coefficients de pénalisation arbitraires, qui doivent être définis par l'utilisateur et qui dépendent du problème) ;
- l'absence d'une stratégie numérique très générale, capable d'optimiser simultanément le *nombre* ainsi que la *valeur* des paramètres constitutifs (i.e. les *variables de conception*) définissant la forme de la courbe NURBS / surface.

Pour surmonter les restrictions précédentes, nous proposons dans ce chapitre une approche innovante du problème d'approximation des courbes. Une nouvelle formulation du problème mathématique a été développée : le problème du fitting de courbes est défini comme une CNLPP en introduisant une contrainte sur la valeur maximale de la courbure. Les principales caractéristiques de la stratégie proposée sont les suivantes :

- Le problème du fitting de courbes est formulé en incluant le nombre de points de contrôle et le degré des fonctions de base parmi les inconnues. Ensuite, le *nombre global* de variables de conception (c'est-à-dire le nombre total de paramètres définissant la forme de la courbe) pour le problème en question est défini sur un espace de recherche de dimension variable.
- Les variables d'optimisation du CNLPP sont de nature différente (continue et discrète).

- La stratégie numérique envisagée pour la recherche de solution d'un tel CNLPP est basée sur la version MATLAB de le GA BIANCA, présentée au chapitre 3. Il est capable de gérer des variables de conception de nature différente et d'optimiser à la fois la dimension du domaine de conception et la valeur de chaque paramètre constitutif de la courbe NURBS.

En outre, le problème de la reconstruction de surfaces a également été résolu. Après une brève discussion sur les stratégies de paramétrage, un algorithme basé sur le gradient est proposé pour l'approximation de surfaces en deux étapes. Tout d'abord, les coordonnées des points de contrôle sont optimisées, puis les poids des NURBS.

Ce chapitre est décrit comme suit. La stratégie générale innovante pour résoudre le problème d'approximation de courbes est décrite dans le paragraphe 7.2 : la formulation mathématique et la stratégie numérique associée sont discutées en profondeur. Les résultats sont présentés sur des benchmarks de difficulté différente et sur un problème d'ingénierie réel plus difficile. Le paragraphe 7.3 se concentre sur les développements préliminaires de la reconstruction de surfaces : les problèmes liés à la généralisation du problème d'approximation de courbes au cas des surfaces sont identifiés et les solutions possibles sont discutées. Enfin, le paragraphe 7.4 termine le chapitre avec des conclusions et des perspectives.

## 7.2 Une stratégie d'Optimisation Hybride Générale pour le Fitting des Courbes dans le cadre NURBS

### 7.2.1 L'énoncé mathématique du problème de Fitting de Courbes

Considérons la forme classique du problème d'approximation de l'éq. (7.1), où  $(\mu + 1)$  est le nombre de TPs,  $\mathbf{Q}_k$  le  $k$ -ème point,  $\mathbf{Q}_k = \{\bar{x}_1^k, \bar{x}_2^k, \bar{x}_3^k\}$  sont les coordonnées cartésiennes des TPs, tandis que  $\mathbf{C}(u_1^k) = \{C_{x_1}(u_1^k), C_{x_2}(u_1^k), C_{x_3}(u_1^k)\}$  sont leur contrepartie appartenant à la courbe paramétrique lorsque le paramètre adimensionnel  $u_1$  prend la valeur  $u_1^k$ . Dans la même équation, le vecteur  $\boldsymbol{\xi}$  recueille toutes les variables d'optimisation, c'est-à-dire l'ensemble complet des paramètres (de nature différente) définissant la forme de la courbe. Dans le cas le plus général, lorsque la courbe paramétrique de l'éq. (7.1) est représentée dans le cadre mathématique des fonctions de base NURBS, sa forme dépend des paramètres suivants :

- *paramètres discrets*, i.e. le nombre de points de contrôle  $n_1 + 1$ , le nombre de knots  $m_1 + 1$  et le degré des fonctions de base  $p_1$  ;
- *paramètres continus*, i.e. la séquence non décroissante des composants du *knot vector*  $U_j^{(1)}$ ,  $j \in [p_1 + 1, m_1 - p_1 - 1]$ , les coordonnées des points de contrôle  $\mathbf{P}_{i_1} =$

$\{X_{i_1}^1, X_{i_1}^2, X_{i_1}^3\}$ ,  $i_1 \in [0, n_1]$ , les poids  $w_{i_1}$ ,  $i_1 \in [0, n_1]$  et l'ensemble des valeurs appropriées du paramètre sans dimension de la courbe  $u_1^k$ ,  $k \in [0, \mu]$ .

Considérons les paramètres discrets : éq. (2.6) donne la relation entre  $m_1$ ,  $p_1$  et  $n_1$ . Contrairement à l'approche standard, aucune hypothèse n'est faite sur les paramètres discrets de la courbe NURBS. En particulier,  $m_1$  et  $p_1$  sont inclus dans le vecteur des variables de conception, tandis que  $n_1$  sera calculé selon l'éq.(2.6). Concernant les paramètres continus, les valeurs  $u_1^k$  du paramètre adimensionnel de la courbe sont calculées via la méthode *chord length* [68]. Elles ne sont donc plus des variables de conception. En particulier, la *chord length*  $L_{TP}$  de la courbe peut être définie par l'éq. (7.2). Le paramètre  $u_1^k$  peut être calculé au moyen de l'éq. (7.3).

De plus, la valeur optimale des coordonnées des points de contrôle peut être obtenue par l'approche analytique de Ueng *et al.* [153]. Soient  $\mathbf{X}_P^1, \mathbf{X}_P^2, \mathbf{X}_P^3 \in \mathbb{R}^{n_1+1}$  des vecteurs colonne collectant les coordonnées  $x_1$ ,  $x_2$  et  $x_3$  des points de contrôle et  $\bar{\mathbf{X}}_Q^1, \bar{\mathbf{X}}_Q^2, \bar{\mathbf{X}}_Q^3 \in \mathbb{R}^{\mu+1}$  les contreparties des TPs. De plus, la matrice  $[\mathbf{A}] \in \mathbb{R}^{(\mu+1) \times (n_1+1)}$  et la matrice  $[\mathbf{B}] \in \mathbb{R}^{(n_1+1) \times (n_1+1)}$  peuvent être définies comme indiqué dans les éqs. (7.4) et (7.5).

Par conséquent, la proposition suivante s'applique.

**Proposition 7.2.1** Pour une courbe NURBS de degré assigné  $p_1$ , nombre de points de contrôle  $(n_1 + 1)$ , *knot vector*  $\mathbf{U}^{(1)}$  et poids  $w_{i_1}$  ( $i_1 = 0, \dots, n_1$ ), les coordonnées du point de contrôle minimisant la fonction de coût  $f$  du problème (7.1) sont données par l'éq. (7.6).

Les variables de conception peuvent être réparties en deux vecteurs  $\boldsymbol{\xi}_1$  et  $\boldsymbol{\xi}_2$  :

- $\boldsymbol{\xi}_1$ , défini dans l'éq. (7.10), recueille les variables *discrètes*, c'est-à-dire l'index maximal du *knot vector*  $m_1$  et le degré de la courbe  $p_1$  ;
- $\boldsymbol{\xi}_2$ , défini dans l'éq. (7.11), recueille les *variables continues*, i.e. les composantes du *knot vector*  $U_j^{(1)}$  et les poids  $w_{i_1}$ .

Le nombre total de variables de conception est  $2m_1 - 3p_1 + 1$ .

Comme indiqué précédemment, dans ce travail, le problème de l'approximation de courbes est toujours présenté comme un problème d'optimisation, mais une formulation plus générale est introduite. D'une part, la fonction objectif a été modifiée par rapport à l'éq.(7.1), et elle est donnée par l'éq. (7.13). Néanmoins, la contrainte d'optimisation sur le rayon de courbure maximal de la courbe NURBS est introduite comme indiqué dans les éqs. (7.14)-(7.16). Dans l'éq. (7.14),  $\kappa_{adm}$  est la valeur admissible pour la courbure qui doit être établie en fonction du problème rencontré. Comme on peut le déduire de la fig. 7.1, l'augmentation du degré implique une valeur plus élevée de la courbure maximale pour une courbe polynomiale simple sous la forme  $\Gamma$ . Étant donné que les courbes NURBS sont définies par des fonctions de base polynomiales, on peut intuitivement affirmer qu'imposer une contrainte sur la valeur de courbure maximale signifie également limiter le degré maximal de la courbe.

Enfin, le problème d'approximation de courbes peut être énoncé sous la forme standard d'un CNLPP de systèmes modulaires (voir chapitre 3) comme indiqué dans l'éq. (7.19), où  $\xi_{i-lb}$  et  $\xi_{i-ub}$  ( $i = 1, 2$ ) représentent respectivement les limites inférieure et supérieure du vecteur  $\xi_i$ .

Le problème vise à optimiser à la fois les variables discrètes et continues : les méthodes basées sur un gradient pur ne sont pas en mesure de fournir la solution dans de tels cas. Une stratégie hybride est donc considérée. Étant donné que la dimension des variables de conception continues  $\xi_2$  dépend des variables de conception discrètes collectées dans  $\xi_1$ , le problème (7.19) doit être résolu sur un domaine ayant une dimension variable. Une solution approximée, pseudo-optimale, du problème (7.19) peut être trouvée en utilisant une stratégie hybride combinant des méthodes classiques basées sur le gradient et le GA spécial, toutes deux discutées dans le chapitre 3.

### 7.2.2 Stratégie Numérique

En prenant en compte les caractéristiques mathématiques du problème (7.19), un outil d'optimisation hybride a été développé, voir la fig. 7.2. Il est composé de la version MATLAB du GA BIANCA illustrée au chapitre 3 et de l'algorithme *fmincon* [76]. Comme le montre la fig. 7.2, la procédure d'optimisation du problème (7.19) est divisée en deux phases. Au cours de la première phase, seulement le GA BIANCA est utilisée pour effectuer la recherche de solutions et l'ensemble complet des variables de conception est pris en compte. La structure du génotype de l'individu pour le problème (7.19) est illustrée à la fig. 7.3.

En raison de la forte non-linéarité du problème (7.19), le but du calcul génétique est de fournir un point potentiel sous-optimal dans l'espace de conception, qui constitue la première estimation pour la phase suivante, c'est-à-dire l'optimisation locale, où l'algorithme *fmincon* basé sur le gradient est utilisé pour finaliser la recherche de la solution. Au cours de cette seconde phase, seules les composantes du *knot vector* et les poids sont considérés comme des variables de conception, voir la fig. 7.2.

### 7.2.3 Cas d'étude et Résultats pour le Fitting de Courbes

L'ensemble des paramètres génétiques réglant le comportement du GA (pour chaque cas) est reporté dans le Tableau 7.1. Il est intéressant de noter que le nombre d'individus et de générations est choisi pour obtenir  $N_{ind} \times N_{gen} = 80000$  évaluations de la fonction objectif [164] pour chaque problème considéré. De plus, le Tableau 7.2 résume les variables de conception avec leurs limites pour le problème (7.19).

En ce qui concerne l'outil d'optimisation *fmincon* utilisé pour la recherche de solutions locales à la fin de la première étape, l'algorithme numérique choisi pour effectuer les calculs est la méthode AS avec des contraintes non linéaires. Les résultats numériques, pour chaque cas, sont rassemblés dans les Tableaux 7.3 et 7.4. La contrainte de courbure

est toujours active mais la fonction objectif a été légèrement simplifiée :  $\Phi_{grad}(\xi_2)$  est simplement la somme des distances carrées.

Puisque la fonction objectif doit être fournie sous forme adimensionnelle (voir chapitre 3),  $\Phi_{grad}(\xi_2)$  est divisé par la quantité de référence de l'éq. (7.20) pendant les itérations. Dans l'éq. (7.20),  $\Phi^{opt}$  est la valeur de la fonction objectif  $\Phi$  à la fin de la première étape d'optimisation. Enfin, dans le Tableau 7.4, la quantité  $d_{average}$  est définie par l'éq. (7.21) et représente une distance moyenne entre les TPs et la courbe approximante.

### Le Folium de Descartes

Le Folium de Descartes est une courbe plane ouverte, dont la représentation paramétrique est donnée par l'éq. (7.22). L'ensemble de  $\mu + 1 = 50$  TPs est extrait de l'éq.(7.22) en sélectionnant  $a = 2$  et il est montré à la fig. 7.4a. Comme on peut le voir sur la fig. 7.4b, la présence de la boucle n'affecte pas la qualité finale de la courbe d'approximation. Du Tableau 7.3, on peut remarquer que, en raison de la nouvelle forme de la fonction objectif et de la présence de la contrainte sur la courbure maximale, les valeurs optimales de  $p_1$  et  $m_1$  sont automatiquement déterminés par le GA.

### Le Trèfle à quatre feuilles

Le trèfle à quatre feuilles est une courbe plane fermée décrite par l'éq. (7.23). Dans ce cas,  $\mu + 1 = 211$  TPs ont été extraits de l'équation précédente. La courbe approximante optimale est illustrée à la fig. 7.5b, tandis que les résultats numériques associés sont recensés dans les Tableaux 7.3 et 7.4. En ce qui concerne la valeur optimale de  $p_1$  et  $m_1$ , les mêmes considérations que celles de l'exemple du Folium peuvent être répétées ici.

### La Flamme

Le troisième test est une courbe fermée non paramétrées. 315 TP ont été échantillonnés pour l'image d'une flamme, voir la fig. 7.6a. Ceci est un cas d'étude très difficile en raison de la forme complexe et de la discontinuité en tangence. En effet, les limites des deux premières variables de conception ont été élargies afin de permettre à la courbe d'évoluer correctement (voir le Tableau 7.2).

