



**HAL**  
open science

# Calcul de gradient sur des paramètres CAO pour l'optimisation de forme

Timothée Leblond

► **To cite this version:**

Timothée Leblond. Calcul de gradient sur des paramètres CAO pour l'optimisation de forme. Autre. Université Paris Saclay (COmUE), 2017. Français. NNT : 2017SACLC017 . tel-01503189

**HAL Id: tel-01503189**

**<https://theses.hal.science/tel-01503189>**

Submitted on 6 Apr 2017

**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.

NNT : 2017SACL017

THÈSE DE DOCTORAT  
DE L'UNIVERSITÉ PARIS-SACLAY  
PRÉPARÉE À  
CENTRALESUPÉLEC

École doctorale n°579

SMEMAG : Sciences mécaniques et énergétiques, matériaux et géosciences

Spécialité de doctorat : génie mécanique

par

**M. TIMOTHÉE LEBLOND**

Calcul de gradient sur des paramètres CAO pour l'optimisation de forme

Thèse présentée et soutenue à Palaiseau, le 22 mars 2017.

**Composition du Jury :**

Pr. Pierre-Alain BOUCARD	Professeur des Universités, ENS Paris-Saclay	Président du jury
Pr. Gaël CHEVALIER	Professeur des Universités, FEMTO-ST	Directeur de thèse
Pr. Jean-François DEU	Professeur des Universités, CNAM Paris	Rapporteur
Dr. Thomas ELGUEDJ	Maître de conférences (HDR), INSA Lyon	Examinateur
Dr. Pierre FROMENT	Ingénieur de recherche, IRT SystemX	Co-encadrant
Dr. Paul DE NAZELLE	Ingénieur de recherche, IRT SystemX	Co-encadrant
Pr. Jean-François REMACLE	Professeur des Universités, UCL	Rapporteur
Pr. Christophe TOURNIER	Professeur des Universités, ENS Paris-Saclay	Examinateur



Cette thèse a été proposée et financée dans le cadre du projet « Réduction de modèles et Optimisation Multiphysique » (ROM) de l'Institut de Recherche Technologique SystemX.

Elle a été encadrée par Paul DE NAZELLE et Pierre FROMENT de l'IRT SystemX en collaboration avec Philippe SERRÉ et Réda SELLA KH du laboratoire QUARTZ de Supméca et sous la direction de Gaël CHEVALLIER du FEMTO-ST.





à PAPUM,



# Remerciements

---

Une thèse c'est comme un film dont le script final est écrit après en avoir vécu les images. Et derrière tout film, de nombreuses personnes se cachent pour que tout soit réglé comme une horloge et que l'intrigue nous tienne en haleine jusqu'à la fin. Timothée, l'acteur principal, tient à remercier l'ensemble de la distribution pour leur présence et leur soutien.

Lorsqu'on lui demande si être doctorant est une bonne situation, il répond : « *Vous savez, moi je ne crois pas qu'il y ait de bonne ou de mauvaise situation. Moi, si je devais résumer ma vie aujourd'hui avec vous, je dirais que c'est d'abord des rencontres.* Tout a commencé avec Pierre et Paul, deux passionnés à qui les rôles de scénariste, de directeur de casting et de réalisateur convenaient parfaitement. Leurs compétences, leurs connaissances, leur exigence et leur maîtrise de l'intrigue n'avaient d'égal que la formation, l'amitié et la confiance qu'ils m'ont accordées. Marc a aussi beaucoup contribué à l'histoire que j'ai eu la chance de jouer. »

Pour mener à bien leur projet, ils ont trouvé en la personne de Gaël un producteur exécutif idéal. Avec son équipe rapprochée, ce dernier a accompagné l'acteur dans les moments de doute et lui a fait découvrir une région parfaite pour envisager un ou deux plans-séquences.

L'IRT SystemX, la boîte de production, a constitué le plateau de tournage idéal pour voir évoluer l'acteur. Son équipe de communication ne s'est pas uniquement chargée de la promotion du film, elle a apporté à l'acteur un support sans faille. « *Virginie est toujours à l'écoute même avec un emploi du temps familial et professionnel chargé, j'en suis extrêmement reconnaissant. Mon rôle n'aurait pas eu la même dimension s'il n'y avait pas eu Aurélie, elle a mis du relief et le petit grain de folie à l'histoire. Julie et Marine ont apporté leur touche personnelle à une équipe déjà au top!* » précise l'interprète.

Parmi les rôles incontournables, on retrouve les talentueux Frédéric, Jean-Alexis, François, Idil, Aurélien, Félix, Laura, Karim, Sophie et Nathalie. L'acteur semble s'être éclaté sur les planches avec eux. Les duplex en Australie et en Crète ont permis un jeu d'acteur idéal avec Laurent et Sarah, l'acteur se souviendra longtemps de ces épopées! Les duplex aux USA, en République tchèque, sur la côte méditerranéenne ou sur les sommets enneigés ont également beaucoup compté pour le comédien.

Celui-ci a beaucoup joué à l'extérieur, il tient aussi à remercier les équipes de tournage :

- du Technocentre : Frédéric, Sylvain, Pascal, Cricri.
- de Supméca : Marco, Adrien, Adrien, Jean-Luc, Nicolas, Réda, Philippe, Antoine, Thomas.
- de la province bisontine : Pauline, Thomas, Kévin.

De nombreux acteurs ont joué un second rôle mais ils ont été tout aussi importants dans l'histoire. Parmi eux, l'interprète souhaite nommer : Lex, Bahaa, Hervé, Vincent, Kim, Mathieu, Julie, Michel, Makhlof, Benjamin, Romain, Hadrien, Gauthier, Paul, Raïssa, Marvin, Maxime, Cédric, Jérémy, Thomas, Cédric, Patrice, Étienne, Gaya... L'acteur a aussi tenu à nommer Yann,

Ludo, Eddy, Sabine, JC, Claudy, Bernard, Jérôme, Didier... – nous cherchons encore à les identifier, quelqu'un a une idée ? – La distribution sait qu'elle peut s'appuyer sur une équipe de son et lumière constituée de JB, Nico et Sam et dans laquelle l'acteur adore prendre sa place quand l'occasion se présente. Pour la rédaction du script final, l'équipe s'est vu renforcée par les contributions de Mouadh et d'Émeline. Quel beau résultat !

L'acteur est honoré que ce résultat ait été évalué par les critiques renommés : Pierre-Alain Boucard, Jean-François Deü, Thomas Elguedj, Jean-François Remacle et Christophe Tournier. Ils ont apprécié le script et ont récompensé la prestation de l'acteur.

L'interprète ne serait pas sous les projecteurs aujourd'hui s'il n'était pas entouré de sa famille qui le soutient de près ou de plus loin. Il leur doit beaucoup.

Finalement, Timothée se confie : « Je n'aurais pu rêver meilleur premier rôle féminin pour m'accompagner que Mathilde. Quel bonheur de lui donner la réplique au quotidien ! »

# Table des matières

---

<b>Remerciements</b>	<b>vii</b>
<b>Introduction</b>	<b>1</b>
<b>1 Introduction à l'optimisation paramétrique de forme</b>	<b>3</b>
1.1 Deux formulations possibles d'un problème . . . . .	4
1.1.1 Formulation du problème direct . . . . .	4
1.1.2 Formulation du problème inverse . . . . .	5
1.2 Stratégies d'optimisation . . . . .	6
1.2.1 Critères de choix de la méthode . . . . .	6
1.2.2 Méthodes exactes . . . . .	7
1.2.3 Méta-heuristiques . . . . .	8
1.3 Familles de paramétrage . . . . .	10
1.3.1 Optimisation topologique . . . . .	11
1.3.2 Optimisation géométrique . . . . .	12
1.3.3 Optimisation isogéométrique . . . . .	13
1.3.4 Optimisation paramétrique . . . . .	14
1.4 Calcul du gradient d'une fonction objectif par rapport à la forme . . . . .	17
1.4.1 Calcul par différences finies . . . . .	18
1.4.2 Calcul par dérivation directe . . . . .	18
1.4.3 Calcul par la méthode de l'état adjoint . . . . .	19
1.5 Calcul de sensibilité d'une forme par rapport aux paramètres CAO . . . . .	22
1.5.1 Méthode analytique . . . . .	22
1.5.2 Méthodes numériques sur les géométries CAO . . . . .	23
1.5.3 Méthodes numériques sur le maillage . . . . .	24
1.6 Synthèse . . . . .	29
Bibliographie du 1 <sup>er</sup> chapitre . . . . .	30

<b>2</b>	<b>Calcul de la sensibilité d'une forme sur les maillages</b>	<b>33</b>
2.1	Utilisation de la projection harmonique . . . . .	34
2.1.1	Détection de la frontière . . . . .	35
2.1.2	Conditions aux limites de Dirichlet . . . . .	37
2.1.3	Développement de la projection harmonique . . . . .	37
2.1.4	Reconstruction de l'isoconnectivité . . . . .	46
2.2	Adaptation de la projection harmonique appliquée aux géométries multifrontières	47
2.2.1	Déplacement des frontières intérieures . . . . .	49
2.2.2	Utilisation de la double projection harmonique . . . . .	54
2.2.3	Utilisation de la triangulation de Delaunay et des coefficients barycentriques	56
2.2.4	Identification des nœuds particuliers . . . . .	60
2.3	Amélioration de l'interpolation . . . . .	61
2.3.1	Interpolation par connectivité . . . . .	61
2.3.2	Interpolation aux frontières . . . . .	62
2.4	Synthèse et discussion des résultats . . . . .	65
	Bibliographie du 2 <sup>e</sup> chapitre . . . . .	67
<b>3</b>	<b>Calcul de la sensibilité d'une forme sur les géométries CAO</b>	<b>69</b>
3.1	Rappel des notions fondamentales des courbes et des surfaces . . . . .	70
3.1.1	Les courbes . . . . .	70
3.1.2	Les surfaces . . . . .	73
3.2	Calcul semi-analytique . . . . .	76
3.2.1	Utilisation des équations paramétriques et des données STEP . . . . .	76
3.2.2	Calcul de la sensibilité d'un point de coordonnées $(u,v)$ . . . . .	78
3.2.3	Calcul des sensibilités sur un maillage de calcul . . . . .	80
3.3	Comparaison des méthodes de calcul de sensibilité . . . . .	81
3.3.1	Sur un patch . . . . .	81
3.3.2	Dans une optimisation en CFD . . . . .	86
3.3.3	Bilan sur la comparaison des deux méthodes . . . . .	92
3.4	Synthèse . . . . .	92
	Bibliographie du 3 <sup>e</sup> chapitre . . . . .	94

<b>4 Applications des calculs de sensibilité dans une boucle d'optimisation</b>	<b>95</b>
4.1 Présentation de la boucle d'optimisation	96
4.1.1 Création du modèle CAO	96
4.1.2 Génération des géométries CAO et maillage	97
4.1.3 Calculs de gradients	98
4.1.4 Algorithme du gradient	98
4.1.5 Gestion d'une optimisation multiobjectif	106
4.2 Spécificités liées à la CFD	107
4.3 1 <sup>er</sup> cas-test : Coude à 90 °	108
4.3.1 Mises en données	108
4.3.2 Post-traitement et résultats	112
4.3.3 Comparaison avec une méthode d'optimisation par plan d'expériences et surface de réponse	116
4.4 2 <sup>e</sup> cas-test : Collecteur de type industriel	119
4.4.1 Mises en données	120
4.4.2 Post-traitement et résultats	128
4.5 Discussion des résultats et synthèse	131
Bibliographie du 4 <sup>e</sup> chapitre	132
 <b>Conclusion et perspectives</b>	 <b>133</b>
 <b>Annexes</b>	 <b>135</b>
A Détection des frontières	135
B Fonction de paramétrisation $\varphi$	137
C Spécificités liées à la CFD	139
C.1 Critères de qualité du maillage	139
C.2 Types de maillage	141
C.3 Remarque sur le nombre d'itérations et la convergence du calcul CFD	142
Bibliographie des annexes	144
 <b>Bibliographie globale</b>	 <b>149</b>
 <b>Liste des acronymes et abréviations</b>	 <b>152</b>



# Table des figures

1.1	Illustration du front de Pareto pour un problème bi-objectif . . . . .	9
1.2	Schéma de mise en œuvre d'une méthode d'optimisation par plan d'expériences .	10
1.3	Optimisation topologique sur un cantilever avec contraintes d'épaisseur minimale	12
1.4	Optimisation géométrique sur un cantilever . . . . .	13
1.5	Visualisation des points de contrôle d'un patch CAO . . . . .	14
1.6	Paramétrage CAO sur un pavillon de véhicule . . . . .	16
1.7	Explication schématique de l'optimisation par gradient avec utilisation des paramètres CAO . . . . .	21
1.8	Génération CAO . . . . .	23
1.9	Génération des géométries suite à la variation de chacun des paramètres . . . . .	24
1.10	Illustration du concept . . . . .	25
1.11	Discrétisation de la forme . . . . .	26
1.12	Illustration de la méthode de projection suivant la normale . . . . .	26
1.13	Problèmes liés à la projection suivant la normale . . . . .	27
1.14	Illustration du dépliage du masque de Toutânkhamon sur un disque . . . . .	28
1.15	Découpage de la géométrie en deux parties homéomorphes à un disque . . . . .	28
2.1	Illustration de la projection harmonique appliquée à une géométrie monofrontière	35
2.2	Projection de la frontière principale sur le bord du disque . . . . .	37
2.3	Triangle unité, fonctions de projection et coordonnées locales . . . . .	39
2.4	Méthode de projection – 1 . . . . .	46
2.5	Méthode de projection – 2 . . . . .	47
2.6	Méthode de projection – 3 . . . . .	47
2.7	Illustration de la projection harmonique appliquée à une géométrie avec deux frontières . . . . .	48
2.8	Illustration du changement de position d'une frontière suite à une variation de paramètre . . . . .	48
2.9	Projection d'une géométrie avec 2 frontières . . . . .	50
2.10	Représentations simplifiées de maillages obtenus après la projection harmonique .	51

## Table des figures

---

2.11	Isolement des frontières intérieures des projections de la figure 2.10 . . . . .	51
2.12	Opérations sur la frontière intérieure du maillage de la forme initiale . . . . .	51
2.13	Opérations sur la frontière intérieure du maillage de la forme perturbée . . . . .	52
2.14	Superposition des frontières suite à leur circularisation et au déplacement de barycentre . . . . .	53
2.15	Représentations simplifiées de maillages obtenus après la projection harmonique et le déplacement des frontières . . . . .	53
2.16	Illustration de la méthode par double projection harmonique . . . . .	55
2.17	Problème de résolution de la deuxième projection harmonique . . . . .	55
2.18	Illustration de la triangulation de Delaunay . . . . .	56
2.19	Illustration du déplacement d'un point du maillage avec la triangulation de Delaunay	57
2.20	Superposition des disques d'une géométrie avec 2 frontières . . . . .	58
2.21	Amélioration de l'interpolation des disques . . . . .	62
2.22	Illustration de l'interpolation sur la frontière . . . . .	62
2.23	Amélioration de l'interpolation des disques . . . . .	63
2.24	Amélioration de l'interpolation des disques . . . . .	64
2.25	Sensibilité calculée avec les 2 méthodes d'interpolation . . . . .	64
2.26	Modèle CAO . . . . .	65
2.27	Sensibilité du paramètre $\partial\vec{x}/\partial(Thead)$ (épaisseur de la tête) . . . . .	65
2.28	Visualisations de la sensibilité . . . . .	66
3.1	Courbe de Bézier de degré 3 . . . . .	71
3.2	Courbes de Bézier de différents degrés . . . . .	71
3.3	Courbes de Bézier rationnelles de degré 3 . . . . .	72
3.4	Différentes courbes B-Spline de degré 3 et avec 7 points de contrôle . . . . .	74
3.5	Courbe NURBS . . . . .	74
3.6	Surface de Bézier . . . . .	75
3.7	Déformation d'un patch . . . . .	76
3.8	Déplacement des points $\tilde{\mathcal{P}}$ d'une forme suite à la variation d'un paramètre . . . . .	79
3.9	Illustration du processus de la méthode semi-analytique avec le résultat sur le maillage de calcul . . . . .	80
3.10	Visualisation de la répartition des points de passage sur le patch . . . . .	81
3.11	Emplacement des paramètres sur le patch vu de face . . . . .	81
3.12	Visualisation de la variation d'un paramètre . . . . .	82
3.13	Visualisation de la sensibilité de la forme pour une variation de 1 mm sur le paramètre 1 . . . . .	82

3.14	Visualisation de la sensibilité de la forme pour une variation de 1 mm sur le paramètre 2 . . . . .	82
3.15	Visualisation de la sensibilité de la forme pour une variation de 1 mm sur le paramètre 5 . . . . .	83
3.16	Justification de l'écart de valeur . . . . .	83
3.17	Courbes sur la norme de la différence des vecteurs de sensibilité . . . . .	84
3.18	Courbe du nombre de mailles en fonction de la taille de maille . . . . .	85
3.19	Courbes sur la norme de la différence des vecteurs de sensibilité . . . . .	85
3.20	Visualisation de la pyramide à l'état initial . . . . .	86
3.21	Plan de coupe de la veine dans laquelle est placée la pyramide . . . . .	87
3.22	Coupe du maillage volumique pour différentes tailles de maillage . . . . .	88
3.23	Courbes pour l'étude de convergence de maillage . . . . .	89
3.24	Optimisation de la force de trainée . . . . .	89
3.25	Visualisation de la forme à l'état initial et aux états finaux . . . . .	90
3.26	Évolution des paramètres . . . . .	91
3.27	Visualisation de la norme de la vitesse à l'état initial et aux états finaux . . . . .	92
3.28	Visualisation de la pression à l'état initial et aux états finaux . . . . .	93
4.1	Boucle d'optimisation proposée . . . . .	96
4.2	Visualisation des sections de la forme coudée . . . . .	109
4.3	Exemple sur un coude du paramétrage à 2 paramètres par section . . . . .	109
4.4	Visualisations du maillage du coude . . . . .	112
4.5	Visualisation des résidus du calcul d'initialisation . . . . .	113
4.6	Évolution de la perte de charge . . . . .	113
4.7	Visualisation des gradients CFD à l'état initial et à l'état final . . . . .	114
4.8	Courbes de valeurs de paramètres et de leur gradient . . . . .	115
4.9	Visualisation de l'évolution de la forme . . . . .	115
4.10	Visualisation des vecteurs vitesse à l'état initial et à l'état final . . . . .	116
4.11	Visualisation des vecteurs vitesse . . . . .	119
4.12	Visualisation des collecteurs étudiés . . . . .	119
4.13	Schéma des paramètres d'une section CAO . . . . .	121
4.14	Visualisation du paramétrage à 5 paramètres . . . . .	121
4.15	Visualisations particulières du collecteur . . . . .	123
4.16	Visualisation des résidus du calcul d'initialisation - branche $R_1$ active . . . . .	126
4.17	Visualisation du maillage du collecteur - branche $R_2$ active . . . . .	126
4.18	Visualisation du maillage du collecteur - branche $R_2$ active . . . . .	128

## Table des figures

---

4.19 Évolutions de la fonction objectif et des pertes de charge des branches du collecteur	129
4.20 Visualisation de l'évolution de la forme . . . . .	130
4.21 Visualisation des champs de vitesse à l'état initial et à l'état final de la branche $R_1$ active . . . . .	130
4.22 Visualisation des champs de vitesse à l'état initial et à l'état final de la branche $R_2$ active . . . . .	131
A.1 Maillage et matrice de connectivité des éléments associée . . . . .	135
A.2 Construction de la matrice $A$ . . . . .	135
A.3 Sélection d'un segment et reconstruction de la frontière en éliminant les segments symétriques . . . . .	136
C.1 Illustration du calcul de l'angle d'asymétrie entre cellules . . . . .	139
C.2 Illustration du calcul de l'angle d'asymétrie aux frontières . . . . .	140
C.3 Illustration du critère de validité des faces . . . . .	140
C.4 Illustration du critère de qualité des cellules . . . . .	140
C.5 Illustration du critère de changement de volume . . . . .	141
C.6 Visualisation de l'indicateur de mauvaise cellule . . . . .	142

# Liste des tableaux

---

1.1	Comparaison entre les différents types de paramétrage . . . . .	17
2.1	Ordre de grandeur des caractéristiques des géométries pour les nombres de frontières intérieures . . . . .	59
3.1	Tableau récapitulatif des types de courbes et surfaces . . . . .	75
3.2	Tableau de répartition du nombre d'itérations . . . . .	87
4.1	Table de géométries pour 4 paramètres . . . . .	97
4.2	Table du paramétrage initial des sections . . . . .	110
4.3	Pourcentage d'amélioration par itération . . . . .	113
4.4	Table de paramétrage initial des sections . . . . .	122
4.5	Configurations du maillage pour le collecteur . . . . .	128
4.6	Pourcentage d'amélioration de la fonction objectif et des pertes de charge . . . .	129



# Liste des Algorithmes

---

2.1	Algorithme de détection des frontières . . . . .	36
2.2	Algorithme de déplacement des nœuds du maillage . . . . .	58
4.1	Algorithme de recherche d'un pas de Wolfe . . . . .	101



# Introduction

---

Dans ces travaux, nous nous intéressons aux problèmes d'optimisation de forme en grandes dimensions pour la conception industrielle. La conception de nouvelles pièces ou de nouveaux organes vise à respecter les contraintes détaillées dans les différents cahiers des charges. Ces contraintes traduisent les exigences que doit satisfaire la pièce en matière de durée de vie, de comportement physique (crash, acoustique, aérodynamique...) ou encore de coûts pour sa fabrication... Ces objectifs de conception sont en général antagonistes, il est donc difficile de déterminer la conception optimale qui satisfait au mieux l'ensemble des contraintes, notamment en matière de forme, d'architecture ou de matériaux. Pour l'optimisation de forme, un important éventail de méthodes est disponible pour aider les concepteurs à se rapprocher de la solution optimale. Celles-ci profitent de l'essor des puissances de calcul intensif (HPC), notamment afin d'éviter le prototypage souvent coûteux et chronophage pendant les phases de conception. Toutefois, le concepteur doit faire un choix parmi ces méthodes en fonction de ses attentes, de la définition de son problème (variables d'optimisation, contraintes, fonctions à optimiser), de sa capacité de calcul, des contraintes de son environnement comme par exemple son intégration dans un processus déjà existant. Le premier chapitre (p. 3) propose un état de l'art des principales méthodes actuelles pour déterminer celle qui répond au mieux à la problématique industrielle posée : faire de l'optimisation de forme en utilisant les paramètres CAO.

En effet, dans l'industrie, la **conception assistée par ordinateur (CAO)** est devenue incontournable car elle fournit des modèles qui contiennent de nombreuses informations utilisées tout au long du cycle de conception, en particulier les paramètres de dimensionnement. Ces paramètres (aussi appelés paramètres CAO dans la suite du manuscrit) sont, par exemple, des longueurs ou des angles appliqués à la géométrie sous formes de contraintes d'esquisse, de distances entre des éléments géométriques... Présentées dans la deuxième partie du chapitre 1, les méthodes permettant de conduire une optimisation à partir des paramètres CAO sont de deux types : l'optimisation par méta-heuristiques et l'optimisation basée sur le gradient. Cette dernière nécessite une discrétisation de la forme CAO pour effectuer les simulations numériques à l'aide de la méthode des éléments finis et pour calculer le gradient. Les solveurs permettant de le calculer, lorsque cela est possible, fournissent la sensibilité de la fonction étudiée par rapport à la position des nœuds du maillage de calcul. Cette sensibilité est indispensable pour effectuer des optimisations par gradient. En contrepartie, la discrétisation supprime le lien avec la CAO et donc de nombreuses informations telles que les paramètres que nous cherchons à manipuler. D'autre part, les méta-heuristiques comme les plans d'expériences et les surfaces de réponse effectuent l'optimisation directement sur le modèle CAO. Ces méthodes sont appréciées par les industriels car elles permettent d'estimer en "boîte noire" les fonctions objectif tout en conservant le lien avec la CAO. Néanmoins, lorsque le solveur mécanique nécessite des temps de calcul importants ou que le nombre de paramètres devient élevé, les temps d'optimisation sont alors prohibitifs. Les méthodes par gradient n'étant pas limitées, quant à elles, par le nombre de paramètres, nous cherchons dans ces travaux à mettre au point une méthode d'optimisation d'une fonction objectif

liée à des paramètres **CAO** (longueurs, angles...) qui exploite le gradient donné sur le maillage, ce qui évite les inconvénients des deux approches précédentes.

La première méthode présentée dans ce manuscrit consiste à effectuer une différence finie sur le déplacement des nœuds d'une forme à l'autre. Ainsi, deux maillages différents sont comparés pour évaluer le déplacement des nœuds. Ces maillages sont générés avant et après la variation d'un paramètre. Leurs configurations respectives sont différentes en terme de connectivité. Pour pallier cela, une projection de ces deux maillages est effectuée dans un domaine commun. La comparaison dans ce domaine lève le verrou de l'absence d'isoconnectivité mais pose différentes problématiques. La première concerne la projection des formes ayant plusieurs frontières. La deuxième est posée par la précision liée à l'interpolation dans le domaine de projection. Nous proposons dans le chapitre 2 (p. 33) différentes méthodes pour répondre à ces questions.

La deuxième méthode se place avant la discrétisation de la forme. En effet, elle utilise la définition intrinsèque des formes **CAO** pour évaluer cette sensibilité. Une **CAO** est définie à l'aide de surfaces paramétriques. L'évolution de ces surfaces pour un patch donné nous renseigne sur l'impact des paramètres sur la forme. La méthode présentée dans le chapitre 3 (p. 69) s'appuie sur ces coordonnées paramétriques pour évaluer la sensibilité en tout point de la surface. Une comparaison avec la méthode basée sur les maillages est effectuée sur une surface simple et dans une boucle d'optimisation en mécanique des fluides.

Le dernier chapitre (p. 95) présente deux optimisations en mécanique des fluides. Ces deux optimisations utilisent le gradient issu d'un solveur adjoint qui est ensuite reporté sur les paramètres **CAO** avec la méthode de calcul de sensibilité sur les maillages. La première optimisation vise à minimiser la perte de charge dans un conduit de ventilation. Les gains constatés sont importants et confirment l'intérêt de l'approche proposée. À titre de comparaison, une méthode par plans d'expériences et surface de réponse est aussi effectuée sur ce coude. Le deuxième cas test permet d'effectuer une optimisation sur une géométrie complexe. Il s'agit d'un collecteur d'échappement à deux branches dont les pertes de charge respectives sont à optimiser. La complexité des modèles **CAO** optimisés, proches de cas industriels, montrent la pertinence et la validité de la méthode.

# Introduction à l'optimisation paramétrique de forme

---

<b>1.1</b>	<b>Deux formulations possibles d'un problème . . . . .</b>	<b>4</b>
1.1.1	Formulation du problème direct . . . . .	4
1.1.2	Formulation du problème inverse . . . . .	5
<b>1.2</b>	<b>Stratégies d'optimisation . . . . .</b>	<b>6</b>
1.2.1	Critères de choix de la méthode . . . . .	6
1.2.2	Méthodes exactes . . . . .	7
1.2.3	Méta-heuristiques . . . . .	8
<b>1.3</b>	<b>Familles de paramétrage . . . . .</b>	<b>10</b>
1.3.1	Optimisation topologique . . . . .	11
1.3.2	Optimisation géométrique . . . . .	12
1.3.3	Optimisation isogéométrique . . . . .	13
1.3.4	Optimisation paramétrique . . . . .	14
1.3.4.1	Les boîtes de morphing . . . . .	14
1.3.4.2	Paramètres CAO . . . . .	15
<b>1.4</b>	<b>Calcul du gradient d'une fonction objectif par rapport à la forme .</b>	<b>17</b>
1.4.1	Calcul par différences finies . . . . .	18
1.4.2	Calcul par dérivation directe . . . . .	18
1.4.3	Calcul par la méthode de l'état adjoint . . . . .	19
<b>1.5</b>	<b>Calcul de sensibilité d'une forme par rapport aux paramètres CAO</b>	<b>22</b>
1.5.1	Méthode analytique . . . . .	22
1.5.2	Méthodes numériques sur les géométries CAO . . . . .	23
1.5.3	Méthodes numériques sur le maillage . . . . .	24
1.5.3.1	Méthode de projection suivant la normale . . . . .	25
1.5.3.2	Méthode de projection harmonique appliquée aux formes monofrontières . . . . .	27
<b>1.6</b>	<b>Synthèse . . . . .</b>	<b>29</b>
	<b>Bibliographie du 1<sup>er</sup> chapitre . . . . .</b>	<b>30</b>

---

Afin d'avoir une vision globale du domaine de l'optimisation de forme, nous présentons les notions et les méthodes numériques disponibles à différentes étapes du processus de conception. Tout commence par la définition du problème. D'une manière générale, il est possible de poser un problème suivant deux formulations : la formulation directe et la formulation inverse. Le paragraphe 1.1 vise à renseigner le lecteur sur les principes de ces deux formulations. Nous nous intéressons ensuite plus particulièrement à la formulation inverse, plus adaptée à notre problème d'optimisation de forme. Une fois que le problème est posé, différentes stratégies d'optimisation sont présentées pour trouver un optimum (cf. paragraphe 1.2). En parallèle de cela, le paramétrage est un élément clé dans le domaine de l'optimisation de forme (cf. paragraphe 1.3). Choisi en cohérence avec le problème posé, il impacte de façon significative la démarche effectuée pour la résolution mais aussi son intégration dans un processus global de conception. Fréquente dans les différents paragraphes, la notion de gradient et son calcul sont détaillés dans le paragraphe 1.4. Finalement, les méthodes actuelles pour le calcul de la sensibilité d'une forme par rapport aux paramètres CAO sont présentées dans le paragraphe 1.5.

### 1.1 Deux formulations possibles d'un problème

D'après [Keller 1976], deux formulations d'un problème sont possibles. La première, appelée formulation directe, permet de déduire les effets en connaissant les causes. La deuxième consiste à déterminer les causes en connaissant des effets et est appelée formulation inverse. Les deux formulations sont présentées dans le paragraphe suivant sur un exemple d'application simple.

#### 1.1.1 Formulation du problème direct

La formulation directe est la plus naturelle et nous l'utilisons tous les jours, elle peut être assimilée à la notion de causalité. Nous connaissons l'ensemble des paramètres de cause et ainsi en déduire les effets. Il est possible de prédire l'état futur d'un problème en connaissant son état actuel. Si nous prenons un exemple industriel, cela revient à déterminer la fréquence des modes propres d'un élément de carrosserie ou la perte de charge d'un conduit de ventilation quand nous connaissons leur géométrie, les matériaux employés, les conditions aux limites, les conditions initiales...

La résolution d'un problème direct s'appuie sur deux types d'équations : les équations d'état et les équations d'observations.

Les équations d'état traduisent le phénomène physique impliqué dans notre problème. Elles sont données sous la forme de l'équation (1.1) :

$$F(\Omega, u) = 0 \tag{1.1}$$

Dans l'équation (1.1),  $F$  est un opérateur algébrique ou aux dérivées partielles comme, par exemple, les équations de Navier-Stokes pour la mécanique des fluides.  $\Omega$  est un ensemble de paramètres comme la forme d'une pièce donnée par la position des nœuds d'un maillage.  $u$  désigne un champ ou un ensemble de champs d'états physiques (température, vitesse, contraintes, etc.). Dans le cas des équations de Navier-Stokes, il correspond à la pression et à la vitesse d'un écoulement. Toutefois, nous ne cherchons pas à connaître uniquement la vitesse ou la pression

d'un fluide. De plus, cette information n'est souvent connue qu'en certains points de l'espace lors d'une résolution par éléments finis. Le concepteur se reporte plus souvent sur des fonctions plus précises mais moins triviales au regard des équations d'état présentées plus haut.

Pour faire correspondre les grandeurs physiques obtenues au problème à résoudre, on utilise des équations d'observations :

$$d = Hu \tag{1.2}$$

Dans cette équation,  $d$  appartient à l'espace des observations. L'équation (1.2) traduit une caractéristique<sup>a</sup> que le concepteur peut exploiter plus facilement (comme une perte de charge, l'uniformité d'un écoulement...) et permet le post-traitement du problème étudié. L'opérateur  $H$  utilise les informations provenant des équations d'état pour déterminer la valeur de la fonction observée. Dans le cas d'une perte de charge, cet opérateur est la différence entre l'intégrale surfacique de la pression en entrée et celle en sortie.

En résumé, la résolution d'un problème direct utilise successivement les deux équations présentées précédemment. Tout d'abord, la résolution de l'équation d'état permet de connaître les grandeurs physiques d'une forme lorsque celle-ci est fixée. Ensuite, les équations d'observations traduisent ces grandeurs physiques en fonctions plus faciles à interpréter ou correspondant aux exigences à satisfaire. Celles-ci sont définies par les ingénieurs de bureaux d'études dans les cahiers des charges.

### 1.1.2 Formulation du problème inverse

Dans le cas du problème inverse, la démarche est plus délicate. On cherche à déterminer quels sont les effets qui ont permis d'obtenir un état connu. Cela peut être utile dans diverses problématiques comme par exemple la reconstitution de l'état passé d'un système connaissant son état actuel ou l'identification de paramètres. En optimisation de forme, c'est cette problématique qui nous concerne puisque nous cherchons ici à trouver les valeurs de paramètres permettant d'atteindre une forme optimale.

[Kern 2016] présente de façon détaillée la résolution des problèmes inverses. Il distingue deux catégories principales : les problèmes linéaires et non linéaires. Il faut noter que la non-linéarité ne fait pas référence au problème direct (lorsque les paramètres sont connus) mais bien au problème inverse lui-même. Après avoir présenté les propriétés mathématiques des problèmes linéaires, il donne des méthodes permettant de les résoudre comme par exemple la décomposition en valeurs singulières (SVD). Pour les problèmes non-linéaires, l'approche est différente et la résolution de la fonctionnelle à minimiser est effectuée par des méthodes plus complexes. Les valeurs des paramètres étant liées de façon non-linéaires à nos phénomènes physiques, c'est cette dernière approche qui nous intéresse dans la suite de ce paragraphe.

Ainsi, nous présentons une fonctionnelle dans l'équation (1.3). Elle cherche à minimiser l'écart d'une fonction étudiée avec l'objectif fixé.

$$\min_{\Omega} J(\Omega) = \frac{1}{2} \|d_{\Omega} - d_{obj}\|^2 \tag{1.3}$$

---

a. En fonction de l'observation à effectuer, l'équation (1.2) n'est pas nécessairement linéaire ou algébrique.

$$\text{avec } d_\Omega = Hu_\Omega \tag{1.4}$$

$$\text{et } F(\Omega, u_\Omega) = 0 \tag{1.5}$$

$$\text{et } C(\Omega) \leq 0 \tag{1.6}$$

Nous retrouvons les termes de la formulation du problème direct mais, cette fois-ci, les paramètres sont à déterminer pour atteindre l'objectif fixé ou au moins s'en approcher. Avant d'écrire la formulation inverse d'un problème, il est préférable de bien connaître le problème direct car les équations directes d'état et d'observations deviennent des équations de contraintes qu'il faut respecter lors de la recherche des paramètres. Dans le cas présent, les paramètres restent la forme de la pièce. Nous utilisons aussi une inéquation portant sur la forme qui traduit les contraintes à prendre en compte dans l'optimisation comme par exemple les contraintes de fabrication. Nous verrons dans le paragraphe 1.3, les différents types de paramètres pouvant définir une forme et aussi leur capacité vis-à-vis des contraintes.

Avant cela, il est important de faire un choix quant à la stratégie d'optimisation utilisée pour atteindre la solution optimale. Le concepteur dispose de nombreuses méthodologies qu'il doit sélectionner en fonction de la nature mathématique de son problème, de ses ressources et de ses exigences.

## 1.2 Stratégies d'optimisation

Afin de déterminer la ou les solutions permettant de satisfaire un objectif tout en respectant d'éventuelles contraintes, l'optimisation permet de résoudre analytiquement ou numériquement les problèmes. Le choix parmi ces méthodes s'effectue suivant différents critères dont les ordres de priorité peuvent varier. Nous en citons quelques-uns dans la partie suivante.

### 1.2.1 Critères de choix de la méthode

#### – Nature de l'espace de recherche du problème

Les méthodes d'optimisation combinatoire (ou méthodes de recherche opérationnelle) sont bien adaptées pour résoudre les problèmes caractérisés par un espace de recherche discret. Les méthodes continues, quant à elles, permettent d'explorer des espaces de recherche avec des variables continues.

#### – Exploration du domaine de recherche

Les méthodes stochastiques s'appuient sur une répartition aléatoire des points de calculs d'une itération. Dans le cas des méthodes déterministes, les nouveaux points de calculs sont déterminés en fonction des itérations précédentes.

### – Solution unique ou population de solutions

Un autre critère pour choisir les méthodes d'optimisation peut être le nombre de points évoluant durant le processus de recherche des solutions optimales. Nous distinguons deux familles principales : les algorithmes faisant évoluer une solution unique (recherche locale, recherche tabou...) et les algorithmes basés sur l'évolution d'une population de points (algorithmes génétiques, réseau de neurones...). L'utilisation des méthodes basées sur une population de points requiert un temps de calcul très important. Il faut donc, pour cela, disposer de ressources **HPC**.

### – Différentiabilité de la fonction objectif

Nous distinguons deux familles de problèmes : la première s'intéresse aux problèmes pour lesquels la dérivée de  $J$  est calculable (méthodes exactes). Tandis que la deuxième s'attaque aux problèmes pour lesquels nous ne disposons pas d'informations sur la fonction  $J$  (problèmes dit "boîte noire"). À partir de ce constat, nous pouvons remarquer que ce critère est déterminant pour le choix de la méthode, principalement pour les problèmes d'optimisation industriels. Nous proposons de détailler ces deux familles de méthodes dans ce qui suit.

#### 1.2.2 Méthodes exactes

Aussi appelées *méthodes basées sur le gradient*, les méthodes exactes s'appuient sur des informations connues pour déterminer le prochain point de recherche. Pour cela, ces algorithmes nécessitent de connaître le gradient de la fonction étudiée qui peut être une direction de descente si le premier n'est pas calculable explicitement. Pour plus de détails sur les méthodes basées sur le gradient, le lecteur peut se reporter aux livres [Culioli 1994, Kelley 1999].

Connaissant le gradient, l'algorithme détermine une direction dans laquelle le calcul d'un nouveau point doit fournir une valeur meilleure que celle du point précédent. Pour ces méthodes, les variables d'optimisation doivent être continues mais leur dimension n'est pas limitée. En ce qui concerne les fonctions d'optimisation ou les contraintes, qu'elles soient bruitées ou non, elles doivent être différentiables afin de calculer le gradient. Dans le cas où la fonction  $J(\Omega)$  est différentiable et convexe (au moins localement), le minimum (local) de la fonction notée :

$$\min_{\Omega} J(\Omega) \tag{1.7}$$

est obtenu lorsque

$$\nabla J(\Omega) = 0 \tag{1.8}$$

Les contraintes jouent aussi un rôle dans la recherche de l'optimum. En effet, le respect de ces contraintes peut entraîner que l'optimum du problème soit atteint sans vérifier l'équation (1.8). Dans ce cas, on s'intéresse aux conditions d'optimalité de Karush–Kuhn–Tucker.

Lorsque nous évaluons un seul point de recherche par itération, c'est-à-dire que la population est égale à 1, chaque calcul est censé améliorer la solution de la fonction à optimiser. Dans le cas d'un nombre limité de calculs, nous sommes assurés d'obtenir une meilleure configuration. Cependant, une telle méthode est sensible aux minima locaux, c'est-à-dire lorsque la fonction

objectif est non-convexe. Afin de pallier cela, il est possible d'augmenter la taille de la population et d'effectuer ainsi différentes initialisations de l'algorithme pour éviter de tomber dans un minimum local.

Toutefois, l'obtention du gradient n'est pas toujours évidente car le coût impliqué par son calcul est souvent trop important et il se peut qu'il ne soit pas calculable. Les méthodes exactes ne sont donc pas adéquates dans ces cas-là.

### 1.2.3 Méta-heuristiques

Les méta-heuristiques sont des méthodes qui permettent de résoudre, en un temps raisonnable, des problèmes pour lesquels nous ne disposons pas d'information *a priori* sur la fonction  $J$  à optimiser. Ces méthodes cherchent à réduire le nombre de calculs nécessaires en analysant les solutions déjà parcourues. Cette analyse, basée sur des règles empiriques, détermine les corrélations qu'il peut y avoir entre différentes variables ou les zones dans lesquelles il est plus intéressant d'orienter la recherche. L'initialisation d'un processus heuristique est souvent stochastique. En effet, on peut avoir recours à une exploration aléatoire du domaine avant d'effectuer l'analyse des données et ensuite réduire le nombre de calculs. Les méthodes méta-heuristiques effectuent donc un tri sur une population de points de recherche et limitent ainsi les chemins à envisager pour atteindre un optimum. Les opérations de tri diffèrent selon la méthode utilisée, la nature des variables...

Il existe un nombre important de méthodes méta-heuristiques parmi lesquelles on peut citer : le recuit simulé, les algorithmes génétiques, la recherche tabou et l'optimisation par essais particuliers. [Talbi 2009] apporte une vision détaillée sur ces méthodes. Elles sont peu sensibles aux minima locaux mais la solution trouvée est, dans la plupart des cas, une solution approchée de la solution optimale du problème. Ces méthodes ne requièrent pas obligatoirement la différentiabilité de la fonction  $J$ .

Dans le cas d'un problème multiobjectif, l'ensemble des configurations qui dominent les autres vis-à-vis des objectifs forment ce que l'on appelle un front de Pareto. En effet, les fonctions objectif sont souvent antagonistes ce qui implique l'absence de configurations minimisant les deux objectifs simultanément. On obtient donc une famille de configurations optimales qui améliore un objectif au détriment du deuxième. L'illustration 1.1 représente ce front pour un cas bi-objectif. En industrie, le concepteur s'appuie souvent sur des méthodologies d'aide à la décision qui utilisent ce front pour déterminer la solution optimale qui leur convient le mieux. Les travaux récents de [Binois 2015] permettent de quantifier les incertitudes sur ces fronts.

### Plans d'expériences et surfaces de réponse

La méthode d'optimisation par plans d'expériences consiste à explorer l'ensemble d'un espace de conception admissible de façon structurée. Cette exploration permet de construire une approximation d'un critère sous la forme d'un méta-modèle, aussi appelé surface de réponse [Benoist 1994]. Cette surface de réponse est une représentation continue obtenue suite à l'interpolation d'essais donnant la valeur, à des points du plan, du critère considéré. L'optimisation est ensuite effectuée sur cette surface afin de trouver des points optimaux. La figure 1.2 présente les différentes étapes de la conduite d'une optimisation par plans d'expériences et surfaces de réponse.

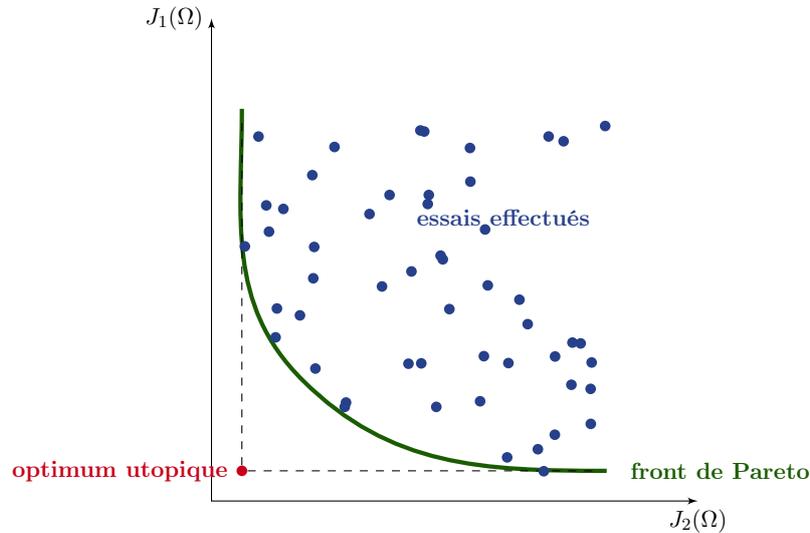


FIGURE 1.1 – Illustration du front de Pareto pour un problème bi-objectif

Un des principaux avantages de cette méthode est qu'elle n'est pas intrusive vis-à-vis des outils utilisés par le concepteur car elle permet d'estimer en "boîte noire" des fonctions, il n'y a donc pas de condition sur le type de la fonction optimisée. Il faut toutefois porter une attention particulière au choix du plan et de l'optimisation de la surface de réponse. En effet, ces choix doivent être effectués en cohérence avec le modèle, les paramètres, leur nombre ou encore la physique étudiée dans le but de mener à bien ces optimisations. La liste non-exhaustive ci-dessous donne un rapide aperçu des plans et des méthodes d'exploitation des surfaces de réponse. De plus amples informations sont disponibles dans la littérature [Benoist 1994, Goupy 2013] et permettent la compréhension de leur fonctionnement et de leur mise en place.

- Types de plans d'expériences :
  - les plans complets ;
  - les plans optimaux (D-optimaux, G-optimaux) ;
  - les plans fractionnaires ;
  - les plans LHS (pour « latin hypercube sampling »).
- Méthode de construction de surfaces de réponse :
  - modèles polynomiaux ;
  - modèles MARS et poly-MARS ;
  - modèle de krigeage ;
  - mélange d'expert.

Il faut noter que l'augmentation du nombre de paramètres (aux alentours de 100) rend plus difficile la convergence de la surface de réponse et nécessite aussi de considérer plus de points de calcul.

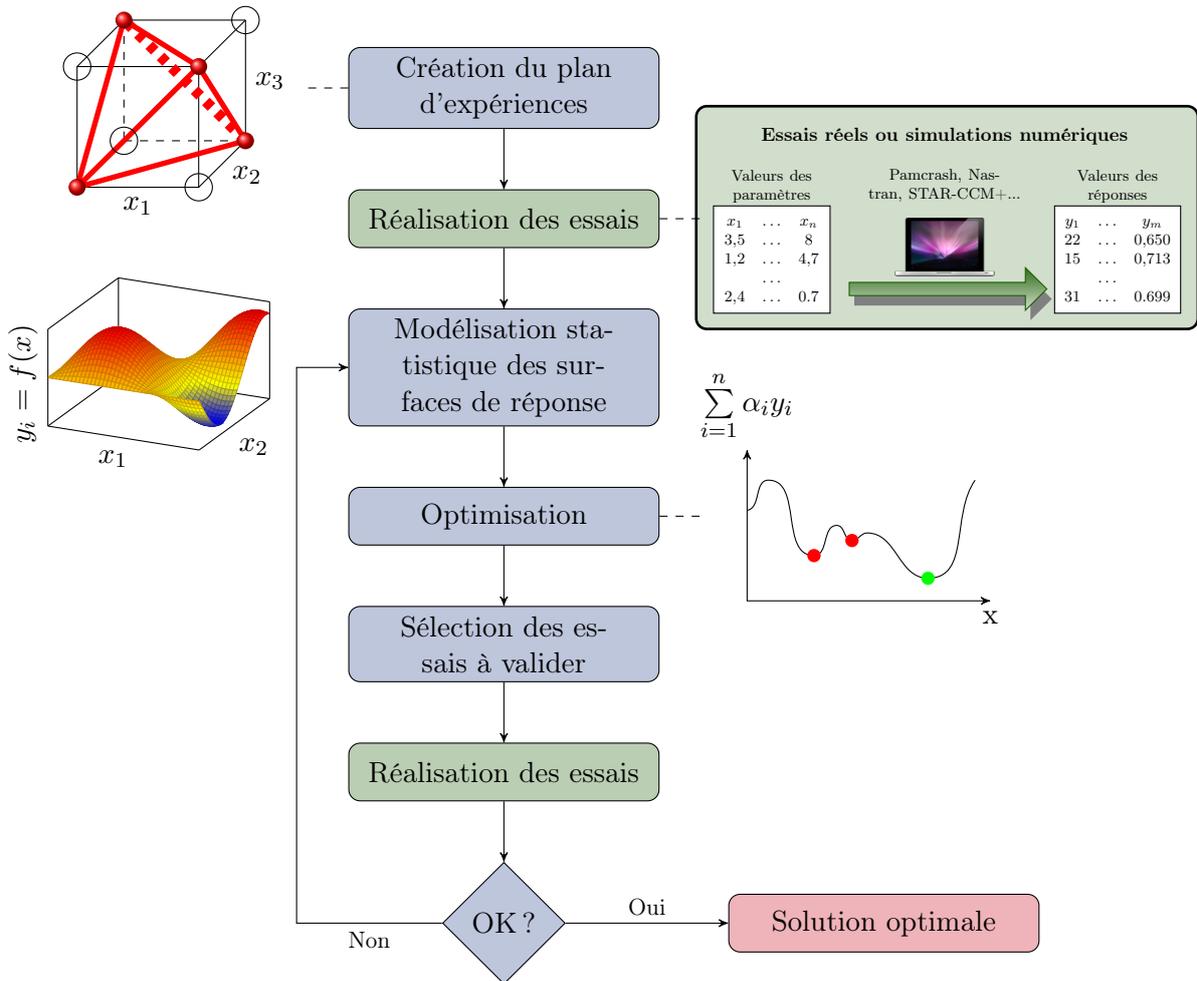


FIGURE 1.2 – Schéma de mise en œuvre d’une méthode d’optimisation par plan d’expériences  
 Source : [Froment 2014]

### Synthèse

Les méta-heuristiques et plus particulièrement les plans d’expériences et la construction de surfaces de réponse sont largement utilisés dans l’industrie car ils ne requièrent pas de connaissance particulière sur la fonction étudiée. Toutefois, ces méthodes nécessitent un nombre de calculs important, ce qui se traduit par des temps d’étude très longs surtout dans le cas de calculs lourds comme la mécanique des fluides. L’utilisation de méthodes exactes réduit significativement le nombre de calculs nécessaires et donc la durée des phases d’optimisation mais le gradient doit être calculable, ce qui n’est pas toujours le cas.

### 1.3 Familles de paramétrage

Nous cherchons à effectuer une optimisation de forme pour la conception industrielle. Dans ce cas, la forme de la structure est la variable de notre problème. Nous allons donc nous in-

téresser aux différentes manières de paramétrer une forme. Les différentes familles de paramétrage sont adaptées aux méthodes d'optimisation les plus courantes pour l'optimisation de forme [Saitou 2005, Allaire 2007]. En accord avec ces travaux, ces familles vont être présentées dans l'ordre<sup>b</sup> décroissant du nombre de degrés de liberté.

### 1.3.1 Optimisation topologique

Dans le cadre de l'optimisation topologique, on définit un volume admissible de conception dans lequel on souhaite déterminer la forme optimale. Par exemple, dans le cas de l'optimisation d'un conduit moteur, le volume admissible est donné par l'espace disponible entre les autres éléments déjà présents. L'ensemble de ce volume est discrétisé et la densité de chaque élément constitue un degré de liberté. **Les paramètres sont la densité de matière associée à chaque élément.** En effet, chaque cellule peut avoir une densité allant de 0, caractérisant un matériau vide, à 1, caractérisant un matériau plein. Pour ce paramétrage, le problème d'optimisation de forme de l'équation (1.3) revient à écrire l'équation (1.9).

$$\min_R J(R) \quad \text{avec} \quad R = [\rho_1, \rho_2, \dots, \rho_K] \quad (1.9)$$

Où  $K$  désigne le nombre d'éléments  $R$  du volume admissible.

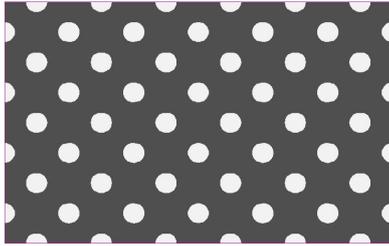
Le passage continu de la densité de 0 à 1 est nécessaire pour effectuer une relaxation du problème combinatoire obtenu, impossible à gérer autrement [Bendsøe 2003]. Le calcul du gradient est donc possible ainsi que la résolution du problème avec des méthodes classiques d'optimisation par gradient. Toutefois, différentes approches sont mises en place pour traiter les éléments ayant des densités intermédiaires qui rendent l'interprétation de la solution compliquée. Actuellement, les méthodes les plus connues sont appliquées en mécanique du solide :

- la méthode SIMP (pour **solid isotropic material with penalization**) utilise une pénalisation sur la densité des éléments. Cette méthode est détaillée dans [Bendsøe 2003].
- la méthode présentée par [Allaire 2004] est basée sur l'utilisation de la théorie des lignes de niveaux, plus couramment appelées *level-set* [Osher 1988].

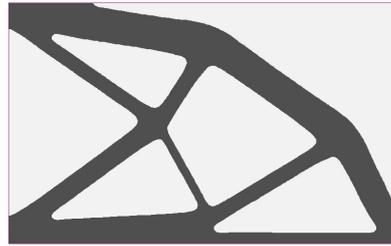
Ce paramétrage est intéressant car la topologie n'est pas fixée *a priori*. L'espace de conception disponible est donc exploré dans sa globalité. Toutefois, il est difficile de formuler des contraintes géométriques complexes pour tenir compte des problèmes liés à l'esthétique ou à la fabrication (cf. équation (1.6)). La méthode des *level-set* a récemment été utilisée dans le projet FUI RODIN<sup>®</sup> [Albertelli 2015, Dapogny 2013, Michailidis 2014] et permet de connaître les frontières de la forme au cours de son optimisation. En outre, avec cette méthode, il est possible de traduire mathématiquement certaines contraintes géométriques comme des épaisseurs minimales ou maximales de matière (cf. figure 1.3).

Pour obtenir un modèle CAO utilisable dans la suite des processus de développement, il est nécessaire de traduire la forme donnée par les densités de chaque élément en un modèle CAO constitué de primitives telles que des volumes élémentaires ou des surfaces de Bézier. Ici, cette étape, aussi appelée le retour CAO, est difficile à effectuer et peut déprécier fortement l'optimum trouvé car la topologie de la forme doit être adaptée aux standards CAO.

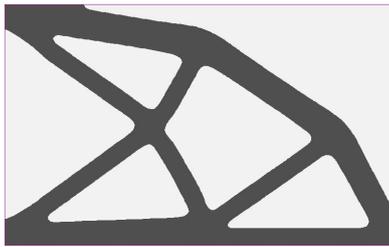
b. Pour effectuer ce classement, on considère un problème équivalent pour toutes les méthodes.



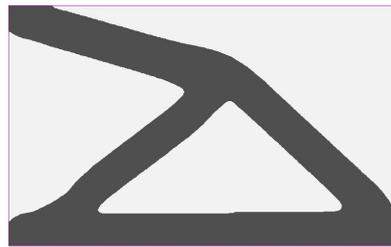
1.3.a – Initialisation du problème



1.3.b – Forme optimale sans contrainte d'épaisseur minimale



1.3.c – Forme optimale avec la contrainte d'épaisseur minimale  $d_{min} = 0.15$



1.3.d – Forme optimale avec la contrainte d'épaisseur minimale  $d_{min} = 0.3$

FIGURE 1.3 – Optimisation topologique sur un cantilever avec contraintes d'épaisseur minimale  
*Source : [Michailidis 2014]*

En mécanique des fluides, l'approche topologique n'est pas basée sur la densité des éléments mais sur le coefficient de Darcy. Ce terme traduit l'écoulement d'un fluide dans un milieu poreux [Darcy 1857]. De la même façon que pour la densité d'un élément, il est possible de faire varier la porosité des éléments d'un volume pour avoir des éléments totalement perméables ou au contraire imperméables [Borrvall 2003]. Toutefois, l'absence de frontière nette est préjudiciable pour l'utilisation de ces méthodes en CFD, dont les calculs requièrent de bien connaître les phénomènes aux parois. Dans ce cas, une valeur seuil est choisie pour définir la frontière du domaine. D'autres outils, comme par exemple le logiciel TOSCA Fluid<sup>®</sup> (édité par DS SIMULIA<sup>®</sup>), se basent sur la prévention des retours d'écoulement pour déterminer les éléments du domaine à sédimenter pour éviter les refoulements [Moos 2004]. [Froment 2014] présente le fonctionnement de cette méthode qui se passe de l'utilisation du gradient pour déterminer la forme optimale en aérodynamique interne mais qui impose au concepteur de sélectionner un niveau de coupure adéquate.