Il faut souligner que la contrainte sur la courbure est plus faible que dans les cas précédents (Tableau 7.3) : cela est dû, bien sûr, à la présence de pics. Ce n'est que pour cet exemple que le *knot vector* et les poids résultants sont fournis dans l'Annexe E pour souligner l'efficacité de la stratégie adoptée : certains éléments sont en caractères gras car ils sont très proches, même identiques. Cette remarque met en lumière une propriété NURBS bien connue : si une composante du *knot vector* a une multiplicité égale à  $\lambda$ , alors la courbe est  $p_1 - \lambda$  fois différentiable au knot.

## La Couture de la balle de tennis

La couture de la balle de tennis est une courbe paramétrique tridimensionnelle. La forme paramétrique est donnée par l'éq.(7.24). Les  $\mu + 1 = 201$  TPs sont extraits de l'éq.(7.24) en mettant  $a = 2$ ,  $b = 1$  et  $c = 2\sqrt{2}$ . Les TPs ainsi que la courbe optimale sont illustrés respectivement à la fig. 7.8a et à la fig. 7.8b.

## La Pagaie

Dans ce sous-paragraphe, un problème d'ingénierie réel est rencontré. Une pagaie a été scannée et tous les points représentant sa surface externe sont représentés sur la fig. 7.9. Ainsi, douze sous-ensembles de TPs ont été extraits (voir la fig. 7.10a) : chaque ensemble est censé constituer une courbe tridimensionnelle primitive qui sera utilisée lors de la reconstruction CAO de la pagaie. Pour chaque courbe, une contrainte technologique sur la courbure a été prise en compte, comme indiqué dans le Tableau 7.5. La fig. 7.10b donne un aperçu global de la forme des courbes primitives de la pagaie fournies par la procédure d'optimisation proposée.

## Discussion sur la Méthodologie présentée

Dans cette section, quelques remarques sont dédiées à la définition des limites pour les variables de conception, qui ont été établies en fonction des considérations suivantes.

Les limites des *paramètres continus* sont simples à définir. Les composantes du *knot vector* sont définies entre 0 et 1, donc  $U_{j-lb}^{(1)} = 0$  et  $U_{j-ub}^{(1)} = 1$ . On a observé que la forme de la courbe est affectée par le rapport  $w_{ub}/w_{lb}$  plutôt que par la valeur unique du poids lié à chaque point de contrôle. De plus, comme le montre clairement l'Annexe E, les poids sont responsables d'ajustements mineurs, qui ne deviennent significatifs qu'en présence de singularités (comme dans le cas du problème de la flamme). Compte tenu de ces considérations, il a été défini  $w_{lb} = 1$  et  $w_{ub} = 3$ .

Contrairement aux poids, les *paramètres discrets* ont une influence majeure sur la forme de la courbe NURBS et leurs limites doivent être soigneusement définies. Le degré minimum est, bien sûr,  $p_{1-lb} = 1$ . Le degré maximum a été fixé à  $p_{1-ub} = 8$  pour tous les exemples à l'exception du premier test, où  $p_{1-ub} = 6$ , afin d'éviter l'introduction de bruit. Bien entendu, les limites de la variable  $m_1$  peuvent être déduites selon des règles empiriques (issues de la pratique), utilisées pour définir un critère de définition du nombre minimal et maximal de points de contrôle. En particulier, les limites de  $n_1$  peuvent être définies selon les règles suivantes :

- 1) le nombre de points cibles ( $\mu + 1$ ) doit être au moins trois fois le nombre de points de contrôle (afin d'assurer la redondance). Par conséquent, le nombre moyen de points de contrôle peut être supposé égal à  $(\mu + 1)/3$ ;



- 2) un intervalle approprié peut être défini autour de cette valeur moyenne. En particulier, le nombre maximal de points de contrôle doit être inférieur au nombre de points cibles, tandis que le minimum doit toujours être supérieur ou égal à 2.

On peut donc affirmer que l'utilisateur externe a moins d'influence dans le contexte de l'approche proposée que les approches classiques.

La discussion précédente sur le choix des limites pour le nombre de composantes du *knot vector* suggère d'étudier ce qui se produit lorsque le nombre de points cible est réduit, c'est-à-dire lorsque l'algorithme bénéficie de moins d'informations. Une analyse de sensibilité du nombre de TPs est fournie ci-après pour l'exemple du trèfle à quatre feuilles (fig. 7.11). Les solutions illustrées aux figs. 7.11b-7.11d ont été obtenus avec un nombre de TPs réduit par rapport à la solution de référence de la fig. 7.11a et en utilisant la même valeur de courbure maximale ( $\kappa_{adm} = 6.0000$ ).

Comme on peut le déduire du Tableau 7.6, si la limite supérieure de  $m_1$  est définie selon le critère proposé, plus  $(\mu + 1)$  diminue, plus la qualité de la solution se dégrade : la quantité  $d_{average}$  augmente et la courbe NURBS approximante n'est pas satisfaisante.

Enfin, la gestion des points cible est une opération qui doit être soigneusement évaluée : certaines informations cruciales pourraient être supprimées et cette opération pourrait avoir un impact important sur la définition du problème (par exemple, supprimer les pics de singularité dans l'exemple de la flamme peut conduire à des résultats trompeurs).

## 7.3 Reconstruction de Surfaces le cadre des NURBS

Comme indiqué ci-dessus, le problème de reconstruction surfaces implique à la fois le paramétrage de la surface et le fitting de la surface (voir la fig. 7.12).

De même que pour le problème d'approximation de courbes, le problème de fitting de surfaces peut être exprimé comme indiqué dans l'éq. (7.25), où le même formalisme de l'éq. (7.1) est utilisé. Dans le cas de l'éq. (7.25), le vecteur  $\xi$  recueille toutes les variables d'optimisation, c'est-à-dire tous ou partie des paramètres (de nature différente) définissant la forme de la surface NURBS. Une brève description des deux étapes caractérisant le problème de reconstruction de surfaces est fournie ci-dessous.

### 7.3.1 Paramétrage de Surfaces

Le paramétrage est le processus de maillage d'une surface sur une région du plan paramétrique. Lorsque le formalisme NURBS de l'éq. (7.25) est utilisé, cette phrase signifie déterminer les valeurs appropriées pour les paramètres adimensionnels  $u_1$  et  $u_2$  pour un point appartenant à la surface NURBS à mettre en relation au TP  $\mathbf{Q}_k$ .

Le paramétrage le plus simple est la *projection* [68] mais il ne fonctionne que pour des surfaces non pliées très simples. Si l'on considère des nuages de TPs plus complexes,

l'algorithme développé par Floater dans [111] peut être utilisé pour les surfaces ouvertes. L'avantage de l'algorithme de Floater est que le paramétrage résultant imite la topologie physique de la surface utilisée pour approximer les TPs (*shape preserving parametrisation*).

Un sujet de recherche important consiste à trouver une division correcte des nuages de TPs fermés et à l'approcher par des « patches » de surfaces ouvertes [165]. Le véritable défi de cette approche consiste à imposer des conditions de continuité appropriées sur les limites des patches.

Par contre, dans le cas de surfaces fermées, on pourrait chercher à approximer les TPs avec une entité unique. Lorsque la surface est du genre zéro, l'approximation peut être effectuée en utilisant une projection de Mercator particulière, dans laquelle chaque TP est associé à des angles de latitude et de longitude spécifiques [150]. Le problème devient beaucoup plus complexe lorsque la topologie de la surface présente des trous. Une stratégie générale peut être la cartographie conforme globale proposée dans [110].

Dans ce qui suit, on suppose qu'un paramétrage approprié est fourni par des routines externes.

### 7.3.2 Fitting de Surfaces

Contrairement à la stratégie innovante de fitting de courbes décrite dans la section 7.2, l'approximation de surfaces proposée ci-dessous est effectuée dans le cadre d'un algorithme purement basé sur le gradient. Pour un nuage de TPs donné, on choisit tout d'abord une simple surface B-Spline : les seules variables de conception sont les coordonnées des points de contrôle. Une fois les coordonnées des points de contrôle ont été identifiées, une deuxième étape d'optimisation est effectuée afin d'affiner la qualité de la solution (voir Algorithme 1).

La solution du problème (7.26) est donnée par l'éq. (7.28), où  $\mathbf{X}_p^j$  est le vecteur contenant les coordonnées  $x_j$  des points de contrôle, tandis que  $\bar{\mathbf{X}}_Q^j$  est le vecteur des coordonnées de TPs selon la direction  $x_j$ . La matrice  $[\mathbf{D}] \in \mathbb{R}^{(n_1+1)(n_2+1) \times (\mu+1)}$  est définie comme à l'éq. (7.29).

Par conséquent, la solution du problème (7.26) est simplement obtenue en inversant la matrice  $([\mathbf{D}][\mathbf{D}]^T)$  et en appliquant l'éq. (7.30).

Le CNLPP (7.27) doit être résolu par un algorithme d'optimisation de ceux proposés dans le chapitre 3 en raison de la contribution non linéaire des poids NURBS, qui sont inclus parmi les variables de conception. En effet, le gradient de la fonction objectif par rapport aux poids peut être fourni par éq. (7.31).

### 7.3.3 Résultats du Fitting de Surfaces

Les résultats sur l'approximation de surfaces sont présentés ici sur certains benchmarks pris de la littérature [160]. Les données concernant chaque cas d'étude sont reportées dans le Tableau 7.7. Les résultats sont indiqués sur les figs. 7.13-7.16.

Afin de mettre en évidence les effets de la seconde étape dans l'algorithme d'optimisation, des résultats numériques ont été collectés dans le Tableau 7.8 et la fonction objectif à la fin de la première étape d'optimisation ( $f_{obj-A}$ ) est comparé à celui résultant du second ( $f_{obj-B}$ ).

Comme il peut être facilement remarqué par les résultats numériques, l'effet des poids NURBS est plus évident lorsque la forme de la surface approximante devient plus complexe. En fait, le rapport entre  $f_{obj-B}$  et  $f_{obj-A}$  prend la plus petite valeur pour la surface du trèfle et la plus élevée pour le paraboloïde simple. Bien sûr, si un nombre plus élevé de points de contrôle est utilisé, l'impact des poids sera moins important.

## 7.4 Conclusions et Perspectives sur les problèmes d'approximation

Dans ce chapitre, les problèmes d'approximation de courbes et de surfaces ont été présentés. En particulier, une formalisation mathématique générale du problème de fitting de courbes ainsi qu'une procédure d'optimisation originale pour effectuer la recherche de la solution dans le cadre des courbes NURBS ont été discutées. L'approche proposée repose sur les fonctionnalités suivantes.

1. Une nouvelle expression de la fonction objectif, associée à une contrainte appropriée sur la valeur maximale de la courbure, a été introduite. Ces modifications impliquent une restriction sur les variables de conception discrètes définissant la forme de la courbe NURBS. De plus, le problème est déclaré comme un CNLPP dans lequel le nombre de variable de conception est lui-même une variable de conception.
2. La non-convexité du problème, associée à un domaine de définition de dimension variable, justifie l'utilisation de méthodes numériques non standard. À cette fin, la recherche de solution du problème d'approximation de courbes est effectuée au moyen d'un outil d'optimisation hybride (un GA couplé à une méthode basée sur le gradient).

La méthode présentée permet d'adapter la courbe approximante au niveau du lissage souhaité, défini par la contrainte de courbure : en fait, l'algorithme est capable d'approximer avec succès à la fois courbes lisses et celles présentant une discontinuité des dérivées.

D'autre part, le problème de l'approximation de surfaces a également été résolu. Le paramétrage de la surface constitue l'aspect le plus difficile lorsque le problème de la

reconstruction de surface est abordé dans des applications d'ingénierie réelles. Cependant, à partir des résultats préliminaires présentés dans ce chapitre sur le fitting de surfaces dans le cadre NURBS, certains points intéressants méritent une attention particulière et constituent les principaux besoins pour des recherches futures.

- Une formulation du problème d'approximation de surfaces dans le cas le plus général, c'est-à-dire en considérant comme variables de conception tous les paramètres définissant la forme d'une surface NURBS, n'existe pas. De manière analogue, les composants des *knot vectors* sont choisis uniformément dans l'intervalle  $[0, 1]$ .
- L'importance des poids dans le fitting de surfaces doit être étudiée plus précisément. Les poids sont-ils vraiment importants pour l'approximation de surfaces ou est-il préférable d'augmenter le nombre de points de contrôle ?

Bien entendu, l'élaboration d'une stratégie plus générale pour l'approximation de surfaces, similaire à celle proposée pour l'approximation des courbes, permettrait de réduire le nombre de décisions arbitraires généralement prises par le concepteur lorsqu'il s'agit de résoudre le problème de la reconstruction de surfaces. Cet outil ne serait complet qu'en étudiant plus en détail les points suivants.

- a) Une contrainte appropriée doit être formulée sur la courbure maximale admissible de la surface, comme cela a été fait pour les courbes. Cette contrainte est supposée jouer un rôle important en évitant des degrés de fonctions de base trop élevés.
- b) Des recherches sont en cours pour inclure une stratégie automatique ou semi-automatique de paramétrage de surfaces. L'algorithme d'approximation de surfaces proposé peut être couplé à la méthode de cartographie de Mercator pour les surfaces fermées ou à la méthode de cartographie conforme pour les surfaces de genre 1. Une extension naturelle aux surfaces de genre  $N$  est possible mais la recherche devrait être orientée vers l'étude de la topologie, de la géométrie différentielle (en mettant l'accent sur la géométrie Riemannienne) et de l'analyse complexe.
- c) Finalement, la technique des « patches » peut être utilisée pour approximer des composants mécaniques (par exemple le résultat obtenu dans le cadre d'un problème d'OT). Dans ce cas, la méthode shape preserving de Floater pour les surfaces ouvertes peut être utilisée pour le paramétrage de chaque surface composant la frontière de la pièce. La méthode doit pouvoir imposer automatiquement les conditions de continuité souhaitées entre les patches (continuité simple, continuité tangentielle, continuité de courbure). Cette méthode est mathématiquement moins élégante mais elle serait extrêmement intuitive pour le concepteur d'un point de vue pratique.