### 1.3.2 Optimisation géométrique

Les calculs par éléments finis utilisent la discrétisation de la forme pour calculer les grandeurs physiques. Ce maillage représente donc la géométrie étudiée. L'optimisation géométrique fait évoluer la discrétisation de la forme au cours du processus d'optimisation. **Les paramètres sont les coordonnées de chaque point du maillage.** Ici, le problème (1.3) peut être traduit par l'équation (1.10).

$$\min_X J(X) \quad \text{avec} \quad X = [x_1, y_1, z_1, x_2, y_2, z_2, \dots, x_L, y_L, z_L] \quad (1.10)$$

avec  $L$  : le nombre de nœuds  $X$  du maillage.

Afin de déplacer chacun des nœuds du maillage, l'optimisation s'appuie sur le gradient de la fonction objectif. Sous réserve d'être capable de le calculer, ce gradient est la sensibilité de la fonction objectif par rapport au déplacement des nœuds, il renseigne ainsi sur la direction dans laquelle il faut déplacer les points pour améliorer la fonction objectif. L'utilisation brute du gradient pour effectuer un morphing risque de dégrader le maillage au fil des itérations car chaque nœud se déplace sans prendre en compte le déplacement de ses voisins. Dans la pratique, le gradient est souvent lissé par un opérateur de type laplacien pour éviter les variations trop importantes entre deux nœuds. Dans la majorité des cas, le maillage de départ est obtenu par la discrétisation d'une géométrie CAO. Afin de revenir à ce modèle CAO après l'optimisation, il faut modifier la géométrie pour qu'elle corresponde au maillage final en jouant sur les paramètres CAO ou en changeant sa construction. Cette étape représente un travail important pour le concepteur.

Depuis peu, [Dapogny 2013] propose une méthode basée sur l'utilisation des lignes de niveaux (ou « *level-set* ») et qui permet de faire évoluer le maillage tout en garantissant une haute qualité comme le montre la figure 1.4. Cette méthode permet aussi de modifier le nombre de frontières au cours de l'optimisation en fonction d'un critère de proximité de surface. Cela permet de s'orienter vers de meilleures solutions.

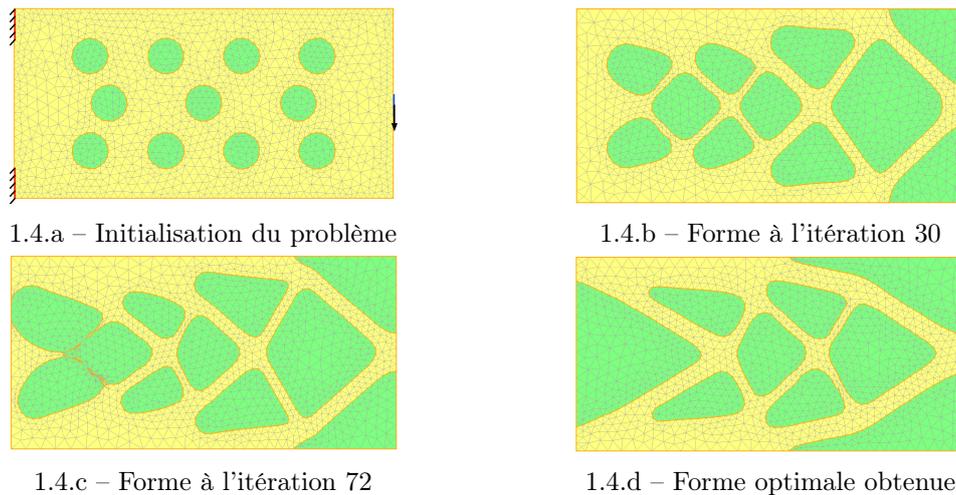


FIGURE 1.4 – Optimisation géométrique sur un cantilever

Source : [Dapogny 2013]

### 1.3.3 Optimisation isogéométrique

L'analyse isogéométrique a été introduite par Hughes [Hughes 2005]. Contrairement aux éléments finis classiques, les calculs ne sont pas effectués sur le maillage de la forme mais directement sur la définition CAO de celle-ci.

Pour cela, cette méthode éléments finis d'ordre élevé (*high-order FEM*) impose d'utiliser des polynômes rationnels pour représenter des formes canoniques comme des cercles, des surfaces, des cylindres... Les logiciels de CAO s'appuient communément sur cette définition de la forme :

les patches constituent donc les éléments et les fonctions (splines, B-splines, NURBS...), qui permettent de les construire, sont les fonctions d'interpolation. La méthode présentée dans les travaux de [du **Cauzé de Nazelle 2013**] utilise **les positions des points de contrôle des splines comme paramètres** (cf. figure 1.5). L'équation (1.11) est une réécriture du problème (1.3) pour ce type de paramétrage :

$$\min_P J(P) \quad \text{avec } P = [x_{P_1}, y_{P_1}, z_{P_1}, x_{P_2}, y_{P_2}, z_{P_2}, \dots, x_{P_M}, y_{P_M}, z_{P_M}] \quad (1.11)$$

avec  $M$  : le nombre de points de contrôle  $P$  des patches CAO.

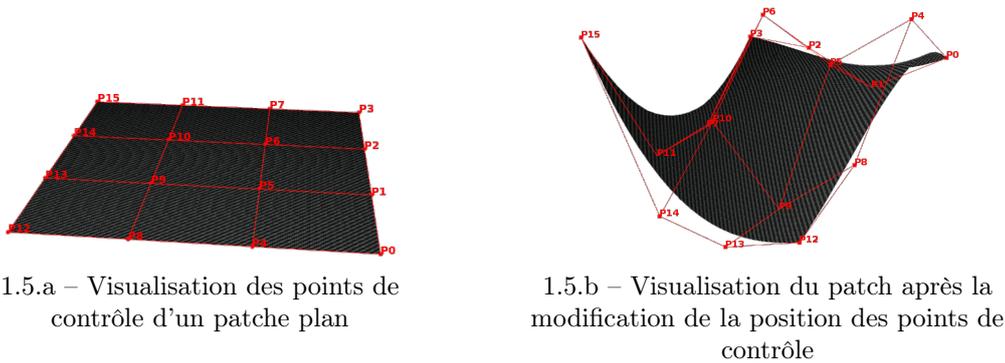


FIGURE 1.5 – Visualisation des points de contrôle d'un patch CAO

Source : [Julisson 2016]

Le fait de se reporter à la définition de la CAO évite l'utilisation d'un outil de maillage et l'approximation que la discrétisation implique sur le calcul des grandeurs physiques. En contrepartie, il faut être capable d'obtenir cette définition, ce qui est généralement compliqué. Une des principales raisons vient du fait qu'en ingénierie, l'utilisation des carreaux de splines est rare en dehors des solveurs géométriques et ces derniers ne permettent pas de récupérer facilement ces informations. L'exploitation des formats neutres comme les fichiers STEP ou IGES est relativement efficace pour obtenir les informations nécessaires mais le lien avec le modèle original et son paramétrage haut niveau s'en trouve rompu.

Le lien entre les paramètres utilisés pour ce type d'optimisation – à savoir les points de contrôle des courbes – et les contraintes géométriques est généralement difficile à déterminer.

### 1.3.4 Optimisation paramétrique

D'après [Allaire 2007], l'optimisation paramétrique s'appuie sur un nombre réduit de variables. Dans ce paragraphe, nous présentons deux paramétrages qui répondent à cette dénomination : le premier utilise des **boîtes de morphing** sur le maillage et le second se base sur des **paramètres du modèle CAO**.

#### 1.3.4.1 Les boîtes de morphing

Le nombre de paramètres à manipuler dans le cas d'une optimisation géométrique est très important. Afin de réduire le nombre de paramètres tout en utilisant les maillages, il est possible

de le paramétrer [Yang 1992]. Cela consiste à mettre en place des *boîtes de morphing* afin de piloter une quantité assez large de nœuds avec un nombre de paramètres limité. **Les paramètres sont les dimensions des boîtes** qui peuvent être cubiques, cylindriques... La position des nœuds à l'intérieur varie en fonction des dimensions de la boîte. Le choix du nombre de boîtes placées sur le maillage doit être fait en corrélation avec la finesse de modification de la forme attendue. Toutefois, les problématiques de retour CAO et de contraintes géométriques ne sont pas plus aisées avec ce type de paramétrage.

#### 1.3.4.2 Paramètres CAO

Les logiciels de CAO sont largement utilisés dans l'industrie pour modéliser les pièces ou les ensembles de pièces. Ainsi, ils permettent de modéliser géométriquement la pièce en elle-même mais aussi de gérer son intégration dans un ensemble plus important. Les suites logicielles les plus connues sont CATIA<sup>®</sup> édité par Dassault Systèmes, NX<sup>®</sup> publié par Siemens PLM Software, Creo<sup>®</sup> fait par PTC ou encore des plates-formes Open Source comme Salomé<sup>®</sup> et le noyau géométrique Open-Cascade<sup>®</sup>.

La construction des modèles CAO est effectuée à partir de zéro ou à partir de surfaces déjà existantes pouvant provenir d'une ancienne pièce ou du département design. Dans ce cas, on parle plus souvent de "CAO morte" puisque la définition de ses patches n'est pas explicite. [Froment 2014] propose une méthode pour aider le concepteur à reconstruire un modèle CAO à partir de ces surfaces mortes.

Une fois le modèle construit ou au cours de sa construction, il est possible de définir **des paramètres de dimensionnement**. Ces paramètres sont de différentes natures et répondent à des exigences différentes. Ils peuvent correspondre à des exigences architecturales comme par exemple :

- l'empattement d'un véhicule ou la hauteur de son hayon ;
- des contraintes de fabrication comme une épaisseur minimale de matière ou un critère d'angle pour le démoulage ;
- des exigences du design comme la position d'une ligne de design sur une portière ou le découpage d'un phare de voiture.

Ces paramètres peuvent être des paramètres géométriques élémentaires (épaisseur, rayon de congé, longueur, angle...) comme sur la figure 1.6, des relations en fonction d'autres paramètres (rapport hauteur/largeur...) et aussi des présences/absences d'éléments (nombre de raidisseurs...). L'ensemble de ces paramètres peuvent être gérés à partir d'une table de paramétrage. Lors de l'optimisation d'un modèle CAO, il est possible de sélectionner tous les paramètres ou seulement une partie. Cela permet de ne pas effectuer l'optimisation vis-à-vis de paramètres moins influents ou même inutiles pour le cas étudié. L'adaptation du problème (1.3) pour ce type de paramétrage donne le problème (1.12).

$$\min_{\alpha} J(\alpha) \quad \text{avec } \alpha = [\alpha_1, \alpha_2, \dots, \alpha_N] \quad (1.12)$$

avec  $N$  : le nombre de paramètres  $\alpha$  du modèle CAO.

L'exploration de l'espace de conception est directement liée au nombre de paramètres de la géométrie : plus il y a de paramètres, plus la géométrie est flexible et balaye l'espace de

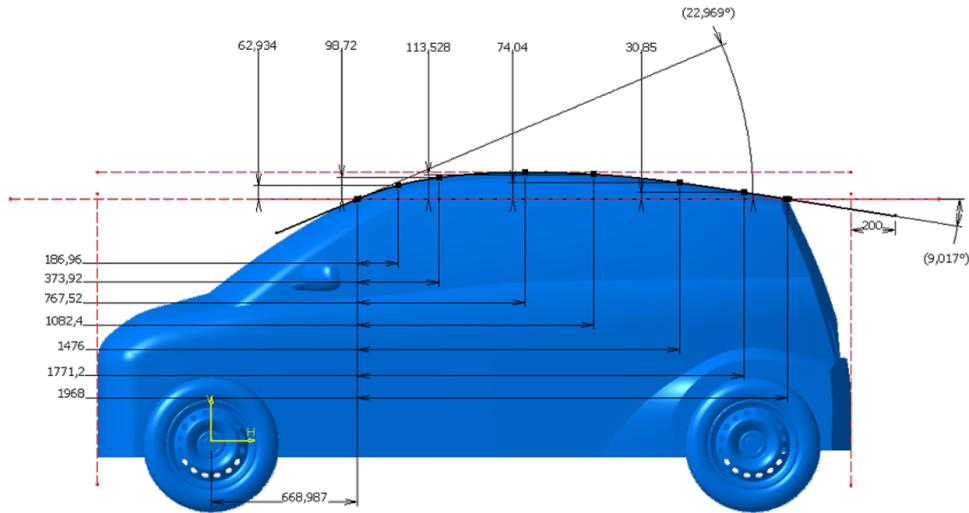


FIGURE 1.6 – Paramètres CAO sur un pavillon de véhicule

conception. Toutefois, l'augmentation du nombre de paramètres peut réduire la robustesse du modèle. En effet, le taux de reconstruction <sup>c</sup> d'une géométrie peut vite être impacté par un nombre trop important de paramètres. Ce taux de reconstruction traduit les échecs de génération d'une géométrie pour différents jeux de paramètres. Ces échecs sont dus à des incompatibilités entre des paramètres (intersections de surfaces...), à l'absence ou à la multiplicité de solutions du problème de géométrie sous-jacent, des systèmes d'équations sous et sur-contraints, des bogues intrinsèques aux logiciels... Il est donc nécessaire de trouver un compromis raisonnable entre le nombre de paramètres du modèle et l'exploration du domaine de conception. La plupart des solveurs physiques se basent sur un maillage pour effectuer les simulations numériques. Il est donc nécessaire d'utiliser un mailleur pour discrétiser la forme sur laquelle les calculs sont effectués. Cette étape ne permet pas de garder le lien avec le paramétrage mis en place.

## Synthèse

Dans cette partie, nous avons vu les différents paramétrages à disposition du concepteur pour effectuer des optimisations de forme et ainsi répondre de la meilleure façon à ses attentes en fonction de ses besoins et de ses contraintes. Afin de faciliter la comparaison de ces méthodes, nous proposons de les rassembler dans le tableau 1.1 avec différents critères de comparaison. Une optimisation topologique ou géométrique s'appuie sur l'ensemble des éléments d'un domaine, le nombre de paramètres est donc très important contrairement aux autres familles de paramétrage (cf. 2<sup>ème</sup> colonne du tableau 1.1). De plus, lorsqu'un modèle CAO est construit ou lors de la mise en place de boîtes de morphing, le paramétrage est laissé à l'appréciation des concepteurs et peut varier suivant les attentes de chacun. Les optimisations topologique et géométrique sont, quant à elles, peu dépendantes du concepteur (cf. 3<sup>ème</sup> colonne du tableau 1.1). Comme nous l'avons vu dans les précédents paragraphes, la prise en compte de contraintes géométriques est

c. Le taux de reconstruction est calculé en fonction du nombre de configurations générées par rapport au nombre de configurations données en entrée.

## 1.4. Calcul du gradient d'une fonction objectif par rapport à la forme

plus ou moins facilitée en fonction du paramétrage utilisé. Cette information est synthétisée dans la quatrième colonne du tableau 1.1.

Type d'optimisation	Types de paramètres	Nombre de paramètres	Dépendance au concepteur	Capacité à respecter des procédés de fabrication	Prérequis
<b>Topologique</b>	densité/porosité des éléments	$10^6, 10^7$	faible	–	calcul des gradients
<b>Géométrique</b>	position des nœuds	$10^6, 10^7$	faible	–	calcul des gradients & déformation de maillage
<b>Isogéométrique</b>	points de contrôle	$10^2, 10^3$	moyenne	–	géométrie définie par carreaux de splines & solveur spécifique
<b>Paramétrique</b>					
Boîtes de morphing	dimensions	$10^1, 10^2$	très forte	+	paramétrisation & automatisation
Paramètres CAO	dimensions	$10^1, 10^2$	très forte	+++	paramétrisation & automatisation

Tableau 1.1 – Comparaison entre les différents types de paramétrage

Dans notre cas, nous nous appuyons sur les avantages qu'apportent les modèles CAO en terme d'intégration dans les processus industriels et en particulier la prise en compte des contraintes de fabrication et la simplicité de mise en œuvre. Dans la suite de ces travaux, nous cherchons à effectuer des optimisations paramétriques sur les paramètres CAO.

## 1.4 Calcul du gradient d'une fonction objectif par rapport à la forme

Actuellement, dans l'industrie, les optimisations paramétriques sont largement utilisées pour profiter de l'intelligence embarquée dans les modèles CAO. Toutefois, ces modèles sont le plus souvent optimisés avec des méta-heuristiques comme des plans d'expériences et la construction de surfaces de réponse (cf. partie 1.2.3). L'utilisation des fonctions en "boîte noire" a favorisé la généralisation de ces méthodes. Pourtant, il est possible de calculer le gradient sur des critères d'optimisation couramment utilisés. De plus, dans le paragraphe 1.2.2, nous avons vu l'intérêt de connaître le gradient pour la résolution d'un problème d'optimisation par des méthodes exactes. D'après [Othmer 2005, Kern 2016], il est possible de le calculer suivant trois approches présentées ci-après.

Afin d'éviter toute confusion avec les types de paramétrage présentés précédemment, nous ne ferons pas de choix quant à la paramétrisation dans cette partie. La forme  $\Omega$  est paramétrée par un nombre  $P$  de paramètres notés  $\omega$ . Nous rappelons ici l'équation d'état (1.1) en utilisant les paramètres  $\omega$ .

$$F(\omega, u(\omega)) = 0 \tag{1.1}$$

Dans cette partie, la fonction objectif est donnée par (1.13) et traduit le problème d'optimisation :

$$J(\omega, u(\omega)) \tag{1.13}$$

**Remarque 1.4.1** *Dans le cas général, les méthodes suivantes ne peuvent pas être appliquées pour tout problème d'optimisation mais dans de nombreux cas la dérivation par rapport aux paramètres est possible. Ainsi, les termes utilisés dans les parties suivantes sont calculables soit de manière explicite lorsque le lien entre la fonction et ses paramètres est connu, soit à l'aide d'un logiciel de calcul.*

### 1.4.1 Calcul par différences finies

Le calcul par différences finies [Boole 1860] est assez répandu car il est relativement simple à mettre en œuvre. Toutefois, son coût peut vite devenir une limite pour son utilisation puisqu'il est proportionnel au nombre de paramètres à évaluer. De plus, cette méthode ne donne qu'un résultat approché du gradient de la fonction. L'équation (1.14) montre le calcul par cette méthode :

$$\frac{\partial J}{\partial \omega_i} \approx \frac{J(\omega + \delta \omega_i) - J(\omega)}{\delta \omega_i}, \quad i = 1 \dots P \tag{1.14}$$

Nous constatons dans l'équation (1.14) que le nombre d'évaluations de  $J$  est égal au nombre de paramètres du problème sans compter l'évaluation de la solution courante  $J(\omega)$ , soit  $P + 1$  lancements de calcul physique. Cela représente donc un effort de calcul trop important et en particulier en mécanique des fluides où les temps de calculs sont très longs.

### 1.4.2 Calcul par dérivation directe

Le calcul par fonctions de sensibilité règle le problème de précision du calcul du gradient sans pour autant réduire le nombre de calculs. En effet, cette méthode [Adelman 1986] consiste à dériver l'équation d'état (1.1) explicitement par rapport à chaque paramètre :

$$\frac{dF}{d\omega_i} = \frac{\partial F}{\partial u} \frac{\partial u}{\partial \omega_i} + \frac{\partial F}{\partial \omega_i} = 0 \quad \text{car} \quad F(\omega, u(\omega)) = 0, \quad \forall \omega \tag{1.15}$$

Il est ensuite possible de déduire le terme  $\frac{\partial u}{\partial \omega_i}$  en résolvant pour chaque paramètre  $\omega_i$  le système linéaire :

$$\frac{\partial F}{\partial u} \frac{\partial u}{\partial \omega_i} = - \frac{\partial F}{\partial \omega_i} \tag{1.16}$$

Le terme  $\frac{\partial u}{\partial \omega_i}$  est utilisé dans l'équation de la dérivée de l'équation (1.13) par rapport à chaque paramètre :

$$\frac{dJ}{d\omega_i} = \frac{\partial J}{\partial u} \frac{\partial u}{\partial \omega_i} + \frac{\partial J}{\partial \omega_i} \tag{1.17}$$

Il est donc possible d'obtenir la sensibilité de la fonction objectif  $J$  en fonction de chaque paramètre. Toutefois, on remarque que le nombre de lancements du solveur est proportionnel au nombre de paramètres.

### 1.4.3 Calcul par la méthode de l'état adjoint

Différentes formulations sont possibles afin d'obtenir les équations adjointes. Nous utilisons le formalisme proposé par [Cacuci 1981, Céa 1986] qui fait intervenir un lagrangien associé à la fonction objectif  $J$  et à l'équation d'état au travers d'un multiplicateur de Lagrange  $\lambda$ . Ce formalisme permet ainsi d'automatiser l'obtention d'un état adjoint. Celui-ci est donné par :

$$\mathcal{L}(\omega_i, u(\omega_i)) = J(\omega_i, u(\omega_i)) + \lambda^t F(\omega_i, u(\omega_i)) \quad (1.18)$$

En différentiant le lagrangien par rapport aux paramètres  $\omega_i$  et en imposant  $\frac{d\mathcal{L}}{d\omega_i} = 0$ , on obtient :

$$\begin{aligned} \frac{d\mathcal{L}}{d\omega_i} &= \left( \frac{\partial J}{\partial u} \frac{\partial u}{\partial \omega_i} + \frac{\partial J}{\partial \omega_i} \right) + \lambda^t \left( \frac{\partial F}{\partial u} \frac{\partial u}{\partial \omega_i} + \frac{\partial F}{\partial \omega_i} \right) \\ &= \left( \frac{\partial J}{\partial \omega_i} + \lambda^t \frac{\partial F}{\partial \omega_i} \right) + \left( \frac{\partial J}{\partial u} + \lambda^t \frac{\partial F}{\partial u} \right) \frac{\partial u}{\partial \omega_i} \end{aligned} \quad (1.19)$$

On choisit ensuite  $\lambda$  afin d'annuler le terme de droite :

$$\frac{\partial J}{\partial u} + \lambda^t \frac{\partial F}{\partial u} = 0 \quad (1.20)$$

Ce qui donne :

$$\lambda^t = - \frac{\partial J}{\partial u} \left( \frac{\partial F}{\partial u} \right)^{-1} \quad (1.21)$$

ou plus communément :

$$\left[ \frac{\partial F}{\partial u} \right]^t \lambda = - \left[ \frac{\partial J}{\partial u} \right]^t \quad (1.22)$$

En utilisant les équations (1.15) et (1.20), l'équation (1.17) s'écrit :

$$\frac{dJ}{d\omega_i} = \frac{\partial J}{\partial \omega_i} + \lambda^t \frac{\partial F}{\partial \omega_i} = \frac{d\mathcal{L}}{d\omega_i} \quad (1.23)$$

L'équation (1.22) est appelée équation adjointe à  $F(\omega_i, u(\omega_i))$  et le terme  $\lambda^t$  est appelé vecteur adjoint.

On remarque que l'approche adjointe ne nécessite que deux calculs quel que soit le nombre de paramètres. En effet, il est nécessaire de résoudre l'équation d'état et l'équation adjointe. La résolution de l'équation d'état est effectuée par un solveur primaire. L'équation adjointe est résolue par un solveur adjoint soit de façon discrète, soit de façon continue. Nous ne présentons pas les différences entre ces deux approches dans ce manuscrit mais le lecteur peut se reporter aux travaux de [Peter 2006, Anderson 1999].

Actuellement, des solveurs adjoints sont disponibles en mécanique du solide (Nastran<sup>®</sup> (MSC), Abaqus<sup>®</sup> (Dassault Systèmes)) et en mécanique des fluides (Fluent<sup>®</sup> (ANSYS), STAR-CCM+<sup>®</sup> (CD-adapco), Pamflow<sup>®</sup> (ESI), OpenFOAM<sup>®</sup> (ESI)...) et permettent d'évaluer la sensibilité de différentes fonctions objectif par rapport à la forme, notamment :

- en mécanique du solide :
  - une compliance ou une rigidité ;
  - une fréquence propre, une amplitude de vibration (critères plus connus dans l'industrie sous l'appellation anglaise « NVH » pour « noise, vibration and harshness ») ;
- en mécanique des fluides :
  - une perte de charge ;
  - l'uniformité d'un flux sur une section ;
  - une force...

Lorsque le gradient est disponible, les processus les plus répandus l'utilisent pour effectuer un morphing qui déforme la géométrie en faisant varier la position des nœuds [Staten 2011]. Pour limiter les déplacements trop importants entre deux nœuds adjacents, il est courant d'utiliser un opérateur de type laplacien. En outre, cela est problématique en mécanique des fluides notamment à cause des couches limites utilisées pour capter les turbulences au niveau des parois. En effet, au bout de quelques itérations, la couche limite est totalement déformée, ce qui peut fortement perturber les résultats.

Les figures 1.7.a à 1.7.d représentent de façon schématique la démarche appliquée le plus souvent lorsque les gradients sont disponibles. Dans un premier temps, un modèle CAO est paramétré. La figure 1.7.a est une représentation simplifiée d'une géométrie CAO avec trois paramètres : les coordonnées  $x_O$  et  $y_O$  du centre du cercle et son rayon  $R$ . Nous avons vu dans la partie 1.3.4 les caractéristiques du paramétrage CAO. Dans la partie 1.4, nous avons vu que la méthode de l'état adjoint permet de calculer le gradient de façon indépendante du nombre de paramètres. La disponibilité des solveurs adjoints en mécanique des fluides pour l'industrie automobile s'est présentée comme une opportunité intéressante. Toutefois, le gradient fourni par ces solveurs n'est notamment exploitable que par rapport à la position des points du maillage comme le montrent les figures 1.7.b et 1.7.c. Actuellement, la méthode la plus répandue dans l'industrie pour l'exploitation des sensibilités consiste à effectuer une déformation du maillage (cf. 1.7.d). Cela implique un retour CAO coûteux pour que la pièce soit utilisée dans la suite du processus industriel.

Afin de profiter de l'intelligence mise en place dans la CAO (contraintes de fabrication, contraintes de design) et de faciliter son intégration dans des processus répandus chez les industriels en évitant le retour CAO coûteux, nous cherchons dans nos travaux à optimiser la forme par rapport aux paramètres CAO. Dans ce cas, cela revient à obtenir les nouvelles valeurs  $x'_O$ ,  $y'_O$  et  $R'_O$  des paramètres du modèle CAO permettant d'approcher au mieux la forme proposée par la sensibilité (cf. 1.7.e).

Nous cherchons donc à optimiser la fonction objectif  $J$  par rapport à chaque paramètre de forme CAO noté  $\alpha_i$ , leur ensemble étant noté  $\alpha$ , cela s'écrit  $J = J(\alpha, \Omega(\alpha))$ . Nous posons donc :

$$\begin{aligned} d_\alpha J &= \partial_\alpha J + \partial_\Omega J \times \partial_\alpha \Omega \\ &= \partial_\Omega J \times \partial_\alpha \Omega \end{aligned} \tag{1.24}$$

car nous considérons qu'il n'y a pas de dépendance directe entre le critère et nos variables d'optimisation, d'où le fait que le terme  $\partial_\alpha J$  soit nul.

Les travaux présentés dans cette thèse se basent sur l'équation (1.24) pour effectuer les optimisations de forme. Comme il a été vu dans la partie 1.4, la sensibilité de la fonction objectif

## 1.4. Calcul du gradient d'une fonction objectif par rapport à la forme

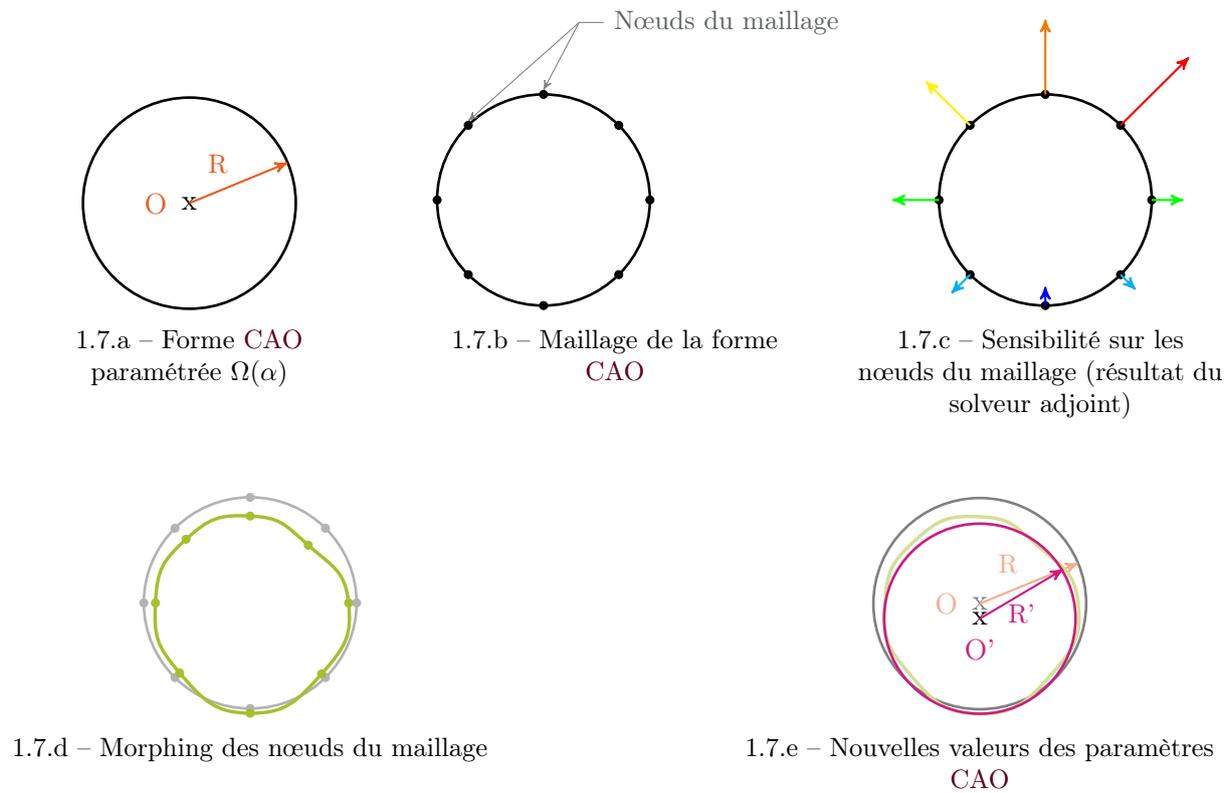


FIGURE 1.7 – Explication schématique de l'optimisation par gradient avec utilisation des paramètres CAO

par rapport à la forme, notée  $\partial_\Omega J$ , est donnée par le solveur adjoint. Il est donc nécessaire de déterminer la sensibilité de la forme par rapport aux paramètres CAO, notée  $\partial_\alpha \Omega$ , dans le but d'obtenir la sensibilité de la fonction objectif vis-à-vis des paramètres CAO, soit  $d_\alpha J$ . Actuellement, cette sensibilité est le maillon manquant dans les outils industriels pour envisager des optimisations par méthodes exactes sur les modèles CAO.

### 1.5 Calcul de sensibilité d'une forme par rapport aux paramètres CAO

Afin de profiter pleinement de l'information donnée par les solveurs adjoints tout en profitant des avantages considérables que représentent les modèles CAO, il est primordial de reporter sur les paramètres CAO les valeurs de sensibilités données par le solveur adjoint. Il existe dans la littérature des méthodes permettant d'évaluer la sensibilité de la forme par rapport aux paramètres CAO, appelée aussi « *design velocity* » en anglais. Leur présentation dans cette partie va nous permettre de peser le pour et le contre des techniques disponibles et de proposer le cas échéant une nouvelle solution.

Il ressort deux voies de calcul : soit de manière analytique, soit de façon numérique. De plus, les solveurs adjoints peuvent fournir les sensibilités de manière discrète ou continue, ce qui permet de s'adapter à la voie choisie pour le calcul de sensibilité du maillage par rapport aux paramètres CAO. À l'heure actuelle, le calcul de sensibilité effectué par les solveurs adjoints est donné au niveau des nœuds du maillage ou bien des barycentres (*centroïdes*) des éléments, selon la normale ou selon les composantes en  $\vec{x}$ ,  $\vec{y}$  et  $\vec{z}$ . Avec un calcul analytique, il est envisageable d'utiliser la version continue des solveurs adjoints et ainsi obtenir des résultats plus précis.

#### 1.5.1 Méthode analytique

L'objectif principal est de calculer de manière formelle la sensibilité de la forme par rapport aux paramètres CAO (cf. figure 1.8). L'avantage principal est que la sensibilité exacte par rapport à chaque paramètre est calculée. En effet, il n'y a pas besoin de discrétiser la forme et ainsi les erreurs numériques liées aux calculs par éléments finis sont évitées. Ce processus est complètement intégrable à un logiciel CAO et est transparent pour l'utilisateur du fait qu'il ne nécessite aucun travail pour son intégration dans une boucle automatisée de calcul.

Le calcul de  $\partial_{\alpha_j} \Omega$  peut donc être posé comme suit :

$$\partial_{\alpha_j} \Omega = \lim_{\delta \alpha_j \rightarrow 0} \frac{\Omega(\alpha_j + \delta \alpha_j) - \Omega(\alpha_j)}{\delta \alpha_j} \quad (1.25)$$

Cette solution nécessite l'utilisation des programmes de résolution des équations liant la géométrie aux paramètres  $\alpha_i$ . Elle est donc intrusive vis-à-vis du logiciel CAO. Ainsi, il est nécessaire

## 1.5. Calcul de sensibilité d'une forme par rapport aux paramètres CAO

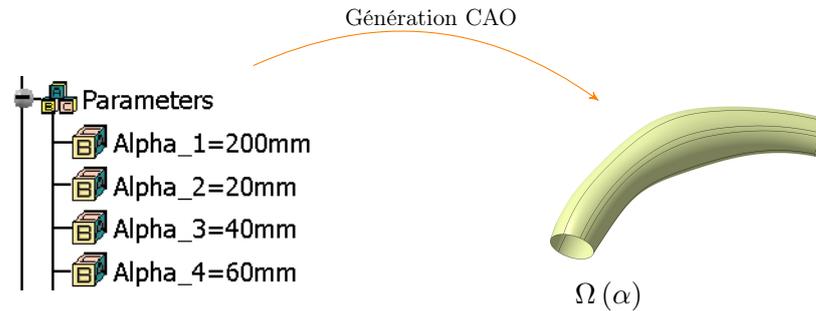


FIGURE 1.8 – Génération CAO

de connaître les équations qui caractérisent la forme et qui sont des fonctions des paramètres. Ces équations et leurs résolutions sont mises en place dans les solveurs géométriques/volumiques. La plupart de ces solveurs (DS CATIA<sup>®</sup>, Siemens NX<sup>®</sup>...) ne donnent pas accès au code source et sont donc des boîtes noires pour l'utilisateur. Le calcul analytique est donc impossible à l'heure actuelle sauf si nous utilisons des solveurs géométriques ouverts (Open CASCADE Technology<sup>®</sup>) ou bien une méthode analytique utilisant des fichiers au format neutre (IGES, STEP...).

[Dannenhoffer III 2015] utilise le logiciel OpenCSM<sup>®</sup> (basé sur le noyau Open CASCADE<sup>®</sup>) pour évaluer la sensibilité d'une géométrie sur les primitives élémentaires (faces, arêtes et nœuds). Pour connaître le lien entre les paramètres définis et les éléments géométriques, cette méthode impose de bien maîtriser le processus de construction du modèle CAO comme la génération des primitives mais aussi les opérations de congés, de chanfrein, etc. La complexité des formes est donc fortement limitée. [Xu 2014] effectue une différentiation en utilisant la *représentation par les bords* ou plus connue sous le nom de "B-Rep". Cette représentation est portée par des surfaces canoniques. Dans le cas de [Xu 2014], les patches utilisés sont exclusivement des NURBS. Les points de contrôle qui pilotent ces NURBS peuvent se déplacer dans toutes les directions et constituent les paramètres de la géométrie, à l'image de l'optimisation isogéométrique de la partie 1.3.3. Toutefois, il n'est pas évident de les utiliser comme paramètres "métier" et surtout de les interpréter vis-à-vis de contraintes de fabrication plus spécifiques. Le lien entre les paramètres définis par le concepteur et les points de contrôle semble difficile à déterminer analytiquement, l'absence de cette information ne permet pas d'envisager d'utiliser ce type de méthode pour effectuer le calcul de sensibilité de la forme par rapport aux paramètres CAO comme des longueurs, des angles...

### 1.5.2 Méthodes numériques sur les géométries CAO

Ces méthodes numériques consistent à utiliser les éléments géométriques des CAO pour comparer deux formes suite à la variation d'un paramètre et ce sans avoir recours à une étape de maillage qui introduit une approximation due à la discrétisation (cf. figure 1.9). Contrairement aux méthodes analytiques pour lesquelles il n'est pas évident de connaître le lien entre les paramètres "métier" et la définition des surfaces, ces méthodes comparent les formes résultant de la variation de chaque paramètre avec la forme initiale [Armstrong 2007]. L'étude des changements sur les éléments de définition entre les deux formes renseigne sur l'impact de la perturbation d'un

paramètre. Il est possible d'utiliser les différents éléments de la définition CAO comme les sommets des patches, leurs bords ou les patches eux-mêmes [Chen 1997]. Cela nécessite donc d'avoir accès à la définition de ces éléments, les formats neutres tels que les fichiers STEP ou IGES sont donc à privilégier lorsque le solveur géométrique ne permet pas de récupérer ces informations. Cela n'évite pas les changements de topologie de cette définition CAO comme le changement du nombre de patches ou encore leur identifiant [Kripac 1997] lors de la variation d'un paramètre. Ces changements rendent plus difficiles l'association patches à patches pour la comparaison de deux géométries entre elles. De plus, la modification d'un paramètre peut entraîner le changement des caractéristiques intrinsèques du patch en question avec le changement de degré ou du nombre de pôles de la surface. Nous reviendrons sur cet aspect dans le chapitre 3 avec une approche par ce type de méthode.

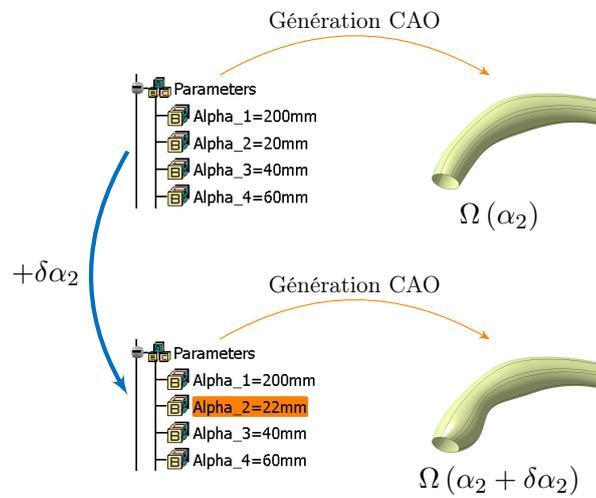


FIGURE 1.9 – Génération des géométries suite à la variation de chacun des paramètres

### 1.5.3 Méthodes numériques sur le maillage

La discrétisation de la forme permet de ne plus être confronté aux différentes définitions d'une même surface CAO suite à la simple variation d'un paramètre. On peut donc discrétiser l'équation (1.25) :

$$\partial_{\alpha_j} \Omega \simeq \frac{\tilde{\Omega}(\alpha_j + \delta\alpha_j) - \tilde{\Omega}(\alpha_j)}{\delta\alpha_j} \quad \forall j \in \llbracket 1, p \rrbracket \quad (1.26)$$

Pour calculer le terme  $\tilde{\Omega}(\alpha_j + \delta\alpha_j)$ , il est nécessaire d'avoir une isoconnectivité. C'est-à-dire une identification nœud à nœud explicite entre le maillage de la forme initiale et celui généré suite à la variation d'un paramètre (cf. figure 1.10).

Afin d'illustrer nos propos, nous nous appuyons sur la figure 1.10. La forme initiale  $\Omega(\alpha_j)$  – un disque dans cet exemple – est maillée en  $\tilde{\Omega}(\alpha_j)$ . En perturbant la forme initiale par une légère variation  $\delta\alpha_j$  d'un paramètre  $\alpha_j$ , nous obtenons la forme  $\Omega(\alpha_j + \delta\alpha_j)$ , représentée par l'ellipse

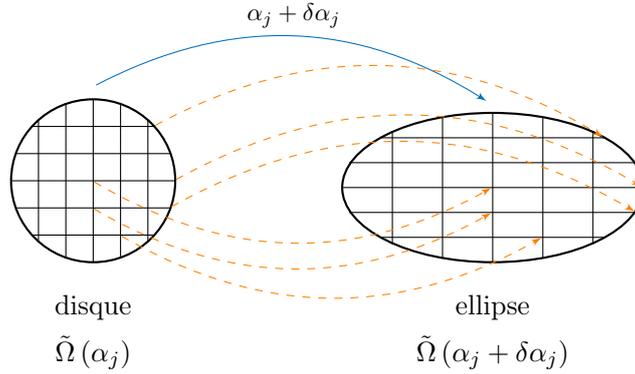


FIGURE 1.10 – Illustration du concept

non maillée. Nous souhaitons donc étudier l'influence de cette perturbation sur le maillage de la forme initiale  $\tilde{\Omega}(\alpha_j)$ , c'est-à-dire sur le maillage du disque. Cela revient à déterminer  $\tilde{\Omega}(\alpha_j + \delta\alpha_j)$  qui est représenté dans notre exemple par le maillage de l'ellipse. Pour cela, il est nécessaire de conserver l'isoconnectivité des maillages, ce qui est symbolisé par les flèches  $\rightarrow$  : le déplacement de chaque nœud du maillage initial est connu. Comme le déplacement de chaque nœud est connu, il est commode de calculer le gradient par différences finies :

$$\nabla_{\alpha_j} \vec{x} = \left. \frac{\partial x_i}{\partial \alpha_j} \right|_{\substack{i=1..n \\ j=1..p}} = \frac{x_i^{ellipse} - x_i^{disque}}{\delta\alpha_j} \quad (1.27)$$

Pour obtenir ce maillage de la forme perturbée, il est envisageable de déformer le maillage de la forme initiale [Toivanen 2006]. Mais cette opération est susceptible d'introduire des distorsions ce qui peut fausser le résultat de la sensibilité. Différentes méthodes sont proposées par [Choi 1994, Choi 2005, Lindby 1999]. Par exemple, le dernier auteur s'appuie sur de petites variations de paramètres et sur un lissage de type laplacien, il projette ainsi les nœuds sur la nouvelle CAO pendant l'étape de lissage.

La génération indépendante des maillages comme le montre la figure 1.11 semble être la plus facile à mettre en place. Malheureusement, lors de cette génération indépendante, la configuration du maillage comme le nombre de cellules ou la disposition des éléments n'est pas nécessairement identique entre les deux maillages.

Afin d'obtenir l'évaluation de la sensibilité avec deux maillages différents, des méthodes sont disponibles dans la littérature. Celles-ci sont présentées dans les paragraphes suivants.

### 1.5.3.1 Méthode de projection suivant la normale

Suite à la génération indépendante des maillages des deux formes, [Robinson 2012] propose une méthode pour évaluer le déplacement de chacun des points du maillage de la forme initiale en utilisant le maillage de la forme déformée. Le calcul de la sensibilité se déroule en deux grandes étapes :

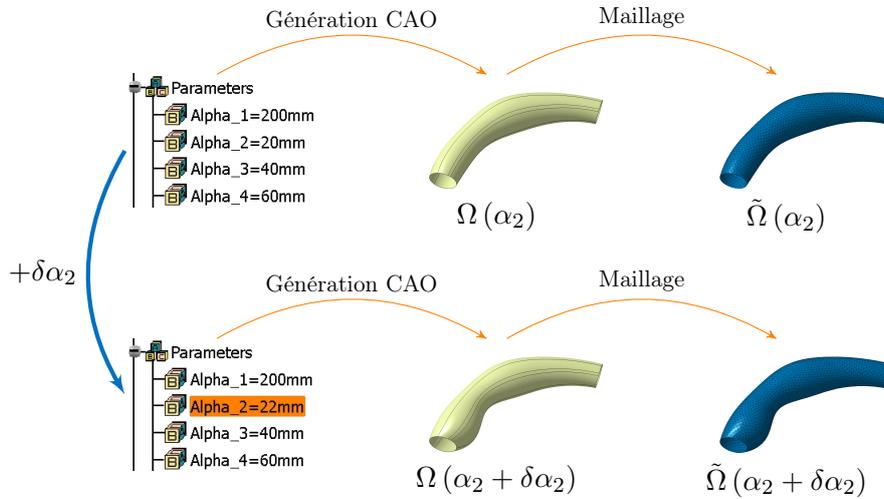


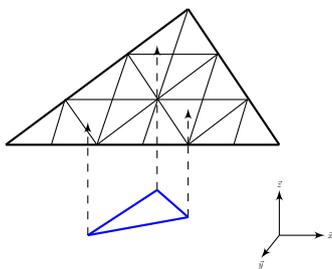
FIGURE 1.11 – Discrétisation de la forme

o **Projection du maillage initial sur un maillage plus fin**

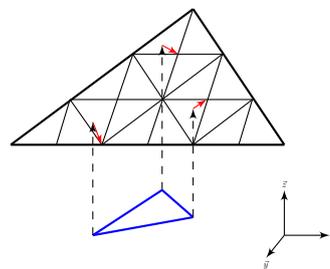
La forme non perturbée est maillée de manière grossière, alors que le maillage de la forme perturbée est quant à lui plus fin. La différence de raffinement permet une meilleure projection des nœuds du maillage grossier sur le maillage raffiné. Ainsi, les nœuds projetés seront répartis sur les éléments plus fins du maillage raffiné (cf. figure 1.12.a). La qualité de la projection dépend du rapport entre le nombre d'éléments du maillage perturbé et celui du maillage non perturbé. Ici, la projection d'un nœud du maillage initial s'effectue suivant la normale de l'élément auquel il appartient.

o **Évaluation du déplacement**

La projection de chaque nœud du maillage initial est ensuite associée au nœud le plus proche du maillage de la forme perturbée (cf. figure 1.12.b). Le déplacement de chaque point suite à la perturbation est donné par la différence entre les coordonnées du point initial et celles de son projeté associé. Des vérifications numériques permettent d'évaluer les erreurs qui peuvent apparaître lors de la projection suivant la normale [Vasilopoulos 2016].



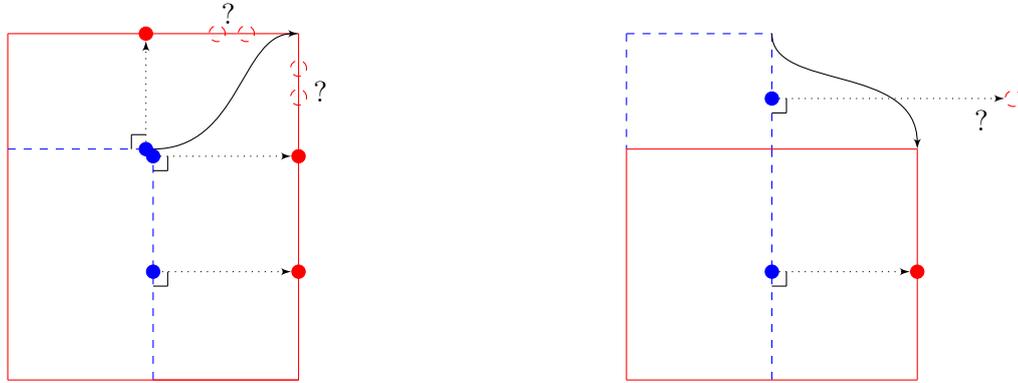
1.12.a – Projection suivant la normale



1.12.b – Recherche du point le plus proche

FIGURE 1.12 – Illustration de la méthode de projection suivant la normale

Toutefois, comme le montrent les schémas de la figure 1.13, illustrant la variation d'un paramètre permettant d'obtenir la forme en trait plein à partir de la forme en trait discontinu, la projection des nœuds d'un maillage n'est pas toujours évidente.



1.13.a – Projection non surjective

1.13.b – Projection non injective

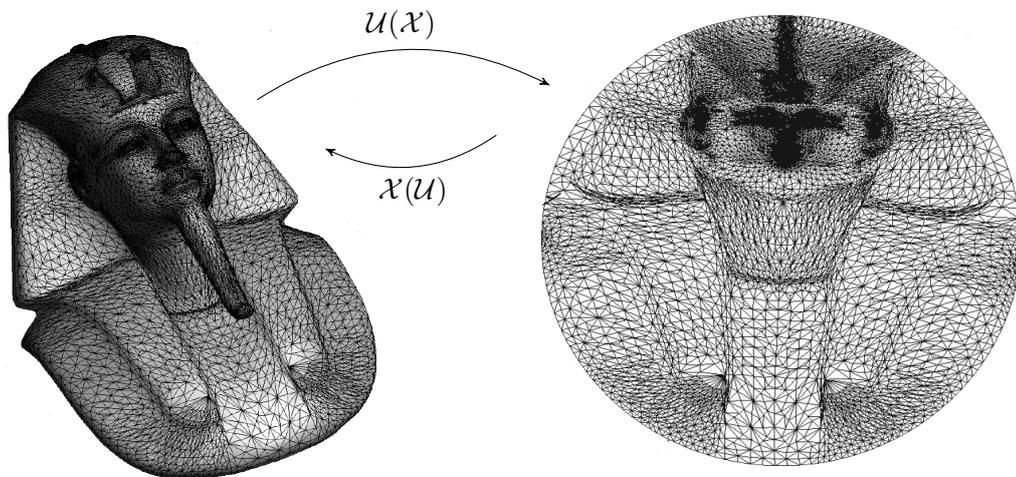
FIGURE 1.13 – Problèmes liés à la projection suivant la normale

Pour la configuration de la figure 1.13.a, si les points sont pris sur la forme en trait discontinu représentant des nœuds du maillage initial, il est aisé de les projeter sur le maillage de la forme perturbée, représenté par les points de la forme en trait plein, par contre il est impossible de déterminer l'origine des nœuds situés dans le coin supérieur gauche de la forme perturbée représentés par les cercles en pointillés rouges. De même, pour la configuration de la figure 1.13.b, les nœuds du maillage initial représentés par les points sur la forme en trait discontinu situés sur la partie supérieure ne sont pas projetables sur la forme en trait plein.

### 1.5.3.2 Méthode de projection harmonique appliquée aux formes monofrontières

Suite au constat effectué sur les méthodes de projections suivant la normale, [Froment 2014] propose une approche différente pour le calcul de cette sensibilité. Largement utilisées dans le domaine de la visualisation 3D [Gu 2008], les techniques de mapping permettent de projeter des formes 3D dans un domaine 2D comme le montre la figure 1.14. Il est donc possible de connaître les coordonnées dans un domaine 2D d'un nœud de la forme 3D. Cette méthode consiste à projeter deux formes d'une même géométrie CAO. La première forme, appelée forme initiale, est celle dont les valeurs de paramètres sont fixées à leur valeur nominale. La deuxième résulte de la variation d'un des paramètres. Suite à la projection de chacune des formes sur le disque, il est possible d'interpoler les nœuds projetés sur le disque unité. Cette interpolation permet de projeter les nœuds du maillage de la forme initiale sur la forme perturbée.

Pour effectuer cette projection, l'auteur s'est inspiré d'une fonctionnalité du logiciel Gmsh<sup>®</sup> [Geuzaine 2009]. La première version implémentée à l'aide des travaux de [Remacle 2010] limite la maîtrise de la fonction de projection et en particulier lors d'un nombre de frontières supérieur à un. Lorsqu'une géométrie ayant plus d'une frontière (par exemple un tube ou un collecteur) est projetée sur un disque, une frontière est projetée sur le bord du disque et les autres sont projetées sur le disque. L'emplacement des trous varie en fonction des valeurs de paramètres et il est impossible de comparer les disques entre eux. L'utilisation de la fonction ne permet pas



1.14.a – Masque de Toutânkhamon en 3D

1.14.b – Masque de Toutânkhamon déplié sur un disque

FIGURE 1.14 – Illustration du dépliage du masque de Toutânkhamon sur un disque  
 Source : [Remacle 2010]

de gérer la position des frontières. Pour pallier cela, un découpage de la géométrie est effectué pour avoir des surfaces homéomorphes à un disque, c'est-à-dire ayant une seule frontière (cf. figure 1.15). Toutefois, le découpage peut être difficile à mettre en place suivant la complexité de la forme. Cette opération supplémentaire peut vite s'avérer contraignante car elle est susceptible de diminuer le taux de reconstruction de la géométrie.

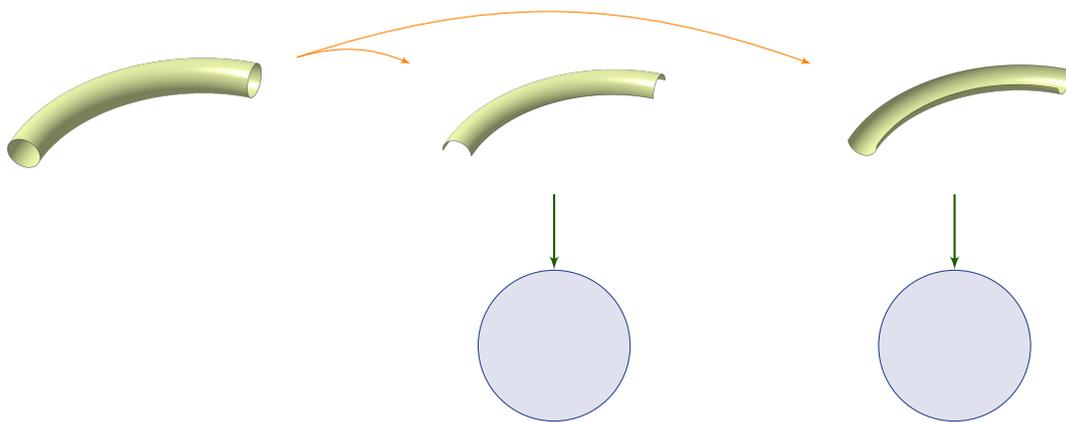


FIGURE 1.15 – Découpage de la géométrie en deux parties homéomorphes à un disque  
 Source : [Froment 2014]

Une autre question se pose sur la justesse des gradients lors de l'assemblage des sensibilités de chaque partie. Comment est calculée la valeur du gradient sur les coutures de la surface découpée ?

## 1.6 Synthèse

Dans cet état de l'art, nous avons parcouru les principaux outils disponibles à l'heure actuelle dans l'industrie pour effectuer des optimisations de forme. Les méthodes d'optimisation sont le plus souvent posées suivant la formulation inverse. Nous souhaitons obtenir un objectif et pour cela il est nécessaire de déterminer le paramétrage adéquat pour formuler au mieux la fonction objectif et les contraintes. Les méthodes à disposition du concepteur sont nombreuses du fait de la diversité des stratégies d'optimisation, du paramétrage utilisé et de la connaissance de la fonction à optimiser.

Dans notre cas, nous utilisons les paramètres géométriques que le concepteur ajoute aux modèles CAO pour traduire des contraintes de fabrication ou de design. En parallèle de cela, nous nous appuyons sur l'opportunité que représentent les solveurs adjoints récemment disponibles pour avoir l'information du gradient de la fonction objectif étudiée. Actuellement, peu de méthodes permettent de travailler sur les deux éléments précédents dans le même temps. La méthode proposée par [Froment 2014] nous apparaît comme la plus adaptée pour effectuer ce type d'optimisation. Toutefois, les limites qu'elle présente peuvent nuire à son utilisation dans les processus répandus chez les industriels. Ces limites sont le découpage de la forme pour obtenir des surfaces homéomorphes à un disque mais aussi l'utilisation de maillages pour effectuer le calcul de sensibilité.

Nous proposons dans la suite de ce manuscrit des améliorations significatives permettant de réduire les difficultés liées à son industrialisation et à sa transparence vis-à-vis de l'utilisateur. Tout d'abord, dans le chapitre 2, nous proposons des développements qui permettent de gérer la projection de surfaces non-homéomorphes à un disque. Ensuite, des changements du processus d'interpolation se traduisent par l'amélioration du calcul de sensibilité basé sur les maillages. Dans le chapitre 3, une approche différente, basée sur la définition des surfaces CAO, représente une alternative à l'utilisation de maillages. Finalement, dans le chapitre 4, deux cas-test confirment les avantages d'un processus d'optimisation paramétrique avec l'utilisation du gradient donné par un solveur adjoint.