# Conclusions and Perspectives

Detailed conclusions and perspectives concerning each topic addressed in this manuscript have been provided at the end of each Chapter. More wide-ranging considerations are discussed in this last Chapter about the conclusions that can be deduced from this PhD thesis and the main general ideas for future researches.

## General Conclusions

The work related to this PhD thesis was born in the framework of the FUTURPROD project but the proposed design methodologies are not circumvented to the SLM technology.

The main contribution of this work is the formulation of the well-known SIMP method by using NURBS surfaces and hyper-surfaces to solve TO problems. In particular, a precise mathematical form is given to the pseudo-density field describing the topology of the structure. The choice of the NURBS blending functions has been driven by the needs expressed through the main objective of this thesis, i.e. providing an innovative and effective TO tool, capable of improving the integration of design and TO for structures to be realised by means of ALM. The design phase takes naturally place in a CAD environment, whilst the TO task is usually performed through a dedicated FE-based software.

It has been proven in Chapter 4 that, thanks to the NURBS entities intrinsic CAD-compatibility and to their mathematical properties, the NURBS-based SIMP algorithm exhibits some advantages when compared to well-established commercial TO software. The implicitly defined filter zone and the NURBS local support property allow for variables saving and computational efficiency. Furthermore, the geometry reconstruction phase (postprocessing of the TO analysis) is completely automatic and it can be performed through trivial operations in CAD environment for 2D structures. The results of this work mainly aim at showing that a NURBS-based formulation is possible for TO problems; in this background, an extensive and detailed sensitivity analysis to the NURBS discrete parameters (number of control points and blending functions degrees) has been carried out. The purpose is to make the designer aware of the effects of these parameters on the final topology and on the performances of the structures.

As it has been frequently stated in this document, the importance of the NURBS formulation for TO problems is mainly related to the possibility of having a purely geometric description of the topology and its boundary during the whole process. When the pseudo-density field is described by means of a NURBS geometric entity, the final optimised topology is totally unrelated from the mesh of the FE model. Being the geometric entity defined on the whole design domain, it can be properly used to establish suitable geometric constraints, as shown in Chapter 5. This idea is put into practice in formulating the minimum length scale requirement, that can be effectively imposed without the introduction of any particular (and problem-dependent) numerical artifact. In fact, the minimum length scale is handled by properly modifying the NURBS discrete parameters. Moreover, a specific sensitivity analysis of the knot vectors distribution on the final topology has been performed.

It is evident that the new formulation of the SIMP method in the NURBS framework presents several similarities to the LSM for TO when the updating procedure is not the HJ equation, but it is carried out by means of mathematical programming techniques. In particular, a first effort has been done in this work in order to take into account a local quantity, as the curvature radius. Usually, the SIMP method results very efficient when global quantities are considered in both objective and constraints functions (e.g. the compliance) or when particular constraints can be formulated in global sense (as the maximum length scale). However, the standard SIMP method cannot successfully handle local quantities because the local value is flawed by the non-realistic geometry representation due to the bad quality of the geometry description: the geometry is just given in a discretised form by means of the mesh elements and its boundary is described through the FE contour, that presents, at least, tangent discontinuities. Even if it is premature to affirm that the NURBS-based SIMP method is as able as the LSM to deal with local quantities/phenomena, a first example has been successfully provided with the implementation of the curvature radius constraint.

The developments discussed in this manuscript are not restrained only to consider the compliance of the structures in the NURBS-based TO algorithm. Other mechanical quantities have been taken into account, namely linear buckling loads and eigen-frequencies. The most important difficulty to be faced is the derivatives provision. Eigenvalues problems deserve a particular attention when the sensitivity analysis is provided through a commercial FE software, without a direct access to information like shape functions, Gauss points, geometric stiffness matrix assembly, etc.: in this case, the derivatives should be given in a special form, as it has been done for buckling loads and eigen-frequencies in Chapter 6. Often, an energetic (weak) form of the state equations gives the solution to this issue, but the procedure is not general. For instance, in the case of buckling loads, the derivatives expression can be given only by making some hypotheses. In any case, it has been shown that considering buckling loads and eigen-frequencies is possible in the NURBS-based algorithm and the underlined difficulties do not arise from the introduction

of NURBS entities in the TO problem.

Furthermore, it has not been possible to completely satisfy the required CAD-compatibility when performing the TO of 3D structures. The major struggles about this topic have been mentioned in Chapter 4 and further detailed in Chapter 7. Particularly, the surface parametrisation (or mapping) presents significant mathematical difficulties, which could not be effectively solved in the framework of this thesis. However, an important contribution has been made in the curve fitting context by proposing an innovative methodology, capable of automatically determining both the continuous and the discrete parameters of the approximating NURBS curve. To this purpose, the curve fitting problem has been reformulated as a CNLPP with variable number of variables and solved through a special hybrid (genetic and gradient-based) algorithm, described in Chapter 7. This one can be seen as a complement to the postprocessing phase of the algorithm *SANTO*, to perform the geometric reconstruction of optimised components.

In this document, the effectiveness of both the algorithm *SANTO* and the curves/surfaces fitting tools has been shown on more or less complicated benchmark. However, they have been applied to more challenging problems as well: in particular, some industrial applications have been faced in the framework of the FUTURPROD project. The results for both the TO algorithm and for the fitting tools are encouraging and no evident limitations or insurmountable difficulties have been found. Of course, the current versions of the algorithms allow the external user to deal with problems of industrial difficulties, but the versatility and the user-friendliness need to be improved.

Finally, it is noteworthy that all the thesis work has been carried out in order to fulfil the objectives stated in the introduction, from both an academic and industrial viewpoint. Therefore, on the one hand, an innovative TO algorithm has been set up with the precise purpose to provide an effective and general tool to perform TO. The designer is guided by robust optimisation algorithms in taking decisions and the arbitrariness is strongly reduced. On the other hand, the aim is not to develop a “black box” tool. The designer should be always aware of the effects of changing some parameters in both TO and in the curve/surface reconstruction frameworks. Future improvements on the integration of TO in the design process are possible only by suitably combining optimisation strategies and the knowledge and the awareness of the designer.

## Perspectives of this Thesis

Results obtained in this thesis are far from being exhaustive on the topic of TO in the NURBS framework. Indeed, many questions naturally follow from this work and they constitute just as many opportunities for further researches.

There is no doubt that the integration of design and TO is not complete yet. The basic step of providing optimised 3D structures in a CAD-compatible format has been only partially faced in this work and the main difficulties have been pointed out. As



previously told, the automatic (or semi-automatic) reconstruction of the boundary of an optimised 3D component can be carried out through two techniques. The first one consists of exploiting the NURBS hyper-surface intrinsic parametrisation: after the threshold operation, the surfaces constituting the boundary of the optimised component are retrieved and suitable techniques should be developed to deduce their parametrisation from the dimensionless parameters  $u_1$ ,  $u_2$  and  $u_3$  and from the hyper-surface itself. The second possibility is to improve the state of the art on the topic of surface fitting by using parametrisation techniques discussed in Chapter 7 (e.g. by performing a special “patch by patch” surface fitting with the required continuity conditions). In any case, the mathematics behind this topic is, on the one hand, really powerful and, on the other hand, extremely tricky to deal with. In the light of the studies presented in this manuscript, it can be stated that concrete improvements in the integration of TO and design (i.e. CAD-compatibility) can be achieved only by means of the collaboration between engineers and mathematicians.

The second open problem is the integration of ALM oriented constraints in TO. In this PhD thesis, the constraints developed in the framework of the algorithm *SANTO* are general geometric constraints. Although they are basic, the minimum / maximum length scale or the minimum curvature radius are not necessarily related to ALM requirements. Concerning more ALM oriented constraints, the most of the efforts in literature are focused on reducing or eliminating the support material in producing ALM structures. Indeed, it has been noted that the topologies proposed by algorithms when the support material constraint is active are very far from the ideal optimum (without the support material constraint). Moreover, the support volume is not the only critical aspect in ALM, because of the intrinsic complexity of the technology. Therefore, there are two main difficulties in formulating suitable constraints for ALM. Firstly, a ranking of priorities should be clearly provided to the designer: this is often an underestimated point because it is not always clear which aspect is more important (performances, costs minimisation, minimisation of support material, minimisation of the production time, etc.). Moreover, the decision is not unique and it strongly depends on the problem at hand. Once the driving criteria have been chosen, the second difficulty must be faced: each technological requirement should be suitably translated in a meaningful constraint for the TO algorithm. The mathematical formulation should be representative enough of the need expressed by the requirement at hand and, meanwhile, it should not introduce too strong non-linearities. On top of that, the constraint formulation for a gradient-based TO algorithm should take into account the sensitivity analysis: the derivatives formulae must be provided in a reasonably straightforward form. However, it can be remarked that even an apparently simple constraint can introduce non-linearities and can make a convex problem non-convex. Research is ongoing on these topics: a particular attention should be dedicated to select those requirements needing a constraint formulation and those that can be handled *a priori*, by properly setting some optimisation parameters or by properly modifying, for instance, the part orientation in the machine.

From the previous considerations, it is evident that the impact of the parameters intervening in TO must be clarified. Particularly, when a TO problem is stated in the NURBS framework, a significant amount of free parameters is added (e.g. number of control points, blending function degrees, knot vectors components distribution). As it has been done for the minimum length scale in Chapter 5, the optimisation constraints should be developed by directly considering the NURBS parameters: only in this way, these quantities can gain a physical meaning and, therefore, can be better controlled by the designer. Otherwise, their role is meaningful only from a mathematical viewpoint and their impact on the final topology cannot be easily forecast. This aspect can be seen as a shortcoming introduced by the NURBS formalism. Indeed, it is just the price to pay when disposing of a versatile geometric entity defined on the whole design domain. Further, the NURBS-based representation offers much more advantages, as briefly anticipated in Chapter 5. The possibility of distributing the knot vectors components according to specific needs and in order to locally control the minimum length scale is just an application. It can be imagined to perform the TO of gradient structures by precisely indicate the portion of design domain wherein thinner elements should appear.

The effectiveness of the presented algorithm must be improved by enhancing the library of mechanical quantities. Particularly, in this manuscript, stiffness, buckling loads and eigen-frequencies have been considered. A key criterion that must be developed is, of course, a strength constraint in the NURBS-based approach. Constraints like those on the maximum Von Mises stress already exist in standard density-based TO problems. However, the Von Mises stress constraint suffers from two main drawbacks: the first one is that, when the standard SIMP method is used, the stress values close to a hole (transition from high to low density regions) may be not representative of the actual stress in the mechanical component. The second issue is related to the calculation of the Von-Mises stress gradient when using FE commercial software, which could be a really hard task. Overcoming these two issues is one of the most important topic of the ongoing research.

Once the effectiveness of the NURBS-based approach will be definitively proven, more complicated and challenging problems can be addressed. For instance, multi-physics and multi-scale problems can be stated in the NURBS framework. In multi-scale problems, the difficulty relies on the sensitivity analysis that must take into account the effects of the NURBS established on the lower scale to evaluate the impact on the upper scale.

All the aforementioned aspects constituting the future perspectives related to this work are supposed to be merged in a unique and powerful optimisation tool, capable of definitively bridge TO and design. This objective can be achieved in general terms, not only in the context of structures produced through ALM. In order to make all the optimisation tools robust and suitable for industrial applications, *ad hoc* Graphic User Interfaces (GUIs) should be developed to let the designer immediately intervene and set the optimisation/tools parameters in direct and nearly interactive way.



# Conclusions et Perspectives

Des conclusions détaillées et des perspectives concernant chaque sujet abordé dans ce manuscrit ont été fournies dans les chapitres respectifs. Dans ce dernier chapitre, des réflexions plus larges sont discutées sur les conclusions qui peuvent être déduites de cette thèse et les principales idées générales pour des recherches futures.

## Conclusions Générales

Le travail lié à cette thèse est né dans le cadre du projet FUTURPROD mais les méthodologies de conception proposées ne sont pas contournées à la technologie SLM.

La principale contribution de ce travail est la formulation de la méthode SIMP bien connue en utilisant des entités NURBS (surfaces et hyper-surfaces) pour résoudre les problèmes d'OT. En particulier, une forme mathématique précise est donnée au champ de pseudo-densité décrivant la topologie de la structure. Le choix des fonctions de base NURBS a été dicté par les besoins exprimés à travers l'objectif principal de cette thèse, à savoir fournir un outil d'OT innovant et efficace, capable d'améliorer l'intégration de la conception et de l'OT pour les structures à réaliser par ALM. La phase de conception se déroule naturellement dans un environnement CAO, tandis que la tâche d'OT est généralement exécutée via un logiciel basé sur les EF dédié.

Il a été prouvé dans le chapitre 4 que, grâce à la compatibilité CAO intrinsèque des entités NURBS et à leurs propriétés mathématiques, l'algorithme SIMP basé sur les NURBS présente certains avantages par rapport à des logiciels commerciaux bien établis. La zone de filtre implicitement définie et la propriété de support local des NURBS permettent de sauvegarder les variables et d'améliorer l'efficacité informatique. De plus, la phase de reconstruction géométrique (post-traitement de l'OT) est entièrement automatique et peut être réalisée via des opérations triviales dans un environnement CAO pour des structures 2D. Les résultats de ce travail visent principalement à montrer qu'une formulation basée sur les NURBS est possible pour les problèmes d'OT ; dans ce contexte, une analyse de sensibilité détaillée des paramètres discrets des entités NURBS (nombre de points de contrôle et degrés de fonctions de base) a été effectuée. L'objectif est de sensibiliser le concepteur aux effets de ces paramètres sur la topologie finale et sur les performances des structures.