## Bibliographie du 1<sup>er</sup> chapitre

- [Adelman 1986] Howard M. Adelman et Raphael T. Haftka. *Sensitivity Analysis of Discrete Structural Systems*. AIAA Journal, vol. 24, no. 5, pages 823–832, May 1986. (cité page 18)
- [Albertelli 2015] Marc Albertelli, Thomas Abballe, Grégoire Allaire, Axelle Caron, Philippe Conraux, Luca Dall’Olio, Cécile Dobrzynski Charles Dapogny, Benoit Jeannin, Damien Lachouette François Jouve, Thierry Le Sommer, Kevin Maquin, Georgios Michailidis, Maximilien Siavelis et Vassili Srithammavanh. *RODIN project, Topology Optimization 2.0 ?* SIA, Congrès Simulation, 2015. (cité page 11)
- [Allaire 2004] Grégoire Allaire, François Jouve et Anca-Maria Toader. *Structural Optimization Using Sensitivity Analysis and a Level-set Method*. J. Comput. Phys., vol. 194, no. 1, pages 363–393, February 2004. (cité page 11)
- [Allaire 2007] Grégoire Allaire. *Conception optimale de structures*, volume 58. Springer Berlin Heidelberg New York, 2007. (cité pages 11 et 14)
- [Anderson 1999] W.K. Anderson et V. Venkatakrishnan. *Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation*. Computers and Fluids, vol. 28, page 443–480, 1999. (cité page 19)
- [Armstrong 2007] C.G. Armstrong, T.T. Robinson, H Ou et C. Othmer. *Linking adjoint sensitivity maps with CAD parameters*. Evolutionary and Deterministic Methods for Design, Optimization and Control, vol. Eurogen 2007, pages 234–239, 2007. (cité page 23)
- [Bendsøe 2003] Martin P. Bendsøe et Ole Sigmund. *Topology optimization : Theory, methods and applications*. Berlin Heidelberg, 2003. (cité page 11)
- [Benoist 1994] D. Benoist, Y. Tourbier et S. Germain-Tourbier. *Plans d’expériences : construction et analyse*. Technique & Documentation-Lavoisier, 1994. (cité pages 8, 9 et 116)
- [Binois 2015] Mickaël Binois. *Uncertainty quantification on pareto fronts and high-dimensional strategies in bayesian optimization, with applications in multi-objective automotive design*. Theses, Ecole Nationale Supérieure des Mines de Saint-Etienne, December 2015. (cité page 8)
- [Boole 1860] G. Boole. *A treatise on the calculus of finite differences*. Macmillan, 1860. (cité page 18)
- [Borrvall 2003] Thomas Borrvall et Joakim Petersson. *Topology optimization of fluids in Stokes flow*. International Journal for Numerical Methods in Fluids, vol. 41, no. 1, pages 77–107, 2003. (cité page 12)
- [Cacuci 1981] Dan G. Cacuci. *Sensitivity theory for nonlinear systems. I. Nonlinear functional analysis approach*. Journal of Mathematical Physics, vol. 22, no. 12, pages 2794–2802, 1981. (cité page 19)
- [Céa 1986] Jean Céa. *Conception optimale ou identification de formes, calcul rapide de la dérivée directionnelle de la fonction coût*. Modélisation Mathématique et Analyse Numérique, vol. 20, no. 3, pages 371–402, 1986. (cité page 19)
- [Chen 1997] S. Chen et D.A. Tortorelli. *Three-dimensional shape optimization with variational geometry*. Structural optimization, vol. 13, no. 2-3, pages 81–94, 1997. (cité page 24)

- [Choi 1994] Kyung K. Choi et Kuang-Hua Chang. *A study of design velocity field computation for shape optimal design*. Finite Elements in Analysis and Design, vol. 15, no. 4, pages 317 – 341, 1994. (cité page 25)
- [Choi 2005] Kyung K. Choi et Nam-Ho Kim. Structural sensitivity analysis and optimization 1. Springer, 2005. (cité page 25)
- [Culioli 1994] J.C. Culioli. Introduction à l’optimisation. Ellipses, 1994. (cité pages 7 et 99)
- [Dannenhoffer III 2015] J.F. Dannenhoffer III et R. Haimes. *Design Sensitivity Calculations Directly on CAD-based Geometry*. In 53rd AIAA Aerospace Sciences Meeting, page 1370, 2015. (cité pages 23, 65 et 66)
- [Dapogny 2013] C. Dapogny. *Shape optimization, level set methods on unstructured meshes and mesh evolution*. PhD thesis, UPMC, 2013. (cité pages 11 et 13)
- [Darcy 1857] Henry Darcy. Recherches expérimentales relatives au mouvement de l’eau dans les tuyaux, volume 1. Mallet-Bachelier, 1857. (cité page 12)
- [du Cauzé de Nazelle 2013] P. du Cauzé de Nazelle. *Paramétrage de formes surfaciques pour l’optimisation*. PhD thesis, École Centrale de Lyon, 2013. (cité page 14)
- [Froment 2014] Pierre Froment. *Optimisation de formes paramétriques en grande dimension*. PhD thesis, École Centrale de Lyon, 2014. (cité pages 10, 12, 15, 27, 28, 29, 34, 45, 47 et 97)
- [Geuzaine 2009] C. Geuzaine et J.F. Remacle. *Gmsh : a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities*. International Journal for Numerical Methods in Engineering, vol. 79, no. 11, pages 1309–1331, 2009. (cité page 27)
- [Goupy 2013] Jacques Goupy. Introduction aux plans d’expériences. Dunod/L’usine nouvelle, 5th édition, 2013. (cité page 9)
- [Gu 2008] X.D. Gu et S.T. Yau. Computational conformal geometry, volume 3 of *Advanced lectures in mathematics*. International Press, 2008. (cité pages 27, 34, 54 et 133)
- [Hughes 2005] T.J.R. Hughes, J.A. Cottrell et Y. Bazilevs. *Isogeometric analysis : CAD, finite elements, NURBS, exact geometry and mesh refinement*. Computer Methods in Applied Mechanics and Engineering, vol. 194, no. 39-41, pages 4135–4195, 2005. (cité page 13)
- [Julisson 2016] Sarah Julisson. *Optimisation de formes de coques minces pour des géométries complexes*. PhD thesis, UVSQ, Décembre 2016. (cité page 14)
- [Keller 1976] J. B. Keller. *Inverse Problems*. The American Mathematical Monthly, vol. 83, no. 2, pages 107–118, 1976. (cité page 4)
- [Kelley 1999] Carl T Kelley. Iterative methods for optimization, volume 18. Siam, 1999. (cité page 7)
- [Kern 2016] Michel Kern. Méthodes numériques pour les problèmes inverses. Collection Mathématiques et statistiques. ISTE Éditions, Mars 2016. (cité pages 5 et 17)
- [Kripac 1997] Jiri Kripac. *A mechanism for persistently naming topological entities in history-based parametric solid models*. Computer-Aided Design, vol. 29, no. 2, pages 113 – 122, 1997. (cité pages 24, 34 et 76)
- [Lindby 1999] T. Lindby et J.L.T. Santos. *Shape optimization of three-dimensional shell structures with the shape parametrization of a CAD system*. Structural optimization, vol. 18, no. 2-3, pages 126–133, 1999. (cité page 25)

- [Michailidis 2014] Georgios Michailidis. *Manufacturing Constraints and Multi-Phase Shape and Topology Optimization via a Level-Set Method*. Theses, Ecole Polytechnique X, January 2014. (cité pages 11 et 12)
- [Moos 2004] O. Moos, F. Klimetzek et R. Rossmann. *Bionic optimization of air-guiding systems*. SAE Technical Paper, 2004. (cité page 12)
- [Osher 1988] Stanley Osher et James A. Sethian. *Fronts Propagating with Curvature Dependent Speed : Algorithms Based on Hamilton-Jacobi Formulations*. Journal of Computational Physics, vol. 79, no. 1, pages 12–49, 1988. (cité page 11)
- [Othmer 2005] C. Othmer et Th. Grahls. *Approaches to fluid dynamic optimization in the car development process*. EUROGEN, 2005. (cité page 17)
- [Peter 2006] J. Peter, S. Burguburu et M. Marcelet. Introduction à l’optimisation de forme en aérodynamique et quelques exemples d’application. ONERA, 2006. (cité page 19)
- [Remacle 2010] J.-F. Remacle, C. Geuzaine, G. Compère et E. Marchandise. *High-quality surface remeshing using harmonic maps*. International journal for numerical methods in engineering, vol. 83, pages 403 – 425, 2010. (cité pages 27, 28 et 34)
- [Robinson 2012] Trevor T. Robinson, Cecil G. Armstrong, Hung Soon Chua, Carsten Othmer et Thorsten Grahls. *Optimizing Parameterized CAD Geometries Using Sensitivities Based on Adjoint Functions*. Computer-Aided Design and Applications, vol. 9, no. 3, pages 253–268, 2012. (cité page 25)
- [Saitou 2005] Kazuhiro Saitou, Kazuhiro Izui, Shinji Nishiwaki et Panos Papalambros. *A Survey of Structural Optimization in Mechanical Product Development*. Journal of Computing and Information Science in Engineering, vol. 5, no. 3, pages 214–226, 2005. (cité page 11)
- [Staten 2011] M.L. Staten, S.J. Owen, S.M. Shontz, A.G. Salinger et T.S. Coffey. *A comparison of mesh morphing methods for 3D shape optimization*. Proceedings of the 20<sup>th</sup> international meshing round-table, pages 293–311, 2011. (cité page 20)
- [Talbi 2009] El-Ghazali Talbi. *Metaheuristics : from design to implementation*, volume 74. John Wiley & Sons, 2009. (cité page 8)
- [Toivanen 2006] Jukka I. Toivanen et Janne Martikainen. *A new method for creating sparse design velocity fields*. Computer Methods in Applied Mechanics and Engineering, vol. 196, pages 528 – 537, 2006. (cité page 25)
- [Vasilopoulos 2016] I Vasilopoulos, D Agarwal, M Meyer, TT Robinson et CG Armstrong. *Linking Parametric CAD with Adjoint Surface Sensitivities*. 2016. (cité page 26)
- [Xu 2014] Shenren Xu, Wolfram Jahn et Jens-Dominik Müller. *CAD-based shape optimisation with CFD using a discrete adjoint*. International Journal for Numerical Methods in Fluids, vol. 74, no. 3, pages 153–168, 2014. (cité page 23)
- [Yang 1992] R.J. Yang, A. Lee et D.T. McGeen. *Application of basis function concept to practical shape optimization problems*. Structural optimization, vol. 5, pages 55–63, 1992. (cité page 15)

# Calcul de la sensibilité d'une forme sur les maillages

---

<b>2.1</b>	<b>Utilisation de la projection harmonique . . . . .</b>	<b>34</b>
2.1.1	Détection de la frontière . . . . .	35
2.1.2	Conditions aux limites de Dirichlet . . . . .	37
2.1.3	Développement de la projection harmonique . . . . .	37
2.1.4	Reconstruction de l'isoconnectivité . . . . .	46
<b>2.2</b>	<b>Adaptation de la projection harmonique appliquée aux géométries multifrontières . . . . .</b>	<b>47</b>
2.2.1	Déplacement des frontières intérieures . . . . .	49
2.2.2	Utilisation de la double projection harmonique . . . . .	54
2.2.3	Utilisation de la triangulation de Delaunay et des coefficients barycen- triques . . . . .	56
2.2.4	Identification des nœuds particuliers . . . . .	60
<b>2.3</b>	<b>Amélioration de l'interpolation . . . . .</b>	<b>61</b>
2.3.1	Interpolation par connectivité . . . . .	61
2.3.2	Interpolation aux frontières . . . . .	62
<b>2.4</b>	<b>Synthèse et discussion des résultats . . . . .</b>	<b>65</b>
	<b>Bibliographie du 2<sup>e</sup> chapitre . . . . .</b>	<b>67</b>

---

Comme mentionné dans l'état de l'art, l'utilisation des maillages pour calculer la sensibilité d'une forme CAO surfacique est à présent une méthode qui a montré son utilité et sa faisabilité. Le fait de travailler sur des maillages évite principalement les problématiques liées à la gestion des patches CAO [Kripac 1997]. La méthode développée par [Froment 2014] a un fort potentiel et la démonstration de sa compatibilité avec des boucles d'optimisation industrielles a été présentée. Toutefois, nous remarquons des limites importantes qui peuvent rendre son utilisation compliquée dans le cadre de développement de pièces. Le fait de devoir découper la pièce (cf. partie 1.5.3.2) pour effectuer le calcul de sensibilité peut entraîner des erreurs importantes dans la génération de la CAO. Il faut aussi noter que la mise en place du plan de découpe peut se révéler contraignante pour l'utilisateur.

Nous proposons donc d'utiliser les travaux mis en place par [Froment 2014] dans sa thèse et d'améliorer le calcul de sensibilité pour permettre son application à des surfaces multifrontières et ainsi rendre son intégration plus aisée chez les industriels.

## 2.1 Utilisation de la projection harmonique

La méthode présentée par [Froment 2014] consiste à mettre en correspondance une surface 3D avec un domaine 2D (cf. figure 1.14). Pour cela, il s'appuie sur une paramétrisation qui à chaque point  $(x,y,z)$  d'une surface 3D associe une coordonnée  $(u,v)$  dans le domaine 2D. [Gu 2008, Floater 2002] présentent différentes méthodes de paramétrisation dont les premiers théorèmes remontent à l'antiquité et les plus récents permettent de respecter différentes contraintes (comme les aires, les angles) nécessaires dans différents domaines tels que la visualisation, la projection de texture en graphisme [Lévy 2000] ou encore le remaillage [Remacle 2010]. Parmi ces méthodes, la projection harmonique correspond à nos besoins puisqu'elle garantit la bijectivité entre les régions convexes et sa résolution est assez simple [Floater 2002]. En effet, cette méthode correspond à un système d'équations aux dérivées partielles linéaires et elliptiques qui peut être résolu par éléments finis. Afin d'implémenter cette méthode, nous présentons les différentes étapes pour la résolution de la projection harmonique.

Dans [Gu 2008, p. 140-141], l'auteur présente la projection harmonique comme une minimisation de l'énergie harmonique avec deux problèmes de Laplace indépendant. Nous pouvons donc écrire la définition du problème de la projection harmonique donnée par le système (2.1) appliquée à une surface  $\mathcal{S}$  ayant des frontières  $\partial\mathcal{S}$ .

$$\begin{cases} \nabla^2 u = 0, & \nabla^2 v = 0 & \text{sur} & \mathcal{S}, \\ u = \bar{u}(\vec{x}), & v = \bar{v}(\vec{x}) & \text{sur} & \partial\mathcal{S}_1. \\ \frac{\partial u}{\partial n} = 0, & \frac{\partial v}{\partial n} = 0 & \text{sur} & \partial\mathcal{S}/\partial\mathcal{S}_1. \end{cases} \quad (2.1)$$

Les termes  $u$  et  $v$  sont les coordonnées dans l'espace de projection. Le système (2.1) fait appel à une condition de Dirichlet et une condition de Neumann sur chacune des coordonnées  $u$  et  $v$ . L'illustration de la figure 2.1 montre la projection d'un cube (avec la face inférieure ouverte) sur un disque.

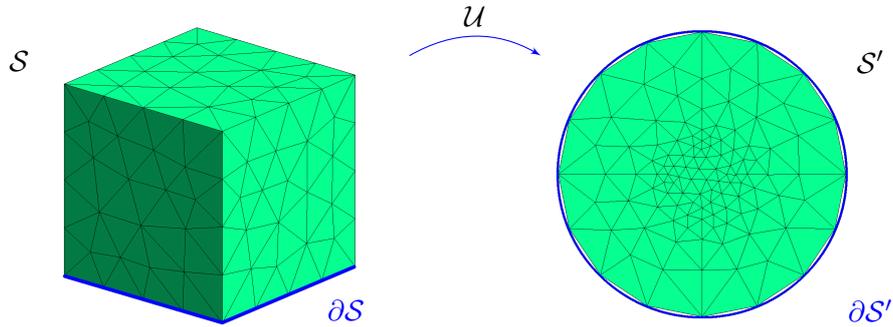


FIGURE 2.1 – Illustration de la projection harmonique appliquée à une géométrie monofrontière

La projection harmonique s’appuie sur la frontière de la surface pour projeter les points à l’intérieur du disque. Il est donc important de fixer la frontière sur le bord du disque pour appliquer la condition de Dirichlet. Pour cela, les frontières des géométries et plus particulièrement les segments qui les composent sont identifiés.

### 2.1.1 Détection de la frontière

Dans cette partie, nous travaillons sur des formes discrétisées. Les frontières sont donc déterminées à partir de la matrice des éléments. Nous illustrons la méthode présentée ici dans l’annexe A.

Soit  $m$  le nombre de nœuds du maillage. Nous construisons une matrice  $A \in \mathcal{M}_m(\mathbb{N})$  nulle. Nous lisons ensuite la matrice de connectivité en ajoutant 1 au coefficient  $a_{i,j}$  de la matrice où  $i$  est le numéro du premier nœud de l’arête et  $j$  celui du deuxième. Au final, dans la matrice obtenue, un segment  $[ij]$  est comptabilisé une seule fois par le coefficient  $a_{i,j}$ . Son segment symétrique  $[ji]$  n’est donc pas pris en compte. Il suffit pour remédier à cela d’ajouter la matrice transposée :  $A = A + A^t$  (cf. figure A.2).

Trois cas se distinguent alors :

- $a_{i,j} = 0$  : l’arête  $[ij]$  n’existe pas ;
- $a_{i,j} = 1$  : l’arête  $[ij]$  est sur le bord du maillage ;
- $a_{i,j} > 1$  : l’arête  $[ij]$  est à l’intérieur du maillage.

À ce stade, on récupère donc aisément les arêtes situées sur le bord du maillage en lisant consécutivement chaque ligne de la matrice  $A$ . Néanmoins, chaque segment apparaît deux fois ( $[ij]$  et  $[ji]$ ) dans la liste obtenue. Il est maintenant nécessaire de les trier afin de reconstruire toutes les arêtes d’une frontière dans l’ordre. Pour cela, le premier segment  $[ij]$  de la liste est sélectionné et son symétrique  $[ji]$  est éliminé tout en cherchant l’autre segment commençant par  $j$  :  $[jk]$ . On itère jusqu’à revenir à un segment dont l’extrémité est le nœud  $i$  (voir la figure A.3). La procédure est répétée tant que des éléments ne sont pas attribués à une frontière. L’algorithme 2.1 présente les étapes de reconstruction des frontières à partir d’un maillage.

**Données :** maillage triangulaire  $\tilde{\Omega}$  avec  $m$  nœuds et  $n$  éléments

**Début de l'algorithme**

```

    création d'une matrice  $A$  de dimension  $m \times m$ 
    initialisation des termes de la matrice  $A : a_{ij} = 0$ 
    pour chaque élément de la table de connectivité, faire
    |   pour chaque côté  $[i,j]$  de l'élément courant, faire
    |   |    $a_{ij} = a_{ij} + 1$ 
    |   fin
    fin
    ajout de la transposée  $A = A + A^t$ 
    pour chaque terme de la matrice  $A$ , faire
    |    $i_A = 1$ 
    |   si  $a_{i,j} = 1$  alors
    |   |    $F(i_A,1) = i$  et  $F(i_A,2) = j$ 
    |   |    $i_A = i_A + 1$ 
    |   fin
    fin
     $k = 1$ 
    tant que  $F \neq [ ]$ , faire
    |    $i = F(1,1)$  et  $j = F(1,2)$ 
    |    $i_m = 1$ 
    |    $F_k(i_m,1) = i$  et  $F_k(i_m,2) = j$ 
    |   suppression dans  $F$  de la ligne où se trouve  $[j,i]$ 
    |   tant que  $j \neq F(1,1)$ , faire
    |   |    $i_F \leftarrow$  trouver  $j$  dans  $F(:,1)$ 
    |   |    $i = F(i_F,1)$  et  $j = F(i_F,2)$ 
    |   |    $i_m = i_m + 1$ 
    |   |    $F_k(i_m,1) = i$  et  $F_k(i_m,2) = j$ 
    |   |   suppression dans  $F$  des lignes où se trouvent  $[i,j]$  et  $[j,i]$ 
    |   fin
    |   suppression de  $F(1, :)$ 
    |    $k = k + 1$ 
    fin

```

**Fin de l'algorithme**

**Résultat :** frontières  $F_1, F_2 \dots$

**Algorithme 2.1 :** Algorithme de détection des frontières

### 2.1.2 Conditions aux limites de Dirichlet

La partie précédente nous a permis de détecter les frontières et l'ensemble des nœuds les composant ainsi que l'ordre dans lequel ils apparaissent. Une condition de Dirichlet est imposée sur une frontière. Celle-ci est considérée comme frontière principale et elle est projetée sur le bord du disque. Dans un premier temps, la longueur totale  $L$  de cette frontière est déterminée par :

$$L = \sum_{i,j} |x_i - x_j| \quad (2.2)$$

Un point initial  $V_s$  est ensuite pris comme point de repère pour la projection comme on peut le voir sur la figure 2.2.

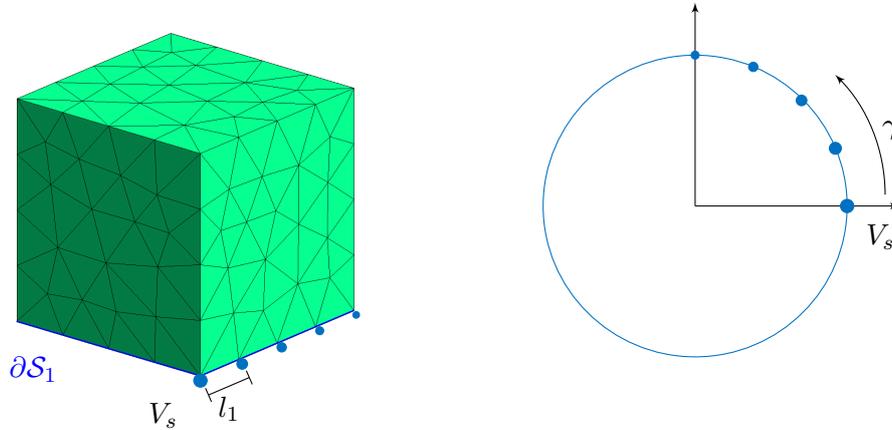


FIGURE 2.2 – Projection de la frontière principale sur le bord du disque

En partant de ce point et en parcourant chacun des segments dans le sens défini par  $\gamma_s$ , les coordonnées des points sur le bords du disque sont données par :

$$\begin{cases} u(\vec{x}_i) = \cos \frac{2\pi l_i}{L} \\ v(\vec{x}_i) = \sin \frac{2\pi l_i}{L} \end{cases} \quad (2.3)$$

avec  $L$  la longueur totale et  $l_i$  la distance suivant la frontière, séparant le point courant  $V_i$  du point initial  $V_s$ .

Ainsi, la frontière 3D est projetée sur la frontière du disque comme le montre la figure 2.2 et il est possible de projeter l'ensemble des points de la surface.

### 2.1.3 Développement de la projection harmonique

Dans cette partie, on considère la surface  $\mathcal{S} \subset \mathbb{R}^3$ . Le disque unité constitue le domaine paramétrique  $\mathcal{S}' \subset \mathbb{R}^2$  ayant la même topologie que  $\mathcal{S}$ . L'équation (2.1) est le problème à résoudre pour déterminer les coordonnées  $(u,v)$ . Sachant que ce problème est découpé, on ne s'intéresse qu'au terme  $u$ , la démarche étant identique pour le terme  $v$ .

On cherche donc  $u \in H^1(\mathcal{S})$  tel que<sup>a</sup> :

$$\begin{cases} \Delta u(x,y,z) = 0 & \text{sur } \mathcal{S} \\ u(x,y,z) = u_0(x,y,z) & \text{sur } \partial\mathcal{S}_1 \\ \frac{\partial u}{\partial n}(x,y,z) = 0 & \text{sur } \partial\mathcal{S} \setminus \partial\mathcal{S}_1 \end{cases} \quad (2.4)$$

où  $u_0 \in H^{1/2}(\partial\mathcal{S}_1)$  une fonction donnée. Par commodité, on note  $\partial\mathcal{S}_2 = \partial\mathcal{S} \setminus \partial\mathcal{S}_1$  dans la suite du développement.

On considère  $\mathcal{V}_1$  l'ensemble des fonctions  $w = u - \bar{u}$  où  $\bar{u} \in H^1(\mathcal{S})$  telles que  $\bar{u} = u_0$  sur  $\partial\mathcal{S}_1$ . En d'autres termes,  $w = 0$  sur  $\partial\mathcal{S}_1$ .

$\forall w \in \mathcal{V}_1$ , on a :

$$\int_{\mathcal{S}} \Delta u w d\mathcal{S} = 0 \quad (2.5)$$

On utilise la formule de Green-Ostrogradski qui généralise la notion d'intégration par parties à des dimensions supérieures. L'équation (2.5) devient :

$$\int_{\partial\mathcal{S}} w \nabla u \cdot n dl - \int_{\mathcal{S}} \nabla u \nabla w d\mathcal{S} = 0 \quad (2.6)$$

où  $* \cdot *$  désigne le produit scalaire canonique de deux vecteurs de  $\mathbb{R}^n$ .

En distinguant les frontières  $\partial\mathcal{S}_1$  et  $\partial\mathcal{S}_2$ , l'équation (2.6) devient :

$$\int_{\mathcal{S}} \nabla u \nabla w d\mathcal{S} = \int_{\partial\mathcal{S}_1} w \nabla u \cdot n dl + \int_{\partial\mathcal{S}_2} w \nabla u \cdot n dl \quad (2.7)$$

Or  $w = 0$  sur  $\partial\mathcal{S}_1$  et  $\nabla u \cdot n = \partial_n u = 0$  sur  $\partial\mathcal{S}_2$ , on a donc :

$$\int_{\mathcal{S}} \nabla u \nabla w d\mathcal{S} = 0 \quad \forall w \in \mathcal{V}_1 \quad (2.8)$$

L'expression (2.8) est symétrique en  $u$  et  $w$ , on peut donc demander les mêmes conditions sur ces fonctions. On cherche donc  $u \in \mathcal{V}_1$  tel que l'équation (2.8) soit vérifiée.

Par analogie, on fait de même pour la coordonnée  $v$  en considérant  $\mathcal{V}_2$ , l'ensemble des fonctions  $w = v - \bar{v}$  telles que  $\bar{v} = v_0$  sur  $\partial\mathcal{S}_1$ .

In fine, le problème (2.1) revient à chercher  $u \in \mathcal{V}_1$  et  $v \in \mathcal{V}_2$  tels que :

$$\begin{cases} \int_{\mathcal{S}} \nabla u \nabla w d\mathcal{S} = 0 & \forall w \in \mathcal{V}_1 \\ \int_{\mathcal{S}} \nabla v \nabla w d\mathcal{S} = 0 & \forall w \in \mathcal{V}_2 \end{cases} \quad (2.9)$$

À ce stade, nous cherchons à discrétiser la formulation variationnelle de notre problème en se ramenant au domaine de référence. Cette résolution s'effectue de manière analogue pour  $u$  et pour  $v$ . Par la suite, nous continuons les développements uniquement pour les coordonnées  $u$ .

---

a.  $H^1(\mathcal{S})$  est l'espace de Sobolev.

## 2.1. Utilisation de la projection harmonique

Nous considérons que la surface  $\mathcal{S}$  est munie d'une triangulation  $\mathcal{S}_{\mathcal{T}}$  ( $\mathcal{T} = \{\mathcal{T}_1, \dots, \mathcal{T}_N\}$ ). Les triangles  $\mathcal{T}_j$  ont donc une représentation dans  $\mathbb{R}^3$  et dans  $\mathbb{R}^2$  : chaque sommet  $A_i|_{i=\llbracket 1, N_{\text{noeud}} \rrbracket}$  du maillage  $\mathcal{S}_{\mathcal{T}}$  a donc des coordonnées dans l'espace  $A_i = (x_i, y_i, z_i) \in \mathcal{S}$  et des coordonnées paramétriques  $B_i = (u_i, v_i) \in \mathcal{S}'$ .

Isolons un triangle  $\mathcal{T}_j$  dont les sommets sont notés  $A_1, A_2$  et  $A_3$  dans  $\mathbb{R}^3$  avec  $A$  un point de ce triangle. De manière analogue, on note  $B_1, B_2$  et  $B_3$  les sommets dans  $\mathbb{R}^2$  et  $B$  le point correspondant au point  $A$  dans ce triangle. La figure 2.3 reprend les notations qui sont utilisées par la suite :  $\mathcal{M}(\xi, \eta)$ ,  $\mathcal{M}^*(x, y, z)$  et  $m(\xi, \eta)$ . Le détail des calculs des matrices jacobiniennes des fonctions  $\varphi(u, v)$  et  $m^{-1}(u, v)$  est donné dans l'annexe B.

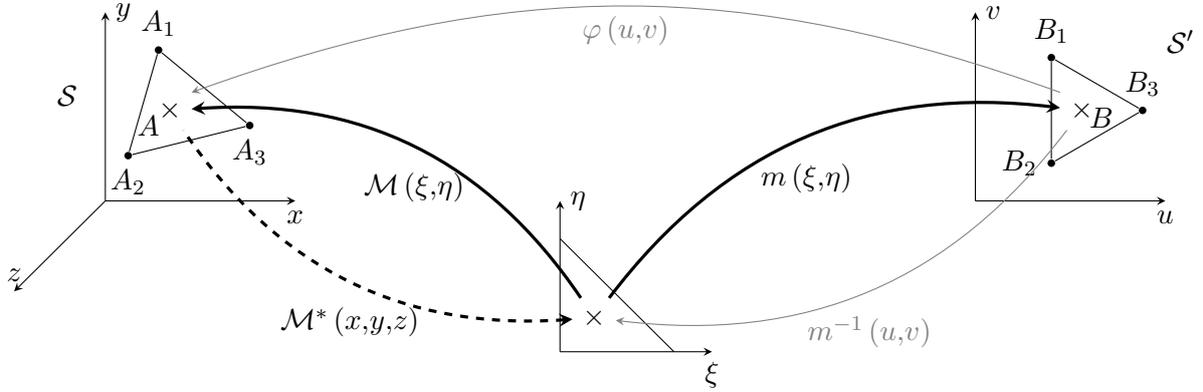


FIGURE 2.3 – Triangle unité, fonctions de projection et coordonnées locales

On pose  $\mathcal{M}(\xi, \eta)$  une fonction classique d'interpolation linéaire qui donne les coordonnées courantes d'un point  $A(x, y, z)$  du triangle  $\mathcal{T}_j$  dans  $\mathbb{R}^3$  :

$$\begin{aligned} \mathcal{M} : [0,1] \times [0,1] &\rightarrow \mathbb{R}^3 \\ (\xi, \eta) &\mapsto \mathcal{M}(\xi, \eta) = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \end{aligned}$$

$$\mathcal{M}(\xi, \eta) = (1 - \xi - \eta) A_1 + \xi A_2 + \eta A_3 \quad (2.10)$$

On retrouve bien les relations suivantes :

$$\left| \begin{array}{l} \mathcal{M}(0,0) = A_1 \\ \mathcal{M}(1,0) = A_2 \\ \mathcal{M}(0,1) = A_3 \end{array} \right.$$

La matrice jacobienne de  $\mathcal{M}$  vaut :

$$\begin{aligned}
 J_{\mathcal{M}} &= \nabla \mathcal{M}(\xi, \eta) \\
 &= \begin{bmatrix} \frac{\partial \mathcal{M}_1}{\partial \xi} & \frac{\partial \mathcal{M}_1}{\partial \eta} \\ \frac{\partial \mathcal{M}_2}{\partial \xi} & \frac{\partial \mathcal{M}_2}{\partial \eta} \\ \frac{\partial \mathcal{M}_3}{\partial \xi} & \frac{\partial \mathcal{M}_3}{\partial \eta} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} \end{bmatrix} \\
 &= \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \\ z_2 - z_1 & z_3 - z_1 \end{bmatrix}
 \end{aligned} \tag{2.11}$$

D'autre part, on pose la fonction  $\mathcal{M}^*(x, y, z)$  comme :

$$\begin{aligned}
 \mathcal{M}^* : \quad \mathbb{R}^3 &\rightarrow [0,1] \times [0,1] \\
 (x, y, z) &\mapsto \mathcal{M}^*(x, y, z) = \begin{pmatrix} \xi \\ \eta \end{pmatrix}
 \end{aligned} \tag{2.12}$$

La matrice jacobienne de  $\mathcal{M}^*$  vaut :

$$\begin{aligned}
 J_{\mathcal{M}^*} &= \nabla \mathcal{M}^*(x, y, z) \\
 &= \begin{bmatrix} \frac{\partial \mathcal{M}_1^*}{\partial x}(x, y, z) & \frac{\partial \mathcal{M}_1^*}{\partial y}(x, y, z) & \frac{\partial \mathcal{M}_1^*}{\partial z}(x, y, z) \\ \frac{\partial \mathcal{M}_2^*}{\partial x}(x, y, z) & \frac{\partial \mathcal{M}_2^*}{\partial y}(x, y, z) & \frac{\partial \mathcal{M}_2^*}{\partial z}(x, y, z) \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} & \frac{\partial \xi}{\partial z} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} & \frac{\partial \eta}{\partial z} \end{bmatrix}
 \end{aligned} \tag{2.13}$$

et on en déduit le gradient de  $u$  :

$$\begin{aligned}
 \nabla u &= \begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \\ \frac{\partial u}{\partial z} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\partial \tilde{u}}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \tilde{u}}{\partial \eta} \frac{\partial \eta}{\partial x} \\ \frac{\partial \tilde{u}}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial \tilde{u}}{\partial \eta} \frac{\partial \eta}{\partial y} \\ \frac{\partial \tilde{u}}{\partial \xi} \frac{\partial \xi}{\partial z} + \frac{\partial \tilde{u}}{\partial \eta} \frac{\partial \eta}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \\ \frac{\partial \xi}{\partial z} & \frac{\partial \eta}{\partial z} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial \tilde{u}}{\partial \xi} \\ \frac{\partial \tilde{u}}{\partial \eta} \end{bmatrix} \\
 &= J_{\mathcal{M}^*}^t \nabla \tilde{u}
 \end{aligned} \tag{2.14}$$

avec  $\tilde{u}(\xi, \eta)$  les coordonnées de  $u(x, y, z)$  dans l'élément de référence :

$$u(x, y, z) = u(\mathcal{M}_1(\xi, \eta), \mathcal{M}_2(\xi, \eta), \mathcal{M}_3(\xi, \eta)) = \tilde{u}(\xi, \eta) \tag{2.15}$$

Nous remarquons que :

$$\begin{aligned}
 J_{\mathcal{M}} J_{\mathcal{M}^*} &= \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} & \frac{\partial \xi}{\partial z} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} & \frac{\partial \eta}{\partial z} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\partial x}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial x}{\partial \eta} \frac{\partial \eta}{\partial x} & \frac{\partial x}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial x}{\partial \eta} \frac{\partial \eta}{\partial y} & \frac{\partial x}{\partial \xi} \frac{\partial \xi}{\partial z} + \frac{\partial x}{\partial \eta} \frac{\partial \eta}{\partial z} \\ \frac{\partial y}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial y}{\partial \eta} \frac{\partial \eta}{\partial x} & \frac{\partial y}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial y}{\partial \eta} \frac{\partial \eta}{\partial y} & \frac{\partial y}{\partial \xi} \frac{\partial \xi}{\partial z} + \frac{\partial y}{\partial \eta} \frac{\partial \eta}{\partial z} \\ \frac{\partial z}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial z}{\partial \eta} \frac{\partial \eta}{\partial x} & \frac{\partial z}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial z}{\partial \eta} \frac{\partial \eta}{\partial y} & \frac{\partial z}{\partial \xi} \frac{\partial \xi}{\partial z} + \frac{\partial z}{\partial \eta} \frac{\partial \eta}{\partial z} \end{bmatrix} \\
 &= I_3
 \end{aligned} \tag{2.16}$$

$$\begin{aligned}
 J_{\mathcal{M}^*} J_{\mathcal{M}} &= \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} & \frac{\partial \xi}{\partial z} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} & \frac{\partial \eta}{\partial z} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\partial \xi}{\partial x} \frac{\partial x}{\partial \xi} + \frac{\partial \xi}{\partial y} \frac{\partial y}{\partial \xi} + \frac{\partial \xi}{\partial z} \frac{\partial z}{\partial \xi} & \frac{\partial \xi}{\partial x} \frac{\partial x}{\partial \eta} + \frac{\partial \xi}{\partial y} \frac{\partial y}{\partial \eta} + \frac{\partial \xi}{\partial z} \frac{\partial z}{\partial \eta} \\ \frac{\partial \eta}{\partial x} \frac{\partial x}{\partial \xi} + \frac{\partial \eta}{\partial y} \frac{\partial y}{\partial \xi} + \frac{\partial \eta}{\partial z} \frac{\partial z}{\partial \xi} & \frac{\partial \eta}{\partial x} \frac{\partial x}{\partial \eta} + \frac{\partial \eta}{\partial y} \frac{\partial y}{\partial \eta} + \frac{\partial \eta}{\partial z} \frac{\partial z}{\partial \eta} \end{bmatrix} \\
 &= I_2
 \end{aligned} \tag{2.17}$$

$J_{\mathcal{M}^*}$  est donc la matrice pseudo-inverse de  $J_{\mathcal{M}}$ . En outre, on pose :

$$\begin{aligned}
 M_{\mathcal{M}} &= (J_{\mathcal{M}^*} J_{\mathcal{M}}^t)^{-1} \\
 &= (J_{\mathcal{M}^*}^t)^{-1} (J_{\mathcal{M}})^{-1} \\
 &= (J_{\mathcal{M}}^{-1})^t J_{\mathcal{M}^*}^{-1} \\
 &= (J_{\mathcal{M}})^t J_{\mathcal{M}} \\
 &= \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \\ z_2 - z_1 & z_3 - z_1 \end{bmatrix}^t \cdot \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \\ z_2 - z_1 & z_3 - z_1 \end{bmatrix}
 \end{aligned} \tag{2.18}$$

L'élément d'aire  $d\mathcal{S}$  de la surface est donné par :

$$d\mathcal{S} = |\vec{a}_1 \wedge \vec{a}_2| d\xi d\eta \tag{2.19}$$

avec  $\vec{a}_1$  et  $\vec{a}_2$  les vecteurs de base du plan tangent au point  $(\xi, \eta)$  :

$$\begin{cases} \vec{a}_1 = \frac{\partial \mathcal{M}}{\partial \xi}(\xi, \eta) \\ \vec{a}_2 = \frac{\partial \mathcal{M}}{\partial \eta}(\xi, \eta) \end{cases} \tag{2.20}$$

$$\begin{aligned}
 \vec{a}_1 &= \frac{\partial \mathcal{M}}{\partial \xi}(\xi, \eta) \\
 &= \begin{bmatrix} \frac{\partial \mathcal{M}_1}{\partial \xi}(\xi, \eta) \\ \frac{\partial \mathcal{M}_2}{\partial \xi}(\xi, \eta) \\ \frac{\partial \mathcal{M}_3}{\partial \xi}(\xi, \eta) \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} \\ \frac{\partial y}{\partial \xi} \\ \frac{\partial z}{\partial \xi} \end{bmatrix}
 \end{aligned} \tag{2.21}$$

$$\begin{aligned}
 \vec{a}_2 &= \frac{\partial \mathcal{M}}{\partial \eta}(\xi, \eta) \\
 &= \begin{bmatrix} \frac{\partial \mathcal{M}_1}{\partial \eta}(\xi, \eta) \\ \frac{\partial \mathcal{M}_2}{\partial \eta}(\xi, \eta) \\ \frac{\partial \mathcal{M}_3}{\partial \eta}(\xi, \eta) \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \eta} \\ \frac{\partial z}{\partial \eta} \end{bmatrix}
 \end{aligned} \tag{2.22}$$

Il vient :

$$\vec{a}_1 \wedge \vec{a}_2 = \begin{bmatrix} \frac{\partial y}{\partial \xi} \frac{\partial z}{\partial \eta} - \frac{\partial z}{\partial \xi} \frac{\partial y}{\partial \eta} \\ \frac{\partial z}{\partial \xi} \frac{\partial x}{\partial \eta} - \frac{\partial x}{\partial \xi} \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial y}{\partial \xi} \frac{\partial x}{\partial \eta} \end{bmatrix} \tag{2.23}$$

$$\begin{aligned}
 |\vec{a}_1 \wedge \vec{a}_2| &= \sqrt{\left[ \left[ \frac{\partial y}{\partial \xi} \frac{\partial z}{\partial \eta} - \frac{\partial z}{\partial \xi} \frac{\partial y}{\partial \eta} \right]^2 + \left[ \frac{\partial z}{\partial \xi} \frac{\partial x}{\partial \eta} - \frac{\partial x}{\partial \xi} \frac{\partial z}{\partial \eta} \right]^2 + \left[ \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial y}{\partial \xi} \frac{\partial x}{\partial \eta} \right]^2 \right]} \\
 &= \sqrt{\det(J_{\mathcal{M}}^t J_{\mathcal{M}})} \\
 &= \sqrt{\det(M_{\mathcal{M}})}
 \end{aligned} \tag{2.24}$$

L'élément d'aire vaut donc :

$$dS = \sqrt{\det(M_{\mathcal{M}})} d\xi d\eta \tag{2.25}$$

On note  $\mathcal{S}_e$  la surface de l'élément  $\mathcal{T}_e$  du maillage, ce qui donne  $\mathcal{S} \simeq \sum_e \mathcal{S}_e$ . L'équation (2.8) devient donc :

$$0 = \int_{\mathcal{S}} \nabla u \cdot \nabla w dS \simeq \sum_e \left( \int_{\mathcal{S}_e} \nabla u \cdot \nabla w dS \right) \quad \forall w \tag{2.26}$$

En particulier, en utilisant l'écriture matricielle du produit scalaire puis en intégrant sur l'élément de référence, il vient :

$$\begin{aligned}
 \int_{\mathcal{S}_e} \nabla u \cdot \nabla w dS &= \int_{\mathcal{S}_e} (J_{\mathcal{M}^*}^t \nabla \tilde{u})^t J_{\mathcal{M}^*}^t \nabla \tilde{w} \sqrt{\det(M_{\mathcal{M}})} d\xi d\eta \\
 &= \int_{\xi} \int_{\eta} \nabla \tilde{u}^t J_{\mathcal{M}^*} J_{\mathcal{M}^*}^t \nabla \tilde{w} \sqrt{\det(M_{\mathcal{M}})} d\xi d\eta \\
 &= \int_{\xi} \int_{\eta} \nabla \tilde{u}^t (M_{\mathcal{M}})^{-1} \nabla \tilde{w} \sqrt{\det(M_{\mathcal{M}})} d\xi d\eta
 \end{aligned} \tag{2.27}$$

Afin de calculer la valeur de  $\nabla \tilde{u}$  en fonction des points  $B_1$ ,  $B_2$  et  $B_3$ , on introduit la fonction bijective  $m(\xi, \eta)$  qui donne les coordonnées courantes du point  $B(u, v)$  du triangle  $\mathcal{T}_j$  à partir de l'élément de référence :

$$m : [0,1] \times [0,1] \rightarrow \mathbb{R}^2$$

$$(\xi, \eta) \mapsto m(\xi, \eta) = \begin{pmatrix} u \\ v \end{pmatrix} \quad (2.28)$$

$$m(\xi, \eta) = \underbrace{(1 - \xi - \eta)}_{N_1} B_1 + \underbrace{\xi}_{N_2} B_2 + \underbrace{\eta}_{N_3} B_3 \quad (2.29)$$

avec  $N_i$  les fonctions de forme en éléments finis. Les nœuds sommets de l'élément de référence se transforment en les sommets  $B_i$  :

$$\begin{cases} m(0,0) = B_1 \\ m(1,0) = B_2 \\ m(0,1) = B_3 \end{cases} \quad (2.30)$$

Il vient :

$$\tilde{u}(u, v) = \tilde{u}(m_1(\xi, \eta), m_2(\xi, \eta)) = \tilde{u}(\xi, \eta) \quad (2.31)$$

La matrice jacobienne de  $m$  vaut :

$$\begin{aligned} J_m &= \nabla m(\xi, \eta) \\ &= \begin{bmatrix} \frac{\partial m_1}{\partial \xi} & \frac{\partial m_1}{\partial \eta} \\ \frac{\partial m_2}{\partial \xi} & \frac{\partial m_2}{\partial \eta} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\partial u}{\partial \xi} & \frac{\partial u}{\partial \eta} \\ \frac{\partial v}{\partial \xi} & \frac{\partial v}{\partial \eta} \end{bmatrix} \\ &= \begin{bmatrix} u_2 - u_1 & u_3 - u_1 \\ v_2 - v_1 & v_3 - v_1 \end{bmatrix} \end{aligned} \quad (2.32)$$

De l'équation (2.29), on tire que :

$$\begin{aligned} \begin{bmatrix} \frac{\partial \tilde{u}}{\partial \xi} \\ \frac{\partial \tilde{u}}{\partial \eta} \end{bmatrix} &= J_m^t \\ &= \underbrace{\begin{bmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}}_D \cdot \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix} \end{aligned} \quad (2.33)$$

L'équation (2.27) devient donc :

$$\begin{aligned}
 \int_{S_e} \nabla u \cdot \nabla w \, dS &= \int_{\xi} \int_{\eta} \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix}^t D^t (M_{\mathcal{M}})^{-1} D \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \sqrt{\det(M_{\mathcal{M}})} \, d\xi \, d\eta \\
 &= \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix}^t \int_{\xi} \int_{\eta} D^t (M_{\mathcal{M}})^{-1} D \sqrt{\det(M_{\mathcal{M}})} \, d\xi \, d\eta \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \\
 &= \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix}^t D^t (M_{\mathcal{M}})^{-1} D \sqrt{\det(M_{\mathcal{M}})} \int_0^1 \int_0^{1-\xi} d\xi \, d\eta \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \\
 &= \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix}^t \frac{D^t (M_{\mathcal{M}})^{-1} D \sqrt{\det(M_{\mathcal{M}})}}{2} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}
 \end{aligned} \tag{2.34}$$

On remarque que la matrice élémentaire  $\mathcal{A}_{\mathcal{T}_e}$  du triangle  $\mathcal{T}_e$  est donnée par :

$$\mathcal{A}_{\mathcal{T}_e} = \frac{D^t (M_{\mathcal{M}})^{-1} D \sqrt{\det(M_{\mathcal{M}})}}{2} \tag{2.35}$$

On assemble les matrices élémentaires  $\mathcal{A}_{\mathcal{T}_e}$  afin d'obtenir la matrice globale  $\mathcal{A}$  qui correspond à l'opérateur de passage de la forme 3D facettée vers le disque.

La relation (2.26) est vérifiée si et seulement si :

$$B^t \cdot \mathcal{A} \cdot W = \sum_e B_e^t \cdot \mathcal{A}_{\mathcal{T}_e} \cdot W_e = 0 \quad \forall W \tag{2.36}$$

$\mathcal{A}$  étant symétrique, l'équation (2.8) revient à résoudre le système linéaire :

$$\mathcal{A} \cdot B = 0 \tag{2.37}$$

avec  $B$  les solutions du problème discrétisé et la condition aux limites  $B = u_0$  sur  $\partial S_1$ .

La démarche ci-dessus est identique pour les coordonnées  $v$ . La matrice  $\mathcal{A}$  obtenue est donc identique pour les deux coordonnées. Le système (2.9) revient à résoudre indépendamment les équations du système :

$$\begin{cases} \mathcal{A} \cdot B_u = 0 \\ \mathcal{A} \cdot B_v = 0 \end{cases} \tag{2.38}$$

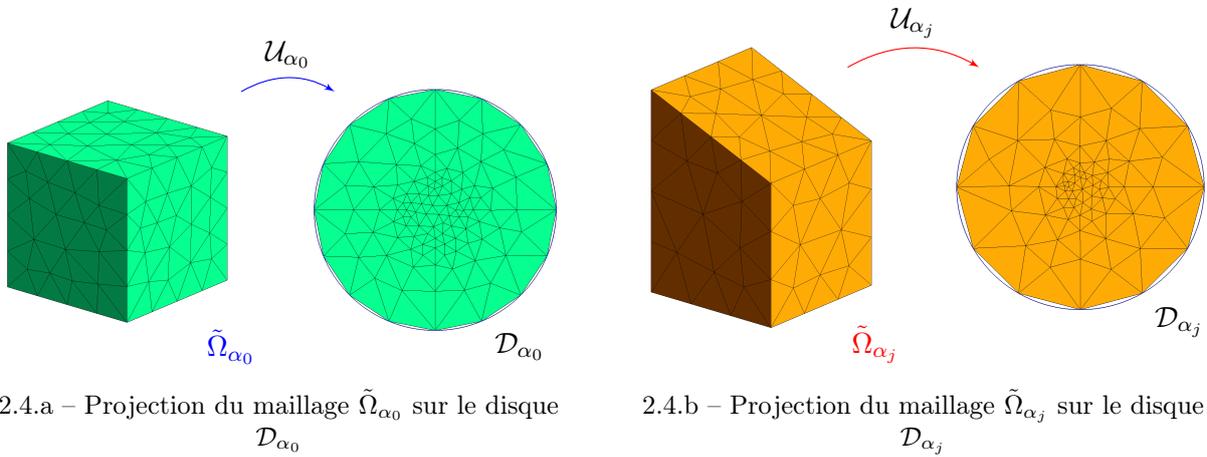
Afin de déterminer les valeurs  $(u, v)$ , on utilise une méthode classique de résolution de problèmes linéaires. Lors de l'implémentation, nous nous sommes appuyés sur la librairie *umfpack* notamment disponible dans le logiciel **Scilab**<sup>®</sup>. Pour prendre en compte la condition  $u = u_0$  et  $v = v_0$ , il est possible d'éliminer les degrés de liberté ou d'appliquer un multiplicateur de Lagrange. Les deux méthodes donnent des résultats équivalents.

Le développement de la méthode de projection harmonique permet de comprendre son fonctionnement et son implémentation est facilitée par cette démarche. Cette projection est ensuite utilisée pour évaluer le déplacement des nœuds d'un maillage entre deux formes [Froment 2014]. Nous présentons dans le prochain paragraphe le fonctionnement global de ce processus.

### 2.1.4 Reconstruction de l'isoconnectivité

La méthode de projection harmonique permet de passer d'une forme 3D à un domaine 2D. Dans notre cas, le domaine choisi est un disque unité. Le calcul de sensibilité sur maillage a pour but d'évaluer le déplacement d'un point d'un maillage suite à la variation d'un paramètre. Toutefois, comme nous l'avons vu dans la partie 1.5.3, il n'y pas d'isoconnectivité entre les maillages générés sur deux formes différentes, c'est-à-dire qu'il n'y a pas le même nombre de nœuds et d'éléments.

Pour effectuer le calcul de sensibilité, l'isoconnectivité doit être reconstruite entre les deux formes. Les deux maillages sont respectivement dépliés sur des disques paramétriques par la projection harmonique présentée ci-dessus. Les cercles obtenus sont :  $\mathcal{D}_{\alpha_0}$  pour  $\tilde{\Omega}_{\alpha_0}$  et  $\mathcal{D}_{\alpha_j}$  pour  $\tilde{\Omega}_{\alpha_j}$ . Ces opérations sont décrites par les équations (2.39) pour la figure 2.4.a et (2.40) pour la figure 2.4.b.



2.4.a – Projection du maillage  $\tilde{\Omega}_{\alpha_0}$  sur le disque  $\mathcal{D}_{\alpha_0}$

2.4.b – Projection du maillage  $\tilde{\Omega}_{\alpha_j}$  sur le disque  $\mathcal{D}_{\alpha_j}$

FIGURE 2.4 – Méthode de projection – 1 : projection des maillages sur un disque 2D

$$\begin{aligned} \mathcal{U}_{\alpha_0} : \tilde{\Omega}_{\alpha_0} \subset \mathbb{R}^3 &\longrightarrow \tilde{\mathcal{D}}_{\alpha_0} (0,1) \\ (x,y,z) &\longmapsto \begin{cases} u_{\alpha_0}(x,y,z) \\ v_{\alpha_0}(x,y,z) \end{cases} \end{aligned} \quad (2.39)$$

$$\begin{aligned} \mathcal{U}_{\alpha_j} : \tilde{\Omega}_{\alpha_j} \subset \mathbb{R}^3 &\longrightarrow \tilde{\mathcal{D}}_{\alpha_j} (0,1) \\ (x,y,z) &\longmapsto \begin{cases} u_{\alpha_j}(x,y,z) \\ v_{\alpha_j}(x,y,z) \end{cases} \end{aligned} \quad (2.40)$$

Connaissant la fonction  $\mathcal{U}_{\alpha_j}$  qui nous a permis de passer du maillage de la forme déformée à son disque, nous appliquons son inverse  $\mathcal{U}_{\alpha_j}^{-1}$  sur le disque  $\mathcal{D}_{\alpha_0}$  (la fonction est bijective du fait que l'on passe d'une forme 3D discrétisée à un domaine 2D). Connaissant les coordonnées d'un nœud du maillage de la forme initiale  $\tilde{\Omega}_{\alpha_0}$  dans l'espace et sur le plan porté par le disque  $\mathcal{D}_{\alpha_0}$ , il est possible de connaître sa projection par la fonction inverse et de déterminer ses nouvelles coordonnées dans l'espace (cf. figure 2.5). L'évaluation du déplacement est calculée par la différence finie

## 2.2. Adaptation de la projection harmonique appliquée aux géométries multifrontières

de ses coordonnées dans l'espace (cf. figure 2.6). En appliquant cela à l'ensemble des points et pour chaque paramètre  $\alpha_j$ , il est possible d'évaluer la sensibilité du maillage par rapport aux paramètres, ce qui est décrit par l'équation (2.41).

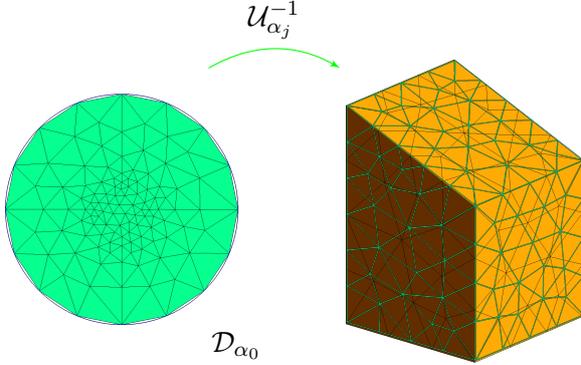


FIGURE 2.5 – Méthode de projection – 2  
Projection du maillage de la forme initiale  
sur la forme perturbée

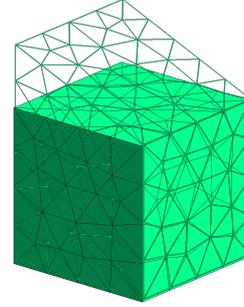


FIGURE 2.6 – Méthode de projection – 3  
Comparaison des coordonnées dans  
l'espace 3D

$$\partial_{\alpha_j} \tilde{\Omega}_{\alpha_0} \simeq \frac{\mathcal{U}_{\alpha_j}^{-1} \left( \tilde{\mathcal{D}}_{\alpha_0} (0,1) \right) - \tilde{\Omega}_{\alpha_0}}{\delta \alpha_j} \quad (2.41)$$

Le processus présenté ici permet d'évaluer la sensibilité d'un maillage par rapport à la variation d'un paramètre de la CAO. Toutefois, la principale limitation donnée par [Froment 2014] concernant le nombre de frontière limité à 1 porte un préjudice important à la méthode puisqu'il est nécessaire d'effectuer un découpage de la géométrie pour avoir des surfaces homéomorphes au disque. Cela implique donc la mise en place de plans de découpe pendant la création de la CAO et se traduit par des étapes supplémentaires dans les processus industriels. Afin d'obvier à cela, nous proposons des adaptations de la méthode pour que son utilisation soit faite de façon transparente.

## 2.2 Adaptation de la projection harmonique appliquée aux géométries multifrontières

Le but principal de cette adaptation est de pouvoir appliquer notre méthode à des géométries plus complexes sans avoir recours à des étapes supplémentaires coûteuses à effectuer dès la création de la géométrie CAO. Cela permet donc une plus grande transparence, une plus grande facilité dans la mise en place de nouveaux cas d'études et donc une meilleure intégration dans les boucles de développement actuelles.