Comme cela a été fréquemment indiqué dans ce document, l'importance de la formulation NURBS pour les problèmes d'OT est principalement liée à la possibilité d'avoir une description purement géométrique de la topologie et de sa frontière pendant tout le processus. Lorsque le champ de pseudo-densité est décrit au moyen d'une entité géométrique NURBS, la topologie optimisée finale est totalement indépendante du maillage du modèle EF. En tant qu'entité géométrique définie sur l'ensemble du domaine de conception, elle peut être utilisée correctement pour satisfaire des contraintes géométriques appropriées, comme indiqué au chapitre 5. Cette idée est mise en pratique lors de la formulation de la contrainte d'épaisseur minimale, qui peut être effectivement imposée sans introduction d'artefact numérique particulier (et dépendant du problème). En fait, l'épaisseur minimale est gérée en modifiant correctement les paramètres discrets NURBS. De plus, une analyse de sensibilité spécifique de la distribution des *knot vectors* sur la topologie finale a été effectuée.

Il est évident que la nouvelle formulation de la méthode SIMP dans le cadre NURBS présente plusieurs similitudes avec la LSM pour l'OT lorsque la procédure de mise à jour n'est pas l'équation d'HJ mais réalisée au moyen de techniques de programmation mathématique. En particulier, un premier effort a été effectué dans ce travail afin de prendre en compte une quantité locale, comme le rayon de courbure. Habituellement, la méthode SIMP est très efficace lorsqu'une quantité globale est prise en compte à la fois dans les fonctions objectif et dans les contraintes (par exemple la compliance). Cependant, la méthode SIMP standard ne peut pas gérer avec succès les quantités locales car la valeur locale est faussée par la représentation géométrique non réaliste (la géométrie est simplement donnée sous forme discrète par les éléments de maillage et sa frontière est décrite par le contour des EF, qui présente au moins des discontinuités de tangence). Même s'il est prématuré d'affirmer que la méthode SIMP basée sur les NURBS est aussi capable que la LSM de traiter des quantités / phénomènes locaux, un premier exemple a été fourni avec succès avec l'implémentation de la contrainte de rayon de courbure.

Les développements discutés dans ce manuscrit ne sont pas limités seulement à considérer la compliance des structures dans l'algorithme d'OT basé sur les NURBS. D'autres grandeurs mécaniques ont été prises en compte, à savoir les charges de flambage linéaires et les fréquences propres. La principale difficulté à surmonter est constituée par les dérivées. Les problèmes de valeurs propres méritent une attention particulière lorsque l'analyse de sensibilité est fournie par un logiciel commercial, sans accès direct à des informations telles que les fonctions de forme, les points de Gauss, l'assemblage de la matrice de rigidité géométrique, etc.. Dans ce cas, les dérivées doivent être fournies sous une forme spéciale, comme il a été fait pour les charges de flambage et les fréquences propres dans le chapitre 6. Souvent, une forme énergétique (faible) des équations d'état donne la solution à ce problème, mais la procédure n'est pas générale. Par exemple, dans le cas de charges de flambage, l'expression des dérivées ne peut être donnée qu'en faisant des hypothèses. En tout état de cause, il a été démontré que l'algorithme basé sur les NURBS permet de

gérer des charges de flambage et des fréquences propres et que les difficultés soulignées ne découlent pas de l'introduction d'entités NURBS dans le problème d'OT.

De plus, il n'a pas été possible de satisfaire complètement la compatibilité CAO requise lors de l'exécution de l'OT des structures 3D. Les principaux problèmes sur ce sujet ont été mentionnés dans le chapitre 4 et plus détaillés dans le chapitre 7. En particulier, le paramétrage de la surface présente des difficultés mathématiques importantes, qui ne peuvent être résolues efficacement dans le cadre de cette thèse. Cependant, une contribution importante a été apportée dans le contexte de l'approximation de courbes en proposant une méthodologie innovante, capable de déterminer automatiquement à la fois les paramètres continus et discrets de la courbe NURBS approximante. À cette fin, le problème d'approximation de courbes a été reformulé en tant que CNLPP avec un nombre variable de variables et résolu grâce à un algorithme hybride (génétique et à gradient), décrit au chapitre 7. Celui-ci peut être vu comme un complément à la phase de post-traitement de l'algorithme *SANTO*, pour effectuer la reconstruction géométrique des pièces optimisées.

Dans ce document, l'efficacité de l'algorithme *SANTO* et des outils d'approximation de courbes / surfaces a été montrée sur des benchmarks plus ou moins complexes. Toutefois, ils ont également été appliqués à des problèmes plus difficiles : notamment, certaines applications industrielles ont été rencontrées dans le cadre du projet FUTURPROD. Les résultats pour l'algorithme d'OT et pour les outils d'approximation sont encourageants et aucune limitation évidente ou difficulté insurmontable n'a été trouvée. Bien entendu, les versions actuelles des algorithmes permettent à l'utilisateur externe de résoudre les problèmes de difficultés industrielles, mais leur polyvalence et l'ergonomie doivent être améliorées.

Enfin, il convient de noter que tous les travaux de cette thèse ont été réalisés dans le but de satisfaire les objectifs énoncés dans l'introduction, tant du point de vue académique qu'industriel. Par conséquent, un algorithme innovant d'OT a été mis au point dans le but précis de fournir un outil efficace et général. Le concepteur est guidé par des algorithmes d'optimisation robustes pour prendre des décisions et l'arbitraire est fortement réduit. Par contre, l'objectif n'est pas de développer un outil de type «boîte noire». Le concepteur doit toujours être conscient des effets de la modification de certains paramètres à la fois dans l'OT et dans les cadres de reconstruction de courbes / surfaces. Des améliorations futures sur l'intégration de l'OT dans le processus de conception ne sont possibles qu'en combinant de manière appropriée les stratégies d'optimisation et les connaissances et la sensibilité du concepteur.

## Les Perspectives de la Thèse

Les résultats obtenus dans cette thèse sont loin d'être exhaustifs sur le sujet de l'OT dans le cadre des NURBS. En effet, de nombreuses questions découlent de ce travail et

constituent autant d'opportunités pour d'autres recherches.

Il ne fait aucun doute que l'intégration de la conception et de l'OT n'est pas encore complète. L'étape de base consistant à fournir des structures 3D optimisées dans un format compatible avec la CAO n'a été que partiellement traitée dans ce travail et les principales difficultés ont été mises en évidence. Comme indiqué précédemment, la reconstruction automatique (ou semi-automatique) de la frontière d'une part 3D optimisée peut être réalisée à l'aide de deux techniques. La première consiste à exploiter le paramétrage intrinsèque de l'hyper-surface NURBS : après l'opération de seuil, les surfaces constituant la frontière du composant optimisé sont récupérées et des techniques appropriées doivent être développées pour en déduire leur paramétrage à partir des paramètres sans dimension  $u_1$ ,  $u_2$  et  $u_3$  et de l'hyper-surface elle-même. La deuxième possibilité est d'améliorer l'état de l'art en matière d'approximation de surfaces en utilisant les techniques de paramétrage décrites au chapitre 7 (par exemple, en effectuant une approximation de surfaces par patches avec conditions de continuité spécifiques). Dans tous les cas, le mathématique sous-jacente est, d'une part, très puissante et, d'autre part, extrêmement difficile à traiter. À la lumière des études présentées dans ce manuscrit, on peut affirmer que des améliorations concrètes dans l'intégration de l'OT et de la conception (compatibilité avec la CAO) ne peuvent être réalisées que grâce à la collaboration entre ingénieurs et mathématiciens.

Le deuxième problème ouvert est l'intégration des contraintes orientées vers l'ALM dans l'OT. Dans cette thèse, les contraintes développées dans le cadre de l'algorithme *SANTO* sont des contraintes géométriques générales. Bien qu'elles soient basiques, l'épaisseur minimale / maximale ou le rayon minimal de courbure ne sont pas nécessairement liés aux exigences de l'ALM. En ce qui concerne les contraintes plus orientées vers la fabrication additive, la plupart des efforts en littérature se concentrent sur la réduction ou l'élimination du matériau de support dans la production. En effet, il a été constaté que les topologies proposées par les algorithmes lorsque la contrainte du matériau de support est active sont très loin de l'optimum idéal (sans contrainte de matériau de support). De plus, le volume de support n'est pas le seul aspect critique de l'ALM, en raison de la complexité intrinsèque de la technologie. Donc deux difficultés principales existent dans la formulation de contraintes appropriées. Tout d'abord, un classement des priorités doit être clairement fourni au concepteur : il s'agit souvent d'un point sous-estimé car il n'est pas toujours clair quel aspect est le plus important (performances, minimisation des coûts, minimisation du temps de production, etc.). De plus, la décision n'est pas unique et dépend fortement du problème à résoudre. Une fois que les critères ont été choisis, la deuxième difficulté doit être surmontée : chaque exigence technologique doit être traduite de manière appropriée en une contrainte significative pour l'algorithme d'OT. La formulation mathématique devrait être suffisamment représentative du besoin exprimé par l'exigence en question et, dans l'intervalle, elle ne devrait pas introduire de non-linéarités trop fortes. De plus, la formulation des contraintes pour un algorithme d'OT basé sur un gradient devrait prendre en compte le calcul des dérivées : de telles formules doivent être fournies sous une forme as-

sez simple. Cependant, on peut remarquer que même une contrainte apparemment simple peut introduire des non-linéarités et peut rendre un problème convexe non-convexe. Des recherches sont en cours sur ces sujets : une attention particulière devrait être accordée à la sélection des exigences nécessitant une formulation de contraintes et celles pouvant être traitées *a priori*, en définissant correctement certains paramètres d'optimisation ou en modifiant, par exemple, l'orientation de la pièce dans la machine.

D'après les considérations précédentes, il est évident que l'impact des paramètres intervenant dans l'OT doit être clarifié. En particulier, lorsque l'OT est indiqué dans le cadre des NURBS, une quantité importante de paramètres libres est ajoutée (par exemple, le nombre de points de contrôle, le degré des fonctions de base, la distribution des composantes des *knot vectors*). Comme cela a été fait pour l'épaisseur minimale dans le chapitre 5, les contraintes d'optimisation doivent être développées en considérant directement les paramètres NURBS : de cette manière seulement, ces quantités peuvent avoir une signification physique et, par conséquent, ils peuvent être mieux contrôlés par le concepteur. Sinon, leur rôle n'a de sens que dans un contexte de point de vue mathématique et leur impact sur la topologie finale ne peut pas être facilement prévisible. Bien entendu, cet aspect peut être considéré comme une lacune introduite par le formalisme NURBS. En effet, il ne s'agit que du prix à payer pour disposer d'une entité géométrique polyvalente définie sur l'ensemble du domaine de conception. De plus, la représentation basée sur les NURBS offre beaucoup plus d'avantages, comme vu brièvement dans le chapitre 5. La possibilité de distribuer les composants des *knot vectors* en fonction des besoins spécifiques et de contrôler localement l'épaisseur minimale n'est qu'une application. On peut imaginer effectuer l'OT des « gradient structures » en indiquant précisément la partie du domaine de conception dans laquelle des éléments plus fins doivent apparaître.

De plus, l'efficacité de l'algorithme présenté doit être améliorée en élargissant la bibliothèque de grandeurs mécaniques. En particulier, dans ce manuscrit, la rigidité, les charges de flambage et les fréquences propres ont été prises en compte. Un critère clé qui doit être développé dans l'approche basée sur les NURBS concerne, bien sûr, la résistance. Des contraintes telles que celles sur la contrainte maximale de Von Mises existent déjà dans les problèmes d'OT standard basés sur la densité. Cependant, la contrainte de Von Mises présente deux inconvénients principaux : le premier est que, lorsque la méthode SIMP standard est utilisée, les valeurs de contrainte proches d'un trou (transition des régions de haute à basse densité) peuvent ne pas être représentatives de la contrainte dans la pièce mécanique. Le second problème est lié au calcul du gradient de la contrainte de Von-Mises lors de l'utilisation de logiciels commerciaux aux EF, ce qui pourrait être une tâche très difficile. Surmonter ces deux problèmes est l'un des thèmes les plus importants de la recherche en cours.

Une fois que les aspects susmentionnés seront traités et que l'efficacité de l'approche basée sur les NURBS sera définitivement prouvée, des problèmes plus complexes pourront être résolus. Par exemple, les problèmes multi-physiques et multi-échelles peuvent être



énoncés dans le cadre des NURBS. Dans les problèmes multi-échelles, la difficulté repose sur l'analyse de sensibilité qui, bien entendu, doit prendre en compte les effets des NURBS établis à l'échelle inférieure pour évaluer l'impact sur l'échelle supérieure.

Tous les aspects précités constituant les perspectives liées à ce travail sont supposés être intégrés dans un outil d'optimisation unique et puissant, capable de relier définitivement l'OT et la conception. Cet objectif peut être atteint en termes généraux, pas seulement dans le contexte des structures produites par ALM. Afin de rendre tous les outils d'optimisation robustes et adaptés aux applications industrielles, des interfaces graphiques *ad hoc* doivent être développées pour permettre au concepteur d'intervenir immédiatement et de définir les paramètres d'optimisation / ou des entités NURBS de manière quasi interactive.