Comme nous l'avons vu dans la partie 2.1.3, la projection harmonique s'applique à des géométries sans faire de distinction au niveau du nombre de frontières à partir du moment où elle possède au moins une frontière. Néanmoins, la surface 3D et celle dans le domaine 2D restent

homéomorphes. Ainsi, lors de la projection d'une géométrie ayant un nombre de frontières supérieur à 1, on obtient un disque avec des trous. La figure 2.7 présente la projection appliquée à une géométrie ayant deux frontières. Pour effectuer cette projection, nous continuons de travailler avec les maillages, le processus de la partie 2.1.1 permet de détecter l'ensemble des frontières du maillage et cela quel que soit leur nombre. Toutefois, une seule frontière est fixée sur le bord du disque par la condition de Dirichlet, les autres sont laissées libres avec une condition de Neumann. Ces frontières, notées  $\partial\mathcal{S}\setminus\partial\mathcal{S}_1$  dans le système (2.1), se projettent à l'intérieur du disque et un changement de paramètre influence leur position comme illustré de façon schématique par la figure 2.8.

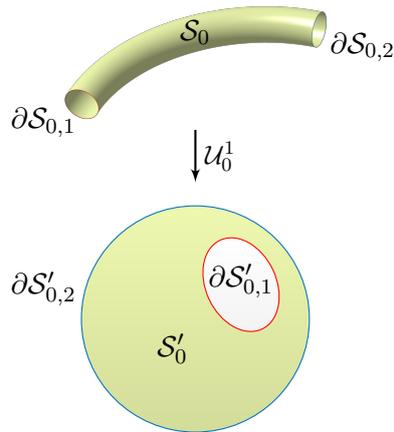


FIGURE 2.7 – Illustration de la projection harmonique appliquée à une géométrie avec deux frontières

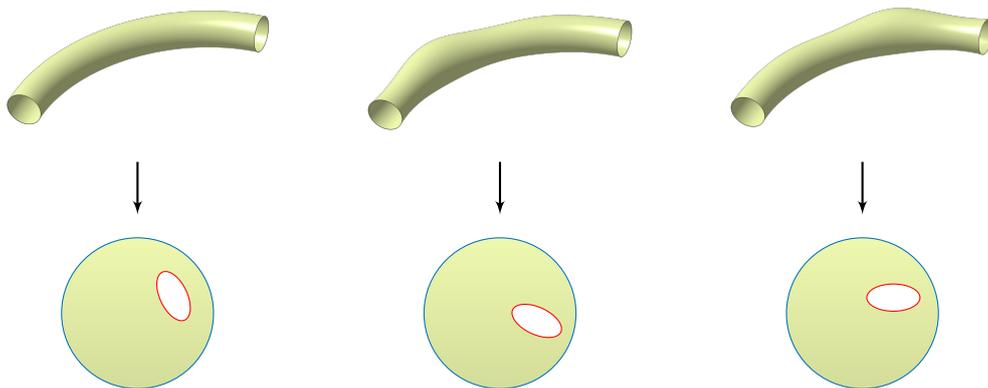


FIGURE 2.8 – Illustration du changement de position d'une frontière suite à une variation de paramètre

Afin de confirmer les schémas de la figure 2.8, les maillages 2.9.a et 2.9.b sont projetés sur le disque unité par la fonction de projection présentée précédemment. Nous obtenons respectivement les figures 2.9.c et 2.9.d. En superposant les deux disques obtenus (cf. figure 2.9.e), nous remarquons un décalage entre les frontières intérieures. Il est donc impossible dans ce cas d'effectuer une comparaison entre les deux disques. C'est pour cette raison que le nombre de frontières

## 2.2. Adaptation de la projection harmonique appliquée aux géométries multifrontières

est jusqu'ici limité à 1. La gestion de la position des frontières rend possible cette comparaison pour des géométries ayant plus d'une frontière.

Afin d'illustrer les explications des prochains paragraphes, nous nous appuyons sur les illustrations de la figure 2.10 qui sont des représentations schématiques des maillages 2.9.c et 2.9.d. La figure 2.10.a est assimilée au maillage de la géométrie initiale (nœuds circulaires) et la figure 2.10.b à celui de la forme perturbée (nœuds rectangulaires). Les deux figures montrent les nœuds – en bleu – de la frontière extérieure situés sur le bord du disque et les nœuds – en rouge – de la frontière intérieure.

Il est donc nécessaire d'imposer l'emplacement de ces trous pour avoir une superposition des frontières respectives. Il faut ensuite répercuter la modification des frontières sur l'ensemble des points du disque. Cela est indispensable pour effectuer l'interpolation et calculer les déplacements des nœuds.

### 2.2.1 Déplacement des frontières intérieures

Dans cette partie, seules les frontières intérieures sont concernées. Elles sont donc isolées comme le montre les figures 2.11.

Les opérations effectuées sur la frontière intérieure de la géométrie initiale sont représentées par les figures 2.12 et celles effectuées sur la frontière intérieure de la géométrie perturbée sont représentées par les figures 2.13. Les paragraphes suivants détaillent les différentes étapes qui permettent d'obtenir la superposition des frontières.

Dans un premier temps, pour faciliter la superposition des frontières, celles-ci sont rendues circulaires. En effet, il est plus aisé d'effectuer la superposition sur des formes circulaires que sur les formes convexes non régulières données par la projection harmonique.

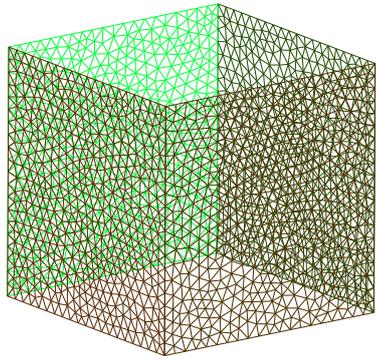
Pour circulariser la frontière, deux informations sont nécessaires : le centre et le rayon. Pour cela, les coordonnées des nœuds situés sur la frontière intérieure sont utilisées. Lors de la projection harmonique, l'identification de ces nœuds est conservée telle qu'elle était originellement.

En premier lieu, la position du barycentre  $X_{G_0}$  de la frontière de la forme initiale projetée est déterminée avec les coordonnées de chacun des points  $V_i \in \partial\mathcal{S}_2$ . Toutefois, si on ne prend en compte que la position des nœuds de la frontière, le barycentre sera plus proche d'une zone ayant une concentration de points plus forte. Dans le but d'éviter cela, une pondération par rapport à la concentration sur cette même frontière est utilisée. Les coordonnées  $X_{V_i}$  d'un point sont donc pondérées par la distance  $l_{(V_{i-1}, V_{i+1})}$  entre ses voisins directs comme le montre l'équation (2.42). Cela permet d'avoir une distance homogène entre le barycentre et l'ensemble des points de la frontière.

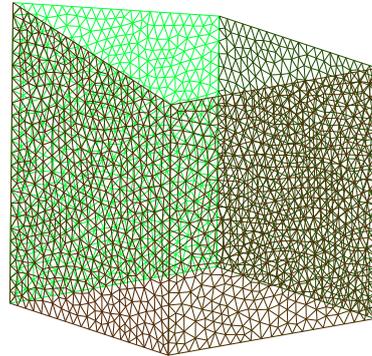
$$X_{G_0} = \frac{\sum_{i=1}^n l_{(V_{i-1}, V_{i+1})} X_{V_i}}{\sum_{i=1}^n l_{(V_{i-1}, V_{i+1})}} \quad \text{avec } V_i \in \partial\mathcal{S}_2 \quad \text{et} \quad \partial\mathcal{S}_2 \in \tilde{\Omega}_{\alpha_0} \quad (2.42)$$

Cette distance est ensuite évaluée pour chacun des points. La plus petite d'entre elles, notée  $r_{min}$  constitue le rayon de notre frontière intérieure (2.43).

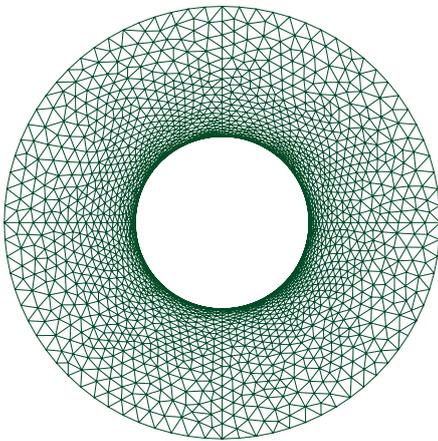
$$r_{min} = \min(X_{V_i} - X_{G_0}) \quad (2.43)$$



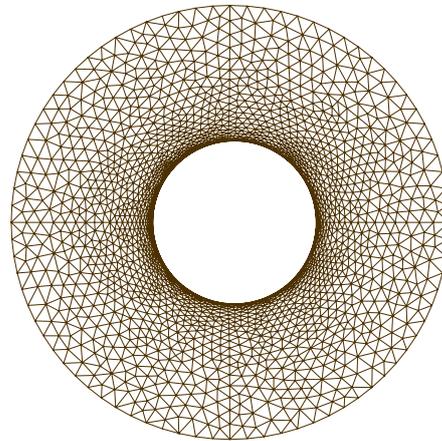
2.9.a – Maillage de la géométrie initiale



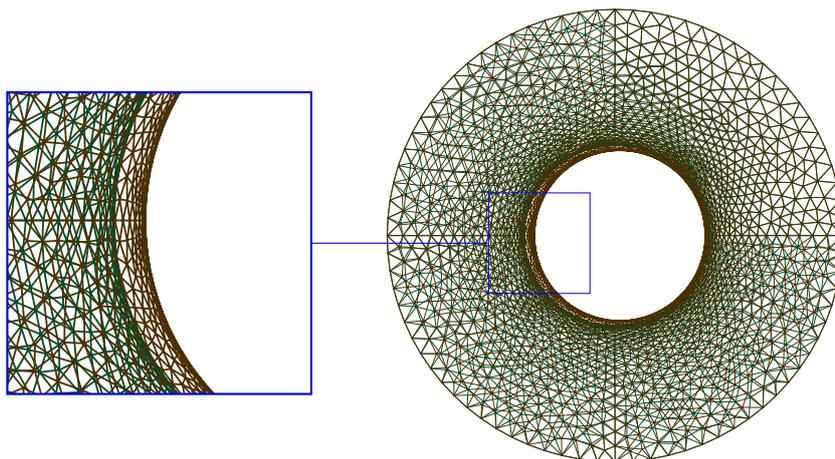
2.9.b – Maillage de la géométrie perturbée



2.9.c – Projection de la forme initiale



2.9.d – Projection de la forme perturbée



2.9.e – Superposition des deux projections

FIGURE 2.9 – Projection d'une géométrie avec 2 frontières

## 2.2. Adaptation de la projection harmonique appliquée aux géométries multifrontières

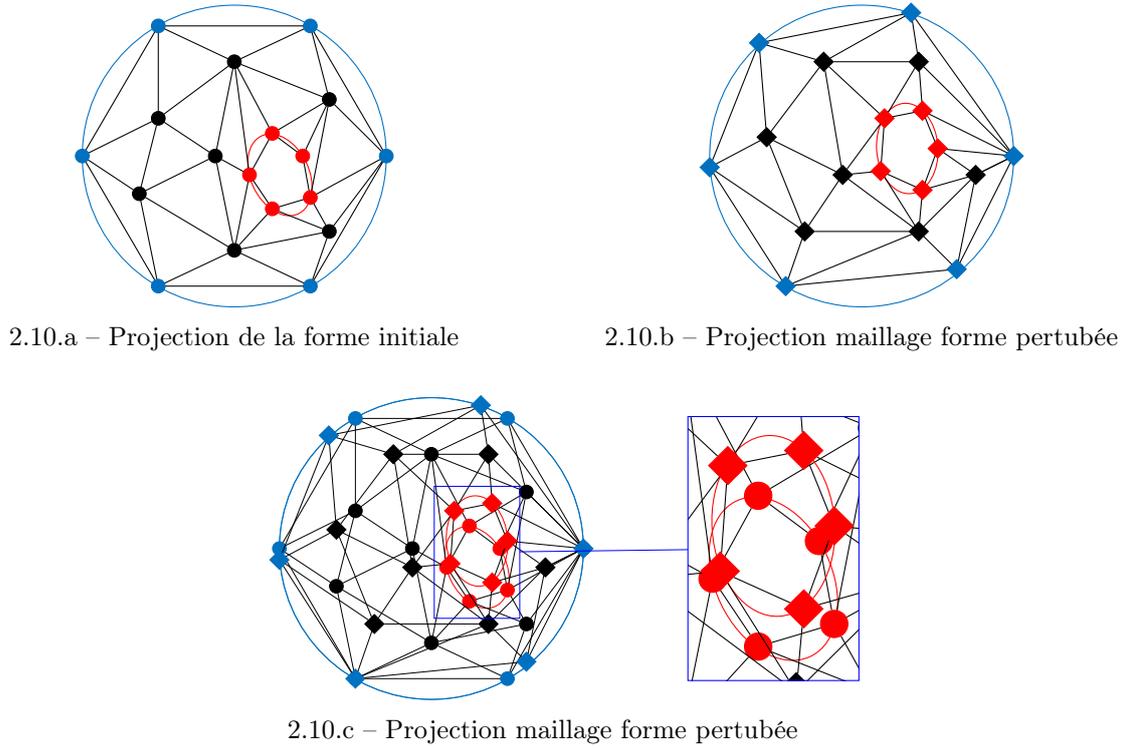


FIGURE 2.10 – Représentations simplifiées de maillages obtenus après la projection harmonique

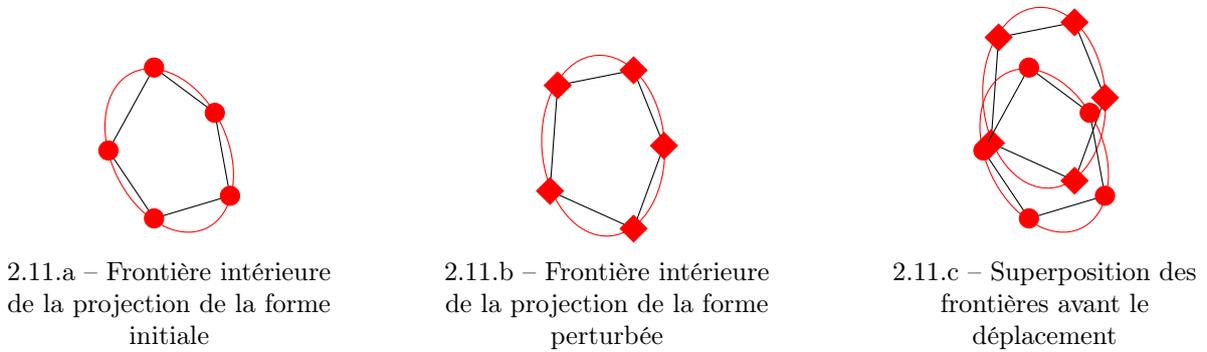


FIGURE 2.11 – Isolement des frontières intérieures des projections de la figure 2.10

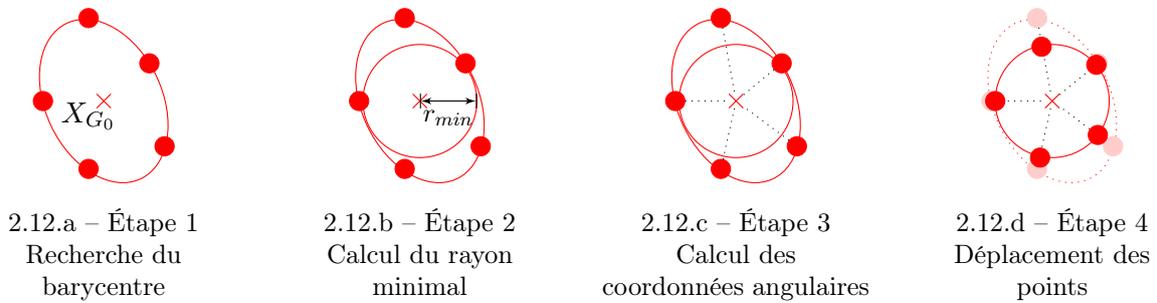


FIGURE 2.12 – Opérations sur la frontière intérieure du maillage de la forme initiale

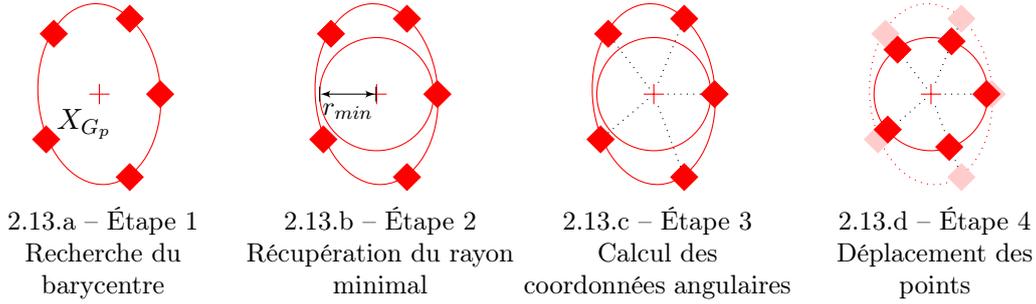


FIGURE 2.13 – Opérations sur la frontière intérieure du maillage de la forme perturbée

Il reste ensuite à répartir les points sur la frontière en fonction de leur placement d'origine. Pour cela, les coordonnées cartésiennes de chacun des points sont converties en coordonnées polaires pour ne s'intéresser qu'à la coordonnée angulaire  $\theta_{V_i}$  donnée par l'équation (2.44) avec pour point d'origine le barycentre déterminé à l'étape précédente. Étant donné que les frontières intérieures sont convexes, il n'est pas possible d'avoir deux points pour une même valeur d'angle.

$$\theta_{V_i} = \arctan \left( \frac{v_{V_i} - v_{G_0}}{u_{V_i} - u_{G_0}} \right) \quad (2.44)$$

Les conditions pour le calcul de  $\theta_{V_i}$  sont rappelées par l'équation (2.45).

$$\theta_{V_i} = \begin{cases} \arctan\left(\frac{v}{u}\right) & \text{si } u_{V_i} > 0 \\ \arctan\left(\frac{v}{u}\right) + \pi & \text{si } u_{V_i} < 0 \text{ et } v_{V_i} \geq 0 \\ \arctan\left(\frac{v}{u}\right) - \pi & \text{si } u_{V_i} < 0 \text{ et } v_{V_i} < 0 \\ \frac{\pi}{2} & \text{si } u_{V_i} = 0 \text{ et } v_{V_i} > 0 \\ -\frac{\pi}{2} & \text{si } u_{V_i} = 0 \text{ et } v_{V_i} < 0 \end{cases} \quad (2.45)$$

Avec cette coordonnée angulaire et la coordonnée radiale donnée par le rayon  $r_{min}$ , il est possible de déterminer les nouvelles coordonnées cartésiennes des nœuds de la frontière intérieure avec l'équation (2.46).

$$\begin{cases} u_{V'_i} = u_{G_0} + (r_{min} \cos \theta_{V_i}) \\ v_{V'_i} = v_{G_0} + (r_{min} \sin \theta_{V_i}) \end{cases} \quad (2.46)$$

Dans le cas des frontières des géométries perturbées, nous utilisons l'équation (2.42). Elle devient l'équation (2.47) lorsqu'elle est appliquée aux maillages projetés des formes perturbées pour déterminer le barycentre  $X_{G_p}$  de la frontière.

$$X_{G_p} = \frac{\sum_{i=1}^n l_{(V_{i-1}, V_{i+1})} X_{V_i}}{\sum_{i=1}^n l_{(V_{i-1}, V_{i+1})}} \quad \text{avec } V_i \in \partial \mathcal{S}_2 \text{ et } \partial \mathcal{S}_2 \in \tilde{\Omega}_{\alpha_i} \quad (2.47)$$

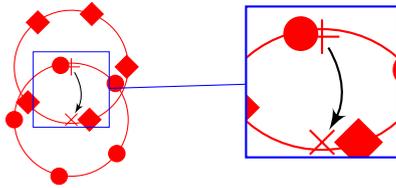
Le barycentre  $X_{G_p}$  permet de calculer les coordonnées angulaires  $\theta_{V_i}$  des points de la frontière en adaptant l'équation (2.44). Celle-ci devient l'équation (2.48) avec les mêmes conditions données

## 2.2. Adaptation de la projection harmonique appliquée aux géométries multifrontières

par l'équation (2.45).

$$\theta_{V_i} = \arctan \left( \frac{v_{V_i} - v_{G_p}}{u_{V_i} - u_{G_p}} \right) \quad (2.48)$$

Finalement, les valeurs du barycentre  $X_{G_0}$  et du rayon  $r_{min}$  de la géométrie initiale sont utilisées pour déplacer les points des frontières des géométries perturbées avec l'équation (2.46) en utilisant cette fois les valeurs de  $\theta_{V_i}$  déterminées à l'équation (2.48). Les nœuds de chaque frontière intérieure sont placés sur un cercle de même rayon  $r_{min}$  et de même centre  $X_{G_0}$  tout en gardant leur propre répartition.



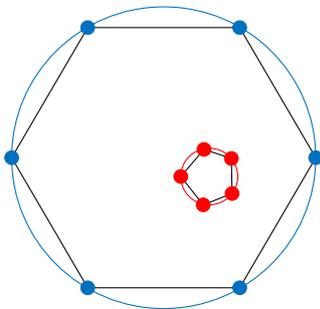
2.14.a – Superposition des frontières avant le déplacement



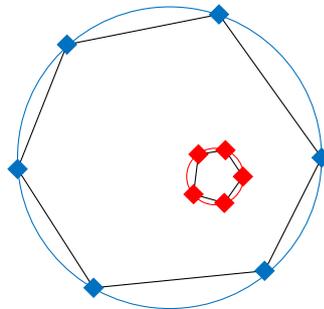
2.14.b – Superposition des frontières suite au déplacement

FIGURE 2.14 – Superposition des frontières suite à leur circularisation et au déplacement de barycentre

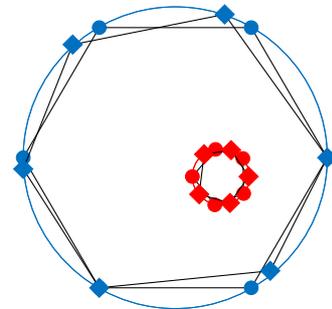
La position souhaitée des frontières intérieures aux disques est donc obtenue comme illustré par la figure 2.14. Il faut maintenant répercuter ce déplacement à l'ensemble des nœuds du disque.



2.15.a – Projection de la forme initiale



2.15.b – Projection maillage forme perturbée



2.15.c – Superposition des maillages

FIGURE 2.15 – Représentations simplifiées de maillages obtenus après la projection harmonique et le déplacement des frontières

Dans les paragraphes précédents, le nombre de frontières est au maximum égal à deux. Dans le cas d'un nombre de frontières supérieur à 2, une identification des frontières doit être effectuée. La frontière contenant le nœud  $V_s$  constitue la frontière projetée sur le bord du disque. Les autres frontières, c'est-à-dire celles situées à l'intérieur du disque, sont en revanche difficilement identifiables. Cela pose problème pour la superposition entre des frontières correspondantes. Il faut donc réussir à récupérer les identifiants des frontières sur le maillage en les ayant préalablement renseignés dans le modèle CAO.

Le processus présenté dans cette partie superpose les frontières respectives entre elles. Il faut ensuite répercuter cette transformation sur l'ensemble des nœuds du maillage.

### 2.2.2 Utilisation de la double projection harmonique

Après avoir défini la position des frontières pour chacun des disques, il est nécessaire de répercuter le déplacement des frontières sur l'ensemble des nœuds du maillage. En s'inspirant de la définition du problème de projection harmonique, il est envisagé d'utiliser une condition de Dirichlet sur les frontières qui se projettent à l'intérieur du disque en plus de celle appliquée sur la frontière qui se projette sur le bord du disque. En effet, la partie précédente nous a permis de déterminer les coordonnées des nœuds des frontières intérieures pour qu'elle se superposent respectivement sur les frontières de la projection de géométrie de référence. Le système (2.1) est donc adapté en prenant en compte la condition de Dirichlet pour les frontières intérieures, ce qui donne le système (2.49). La nouvelle résolution permet d'obtenir les disques de projection de l'ensemble des formes générées tout en ayant contrôlé l'emplacement de la frontière intérieure comme illustré par la figure 2.16.

$$\begin{cases} \nabla^2 u = 0, & \nabla^2 v = 0 & \text{sur } \mathcal{S}, \\ u = \bar{u}(\vec{x}), & v = \bar{v}(\vec{x}) & \text{sur } \partial\mathcal{S}_1, \\ u = \bar{u}'(\vec{x}), & v = \bar{v}'(\vec{x}) & \text{sur } \partial\mathcal{S} \setminus \partial\mathcal{S}_1. \end{cases} \quad (2.49)$$

Le changement de conditions aux limites impacte fortement la façon de résoudre le problème et la solution obtenue ne correspond pas toujours à nos attentes. Les essais sur différentes géométries ont montré deux cas de figure. Dans le cas où la frontière de référence est plus petite que celle à adapter et qu'elle se situe à l'intérieur, la résolution du problème harmonique se fait correctement. Toutefois, lorsque la frontière de référence est plus grande que celle à adapter ou qu'elle ne se situe pas au même emplacement, des erreurs sont obtenues lors de la deuxième résolution du problème harmonique. La projection observée démarre correctement de la frontière intérieure fixée mais les points se dirigent vers l'intérieur de celle-ci avant de rejoindre la frontière extérieure. Comme le montre la figure 2.17, la concentration à gauche de la figure (maillage vert) correspond au maillage de la forme initiale, la première projection de la forme perturbée donne la concentration à droite (maillage noir) et le maillage résultant de la deuxième étape de projection, après avoir fixé la frontière intérieure de la projection de la géométries perturbées, est mis en évidence par le zoom (maillage marron et jaune).

Si on se réfère à l'énoncé du problème donné par le système (2.49), celui-ci n'est plus un problème elliptique mais parabolique et nécessite de modifier la métrique de la projection [Gu 2008]. La méthode de résolution linéaire ne permet pas de résoudre directement ce problème avec de telles conditions. Afin de pallier cela, il serait nécessaire d'utiliser des méthodes itératives comme par exemple les méthodes basées sur le flot de Ricci<sup>b</sup> pour déterminer les coordonnées des points sur le disque qui respectent les conditions de Dirichlet sur chacune des frontières. Toutefois, les méthodes itératives sont plus coûteuses et leur mise en place implique de définir différents paramètres comme des critères de convergence [Gu 2008, p. 255].

Au final, l'utilisation d'une condition de Dirichlet sur les autres frontières ne correspond pas aux attentes puisqu'elle se traduit par une plus grande complexité de résolution (temps

---

b. Ces méthodes sont basées sur les travaux de [Hamilton 1982].

## 2.2. Adaptation de la projection harmonique appliquée aux géométries multifrontières

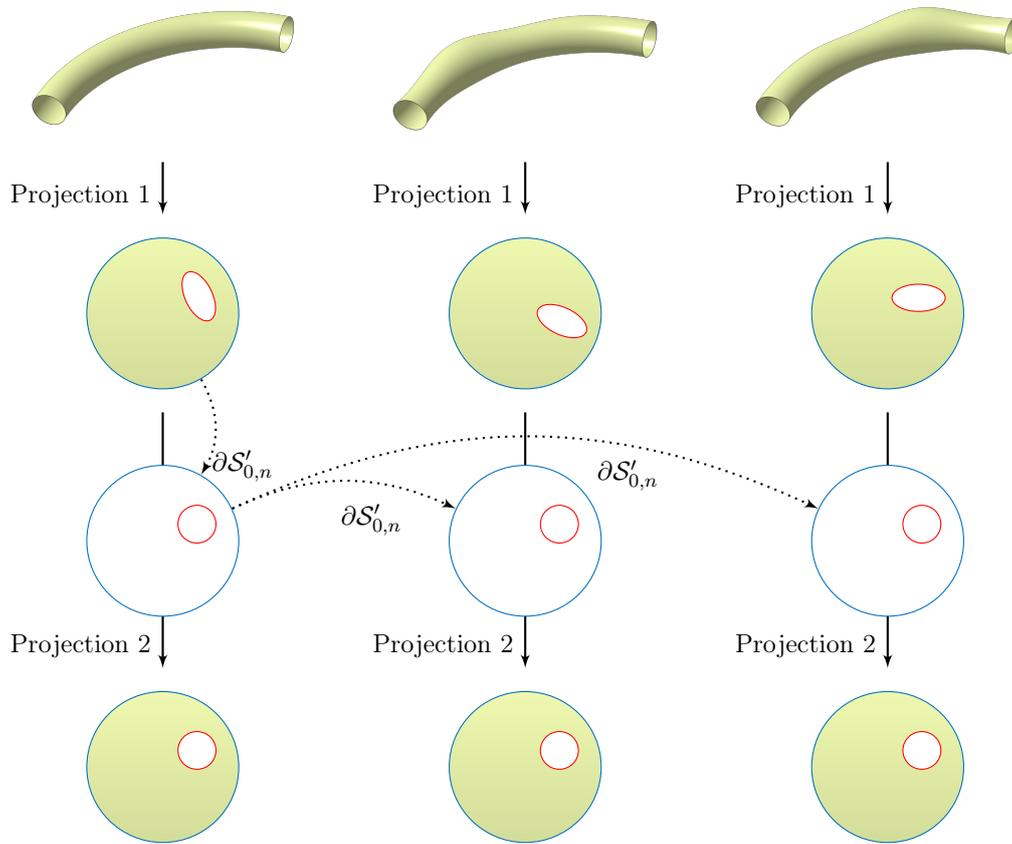


FIGURE 2.16 – Illustration de la méthode par double projection harmonique

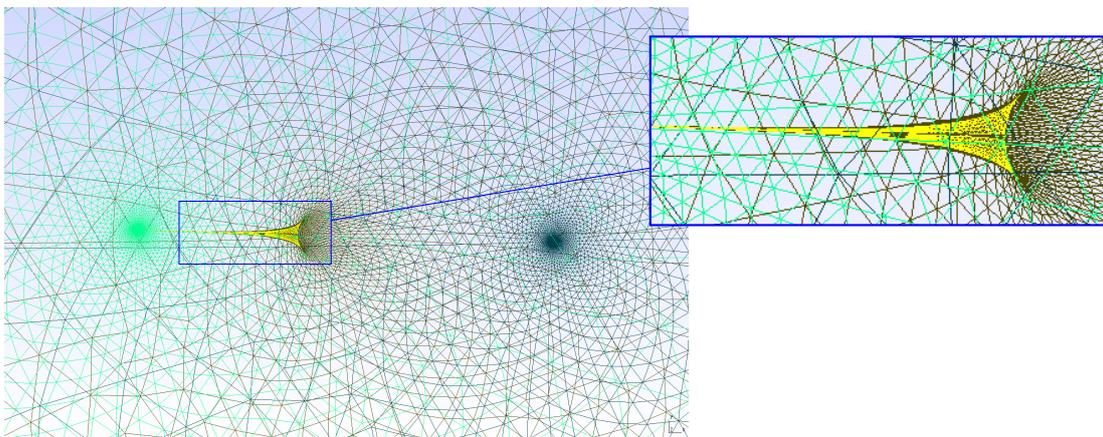


FIGURE 2.17 – Problème de résolution de la deuxième projection harmonique

de résolution, paramètres supplémentaires à gérer...). Cette complexité est donc susceptible de freiner l'intégration de la méthode dans des processus industriels.

### 2.2.3 Utilisation de la triangulation de Delaunay et des coefficients barycentriques

L'utilisation d'une condition de Dirichlet sur les frontières intérieures ne correspond pas à nos attentes. Il est donc nécessaire de trouver une méthode différente pour déplacer les points du maillage en fonction du déplacement des nœuds de la frontière intérieure. Nous proposons d'utiliser une triangulation de Delaunay pour effectuer ce déplacement. En premier lieu, la projection harmonique permet de passer d'une forme 3D à un domaine paramétrique 2D (à savoir un disque unité dans notre cas) comme présenté dans la partie 2.1.

À ce stade, nous connaissons la position des nœuds des frontières intérieures suite à la projection harmonique mais aussi leur position souhaitée pour la superposition avec la frontière de la forme initiale, déterminée dans la partie 2.2.1. Nous connaissons donc le déplacement à effectuer aux nœuds de chacune des frontières pour obtenir les frontières à l'emplacement souhaité.

Pour répercuter ce déplacement sur l'ensemble des points, nous commençons par effectuer une triangulation de Delaunay uniquement sur les points situés sur les frontières extérieures et intérieures. Nous nous appuyons sur la méthode de [Delaunay 1934] pour effectuer la triangulation d'un ensemble de points  $P$  du plan. Cette triangulation, notée  $T_{\mathcal{D}}$ , est effectuée telle qu'aucun point  $P$  ne soit à l'intérieur du cercle circonscrit d'un des triangles de  $T_{\mathcal{D}}$ .

Cette triangulation donne un maillage très grossier du disque. La figure 2.18.a représente la triangulation avant le déplacement des frontières et la figure 2.18.b représente cette même triangulation après l'étape de transformation.

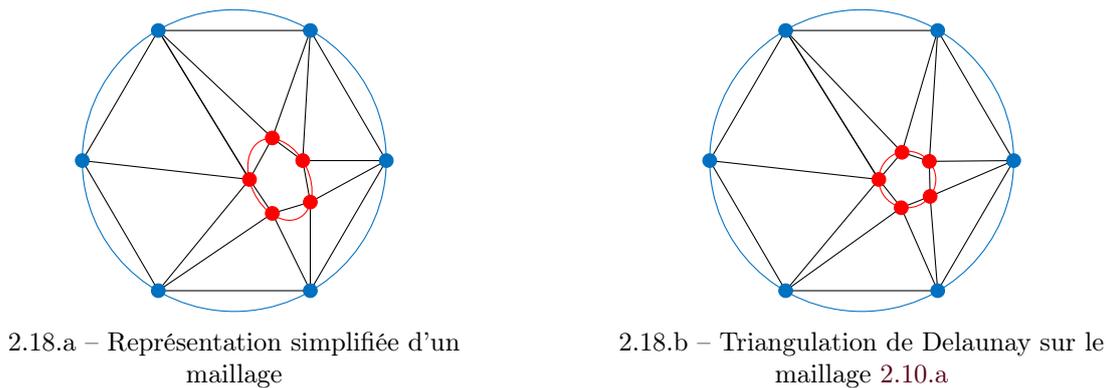


FIGURE 2.18 – Illustration de la triangulation de Delaunay : la triangulation de Delaunay reste identique pour une même géométrie.

Les coefficients barycentriques  $(\lambda_1, \lambda_2, \lambda_3)$  des nœuds du maillage sont ensuite calculés dans chacun des triangles obtenus par la triangulation de Delaunay. Les coefficients barycentriques sont donnés par le système d'équations (2.50) où  $(u_1, v_1), (u_2, v_2), (u_3, v_3)$  sont les coordonnées

## 2.2. Adaptation de la projection harmonique appliquée aux géométries multifrontières

respectives des sommets d'un triangle  $T_k$  et  $(u,v)$  les coordonnées d'un nœud  $V_i$ .

$$\begin{cases} \lambda_1 = \frac{(v_2 - v_3)(u - u_3) + (u_3 - u_2)(v - v_3)}{(v_2 - v_3)(u_1 - u_3) + (u_3 - u_2)(v_1 - v_3)} \\ \lambda_2 = \frac{(v_3 - v_1)(u - u_3) + (u_1 - u_3)(v - v_3)}{(v_2 - v_3)(u_1 - u_3) + (u_3 - u_2)(v_1 - v_3)} \\ \lambda_3 = 1 - \lambda_1 - \lambda_2 \end{cases} \quad (2.50)$$

Le nœud  $V_i$  se situe dans le triangle  $T_k$ , sur un de ses coins ou sur une de ses arêtes si et seulement si :

$$0 \leq \lambda_j \leq 1 \quad \forall j \in 1,2,3 \quad (2.51)$$

Une fois que le point est identifié, il ne peut pas appartenir à un autre triangle, ce qui évite qu'un point situé sur un côté du triangle soit identifié dans le triangle adjacent.

Si un nœud est identifié dans un triangle, les coefficients barycentriques sont utilisés dans l'équation (2.52) avec les coordonnées des sommets du triangle après la transformation de la frontière.

$$\begin{cases} u' = \lambda_1 u'_1 + \lambda_2 u'_2 + \lambda_3 u'_3 \\ v' = \lambda_1 v'_1 + \lambda_2 v'_2 + \lambda_3 v'_3 \end{cases} \quad (2.52)$$

Sachant qu'ici, au moins un couple  $(u'_i, v'_i)$  avec  $i \in 1,2,3$  est égal au couple  $(u_i, v_i)$  de même  $i$ .

Ainsi, les coordonnées  $(u', v')$  du point sont obtenues en relation avec la nouvelle position de la frontière intérieure comme on peut le voir sur la figure 2.19.

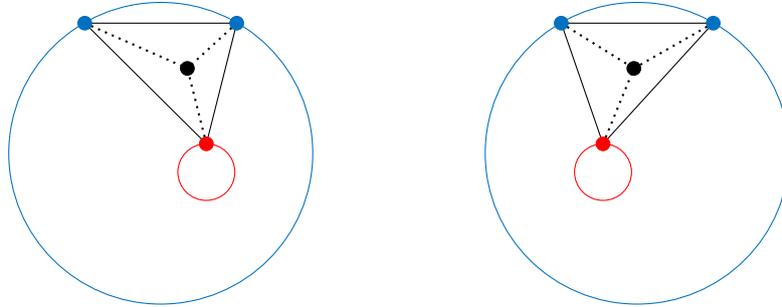


FIGURE 2.19 – Illustration du déplacement d'un point du maillage avec la triangulation de Delaunay

Cette opération est répétée sur l'ensemble des nœuds du maillage conduisant à l'obtention des nouvelles coordonnées des nœuds du disque après le déplacement des frontières intérieures. La superposition obtenue est parfaite pour les frontières intérieures de chacun des disques. L'algorithme 2.2 détaille les étapes effectuées. Si on reprend le cas d'application présenté par les figures 2.9, on observe sur la figure 2.20 que les deux frontières intérieures sont bien superposées.

La triangulation de Delaunay et le calcul des coefficients barycentriques donnent les résultats attendus tout en utilisant des méthodes de calcul simples. Contrairement à la méthode qui utilise une condition de Dirichlet sur les frontières intérieures, le processus ne requiert pas de moyens de résolution importants, ni de gestion de paramètres de convergence. Son intégration et sa transparence sont donc accentuées.

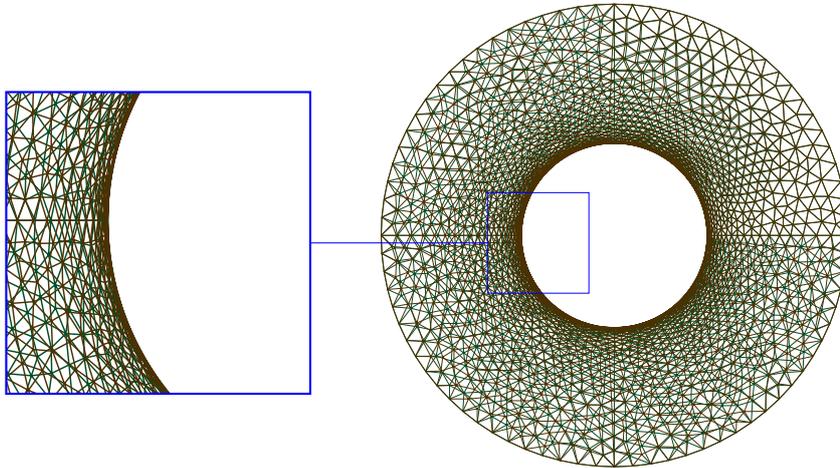


FIGURE 2.20 – Projection d'une géométrie avec 2 frontières

**Données :** maillage triangulaire  $\tilde{\Omega}$  et nouvelles coordonnées des nœuds des frontières

**Début de l'algorithme**

    triangulation de Delaunay sur les points des frontières uniquement

**pour chaque** nœud du maillage, **faire**

**pour chaque** élément de la triangulation, **faire**

            calcul des coefficients barycentrique  $\lambda_1, \lambda_2, \lambda_3$

**si**  $0 \leq \lambda_j \leq 1$  **et**  $0 \leq \lambda_2 \leq 1$  **et**  $0 \leq \lambda_3 \leq 1$  **alors**

                calcul des nouvelles coordonnées du nœud courant

**fin**

**fin**

**fin**

**Fin de l'algorithme**

**Résultat :** Nouvelles coordonnées de l'ensemble des nœuds du maillage

**Algorithme 2.2 :** Algorithme de déplacement des nœuds du maillage

## 2.2. Adaptation de la projection harmonique appliquée aux géométries multifrontières

### Aspect ratio élevé

L'aspect ratio (ou facteur de forme) donne le rapport entre deux dimensions d'une géométrie. Par exemple, l'aspect ratio d'une conduite traduit le rapport entre la longueur de la conduite et son rayon.

La méthode présentée précédemment montre ses limites dans le cadre d'un aspect ratio élevé, cela à cause de la précision numérique. En effet, lors de la projection de géométries avec un aspect ratio important, le diamètre de la frontière intérieure du disque est très faible. Les triangles de Delaunay obtenus sont très étirés entre les deux frontières et deux de leurs côtés sont disproportionnés par rapport au troisième. Les erreurs numériques du calcul des coefficients barycentriques sont donc amplifiées et les conditions ne sont pas respectées à proximité des sommets. Ce qui entraîne une mauvaise répartition des nœuds dans leur triangle respectif.

Pour éviter cela, la dimension des triangles est réduite en fonction de la configuration du disque. Si la frontière intérieure a un diamètre de l'ordre de  $10^{-7}$ , aucune intervention n'est nécessaire. Une frontière intérieure avec un diamètre proche de cet ordre de grandeur peut être obtenue en projetant un tube ayant un rayon  $R = 1 \text{ cm}$  et une hauteur  $H = 12 \text{ cm}$ , soit un aspect ratio  $H/R = 12^c$ .

Si le diamètre est plus faible, des cercles intermédiaires sont ajoutés et permettent de limiter les erreurs numériques. Afin d'être homogène avec la frontière intérieure d'origine, celle-ci sert de référence pour la construction des cercles intermédiaires. Ces cercles comporteront le même nombre de nœuds et le même centre que la frontière intérieure. On utilise pour cela les valeurs  $\theta_{V_i}$  calculées respectivement aux équations (2.44) pour la forme initiale et (2.48) pour les formes perturbées. Les nœuds sont ensuite répartis sur des cercles de différentes tailles qui dépendent du diamètre de la frontière et donc du nombre de cercle requis. Les données du tableau 2.1 donnent les ordres de grandeurs correspondants aux nombre de cercles supplémentaires. Ces valeurs proviennent d'essais effectués sur un cylindre ouvert avec une taille de cellule fixée. Néanmoins, les valeurs sont susceptibles de varier en fonction des formes projetées, des tailles de maillages, du nombre de frontières... Les valeurs du tableau sont des valeurs seuil et les diamètres des cercles intermédiaires sont adaptées automatiquement en fonction de la configuration. Il faut noter que la limitation à  $10^{-16}$  correspond à la précision machine liée à l'utilisation de la double précision pour les variables. Cette précision ne permet pas d'effectuer d'opérations entre des éléments avec de tels ordres de grandeur.

Aspect ratio $H/R$ de la géométrie projetée	Rayon de la frontière intérieure minimal	Nombre de cercles supplémentaires
12	$10^{-7}$	0
21	$10^{-13}$	1
24	$10^{-16}$	2

Tableau 2.1 – Ordre de grandeur des caractéristiques des géométries pour les nombres de frontières intérieures

De la même façon que pour les géométries avec un aspect ratio faible, l'étape de création des

c. Les valeurs données ici peuvent varier en fonction de différents paramètres telle que la taille des éléments de maillage par exemple.

cercles intermédiaires est effectuée avant le déplacement des frontières intérieures. Il faut ensuite répercuter le déplacement des frontières intérieures aux nœuds des cercles de la même façon que pour les nœuds du reste du maillage. Dans le cas présent, le processus se décompose en deux étapes.

La première consiste à déterminer la position des nœuds des cercles intermédiaires après le déplacement des frontières intérieures. Une triangulation de Delaunay est effectuée seulement sur les nœuds des frontières. Certes, les triangles ont la même configuration, qui pose problème pour les conditions sur les coefficients barycentriques, mais les points à déplacer ne sont plus à proximité des sommets des triangles ce qui permet d'éviter les erreurs. Le calcul des nouvelles coordonnées est mené aisément et détermine la position des nœuds des cercles intermédiaires suite au mouvement des frontières. La deuxième étape fonctionne de manière similaire sauf que cette fois la triangulation est effectuée sur les frontières en prenant aussi les nœuds des cercles intermédiaires. Les triangles formés ont donc des proportions plus adéquates pour effectuer le déplacement de l'ensemble des points.

### 2.2.4 Identification des nœuds particuliers

Pour effectuer la comparaison des différents disques superposables, il est indispensable de fixer un référentiel commun à l'ensemble des disques. Dans notre cas, c'est le point  $V_s$  qui constitue cet invariant (cf. partie 2.1.2). Toutefois, l'étape de maillage n'est pas maîtrisée ainsi que la numérotation des nœuds. Lors du processus de détection de frontière présenté à la partie 2.1.1, il n'y a pas de prise en compte de paramètre géométrique ce qui impose de fixer le nœud correspondant au point  $V_s$ . De plus, le sens de parcours de la frontière pour le calcul du système (2.3) doit aussi être imposé. Sans cela, des retournements de disque sont observés et il n'est pas possible de les comparer entre eux.

Si on récapitule, il faut imposer deux conditions :

- un point invariant  $V_s$  qui est le point initial de la projection harmonique ;
- le sens de parcours de la frontière qui doit aussi être invariant quelle que soit la variation de la géométrie.

Le fait d'imposer ces caractéristiques revient à repérer deux nœuds particuliers sur le maillage. Le premier constitue le point initial. Le deuxième nœud, consécutif au premier, impose le sens de parcours de la frontière. Il faut noter que ces points peuvent se situer sur une partie déformable de la géométrie. Les points sont donc invariants de par leur position sur la géométrie et non pas au point de vue de leurs coordonnées. Il est difficilement envisageable de repérer ces points sur le maillage sans référence à des éléments géométriques de la CAO. C'est pourquoi, deux composants sont ajoutés en amont dans la CAO. Un premier point sur la géométrie permet de définir l'emplacement du point initial de la projection. Ainsi, le nœud de la frontière le plus proche est le point  $V_s$ . La définition d'un segment impose le sens de parcours de la frontière pour repérer le point adjacent au point  $V_s$ .

Comme énoncé à la fin de la partie 2.2.1, il est nécessaire d'identifier les frontières lorsque leur nombre est supérieur à deux. En effet, si elles sont plusieurs, il faut superposer les frontières intérieures respectives et aucune information du maillage ne permet de s'en assurer. Pour cela, un nœud du maillage est repéré sur chacune d'elles à l'aide d'un point placé pendant la création

de la CAO. Le nœud le plus proche de chaque élément est identifié et les frontières respectives entre les formes sont identifiées.

Certes, l'utilisation de ces éléments et surtout leur mise en place dans la CAO ajoutent une étape à l'utilisateur mais ceux-ci peuvent être ajoutés à la fin de la conception du modèle CAO pendant la préparation de l'optimisation. Cela représente un avantage très important comparé au plan de découpe requis jusqu'à présent dans le cas des multifrontières. En effet, il était nécessaire de mettre en place ce plan de découpe au plus tôt dans la conception de la géométrie ce qui ajoutait une complexité importante.

### 2.3 Amélioration de l'interpolation

La comparaison des disques dans le but de calculer les valeurs du maillage de référence dans l'espace 3D est effectuée à l'aide d'une interpolation. Jusqu'ici, nous utilisons l'interpolation cubique bidimensionnelle de Shepard [Shepard 1968]. Celle-ci permet d'assigner une valeur à un espace inconnu à partir d'un ensemble de points connus. Les coordonnées 3D d'un point quelconque du disque sont donc déterminées en utilisant les autres points du disque. Toutefois, nous rencontrons des limites avec cette interpolation. Elles sont principalement dues au changement de métrique lors du passage de l'espace 3D à l'espace 2D et la reconstruction des frontières.

#### 2.3.1 Interpolation par connectivité

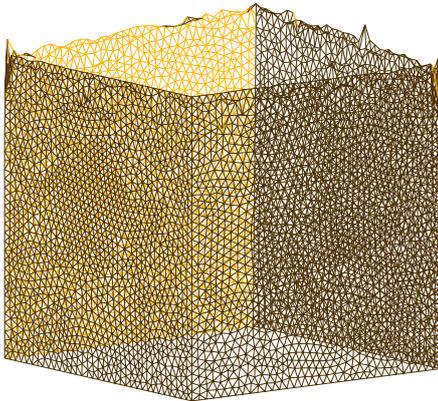
L'interpolation cubique bidimensionnelle de Shepard [Shepard 1968] se base sur un nuage de points. Elle ne prend pas en compte la connectivité du maillage. Quand le maillage est projeté sur le disque par la projection harmonique, les valeurs des angles des triangles ne sont pas conservées<sup>d</sup>, ce qui a pour effet de déformer les éléments. Il arrive donc qu'un point à interpoler soit plus proche du sommet d'un élément dont il ne fait pas partie. L'interpolation est donc influencée par ce point à tort.

Ce que nous proposons ici, c'est de prendre en compte la connectivité. En effet, nous calculons la nouvelle position d'un point en interpolant uniquement avec les sommets de l'élément dans lequel il est contenu. Nous calculons les coefficients barycentriques de chaque nœud par rapport aux sommets du triangle 2D dans lequel il est contenu et nous appliquons ces mêmes coefficients aux coordonnées 3D des mêmes sommets. Cela nous donne les coordonnées de chacun des nœuds dans l'espace 3D.

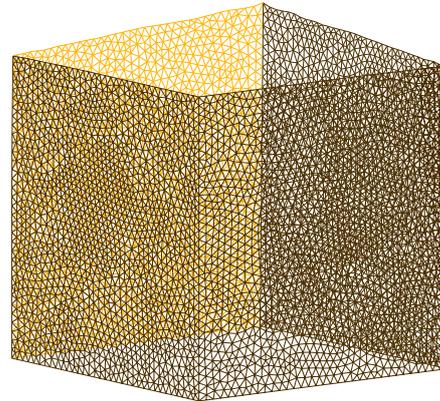
**Remarque 2.3.1** *La problématique de l'aspect ratio élevé présentée dans la partie 2.2.3 est aussi limitante pour l'interpolation. La concentration des points obtenue suite à la projection des géométries se traduit par des éléments de petite taille. Qu'elles soient monofrontières ou multifrontières, lorsque les géométries ayant un aspect ratio élevé sont projetées, la taille des éléments obtenus peut dépasser la précision machine liée à l'utilisation de variables de type double. Dans le cas de ces géométries, il est nécessaire de mettre en place des moyens pour contourner cette limitation comme par exemple l'utilisation de la quadruple précision.*

---

d. La projection harmonique n'est en général pas conforme et ne conserve pas les angles.



2.21.a – Interpolation des disques par la méthode de Shepard



2.21.b – Interpolation des disques par connectivité

FIGURE 2.21 – Amélioration de l'interpolation des disques

Ce changement d'interpolation permet d'avoir une meilleure information et de ne pas être influencé par les nœuds qui seraient placés plus près lors de la projection harmonique. La figure 2.21.a représente le maillage de la forme initiale auquel la déformation calculée par l'interpolation de Shepard est appliquée. La même opération est effectuée pour la figure 2.21.b mais avec la sensibilité calculée avec l'interpolation par connectivité. La différence est flagrante entre les deux figures. Toutefois, la frontière ne correspond pas encore tout à fait au maillage de la forme initiale.

### 2.3.2 Interpolation aux frontières

Travaillant sur un disque, nous nous heurtons à la forme arrondie de la frontière de la surface de projection. La figure 2.22 présente ce qu'il se passe sur la frontière. Les points de forme rectangulaire sont les points à interpoler avec les informations données par les points ronds. Toutefois, ces points rectangulaires sont à l'extérieur du triangle formé par les points ronds du fait de la frontière du disque. L'extrapolation effectuée induit des erreurs dans le calcul de sensibilité comme on peut le voir sur la figure 2.21.b. Nous proposons une méthode permettant de calculer plus finement la déformation aux frontières.

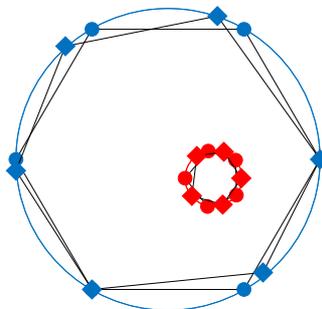


FIGURE 2.22 – Illustration de l'interpolation sur la frontière

En effectuant un zoom sur le maillage obtenu suite à l'interpolation par connectivité, on remarque que les arêtes ne sont pas définies correctement (cf. figure 2.23).

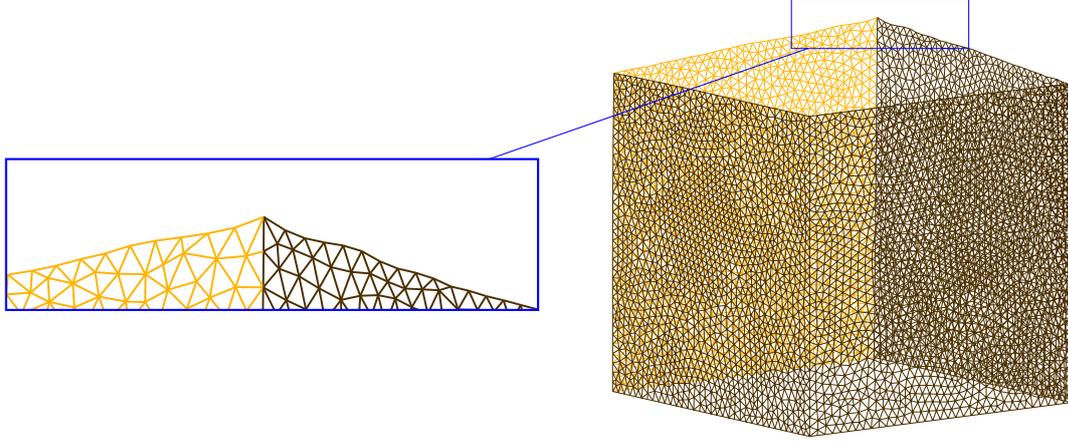


FIGURE 2.23 – Amélioration de l'interpolation des disques

Pour éviter cet effet, l'interpolation est effectuée en suivant la frontière. En effet, un point situé sur le bord du disque entre deux points est considéré être placé sur le segment reliant ces deux points en respectant le rapport de distance.

Les nœuds  $V_i$  d'une frontière doivent être interpolés avec les nœuds  $V'_j$ . On utilise les coordonnées angulaires pour effectuer cette interpolation. Si la coordonnée angulaire est comprise entre deux coordonnées angulaires des points  $V'_j$ , on effectue les étapes suivantes. Tout d'abord, la coordonnée angulaire  $\theta$  de chacun des points de la frontière est calculée en fonction du barycentre correspondant. Pour la frontière extérieure, le barycentre est le centre du cercle unité et pour les frontières intérieures, le barycentre calculé dans la partie 2.2.1 ( $x_{G_0}$ ) est repris. Avec cette coordonnée angulaire, les points  $V'_j$  et  $V'_{j+1}$  qui encadrent le point  $V_i$  sont identifiés. Le calcul du rapport entre les angles permet de déterminer la position de  $V_i$  par rapport à  $V'_j$  et  $V'_{j+1}$  :

$$\rho_\theta = \frac{\theta_i - \theta'_j}{\theta'_{j+1} - \theta'_j} \quad (2.53)$$

Ce qui conduit au calcul des coordonnées :

$$\begin{cases} u_{V_i} = \rho_\theta u_{V'_j} + (1 - \rho_\theta) u_{V'_{j+1}} \\ v_{V_i} = \rho_\theta v_{V'_j} + (1 - \rho_\theta) v_{V'_{j+1}} \end{cases} \quad (2.54)$$

Cette interpolation évite les erreurs liées à l'extrapolation et donne ainsi une meilleure définition au niveau de la frontière. Les arêtes sont plus nettes sur la figure 2.24 qui représente le maillage reconstruit à partir du maillage initial auquel la sensibilité calculée est appliquée (cf. figure 2.25).

Il faut noter que la taille de cellule a une grande importance dans la précision du calcul de la différence finie. En effet, il est nécessaire d'utiliser une taille de maille permettant une bonne

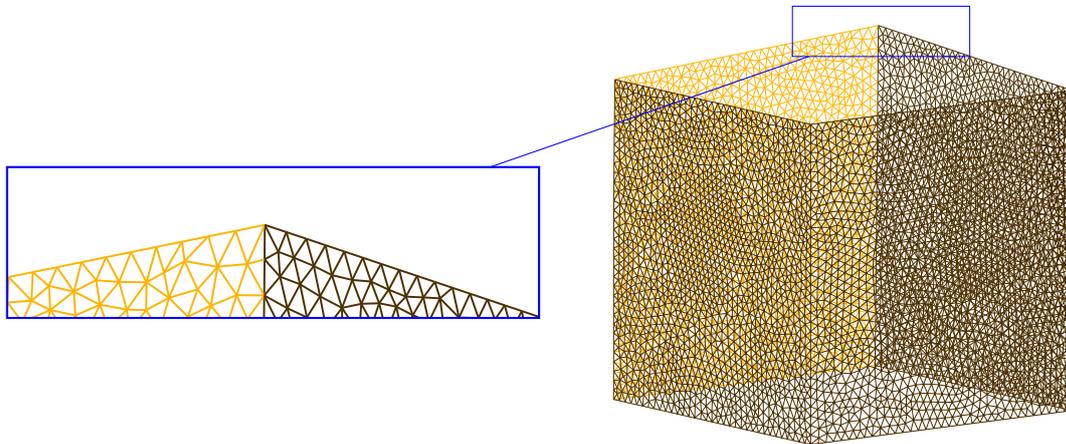


FIGURE 2.24 – Amélioration de l'interpolation des disques

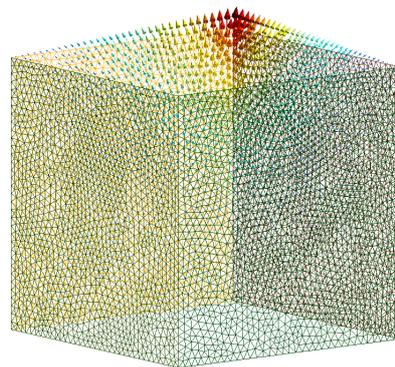


FIGURE 2.25 – Sensibilité calculée avec les 2 méthodes d'interpolation

définition de la surface pour détecter la variation des paramètres mais aussi pour favoriser une bonne représentativité des arêtes vives et des lignes de design (« *feature lines* »).

## 2.4 Synthèse et discussion des résultats

Les développements effectués dans ce chapitre ont pour but d'étendre la méthode de calcul de sensibilité par projection harmonique à des géométries plus complexes tout en améliorant la qualité de l'évaluation vis-à-vis d'un paramètre. Pour cela, la gestion de la projection des géométries ayant plusieurs frontières sans opération supplémentaire facilite l'utilisation de la méthode de calcul de sensibilité. En effet, il est possible de construire le modèle CAO sans qu'il soit nécessaire de mettre en place des opérations de découpage de la forme pour n'avoir que des surfaces homéomorphes au disque unité. La superposition de chacun des disques sur lesquels les géométries sont projetées, et cela quel que soit le nombre de frontières contenues, supprime cette contrainte.

Dans cette partie, nous avons utilisé les maillages des formes CAO pour effectuer notre calcul de sensibilité. [Dannenhoffer III 2015], quant à lui, présente une méthode de sensibilité basée sur la définition CAO. Cette méthode est appliquée sur le modèle d'une vis représentée par la figure 2.26 dont la construction est détaillée dans cet article. La CAO est régie par différents paramètres comme par exemple l'épaisseur de la tête ou la longueur de la tige. Extraite de [Dannenhoffer III 2015], la figure 2.27 présente la sensibilité de la vis par rapport à la variation de l'épaisseur de la tête.

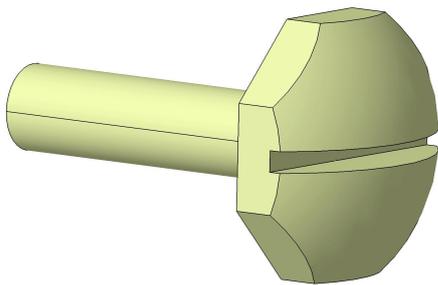


FIGURE 2.26 – Modèle CAO

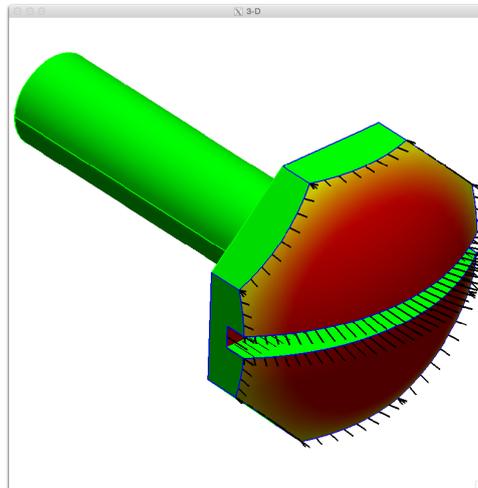
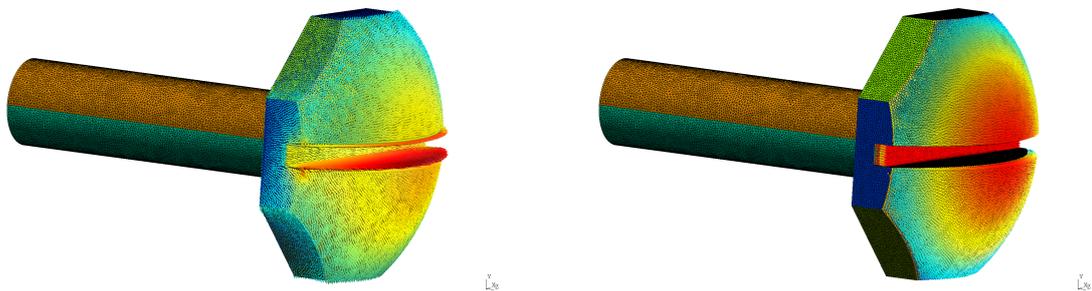


FIGURE 2.27 – Sensibilité du paramètre  $\partial \vec{x} / \partial (Thead)$  (épaisseur de la tête)  
Source : [Dannenhoffer III 2015]

À titre de comparaison, nous avons effectué un calcul de sensibilité sur le même paramètre. Les résultats obtenus sont présentés par la figure 2.28.a. On remarque sur cette figure de nombreux déplacements tangentiels à la surface. Ceux-ci viennent de l'utilisation de maillages pour le calcul de la sensibilité. Lors de la reconstruction de l'isoconnectivité, le maillage de la forme initiale

se déploie sur la forme déformée. On obtient donc un glissement des nœuds sur la surface de la géométrie. Comme on peut le voir sur la figure 2.28.b, pour faciliter la comparaison des résultats, nous avons extrait le gradient suivant la normale au barycentre (centroïde) de l'élément, ce qui révèle une sensibilité proche de celle présentée par la figure 2.27. Toutefois, la sensibilité utilisée dans notre méthode est celle de la figure 2.28.a. Ainsi, lors du calcul (1.24), elle est multipliée à celle de la fonction objectif par rapport à la forme et ce glissement sera implicitement filtré. En effet, [Sokolowski 1992] montre que les dérivées de forme dépendent de la composante normale du champ de vecteur vitesse sur la frontière. Dans notre cas, le gradient donné par le solveur adjoint est donc porté par la normale à la paroi. Il précise que cette propriété est cruciale pour l'optimisation de forme.



2.28.a – Sensibilité obtenue par la méthode de projection harmonique

2.28.b – Extraction de la sensibilité suivant les normales aux barycentres (centroïdes)

FIGURE 2.28 – Visualisation de la sensibilité

[Dannenhoffer III 2015] présente une méthode de calcul de sensibilité basée directement sur les géométries CAO. Cependant, comme décrit dans cet article, pour sa mise en oeuvre, il est nécessaire de différentier un solveur géométrique et de construire la CAO pour s'assurer qu'elle utilise les opérations différentiées. Cela représente une contrainte importante pour généraliser son utilisation. En particulier, la méthode présentée s'appuie sur un code open source peu répandu dans l'industrie. Dans le chapitre 3, nous présentons une méthode basée elle aussi sur la définition CAO tout en étant compatible avec les outils utilisés par les industriels.

Bibliographie du 2<sup>e</sup> chapitre

- [Dannenhoffer III 2015] J.F. Dannenhoffer III et R. Haimes. *Design Sensitivity Calculations Directly on CAD-based Geometry*. In 53rd AIAA Aerospace Sciences Meeting, page 1370, 2015. (cité pages 23, 65 et 66)
- [Delaunay 1934] Boris Delaunay. *Sur la sphere vide*. Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk, vol. 7, no. 793-800, pages 1–2, 1934. (cité page 56)
- [Floater 2002] Michael S. Floater et Kai Hormann. *Surface Parameterization : a Tutorial and Survey*. Article, 2002. (cité page 34)
- [Froment 2014] Pierre Froment. *Optimisation de formes paramétriques en grande dimension*. PhD thesis, École Centrale de Lyon, 2014. (cité pages 10, 12, 15, 27, 28, 29, 34, 45, 47 et 97)
- [Gu 2008] X.D. Gu et S.T. Yau. Computational conformal geometry, volume 3 of *Advanced lectures in mathematics*. International Press, 2008. (cité pages 27, 34, 54 et 133)
- [Hamilton 1982] Richard S. Hamilton. *Three-manifolds with positive Ricci curvature*. J. Differential Geom., vol. 17, no. 2, pages 255–306, 1982. (cité page 54)
- [Kripac 1997] Jiri Kripac. *A mechanism for persistently naming topological entities in history-based parametric solid models*. Computer-Aided Design, vol. 29, no. 2, pages 113 – 122, 1997. (cité pages 24, 34 et 76)
- [Lévy 2000] Bruno Lévy et J-L Mallet. *Paramétrisation des surfaces triangulées*. Revue internationale de CFAO et d’informatique graphique, vol. 15, no. 1, pages 25–42, 2000. (cité page 34)
- [Remacle 2010] J.-F. Remacle, C. Geuzaine, G. Compère et E. Marchandise. *High-quality surface remeshing using harmonic maps*. International journal for numerical methods in engineering, vol. 83, pages 403 – 425, 2010. (cité pages 27, 28 et 34)
- [Shepard 1968] Donald Shepard. *A Two-dimensional Interpolation Function for Irregularly-spaced Data*. In Proceedings of the 1968 23rd ACM National Conference, ACM '68, pages 517–524, New York, NY, USA, 1968. ACM. (cité page 61)
- [Sokolowski 1992] Jan Sokolowski et Jean-Paul Zolesio. Introduction to shape optimization : shape sensitivity analysis. Springer series in computational mathematics. Springer, New York, Heidelberg, Paris, 1992. (cité page 66)



# Calcul de la sensibilité d'une forme sur les géométries CAO

---

<b>3.1</b>	<b>Rappel des notions fondamentales des courbes et des surfaces . . .</b>	<b>70</b>
3.1.1	Les courbes . . . . .	70
3.1.1.1	Courbes de Bézier . . . . .	70
3.1.1.2	Courbes rationnelles . . . . .	71
3.1.1.3	Courbes B-Splines . . . . .	72
3.1.1.4	Courbes NURBS . . . . .	73
3.1.2	Les surfaces . . . . .	73
3.1.2.1	Surfaces de Bézier . . . . .	73
3.1.2.2	Surfaces NURBS . . . . .	75
<b>3.2</b>	<b>Calcul semi-analytique . . . . .</b>	<b>76</b>
3.2.1	Utilisation des équations paramétriques et des données STEP . . . . .	76
3.2.2	Calcul de la sensibilité d'un point de coordonnées $(u,v)$ . . . . .	78
3.2.3	Calcul des sensibilités sur un maillage de calcul . . . . .	80
<b>3.3</b>	<b>Comparaison des méthodes de calcul de sensibilité . . . . .</b>	<b>81</b>
3.3.1	Sur un patch . . . . .	81
3.3.2	Dans une optimisation en CFD . . . . .	86
3.3.2.1	Calcul physique . . . . .	86
3.3.2.2	Maillage . . . . .	87
3.3.2.3	Post-traitement et résultats . . . . .	88
3.3.3	Bilan sur la comparaison des deux méthodes . . . . .	92
<b>3.4</b>	<b>Synthèse . . . . .</b>	<b>92</b>
	<b>Bibliographie du 3<sup>e</sup> chapitre . . . . .</b>	<b>94</b>

---

Dans le chapitre précédent, le calcul de sensibilité est réalisé en s'appuyant sur la discrétisation de chaque forme générée. L'absence d'isoconnectivité entre les différents maillages ne permet pas d'évaluer le déplacement de chacun des nœuds directement. La méthode consiste donc à interpoler sur un domaine commun (un disque unité dans  $\mathbb{R}^2$ ) les images obtenues de deux formes 3D par la projection harmonique. L'utilisation de cette méthode de projection ainsi que l'interpolation entre les coordonnées des nœuds conduisent à projeter les nœuds du maillage de la première forme sur la deuxième et ainsi évaluer le déplacement de chacun des points. L'utilisation de maillages, qui sont des discrétisations de formes géométriques, est susceptible d'induire des imprécisions en fonction de différents paramètres comme la taille de maille, les tailles de géométries à projeter... Comme nous l'avons vu dans l'état de l'art sur les méthodes de sensibilité (cf. partie 1.5), différents auteurs proposent d'utiliser la définition CAO de la géométrie. Cette définition permet de connaître en tout point de la géométrie les coordonnées  $(x,y,z)$  et ainsi éviter les approximations dues à la discrétisation du maillage. Actuellement, les processus mis en place dans l'industrie utilisent des solveurs géométriques qui ne permettent pas de connaître la définition analytique de la géométrie. La différentiation des équations par rapport aux paramètres définis par l'utilisateur, trop intrusives vis-à-vis de ces logiciels, n'est donc pas envisageable. Le recours à des formats d'échange neutres donne accès aux équations de la définition CAO mais pas aux informations du lien qu'elles ont avec les paramètres de l'utilisateur. Il est donc préférable d'effectuer un calcul par différences finies entre deux formes mais, cette fois, en utilisant la définition exacte des formes. Avant de présenter la méthode que nous proposons, nous rappelons les notions des courbes et des surfaces nécessaires à la bonne compréhension de ce chapitre.

### 3.1 Rappel des notions fondamentales des courbes et des surfaces

Dans cette partie, nous allons nous intéresser aux différents types de courbes et de surfaces utilisées pour la modélisation géométrique. Nous allons aussi voir les différentes notions nécessaires à la compréhension de leur fonctionnement. Pour plus de précisions sur ces notions, le lecteur peut se reporter aux travaux de [Piegl 1995, Foretnik 2010].

#### 3.1.1 Les courbes

##### 3.1.1.1 Courbes de Bézier

Les courbes de Bézier sont des courbes polynomiales. Chaque point d'une courbe de Bézier de degré  $n$  est déterminé par une somme pondérée donnée par :

$$C(u) = \sum_{i=0}^n B_{i,n}(u)P_i \quad 0 \leq u \leq 1 \quad (3.1)$$

où le terme  $B_{i,n}(u)$  représente le polynôme de Bernstein de degré  $n$ , donné par :

$$B_{i,n}(u) = \frac{n!}{i!(n-i)!} u^i (1-u)^{n-i} \quad (3.2)$$

Les termes  $P_i$  sont les coefficients géométriques et sont couramment appelés *points de contrôle*.

### 3.1. Rappel des notions fondamentales des courbes et des surfaces

Les courbes de Bézier restent limitées en ce qui concerne le nombre de formes caractérisables. En effet, il est souvent nécessaire de monter en degré pour obtenir des formes complexes. Même si les courbes de Bézier peuvent être déterminées par une moyenne des points de contrôle (avec leur possible pondération), le contrôle de la courbe n'est pas assez local.

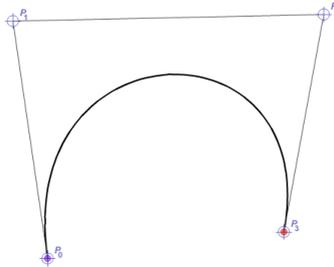


FIGURE 3.1 – Courbe de Bézier de degré 3  
Source : [Foretnik 2010]

#### Degré des courbes

Le degré d'une courbe se réfère au plus grand exposant du polynôme utilisé pour définir la courbe de Bézier. Plus le degré est élevé, plus il y a de point de contrôle et plus la complexité de la forme est élevée comme on peut le voir sur les figures 3.2. Toutefois, il n'est pas évident d'utiliser des courbes avec un degré élevé car chaque point de contrôle influence l'ensemble de la courbe et le contrôle local est difficile.

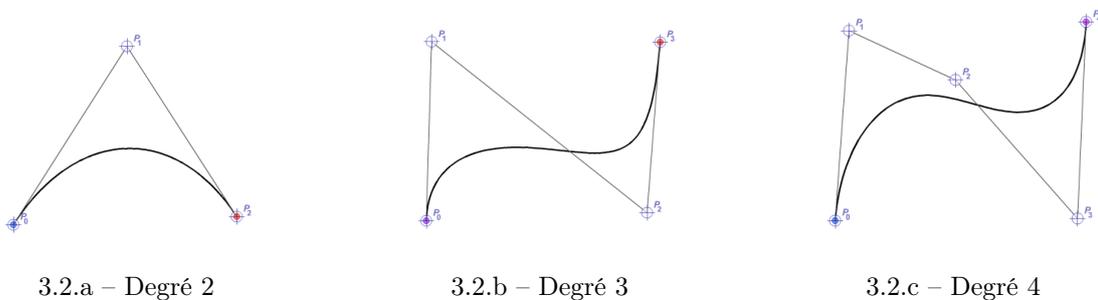


FIGURE 3.2 – Courbes de Bézier de différents degrés  
Source : [Foretnik 2010]

Dans la pratique, le degré des courbes est souvent limité à 3 pour faciliter la manipulation des courbes et leur précision. Le recours à des assemblages de courbes de Bézier permet d'augmenter considérablement la complexité des formes tout en garantissant un contrôle local suffisant. Avant cela, l'utilisation d'une pondération sur les courbes de Bézier étend leur champ d'action.

#### 3.1.1.2 Courbes rationnelles

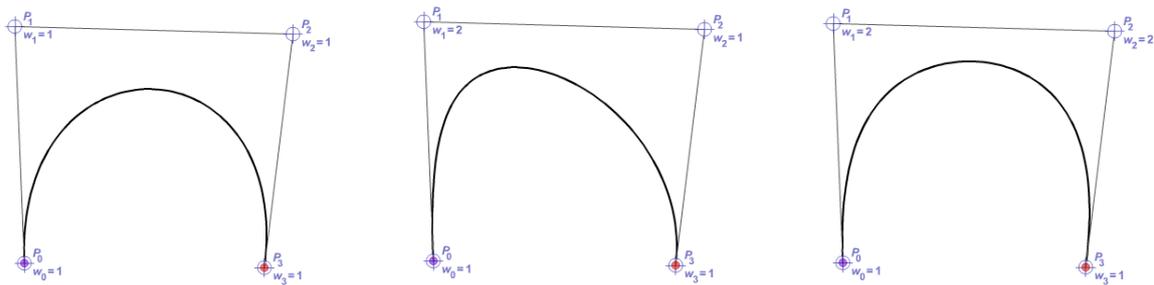
Dans le cas des courbes rationnelles, les points de contrôle ne sont pas seulement caractérisés par les coordonnées mais aussi par un poids. Ce poids va définir comment le point de contrôle

attire la courbe. Pour exemple, une courbe de Bézier rationnelle est donnée par :

$$C(u) = \sum_{i=0}^n R_{i,n}(u)P_i \quad 0 \leq u \leq 1 \tag{3.3}$$

où  $R_{i,n}(u)$  est une fonction rationnelle donnée par :

$$R_{i,n}(u) = \frac{B_{i,n}(u)w_i}{\sum_{j=0}^n B_{j,n}(u)w_j} \tag{3.4}$$



3.3.a – Courbe de Bézier d'ordre 3 avec les poids identiques    3.3.b – Changement du poids de P1    3.3.c – Changement du poids de P1 et de P2

FIGURE 3.3 – Courbes de Bézier rationnelles de degré 3

Source : [Foretnik 2010]

Ici, ce n'est pas la valeur des poids qui est prépondérante mais le rapport entre chacun des poids.

### 3.1.1.3 Courbes B-Splines

Les B-Splines (pour basis splines) sont la généralisation des courbes de Bézier. Ce sont des fonctions définies par morceaux et elles permettent d'obtenir des formes plus complexes. Afin de déterminer où et comment les points de contrôle vont affecter la courbe, les B-Splines s'appuient sur une séquence de valeurs appelés nœuds. Pour une B-Spline de degré  $n$  contenant  $p$  points de contrôle, le nombre de nœuds  $m$  est donné par :

$$m = p + n + 1 \tag{3.5}$$

La valeur de chacun des  $m$  nœuds est stockée dans un vecteur nodal  $T$ . Sur les figures 3.4, on observe les nœuds sur les courbes ainsi que la variation des courbes en fonction du vecteur nodal. Les extrémités du vecteur nodal traduisent où commence et se termine la courbe. Le cas le plus répandu est celui dans lequel la courbe commence au premier point de contrôle et se termine au dernier. Dans ce cas, les  $n + 1$  premiers et les  $n + 1$  derniers termes du vecteur nodal sont multiples et respectivement égaux à 0 et 1 :

$$T = (\underbrace{0, \dots, 0}_{n+1}, t_{n+1}, t_{n+2}, \dots, t_{m-n-1}, t_{m-n}, \underbrace{1, \dots, 1}_{n+1}) \tag{3.6}$$

### 3.1. Rappel des notions fondamentales des courbes et des surfaces

Il est aussi possible que la courbe ne débute pas sur un point de contrôle ce qui impacte en conséquence le vecteur nodal.

Les termes intermédiaires sont une suite croissante et traduisent le caractère de la courbe. Si ces termes sont séparés par des intervalles réguliers la courbe obtenue est dite uniforme (cf. figures 3.4) sinon on parle de courbes non-uniformes. L'ensemble du vecteur de nœuds permet de calculer les termes  $N_j$ , définis par la formule doublement récursive de Cox-De Boor :

$$\begin{cases} N_j^0(t) = \begin{cases} 1 & \text{si } t_j \leq t < t_{j+1} \\ 0 & \text{sinon} \end{cases} \\ N_j^d(t) = \frac{t-t_j}{t_{j+d}-t_j} N_j^{d-1}(t) + \frac{t_{j+d+1}-t}{t_{j+d+1}-t_{j+1}} N_{j+1}^{d-1}(t) \end{cases} \quad (3.7)$$

Il est ensuite possible de calculer chacun des points de la courbe par l'équation :

$$C(t) = \sum_{i=0}^{m-n-1} P_i N_i^n(t) \quad (3.8)$$

Il faut aussi noter que la modification d'un des points de contrôle n'influence que localement l'allure de la courbe, cela permet d'avoir un contrôle plus précis de la courbe souhaitée.

#### 3.1.1.4 Courbes NURBS

Nous venons de voir les caractéristiques des B-Splines avec les notions de nœuds. Si nous ajoutons à celles-ci la notion de poids, elle deviennent des B-Splines rationnelles, plus communément appelées NURBS (non-uniform rational basis splines). Elles permettent de représenter un très grand nombre de forme, de complexité différentes.

$$C(t) = \frac{\sum_{i=0}^{m-n-1} \omega_i P_i N_i^n(t)}{\sum_{i=0}^{m-n-1} \omega_i N_i^n(t)} \quad (3.9)$$

### 3.1.2 Les surfaces

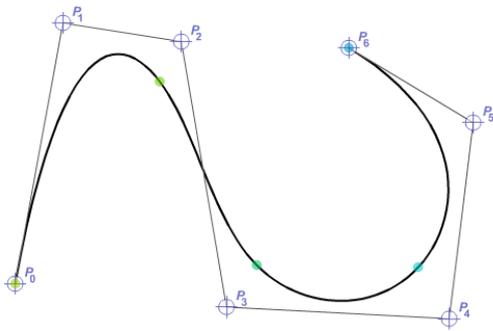
L'ensemble des courbes présentées dans la partie précédente permettent de construire des surfaces. En effet, la construction des surfaces est similaire à celle des courbes car elles sont construites suivant deux paramètres qui ont individuellement le même comportement d'une courbe. Nous retrouvons donc les différentes notions qui caractérisent les courbes appliquées aux termes  $(u,v)$ .

#### 3.1.2.1 Surfaces de Bézier

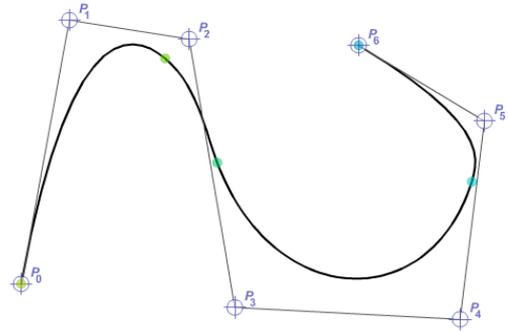
Chaque point d'une surface de Bézier de degré  $(n,m)$  est déterminé par une somme pondérée donnée par :

$$S(u,v) = \sum_{i=0}^n \sum_{j=0}^m B_{i,n}(u) B_{j,m}(v) P_{i,j} \quad (u,v) \in [0,1]^2 \quad (3.10)$$

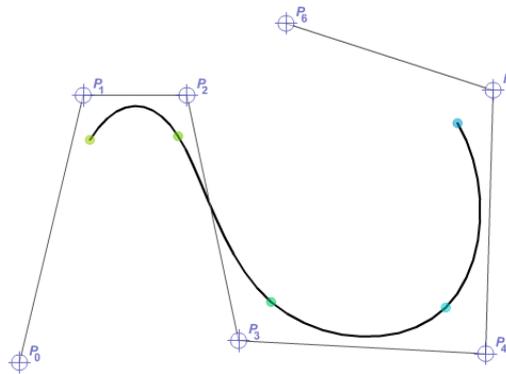
La figure 3.6 est une surface de Bézier de degré  $5 \times 5$ .



3.4.a – Courbe B-Spline uniforme  
 $T = \{0,0,0,0,0.25,0.5,0.75,1,1,1,1\}$



3.4.b – Courbe B-Spline non-uniforme  
 $T = \{0,0,0,0,0.15,0.3,0.85,1,1,1,1\}$



3.4.c – Courbe B-Spline uniforme non fixée  
 $T = \{0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1\}$

FIGURE 3.4 – Différentes courbes B-Spline de degré 3 et avec 7 points de contrôle ( $n = 3$  et  $p = 7$  donc  $m = 11$ ) – Source : [Foretnik 2010]

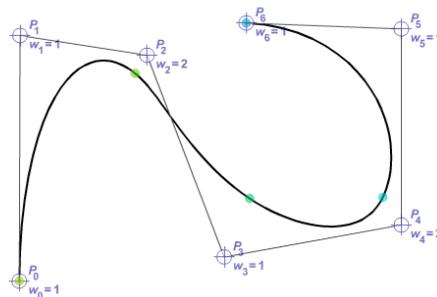


FIGURE 3.5 – Courbe NURBS (modification du poids de  $P_2$  et  $P_4$ )  
 Source : [Foretnik 2010]

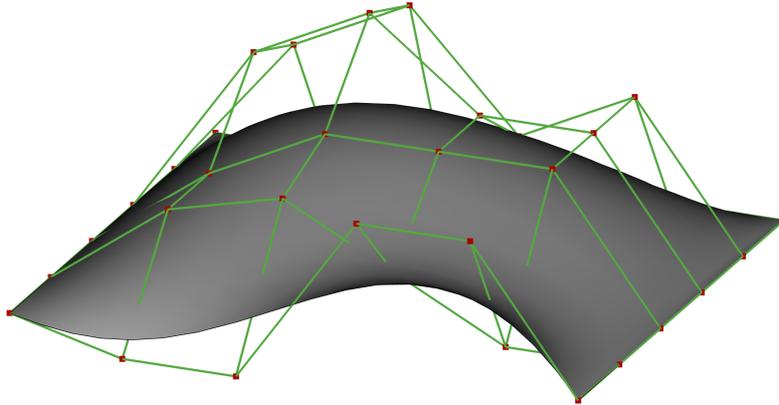


FIGURE 3.6 – Surface de Bézier de degré 5×5

### 3.1.2.2 Surfaces NURBS

Chaque point d'une surface **NURBS** de degré  $n$  est déterminé par une somme pondérée donnée par :

$$S(u,v) = \frac{\sum_{i=0}^{m_u-n_u-1} \sum_{j=0}^{m_v-n_v-1} \omega_{i,j} N_i^{n_u}(u) N_j^{n_v}(v) P_{i,j}}{\sum_{i=0}^{m_u-n_u-1} \sum_{j=0}^{m_v-n_v-1} \omega_{i,j} N_i^{n_u}(u) N_j^{n_v}(v)} \quad (u,v) \in [0,1]^2 \quad (3.11)$$

Les surfaces B-Splines sont un cas particulier des surfaces **NURBS** pour lesquelles les poids appliquées aux pôles sont égaux.

### Récapitulatif

Les principales caractéristiques des courbes et des surfaces présentées ci-dessus sont rassemblées dans le tableau 3.1.

	Splines	Bézier	Bézier rationnelles	B-Splines	NURBS
Points de contrôle/Pôles	-	X	X	X	X
Poids	-	-	X	-	X
Noeuds	-	-	-	X	X

Tableau 3.1 – Tableau récapitulatif des types de courbes et surfaces

Ces différents types de courbes et de surfaces sont utilisés dans les logiciels de **CAO** sans que l'on s'en rende compte. En effet, les solveurs géométriques manipulent ces systèmes d'équations en fonction des opérations effectuées ou des contraintes imposées à la géométrie. La plupart des logiciels ne donnent pas accès à leur fonctionnement interne, il est donc impossible de connaître les informations utiles pour calculer la sensibilité de la forme. Pour calculer cette sensibilité, il faut donc être capable de retrouver ces informations. Certains fichiers d'échange permettent de retrouver aisément ces informations mais les relations avec les paramètres ne sont pas explicites, il faut donc adapter la méthode de sensibilité.

### 3.2 Calcul semi-analytique

L'utilisation des formats d'échange ne permet donc pas d'effectuer des calculs analytiques avec ces informations, c'est pour cela que l'on parle de calcul semi-analytique car on utilise la définition analytique des surfaces dans un processus s'approchant de la différence finie en comparant deux formes entre elles.

#### 3.2.1 Utilisation des équations paramétriques et des données STEP

Pour effectuer le calcul de sensibilité d'une forme CAO, nous utilisons sa définition analytique. Nous prenons une hypothèse assez forte : nous considérons que notre forme est composée d'un seul patch quelle que soit la variation de paramètre. Cela évite d'être confronté à des changements du nombre de patches ou de leur identification [Kripac 1997] qui augmentent de façon significative la complexité du calcul de sensibilité. Une vérification est donc effectuée pour s'assurer qu'aucun patch n'est apparu suite à la variation de paramètre.

Le calcul de sensibilité est effectué en comparant deux formes suite à la variation d'un paramètre. Contrairement à la méthode présentée dans le chapitre 2 basée sur les maillages des géométries, la méthode consiste à utiliser les fichiers STEP dans lesquels on retrouve de nombreuses informations permettant de connaître la forme générée en connaissant différents paramètres comme les coordonnées des points de contrôle, leur pondération... Ces informations deviennent les paramètres des équations présentées dans la partie 3.1. La sensibilité d'une forme par rapport à l'autre est obtenue en effectuant la comparaison des fonctions obtenues pour chaque forme.

Toutefois, en pratique, une variation de la valeur d'un paramètre ne modifie pas uniquement la position des points de contrôle mais possiblement leur nombre donc le degré de la surface ainsi que le type de celle-ci avec l'apparition de nœuds ou de pondération des points de contrôle (dans le cas des surfaces NURBS). Les extraits de code 3.1 et 3.2 correspondent respectivement aux surfaces de la figure 3.7.



FIGURE 3.7 – Déformation d'un patch

Le début de la première ligne de chaque fichier (en rouge) nous indique que la surface est définie par une surface B-Spline (cf. partie 3.1.1.3). On observe aussi que les définitions gardent des similarités mais qu'elles ne contiennent pas le même nombre d'argument. Les lignes 338 à 352 de l'extrait de code 3.1 et les lignes 231-242 de l'extrait de code 3.2 définissent les surfaces et les numéros précédés du symbole # sont les points définis plus tôt dans les fichiers. On remarque que l'extrait de code 3.1 contient 15 lignes de 15 points chacune alors que l'extrait de code 3.2 ne contient que 12 lignes de 12 points. Généralement, on considère que les  $(u,v)$  des patches varient entre 0 et 1, ce qui n'est pas vrai dans le cas de la définition des patches. En effet, les bornes peuvent être différentes comme on peut le voir avec les valeurs en bleu aux lignes 354-355 de

Extrait de code 3.1 – Définition de la surface pour la déformation 1 (figure 3.7.a)

```

337 #53=B_SPLINE_SURFACE_WITH_KNOTS(' ',5,5,
338 ((#54,#55,#56,#57,#58,#59,#60,#61,#62,#63,#64,#65,#66,#67,#68),
339 (#69,#70,#71,#72,#73,#74,#75,#76,#77,#78,#79,#80,#81,#82,#83),
340 (#84,#85,#86,#87,#88,#89,#90,#91,#92,#93,#94,#95,#96,#97,#98),
341 (#99,#100,#101,#102,#103,#104,#105,#106,#107,#108,#109,#110,#111,#112,#113),
342 (#114,#115,#116,#117,#118,#119,#120,#121,#122,#123,#124,#125,#126,#127,#128),
343 (#129,#130,#131,#132,#133,#134,#135,#136,#137,#138,#139,#140,#141,#142,#143),
344 (#144,#145,#146,#147,#148,#149,#150,#151,#152,#153,#154,#155,#156,#157,#158),
345 (#159,#160,#161,#162,#163,#164,#165,#166,#167,#168,#169,#170,#171,#172,#173),
346 (#174,#175,#176,#177,#178,#179,#180,#181,#182,#183,#184,#185,#186,#187,#188),
347 (#189,#190,#191,#192,#193,#194,#195,#196,#197,#198,#199,#200,#201,#202,#203),
348 (#204,#205,#206,#207,#208,#209,#210,#211,#212,#213,#214,#215,#216,#217,#218),
349 (#219,#220,#221,#222,#223,#224,#225,#226,#227,#228,#229,#230,#231,#232,#233),
350 (#234,#235,#236,#237,#238,#239,#240,#241,#242,#243,#244,#245,#246,#247,#248),
351 (#249,#250,#251,#252,#253,#254,#255,#256,#257,#258,#259,#260,#261,#262,#263),
352 (#264,#265,#266,#267,#268,#269,#270,#271,#272,#273,#274,#275,#276,#277,#278)),
353 .UNSPECIFIED.,.F.,.F.,.U.,(6,3,3,3,6),(6,3,3,3,6),
354 (0.,25.1601932939,50.3203865877,75.4805798816,100.640773175),
355 (0.,25.1601932939,50.3203865877,75.4805798816,100.640773175),.UNSPECIFIED.) ;

```

Extrait de code 3.2 – Définition de la surface pour la déformation 2 (figure 3.7.b)

```

230 #53=B_SPLINE_SURFACE_WITH_KNOTS(' ',5,5,
231 ((#54,#55,#56,#57,#58,#59,#60,#61,#62,#63,#64,#65),
232 (#66,#67,#68,#69,#70,#71,#72,#73,#74,#75,#76,#77),
233 (#78,#79,#80,#81,#82,#83,#84,#85,#86,#87,#88,#89),
234 (#90,#91,#92,#93,#94,#95,#96,#97,#98,#99,#100,#101),
235 (#102,#103,#104,#105,#106,#107,#108,#109,#110,#111,#112,#113),
236 (#114,#115,#116,#117,#118,#119,#120,#121,#122,#123,#124,#125),
237 (#126,#127,#128,#129,#130,#131,#132,#133,#134,#135,#136,#137),
238 (#138,#139,#140,#141,#142,#143,#144,#145,#146,#147,#148,#149),
239 (#150,#151,#152,#153,#154,#155,#156,#157,#158,#159,#160,#161),
240 (#162,#163,#164,#165,#166,#167,#168,#169,#170,#171,#172,#173),
241 (#174,#175,#176,#177,#178,#179,#180,#181,#182,#183,#184,#185),
242 (#186,#187,#188,#189,#190,#191,#192,#193,#194,#195,#196,#197)),
243 .UNSPECIFIED.,.F.,.F.,.U.,(6,3,3,6),(6,3,3,6),
244 (-50.,-16.8000012636,11.6999983788,50.),
245 (-50.,-16.6000008583,11.6999983788,50.),.UNSPECIFIED.) ;

```

l'extrait de code 3.1 et aux lignes 244-245 de l'extrait de code 3.2. De plus, on observe des valeurs différentes pour les bornes  $(u,v)$  entre les deux définitions.

Dans le cas particulier d'un patch plan, le type de la surface contenu dans le fichier STEP est différent des autres . En effet, la définition du patch est principalement donnée par la ligne "#57=PLANE(' ',#56);". Le support du patch est donc un plan infini découpé par des éléments géométriques supplémentaires tels que des courbes données par différentes lignes du fichier STEP (non reportées ici).

Au vu de ces différences, cette solution n'est pas adaptée dans notre cas car nous ne pouvons pas nous permettre de rester cantonnés aux courbes d'un même type comme les courbes de Bézier, celles-ci n'épousant pas assez de formes et étant donné qu'il est difficile d'imposer le type de surface en sortie du logiciel CAO. Pour pallier ces problèmes, il est nécessaire de mettre en place une méthode qui ne s'appuie pas sur un système de courbe en particulier. Il est nécessaire que le processus puisse s'adapter en fonction de la sortie du logiciel de CAO.

### 3.2.2 Calcul de la sensibilité d'un point de coordonnées $(u,v)$

L'utilisation de l'aspect paramétrique des surfaces répond à la problématique liée à la généralisation de la méthode. En effet, nous savons que les surfaces sont bornées en  $u$  et en  $v$  quel que soit leur type<sup>a</sup>. Il est donc possible de comparer une surface de Bézier et une surface d'un autre type en utilisant les coordonnées  $(u,v)$  d'un point  $\tilde{\mathcal{P}}$ . Nous créons en quelque sorte une isoconnectivité entre les deux patches.

Considérons  $\mathcal{S}$ , une fonction de définition d'un patch CAO, qui, pour une valeur de  $u$  et de  $v$ , associe des valeurs dans l'espace 3D. Celle-ci est définie comme suit :

$$\begin{aligned} \mathcal{S}_\chi : \tilde{\mathcal{P}} \subset \mathbb{R}^2 &\longrightarrow \tilde{\Omega}_\chi \subset \mathbb{R}^3 \\ (u,v) &\longmapsto \begin{cases} x_\chi(u,v) \\ y_\chi(u,v) \\ z_\chi(u,v) \end{cases} \quad \chi \in \{\alpha_0, \alpha_1, \dots, \alpha_i, \dots, \alpha_p\} \end{aligned} \quad (3.12)$$

Cette opération est effectuée sur l'ensemble des fichiers STEP : le fichier STEP avec l'ensemble des paramètres à leur valeur initiale ( $\alpha_0$ ) et les fichiers STEP correspondants chacun à une variation de paramètres ( $\alpha_i$ ). Nous effectuons donc ce calcul :

$$\begin{aligned} \mathcal{S}_{\alpha_0} : \tilde{\mathcal{P}} \subset \mathbb{R}^2 &\longrightarrow \tilde{\Omega}_{\alpha_0} \subset \mathbb{R}^3 \\ (u,v) &\longmapsto \begin{cases} x_{\alpha_0}(u,v) \\ y_{\alpha_0}(u,v) \\ z_{\alpha_0}(u,v) \end{cases} \end{aligned} \quad (3.13)$$

$$\begin{aligned} \mathcal{S}_{\alpha_i} : \tilde{\mathcal{P}} \subset \mathbb{R}^2 &\longrightarrow \tilde{\Omega}_{\alpha_i} \subset \mathbb{R}^3 \\ (u,v) &\longmapsto \begin{cases} x_{\alpha_i}(u,v) \\ y_{\alpha_i}(u,v) \\ z_{\alpha_i}(u,v) \end{cases} \quad i \in \{1, \dots, p\} \end{aligned} \quad (3.14)$$

---

a. Une normalisation des coordonnées  $(u,v)$  peut être nécessaire pour prendre en compte des différences de borne dans le fichier STEP.

Il est important de noter, ici, qu'il n'y a pas forcément d'équivalence entre deux fonctions  $\mathcal{S}$ . Une fonction  $\mathcal{S}$  peut représenter la définition d'une surface de Bézier alors qu'une autre peut correspondre à une surface NURBS. De plus, contrairement à la méthode de projection harmonique, il n'est pas nécessaire d'effectuer une interpolation entre les surfaces car le point  $\tilde{\mathcal{P}}$  a les mêmes coordonnées paramétriques pour chacune des surfaces.

La sensibilité de la forme par rapport aux paramètres est finalement donnée comme suit :

$$\partial_{\alpha_i} \tilde{\Omega} \simeq \frac{\tilde{\Omega}_{\alpha_i} - \tilde{\Omega}_{\alpha_0}}{\delta \alpha_i} \quad i = 1 \dots p \quad (3.15)$$

La démarche proposée consiste à effectuer les étapes ci-dessous :

- Extraction des données du fichier STEP ;
- Détection du type de surface pour l'utilisation de la fonction  $\mathcal{S}$  adéquate ;
- Récupération des données utiles suivant le type de surface ;
- Calcul des  $(x,y,z)$  correspondants pour tout point de coordonnées  $(u,v)$  ;
- Comparaison avec un autre jeu de  $(x,y,z)$  issu des mêmes coordonnées  $(u,v)$  avec une fonction  $\mathcal{S}$  pouvant être différente.

Cette méthode nécessite de coder l'ensemble des fonctions  $\mathcal{S}$  de définition de surface quels que soient le type et le degré pour être le plus générique possible. En effet, lors de la détection du type de surface, la fonction correspondante est sélectionnée pour calculer l'ensemble des  $(x,y,z)$  respectifs à tout point  $(u,v)$ .

Toutefois, l'interfaçage avec des bibliothèques open source du moteur géométrique Open-Cascade<sup>®</sup> permet d'effectuer cette lecture des fichiers STEP et de récupérer les informations nécessaires à la suite du processus. Celui-ci consiste donc à donner en entrée les coordonnées d'un point  $\tilde{\mathcal{P}}$  et chacun des fichiers STEP correspondants à la géométrie initiale et aux géométries générées suite à la variation des paramètres. Open-Cascade<sup>®</sup> retourne ensuite les coordonnées  $(x,y,z)$  respectives aux coordonnées  $(u,v)$  du point  $\tilde{\mathcal{P}}$ . La comparaison des valeurs retournées permet d'évaluer le déplacement des points  $(x,y,z)$  respectifs. Le post traitement de ces valeurs pour l'ensemble des points  $\tilde{\mathcal{P}}$  est ensuite possible comme on peut le voir sur la figure 3.8.

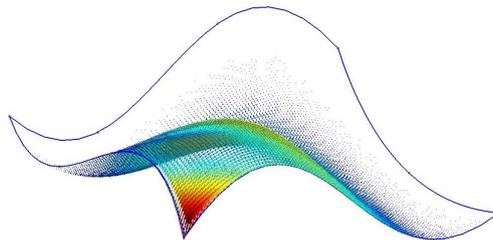


FIGURE 3.8 – Déplacement des points  $\tilde{\mathcal{P}}$  d'une forme suite à la variation d'un paramètre

*In fine*, les sensibilités de la géométrie sont obtenues sur tout point de coordonnées  $(u,v)$ . Toutefois, les logiciels de calculs s'appuient sur des maillages pour effectuer la résolution des problèmes physiques qui sont souvent non-structurés. Afin de faciliter l'intégration de cette méthode, il est donc nécessaire d'obtenir la sensibilité calculée par cette méthode sur le maillage de calcul utilisé par le reste du processus.

3.2.3 Calcul des sensibilités sur un maillage de calcul

Afin d'aller plus loin dans cette démarche et de faciliter la communication de cette méthode avec les logiciels de calcul ou plus généralement son intégration dans un processus d'optimisation complet, il est nécessaire d'avoir la sensibilité sur le maillage de calcul. Ce maillage de calcul est généré sur la géométrie initiale identique à celle du fichier STEP de référence. Le maillage est donc la discrétisation de la surface donnée par le fichier STEP de référence. La première étape consiste à calculer les coordonnées  $(u,v)$  des points du maillage de calcul. Pour cela, les coordonnées  $(x,y,z)$  sont traduites en coordonnées  $(u,v)$  et sont stockées. Ces coordonnées  $(u,v)$  sont ensuite utilisées sur l'ensemble des patches afin de récupérer les coordonnées  $(x,y,z)$  de chacun des patches. La figure 3.9 schématise le fonctionnement du processus permettant d'obtenir cette sensibilité sur le maillage de calcul.

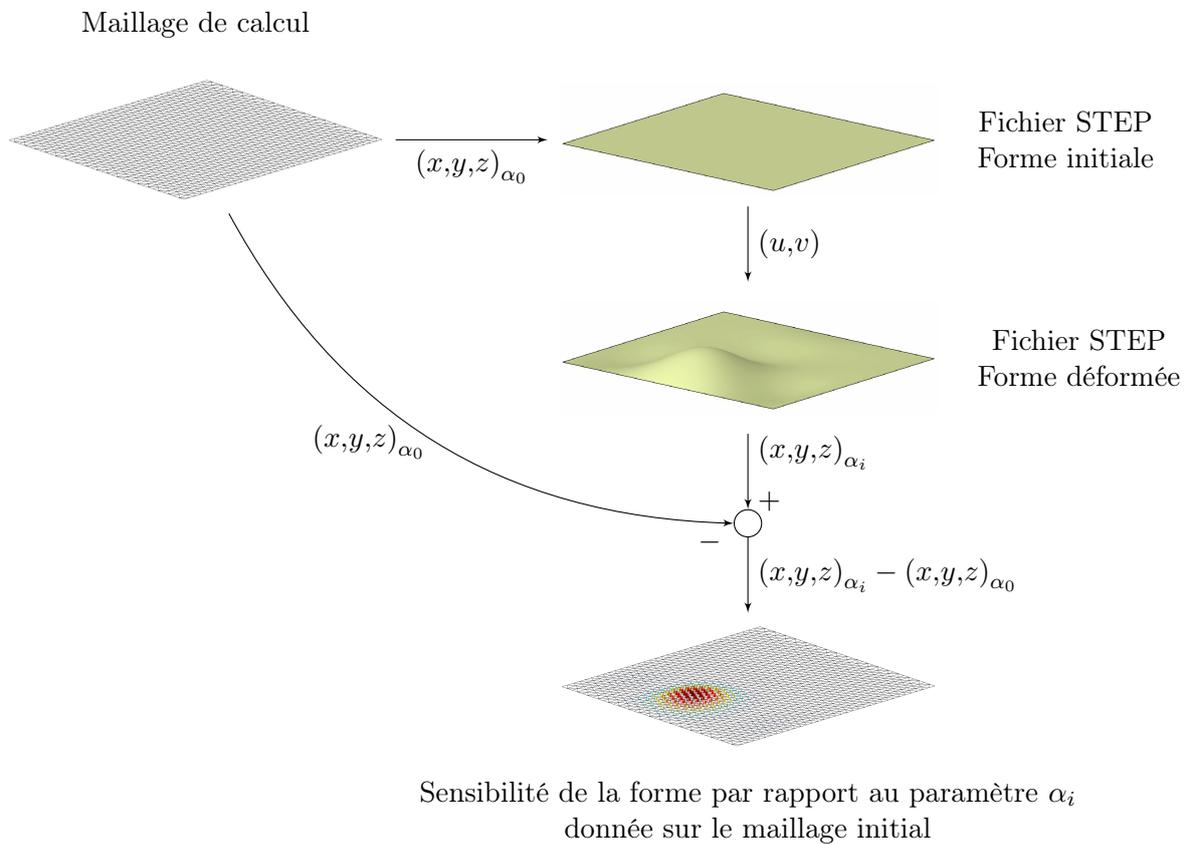


FIGURE 3.9 – Illustration du processus de la méthode semi-analytique avec le résultat sur le maillage de calcul

La méthode présentée dans cette partie permet d'évaluer la variation d'une forme suite à la perturbation d'un paramètre. Même si la limitation du nombre de patches est contraignante, cette méthode donne de bons résultats qu'il est possible de comparer avec la méthode présentée dans le chapitre 2.

### 3.3 Comparaison des méthodes de calcul de sensibilité

Les deux méthodes présentées précédemment évaluent la sensibilité d'une forme par rapport à la variation de paramètres. La méthode par projection harmonique présentée dans le chapitre 2 utilise nativement des maillages pour calculer cette sensibilité. Dans le cas de la méthode présentée dans la partie précédente, ce sont les géométries CAO qui sont utilisées par le biais de format d'échange. Néanmoins, nous venons de voir dans la partie 3.2.3 qu'il est aisé de reporter sur les nœuds d'un maillage la sensibilité calculée sur ces géométries CAO. Dans la suite de cette partie, nous proposons des éléments de comparaison entre les deux méthodes avec dans un premier temps l'étude de la sensibilité d'un patch et ensuite la mise en place d'une optimisation en CFD.

#### 3.3.1 Sur un patch

Les développements actuels de calcul de sensibilité sur les géométries CAO obligent de limiter le nombre de patches car l'association patch à patch n'est pas évidente en cas d'apparition de patch ou de changement d'identifiant. La complexité de la géométrie est volontairement limitée afin de n'avoir qu'un seul patch quelle que soit la variation des paramètres. En effet, la surface résulte de la fonction *remplissage* d'un logiciel de CAO qui se base sur le contour d'un carré de 100 mm de côté et sur 9 points de passage également répartis à l'intérieur du carré comme on peut le voir sur la figure 3.10. Les paramètres utilisés sont le déplacement en  $z$  de chacun des points.

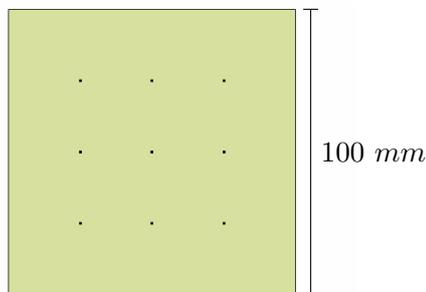


FIGURE 3.10 – Visualisation de la répartition des points de passage sur le patch

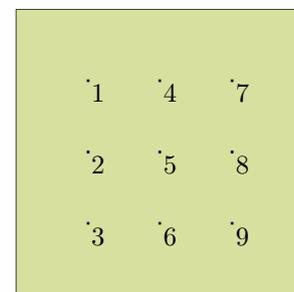
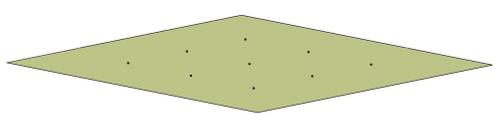


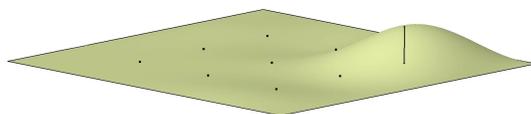
FIGURE 3.11 – Emplacement des paramètres sur le patch vu de face

La figure 3.12 montre deux configurations de ce patch, celle initiale et celle obtenue suite à une variation arbitraire d'un des paramètres.

Nous appliquons les deux méthodes de calcul de sensibilité sur cette surface. Les figures 3.13, 3.14 et 3.15 représentent cette sensibilité pour une variation de 1 mm des paramètres 1, 2 et 5 (cf. figure 3.11). Les sensibilités présentées à droite proviennent de la méthode semi-analytique alors que les sensibilités à gauche sont évaluées avec la méthode de projection harmonique. Nous observons sur la figure 3.13 que la valeur la plus forte est de 1,03 mm alors que la variation appliquée pour chacun des paramètres est de 1 mm. Ceci vient du fait qu'on change la coordonnée en  $z$  d'un point par lequel la surface doit passer. Ce point n'est donc pas nécessairement placé au déplacement maximal de la surface comme on peut le voir sur les figures 3.16. Cela explique

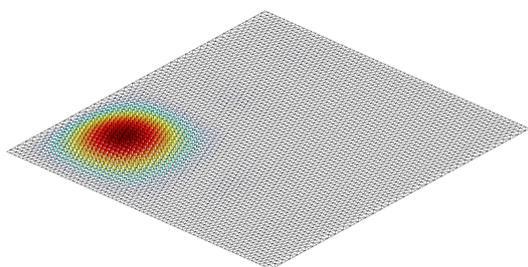


3.12.a – Surface à l'état initial

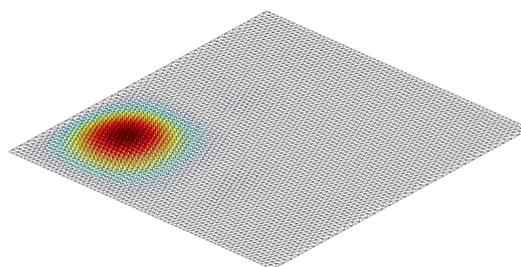


3.12.b – Surface suite à la variation d'un paramètre

FIGURE 3.12 – Visualisation de la variation d'un paramètre



3.13.a – Méthode semi-analytique



3.13.b – Méthode projection harmonique

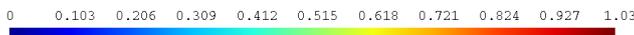
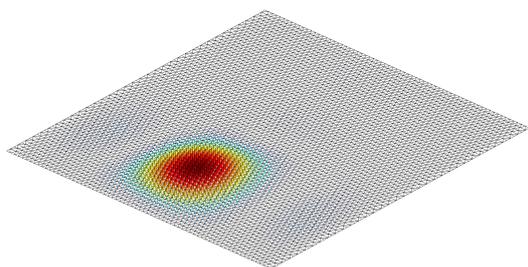
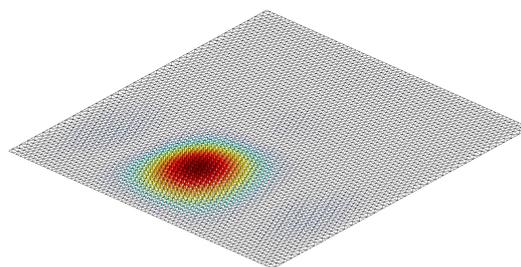


FIGURE 3.13 – Visualisation de la sensibilité de la forme pour une variation de 1 mm sur le paramètre 1



3.14.a – Méthode semi-analytique



3.14.b – Méthode projection harmonique

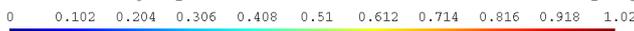


FIGURE 3.14 – Visualisation de la sensibilité de la forme pour une variation de 1 mm sur le paramètre 2

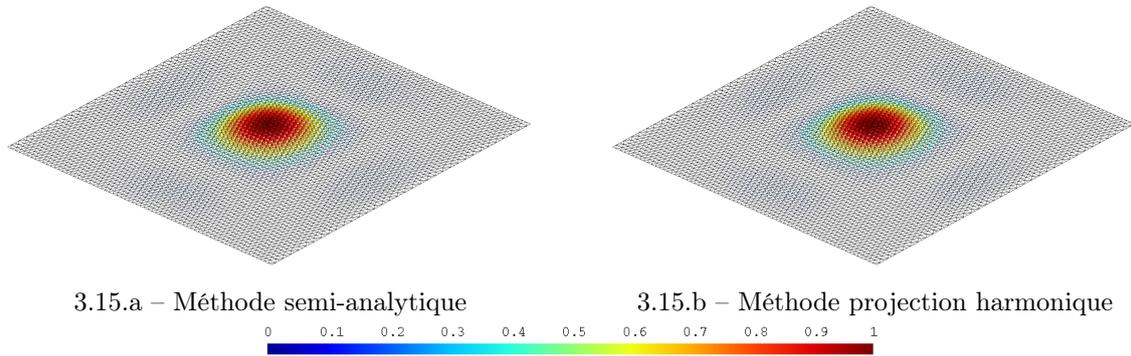


FIGURE 3.15 – Visualisation de la sensibilité de la forme pour une variation de 1 mm sur le paramètre 5

pourquoi la valeur observée lors du post traitement n'est pas identique à celle appliquée au paramètre.