# Appendix A

## NURBS entities : derivatives computation

In order to compute  $\frac{\partial \rho_e}{\partial \Xi_\tau^{(1)}}$  and  $\frac{\partial \rho_e}{\partial \Xi_\tau^{(2)}}$ , the quantity  $B_k^e$  is introduced as

$$B_j^e = \prod_{l=1}^N N_{i_l, p_l}(u_l^e), \quad (1)$$

and, consequently, the general expression of a NURBS entity in terms of linear indices is

$$\rho_e = \frac{\sum_{j=1}^{n_{tot}} B_j^e \Xi_j^{(2)} \Xi_j^{(1)}}{\sum_{j=1}^{n_{tot}} B_j^e \Xi_j^{(2)}}, \quad (2)$$

where the explicit dependence on the dimensionless parameters is omitted for the sake of simplicity. Therefore, the derivative  $\frac{\partial \rho_e}{\partial \Xi_\tau^{(1)}}$  trivially reads

$$\frac{\partial \rho_e}{\partial \Xi_\tau^{(1)}} = \frac{B_\tau^e \Xi_\tau^{(2)}}{\sum_{j=1}^{n_{tot}} B_j^e \Xi_j^{(2)}} = R_\tau^e. \quad (3)$$

The derivative  $\frac{\partial \rho_e}{\partial \Xi_\tau^{(2)}}$  requires to be handled before achieving a synthetic form :

$$\begin{aligned} \frac{\partial \rho_e}{\partial \Xi_\tau^{(1)}} &= \frac{B_\tau^e \Xi_\tau^{(1)} \left( \sum_{j=1}^{n_{tot}} B_j^e \Xi_j^{(2)} \right) - B_\tau^e \left( \sum_{j=1}^{n_{tot}} B_j^e \Xi_j^{(2)} \Xi_j^{(1)} \right)}{\left( \sum_{j=1}^{n_{tot}} B_j^e \Xi_j^{(2)} \right)^2} = \\ &= \frac{\Xi_\tau^{(1)}}{\Xi_\tau^{(2)}} \frac{B_\tau^e \Xi_\tau^{(2)}}{\sum_{j=1}^{n_{tot}} B_j^e \Xi_j^{(2)}} - \frac{\rho_e}{\Xi_\tau^{(2)}} \frac{B_\tau^e \Xi_\tau^{(2)}}{\sum_{j=1}^{n_{tot}} B_j^e \Xi_j^{(2)}} = \frac{R_\tau^e}{\Xi_\tau^{(2)}} \left( \Xi_\tau^{(1)} - \rho_e \right). \end{aligned} \quad (4)$$



# Appendix B

## Sensitivity analysis of the minimum member size constraint (Poulsen's formulation)

Sensitivity of the monotonicity integral  $M_{\gamma_i}(\rho)$  with respect to the control points :

$$\begin{aligned}
 \frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_{\tau}^{(1)}} &= \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{\partial}{\partial \Xi_{\tau}^{(1)}} \left( \sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2} \right) - \left( \frac{\partial}{\partial \Xi_{\tau}^{(1)}} \sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2} \right) = \\
 &= \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{(\rho_{j+1} - \rho_j) \left( \frac{\partial \rho_{j+1}}{\partial \Xi_{\tau}^{(1)}} - \frac{\partial \rho_j}{\partial \Xi_{\tau}^{(1)}} \right)}{\sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2}} - \frac{(\rho_{N_{\mathbf{b}_i}} - \rho_1) \left( \frac{\partial \rho_{N_{\mathbf{b}_i}}}{\partial \Xi_{\tau}^{(1)}} - \frac{\partial \rho_1}{\partial \Xi_{\tau}^{(1)}} \right)}{\sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2}} \quad (1) \\
 &= \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{(\rho_{j+1} - \rho_j) (R_{\tau}^{j+1} - R_{\tau}^j)}{\sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2}} - \frac{(\rho_{N_{\mathbf{b}_i}} - \rho_1) (R_{\tau}^{N_{\mathbf{b}_i}} - R_{\tau}^1)}{\sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2}}.
 \end{aligned}$$

Sensitivity of the monotonicity integral  $M_{\gamma_i}(\rho)$  with respect to the NURBS weights :

$$\begin{aligned}
\frac{\partial M_{\mathbf{b}_i}(\rho)}{\partial \Xi_{\tau}^{(2)}} &= \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{\partial}{\partial \Xi_{\tau}^{(2)}} \left( \sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2} \right) - \left( \frac{\partial}{\partial \Xi_{\tau}^{(2)}} \sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2} \right) = \\
&= \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{(\rho_{j+1} - \rho_j) \left( \frac{\partial \rho_{j+1}}{\partial \Xi_{\tau}^{(2)}} - \frac{\partial \rho_j}{\partial \Xi_{\tau}^{(2)}} \right)}{\sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2}} - \frac{(\rho_{N_{\mathbf{b}_i}} - \rho_1) \left( \frac{\partial \rho_{N_{\mathbf{b}_i}}}{\partial \Xi_{\tau}^{(2)}} - \frac{\partial \rho_1}{\partial \Xi_{\tau}^{(2)}} \right)}{\sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2}} \\
&= \frac{1}{\Xi_{\tau}^{(2)}} \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{(\rho_{j+1} - \rho_j) \left( \frac{\partial \rho_{j+1}}{\partial \Xi_{\tau}^{(1)}} (\Xi_{\tau}^{(1)} - \rho_{j+1}) - \frac{\partial \rho_j}{\partial \Xi_{\tau}^{(1)}} (\Xi_{\tau}^{(1)} - \rho_j) \right)}{\sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2}} + \\
&\quad - \frac{1}{\Xi_{\tau}^{(2)}} \frac{(\rho_{N_{\mathbf{b}_i}} - \rho_1) \left( \frac{\partial \rho_{N_{\mathbf{b}_i}}}{\partial \Xi_{\tau}^{(1)}} (\Xi_{\tau}^{(1)} - \rho_{N_{\mathbf{b}_i}}) - \frac{\partial \rho_1}{\partial \Xi_{\tau}^{(1)}} (\Xi_{\tau}^{(1)} - \rho_1) \right)}{\sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2}} = \\
&\quad \frac{\Xi_{\tau}^{(1)}}{\Xi_{\tau}^{(2)}} \left[ \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{(\rho_{j+1} - \rho_j) \left( \frac{\partial \rho_{j+1}}{\partial \Xi_{\tau}^{(1)}} - \frac{\partial \rho_j}{\partial \Xi_{\tau}^{(1)}} \right)}{\sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2}} - \frac{(\rho_{N_{\mathbf{b}_i}} - \rho_1) \left( \frac{\partial \rho_{N_{\mathbf{b}_i}}}{\partial \Xi_{\tau}^{(1)}} - \frac{\partial \rho_1}{\partial \Xi_{\tau}^{(1)}} \right)}{\sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2}} \right] + \\
&\quad + \frac{1}{\Xi_{\tau}^{(2)}} \left[ \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{(\rho_{j+1} - \rho_j) \left( \rho_j \frac{\partial \rho_j}{\partial \Xi_{\tau}^{(1)}} - \rho_{j+1} \frac{\partial \rho_{j+1}}{\partial \Xi_{\tau}^{(1)}} \right)}{\sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2}} - \frac{(\rho_{N_{\mathbf{b}_i}} - \rho_1) \left( \rho_1 \frac{\partial \rho_1}{\partial \Xi_{\tau}^{(1)}} - \rho_{N_{\mathbf{b}_i}} \frac{\partial \rho_{N_{\mathbf{b}_i}}}{\partial \Xi_{\tau}^{(1)}} \right)}{\sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2}} \right] = \\
&\quad = \frac{\Xi_{\tau}^{(1)}}{\Xi_{\tau}^{(2)}} \frac{\partial M_{\gamma_i}(\rho)}{\partial \Xi_{\tau}^{(1)}} + \\
&\quad + \frac{1}{\Xi_{\tau}^{(2)}} \left[ \sum_{j=1}^{N_{\mathbf{b}_i}-1} \frac{(\rho_{j+1} - \rho_j) (\rho_j R_{\tau}^j - \rho_{j+1} R_{\tau}^{j+1})}{\sqrt{(\rho_{j+1} - \rho_j)^2 + \epsilon^2}} - \frac{(\rho_{N_{\mathbf{b}_i}} - \rho_1) (\rho_1 R_{\tau}^1 - \rho_{N_{\mathbf{b}_i}} R_{\tau}^{N_{\mathbf{b}_i}})}{\sqrt{(\rho_{N_{\mathbf{b}_i}} - \rho_1)^2 + \epsilon^2}} \right]. \tag{2}
\end{aligned}$$

# Appendix C

## Sensitivity analysis of the maximum member size constraint

The gradient of the maximum member size constraint is computed here with respect to the NURBS control points and weights :

$$\begin{aligned}
\frac{\partial g_{d_{max}}}{\partial \Xi_\tau^{(1)}} &= \frac{1}{\Omega_e(1-\psi)} \frac{\partial}{\partial \Xi_\tau^{(1)}} \left( \sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi \right)^{\frac{1}{\chi}} = \\
&= \frac{\left( \sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi \right)^{\frac{1}{\chi} - 1}}{\chi \Omega_e(1-\psi)} \left( \sum_{e=1}^{N_e} \frac{\partial}{\partial \Xi_\tau^{(1)}} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi \right) = \\
&= \frac{\left( \sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi \right)^{\frac{1}{\chi}} \left( \sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^{\chi-1} \left( \sum_{i \in \Omega_e} \beta \rho_i^{\beta-1} \frac{\partial \rho_i}{\partial \Xi_\tau^{(1)}} \mu_i \right) \right)}{\Omega_e(1-\psi) \sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi} = \\
\beta(g_{d_{max}} + 1) &= \frac{\sum_{e=1}^{N_e} \left( \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^{\chi-1} \left( \sum_{i \in \Omega_e} \rho_i^{\beta-1} R_\tau^i \mu_i \right) \right)}{\sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi}.
\end{aligned} \tag{1}$$

$$\begin{aligned}
\frac{\partial g_{d_{max}}}{\partial \Xi_\tau^{(2)}} &= \beta(g_{d_{max}} + 1) \frac{\sum_{e=1}^{N_e} \left( \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^{\chi-1} \left( \sum_{i \in \Omega_e} \rho_i^{\beta-1} \frac{\partial \rho_i}{\partial \Xi_\tau^{(2)}} \mu_i \right) \right)}{\sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi} = \\
&= \beta(g_{d_{max}} + 1) \frac{\sum_{e=1}^{N_e} \left( \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^{\chi-1} \left( \sum_{i \in \Omega_e} \rho_i^{\beta-1} \frac{\partial \rho_i}{\partial \Xi_\tau^{(1)}} \frac{\Xi_\tau^{(1)} - \rho_i}{\Xi_\tau^{(2)}} \mu_i \right) \right)}{\sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi} = \\
&= \frac{\Xi_\tau^{(1)}}{\Xi_\tau^{(2)}} \beta(g_{d_{max}} + 1) \frac{\sum_{e=1}^{N_e} \left( \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^{\chi-1} \left( \sum_{i \in \Omega_e} \rho_i^{\beta-1} \frac{\partial \rho_i}{\partial \Xi_\tau^{(1)}} \mu_i \right) \right)}{\sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi} + \\
&- \frac{1}{\Xi_\tau^{(2)}} \beta(g_{d_{max}} + 1) \frac{\sum_{e=1}^{N_e} \left( \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^{\chi-1} \left( \sum_{i \in \Omega_e} \rho_i^\beta \frac{\partial \rho_i}{\partial \Xi_\tau^{(1)}} \mu_i \right) \right)}{\sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi} = \\
\frac{\Xi_\tau^{(1)}}{\Xi_\tau^{(2)}} \frac{\partial g_{d_{max}}}{\partial \Xi_\tau^{(1)}} - \frac{\beta(g_{d_{max}} + 1)}{\Xi_\tau^{(2)}} \frac{\sum_{e=1}^{N_e} \left( \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^{\chi-1} \left( \sum_{i \in \Omega_e} \rho_i^\beta R_\tau^i \mu_i \right) \right)}{\sum_{e=1}^{N_e} \left( \sum_{i \in \Omega_e} \rho_i^\beta \mu_i \right)^\chi}.
\end{aligned} \tag{2}$$

# Appendix D

## Sensitivity analysis of the minimum curvature radius constraint

The gradient of the local curvature radius is computed with respect to the control points and to the weights. Let us write again Eq. (5.29) in a more convenient form :

$$r = -\frac{1}{a_1 a_2} \frac{r_N}{r_D}. \quad (1)$$

Thus, the derivatives write

$$\frac{\partial r}{\partial \Xi_\tau^{(1)}} = r \left( \frac{1}{r_N} \frac{\partial r_N}{\partial \Xi_\tau^{(1)}} - \frac{1}{r_D} \frac{\partial r_D}{\partial \Xi_\tau^{(1)}} \right), \quad (2)$$

$$\frac{\partial r}{\partial \Xi_\tau^{(2)}} = r \left( \frac{1}{r_N} \frac{\partial r_N}{\partial \Xi_\tau^{(2)}} - \frac{1}{r_D} \frac{\partial r_D}{\partial \Xi_\tau^{(2)}} \right). \quad (3)$$

The complete expression of Eqs. (2) and (3) are not provided here for sake of brevity ; anyway the reader can easily deduce them by using the following formulae to compute the derivatives of each term.

$$\frac{\partial}{\partial \Xi_\tau^{(1)}} \left( \frac{\partial \rho}{\partial u_1} \right) = \frac{\partial R_\tau}{\partial u_1}, \quad (4)$$

$$\frac{\partial}{\partial \Xi_\tau^{(1)}} \left( \frac{\partial \rho}{\partial u_2} \right) = \frac{\partial R_\tau}{\partial u_2}, \quad (5)$$

$$\frac{\partial}{\partial \Xi_\tau^{(1)}} \left( \frac{\partial^2 \rho}{\partial u_1^2} \right) = \frac{\partial^2 R_\tau}{\partial u_1^2}, \quad (6)$$

$$\frac{\partial}{\partial \Xi_\tau^{(1)}} \left( \frac{\partial^2 \rho}{\partial u_1 \partial u_2} \right) = \frac{\partial^2 R_\tau}{\partial u_1 \partial u_2}, \quad (7)$$

$$\frac{\partial}{\partial \Xi_\tau^{(1)}} \left( \frac{\partial^2 \rho}{\partial u_2^2} \right) = \frac{\partial^2 R_\tau}{\partial u_2^2}, \quad (8)$$



$$\frac{\partial}{\partial \Xi_\tau^{(2)}} \left( \frac{\partial \rho}{\partial u_1} \right) = \frac{\Xi_\tau^{(1)} - \rho \partial R_\tau}{\Xi_\tau^{(2)}} \frac{\partial R_\tau}{\partial u_1} - \frac{R_\tau}{\Xi_\tau^{(2)}} \frac{\partial \rho}{\partial u_1}, \quad (9)$$

$$\frac{\partial}{\partial \Xi_\tau^{(2)}} \left( \frac{\partial \rho}{\partial u_2} \right) = \frac{\Xi_\tau^{(1)} - \rho \partial R_\tau}{\Xi_\tau^{(2)}} \frac{\partial R_\tau}{\partial u_2} - \frac{R_\tau}{\Xi_\tau^{(2)}} \frac{\partial \rho}{\partial u_2}, \quad (10)$$

$$\frac{\partial}{\partial \Xi_\tau^{(2)}} \left( \frac{\partial^2 \rho}{\partial u_1^2} \right) = \frac{1}{\Xi_\tau^{(2)}} \left( (\Xi_\tau^{(1)} - \rho) \frac{\partial^2 R_\tau}{\partial u_1^2} - 2 \frac{\partial R_\tau}{\partial u_1} \frac{\partial \rho}{\partial u_1} - R_\tau \frac{\partial^2 \rho}{\partial u_1^2} \right), \quad (11)$$

$$\frac{\partial}{\partial \Xi_\tau^{(2)}} \left( \frac{\partial^2 \rho}{\partial u_1 \partial u_2} \right) = \frac{1}{\Xi_\tau^{(2)}} \left( (\Xi_\tau^{(1)} - \rho) \frac{\partial^2 R_\tau}{\partial u_1 \partial u_2} - \frac{\partial R_\tau}{\partial u_1} \frac{\partial \rho}{\partial u_2} - \frac{\partial R_\tau}{\partial u_2} \frac{\partial \rho}{\partial u_1} - R_\tau \frac{\partial^2 \rho}{\partial u_1 \partial u_2} \right), \quad (12)$$

$$\frac{\partial}{\partial \Xi_\tau^{(2)}} \left( \frac{\partial^2 \rho}{\partial u_2^2} \right) = \frac{1}{\Xi_\tau^{(2)}} \left( (\Xi_\tau^{(1)} - \rho) \frac{\partial^2 R_\tau}{\partial u_2^2} - 2 \frac{\partial R_\tau}{\partial u_2} \frac{\partial \rho}{\partial u_2} - R_\tau \frac{\partial^2 \rho}{\partial u_2^2} \right). \quad (13)$$

# Appendix E

## Details of the NURBS curve of the flame problem

The optimised Knot Vector :

[0, 0, 0, 0, 0, 0.0225, 0.0259, 0.0601, 0.0644, 0.0657, **0.0727**, **0.0727**, 0.0881, **0.0894**, **0.0894**, **0.1145**, **0.1145**, 0.1220, 0.1410, **0.1527**, **0.1527**, **0.1527**, 0.1638, 0.1753, 0.1778, 0.1833, 0.1904, **0.1934**, **0.1934**, **0.1934**, 0.2013, 0.2137, 0.2287, 0.2354, 0.2422, 0.2627, 0.2642, **0.2850**, **0.2855**, **0.2939**, **0.2944**, 0.3054, 0.3327, 0.3457, 0.3535, 0.3633, 0.3692, 0.3743, **0.3750**, **0.3750**, 0.3777, 0.3880, 0.3888, **0.3927**, **0.3927**, **0.4161**, **0.4161**, 0.4327, 0.4405, 0.4547, 0.4595, 0.4736, 0.5001, 0.5075, 0.5148, **0.5362**, **0.5364**, **0.5364**, 0.5452, 0.5470, 0.5613, 0.5665, 0.5712, 0.5948, 0.5995, **0.6029**, **0.6032**, 0.6054, 0.6153, 0.6191, 0.6206, 0.6310, 0.6451, 0.6505, 0.6516, 0.6829, 0.6872, 0.7094, 0.7106, 0.7309, 0.7503, 0.7614, 0.7722, **0.7859**, **0.7859**, 0.7964, 0.8005, 0.8087, 0.8237, 0.8414, **0.8482**, **0.8482**, 0.8655, 0.8687, 0.8821, 0.8837, **0.9231**, **0.9231**, 0.9446, 0.9563, 1, 1, 1, 1, 1, ].

The optimised weights vector :

$w = [1.0029, 0.9997, 0.9992, 0.9857, 1.0179, 0.9971, 1.0009, 0.9978, 1.0109, 0.9895, 1.0040, 1.0111, 0.9769, 0.9893, 1.0297, 1.0127, 0.9786, 0.9974, 0.9847, 1.0164, 1.0162, 0.9817, 1.0005, 1.0005, 0.9965, 1.0023, 1.0000, 1.0071, 0.9898, 1.0014, 0.9967, 0.9961, 1.0357, 0.9640, 1.0114, 1.0065, 0.9907, 0.9893, 1.0323, 0.9857, 0.9919, 0.9969, 1.0058, 0.9998, 0.9996, 0.9982, 1.0008, 1.0020, 1.0000, 1.0003, 0.9961, 0.9912, 1.0246, 0.9788, 0.9959, 1.0669, 0.9176, 1.0248, 1.0124, 0.9721, 1.0761, 0.8686, 1.0859, 0.9872, 0.9891, 1.0072, 1.0080, 1.0115, 0.9620, 1.0305, 0.9737, 1.0232, 0.9907, 1.0006, 1.0001, 1.0017, 1.0014, 0.9866, 1.0179, 0.9985, 0.9819, 1.0079, 1.0018, 1.0276, 0.9920, 0.8576, 1.1315, 1.1224, 0.8604, 0.9518, 1.0663, 1.0000, 0.9807, 1.0491, 0.9341, 0.9720, 1.0871, 0.9647, 1.0268, 0.9844, 0.9642, 1.0530, 0.9769, 0.9789, 1.0117, 1.0152, 0.9919, 0.9820, 1.0146, 1.0015, ].$



# Bibliography

- [1] J. A. Sethian. *Level Set Methods and Fast Marching Methods - Evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*. Cambridge University Press, 1999.
- [2] R. Ponche. *Méthodologie de conception pour la fabrication additive, application à la projection de poudres*. PhD thesis, Ecole centrale de Nantes, 2013.
- [3] T. Wohlers. Wohlers report 2015: Additive manufacturing and 3d printing state of the industry annual worldwide progress report. Technical report, Wohlers Associate, 2015.
- [4] S. Yang and Y. F. Zhao. Additive manufacturing-enabled design theory and methodology: a critical review. *The International Journal of Advanced Manufacturing Technology*, 80(1):327–342, 2015.
- [5] M. Bendsoe and O. Sigmund. *Topology Optimization - Theory, Methods and Applications*. Springer, 2003.
- [6] R. T. Haftka and Z. Gürdal. *Elements of Structural Optimization*. Kluwer Academic Publishers, 1992.
- [7] M. Montemurro, A. Vincenti, and P. Vannucci. The Automatic Dynamic Penalisation method (ADP) for handling constraints with genetic algorithms. *Computer Methods in Applied Mechanics and Engineering*, 256:70 – 87, 2013.
- [8] J. Sokolowski and J.-P. Zolesio. *Introduction to Shape Optimization, Shape Sensitivity Analysis*. Springer-Verlag Berlin Heidelberg, 1992.
- [9] G. Allaire, F. Jouve, and A.-M. Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, 194(1):363 – 393, 2004.
- [10] J. K. Guest, 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*, 61(2):238–254, 2004.

- [11] *OptiStruct 13.0 Reference Guide*, 2014.
- [12] *SIMULIA Tosca Structure Documentation 8.0*, 2013.
- [13] T. Zegard and G. H. Paulino. Bridging topology optimization and additive manufacturing. *Structural and Multidisciplinary Optimization*, 53(1):175–192, 2016.
- [14] I. Gibson, D. Rosen, and B. Stucker. *Additive Manufacturing Technologies, 3D Printing, Rapid Prototyping, and Direct Digital Manufacturing*. Springer-Verlag New York, 2015.
- [15] N. Guo and M. C. Leu. Additive manufacturing: technology, applications and research needs. *Frontiers of Mechanical Engineering*, 8(3):215–243, 2013.
- [16] J.P. Kruth, L. Froyen, J. Van Vaerenbergh, P. Mercelis, M. Rombouts, and B. Lauwers. Selective laser melting of iron-based powder. *Journal of Materials Processing Technology*, 149(1):616 – 622, 2004. 14th International Symposium on Electromachining (ISEM XIV).
- [17] T. H. C. Childs, C. Hauser, and M. Badrossamay. Selective laser sintering (melting) of stainless and tool steel powders: Experiments and modelling. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 219(4):339–357, 2005.
- [18] V. Gunenthiram, P. Peyre, M. Schneider, M. Dal, F. Coste, and R. Fabbro. Analysis of laser melt pool powder bed interaction during the selective laser melting of a stainless steel. *Journal of Laser Applications*, 29(2), 2017.
- [19] E.O. Olakanmi, R.F. Cochrane, and K.W. Dalgarno. A review on selective laser sintering/melting (sls/slm) of aluminium alloy powders: Processing, microstructure, and properties. *Progress in Materials Science*, 74:401 – 477, 2015.
- [20] C. Galy, E. Le Guen, E. Lacoste, and C. Arvieu. Main defects observed in aluminum alloy parts produced by slm: From causes to consequences. *Additive Manufacturing*, 22:165 – 175, 2018.
- [21] F. Abe, E. Costa Santos, Y. Kitamura, K. Osakada, and M. Shiomi. Influence of forming conditions on the titanium model in rapid prototyping with the selective laser melting process. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 217(1):119–126, 2003.
- [22] F. Wang. Mechanical property study on rapid additive layer manufacture Hastelloy<sup>®</sup> X alloy by selective laser melting technology. *The International Journal of Advanced Manufacturing Technology*, 58(5):545–551, 2012.

- [23] K.N. Amato, S.M. Gaytan, L.E. Murr, E. Martinez, P.W. Shindo, J. Hernandez, S. Collins, and F. Medina. Microstructures and mechanical behavior of inconel 718 fabricated by selective laser melting. *Acta Materialia*, 60(5):2229 – 2239, 2012.
- [24] T. G. Spears and S. A. Gold. In-process sensing in selective laser melting (SLM) additive manufacturing. *Integrating Materials and Manufacturing Innovation*, 5(1), 2016.
- [25] P. Mercelis and J. P. Kruth. Residual stresses in selective laser sintering and selective laser melting. *Rapid Prototyping Journal*, 12(5):254–265, 2006.
- [26] H. S. Byun and K. H. Lee. Determination of the optimal build direction for different rapid prototyping processes using multi-criterion decision making. *Robotics and Computer-Integrated Manufacturing*, 22(1):69 – 80, 2006.
- [27] V. Canellidis, J. Giannatsis, and V. Dedoussis. Genetic-algorithm-based multi-objective optimization of the build orientation in stereolithography. *The International Journal of Advanced Manufacturing Technology*, 45(7):714–730, 2009.
- [28] M. Cloots, A. B. Spierings, and K. Wegener. Assessing New Support Minimizing Strategies for the Additive Manufacturing Technology SLM. Solid Freeform Fabrication Proceedings, Annual international solid freeform fabrication symposium, 2013.
- [29] C. Yan, L. Hao, A. Hussein, and D. Raymont. Evaluations of cellular lattice structures manufactured using selective laser melting. *International Journal of Machine Tools and Manufacture*, 62:32 – 38, 2012.
- [30] J.-P. Järvinen, V. Matilainen, X. Li, H. Piili, A. Salminen, I. Mäkelä, and O. Nyrhilä. Characterization of effect of support structures in laser additive manufacturing of stainless steel. *Physics Procedia*, 56:72 – 81, 2014. 8th International Conference on Laser Assisted Net Shape Engineering LANE 2014.
- [31] P. Das, R. Chandran, R. Samant, and S. Anand. Optimum part build orientation in additive manufacturing for minimizing part errors and support structures. *Procedia Manufacturing*, 1:343 – 354, 2015. 43rd North American Manufacturing Research Conference, NAMRC 43, 8-12 June 2015, UNC Charlotte, North Carolina, United States.
- [32] Y.M. Xie and G.P. Steven. A simple evolutionary procedure for structural optimization. *Computers & Structures*, 49(5):885 – 896, 1993.

- [33] B. Bourdin and A. Chambolle. The phase-field method in optimal design. In M. P. Bendsøe and N. Olhoff O. Sigmund, editors, *IUTAM Symposium on Topological Design Optimization of Structures, Machines and Materials*, pages 207–215. Springer Netherlands, 2006.
- [34] M. P. Bendsoe and N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering*, 71(2):197–224, 1988.
- [35] T. A. Poulsen. A new scheme for imposing a minimum length scale in topology optimization. *International Journal for Numerical Methods in Engineering*, 57(6):741–760, 2003.
- [36] O. Sigmund. On the usefulness of non-gradient approaches in topology optimization. *Structural and Multidisciplinary Optimization*, 43(5):589–596, 2011.
- [37] G. I. N. Rozvany. *Structural Design via Optimality Criteria*. Springer Netherlands, 1989.
- [38] G. I. N. Rozvany. Difficulties in truss topology optimization with stress, local buckling and system stability constraints. *Structural optimization*, 11(3):213–217, 1996.
- [39] K. Svanberg. The method of moving asymptotes, a new method for structural optimization. *International Journal for Numerical Methods in Engineering*, 24(2):359–373, 1987.
- [40] K. Svanberg. A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM Journal on Optimization*, 12(2):555–573, 2002.
- [41] O. Sigmund. A 99 line topology optimization code written in Matlab. *Structural and Multidisciplinary Optimization*, 21(2):120–127, 2001.
- [42] F. Wang, B. S. Lazarov, and O. Sigmund. On projection methods, convergence and robust formulations in topology optimization. *Structural and Multidisciplinary Optimization*, 43(6):767–784, 2011.
- [43] J. K. Guest. Imposing maximum length scale in topology optimization. *Structural and Multidisciplinary Optimization*, 37:463–473, 2009.
- [44] G. Allaire and F. Jouve. A level-set method for vibration and multiple loads structural optimization. *Computer Methods in Applied Mechanics and Engineering*, 194(30):3269 – 3290, 2005. Structural and Design Optimization.