FIGURE 3.16 – Justification de l'écart de valeur

La méthode semi-analytique s'appuie sur les coordonnées  $(x,y,z)$  correspondant à des coordonnées  $(u,v)$  communes. Il n'y a donc pas d'approximation sur le calcul des coordonnées contrairement à la méthode de projection harmonique qui utilise une interpolation (cf. partie 2.3). Sur les figures précédentes, il est difficile de distinguer les différences entre les deux méthodes. De même, l'étude des temps d'exécution des deux méthodes ne présente pas d'intérêt puisque les langages utilisés sont différents. En effet, la méthode de projection harmonique s'appuie en grande partie sur du code de prototypage (Scilab®) alors que la méthode semi-analytique fait appel à une librairie compilée. Nous proposons donc d'étudier l'influence de deux paramètres sur l'écart entre les deux méthodes :

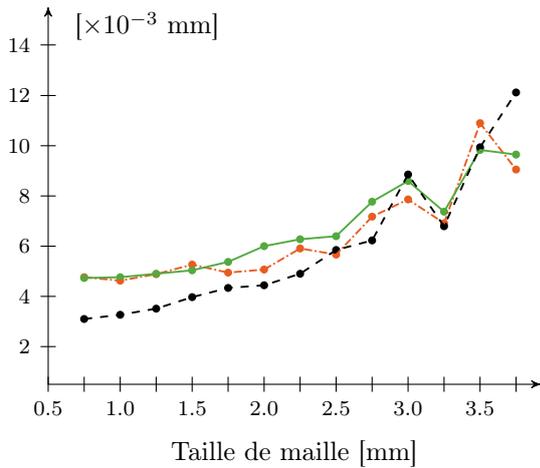
- la taille de maille ;
- la valeur de variation du paramètre CAO.

#### Influence de la taille de maille

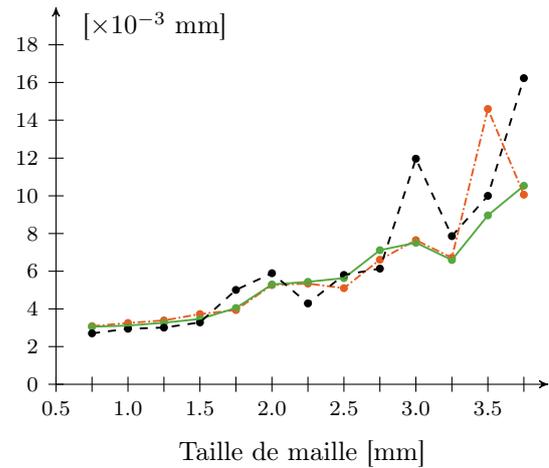
La méthode de projection harmonique se base sur les maillages d'une géométrie pour évaluer sa sensibilité vis-à-vis des paramètres CAO. Le nombre de nœuds et de cellules étant différents entre les différentes formes, une interpolation est utilisée pour reconstruire l'isoconnectivité. La méthode semi-analytique donne les coordonnées  $(x,y,z)$  pour tout couple  $(u,v)$  donné en entrée. Les résultats donnés par cette dernière méthode sont donc plus précis.

Pour observer l'influence de la taille de maille, nous nous concentrons sur les paramètres 1, 2 et 5 de la figure 3.11. Les calculs de sensibilité sont effectués avec chacune des deux méthodes pour différentes valeurs de tailles de maille. Les maillages de référence sont identiques pour les deux calculs, il est donc possible de comparer les valeurs obtenues. Pour chaque valeur de taille de maille, la moyenne (figure 3.17.a) et l'écart type (figure 3.17.b) de la norme des vecteurs résultants de la différence entre les deux sensibilités sont extraits.

Même si la différence observée sur les courbes 3.17 est relativement stable jusqu'à 1,5 mm et qu'elle évolue ensuite légèrement, l'ordre de grandeur reste faible par rapport à la déformation.



3.17.a – Courbe de la moyenne de la différence



3.17.b – Courbe de l'écart type de la différence

FIGURE 3.17 – Courbes sur la norme de la différence des vecteurs de sensibilité :  
 point N°1 : courbe -.-.- | point N°2 : courbe -.-.- | point N°5 : courbe -.-.-

La taille de maille idéale est un compromis entre la précision obtenue et le nombre d'éléments du maillage. En effet, une taille de maille trop faible augmente considérablement le nombre de cellules comme le montre la figure 3.18.

### Influence de la valeur de variation du paramètre

Sur les courbes de la figure 3.19, la différence entre les deux méthodes s'accroît de plus en plus lorsque la variation du paramètre est de plus en plus forte. En effet, la méthode s'appuie sur des différences finies donc les calculs des équations (2.41) et (3.15) ne sont valables que pour des petites perturbations. L'utilisation d'une faible valeur de variation remplit donc les deux conditions.

Comme on peut le voir dans cette partie, les résultats sont assez proches entre les deux méthodes. Toutefois, le calcul par la méthode de projection harmonique utilise une interpolation pour calculer la sensibilité. Cette interpolation peut se révéler préjudiciable si le maillage n'est pas assez fin car la précision en dépend. Étant donné que la méthode basée sur les géométries CAO ne fait pas appel à une interpolation mais aux coordonnées  $(u,v)$  communes aux différents patches, la sensibilité donnée par cette méthode est plus précise. L'observation d'une optimisation utilisant ces sensibilités est intéressante pour comparer les deux méthodes.

### 3.3. Comparaison des méthodes de calcul de sensibilité

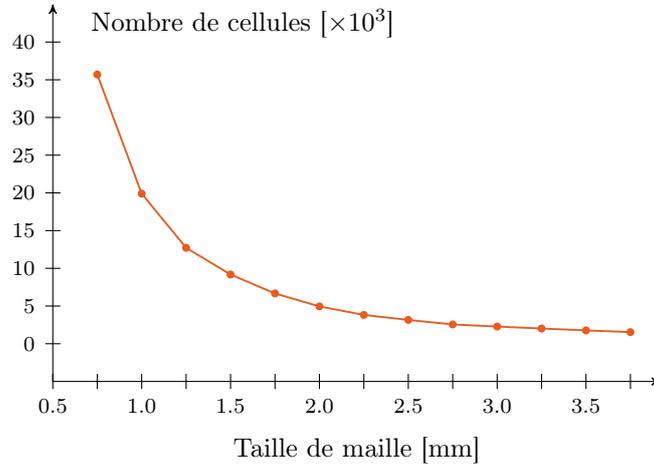
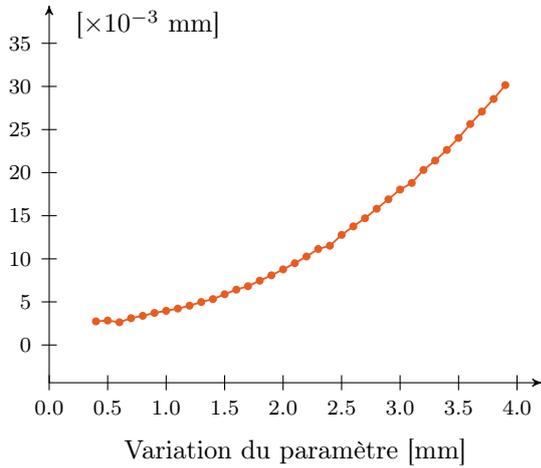
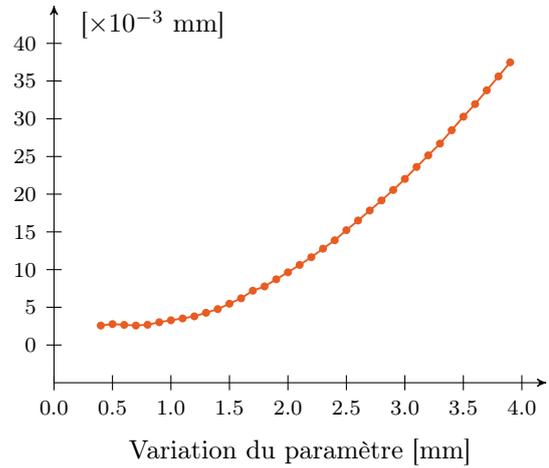


FIGURE 3.18 – Courbe du nombre de mailles en fonction de la taille de maille



3.19.a – Courbe de la moyenne de la différence



3.19.b – Courbe de l'écart type de la différence

FIGURE 3.19 – Courbes sur la norme de la différence des vecteurs de sensibilité au point N°5

### 3.3.2 Dans une optimisation en CFD

Nous étudions maintenant le comportement d'une optimisation CFD avec l'utilisation des deux méthodes de sensibilité. Pour cela, nous menons l'étude sur une pyramide (cf. figure 3.20) placée dans un écoulement et nous cherchons à minimiser la force de traînée induite par cette forme. La base de cette pyramide est placée face au flux d'entrée, ce qui implique une forte force de traînée. L'optimisation porte uniquement sur la surface de cette base. Pour cela, le patch paramétré présenté dans la partie précédente est utilisé comme base de la pyramide (cf. figure 3.20).

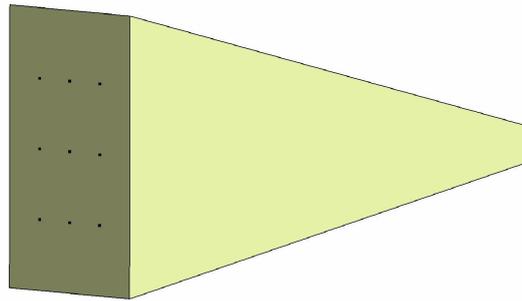


FIGURE 3.20 – Visualisation de la pyramide à l'état initial

Le problème d'optimisation est le suivant :

$$\begin{cases} \min_{\alpha} J(\alpha) \\ s.t. \underline{\alpha}_i \leq \alpha_i \leq \bar{\alpha}_i \quad \forall i = 1 \dots n \end{cases} \quad (3.16)$$

où  $J$  est la force de traînée, soit  $\mathcal{F}_D$ .

L'optimisation est effectuée dans une boucle d'optimisation utilisant un algorithme basé sur une direction de descente. Afin de se concentrer sur la comparaison des deux méthodes, nous ne parlerons pas de la boucle d'optimisation dans ce chapitre. Les détails sur le fonctionnement de la boucle sont donnés au début du chapitre 4. Dans la suite de ce chapitre, nous nous intéressons principalement à la présentation du cas-test et aux résultats obtenus.

#### 3.3.2.1 Calcul physique

Pour effectuer le calcul physique sur cette pyramide, celle-ci est placée dans une veine de dimensions 1,5 m x 1,5 m x 3 m représentée par la figure 3.21.

Le calcul physique utilise un modèle Reynolds-averaged Navier-Stokes (RANS) avec un modèle de turbulence  $k-\varepsilon$ .

Le modèle physique utilisé est constitué des caractéristiques suivantes :

- calcul 3D ;
- stationnaire ;
- solveur couplé ;
- air ;

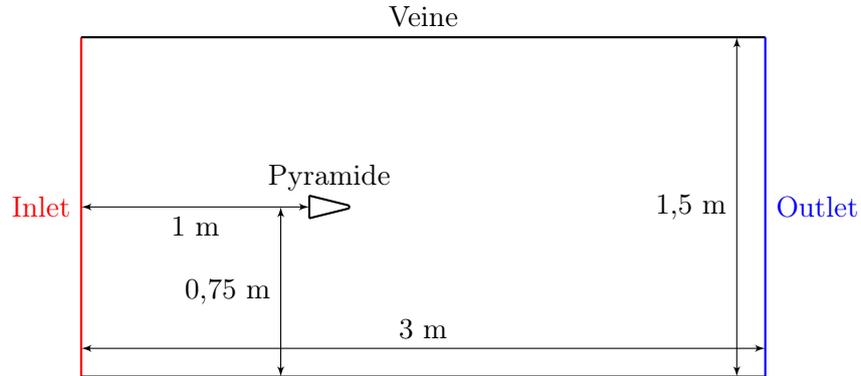


FIGURE 3.21 – Plan de coupe de la veine dans laquelle est placée la pyramide

- modèle Reynolds-averaged Navier-Stokes (RANS) ;
- modèle de turbulence  $k-\varepsilon$ .

La vitesse de l'écoulement en entrée est imposée à  $100 \text{ km.h}^{-1}$  soit environ  $27,8 \text{ m.s}^{-1}$  avec une sortie de pression à l'autre extrémité de la veine. Le nombre d'itérations pour obtenir la convergence est déterminé suite aux calculs de préparation du cas test en prenant une marge assez importante pour prendre en compte un éventuel allongement de la convergence au cours de l'optimisation. De plus, afin de faciliter la convergence du solveur, nous avons décidé d'effectuer une partie des itérations en turbulence gelée après avoir assuré la convergence classique. La répartition du nombre d'itérations pour chacune des étapes du calcul physique est présentée dans le tableau 3.2.

Solveur	Nombre d'itérations
Primaire	2500
Primaire et turbulence gelée	2500
Adjoint	4000

Tableau 3.2 – Tableau de répartition du nombre d'itérations

Pour mener à bien le calcul physique tout en limitant le temps de calcul, il est intéressant d'effectuer une étude de convergence de maillage.

#### 3.3.2.2 Maillage

Afin de choisir la taille de cellule adéquate pour effectuer le calcul dans les meilleures conditions, nous procédons à une étude de convergence de maillage sur la géométrie initiale. Pour cela, le même calcul est lancé avec des tailles de cellules allant de 5,5 à 17 mm et tous les 0,5 mm. La figure 3.22 présente 3 maillages ayant des tailles de cellule différentes. Il faut noter que les tailles de cellules données correspondent aux tailles de cellules obtenues à proximité de la pyramide. En dehors de cette zone, il n'est pas nécessaire d'avoir une telle finesse. La taille de cellules est donc relative à la taille de base. Par exemple, la taille des cellules sur la périphérie de la veine

est fixée à 400% de la taille de base car il n'est pas nécessaire d'avoir un maillage très fin dans ces zones.

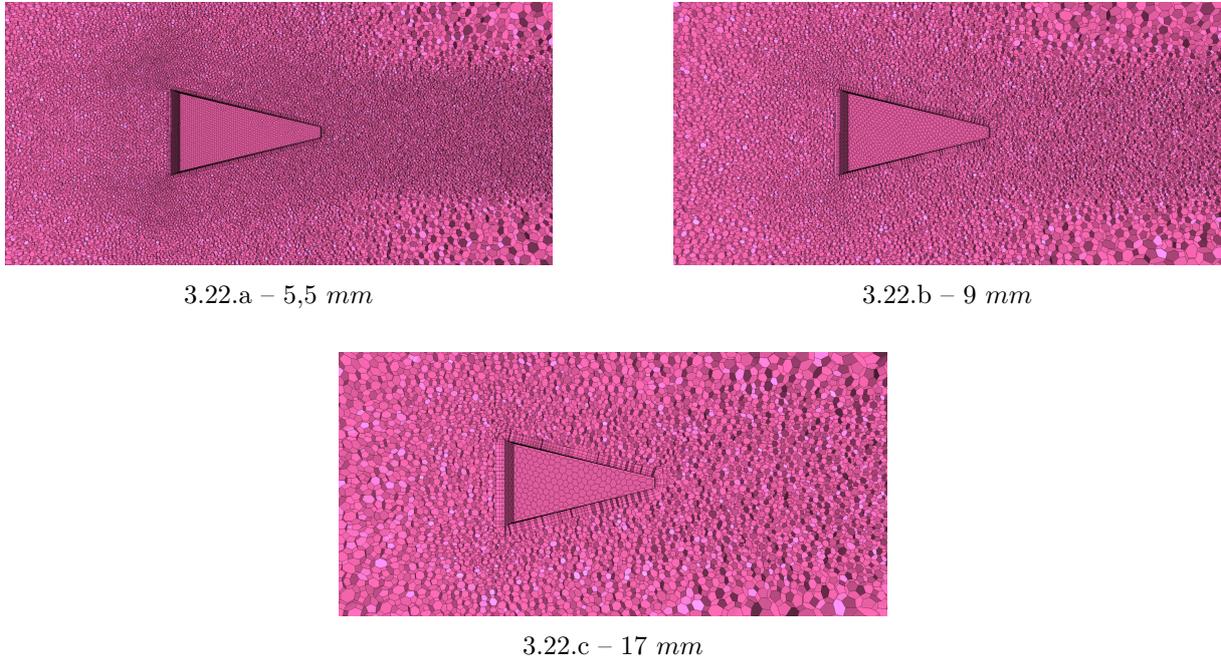


FIGURE 3.22 – Coupe du maillage volumique pour différentes tailles de maillage

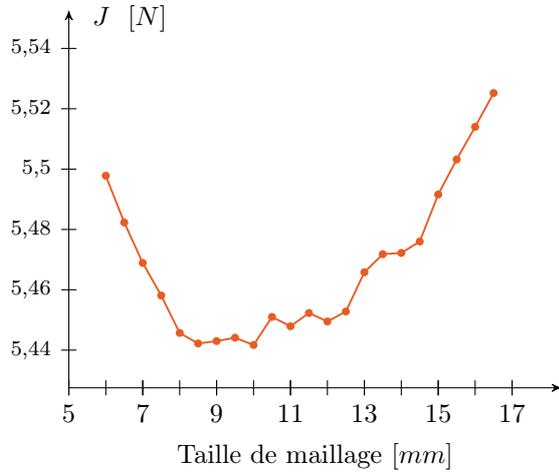
L'étude de la courbe de la figure 3.23.a tracée avec les résultats de la fonction objectif montre qu'un plateau est atteint entre 8 et 10 mm. Ce comportement est caractéristique des simulations numériques. En effet, lorsque le maillage est raffiné, à partir d'une certaine valeur de taille de cellule, le problème physique est indépendant de la taille de maille. En outre, un raffinement trop important provoque une variation de la fonction objectif. En effet, il faut que la taille des cellules soit en adéquation avec le modèle physique retenu comme par exemple le nombre de couches limite. Il est donc nécessaire de choisir une taille de cellules située dans l'intervalle dans lequel la fonction objectif ne varie pas ou peu. Ce choix doit aussi être fait en fonction du nombre de cellules obtenues qui augmente fortement lorsque la taille de base est diminuée (cf. figure 3.23) et qui impacte directement les temps de calcul. Nous sélectionnons ainsi une valeur de 9 mm pour effectuer l'optimisation (cf. figure 3.22.b).

### 3.3.2.3 Post-traitement et résultats

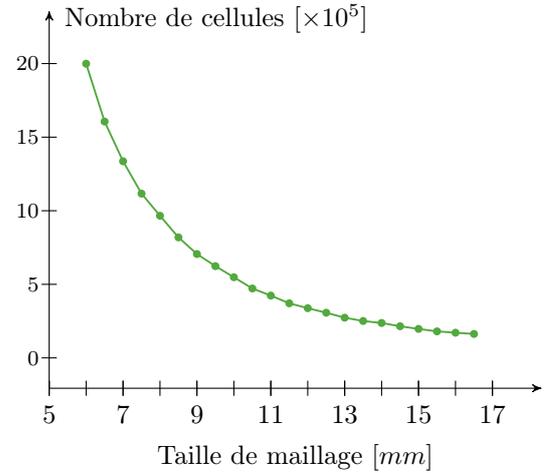
*Dans cette section, pour faciliter la comparaison des résultats, les échelles des post-traitements sont fixes entre les deux états d'une même caractéristique étudiée ( $P$ ,  $v...$ ) ainsi qu'entre les résultats obtenus pour les deux méthodes de calcul de sensibilité.*

Avec la taille de cellule déterminée dans la partie précédente, nous obtenons une force de traînée d'environ 5,44 N pour la géométrie initiale. Les optimisations sont initialisées avec la même forme, c'est-à-dire que l'ensemble des conditions sont identiques pour les deux méthodes afin de

### 3.3. Comparaison des méthodes de calcul de sensibilité



3.23.a – Valeur de la fonction objectif en fonction de la taille de base



3.23.b – Nombre de cellules en fonction de la taille de base

FIGURE 3.23 – Courbes d'étude de convergence de maillage sur la géométrie initiale

ne pas introduire de perturbation. Les calculs physiques sont effectués sur une machine HPC de 16 cœurs et chacun s'exécute en 120 minutes environ.

La figure 3.24 présente l'évolution des courbes respectives aux méthodes de calcul de sensibilité. L'optimisation par la méthode de projection harmonique (courbe  $-\bullet-\bullet-$ ) atteint un optimum à 1,36 N et celle par la méthode semi-analytique (courbe  $\blacklozenge$ ) converge<sup>b</sup> vers 2,14 N.

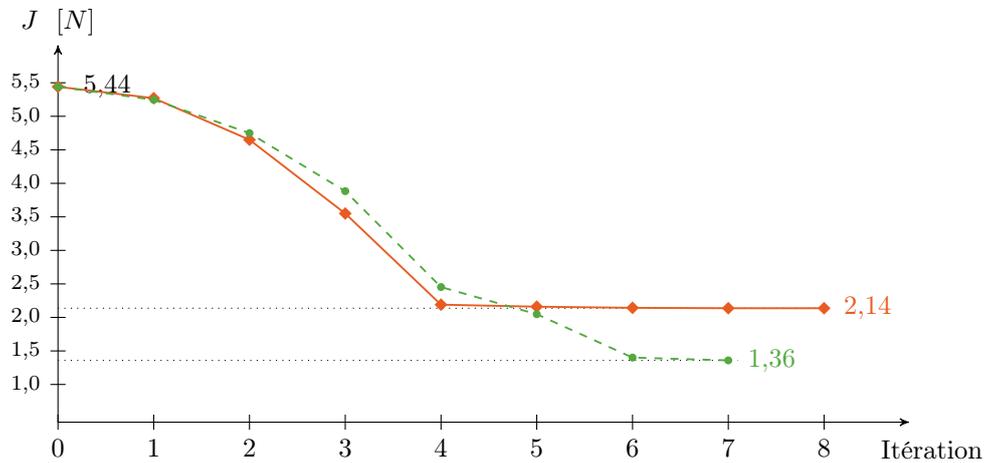


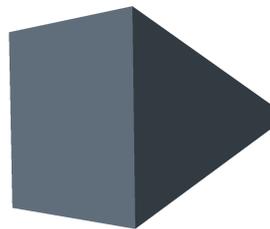
FIGURE 3.24 – Optimisation de la force de trainée par la méthode de projection harmonique (courbe  $-\bullet-\bullet-$ ) et par la méthode semi-analytique (courbe  $\blacklozenge$ )

Jusqu'à l'itération 4, les deux courbes présentent le même profil avec un léger décalage. Ensuite, les deux optimisations subissent un ralentissement. L'optimum obtenu par la méthode analytique est quasiment atteint après ce ralentissement. En effet, l'amélioration est très faible

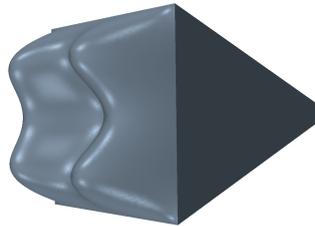
b. Les critères d'arrêt sont présentés dans le chapitre 4.

jusqu'à l'itération 8. Quant à l'optimisation par la méthode de projection harmonique, l'optimisation continue de descendre aux itérations 5 et 6 pour se terminer à l'itération 7. Les deux optimisations n'ont donc pas convergé vers un même minimum local ce qui est souvent le cas dans les méthodes d'optimisation par gradient. Bien que les conditions d'initialisation sont identiques, l'optimum local obtenu n'est pas le même. Toutefois, nous ne pouvons bien évidemment pas conclure sur le fait qu'une méthode soit meilleure que l'autre.

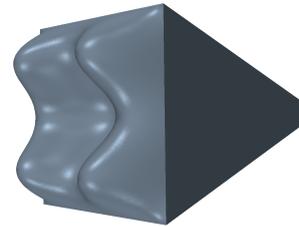
La figure 3.25 montre l'évolution des formes pour les deux optimisations. Nous remarquons que les configurations sont semblables entre les deux méthodes. Les valeurs des paramètres situés sur les coins du patch sont élevées, celles des paramètres au milieu des segments le sont moins et la valeur du paramètre central est plus faible. Toutefois, un décalage important entre les deux formes est flagrant.



3.25.a – Forme initiale :  $\mathcal{F}_D = 5,44 N$



3.25.b – Forme finale avec la méthode de projection harmonique :  $\mathcal{F}_D = 1,36 N$



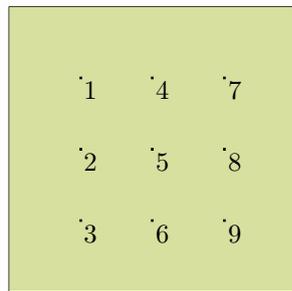
3.25.c – Forme finale avec la méthode semi-analytique :  $\mathcal{F}_D = 2,14 N$

FIGURE 3.25 – Visualisation de la forme à l'état initial et aux états finaux.

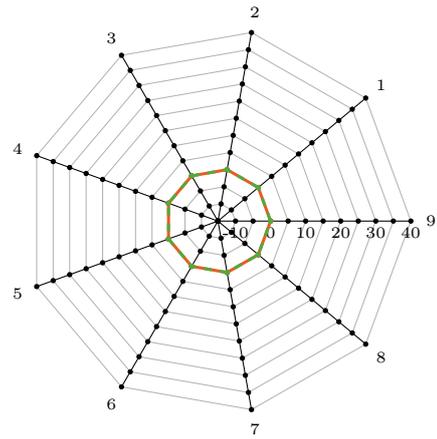
Les diagrammes de la figure 3.26 représentent les valeurs des 9 paramètres pour les itérations 1 (figure 3.26.b), 4 (figure 3.26.c) et 7 (figure 3.26.d). L'ordre des paramètres dans les diagrammes est donné par la figure 3.26.a. À l'itération 4, les valeurs des paramètres sont proches pour les deux méthodes, ce qui justifie la proximité des valeurs de force de traînée à cette itération (cf. figure 3.24). Les motifs obtenus à la figure 3.26.d confirme la ressemblance et le décalage des configurations. La comparaison des courbes pleines (méthode semi-analytique) entre les itérations 4 et 7 montre que la forme n'a pas évolué, ce qui correspond au comportement de la courbe de la figure 3.24.

Les figures 3.27 représentent les valeurs de vitesse autour de la pyramide pour l'état initial (figure 3.27.a), l'état final obtenu par la méthode de projection harmonique (figure 3.27.b) et celui obtenu par la méthode semi-analytique (figure 3.27.c). La traînée induite par la pyramide est beaucoup plus faible pour les états finaux. Celle obtenue par la méthode de projection harmonique est plus effilée que celle obtenue avec le calcul semi-analytique.

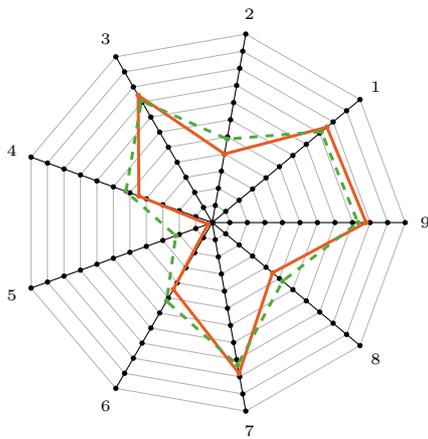
### 3.3. Comparaison des méthodes de calcul de sensibilité



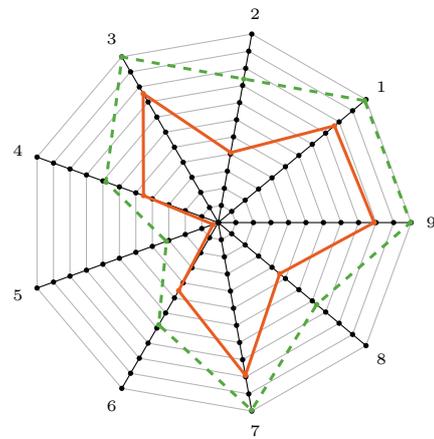
3.26.a – Emplacement des paramètres sur le patch vu de face



3.26.b – Itération 0



3.26.c – Itération 4



3.26.d – Itération 7

FIGURE 3.26 – Évolution des paramètres : méthode de projection harmonique (courbe  $-\cdot-\cdot-$ ) et méthode semi-analytique (courbe  $-\bullet-$ )

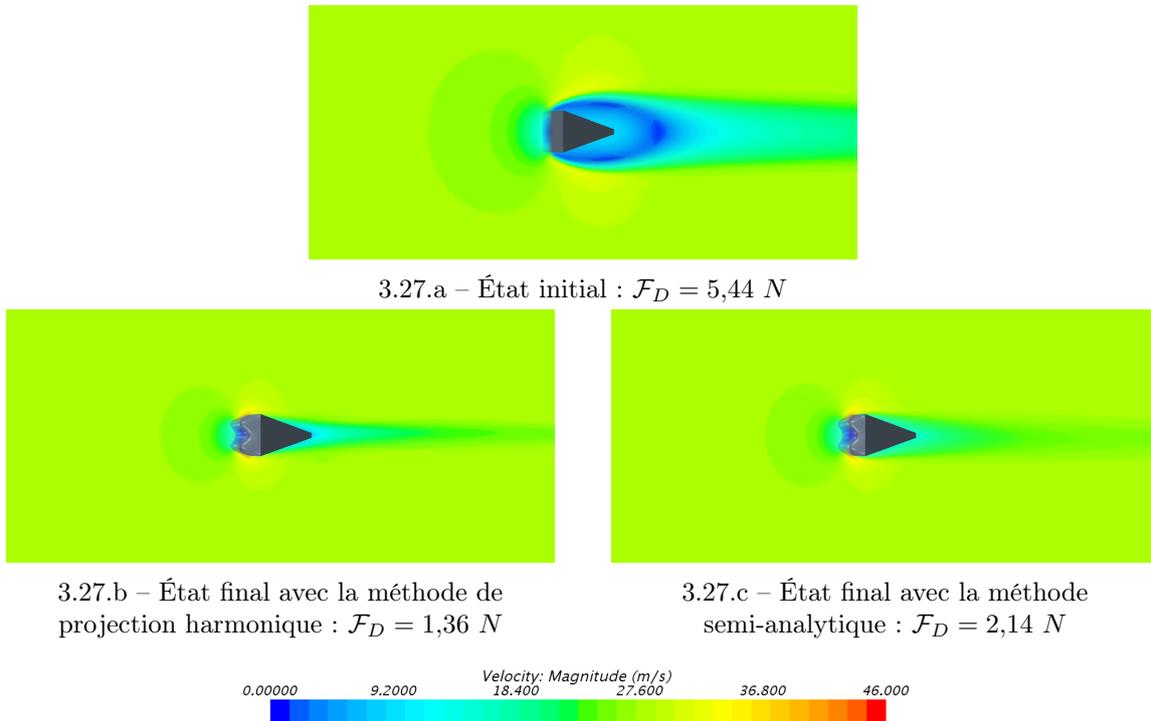


FIGURE 3.27 – Visualisation de la magnitude de la vitesse à l'état initial et aux états finaux.

Les figures 3.28 montrent les valeurs de pression autour de la pyramide pour l'état initial (figure 3.28.a), l'état final obtenu par la méthode de projection harmonique (figure 3.28.b) et celui obtenu par la méthode semi-analytique (figure 3.28.c). On observe que la zone de changement de pression est moins importante aux états finaux qu'à l'état initial. On ne remarque pas de réelles différences entre les deux états finaux.

### 3.3.3 Bilan sur la comparaison des deux méthodes

Les parties précédentes ont montré que les deux méthodes permettent d'effectuer l'évaluation de la sensibilité d'une forme vis-à-vis des paramètres CAO et d'utiliser ces deux méthodes dans une boucle d'optimisation. Toutefois, il est difficile de déterminer laquelle des deux méthodes est plus efficace. En effet, la différence de gain entre les deux méthodes n'est pas seulement due à la méthode de calcul de sensibilité. Le chemin emprunté par une optimisation est susceptible de mener dans un minimum local différent de l'autre optimisation. Un résultat inverse est probablement possible avec des paramètres d'optimisation différents comme par exemple le pas de descente, les valeurs d'initialisation...

## 3.4 Synthèse

Dans ce chapitre, nous avons montré que le calcul de sensibilité est possible en utilisant la définition de la géométrie. Dues à l'impossibilité d'accéder aux données internes du solveur

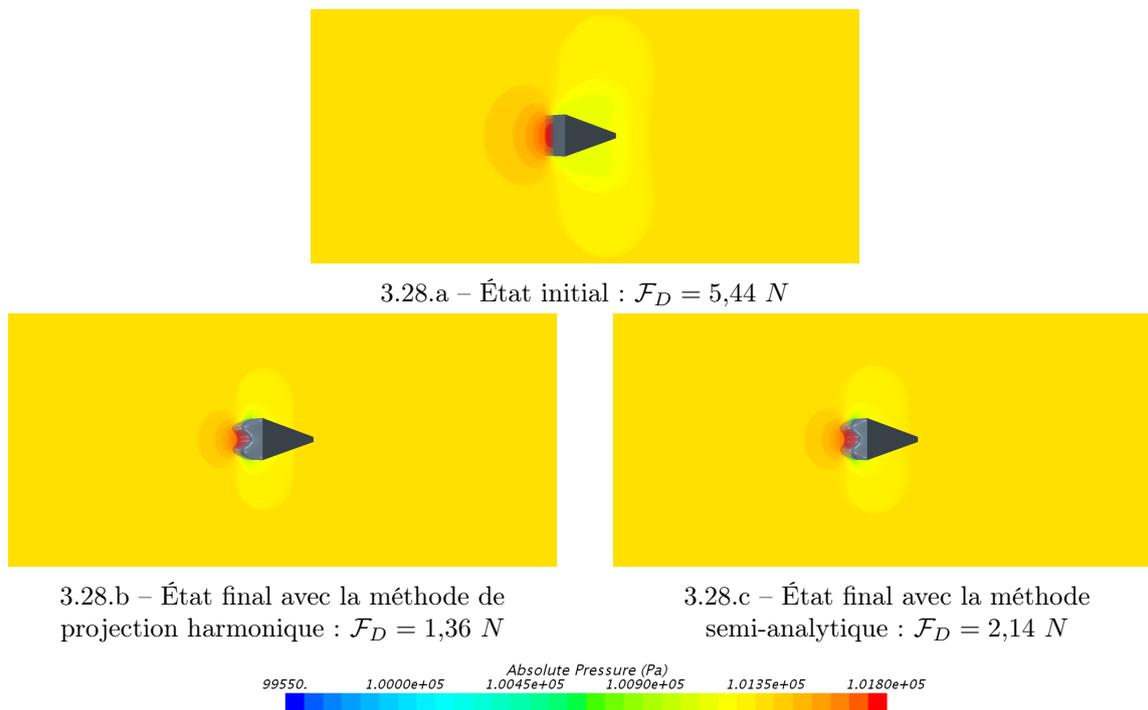


FIGURE 3.28 – Visualisation de la pression à l'état initial et aux états finaux.

géométrique, cette méthode n'exploite pas la définition analytique au sens strict du terme en dérivant l'équation de forme par rapport aux paramètres mais elle donne des résultats satisfaisants en utilisant la définition disponible dans les formats d'échange. L'utilisation de ces formats nécessite de gérer les changements des définitions d'une même surface en fonction des valeurs de paramètres. De plus, la limitation de la complexité des géométries dans le but de garder un même nombre de patches pour ne pas avoir à gérer l'identification patch à patch est un frein important pour le développement de cette méthode. Il serait donc judicieux de mettre en place des moyens permettant d'utiliser cette méthode sur des géométries de plus grande complexité.

Dans la suite du manuscrit, nous optimisons des formes plus complexes que celle optimisée dans cette partie. Cette complexité rend impossible l'utilisation de la méthode de sensibilité par forme CAO. La suite des applications est donc effectuée avec le calcul de sensibilité sur maillage (cf. chapitre 2).

## Bibliographie du 3<sup>e</sup> chapitre

- [Foretnik 2010] Jan Foretnik. Nurbs demo. [geometrie.foretnik.net/files/NURBS-en.swf](http://geometrie.foretnik.net/files/NURBS-en.swf), 2010. (cité pages 70, 71, 72 et 74)
- [Kripac 1997] Jiri Kripac. *A mechanism for persistently naming topological entities in history-based parametric solid models*. Computer-Aided Design, vol. 29, no. 2, pages 113 – 122, 1997. (cité pages 24, 34 et 76)
- [Piegl 1995] L.A. Piegl et W. Tiller. The nurbs book. Monographs in visual communication. Springer, 1995. (cité page 70)

# Applications des calculs de sensibilité dans une boucle d'optimisation

---

<b>4.1</b>	<b>Présentation de la boucle d'optimisation</b>	<b>96</b>
4.1.1	Création du modèle CAO	96
4.1.2	Génération des géométries CAO et maillage	97
4.1.3	Calculs de gradients	98
4.1.4	Algorithme du gradient	98
4.1.4.1	Algorithme de la plus grande descente	99
4.1.4.2	Algorithme du gradient conjugué	102
4.1.4.3	Initialisation du pas	102
4.1.4.4	Critères d'arrêt	103
4.1.4.5	Vérification de la direction de descente et de la norme	103
4.1.5	Gestion d'une optimisation multiobjectif	106
<b>4.2</b>	<b>Spécificités liées à la CFD</b>	<b>107</b>
<b>4.3</b>	<b>1<sup>er</sup> cas-test : Coude à 90°</b>	<b>108</b>
4.3.1	Mises en données	108
4.3.1.1	CAO	109
4.3.1.2	Calcul physique	110
4.3.1.3	Configuration des maillages	111
4.3.2	Post-traitement et résultats	112
4.3.3	Comparaison avec une méthode d'optimisation par plan d'expériences et surface de réponse	116
4.3.3.1	Choix du plan, du modèle de la surface et de l'algorithme	116
4.3.3.2	Résultat et comparaison	118
<b>4.4</b>	<b>2<sup>e</sup> cas-test : Collecteur de type industriel</b>	<b>119</b>
4.4.1	Mises en données	120
4.4.1.1	Modèle CAO	120
4.4.1.2	Calcul physique	122
4.4.1.3	Configuration des maillages	125
4.4.2	Post-traitement et résultats	128
<b>4.5</b>	<b>Discussion des résultats et synthèse</b>	<b>131</b>
	<b>Bibliographie du 4<sup>e</sup> chapitre</b>	<b>132</b>

---

Maintenant que nous connaissons la sensibilité de la forme, soit par l'utilisation de maillages, soit par la définition analytique de la géométrie, il nous est possible d'intégrer ce calcul afin d'effectuer des optimisations paramétriques (cf. 1.3.4). Ces optimisations sont possibles dès lors que l'information du gradient est disponible pour la fonction objectif donnée. Nous verrons tout d'abord le fonctionnement global de la boucle avant de voir son application sur certains cas d'étude représentatifs.

### 4.1 Présentation de la boucle d'optimisation

La boucle d'optimisation, présentée par la figure 4.1, est pilotée par un programme qui gère le lancement des logiciels, le partage des fichiers nécessaires et la communication. Les paragraphes suivants présentent les différentes étapes de cette boucle, les spécificités liées à la robustesse et aux problèmes multiobjectifs.

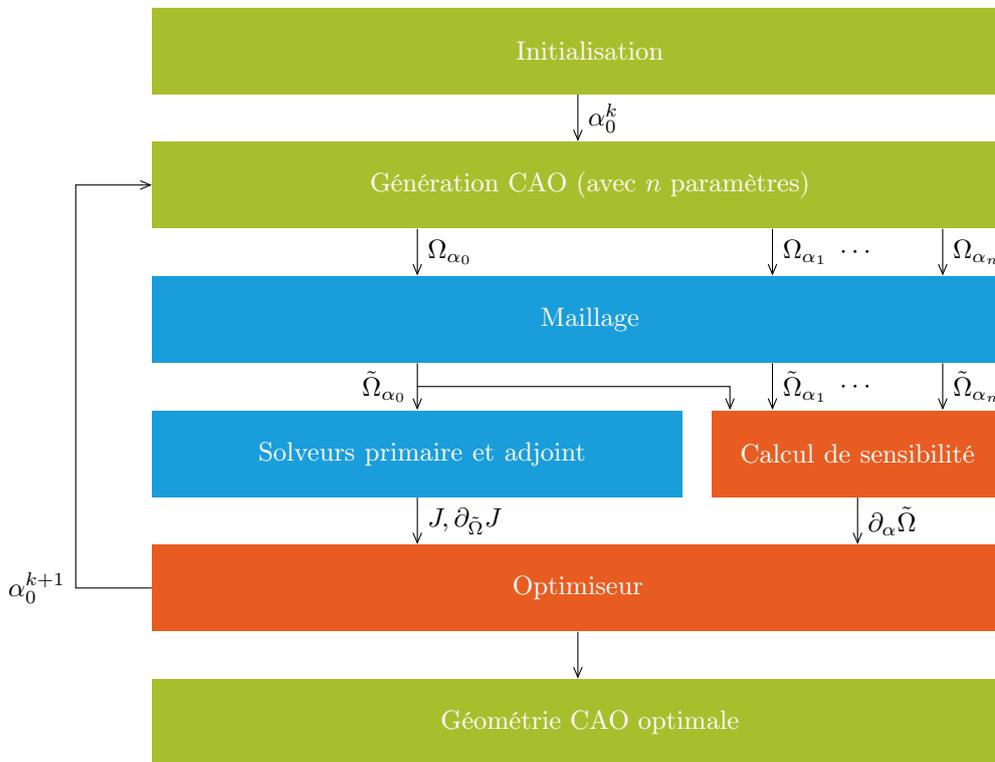


FIGURE 4.1 – Boucle d'optimisation proposée

Ici, la géométrie optimale ne l'est pas forcément au sens propre, il s'agit d'une solution qui améliore le problème posé et qui respecte les critères d'arrêt.

#### 4.1.1 Création du modèle CAO

La première étape de la boucle consiste à créer la géométrie initiale. Le modèle CAO est l'objet central de la boucle et il est le support des paramètres d'optimisation.

Lors de l'élaboration du modèle CAO, la géométrie est créée et les paramètres lui sont appliqués. Ces derniers sont les contraintes géométriques des éléments d'esquisse (longueurs, angles, perpendicularité, tangence...), les contraintes appliquées aux éléments de référence ou des paramètres d'opérations appliquées aux géométries comme des congés tritangents... Un paramètre peut impacter différents éléments à la fois à l'aide de relations. Il est constitué de trois informations : sa valeur initiale  $\alpha_i$ , sa borne inférieure  $\underline{\alpha}_i$  et sa borne supérieure  $\bar{\alpha}_i$ . Les valeurs initiales des paramètres sont fixées arbitrairement. Les  $n$  paramètres sur lesquels l'optimisation est effectuée ainsi que leurs trois valeurs respectives sont contenues dans la table de paramétrage initiale qui sert de référence tout au long de l'optimisation.

En outre, les éléments géométriques nécessaires à l'identification des nœuds particuliers (cf. 2.2.4) sont ajoutés tout à la fin de cette étape.

### 4.1.2 Génération des géométries CAO et maillage

L'étape suivante est la génération de l'ensemble des géométries. En prenant les informations données par la table de paramétrage initiale, le programme génère la table des géométries. Dans cette table, nous retrouvons les valeurs des paramètres de la géométrie initiale  $\Omega_{\alpha_0}$  et ceux de l'ensemble des géométries perturbées  $\Omega_{\alpha_i}$ . Le tableau 4.1 donne un exemple de la table de géométries pour 4 paramètres. Chaque géométrie perturbée correspond à une variation de paramètre. Avec  $n$  paramètres,  $n + 1$  géométries sont ainsi obtenues.

Configuration	Paramètre 1	Paramètre 2	Paramètre 3	Paramètre 4
0 (référence)	1 mm	1 mm	1 mm	1 mm
1	1,1 mm	1 mm	1 mm	1 mm
2	1 mm	1,1 mm	1 mm	1 mm
3	1 mm	1 mm	1,1 mm	1 mm
4	1 mm	1 mm	1 mm	1,1 mm

Tableau 4.1 – Table de géométries pour 4 paramètres avec une variation de 10%

Cette variation de paramètre est donnée par un critère de perturbation. Dans le cadre de nos optimisations, il est défini de manière empirique comme un pourcentage de déformation de la valeur initiale du paramètre. Bien souvent, il est équivalent pour l'ensemble des paramètres.

Le taux de reconstruction d'une géométrie CAO peut se révéler assez faible [Froment 2014]. En effet, suivant les valeurs de paramètres, le solveur géométrique peut rencontrer des singularités et n'a pas la possibilité de générer la forme correspondante au jeu de paramètres donné. Cela est préjudiciable pour le calcul de sensibilité puisque l'évaluation du paramètre ne peut donc pas être effectuée. Fort heureusement, ces erreurs de génération de géométries sont détectables sans pour autant bloquer la poursuite de l'optimisation. Un changement, même minime, de la valeur du paramètre incriminé peut permettre que la génération soit de nouveau possible. Deux solutions ont donc été envisagées pour pallier ce problème. La première consiste à ne pas prendre en compte, à l'itération donnée, le paramètre qui empêche la génération de la forme. Le calcul de sensibilité pour ce paramètre n'est donc pas effectué. Dans la majorité des cas, à l'itération suivante, la variation des autres paramètres permet au solveur géométrique de générer la forme en prenant de nouveau en compte ce paramètre. Une autre solution est de lancer un plan de

Monte-Carlo autour du jeu de paramètres incriminé afin de générer une géométrie. Cette valeur est ensuite répercutée dans le reste de l'optimisation.

Suite à la génération des géométries, l'ensemble des géométries sont maillées. Il faut noter que les qualités de maillage ne sont pas nécessairement identiques entre les géométries perturbées et celle destinée au calcul physique. De même que la géométrie initiale peut être maillée de deux façons : celle respectant les conditions de maillage requises pour le calcul physique et celle correspondante aux valeurs de maillage des géométries perturbées. Toutefois, un tel choix engendre l'utilisation d'une interpolation pour faire concorder les informations données sur les maillages respectifs.

### 4.1.3 Calculs de gradients

Pour ce qui est de la physique, il n'y a pas de restrictions autres que celle liée à la nécessité de connaître le gradient de la fonction objectif. Pour obtenir les gradients d'une fonction objectif sur un maillage, il est nécessaire en CFD d'avoir une très bonne qualité de maillage.

La figure 4.1 présente le calcul du solveur primaire et du solveur adjoint en une seule étape commune. Afin de réduire le temps de calcul et surtout quand cela est possible – par exemple dans le cas d'un algorithme à pas adaptatif –, il est plus judicieux de faire le test d'amélioration de la fonction objectif avant de lancer le solveur adjoint. En effet, dans le cas où la solution n'a pas été améliorée, ce sont les gradients de l'itération précédente qui sont utilisés et ceux de l'itération actuelle ne sont pas gardés. Une telle approche est souvent laborieuse avec les logiciels de simulation car les solveurs primaire et adjoint sont la plupart du temps indissociables.

La faible variation de forme entre deux itérations permet d'initialiser le calcul physique avec les valeurs du calcul précédent. Cela représente un gain de temps considérable car le flux est déjà établi dans la conduite avec des valeurs proches de celles attendues. Dans ce cas, nous récupérons les champs de pression, de vitesse et les paramètres liés aux modèles physiques utilisés comme le taux de dissipation  $\varepsilon$ , l'énergie cinétique de turbulence  $k$ ... Le calcul de sensibilité utilise l'ensemble des géométries maillées pour déterminer la variation de la forme suite à la perturbation de chaque paramètre.

L'équation (4.1) utilise chacun des termes donnés par les calculs précédents :

$$d_{\alpha}J = \partial_x J \times \partial_{\alpha}x \quad (4.1)$$

Les deux étapes précédentes sont indépendantes l'une de l'autre et sont effectuées en parallèle. En CFD, les temps de calcul physique sont largement supérieurs à ceux de la sensibilité de la forme par rapport aux paramètres CAO. La boucle d'optimisation est donc peu affectée par l'augmentation du nombre de paramètres.

### 4.1.4 Algorithme du gradient

Le choix de l'optimiseur conditionne le nombre d'itérations, de calculs et la vitesse de convergence de la boucle. De plus, l'optimiseur a besoin de différentes informations provenant des différentes étapes précédentes. Même si l'étude de l'optimiseur ne constitue pas le cœur de la thèse, la boucle d'optimisation mise en place requiert la connaissance de son fonctionnement. Les

méthodes utilisées restent simples et pourront faire l'objet d'améliorations futures afin de tirer parti au mieux de la méthode présentée dans ce manuscrit.

Les algorithmes présentés dans cette section se basent sur la recherche d'une direction de descente  $d_k$ . En effet, dans le cas de la minimisation<sup>a</sup> de la fonction objectif  $J$ , celle-ci décroît le plus fortement dans la direction opposée à celle du gradient de  $J$ , noté  $\nabla J$  pour le paramètre  $\alpha_i$  et ce localement. Dans notre cas, les sensibilités sont calculées pour chaque paramètre  $\alpha_i$  par l'équation (4.1).

Un algorithme de descente est déterminé par les stratégies de choix des directions de descente successives et du pas effectué à chaque itération dans la direction choisie. La direction de descente  $d_k$  influe directement dans la nomination de l'algorithme.

### 4.1.4.1 Algorithme de la plus grande descente

L'*algorithme de la plus grande descente*, également appelé *algorithme de la plus profonde descente* est très simple à mettre en œuvre. Sa direction de descente  $d_k$  est donnée par :

$$d_k = -\nabla J(\alpha_k) \tag{4.2}$$

Son principal défaut est sa convergence très lente au voisinage des points stationnaires. Cela se caractérise par le comportement en zigzag des itérés. Cela est expliqué par le fait que deux directions de descente successives calculées par l'algorithme du gradient sont orthogonales.

La direction de descente obtenue nous permet de déterminer des nouvelles valeurs de paramètres données par :

$$\alpha^{k+1} = \alpha^k + \beta d_k \tag{4.3}$$

Le choix du pas  $\beta$  peut influencer la rapidité et la stabilité de convergence de l'algorithme. Les paragraphes suivants présentent différentes méthodes relatives aux choix du pas.

Dans les deux cas, les nouveaux paramètres constituent les paramètres de la nouvelle géométrie initiale si chacune des valeurs respecte ses bornes respectives fixées lors de la création du modèle CAO et reportées dans le table de paramétrage initiale. Lorsqu'un paramètre dépasse une de ses bornes, il prend la valeur de celle-ci. La sélection d'un meilleur algorithme n'était pas le but premier de la thèse. Un choix plus approfondi des méthodes proposées par [Culioli 1994] permettront sûrement d'améliorer cette partie de la boucle.

### Algorithme du gradient à pas constant

L'utilisation d'un pas constant pour le calcul des nouveaux paramètres est la méthode la plus simple. En début d'optimisation, le pas est fixé soit de façon arbitraire, soit de façon calculatoire. Il est ensuite gardé fixe tout au long de l'optimisation et n'implique pas de condition en fonction des résultats obtenus que ce soit des améliorations ou des échecs. Le pas constant ne permet pas de converger rapidement vers l'optimum et peut générer des instabilités.

---

a. Dans le cas d'une maximisation, on écrit le problème en  $-J(\alpha)$ , ce qui revient à un problème de minimisation.

### Algorithme du gradient à pas adaptatif

Contrairement à la méthode précédente, il est possible d'adapter le pas en fonction des résultats de l'optimisation. En effet, si la valeur de la fonction objectif  $J^k$  est inférieure à celle de l'itération précédente  $J^{k-1}$ , les nouvelles valeurs des paramètres sont calculées par l'équation (4.3). Dans notre cas, nous proposons de multiplier le pas  $\beta$  par 1,1. Dans le cas contraire, c'est-à-dire  $J^k > J^{k-1}$ , le calcul reprend les précédentes valeurs de paramètres et de gradient et le pas  $\beta$  est divisé par 2. Les méthodes présentées précédemment utilisent des conditions simples pour calculer le pas  $\beta$  d'une itération à l'autre. Ce pas n'est donc pas nécessairement optimal au regard de l'optimisation.

### Recherche linéaire

Afin de déterminer le pas optimal, certaines méthodes proposent d'effectuer une recherche linéaire [Armijo 1966, Wolfe 1971]. Cela consiste à trouver un pas  $\beta_k$  le long d'une direction de descente  $d_k$  de façon à diminuer la fonction  $J$  suivant cette direction. On cherche donc  $\beta$  tel que :

$$\min_{\beta > 0} \{J(\alpha_k) - \beta \nabla J(\alpha_k)\} \tag{4.4}$$

On s'intéresse dans nos cas aux recherches linéaires inexactes, c'est-à-dire que l'on préfère des règles de calcul moins restrictives et par conséquent moins gourmandes en temps de calcul. L'objectif est double :

1. faire décroître  $J$  suffisamment ;
2. empêcher le pas  $\beta_k > 0$  d'être trop petit afin d'empêcher la convergence vers un point non stationnaire.

On trouvera donc tout un intervalle de pas possibles.

#### o Règle d'Armijo

La condition d'Armijo demande à ce que  $J$  décroisse autant qu'une portion  $\omega_1 \in ]0,1[$  de ce que ferait le modèle linéaire de  $J$  en  $\alpha_k$ , ce qui conduit à l'inégalité suivante (condition de décroissance linéaire) :

$$J(\alpha_k + \beta_k d_k) \leq J(\alpha_k) + \omega_1 \beta_k \langle \nabla J(\alpha_k), d_k \rangle \tag{4.5}$$

avec  $\langle \cdot, \cdot \rangle$  le produit scalaire de  $\mathbb{R}^n$

En pratique, la constante  $\omega_1$  est prise très petite (typiquement,  $\omega_1 = 10^{-4}$ ) afin de satisfaire le plus facilement possible l'équation (4.5). On remarque que cette seule condition ne suffit pas à se prémunir d'un pas qui deviendrait trop petit. [Armijo 1966] propose d'utiliser la technique de rebroussement, tout comme [Goldstein 1965].

o Règle de Wolfe

La règle de Wolfe fait appel au calcul du gradient  $\nabla J(\alpha_k)$  et peut donc devenir plus coûteuse que la règle de Goldstein. Cette règle est très utilisée lorsque le calcul du gradient représente un faible coût additionnel en comparaison du coût d'évaluation de  $J(\alpha)$ .

La première inégalité correspond à la condition de décroissance linéaire (cf. équation (4.5)) tandis que la seconde s'assure que le pas ne soit pas trop petit. Ce dernier est déterminé de façon à satisfaire les deux équations suivantes :

$$\begin{cases} J(\alpha_k + \beta_k d_k) \leq J(\alpha_k) + \omega_1 \beta_k \langle \nabla J(\alpha_k), d_k \rangle \\ \langle \nabla J(\alpha_k + \beta_k d_k), d_k \rangle \geq \omega_2 \langle \nabla J(\alpha_k), d_k \rangle \end{cases} \quad (4.6)$$

où les constantes  $\omega_1$  et  $\omega_2$  sont choisies telles que  $0 < \omega_1 < \omega_2 < 1$ . Typiquement, on prend  $\omega_1 = 10^{-4}$  et  $\omega_2 = 0.99$ .

L'algorithme 4.1 est basé sur la méthode de la bissectrice pour les conditions faibles de Wolfe et s'appuie sur la méthode attribuée à Fletcher et Lemaréchal. Cet algorithme pourrait être accéléré en faisant une recherche linéaire de Wolfe par interpolation quadratique ou cubique.

**Données :**  $\beta_{min} = 0$  et  $\beta_{max} = -1$ ;  $\omega_1$  et  $\omega_2$  telles que  $0 < \omega_1 < \omega_2 < 1$

**Début de l'algorithme**

```

tant que  $J(\alpha_k + \beta_k d_k) > J(\alpha_k) + \omega_1 \beta_k \langle \nabla J(\alpha_k), d_k \rangle$ 
  ou  $\langle \nabla J(\alpha_k + \beta_k d_k), d_k \rangle < \omega_2 \langle \nabla J(\alpha_k), d_k \rangle$ , faire
    si  $J(\alpha_k + \beta_k d_k) > J(\alpha_k) + \omega_1 \beta_k \langle \nabla J(\alpha_k), d_k \rangle$  alors
       $\beta_{max} = \beta_k$ 
       $\beta_k = \frac{1}{2}(\beta_{min} + \beta_{max})$ 
    sinon si  $\langle \nabla J(\alpha_k + \beta_k d_k), d_k \rangle < \omega_2 \langle \nabla J(\alpha_k), d_k \rangle$  alors
       $\beta_{min} = \beta_k$ 
      si  $\beta_{max} < 0$  alors
         $\beta_k = 2\beta_{min}$ 
      sinon
         $\beta_k = \frac{1}{2}(\beta_{min} + \beta_{max})$ 
      fin
    fin
  fin

```

**Fin de l'algorithme**

**Algorithme 4.1 :** Algorithme de recherche d'un pas de Wolfe

Dans notre cas, les durées de calcul et le nombre d'itérations effectuées lors des optimisations peuvent limiter les gains obtenus par de telles méthodes. Nous reviendrons sur ce point lors de la discussion des résultats des différents cas d'étude.