- [45] G. Allaire, F. Jouve, and G. Michailidis. Thickness control in structural optimization via a level set method. *Structural and Multidisciplinary Optimization*, 53(6):1349–1382, 2016.
- [46] N. P. van Dijk, K. Maute, M. Langelaar, and F. van Keulen. Level-set methods for structural topology optimization: a review. *Structural and Multidisciplinary Optimization*, 48(3):437–472, 2013.
- [47] S. Y. Wang and M. Y. Wang. Radial basis functions and level set method for structural topology optimization. *International Journal for Numerical Method in Engineering*, 65(12):2060–2090, 2006.
- [48] Z. Luo, L. Tong, M. Y. Wang, and S. Wang. Shape and topology optimization of compliant mechanisms using a parameterization level set method. *Journal of Computational Physics*, 227:680–705, 2007.
- [49] Z. Luo, L. Tong, and Z. Kang. A level set method for structural shape and topology optimization using radial basis functions. *Computers and Structures*, 87:425–434, 2009.
- [50] T. Yamada, K. Izui, S. Nishiwaki, and A. Takezawa. A topology optimization method based on the level set method incorporating a fictitious interface energy. *Computer Methods in Applied Mechanics and Engineering*, 199(45):2876 – 2891, 2010.
- [51] A.N. Tikhonov, A. Goncharsky, V.V. Stepanov, and A. G. Yagola. *Numerical Methods for the Solution of Ill-Posed Problems*. Springer Netherlands, 1995.
- [52] V. J. Challis. A discrete level-set topology optimization code written in Matlab. *Structural and Multidisciplinary Optimization*, 41(3):453–464, 2010.
- [53] O. Sigmund. Manufacturing tolerant topology optimization. *Acta Mechanica Sinica*, 25(2):227–239, Apr 2009.
- [54] D. Brackett, I. Ashcroft, and R. Hague. Topology optimization for additive manufacturing. In *22nd Annual international solid freeform fabrication symposium*, 2011.
- [55] J. Liu and Y. Ma. A survey of manufacturing oriented topology optimization methods. *Advances in Engineering Software*, 100:161 – 175, 2016.
- [56] M. Leary, L. Merli, F. Torti, M. Mazur, and M. Brandt. Optimal topology for additive manufacture: A method for enabling additive manufacture of support-free optimal structures. *Materials & Design*, 63:678 – 690, 2014.



- [57] A.T. Gaynor and J.K. Guest. Topology optimization for additive manufacturing: considering maximum overhang constraint. In *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, pages 16–20, 2014.
- [58] A. T. Gaynor and J. K. Guest. Topology optimization considering overhang constraints: Eliminating sacrificial support material in additive manufacturing through design. *Structural and Multidisciplinary Optimization*, 54(5):1157–1172, 2016.
- [59] M. Langelaar. Topology optimization of 3d self-supporting structures for additive manufacturing. *Additive Manufacturing*, 12:60 – 70, 2016.
- [60] M. Hoffarth, N. Gerzen, and C. Pedersen. ALM Overhang Constraint in Topology Optimization for Industrial Applications. In *12th World Congress on Structural and Multidisciplinary Optimisation*, 2017.
- [61] A. Clausen. *Topology Optimization for Additive Manufacturing*. PhD thesis, DTU Mechanical Engineering, 2016.
- [62] A. M. Mirzendehtel and K. Suresh. Support structure constrained topology optimization for additive manufacturing. *Computer-Aided Design*, 81:1 – 13, 2016.
- [63] X. Qian. Undercut and overhang angle control in topology optimization: A density gradient based integral approach. *International Journal for Numerical Methods in Engineering*, 111(3):247–272, 2016.
- [64] T. Primo, M. Calabrese, A. Del Prete, and A. Anglani. Additive manufacturing integration with topology optimization methodology for innovative product design. *The International Journal of Advanced Manufacturing Technology*, 93(1):467–479, 2017.
- [65] P. Bézier. *Courbes et surfaces*. Springer-Verlag, Paris: Hermès, 1986.
- [66] C. de Boor. *A practical guide to splines*. Springer-Verlag, New York, 1978.
- [67] G. Farin. *Curves and Surfaces for CAGD: A Practical Guide*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 5th edition, 2002.
- [68] L. Piegl and W. Tiller. *The NURBS book*. Springer-Verlag, Berlin, Heidelberg, New York, 1997.
- [69] The Nature of Mathematical Programming, Mathematical Programming Glossary, 2014, <https://glossary.informs.org>.
- [70] J. Nocedal and S. Wright. *Numerical Optimization*. Springer-Verlag New York, 2006.

- [71] F. W. Glover and G. A. Kochenberger. *Handbook of Metaheuristics*. Springer US, 2003.
- [72] M. Montemurro. *Optimal design of advanced engineering modular systems through a new genetic approach*. PhD thesis, Université Pierre et Marie Curie Paris VI, 2012.
- [73] O. Mangasarian. *Nonlinear Programming*. Society for Industrial and Applied Mathematics, 1994.
- [74] R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, New York, NY, USA, second edition, 1987.
- [75] E. Rizzo. *Optimization Methods Applied to the preliminary design of innovative non conventional aircraft configurations*. ETS, 2009.
- [76] *Optimization Toolbox User's Guide*. The MathWorks Inc, 3 Apple Hill Drive, Natick, September 2011.
- [77] A. Forsgren, P. E. Gill, and M. H. Wright. Interior methods for nonlinear optimization. *SIAM review*, 44(4):525–597, 2002.
- [78] N. Gould, D. Orban, and P. Toint. Numerical methods for large-scale nonlinear optimization. *Acta Numerica*, 14:299–361, 2005.
- [79] Z. Michalewicz. *Genetic Algorithm + Data Structures = Evolution Programs*. Springer-Verlag Berlin Heidelberg, 1994.
- [80] L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial intelligence through simulated evolution*. Oxford, England: John Wiley & Sons, 1966.
- [81] F. Glover. Heuristic for integer programming using surrogate constraints. *Decision Sciences*, 8(1):156–166, 1977.
- [82] J.H. Holland. *Adaptation in natural and artificial systems*. Ann-Arbor, 1975.
- [83] D.E. Goldberg. *Genetic algorithm*. New York, 1994.
- [84] M. Dorigo, V. Maniezzo, and A. Colorni. Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 26(1):29–41, 1996.
- [85] J. Kennedy and R. Eberhart. Particle Swarm Optimization. pages 1942–1945, Perth, Australia, 1995. Proceedings of the IEEE International Conference on Neural Networks.

- [86] A. Vincenti. *Conception et optimisation de composites par méthode polaire et algorithmes génétiques*. PhD thesis, Université de Bourgogne, France, 2002.
- [87] A. Vincenti, M. R. Ahmadian, and P. Vannucci. BIANCA: a genetic algorithm to solve hard combinatorial optimisation problems in engineering. *Journal of Global Optimization*, 48(3):399–421, 2010.
- [88] C. A. Coello Coello. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer Methods in Applied Mechanics and Engineering*, 191(11):1245 – 1287, 2002.
- [89] A. Catapano and M. Montemurro. A multi-scale approach for the optimum design of sandwich plates with honeycomb core - Part I: homogenisation of core properties. *Composite Structures*, 118:664 – 676, 2014.
- [90] A. Catapano and M. Montemurro. A multi-scale approach for the optimum design of sandwich plates with honeycomb core - Part II: the optimization strategy. *Composite Structures*, 118:677 – 690, 2014.
- [91] M. Montemurro, A. Vincenti, and P. Vannucci. Design of the elastic properties of laminates with a minimum number of plies. *Mechanics of Composite Materials*, 48(4), 2012.
- [92] M. Montemurro, A. Catapano, and D. Doroszewski. A multi-scale approach for the simultaneous shape and material optimisation of sandwich panels with cellular core. *Composites Part B: Engineering*, 91:458 – 472, 2016.
- [93] M. Montemurro and A. Catapano. *Variational Analysis and Aerospace Engineering: Mathematical Challenges for the Aerospace of the Future*, chapter A New Paradigm for the Optimum Design of Variable Angle Tow Laminates, pages 375–400. Springer, 2016.
- [94] Marco Montemurro and Anita Catapano. On the effective integration of manufacturability constraints within the multi-scale methodology for designing variable angle-tow laminates. *Composite Structures*, 161:145 – 159, 2017.
- [95] W.D. Ueng, J.Y. Lai, and Y.C. Tsai. Unconstrained and constrained curve fitting for reverse engineering. *International Journal of Advanced Manufacturing Technology*, 33:1189–1203, 2007.
- [96] A. Galvez and A. Iglesias. Particle swarm optimization for non-uniform rational B-spline surface reconstruction from clouds of 3D data points. *Information Sciences*, 192:174–192, 2012.

- [97] G. Costa, M. Montemurro, and J. Pailhès. A General Hybrid Optimization Strategy for Curve Fitting in the Non-Uniform Rational Basis Spline Framework. *Journal of Optimization Theory and Applications*, 176(1):225–251, 2018.
- [98] C. J. Turner and R. H. Crawford. N-Dimensional Nonuniform Rational B-Splines for Metamodeling. *Journal of Computing and Information Science in Engineering*, 9(3), 2009.
- [99] Y. Audoux, M. Montemurro, and J. Pailhès. A surrogate model based on Non-Uniform Rational B-Splines hypersurfaces. In *28th CIRP Design Conference, Nantes, France*, 2018.
- [100] B. C. Kim, P. M. Weaver, and K. Potter. Computer aided modelling of variable angle tow composites manufactured by continuous tow shearing. *Composite Structures*, 129:256 – 267, 2015.
- [101] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs. Isogeometric analysis: Cad, finite elements, nurbs, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194(39):4135 – 4195, 2005.
- [102] S. Cai, W. Zhang, J. Zhu, and T. Gao. Stress constrained shape and topology optimization with fixed mesh: A B-spline finite cell method combined with level set function. *Computer Methods in Applied Mechanics and Engineering*, 278:361 – 387, 2014.
- [103] Y. Wang and D. J. Benson. Geometrically constrained isogeometric parametrized level-set based topology optimization via trimmed elements. *Frontiers of Mechanical Engineering*, 11(4):328–343, 2016.
- [104] X. Qian. Topology optimization in B-Spline space. *Computer Methods in Applied Mechanics and Engineering*, 265:15 – 35, 2013.
- [105] M. Wang and X. Qian. Efficient Filtering in Topology Optimization via B-Splines. *ASME, Journal of Mechanical Design*, 137(3):225–251, 2015.
- [106] G. Costa, M. Montemurro, and J. Pailhès. A NURBS-based Topology Optimisation method including additive manufacturing constraints. In J.F. Silva Gomes and Shaker A. Meguid editors, editors, *Proceedings of the 7th International Conference on Mechanics and Materials in Design*, Albufeira, Portugal, 2017. INEGI/FEUP.
- [107] G. Costa, M. Montemurro, and J. Pailhès. A 2D topology optimisation algorithm in NURBS framework with geometric constraints. *International Journal of Mechanics and Materials in Design*, 2017.

- [108] G. Costa, M. Montemurro, and J. Pailhès. A Geometry-based Method for 3D Topology Optimization. In *First International Conference on Mechanics of Advanced Materials and Structures - Proceedings*, Turin, Italy, 2018.
- [109] G. Costa, M. Montemurro, and J. Pailhès. NURBS Hyper-surfaces for 3D Topology Optimisation Problems. *submitted to Computer Methods in Applied Mechanics and Engineering*.
- [110] X. Gu and S.-T. Yau. Global Conformal Surface Parameterization. In Leif Kobbelt, Peter Schroeder, and Hugues Hoppe, editors, *Eurographics Symposium on Geometry Processing*. The Eurographics Association, 2003.
- [111] M. S. Floater. Parametrization and smooth approximation of surface triangulations. *Computer Aided Geometric Design*, 14(3):231 – 250, 1997.
- [112] M. Zhou, Y. K. Shyy, and H. L. Thomas. Checkerboard and minimum member size control in topology optimization. *Structural and Multidisciplinary Optimization*, 21(2):152–158, 2001.
- [113] Xu Guo, Weisheng Zhang, and Wenliang Zhong. Explicit feature control in structural topology optimization via level set method. *Computer Methods in Applied Mechanics and Engineering*, 272:354 – 378, 2014.
- [114] W. Zhang, W. Zhong, and X. Guo. An explicit length scale control approach in simp-based topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 282:71 – 86, 2014.
- [115] G. Michailidis. *Manufacturing Constraints and Multi-Phase Shape and Topology Optimization via a Level-Set Method*. PhD thesis, Ecole Polytechnique X, 2014.
- [116] M. Zhou, B. S. Lazarov, F. Wang, and O. Sigmund. Minimum length scale in topology optimization by geometric constraints. *Computer Methods in Applied Mechanics and Engineering*, 293:266 – 282, 2015.
- [117] Y. Gu and X. Qian. B-Spline Based Robust Topology Optimization. In *Volume 2B: 41st Design Automation Conference, International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2015.
- [118] B. S. Lazarov, F. Wang, and O. Sigmund. Length scale and manufacturability in density-based topology optimization. *Archive of Applied Mechanics*, 86(1):189–218, 2016.
- [119] B. S. Lazarov and F. Wang. Maximum length scale in density based topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 318:826 – 844, 2017.

- [120] M. Jansen, G. Lombaert, M. Schevenels, and O. Sigmund. Topology optimization of fail-safe structures using a simplified local damage model. *Structural and Multidisciplinary Optimization*, 49(4):657–666, 2014.
- [121] T. S. Kim, J. E. Kim, J. H. Jeong, and Y. Y. Kim. Filtering technique to control member size in topology design optimization. *KSME International Journal*, 18(2):253–261, 2004.
- [122] G. Costa, M. Montemurro, and J. Pailhès. Minimum and maximum length scales in NURBS-based Topology Optimisation algorithm. *to be submitted*.
- [123] R. Goldman. Curvature formulas for implicit curves and surfaces. *Computer Aided Geometric Design*, 22(7):632 – 658, 2005.
- [124] G. Kreisselmeier and R. Steinhauser. Systematic control design by optimizing a vector performance index. *IFAC Proceedings Volumes*, 12(7):113 – 117, 1979.
- [125] N.L. Pedersen and A.K. Nielsen. Optimization of practical trusses with constraints on eigenfrequencies, displacements, stresses, and buckling. *Structural and Multidisciplinary Optimization*, 25(5):436–445, 2003.
- [126] M. Zhou. Difficulties in truss topology optimization with stress and local buckling constraints. *Structural optimization*, 11(2):134–136, 1996.
- [127] G. I. N. Rozvany. Difficulties in truss topology optimization with stress, local buckling and system stability constraints. *Structural optimization*, 11(3):213–217, 1996.
- [128] G. D. Cheng and X. Guo.  $\epsilon$ -relaxed approach in structural topology optimization. *Structural optimization*, 13(4):258–266, 1997.
- [129] X. Guo, G. Cheng, and K. Yamazaki. A new approach for the solution of singular optima in truss topology optimization with stress and local buckling constraints. *Structural and Multidisciplinary Optimization*, 22(5):364–373, 2001.
- [130] M. M. Neves, H. Rodrigues, and J. M. Guedes. Generalized topology design of structures with a buckling load criterion. *Structural optimization*, 10(2):71–78, 1995.
- [131] M. M. Neves, O. Sigmund, and M. P. Bendsøe. Topology optimization of periodic microstructures with a penalization of highly localized buckling modes. *International Journal for Numerical Methods in Engineering*, 54(6):809–834, 2002.
- [132] M. Zhou. Topology optimization for shell structures with linear buckling responses. In *WCCM VI in conjunction with APCOM'04, Beijing, China*. Tsinghua University Press & Springer-Verlag, 2004.

- [133] Q. Luo and L. Tong. Structural topology optimization for maximum linear buckling loads by using a moving iso-surface threshold method. *Structural and Multidisciplinary Optimization*, 52(1):71–90, 2015.
- [134] E. Lindgaard and J. Dahl. On compliance and buckling objective functions in topology optimization of snap-through problems. *Structural and Multidisciplinary Optimization*, 47(3):409–421, 2013.
- [135] P. A. Browne, C. Budd, N. I. M. Gould, H. A. Kim, and J. A. Scott. A fast method for binary programming using first-order derivatives, with application to topology optimization with buckling constraints. *International Journal for Numerical Methods in Engineering*, 92(12):1026–1043, 2012.
- [136] B. Bochenek and K. Tajs-Zielińska. Minimal compliance topologies for maximal buckling load of columns. *Structural and Multidisciplinary Optimization*, 51(5):1149–1157, 2015.
- [137] X. Gao and H. Ma. Topology optimization of continuum structures under buckling constraints. *Computers & Structures*, 157:142 – 152, 2015.
- [138] X. Gao, L. Li, and H. Ma. An adaptive continuation method for topology optimization of continuum structures considering buckling constraints. *International Journal of Applied Mechanics*, 9(7), 2017.
- [139] G. I. N. Rozvany. A critical review of established methods of structural topology optimization. *Structural and Multidisciplinary Optimization*, 37(3):217–237, 2009.
- [140] Z.-D. Ma, H.-C. Cheng, and N. Kikuchi. Structural design for obtaining desired eigenfrequencies by using the topology and shape optimization method. *Computing Systems in Engineering*, 5(1):77 – 89, 1994.
- [141] I. Kosaka and C. C. Swan. A symmetry reduction method for continuum structural topology optimization. *Computers & Structures*, 70(1):47 – 61, 1999.
- [142] L. Tenek H. and I. Hagiwara. Eigenfrequency maximization of plates by optimization of topology using homogenization and mathematical programming. *JSME international journal. Ser. C, Dynamics, control, robotics, design and manufacturing*, 37(4):667–677, 1994.
- [143] N.L. Pedersen. Maximization of eigenvalues using topology optimization. *Structural and Multidisciplinary Optimization*, 20(1):2–11, 2000.
- [144] D. Tcherniak. Topology optimization of resonating structures using simp method. *International Journal for Numerical Methods in Engineering*, 54(11):1605–1622, 2002.

- [145] J. S. Jensen and N. L. Pedersen. On maximal eigenfrequency separation in two-material structures: the 1d and 2d scalar cases. *Journal of Sound and Vibration*, 289(4):967 – 986, 2006.
- [146] J. Du and N. Olhoff. Topological design of freely vibrating continuum structures for maximum values of simple and multiple eigenfrequencies and frequency gaps. *Structural and Multidisciplinary Optimization*, 34(2):91–110, 2007.
- [147] B. Niu, J. Yan, and G. Cheng. Optimum structure with homogeneous optimum cellular material for maximum fundamental frequency. *Structural and Multidisciplinary Optimization*, 39(2):115, 2008.
- [148] T. D. Tsai and C. C. Cheng. Structural design for desired eigenfrequencies and mode shapes using topology optimization. *Structural and Multidisciplinary Optimization*, 47(5):673–686, 2013.
- [149] W. Su and S. Liu. Topology design for maximization of fundamental frequency of couple-stress continuum. *Structural and Multidisciplinary Optimization*, 53(3):395–408, 2016.
- [150] S. J. Rahi and K. Sharp. Mapping complicated surfaces onto a sphere. *International Journal of Computational Geometry & Applications*, 17(04):305–329, 2007.
- [151] X. Gu. *Parametrization for Surfaces with Arbitrary Topologies*. PhD thesis, Harvard University, 2002.
- [152] W. Wang, H. Pottmann, and Y. Liu. Fitting b-spline curves to point clouds by squared distance minimization. Technical Report 11, The University of Hong Kong, Department of Computer Science, 2004.
- [153] W.-D. Ueng, J.-Y. Lai, and Y.-C. Tsai. Unconstrained and constrained curve fitting for reverse engineering. *The International Journal of Advanced Manufacturing Technology*, 33(11):1189–1203, 2007.
- [154] W. Zheng, P. Bo, Y. Liu, and W. Wang. Fast b-spline curve fitting by l-bfgs. *Computer Aided Geometric Design*, 29(7):448 – 462, 2012.
- [155] R. M. Bolle and B. C. Vemuri. On three-dimensional surface reconstruction methods. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*. IEEE, 1991.
- [156] C. Olsson and Y. Boykov. Curvature-based regularization for surface approximation. In *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012.



- [157] W. Li, S. Xu, G. Zhao, and L. P. Goh. Adaptive knot placement in b-spline curve approximation. *Computer-Aided Design*, 37(8):791 – 797, 2005.
- [158] A. Limaiem, A. Nassef, and H. A. El-Maraghy. Data fitting using dual kriging and genetic algorithms. *CIRP Annals*, 45(1):129 – 134, 1996.
- [159] H. Kang, F. Chen, Y. Li, J. Deng, and Z. Yang. Knot calculation for spline fitting via sparse optimization. *Computer-Aided Design*, 58:179 – 188, 2015.
- [160] A. Gálvez, A. Iglesias, and J. Puig-Pey. Iterative two-step genetic-algorithm-based method for efficient polynomial b-spline surface reconstruction. *Information Sciences*, 182(1):56 – 76, 2012.
- [161] D. Brujic, I. Ainsworth, and M. Ristic. Fast and accurate nurbs fitting for reverse engineering. *The International Journal of Advanced Manufacturing Technology*, 54(5):691–700, 2011.
- [162] C. H. Garcia-Capulin, F. J. Cuevas, G. Trejo-Caballero, and H. Rostro-Gonzalez. A hierarchical genetic algorithm approach for curve fitting with b-splines. *Genetic Programming and Evolvable Machines*, 16(2):151–166, 2015.
- [163] M. Montemurro, A. Vincenti, and P. Vannucci. A two-level procedure for the global optimum design of composite modular structures—application to the design of an aircraft wing. *Journal of Optimization Theory and Applications*, 155(1):1–23, 2012.
- [164] M. Montemurro, A. Vincenti, and P. Vannucci. A two-level procedure for the global optimum design of composite modular structures—application to the design of an aircraft wing. *Journal of Optimization Theory and Applications*, 155(1):24–53, 2012.
- [165] Kuan-Yuan Lin, Chung-Yi Huang, Jiing-Yih Lai, Yao-Chen Tsai, and Wen-Der Ueng. Automatic reconstruction of b-spline surfaces with constrained boundaries. *Computers & Industrial Engineering*, 62(1):226 – 244, 2012.

# List of Publications issued from this thesis

## Published Articles

1. G. Costa, M. Montemurro, J. Pailhès, A General Hybrid Optimization Strategy for Curve Fitting in the Non-uniform Rational Basis Spline Framework, *Journal of Optimization Theory and Applications*, v. 176(1), pp. 225-251, 2018  
<https://doi.org/10.1007/s10957-017-1192-2>
2. G. Costa, M. Montemurro, J. Pailhès, A 2D topology optimisation algorithm in NURBS framework with geometric constraints, *International Journal of Mechanics and Materials in Design*, 2017  
<https://doi.org/10.1007/s10999-017-9396-z>

## Submitted Articles

1. G. Costa, M. Montemurro, J. Pailhès, NURBS Hyper-Surfaces for 3D Topology Optimisation Problems, *International Journal of Solids and Structures*, under review

## Conferences

1. G. Costa, M. Montemurro, J. Pailhès, A NURBS-based Topology Optimisation Method including Additive-Manufacturing constraints, *7th International Conference of Mechanics and Materials in Design*, Albufeira (Portugal), June 2017
2. G. Costa, M. Montemurro, J. Pailhès, Eigenvalue buckling analysis via a NURBS-framed algorithm, *6th European Conference on Computational Mechanics*, Glasgow (UK), June 2018
3. G. Costa, M. Montemurro, J. Pailhès, A geometry-based method for 3D topology optimisation, *1st International Conference on Mechanics of Advanced Materials and Structures*, Torino (Italy), June 2018





## METHODOLOGIES DE CONCEPTION ET D'OPTIMISATION DE STRUCTURES OBTENUES PAR ADDITIVE LAYER MANUFACTURING

**RESUME :** Les développements récents de l'Additive Layer Manufacturing (ALM) et de l'Optimisation Topologique (OT) ont suscité l'intérêt des chercheurs et des industries. Potentiellement, il est possible de produire des pièces de plus en plus performantes répondant aux exigences du cahier de charges. Toutefois, la liaison entre OT et ALM n'est qu'un aspect d'une question plus générale concernant l'intégration de l'OT dans les procédures de conception. Actuellement, les résultats des logiciels commerciaux d'OT sont fournis sous forme discrète et non en termes d'entités géométriques. La conséquence néfaste est que les contraintes géométriques, même les plus simples, ne sont pas respectées sur une géométrie continue et CAO-compatible. Un post-traitement chronophage de reconstruction CAO est donc nécessaire. Dans cette phase, le concepteur prend des décisions arbitraires. Afin de remédier à ces inconvénients, un algorithme d'OT innovant est présenté dans ce travail: les entités B-Spline / NURBS sont systématiquement utilisées pour représenter la topologie. Ce choix simplifie considérablement la phase de reconstruction CAO en 2D et présente un grand potentiel pour les cas 3D. De plus, des contraintes géométriques liées à la tenue en service et à la fabricabilité de la pièce peuvent être efficacement intégrées dans la formulation du problème. La robustesse de cette méthode a été testée sur plusieurs cas d'étude, impliquant la rigidité, les charges de flambement et les fréquences propres.

**Mots clés :** Optimisation Topologique, NURBS, CAO, Fabrication Additive

## DESIGN AND OPTIMISATION METHODS FOR STRUCTURES PRODUCED BY MEANS OF ADDITIVE LAYER MANUFACTURING PROCESSES

**ABSTRACT :** Recent developments of Additive Layer Manufacturing (ALM) and Topology Optimisation (TO) have raised interest of scientific and industrial communities. Potentially, increasingly efficient parts can be produced, meeting the requirements demanded by the technical specification as well. However, bridging TO and ALM is just one aspect of a more general issue concerning the integration of TO in numerical design procedures. Nowadays, results of TO commercial software are provided in a discrete form and not in terms of geometric entities. The detrimental consequence is that even the simplest geometrical constraints are not met on a continuous, CAD-compatible, geometry. Therefore, a time-consuming CAD reconstruction phase is needed, wherein the designer takes arbitrary decisions. In order to overcome these shortcomings, an innovative TO algorithm is presented in this work: B-Spline/NURBS entities are systematically used to represent the topology. This choice considerably simplifies the CAD reconstruction phase in 2D and it exhibits great potential for 3D cases. Moreover, geometric constraints can be efficiently integrated into the problem formulation. The robustness of this method has been tested on several benchmarks, involving compliance, buckling loads and eigen-frequencies.

**Keywords :** Topology Optimisation, NURBS, CAD, Additive Layer Manufacturing