#### 4.1.4.2 Algorithme du gradient conjugué

Cet algorithme peut être vu comme une modification de l'algorithme du gradient. La direction du gradient conjugué vaut :

$$d_k = \begin{cases} -\nabla J(\alpha_0) & \text{si } k = 0 \\ -\nabla J(\alpha_k) + \lambda_k d_{k-1} & \text{si } k \geq 1 \end{cases} \quad (4.7)$$

avec  $\lambda_k$  un scalaire de  $\mathbb{R}$  qui caractérise les différentes variantes du gradient conjugué. Nous avons utilisé la formulation de Fletcher-Reeves pour le calcul de  $\lambda_k$  :

$$\lambda_k = \frac{\|\nabla J(\alpha_k)\|^2}{\|\nabla J(\alpha_{k-1})\|^2} \quad (4.8)$$

#### 4.1.4.3 Initialisation du pas

Quelle que soit la méthode utilisée, il est nécessaire d'initialiser le pas. Pour cela, on note  $\Delta_k$  la décroissance attendue du critère à l'itération  $k$  :

$$\Delta_k = J(\alpha_{k-1}) - J(\alpha_k) \quad (4.9)$$

On peut initialiser le premier pas en prenant le *pas de Fletcher* :

$$\beta_k = \frac{-2\Delta_k}{\nabla J(\alpha_k) d_k} \quad (4.10)$$

Avec ce pas, la détermination du pas initial est reportée sur la détermination de la décroissance attendue  $\Delta_k$ . Si l'on a une idée de la valeur minimale  $J_{min}$  que l'on peut atteindre, on peut initialiser notre pas  $\beta_k$  avec l'équation (4.10) en prenant :

$$\Delta_k = \gamma (J(\alpha_k) - J_{min}) \quad (4.11)$$

où  $\gamma$  est de l'ordre de  $10^{-2}$  ou  $10^{-1}$ . Globalement, dans les cas traités, le gain envisagé est de 30% à 40%<sup>b</sup>. La décroissance attendue est donc donnée par :

$$\begin{aligned} \Delta_k &= \gamma (J(\alpha_k) - J_{min}) \\ &= \gamma \left( J(\alpha_0) - \frac{6}{10} J(\alpha_0) \right) \\ &= \gamma \left( \frac{2}{5} J(\alpha_0) \right) \\ &= \frac{1}{25} J(\alpha_0) \end{aligned}$$

Une autre façon d'initialiser le pas consiste à fixer la décroissance désirée par rapport au point initial. Si le gain attendu est de l'ordre de 10%, alors on a :

$$\begin{aligned} \Delta_k &= J(\alpha_0) - J(\alpha_1) \\ &= J(\alpha_0) - 0.9J(\alpha_0) \\ &= 0.1J(\alpha_0) \end{aligned}$$

---

b. Ces valeurs ont été retenues suite aux observations effectuées sur les différentes optimisations

Dans le cas où l'on n'a pas d'idée sur un possible gain, on pourrait être tenté de réajuster la valeur de  $\Delta_k$  après la première itération par exemple, en utilisant la valeur de la décroissance de  $J$  réalisée à l'itération précédente : on aurait donc  $\Delta_k = J(\alpha_{k-1}) - J(\alpha_k)$ . Seulement, on ignore le comportement de l'algorithme et dans le cas où ce dernier aurait une convergence superlinéaire,  $J(\alpha_k) - J(\alpha_{k+1})$  serait beaucoup plus petit que  $J(\alpha_{k-1}) - J(\alpha_k)$ .  $\beta_k$  aurait alors une valeur beaucoup trop forte pour la suite.

### 4.1.4.4 Critères d'arrêt

Comme nous l'avons vu dans le chapitre 1, nos problèmes sont contraints au niveau des bornes. Ainsi, un critère d'arrêt portant uniquement sur le gradient tel que  $\nabla J(\alpha_k) \simeq 0$  ne suffit pas. On fait donc souvent appel à des critères fondés sur l'expérience [Rondepierre 2013] :

- valeur du pas :  $\beta < \varepsilon$
- stagnation de la solution :  $\frac{\|\alpha_{k+1} - \alpha_k\|}{1 + \|\alpha_k\|} < \varepsilon$  ;
- stagnation de la valeur courante :  $\frac{|J(\alpha_{k+1}) - J(\alpha_k)|}{1 + |J(\alpha_k)|} < \varepsilon$  ;
- nombre d'itérations de boucle dépassant une valeur fixée :  $k < \text{nombre d'itérations maximum}$ .

### 4.1.4.5 Vérification de la direction de descente et de la norme

En optimisation basée sur le gradient, une erreur sur le gradient met très vite un terme au processus d'optimisation car il propose des solutions aberrantes. En outre, dans cette partie, on souhaite vérifier numériquement que l'on obtient la bonne direction de descente. On prend l'hypothèse que le résultat fourni par le solveur adjoint **STAR-CCM+**<sup>®</sup> est validé, la seule source d'erreur possible serait le gradient reporté sur les paramètres **CAO**. Afin de valider ce calcul, on vérifie par la méthode de différences finies les résultats obtenus par l'utilisation du gradient. Pour cela, on utilise la formule de Taylor-Young qui conduit aux deux formules suivantes :

$$\begin{cases} J(\alpha + hu) = J(\alpha) + h \langle \nabla J(\alpha), u \rangle + \mathcal{O}(h) \\ J(\alpha - hu) = J(\alpha) - h \langle \nabla J(\alpha), u \rangle + \mathcal{O}(h) \end{cases} \quad (4.12)$$

En faisant la moyenne des deux différences finies antérieure et postérieure à  $J(\alpha)$ , on obtient une approximation de  $\nabla J(\alpha)$  du deuxième ordre en  $h$  :

$$\langle \nabla J(\alpha), u \rangle = \frac{J(\alpha + hu) - J(\alpha - hu)}{2h} \quad (4.13)$$

On choisit deux jeux de paramètres  $\alpha_1$  et  $\alpha_2$  tels que :

$$\begin{cases} \alpha_1 = \alpha_0 - \varepsilon \frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|} \\ \alpha_2 = \alpha_0 + \varepsilon \frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|} \end{cases} \quad (4.14)$$

## Chapitre 4. Applications des calculs de sensibilité dans une boucle d'optimisation

---

Par identification, on a  $h = \frac{\varepsilon}{\|\nabla J(\alpha_0)\|}$  et  $u = \nabla J(\alpha_0)$ . On a logiquement  $J(\alpha_1) < J(\alpha_0) < J(\alpha_2)$ . Par soustraction des équations issues de (4.14), on trouve que :

$$\alpha_2 - \alpha_1 = 2\varepsilon \frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|} \quad (4.15)$$

D'après les équations (4.13) et (4.14), on en déduit que :

$$\begin{aligned} \langle \nabla J(\alpha_0), u \rangle &= \frac{J(\alpha_0 + hu) - J(\alpha_0 - hu)}{2h} \\ \langle \nabla J(\alpha_0), \nabla J(\alpha_0) \rangle &= \frac{J\left(\alpha_0 + \varepsilon \frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|}\right) - J\left(\alpha_0 - \varepsilon \frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|}\right)}{2 \frac{\varepsilon}{\|\nabla J(\alpha_0)\|}} \end{aligned} \quad (4.16)$$

$$\|\nabla J(\alpha_0)\|^2 = \|\nabla J(\alpha_0)\| \frac{J(\alpha_2) - J(\alpha_1)}{2\varepsilon}$$

$$\|\nabla J(\alpha_0)\| = \frac{J(\alpha_2) - J(\alpha_1)}{2\varepsilon}$$

Dans notre étude, il n'est pas possible de vérifier directement la norme du gradient. En effet, notre calcul de sensibilité s'appuie sur un gradient composé dont l'une des parties – à savoir  $\nabla J(\alpha_0)$  – est issue d'un logiciel tiers. Or, la norme de  $\nabla J(\alpha_0)$  est déterminée pour nos exemples sur l'ensemble du volume fluide. La relation  $\|\nabla J(\alpha_0)\| = \frac{J(\alpha_2) - J(\alpha_1)}{2\varepsilon}$  n'est donc pas valable car on ne sait quelle norme a été utilisée dans le logiciel et le gradient n'est pas surfacique. En revanche, il est possible de vérifier que la direction du gradient donnée par  $\frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|}$  est bonne. Pour cela, on pose  $\alpha = \alpha_2 - \alpha_1$  et on appelle  $\alpha^*$  son vecteur pseudo-inverse :

$$\alpha^* = \frac{\alpha_2 - \alpha_1}{\|\alpha_2 - \alpha_1\|^2} \quad (4.17)$$

On retrouve bien que :

$$\langle \alpha, \alpha^* \rangle = 1$$

On définit ensuite le vecteur  $\Omega$  comme :

$$\Omega = (J(\alpha_2) - J(\alpha_1)) \frac{\alpha_2 - \alpha_1}{\|\alpha_2 - \alpha_1\|^2} \quad (4.18)$$

En développant, on montre que :

$$\begin{aligned}
 \Omega &= (J(\alpha_2) - J(\alpha_1)) \frac{\alpha_2 - \alpha_1}{\|\alpha_2 - \alpha_1\|^2} \\
 &= 2\varepsilon \|\nabla J(\alpha_0)\| \frac{\alpha_2 - \alpha_1}{\|\alpha_2 - \alpha_1\|^2} \\
 &= 2\varepsilon \|\nabla J(\alpha_0)\| \frac{\alpha_2 - \alpha_1}{4\varepsilon^2} \\
 &= \frac{\|\nabla J(\alpha_0)\|}{2\varepsilon} (\alpha_2 - \alpha_1) \\
 &= \frac{\|\nabla J(\alpha_0)\|}{2\varepsilon} 2\varepsilon \frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|} \\
 &= \nabla J(\alpha_0)
 \end{aligned} \tag{4.19}$$

et que :

$$\begin{aligned}
 \|\Omega\| &= \left\| (J(\alpha_2) - J(\alpha_1)) \frac{\alpha_2 - \alpha_1}{\|\alpha_2 - \alpha_1\|^2} \right\| \\
 &= |J(\alpha_2) - J(\alpha_1)| \frac{\|\alpha_2 - \alpha_1\|}{\|\alpha_2 - \alpha_1\|^2} \\
 &= \frac{|J(\alpha_2) - J(\alpha_1)|}{\|\alpha_2 - \alpha_1\|} \\
 &= \frac{J(\alpha_2) - J(\alpha_1)}{2\varepsilon} \\
 &= \|\nabla J(\alpha_0)\|
 \end{aligned} \tag{4.20}$$

Ainsi, la direction de descente obtenue par notre méthode  $\frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|}$  peut donc être vérifiée en calculant la direction  $\frac{\Omega}{\|\Omega\|}$  obtenue par différences finies. On regarde donc :

$$\xi = \frac{\left\| \frac{\Omega}{\|\Omega\|} - \frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|} \right\|}{\left\| \frac{\nabla J(\alpha_0)}{\|\nabla J(\alpha_0)\|} \right\|} \tag{4.21}$$

### Application numérique

La vérification de la direction de descente est montrée ici dans le cas du coude à 90 degrés (cf. partie 4.3). On se donne un jeu de paramètres initial :

$$\alpha_0 = (20.000 \quad 20.000 \quad 20.000)$$

## Chapitre 4. Applications des calculs de sensibilité dans une boucle d'optimisation

Le calcul primaire nous donne une perte de charge  $J(\alpha_0) = 4.841 Pa$  et le solveur adjoint nous donne le gradient :

$$\nabla J(\alpha_0) = - (77.949 \quad 82.236 \quad 80.036 \quad 72.066 \quad 86.434 \quad 72.012 \quad 89.957 \quad 79.236 \quad 78.101 \quad 66.466)$$

On pose  $\epsilon = 0.1$ . On obtient donc par l'équation (4.14) :

$$\begin{cases} \alpha_1 = (20.031 & 20.033 & 20.032 & 20.029 & 20.035 & 20.029 & 20.036 & 20.032 & 20.031 & 20.027) \\ \alpha_2 = (19.967 & 19.967 & 19.968 & 19.971 & 19.965 & 19.971 & 19.968 & 19.968 & 19.969 & 19.973) \end{cases}$$

et le solveur primaire nous donne  $J(\alpha_1) = 4.803 Pa$  et  $J(\alpha_2) = 4.890 Pa$ . Le calcul d'erreur nous donne  $\xi = 0.001\%$ . D'autres points d'initialisation donnent des erreurs équivalentes. L'erreur obtenue permet de valider sur ce cas l'approche retenue et montre que le gradient obtenu est proche de nos attentes.

### 4.1.5 Gestion d'une optimisation multiobjectif

Nous cherchons à résoudre un problème d'optimisation multiobjectif du type :

$$\begin{cases} \min_{\alpha} [J_1(\alpha), \dots, J_m(\alpha)] \\ \text{s.t. } \underline{\alpha}_i \leq \alpha_i \leq \bar{\alpha}_i \quad \forall i = 1, \dots, n \end{cases} \quad (4.22)$$

Nous utilisons par la suite des méthodes d'agrégation afin de se ramener à un problème mono-objectif car . L'approche la plus basique est d'agrèger les objectifs en une somme pondérée :

$$J(\alpha) = \sum_{i=1}^m \omega_i J_i(\alpha) \quad \text{avec } \sum_{i=1}^m \omega_i = 1 \text{ et } \omega_i > 0 \quad \forall i \quad (4.23)$$

En général, la valeur relative des poids reflète l'importance relative des objectifs et est fixée à l'appréciation de l'utilisateur.

Dans nos problèmes, nous cherchons par exemple à minimiser la perte de charge sur chacun des conduits d'un collecteur d'échappement. Idéalement, nous souhaitons uniformiser cette perte de charge sur l'ensemble des conduits. Le maximum de  $\{|J_1|, \dots, |J_m|\}$ , c'est-à-dire la norme infinie sur  $J$  ( $J$  étant pris comme un vecteur), doit être optimisé. Pour éviter la gestion du maximum, nous cherchons à nous rapprocher au plus près du point idéal au sens d'une norme définie  $\|\cdot\|_p$ . Cela nous donne donc :

$$J(\alpha) = \omega_1 |J_1(\alpha)|^p + \dots + \omega_m |J_m(\alpha)|^p \quad \text{avec } \sum_{i=1}^m \omega_i = 1 \text{ et } \omega_i > 0 \quad \forall i \quad (4.24)$$

Cette méthode est connue sous le nom de *weighted exponential sum* [Marler 2004]. En pratique,  $p$  vaut 2 ou 4.

Les techniques d'agrégation supposent que les objectifs soient commensurables. Par exemple, si deux grandeurs différentes telles qu'une perte de charge et une fréquence propre sont à optimiser, on peut utiliser des transformations :

- normaliser par la valeur absolue du critère au point initial  $\alpha_0$  :  $\frac{1}{|J_i(\alpha_0)|}$  ;
- adimensionner par la norme du gradient  $\nabla J_i$  au point initial  $\alpha_0$  :  $\frac{1}{\|\nabla J_i(\alpha_0)\|}$ .

Cela donnerait donc :

$$J(\alpha) = \omega_1 \left| \frac{J_1(\alpha)}{J_1(\alpha_0)} \right|^p + \dots + \omega_m \left| \frac{J_m(\alpha)}{J_m(\alpha_0)} \right|^p \quad (4.25)$$

Le gradient de la fonction objectif est donné par l'équation (4.26) :

$$\nabla J(\alpha) = p \omega_1 \nabla J_1(\alpha) \left| \frac{J_1(\alpha)}{J_1(\alpha_0)} \right|^{p-1} + \dots + p \omega_m \nabla J_m(\alpha) \left| \frac{J_m(\alpha)}{J_m(\alpha_0)} \right|^{p-1} \quad (4.26)$$

## 4.2 Spécificités liées à la CFD

La boucle d'optimisation développée utilise le terme  $\partial_x J$  (cf. équation 4.1). Ce dernier indique dans quelle direction il faut déplacer les nœuds pour obtenir une amélioration de la fonction. Le calcul de ce terme en CFD requiert des paramètres spécifiques. En effet, cette technique est en développement depuis quelques années pour le secteur de l'automobile et par conséquent se limite à des simulations stationnaires avec un modèle de turbulence de type k- $\Omega$  ou k- $\varepsilon$ . En plus de l'utilisation obligatoire d'un « coupled solver » et de la double précision, il est nécessaire d'avoir le type de maillage adéquat et surtout que celui-ci soit de bonne qualité en respectant notamment les critères détaillés dans l'annexe C. Ceux-ci sont préconisés dans la documentation [CD- 2014] de STAR-CCM+<sup>®</sup>. Des cycles d'optimisation du maillage sont utilisés pour que ce dernier atteigne les niveaux d'exigence attendus.

Dans ce manuscrit, nous nous sommes concentrés sur la perte de charge, fonction fréquemment utilisée dans l'automobile pour le dimensionnement des conduits. En outre, les fonctions objectif en CFD portent couramment sur :

- l'uniformité d'un écoulement ;
- une force ;
- un tumble : mouvement de rotation dont l'axe est normal à l'axe du cylindre et dans le plan de symétrie des orifices des soupapes d'admission (en supposant un moteur symétrique à 4 soupapes) ;
- un swirl : mouvement de rotation autour d'un axe parallèle à celui du cylindre ;
- un débit massique.

### Calcul de la perte de charge

Dans cette partie, nous présentons la démarche effectuée par le logiciel pour calculer la perte de charge. Le calcul de la perte de charge s'effectue entre deux groupes de frontières ou d'inter-

faces qui représente respectivement les hautes pressions (désignées par la lettre  $H$ ) et les basses pressions (désignées par la lettre  $L$ ). Le calcul est donc donné par :

$$\Delta P = \left[ \frac{\sum_f |\dot{m}_f| P_f}{\sum_f |\dot{m}_f|} \right]_H - \left[ \frac{\sum_f |\dot{m}_f| P_f}{\sum_f |\dot{m}_f|} \right]_L \quad (4.27)$$

avec  $P_f$  la pression totale et absolue sur chaque face des cellules composant la frontière ou l'interface et  $\dot{m}_f$  le débit massique<sup>c</sup>.

La perte de charge entre l'entrée et la sortie d'une conduite est induite par différents types de pertes de charge classées dans deux catégories :

- les pertes de charge régulières ou par frottement : lorsque le fluide se déplace dans la conduite, il subit une force de frottement sur les parois, celle-ci peut varier en fonction de la rugosité du matériau sur la paroi, de la viscosité du fluide, de la vitesse de l'écoulement.
- les pertes de charge singulières : elles sont essentiellement dues aux accidents de canalisation. Cela concerne les changements géométriques subis par la conduite, que ce soient des changements de direction, de taille de section, la présence d'éléments comme des pompes ou des vannes.

Dans notre cas, nous cherchons à ce que cette valeur soit la plus faible possible. L'étude n'est pas portée sur la rugosité de la surface mais sur la forme de la conduite.

Les simulations nous ont montré que la perte de charge continue d'évoluer même en cas de bonne convergence du calcul primaire. Afin de limiter l'influence des oscillations qui peuvent subvenir durant le calcul de la fonction objectif, nous prenons une moyenne sur un intervalle d'itérations. Cet intervalle, fixé de façon arbitraire, est de 400 itérations.

### 4.3 1<sup>er</sup> cas-test : Coude à 90 °

Le développement de la boucle en CFD s'est fait sur un exemple académique : un tube coudé à 90 degrés (cf. figure 4.2). Ce tube est modélisé par une surface multisection. Chaque section est dimensionnée suivant la valeur de 2 paramètres (cf. figure 4.3), ce paramétrage simple a permis de se concentrer sur les développements nécessaires au bon fonctionnement et à l'automatisation de la boucle.

On cherche à minimiser la perte de charge entre les sections d'entrée et de sortie, on écrit donc le problème d'optimisation suivant :

$$\begin{cases} \min_{\alpha} \Delta P(\alpha) \\ s.t. \underline{\alpha}_i \leq \alpha_i \leq \bar{\alpha}_i \quad \forall i = 1 \dots 10 \end{cases} \quad (4.28)$$

#### 4.3.1 Mises en données

Les mises en données présentent les caractéristiques des étapes de la boucle. Toutes les informations nécessaires à la mise en place du modèle CAO et du calcul physique sont détaillées ci-après.

---

c. Le débit massique permet de tenir compte des variations de volume liées à un fluide compressible.

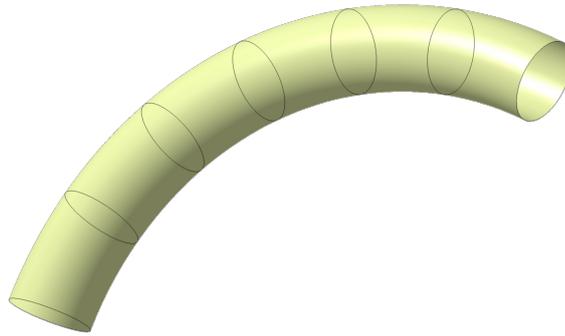


FIGURE 4.2 – Visualisation des sections de la forme coudée

#### 4.3.1.1 CAO

Le coude est modélisé par une surface passant par 7 sections visibles sur la figure 4.2. Nous avons deux paramètres par section. Ils permettent de décentrer la section par rapport à la fibre neutre (vers le haut : paramètre A ou vers le bas : paramètre B) tout en modifiant le diamètre de la section circulaire comme on peut le voir sur les figures 4.3.

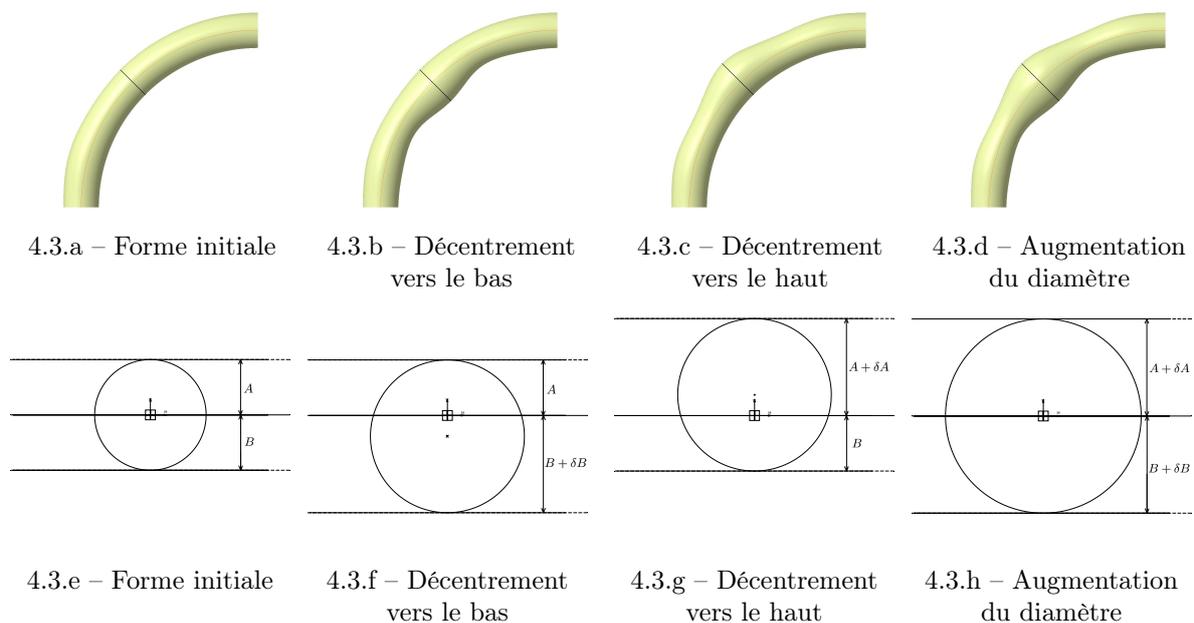


FIGURE 4.3 – Exemple sur un coude du paramétrage à 2 paramètres par section

Afin de ne pas modifier les conditions à l'entrée et à la sortie du coude, les paramètres de la première et de la dernière section sont bloqués pendant toute l'optimisation. De manière analogue, on aurait pu aisément imposer une tangence à ces extrémités si le cahier des charges l'avait imposée. La CAO permet de mettre toute l'intelligence liée aux processus de fabrication. Au final, l'optimisation est menée sur 10 paramètres. Les valeurs du tableau 4.2 sont appliquées à toutes les sections pour la géométrie initiale.

Paramètres	Décentrement vers le haut (A)	Décentrement vers le bas (B)
Valeur nominale	20 mm	20 mm
Borne inférieure	5 mm	5 mm
Borne supérieure	45 mm	45 mm

Tableau 4.2 – Table du paramétrage initial des sections

Ce coude présente l'avantage d'avoir un taux de reconstruction de 100%. Ainsi, il n'existe pas de zones interdites sur l'espace de conception, ce qui nous permet de nous affranchir des bogues que cela aurait pu engendrer (cf. partie 4.1.2).

### 4.3.1.2 Calcul physique

Le coude à 90 degrés est une géométrie simple sur laquelle on observe facilement la perte de charge. On retrouve ce type de géométrie dans de nombreuses applications. L'optimisation a tourné avec la version 10.06 du logiciel STAR-CCM+® avec les configurations suivantes.

#### Modèle physique

Le modèle physique utilisé est constitué des caractéristiques suivantes :

- o calcul 3D ;
- o stationnaire ;
- o solveur couplé : ce modèle résout simultanément les équations de conservation de masse, d'énergie en utilisant une approche pseudo-temporelle ;
- o gaz idéal ;
- o modèle Reynolds-averaged Navier-Stokes (RANS) ;
- o modèle de turbulence  $k-\varepsilon$ .

#### Nombre d'itérations

Pour cette étude, le nombre de Courant (CFL) est laissé à 20 tout au long de la boucle et le CCA est activé pendant 200 itérations. Le calcul adjoint étant sensible à la bonne convergence du calcul primaire, le nombre d'itérations maximum est fixé à 900<sup>d</sup>. En ce qui concerne le solveur adjoint, le nombre d'itérations est fixé à 1100, ce qui fait un total de 2000 itérations pour le calcul physique.

#### Conditions aux limites

Afin de s'affranchir des problèmes de stabilité, il est préférable d'imposer un débit massique comme condition limite en entrée (au lieu d'une vitesse). Nous cherchons donc le débit massique correspondant à une vitesse  $\bar{U}$  de 5 m.s<sup>-1</sup>. Pour cela, on utilise les formules suivantes :

$$Q = \bar{U} \times S \tag{4.29}$$

---

d. Ce nombre est fixé arbitrairement à la suite de calculs préliminaires.

$$\dot{m} = \rho \times \bar{U} \times S \quad (4.30)$$

Pour cette optimisation, on sait que :

$$S_{inlet} = S_{outlet} = \pi r^2 = 1,2566 \times 10^{-3} \text{ m}^2 \quad (4.31)$$

et

$$\rho_{air} = 1,18415 \text{ kg.m}^{-3} \quad (4.32)$$

On obtient le débit volumique suivant :

$$Q = 0,00628 \text{ m}^3.\text{s}^{-1} \quad (4.33)$$

qui se traduit par un débit massique de :

$$\dot{m} = 0,0074402 \text{ kg.s}^{-1} \quad (4.34)$$

*In fine*, les conditions limites sont :

- en inlet de l'extrusion de la section d'entrée : un débit massique de  $7,44.10^{-3} \text{ kg.s}^{-1}$ , soit une vitesse de  $5 \text{ m.s}^{-1}$  ;
- en outlet de l'extrusion de la section de sortie : une pression libre.

**Remarque 4.3.1** *Nous sommes en présence d'un écoulement turbulent. En effet, le nombre de Reynolds est supérieur à 4000 :*

$$Re = \frac{\rho \bar{U} L}{\mu} = \frac{\bar{U} L}{\nu} \simeq 13000 \quad (4.35)$$

où la viscosité cinématique  $\nu$  est donnée par  $\nu = \frac{\mu}{\rho} = 15,6.10^{-6} \text{ St}$  à  $25^\circ \text{C}$  et la dimension caractéristique  $L$  est le diamètre hydraulique, soit  $L = 40 \text{ mm}$ .

#### 4.3.1.3 Configuration des maillages

Le maillage du coude, présenté par les figures 4.4, est constitué de cellules polyédriques avec une taille de base égale à 1,9 mm. Des extrusions ont été utilisées pour stabiliser le fluide en entrée et en sortie (mises en évidence sur la figure 4.4.b). Elles font 200 mm de long, ce qui correspond à environ 5 fois le diamètre de leurs sections respectives. Elles sont composées de 55 couches avec un ratio<sup>e</sup> de 2 afin d'assurer la cohérence du maillage au niveau des interfaces entre les extrusions et le coude. La couche limite du maillage est composée de 8 couches avec un coefficient d'étirement de 1,3 et une hauteur totale de 33% de la taille de base du maillage, soit environ 0,6 mm.

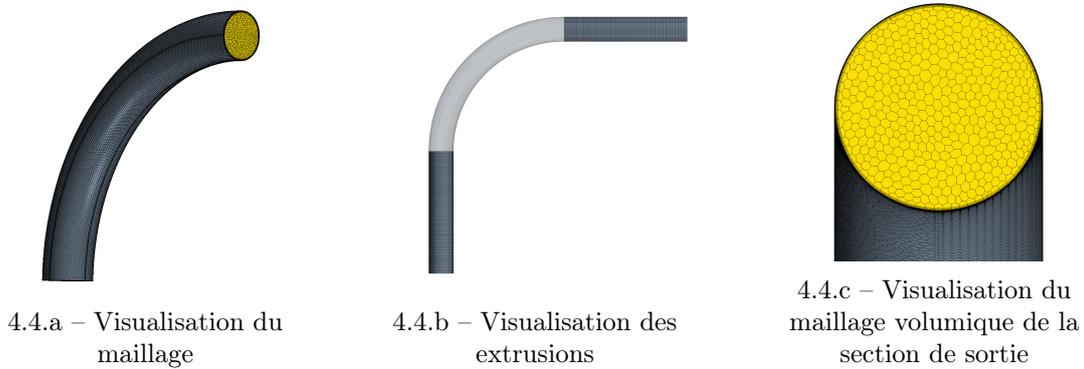


FIGURE 4.4 – Visualisations du maillage du coude

Afin d'obtenir une qualité de cellules permettant d'assurer la validité des gradients, il est nécessaire que le maillage respecte les critères détaillés dans l'annexe C.1. Pour cela, on effectue 4 cycles d'optimisation du maillage, ce qui conduit à un nombre de cellules d'environ 320000 qui respectent au mieux les contraintes de qualité de maillage.

Pour le calcul de sensibilité de forme, nous utilisons une taille de cellules équivalente à celle du calcul physique, c'est-à-dire 1,9 mm. Cette taille de cellule se traduit par des maillages surfaciques contenant environ 26000 éléments.

### 4.3.2 Post-traitement et résultats

*Dans cette section, pour faciliter la comparaison des résultats, les échelles des post-traitements sont fixes entre les deux états d'une même caractéristique étudiée ( $P$ ,  $v...$ ).*

Nous nous intéressons à la bonne convergence des résidus mais aussi à celle de la perte de charge. Celle-ci peut continuer d'osciller même si la convergence du solveur primaire est parfaite. Ainsi, nous utilisons une moyenne de la fonction objectif sur les 400 dernières itérations afin de prévenir tout désagrément. Sur la figure 4.5, les résidus du calcul d'initialisation de la boucle montrent la bonne convergence des calculs primaire et adjoint. On remarque aussi que l'on a gardé une marge au niveau du nombre d'itérations permettant de prévenir les allongements éventuels de durée de convergence au cours de l'optimisation.

Lors de l'initialisation de la boucle, on obtient une perte de charge de 4,86 Pa. Après 11 itérations, l'évolution de la forme par le biais des paramètres CAO a permis d'obtenir une perte de charge de 2,45 Pa (voir figure 4.6 et tableau 4.3). Le temps moyen pour une itération de boucle est d'environ 25 minutes sur 16 cœurs. L'optimisation a permis un gain de l'ordre de 49% pour un temps total d'environ 5 heures.

Les figures 4.7 sont les visualisations des gradients CFD à l'état initial et final. Sur la figure 4.7.a, on observe que les gradients sont uniformément répartis tout au long du tube et qu'ils sont relativement importants comparé à ceux obtenus à l'état final (figure 4.7.b). Cela montre la

---

e. Dans cette mise en données, ce ratio est fixé entre l'épaisseur de la première couche de l'extrusion et la dernière.

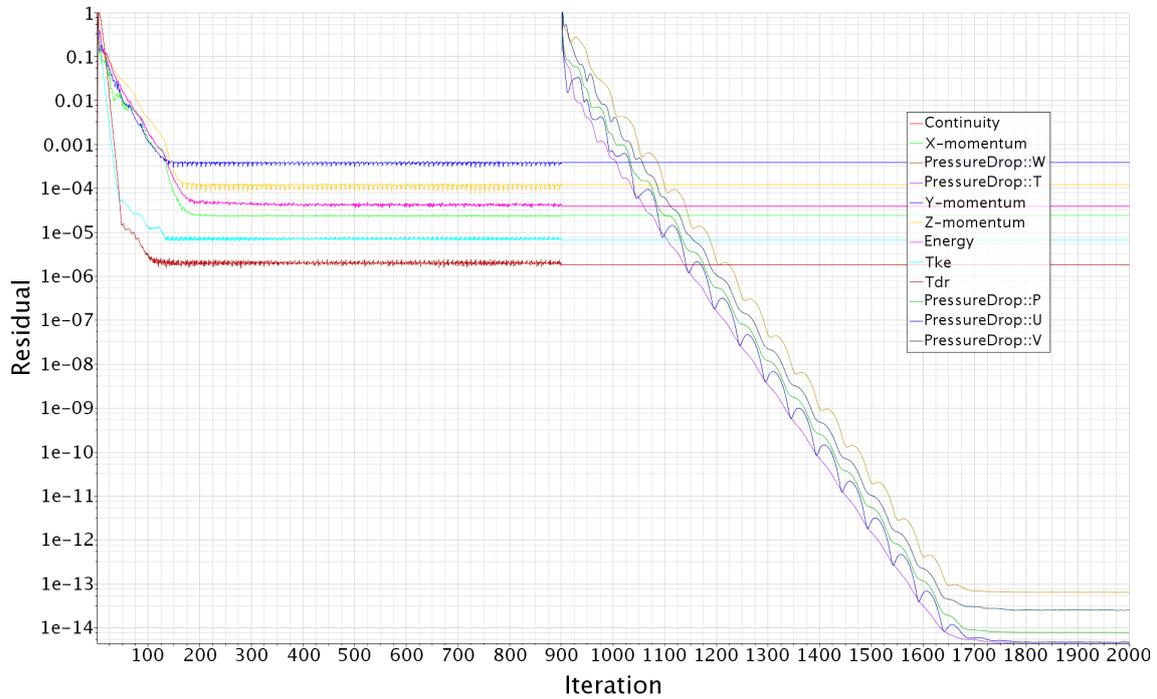


FIGURE 4.5 – Visualisation des résidus du calcul d’initialisation  
*Itérations : solveur primaire = [1; 900] / moyenne sur la perte de charge = [500; 900] / solveur adjoint = [901; 2000]*

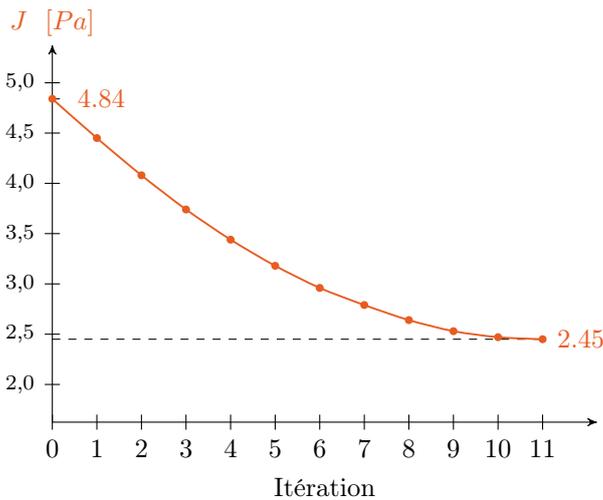


FIGURE 4.6 – Évolution de la perte de charge

Iter	$J$ [Pa]	$\Delta J$ [%]
0	4,84	–
1	4,45	8,2
2	4,08	8,3
3	3,74	8,2
4	3,44	8,0
5	3,18	7,7
6	2,96	6,7
7	2,79	6,0
8	2,64	5,3
9	2,53	4,0
10	2,47	2,5
11	2,45	0,7

Tableau 4.3 – Pourcentage d’amélioration par itération

bonne convergence globale de l'algorithme d'optimisation (cf. figure 4.6). À l'état final, on observe des valeurs de gradients importantes au niveau de la section d'entrée du coude. Le solveur adjoint aimerait modifier la section à ce niveau afin d'améliorer la perte de charge. Toutefois, ces valeurs ne sont pas prises en compte lors de la génération de la nouvelle forme par l'optimiseur étant donné que les paramètres de cette section sont bloqués, ce qui est l'un des points forts de notre méthode. On peut observer sur la figure 4.7.a que les gradients sont orientés vers l'intérieur du tube ce qui suppose de tirer la surface vers l'extérieur pour diminuer la perte de charge.

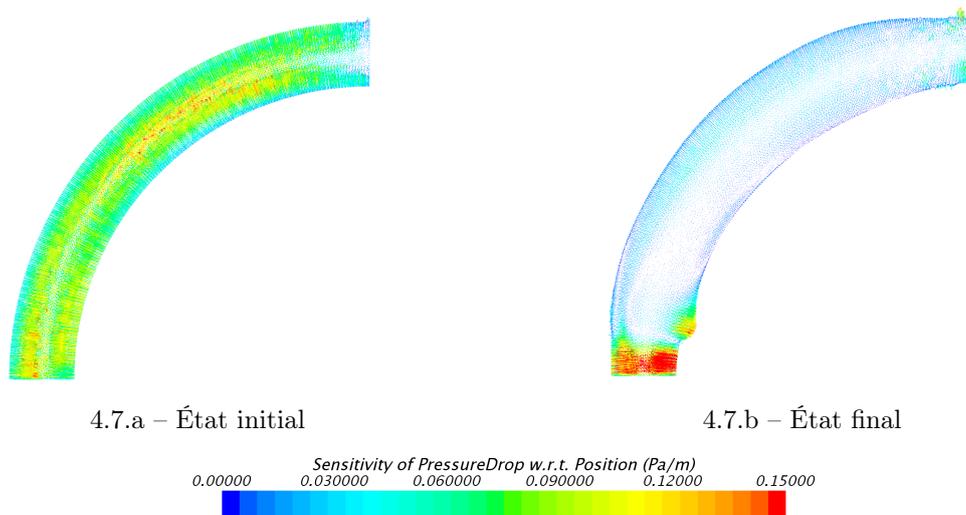
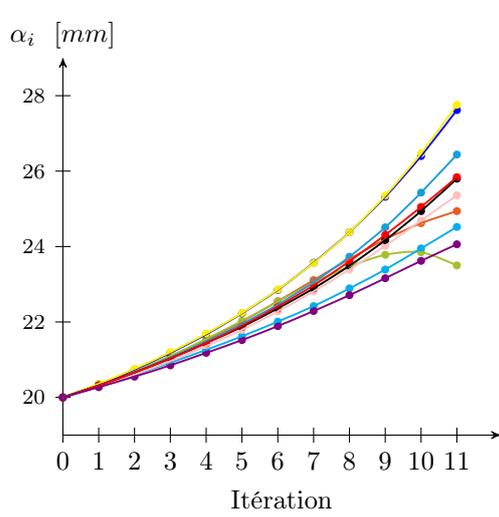


FIGURE 4.7 – Visualisation des gradients CFD à l'état initial et à l'état final

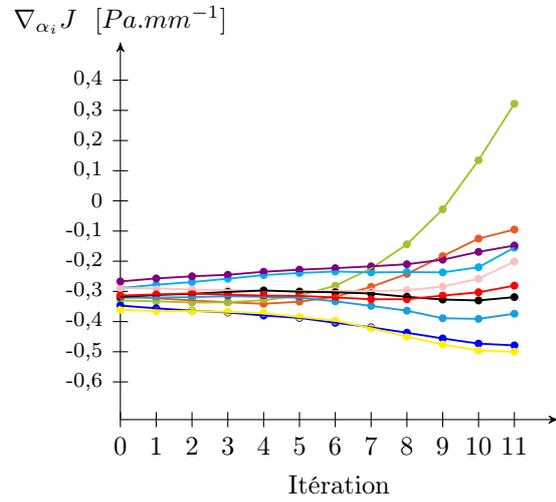
Les graphiques des figures 4.8 montrent l'évolution de chaque paramètre au fil des itérations. Il est possible de comparer aisément l'évolution de ces paramètres car ils sont tous du même type et avec le même ordre de grandeur. Sur la figure 4.8.a, on observe qu'ils évoluent tous dans le même sens dans un premier temps pour ensuite subir des variations à la fin de l'optimisation. L'orientation des gradients observée sur l'image 4.7 est confirmée par la courbe de la figure 4.8.b. Les valeurs de gradient négatives traduisent une augmentation de la valeur du paramètre, ce que l'on observe avec la quasi-totalité des paramètres (cf. figure 4.8.a). Le passage en positif d'un gradient (cf. la courbe verte sur la figure 4.8.b) se traduit par la diminution de la valeur du paramètre.

On observe sur la figure 4.8.a qu'aucun des paramètres n'atteint les bornes fixées dans l'environnement CAO.

Sur les figures 4.9, on observe que la surface reste lisse grâce à la méthode de construction utilisée dans l'environnement CAO. Dans le cadre d'une optimisation par morphing de maillage, le respect de ces mêmes contraintes n'aurait pas été assuré. Qui plus est, l'utilisation d'un opérateur de lissage de type Laplacien n'aurait amélioré que partiellement ces défauts.



4.8.a – Évolution des paramètres



4.8.b – Évolution du gradient de chaque paramètre

FIGURE 4.8 – Courbes de valeurs de paramètres et de leur gradient



FIGURE 4.9 – Visualisation de l'évolution de la forme entre la 1<sup>re</sup> itération et la dernière

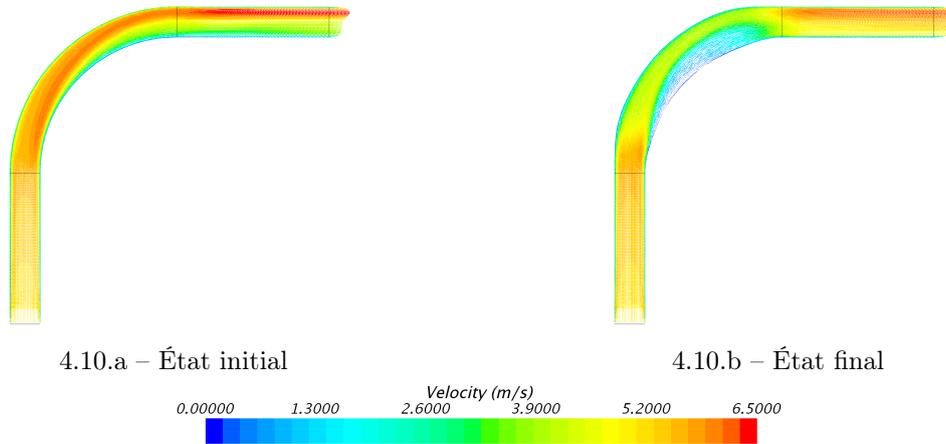


FIGURE 4.10 – Visualisation des vecteurs vitesse à l'état initial et à l'état final. On observe que le champ des vecteurs vitesse tend à s'uniformiser au fil des itérations.

### 4.3.3 Comparaison avec une méthode d'optimisation par plan d'expériences et surface de réponse

La méthode basée sur le gradient utilisée précédemment est opposée à la méthode par plan d'expériences et surface de réponse décrite dans la partie 1.2. Afin de comparer ces deux méthodes, nous avons conduit une campagne de calcul reposant sur une surface de réponse. Cette comparaison permet de comprendre les avantages et les inconvénients liés à chacune des deux méthodes sur un cas concret.

#### 4.3.3.1 Choix du plan, du modèle de la surface et de l'algorithme

Comme nous l'avons vu dans la partie 1.2.3, une optimisation par plan d'expériences et surface de réponse est une succession de différentes étapes. Tout d'abord, la sélection du plan est effectuée de façon cohérente avec le type de géométrie. Dans le cas d'une géométrie ayant des zones interdites (c'est-à-dire un taux de reconstruction inférieur à 100%), les plans optimaux [Benoist 1994], comme par exemple le plan D-optimal, s'adaptent à des domaines de conception dont certaines plages de valeurs sont interdites car ils prennent en compte des contraintes sur les paramètres d'entrée. Des essais préalables sur la géométrie de ce cas test ont montré que son taux de reconstruction est de 100%. Dans ce cas, l'utilisation de plans latin hypercube sampling (LHS), orthogonaux ou sursaturés est possible. Nous choisissons un plan LHS à un niveau. Ce plan explore l'ensemble du domaine admissible et son implémentation est aisée.

Toutefois, s'il n'est pas évident de déterminer *a priori* le taux de reconstruction ou encore les zones interdites, l'utilisation de plans optimaux est conseillée car ils sont moins affectés en cas d'échec d'essais.

Le nombre de points que le plan contient est ensuite déterminé en fonction du nombre de coefficients à évaluer. Le modèle que l'on utilise est un modèle polynomial d'ordre 2 donné par l'équation (4.36). Assez répandu en ingénierie, ce modèle simple permet de prendre en compte les interactions entre les paramètres tout en étant cohérent avec la non-linéarité de la physique

étudiée.

$$y = \beta_0 + \sum_i \beta_i x_i + \sum_i \beta_i x_i^2 + \sum_{i < j} \beta_{ij} x_i x_j + \varepsilon \quad (4.36)$$

Le nombre de coefficients à estimer pour un modèle quadratique – en notant  $p$  le nombre de paramètres – est donné par l'équation (4.37).

$$\underbrace{1}_{\text{constante}} + \underbrace{2}_{\text{termes quadratiques}} + \underbrace{p}_{\text{termes linéaires}} + \underbrace{\frac{p(p-1)}{2}}_{\text{interactions}} \quad (4.37)$$

Avec les 10 paramètres du cas présent, cela fait donc 66 coefficients à estimer. Nous choisissons donc d'effectuer un plan **LHS** contenant 100 points de calcul et ainsi de prendre une marge au niveau du nombre de points pour améliorer l'exploration du domaine de conception admissible. Suite au tirage du plan, les positions des points ont été modifiées dans le but de les optimiser vis-à-vis d'un critère *maximin* [Johnson 1990] qui consiste à maximiser la distance minimale entre les points pour qu'il soient mieux répartis dans l'espace.

Nous résolvons le modèle polynomial par une régression linéaire. Le fonctionnement de cette méthode est détaillé dans la documentation [Pedregosa 2011]. Par exemple, on se donne un modèle polynomial donné par l'équation (4.38) avec deux paramètres  $x_1$  et  $x_2$ .

$$\hat{y}(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + \beta_4 x_1^2 + \beta_5 x_2^2 \quad (4.38)$$

Si on effectue ensuite le changement de variable donné par l'équation (4.39), on obtient le modèle linéaire de l'équation (4.40).

$$z = [x_1, x_2, x_1 x_2, x_1^2, x_2^2] \quad (4.39)$$

$$\hat{y}(x) = \beta_0 + \beta_1 z_1 + \beta_2 z_2 + \beta_3 z_3 + \beta_4 z_4 + \beta_5 z_5 \quad (4.40)$$

Il est donc aisé par ce changement de variable d'estimer rapidement les valeurs des coefficients  $\beta_i$ .

Les calculs **CFD** sont ensuite effectués sur chacun des points du plan. Il faut noter ici que seule l'exécution du solveur primaire est nécessaire pour obtenir la valeur de la perte de charge. Suite à ces calculs, le modèle polynomial permet de construire le modèle statistique, appelé aussi surface de réponse.

Afin de valider les coefficients et donc la qualité de la surface de réponse, on effectue une vérification par validation croisée (plus souvent appelée « cross validation ») et plus particulièrement une « **leave-one-out cross-validation (LOOCV)** » [Hastie 2001]. Cela consiste à séparer l'ensemble des points en deux échantillons : l'échantillon d'apprentissage et l'échantillon de test. Le modèle est ensuite bâti sur l'échantillon d'apprentissage et validé sur l'échantillon de test. Dans le cas de la **LOOCV**, l'échantillon de test ne contient qu'un point et l'opération est effectuée autant de fois qu'il y a de points dans le plan en changeant à chaque fois d'échantillon test. Pour chacune

des opérations, on calcule l'erreur entre le modèle construit et l'échantillon test. La moyenne de ces erreurs sur l'ensemble des tests nous informe sur la qualité de la surface de réponse.

Enfin, afin d'exploiter cette surface de réponse, on utilise un algorithme qui extrait des jeux de paramètres qui sont susceptibles de donner des géométries optimales vis-à-vis de la fonction objectif. L'algorithme utilisé est un **Sequential Least Squares Programming** [Nocedal 2006, Jones 2001]<sup>f</sup>. Étant donné que l'initialisation de l'algorithme peut influencer sur les optimum déterminés par l'algorithme, il est lancé en multistart sur les 100 points du plan. Sur l'ensemble des points minimums donnés par l'algorithme, on retient les 10 meilleurs et les calculs physiques sont effectués sur ces points. Ces nouveaux calculs vont permettre de confirmer les valeurs données par l'algorithme et ainsi de trouver le jeu de paramètres correspondant au minimum. Si les jeux de paramètres donnés par l'algorithme ne sont pas satisfaisants, ces nouveaux points sont ajoutés au modèle et on recommence les étapes précédentes afin de déterminer le jeu de paramètres qui donne la géométrie optimale.

### 4.3.3.2 Résultat et comparaison

La mise en données de l'optimisation du plan d'expériences est celle présentée dans la partie 4.3.1.2. Toutefois, l'exécution du solveur adjoint n'est pas nécessaire. En effet, l'information du gradient n'est pas utile dans ce cas. Le temps de chacun des calculs CFD est donc de 11 minutes en moyenne contre 25 minutes lorsque le calcul du solveur adjoint est effectué<sup>g</sup>.

Parmi les points donnés par l'algorithme du plan d'expériences, l'optimum obtenu est de 2,38 Pa. La valeur déterminée par notre méthode est de 2,45 Pa soit un écart d'environ 3%. Environ 130 calculs ont été nécessaires pour déterminer l'optimum par la méthode par plan d'expériences et d'optimisation par méta-modèles contre 12 calculs pour notre méthode. Cela se traduit par un temps d'optimisation d'environ 24 heures<sup>h</sup> pour le plan d'expériences contre 5 heures pour l'optimisation par gradient. Le gain sur la fonction objectif n'est inférieur que de 3% alors que le temps nécessaire pour son obtention est multiplié par 5. Il faut aussi noter que le nombre de paramètres pour cette optimisation est faible et que l'augmentation du nombre de paramètres se traduit par une augmentation importante du nombre de calculs nécessaires pour la construction de la surface de réponse. La différence de temps de calcul pour atteindre de tels résultats avec une géométrie plus complexe augmente de la même façon. Néanmoins, une optimisation par plan d'expériences et surface de réponse permet de trouver des solutions non atteignables par une méthode exacte. L'augmentation de la taille de la population comme un lancement en multistart pourrait permettre de mieux explorer l'espace de conception (cf. partie 1.2).

La figure 4.11 présente le champ de vitesse sur le plan de coupe des deux géométries. On remarque que le comportement du fluide est assez proche entre les deux optima, ce qui confirme la faible différence entre les deux pertes de charge.

---

f. Les détails de l'algorithme sont donnés dans la partie *Optimize* de la documentation [Jones 2001].

g. Les calculs sont effectués sur 16 cœurs avec des tailles de maillage équivalentes.

h. Ce temps n'inclut pas le temps nécessaire à l'optimisation du plan LHS, au choix de la surface de réponse et au choix de l'algorithme.

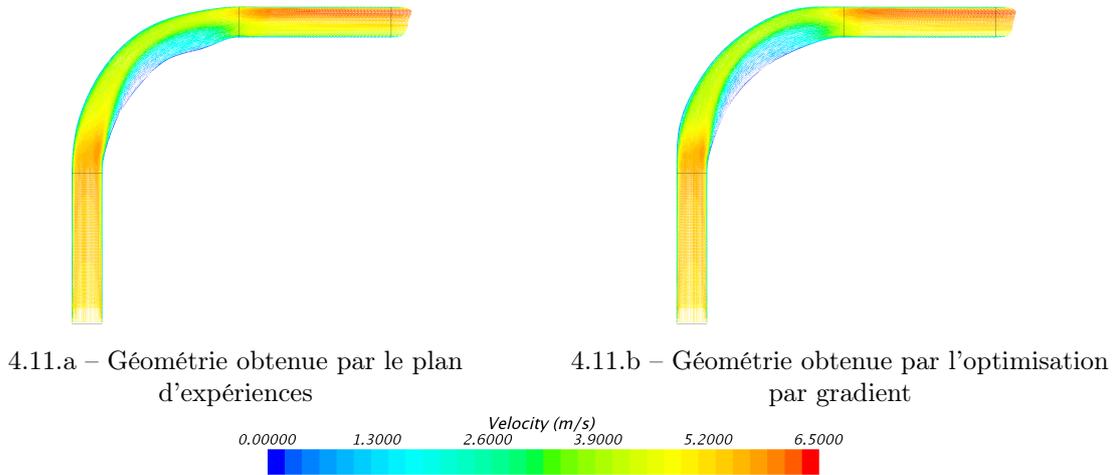


FIGURE 4.11 – Visualisation des vecteurs vitesse

#### 4.4 2<sup>e</sup> cas-test : Collecteur de type industriel

À la suite de l'application présentée dans la partie 4.3, le choix s'est porté sur un collecteur d'échappement. C'est un bon exemple de géométrie dont le nombre de frontières est supérieur à deux. Cela permet non seulement de traiter un cas multifrontière plus complexe mais aussi multiobjectif 4.1.5. En effet, chacune des branches du collecteur peut être traitée indépendamment des autres. Nous limitons le nombre de branches à deux (ce qui correspond à un moteur bicylindre). Plus commune, la version à quatre branches n'a pas été retenue car elle ne présente pas d'intérêts justifiant des efforts supplémentaires en ressource et en temps de calculs.

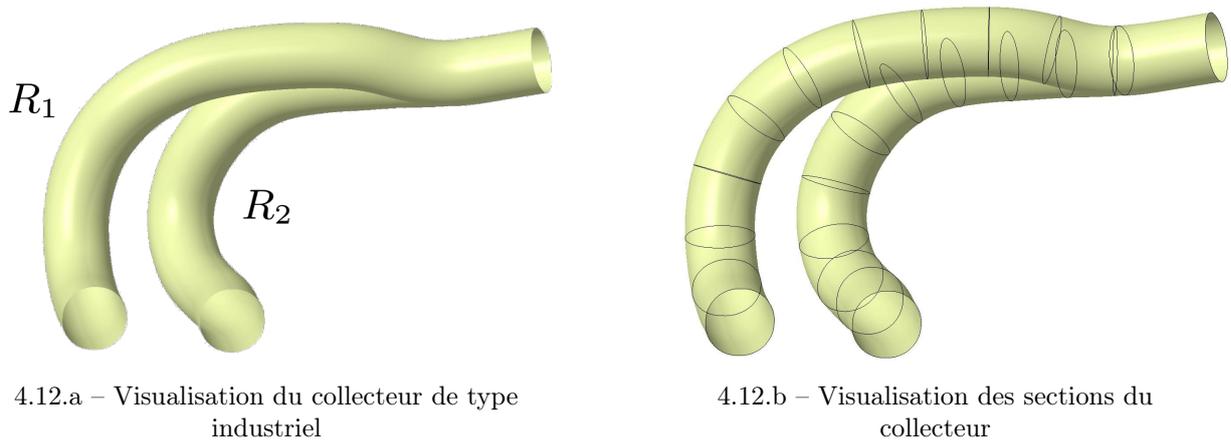


FIGURE 4.12 – Visualisation des collecteurs étudiés

Dans cette section, pour faciliter les explications, les branches sont identifiées. Sur la figure 4.12.a, la branche de gauche est nommée  $R_1$  et celle de droite  $R_2$ . On cherche ici à minimiser la perte de charge de chacune des branches du collecteur. Cette application est une optimisation effectuée sur les deux objectifs suivants :

- $J_{R_1}$  : la perte de charge  $\Delta P$  de la branche  $R_1$  soit  $\Delta P_{R_1}$
- $J_{R_2}$  : la perte de charge  $\Delta P$  de la branche  $R_2$  soit  $\Delta P_{R_2}$

Le problème d'optimisation s'écrit :

$$\left\{ \begin{array}{l} \min_{\alpha} J_{R_1}(\alpha) \\ \min_{\alpha} J_{R_2}(\alpha) \\ \text{s.t. } \underline{\alpha}_i \leq \alpha_i \leq \bar{\alpha}_i \quad \forall i = 1 \dots n \end{array} \right. \quad (4.41)$$

L'agrégation de fonction présentée par la partie 4.1.5 permet de traduire ce problème multiobjectif en un problème mono-objectif. D'après l'équation (4.22), on écrit :

$$J = \frac{1}{2} (J_{R_1}^2 + J_{R_2}^2) \quad (4.42)$$

Le facteur  $1/2$  dans l'équation (4.22) est utilisé uniquement par commodité pour le calcul du gradient.

On peut poser notre problème d'optimisation comme ceci :

$$\left\{ \begin{array}{l} \min_{\alpha} J(\alpha) \\ \text{s.t. } \underline{\alpha}_i \leq \alpha_i \leq \bar{\alpha}_i \quad \forall i = 1 \dots n \end{array} \right. \quad (4.43)$$

#### 4.4.1 Mises en données

Les mises en données présentent les caractéristiques des étapes de la boucle. Toutes les informations nécessaires à la mise en place du modèle CAO et du calcul physique sont détaillées ci-après.

##### 4.4.1.1 Modèle CAO

Le collecteur de la figure 4.12.a est composé de deux surfaces multisections. Celles-ci se rejoignent pour former le collecteur. Chaque branche est composée de 12 sections (cf. figure 4.12.b). Les deux branches partagent la même section en sortie du collecteur. Chacune des sections est pilotée par un jeu de 5 paramètres.

Trois de ces paramètres font varier la taille de la section en jouant sur les rayons (paramètre A sur la figure 4.13) et sur les longueurs des segments (paramètres B et C sur la figure 4.13). De plus, comme on peut le voir à la figure 4.14, ils permettent aussi de passer d'une forme quasi-circulaire, lorsque les valeurs de rayon sont largement supérieures aux valeurs des tailles de segments, à une forme quasi-rectangulaire, lorsque les valeurs de rayon sont largement inférieures. Cela donne une plus grande liberté à l'optimisation. Les deux derniers paramètres pilotent quant à eux le décalage de cette section suivant le repère local du plan auquel elle appartient (paramètres D et E sur la figure 4.13).

Comme nous l'avons vu dans la partie 4.1.1, les bornes de chaque paramètre sont fixées. Dans notre cas, ces valeurs sont choisies de manière arbitraire mais elle peuvent être le support de contraintes architecturale ou de l'environnement dans lequel la pièce est intégrée. De la même

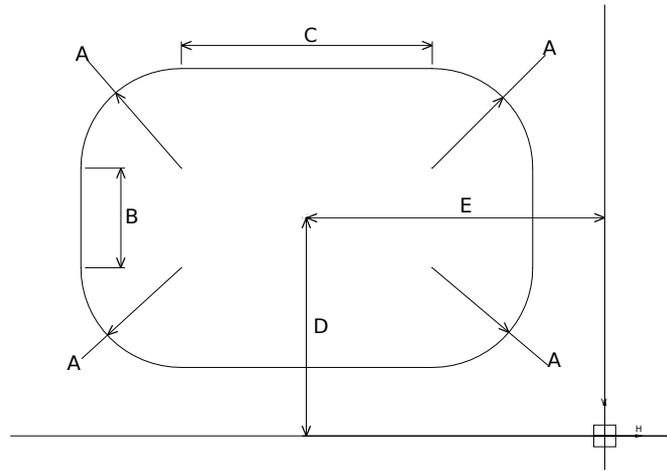


FIGURE 4.13 – Schéma des paramètres d'une section CAO

*A : rayon des angles de la section / B : largeur de la section / C : longueur de la section / D : décentrement suivant l'axe vertical de la section / E : décentrement suivant l'axe horizontal de la section*

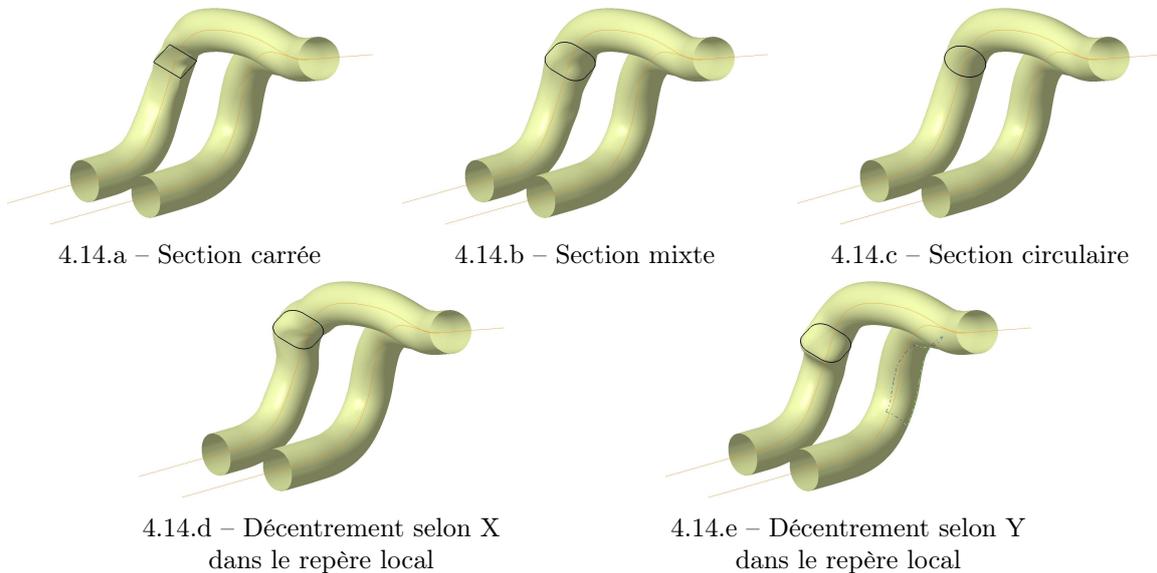


FIGURE 4.14 – Visualisation du paramétrage à 5 paramètres : les sections (en noir) peuvent passer d'une forme circulaire à une forme rectangulaire de manière continue.

## Chapitre 4. Applications des calculs de sensibilité dans une boucle d'optimisation

manière, on peut décider volontairement de ne pas prendre en compte certains paramètres durant l'optimisation, comme ceux des sections d'entrée et de sortie afin de respecter des contraintes géométriques par exemple. De plus, nous imposons à la surface multisection d'être tangente à la normale de ces sections. En résumé, l'optimisation de ce collecteur est effectuée sur 100 paramètres ( $n = 100$ ).

Pour se rapprocher de ce qui est fait actuellement dans l'industrie, les sections sont quasiment circulaires (cf. figure 4.14.c et figure 4.12.a), c'est-à-dire que la valeur des rayons est largement supérieure aux longueurs des segments horizontaux et verticaux comme on peut le voir dans le tableau 4.4.

Paramètres	Rayon (A)	Longueur du segment vertical (B)	Longueur du segment horizontal (C)	Décentrement vertical (D)	Décentrement horizontal (E)
Valeur nominale	20 mm	2 mm	2 mm	100 mm	100 mm
Borne inférieure	1 mm	5 mm	5 mm	80 mm	80 mm
Borne supérieure	40 mm	50 mm	50 mm	120 mm	120 mm

Tableau 4.4 – Table de paramétrage initial des sections

Les valeurs de décentrement sont prises par rapport à une référence décalée de 100 mm pour éviter les confusions entre des valeurs positives et négatives.

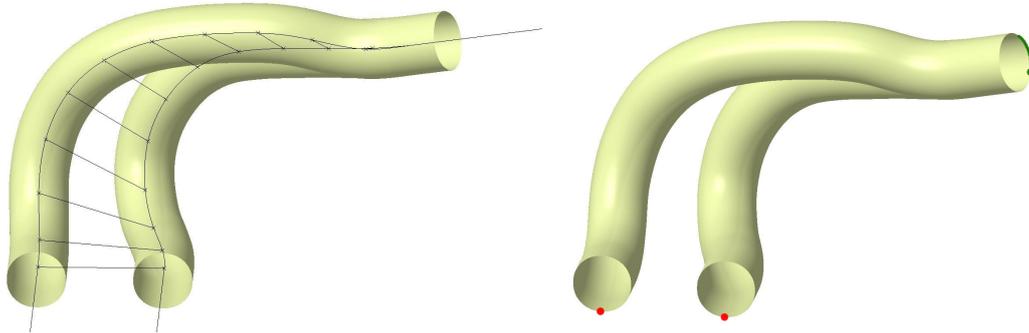
Le paramétrage donné par le tableau 4.4, commun pour toutes les sections à l'état initial de la géométrie donne une aire de section d'environ 1420 mm<sup>2</sup> et un périmètre d'environ 133 mm. Ces valeurs, fixes pour les sections d'entrées et de sortie, sont nécessaires au calcul de la partie 4.4.1.2.

Une des difficultés de cette géométrie est de fixer le repère local de chaque section afin de faire varier de façon homogène les paramètres. Pour cela, nous nous sommes appuyés sur les courbes guides et sur la répartition des sections sur ces courbes. On utilise cette répartition, composée du même nombre de points et faite de façon régulière entre les deux branches, pour relier les points respectifs de chaque branche comme on peut le voir sur la figure 4.15.a. Chaque plan, support des sections, passe par un des points répartis sur la courbe guide et est normal à cette courbe en ce point. Pour l'esquisse de la section, l'axe horizontal est donné par la projection du segment reliant les deux courbes sur le plan dans lequel la section est tracée.

Les éléments de repérage de la frontière extérieure sont placés sur la section de sortie du collecteur. Un point et une courbe sont définis sur la géométrie, en vert sur la figure 4.15.b. Ils permettent de repérer les nœuds dont on déduit le nœud initial et le sens de parcours (cf. partie 2.2.4). Deux points supplémentaires, en rouges sur la figure 4.15.b, sont placés sur chacune des sections d'entrée et identifient les frontières respectives entre les fichiers de maillage.

### 4.4.1.2 Calcul physique

Le cas étudié est assimilé à un collecteur d'échappement (cf. figure 4.12.a). Lorsque le moteur tourne, les soupapes d'échappement de chaque cylindre se coupent alternativement. La simulation



4.15.a – Visualisation du squelette du collecteur

4.15.b – Éléments de repérage des frontières

FIGURE 4.15 – Visualisations particulières du collecteur

sur ce collecteur consiste à avoir une branche active avec un débit élevé tandis que la deuxième branche, celle qui est considérée passive, a un débit largement inférieur à la première mais non nul. Ce débit non nul caractérise un flux résiduel lorsque la soupape est fermée. La simulation de ce flux permet de stabiliser le modèle numérique en évitant les recirculations qui n'auraient pas de sens physique. Dans le but d'optimiser les deux branches, nous appliquons cette mise en données sur les branches tour à tour. Nous effectuons donc deux calculs physiques pour une géométrie.

La méthode d'agrégation présentée dans la partie 4.1.5 permet de traduire en une seule fonction objectif les fonctions objectif calculées pour chacune des simulations. Dans cette partie, les conditions données pour les branches actives et passives s'appliquent alternativement aux branches de la géométrie CAO sans modifications des valeurs.

### Modèle physique

Ce calcul physique est effectué avec le logiciel STAR-CCM+<sup>®</sup>. Le modèle utilisé est de type  $k-\epsilon$  stationnaire basé sur les équations Reynolds-averaged Navier-Stokes. Comme nous l'avons vu dans le paragraphe 4.2, il est nécessaire d'utiliser un solveur « coupled flow » ainsi que la double précision.

**Remarque 4.4.1** *Le modèle  $k-\epsilon$  utilisé est le « realizable  $k-\epsilon$  model ». Il se base sur le modèle  $k-\epsilon$  avec une nouvelle formulation de la viscosité turbulente et une nouvelle équation de transport du taux de dissipation  $\epsilon$ .*

Le fluide utilisé est l'air avec les caractéristiques suivantes :

- Masse volumique :  $\rho = 1,184 \text{ kg.m}^{-3}$
- Viscosité dynamique :  $\mu = 1,8e^{-5} \text{ Pa.s}$

Ces valeurs permettent de calculer le coefficient de viscosité cinématique donné par :

$$\nu = \frac{\mu}{\rho} = 15,2e^{-6} \text{ m}^2.\text{s}^{-1} \quad (4.44)$$

## Chapitre 4. Applications des calculs de sensibilité dans une boucle d'optimisation

---

Des volumes de tranquillisation du fluide sont ajoutés à la géométrie initiale. Ces volumes sont des extrusions des sections d'entrée et de sortie mises en évidence sur la figure 4.18.a. La longueur de ces extrusions est définie en fonction du diamètre hydraulique des sections d'entrée et de sortie. Elle est usuellement comprise entre 5 à 8 fois le diamètre hydraulique. Le diamètre hydraulique représente la longueur caractéristique d'une section. Il est donné par :

$$D_h = \frac{4A}{P} \quad (4.45)$$

Nous avons donc besoin de la valeur de l'aire et du périmètre de la section qui sont fournis par la géométrie CAO de la partie 4.4.1.1. Avec ces valeurs, relevées sur la géométrie, on a  $D_h = 4 \times 1420/133 = 42,70 \text{ mm}$ . Ici, la longueur des extrusions est fixée à 8 fois la valeur du diamètre hydraulique, soit  $L_{extrusion} = 8 \times D_h = 8 \times 42,70 \simeq 340 \text{ mm}$ .

Ce diamètre hydraulique est également nécessaire dans le calcul du nombre de Reynolds afin de déterminer la nature de l'écoulement : laminaire ou turbulent. Cette indication va nous aider à choisir un modèle numérique adapté. Pour une conduite de section circulaire, le passage du laminaire au turbulent se fait entre des valeurs de Reynolds comprises entre 2000 et 3000. Pour chacune des branches, nous avons un débit différent. Nous calculons donc le nombre de Reynolds pour chaque branche en fonction des conditions aux limites définies.

### Conditions aux limites

Les conditions aux limites suivantes sont fixées à l'extrémité de leur extrusion respective.

#### o Entrée de la branche active

Cette branche se situe en sortie du cylindre dont la soupape d'échappement est totalement ouverte. Le débit d'air à l'extrémité de l'extrusion de la branche active est fixé à  $0,05 \text{ kg.s}^{-1}$ . La vitesse moyenne est donnée par :

$$\bar{U} = \frac{\dot{m}}{\rho \times S} = \frac{0.05}{1,184 \times 1,42e^{-3}} = 29,7 \text{ m.s}^{-1} \quad (4.46)$$

Cette vitesse et les données précédentes permettent d'estimer la valeur du nombre de Reynolds donnée par :

$$Re = \frac{\bar{U}L}{\nu} = \frac{29,7 \times 42,7e^{-3}}{15,2e^{-6}} \simeq 84000 \quad (4.47)$$

On a donc un régime fortement turbulent dans la branche active.

#### o Entrée de la branche passive

Contrairement à la précédente, cette branche est passive avec un débit résiduel. Le fluide de la branche active arrive dans la zone de mélange avec une vitesse élevée, si on laisse un fluide statique dans la branche passive, celui-ci peut être mis en mouvement et provoquer des perturbations dans la branche normalement passive. Le fait d'avoir un écoulement déjà présent permet d'éviter ce retour. Le débit d'air dans la branche passive est de  $0,003 \text{ kg.s}^{-1}$ , on peut donc calculer la vitesse moyenne :

$$\bar{U} = \frac{\dot{m}}{\rho \times S} = \frac{0.003}{1,184 \times 1,42e^{-3}} = 1,78 \text{ m.s}^{-1} \quad (4.48)$$

D'où :

$$Re = \frac{\bar{U}L}{\nu} = \frac{1,78 \times 42,7e^{-3}}{15,2e^{-6}} \simeq 5000 \quad (4.49)$$

On en conclut que l'écoulement dans cette branche est à la limite de la turbulence.

Cette entrée n'est pas prise en compte dans le calcul de la perte de charge, on considère que son influence sur le calcul est négligeable du fait de son débit réduit.

◦ **Sortie du collecteur**

Du côté de la sortie du collecteur, la section à l'extrémité de l'extrusion est une sortie de pression libre.

Pour cette optimisation, la perte de charge que nous utilisons comme fonction objectif correspond à la différence de pression entre l'entrée de la branche active et la sortie du collecteur (cf. partie 4.2).

**Nombre d'itérations**

Les observations effectuées lors des calculs préparatoires nous ont renseignés sur le comportement de la convergence de chacun des solveurs. Les optimisations effectuées ont montré que la durée de convergence peut s'allonger au fur et à mesure de l'optimisation. Il faut donc prendre en compte cette augmentation dans la mise en données de départ, sa modification étant difficile au cours de l'optimisation.

Pour le solveur primaire, le nombre d'itérations est de 5000 avec un nombre de Courant fixé à 50, ce qui nous assure à la fois la convergence des résidus mais aussi un nombre suffisant d'itérations pour le calcul de la perte de charge une fois que les résidus ont convergé. Concernant le solveur adjoint, le nombre d'itérations est fixé à 10000. Le nombre total d'itérations du calcul physique est donc de 15000. La figure 4.16 nous donne un aperçu de la convergence des résidus du calcul initial de la branche  $R_1$ .

L'obtention du gradient demande de prêter une attention particulière à la qualité de maillage et à la bonne convergence du solveur. Il faut noter que ces deux conditions sont le plus souvent liées.

**4.4.1.3 Configuration des maillages**

Nous portons une attention particulière à la qualité du maillage qui est nécessaire à la convergence du solveur adjoint et donc à l'obtention du gradient. Les valeurs sont fixées pour respecter au mieux les conditions de maillage requises (cf. annexe C.1).

◦ **Maillage de la branche active**

Ces conditions s'appliquent non seulement à la branche active mais aussi à la jonction entre les deux branches (représentée par la zone verte sur la figure 4.17.b). La taille du maillage dans cette branche est fixée à 1.2 mm et constitue notre taille de base, cela permet de fixer les valeurs des différentes zones de maillage en fonction de cette taille de référence.

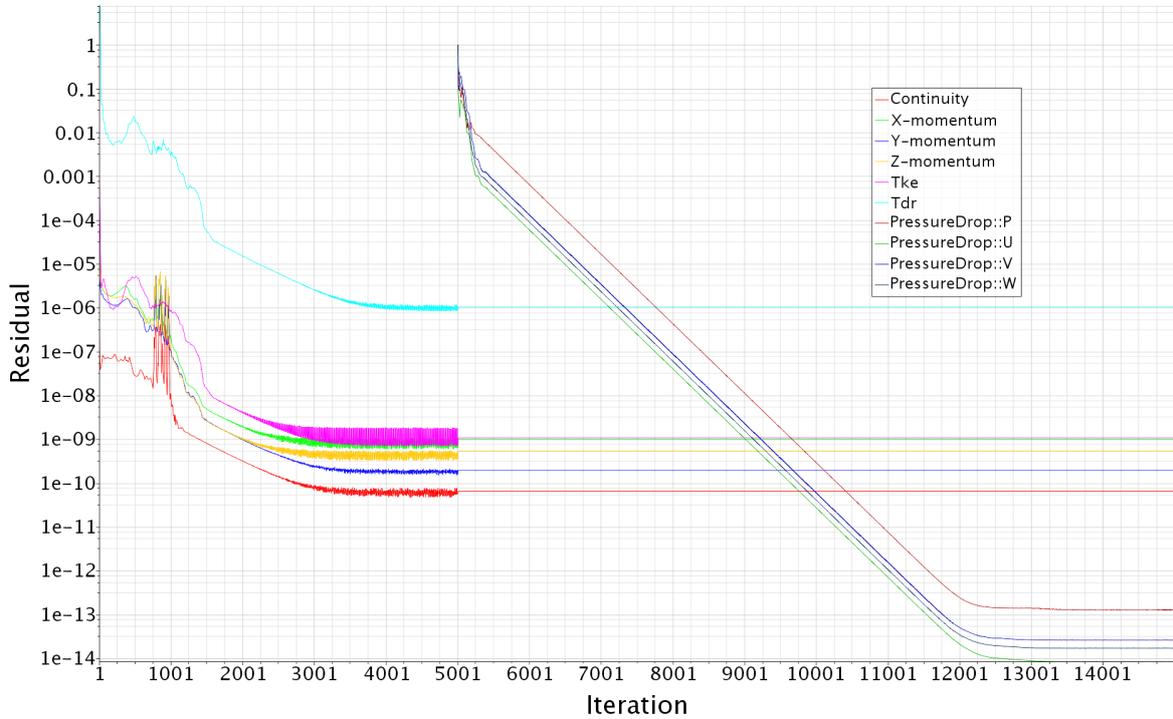
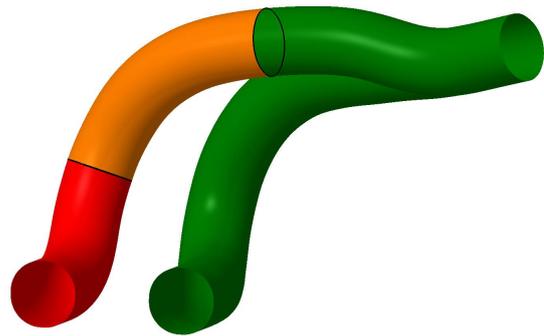


FIGURE 4.16 – Visualisation des résidus du calcul d’initialisation - branche  $R_1$  active  
 Itérations : solveur primaire = [1; 5000] / moyenne sur la perte de charge = [4600; 5000] /  
 solveur adjoint = [5001; 15000]



4.17.a – Visualisation du maillage



4.17.b – Visualisation des différentes zones de maillage : zone active en vert | zone de transition en orange | zone passive en rouge

FIGURE 4.17 – Visualisation du maillage du collecteur - branche  $R_2$  active

En mécanique des fluides, la qualité de la couche limite est primordiale pour un calcul. Dans la branche active, nous avons fixé à 10 le nombre de couches de prismes pour une hauteur totale de 0,72 mm soit 60% de la taille de base du maillage. Le coefficient d'étirement des couches de prismes est fixé à 1,2 quelle que soit la zone de maillage. Ces paramètres permettent d'avoir une taille proche des cellules à l'interface entre les couches limites et le reste du maillage.

Le maillage des extrusions est basé sur celui des interfaces et extrudé selon une direction normale à ces interfaces. Les paramètres requis pour le maillage des extrusions sont le nombre de couches composant l'extrusion et le coefficient d'extension de celle-ci. Ce coefficient permet de fixer l'épaisseur d'une couche par rapport à celle qui la précède. Ce coefficient est fixé de telle sorte que l'épaisseur augmente en s'éloignant de la section extrudée. Dans le cas de l'extrusion à l'entrée de la branche active et à la sortie du collecteur, le nombre de couches est fixé à 140 et le coefficient est de 1,01. Cela permet d'avoir des tailles de cellules assez proches d'un côté et de l'autre de la section extrudée tout en limitant le nombre de cellules nécessaires pour le maillage des extrusions.

##### o **Maillage de la branche passive**

Dans cette branche, on change de conditions de maillage afin d'avoir une configuration de maillage en adéquation avec la physique et le nombre de Reynolds déterminé précédemment. En effet, le modèle **RANS** repose sur une hypothèse de volume représentatif sur lequel les équations de Navier-Stokes sont moyennées. Si le maillage est trop fin, des instabilités de l'écoulement sont capturées et peuvent ralentir voire même perturber la convergence. Les valeurs de maillage ont donc été adaptées pour cette portion de branche (représentée en rouge sur la figure 4.17.b).

La taille de cellule dans cette zone est relative à la taille de base. Cette taille est fixée à 160% de la taille de base, c'est-à-dire aux alentours de 1,9 mm. La couche de prismes est composé de 5 couches et sa hauteur totale est fixée à 5,5 mm.

Concernant le maillage de l'extrusion de la section de la branche passive, le nombre de couches est fixé à 100 avec un coefficient de 1,01.

##### o **Zone de transition**

Les valeurs entre les deux maillages décrites précédemment ont un écart relativement important. Il n'est pas judicieux de laisser une telle différence de maillage, d'autant plus que ce changement de maillage s'effectue dans la zone de mélange des deux branches, juste avant la partie commune, on risque donc d'ajouter des difficultés de convergence. Afin de rendre le changement des caractéristiques du maillage moins brutal, un volume de transition est défini entre les différents maillages (en orange sur la figure 4.17.b).

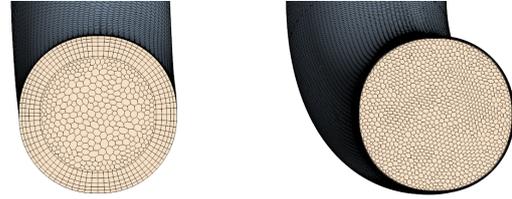
Même si l'on cherche à réduire l'écart entre les zones de la branche active et passive, les valeurs de maillage doivent être adaptées au faible débit car cette zone de transition se situe dans la branche passive.

La taille de cellule est fixée à 140% de la taille de base du maillage soit environ 1,7 mm. La couche de prismes est composée de 8 couches et sa hauteur totale est fixée à 5,0 mm.

En plus des caractéristiques de maillage ci-dessus, 8 cycles d'optimisation de maillage ont été utilisés. Ces cycles permettent de s'approcher de la qualité de maillage nécessaire et en particulier



4.18.a – Visualisation des extrusions



4.18.b – Visualisation du maillage des sections d'entrée

FIGURE 4.18 – Visualisation du maillage du collecteur - branche  $R_2$  active

d'une "qualité minimum de face" de 0,05 et d'un "seuil de qualité" de 0,9. D'autres critères sont présentés dans l'annexe C.

Pour des questions de comparaison, les valeurs présentées dans les paragraphes précédents sont reportées dans le tableau 4.5.

Zone	Branche active	Zone de transition	Branche passive
<b>Taille de maillage</b>	1,2 mm (base)	1,68 mm (140 %)	1,92 mm (160 %)
<b>Couche limite</b>			
Hauteur de couche limite	0.72 mm (60 %)	5,0 mm (absolue)	5,5 mm (absolue)
Nombre de couche limite	10	8	5
Coefficient d'extension	1,2	1,2	1,2
<b>Extrusion</b>			
Nombre de couches	140	–	100
Coefficient d'extension	1,01	–	1,01

Tableau 4.5 – Configurations du maillage pour le collecteur

Les valeurs précédentes concernent uniquement le maillage de calcul à destination des solveurs primaire et adjoint. Le calcul de sensibilité ne requiert pas la même qualité de maillage et il se base uniquement sur le maillage surfacique des formes. Afin d'alléger le processus de calcul de sensibilité, nous avons choisi de ne pas utiliser le maillage du calcul physique dans le calcul de sensibilité. Nous remaillons donc l'ensemble des géométries avec une taille de base fixée à 3,5 mm. Cette taille de base se traduit par des maillages surfaciques composés d'un peu plus de 32000 cellules.

Suite à la création de l'ensemble des mises en données, l'optimisation est effectuée et les résultats sont présentés dans la partie suivante.

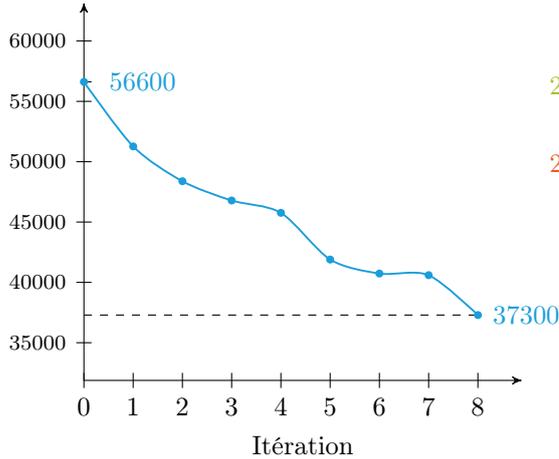
#### 4.4.2 Post-traitement et résultats

*Dans cette section, pour faciliter la comparaison des résultats, les échelles des post-traitements sont fixes entre les deux états d'une même caractéristique étudiée (P, v...).*

#### 4.4. 2<sup>e</sup> cas-test : Collecteur de type industriel

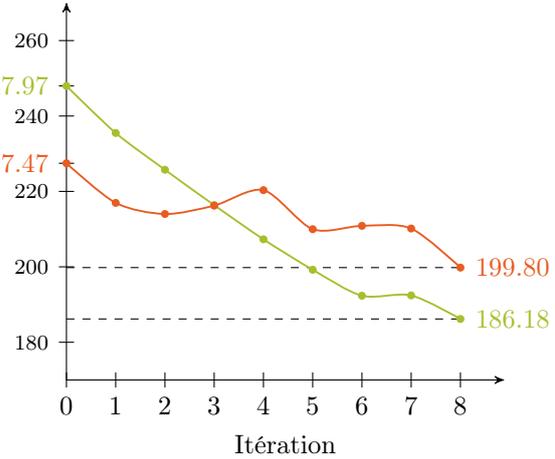
L'optimisation de ce collecteur porte sur la minimisation de la perte de charge de chacune des branches (cf. équations (4.42) et (4.43)). À l'état initial, la perte de charge de la branche  $R_1$  est d'environ 248 Pa et celle de la branche  $R_2$  est d'environ 227 Pa, ce qui nous donne une fonction objectif d'environ 56600 Pa<sup>2</sup>. L'évolution de la fonction objectif au cours des 8 itérations est donnée par la courbe de la figure 4.19.a.

Fonction objectif  $J$  [Pa<sup>2</sup>]



4.19.a – Évolution de la fonction objectif

$J_{R_1}$  [Pa]  $J_{R_2}$  [Pa]



4.19.b – Évolution des pertes de charge

FIGURE 4.19 – Évolutions de la fonction objectif et des pertes de charge des branches du collecteur

La valeur obtenue au bout des 8 itérations est d'environ 37300 Pa<sup>2</sup>, ce qui représente une amélioration de 34% de la valeur initiale de la fonction objectif. Toutefois, ce sont les évolutions de chacune des pertes de charge qui nous intéressent ici.

L'évolution de la perte de charge de chaque branche est donnée par la figure 4.19.b. On observe une amélioration de près de 25% pour la branche  $R_1$  avec une valeur finale d'environ 186 Pa et de plus de 12% pour la branche  $R_2$  avec une perte de charge de moins de 200 Pa. On observe que l'optimiseur pénalise aux itérations 3 et 4 une perte de charge pour améliorer l'autre tout en améliorant la fonction objectif globale (cf. figure 4.19 et tableau 4.6).

Itération	$J$	$\Delta J$ [%]	$R_1$		$R_2$	
			$J_{R_1}$ [Pa]	$\Delta J_{R_1}$ [%]	$J_{R_2}$ [Pa]	$\Delta J_{R_2}$ [%]
0	56614	–	247,97	–	227,47	–
1	51266	9,45	235,50	5,03	216,96	4,62
2	48382	5,63	225,75	4,14	214,01	1,36
3	46785	3,30	216,36	4,16	216,24	- 1,04
4	45762	2,19	207,30	4,19	220,34	- 1,90
5	41892	8,46	199,23	3,89	209,98	4,70
6	40731	2,77	192,34	3,46	210,87	- 0,42
7	40598	0,33	192,43	- 0,05	210,17	0,33
8	37291	8,15	186,18	3,25	199,80	4,93

Tableau 4.6 – Pourcentage d'amélioration de la fonction objectif et des pertes de charge

## Chapitre 4. Applications des calculs de sensibilité dans une boucle d'optimisation

La configuration de maillage fixée dans la mise en données induit des tailles de maillage de l'ordre de 2 millions de cellules. Les calculs de CFD de chaque branche du collecteur sont effectués sur une machine de 60 cœurs et sont complétés en un peu plus de 4 heures.

Comme on peut le remarquer sur les visualisations de la figure 4.20, la variation de forme n'est pas très importante. Les paramètres ont en moyenne été modifiés de 12%. Toutefois, les valeurs de segments sont faibles pour cette mise en données donc leur taux de variation augmente rapidement. C'est pour cela que l'on relève une variation maximale de paramètre de près de 70%.

Pour observer la variation de forme, nous avons superposé les deux formes, la transparence de la forme optimale nous donne un aperçu des zones d'évolution de la forme. On remarque que la forme évolue principalement au niveau de la jonction. On observe que celle-ci se décale suite au grossissement de chacune des branches.

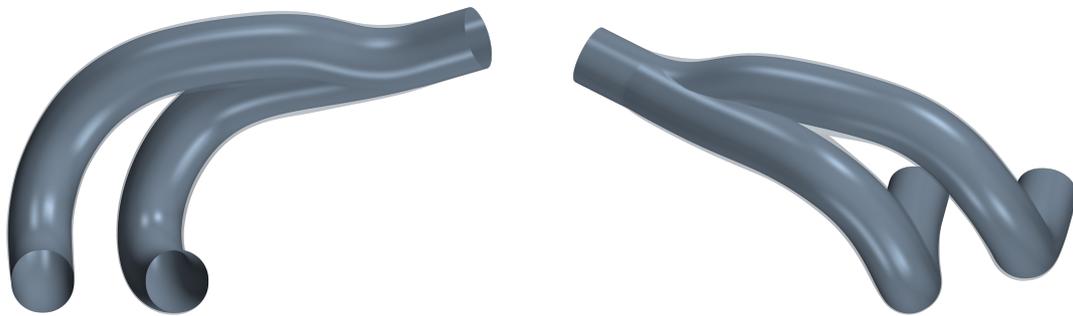


FIGURE 4.20 – Visualisation de l'évolution de la forme.

Les figures 4.21 et 4.22 représentent respectivement les champs de vitesse des états initial et final lorsque les branches  $R_1$  ou  $R_2$  sont actives. On remarque une uniformisation des champs aux états finaux. De plus, on observe la réduction des zones où la vitesse est plus élevée (zone en rouge). Les zones de forte vitesse traduisent des obstacles que l'écoulement rencontre. À l'état final, l'écoulement s'effectue mieux dans la canalisation, ce qui vient diminuer la valeur de perte de charge (cf. section 4.2).

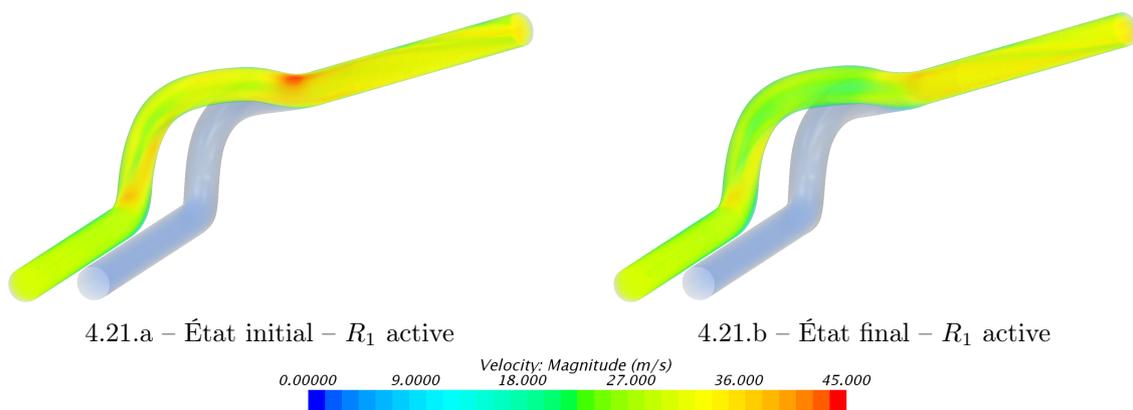


FIGURE 4.21 – Visualisation des champs de vitesse à l'état initial et à l'état final de la branche  $R_1$  active.

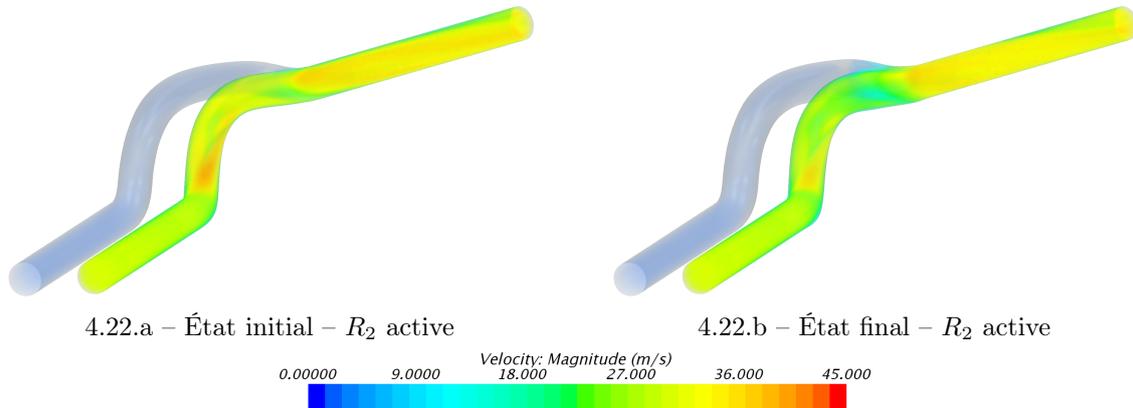


FIGURE 4.22 – Visualisation des champs de vitesse à l'état initial et à l'état final de la branche  $R_2$  active.

## 4.5 Discussion des résultats et synthèse

Les optimisations présentées dans ce chapitre montrent des gains importants sur chacune des fonctions objectif étudiées. On remarque dans les différents cas que la variation des paramètres est relativement faible par rapport aux gains obtenus.

La comparaison avec la méthode par plan d'expériences et surface de réponse montre que la méthode par gradient diminue le nombre de calculs nécessaires pour obtenir un résultat équivalent. De plus, cette comparaison a été effectuée sur un nombre de paramètres assez faible. Si le nombre de paramètres augmente, les écarts en terme de temps de calcul et de ressource vont exploser. En effet, la surface de réponse va nécessiter plus de calcul pour converger alors que la méthode par gradient est indépendante du nombre de paramètres.

L'application de la méthode de projection harmonique sur des formes complexes représente un avantage important pour la mise en place de nouveau cas test. Les applications ont aussi montré l'indépendance des optimisations vis-à-vis du nombre de paramètres. On peut donc envisager d'utiliser ces méthodes sur des modèles CAO ayant plusieurs centaines de paramètres sans que cela représente un frein pour l'étude.

La boucle d'optimisation n'étant pas le point central de la thèse, des développements et des améliorations sont susceptibles d'augmenter significativement son efficacité notamment en ce qui concerne les algorithmes d'optimisation.

Globalement, l'intégration du calcul du gradient de la fonction objectif par rapport aux paramètres CAO dans un processus d'optimisation présente de nombreux avantages. Parmi eux, le fait que le modèle CAO, élément incontournable en industrie, soit l'élément pivot de la boucle facilite les étapes de développement. Ainsi, il est aussi possible de contrôler au cours de l'optimisation les variations sur des paramètres sont définis par l'utilisateur. Mais surtout le modèle optimisé est directement utilisable dans la suite du processus industriel sans nécessiter d'étape de post-traitement.

## Bibliographie du 4<sup>e</sup> chapitre

- [Armijo 1966] Larry Armijo. *Minimization of functions having Lipschitz continuous first partial derivatives*. Pacific J. Math., vol. 16, no. 1, pages 1–3, 1966. (cité page 100)
- [Benoist 1994] D. Benoist, Y. Tourbier et S. Germain-Tourbier. Plans d’expériences : construction et analyse. Technique & Documentation-Lavoisier, 1994. (cité pages 8, 9 et 116)
- [CD- 2014] CD-adapco. *STAR-CCM+ Documentation*, 9.06 édition, October 2014. (cité pages 107 et 139)
- [Culioli 1994] J.C. Culioli. Introduction à l’optimisation. Ellipses, 1994. (cité pages 7 et 99)
- [Froment 2014] Pierre Froment. *Optimisation de formes paramétriques en grande dimension*. PhD thesis, École Centrale de Lyon, 2014. (cité pages 10, 12, 15, 27, 28, 29, 34, 45, 47 et 97)
- [Goldstein 1965] A. A. Goldstein. *On Steepest Descent*. Journal of the Society for Industrial and Applied Mathematics Series A Control, vol. 3, no. 1, pages 147–151, 1965. (cité page 100)
- [Hastie 2001] Trevor Hastie, Robert Tibshirani et Jerome Friedman. The elements of statistical learning. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001. (cité page 117)
- [Johnson 1990] M.E. Johnson, L.M. Moore et D. Ylvisaker. *Minimax and maximin distance designs*. Journal of Statistical Planning and Inference, vol. 26, no. 2, pages 131–148, 1990. (cité page 117)
- [Jones 2001] Eric Jones, Travis Oliphant, Pearu Peterson *et al.* *SciPy : Open source scientific tools for Python*, 2001. [Online ; accessed 2016-06-21]. (cité page 118)
- [Marler 2004] R.T. Marler et J.S. Arora. *Survey of multi-objective optimization methods for engineering*. Structural and Multidisciplinary Optimization, vol. 26, pages 369–395, 2004. (cité page 106)
- [Nocedal 2006] J. Nocedal et S. J. Wright. Numerical optimization. Springer, New York, 2nd édition, 2006. (cité page 118)
- [Pedregosa 2011] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot et E. Duchesnay. *Scikit-learn : Machine Learning in Python*. Journal of Machine Learning Research, vol. 12, pages 2825–2830, 2011. (cité page 117)
- [Rondepierre 2013] Aude Rondepierre et Pierre Weiss. Méthodes standards en optimisation non linéaire déterministe. 2013. (cité page 103)
- [Wolfe 1971] Philip Wolfe. *Convergence Conditions for Ascent Methods. II : Some Corrections*. SIAM Review, vol. 13, no. 2, pages 185–188, 1971. (cité page 100)

# Conclusion et perspectives

---

Les méthodes numériques présentent de réels atouts dans le domaine de l'optimisation de forme. Elles permettent le développement de nouvelles pièces en limitant le nombre de prototypes physiques. Parmi elles, les méthodes d'optimisation par gradient parviennent à un gain appréciable en un nombre réduit de calculs. Mais surtout chaque calcul améliore potentiellement la solution, ce qui présente un avantage indéniable dans un processus industriel où les temps de conception sont souvent limités. En parallèle de cela, l'optimisation paramétrique se base sur un modèle CAO tout au long de la boucle d'optimisation. Ainsi, le retour CAO – étape traditionnellement chronophage et susceptible de dégrader la performance de la solution optimale – est nativement inclus dans ces méthodes. L'espace admissible des formes est sûrement plus restreint que pour des méthodes topologique ou géométrique mais l'optimum obtenu est conservé dans la suite du processus. Dans ce manuscrit, nous avons concilié ces deux approches en proposant une méthode de calcul du gradient d'une fonction objectif par rapport aux paramètres CAO.

Suite aux développements présentés dans le chapitre 2, la méthode de sensibilité sur les maillages a de réels atouts et peut être intégrée aisément aux processus des industriels. En effet, son utilisation nécessite le strict minimum d'opérations pour l'utilisateur, ce qui accentue sa transparence. La gestion des surfaces multifrontières lève un verrou important pour la mise en place de nouvelles optimisations. En particulier, il n'est plus nécessaire de considérer un plan de découpe au début de la création du modèle CAO. La problématique de l'aspect-ratio est toujours présente, le recours à la quad-précision à la place de la double-précision est susceptible de repousser cette limite. De même, les méthodes itératives telles que la méthode de Ricci Flow [Gu 2008] devraient permettre une meilleure répartition des éléments dans le disque au détriment de la rapidité de calcul et donc s'écarter des niveaux de précision qui posent problème.

La méthode semi-analytique sur les formes CAO permet d'évaluer la sensibilité en tout point de la surface et évite toute perte de précision liée à l'interpolation. Toutefois, cette méthode gère difficilement l'apparition de patches ou le changement de leur identification. La mise en place d'un processus d'identification robuste permettrait le déploiement de cette méthode sur des géométries complexes. Une bonne idée pour permettre l'identification des patches serait de combiner les deux méthodes présentées dans ce manuscrit : appliquer la projection du *maillage de patches* pour les identifier puis exploiter l'approche semi-analytique pour évaluer la sensibilité. Une telle méthode permettrait de tirer partie des avantages de chacune des deux approches, notamment la flexibilité de la méthode basée sur les maillages et la précision de la définition CAO.

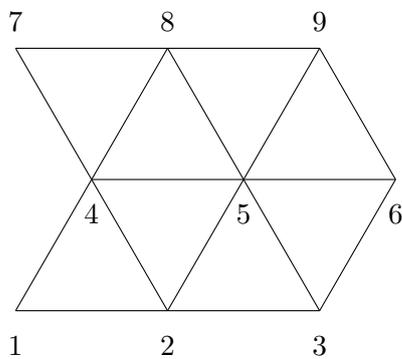
La comparaison avec la méthode par plan d'expériences et surface de réponse montre que les méthodes basées sur le gradient fournissent des solutions plus rapidement et surtout que le nombre de paramètres CAO n'est plus un facteur limitant. Le modèle CAO du collecteur du chapitre 4 confirme cette indépendance en atteignant la centaine de paramètres. Cet ordre de grandeur est caractéristique de modèles répandus chez les industriels. En outre, les surfaces obtenues sont d'excellente qualité et les contraintes de fabrication et de design, difficiles à exprimer

de façon mathématique sont plus faciles à mettre en place à l'aide des paramètres de dimensionnement. En outre, l'approche proposée permet de rester dans l'environnement du concepteur.

Le calcul de sensibilité a été exclusivement utilisé dans le cadre de méthodes exactes (basées sur le gradient). Néanmoins, l'exploitation des informations du gradient dans des méta-heuristiques comme le co-krigeage est susceptible d'améliorer la convergence des surfaces de réponse, et ainsi de diminuer le nombre de calculs requis pour leur construction. Il est aussi possible d'exploiter ces informations pour déterminer quels sont les paramètres qui modifient le plus la forme. Cette méthode ayant été éprouvée sur des cas de mécanique du solide et de mécanique des fluides, son application à des cas multiphysiques semble aisément envisageable. Finalement, l'application à des cas d'aérodynamique externe promet une réduction importante du temps de développement du fait des temps de calculs importants et de la taille des modèles impliqués, ceux-ci pouvant facilement dépasser les 200 paramètres CAO.

## A Détection des frontières

Dans le chapitre 2, on détecte les frontières dans un maillage surfacique. Les figures suivantes illustrent les étapes présentées dans la partie 2.1.1.



A.1.a – Exemple de maillage

1	4	2
2	5	3
2	4	5
6	3	5
4	8	5
8	4	7
5	9	6
9	5	8

A.1.b – Matrice de connectivité des éléments

FIGURE A.1 – Maillage et matrice de connectivité des éléments associée

	1	2	3	4	5	6	7	8	9
1	0	0	0	1	0	0	0	0	0
2	1	0	0	1	1	0	0	0	0
3	0	1	0	0	1	0	0	0	0
4	0	1	0	0	1	0	1	1	0
5	0	1	1	1	0	1	0	1	1
6	0	0	1	0	1	0	0	0	0
7	0	0	0	0	0	0	0	1	0
8	0	0	0	1	1	0	0	0	1
9	0	0	0	0	1	1	0	0	0

A.2.a – Ajout des termes dans la matrice  $A$

	1	2	3	4	5	6	7	8	9
1	0	1	0	1	0	0	0	0	0
2	1	0	1	2	2	0	0	0	0
3	0	1	0	0	2	1	0	0	0
4	1	2	0	0	2	0	1	2	0
5	0	2	2	2	0	2	0	2	2
6	0	0	1	0	2	0	0	0	1
7	0	0	0	1	0	0	0	1	0
8	0	0	0	2	2	0	1	0	1
9	0	0	0	0	2	1	0	1	0

A.2.b – Ajout de la transposée :  $A = A + A^t$

FIGURE A.2 – Construction de la matrice  $A$

1 - 2	<b>1 - 2</b>	<b>1 - 2</b>	<b>1 - 2</b>	<b>1 - 2</b>	<b>1 - 2</b>	<b>1 - 2</b>	<b>1 - 2</b>	<b>1 - 2</b>	<b>1 - 2</b>	<b>1 - 2</b>
1 - 4	1 - 4	<u>1 - 4</u>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>
2 - 1	<del>2 - 1</del>	1 - 4	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>	<b>2 - 3</b>
2 - 3	2 - 3	<b>2 - 3</b>	1 - 4	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>
3 - 2	3 - 2	<del>3 - 2</del>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>	<b>3 - 6</b>
3 - 6	3 - 6	3 - 6	4 - 1	1 - 4	<b>6 - 9</b>	<b>6 - 9</b>	<b>6 - 9</b>	<b>6 - 9</b>	<b>6 - 9</b>	<b>6 - 9</b>
4 - 1	4 - 1	4 - 1	4 - 1	4 - 1	<u>4 - 1</u>	<b>6 - 9</b>	<b>6 - 9</b>	<b>6 - 9</b>	<b>6 - 9</b>	<b>6 - 9</b>
4 - 7	4 - 7	4 - 7	4 - 7	4 - 7	4 - 7	<b>9 - 8</b>	<b>9 - 8</b>	<b>9 - 8</b>	<b>9 - 8</b>	<b>9 - 8</b>
4 - 7	4 - 7	4 - 7	<del>6 - 3</del>	<b>6 - 9</b>	4 - 1	<u>4 - 1</u>	<b>8 - 7</b>	<b>8 - 7</b>	<b>8 - 7</b>	<b>8 - 7</b>
6 - 3	6 - 3	6 - 3	6 - 9	7 - 4	4 - 7	1 - 4	<b>8 - 7</b>	<b>8 - 7</b>	<b>8 - 7</b>	<b>8 - 7</b>
6 - 9	6 - 9	6 - 9	7 - 4	7 - 4	7 - 4	4 - 1	<u>1 - 4</u>	<b>7 - 4</b>	<b>7 - 4</b>	<b>7 - 4</b>
7 - 4	7 - 4	7 - 4	7 - 8	7 - 8	7 - 8	4 - 7	4 - 1	<u>7 - 4</u>	<b>7 - 4</b>	<b>7 - 4</b>
7 - 8	7 - 8	7 - 8	8 - 7	8 - 7	8 - 7	7 - 4	<del>4 - 7</del>	<u>4 - 1</u>	<b>4 - 1</b>	<b>4 - 1</b>
8 - 7	8 - 7	8 - 7	8 - 9	8 - 9	8 - 7	7 - 8	<b>7 - 4</b>	<b>4 - 1</b>	<b>4 - 1</b>	<b>4 - 1</b>
8 - 9	8 - 9	8 - 9	8 - 9	8 - 9	<del>8 - 9</del>	<b>8 - 7</b>	<b>7 - 4</b>	<b>4 - 1</b>	<b>4 - 1</b>	<b>4 - 1</b>
9 - 6	9 - 6	9 - 6	9 - 6	9 - 6	<b>9 - 8</b>	<b>8 - 7</b>	<b>7 - 4</b>	<b>4 - 1</b>	<b>4 - 1</b>	<b>4 - 1</b>
9 - 8	9 - 8	9 - 8	9 - 8	9 - 8	9 - 8	<b>8 - 7</b>	<b>7 - 4</b>	<b>4 - 1</b>	<b>4 - 1</b>	<b>4 - 1</b>

FIGURE A.3 – Sélection d'un segment et reconstruction de la frontière en éliminant les segments symétriques

## B Fonction de paramétrisation $\varphi$

Dans le schéma général 2.3, on a introduit la fonction de paramétrisation  $\varphi$  qui met un sous-ensemble  $\mathcal{S}'$  en correspondance avec  $\mathcal{S}$ . Cette fonction peut servir par exemple à déterminer le tenseur métrique fondamental  $M_\varphi = J_\varphi^t J_\varphi$ . Les lignes ci-dessous détaillent son calcul bien que l'on n'ait pas exploité cet indicateur par la suite. Il sert notamment à vérifier comment la transformation  $\varphi$  altère les distances et les angles.

Ainsi, on appelle  $\varphi$  la fonction de paramétrisation qui met un sous-ensemble  $\mathcal{S}'$  en correspondance avec  $\mathcal{S}$  :

$$\begin{aligned} \varphi : \mathcal{S}' \subset \mathbb{R}^2 &\rightarrow \mathcal{S} \subset \mathbb{R}^3 \\ (u,v) &\mapsto \varphi(u,v) = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \end{aligned} \quad (\text{B.1})$$

Il vient :

$$U(x,y,z) = U(\varphi_1(u,v), \varphi_2(u,v), \varphi_3(u,v)) = \tilde{U}(u,v) \quad (\text{B.2})$$

La matrice jacobienne de  $\varphi$  vaut :

$$\begin{aligned} J_\varphi &= \nabla \varphi(u,v) \\ &= \begin{bmatrix} \frac{\partial \varphi_1(u,v)}{\partial u} & \frac{\partial \varphi_1(u,v)}{\partial v} \\ \frac{\partial \varphi_2(u,v)}{\partial u} & \frac{\partial \varphi_2(u,v)}{\partial v} \\ \frac{\partial \varphi_3(u,v)}{\partial u} & \frac{\partial \varphi_3(u,v)}{\partial v} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \\ \frac{\partial z}{\partial u} & \frac{\partial z}{\partial v} \end{bmatrix} \end{aligned} \quad (\text{B.3})$$

Il est possible de déterminer numériquement  $J_\varphi$  en calculant la matrice jacobienne de  $m^{-1}$

et en utilisant le théorème de dérivation des fonctions composées :

$$\begin{aligned}
 J_{m^{-1}} &= \nabla m^{-1}(u,v) \\
 &= \begin{bmatrix} \frac{\partial m_1^{-1}(u,v)}{\partial u} & \frac{\partial m_1^{-1}(u,v)}{\partial v} \\ \frac{\partial m_2^{-1}(u,v)}{\partial u} & \frac{\partial m_2^{-1}(u,v)}{\partial v} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\partial \xi}{\partial u} & \frac{\partial \xi}{\partial v} \\ \frac{\partial \eta}{\partial u} & \frac{\partial \eta}{\partial v} \end{bmatrix} \\
 &= \frac{1}{(u_2 - u_1)(v_3 - v_1) - (u_3 - u_1)(v_2 - v_1)} \begin{bmatrix} v_3 - v_1 & -(u_3 - u_1) \\ -(v_2 - v_1) & u_2 - u_1 \end{bmatrix}
 \end{aligned} \tag{B.4}$$

$$\begin{aligned}
 J_\varphi &= \nabla \varphi(u,v) = \nabla (\mathcal{M} \circ m^{-1})(u,v) \\
 &= \begin{bmatrix} \frac{\partial \mathcal{M}_1}{\partial \xi}(m^{-1}) \frac{\partial m_1^{-1}}{\partial u} + \frac{\partial \mathcal{M}_1}{\partial \eta}(m^{-1}) \frac{\partial m_2^{-1}}{\partial u} & \frac{\partial \mathcal{M}_1}{\partial \xi}(m^{-1}) \frac{\partial m_1^{-1}}{\partial v} + \frac{\partial \mathcal{M}_1}{\partial \eta}(m^{-1}) \frac{\partial m_2^{-1}}{\partial v} \\ \frac{\partial \mathcal{M}_2}{\partial \xi}(m^{-1}) \frac{\partial m_1^{-1}}{\partial u} + \frac{\partial \mathcal{M}_2}{\partial \eta}(m^{-1}) \frac{\partial m_2^{-1}}{\partial u} & \frac{\partial \mathcal{M}_2}{\partial \xi}(m^{-1}) \frac{\partial m_1^{-1}}{\partial v} + \frac{\partial \mathcal{M}_2}{\partial \eta}(m^{-1}) \frac{\partial m_2^{-1}}{\partial v} \\ \frac{\partial \mathcal{M}_3}{\partial \xi}(m^{-1}) \frac{\partial m_1^{-1}}{\partial u} + \frac{\partial \mathcal{M}_3}{\partial \eta}(m^{-1}) \frac{\partial m_2^{-1}}{\partial u} & \frac{\partial \mathcal{M}_3}{\partial \xi}(m^{-1}) \frac{\partial m_1^{-1}}{\partial v} + \frac{\partial \mathcal{M}_3}{\partial \eta}(m^{-1}) \frac{\partial m_2^{-1}}{\partial v} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\partial x}{\partial \xi} \frac{\partial \xi}{\partial u} + \frac{\partial x}{\partial \eta} \frac{\partial \eta}{\partial u} & \frac{\partial x}{\partial \xi} \frac{\partial \xi}{\partial v} + \frac{\partial x}{\partial \eta} \frac{\partial \eta}{\partial v} \\ \frac{\partial y}{\partial \xi} \frac{\partial \xi}{\partial u} + \frac{\partial y}{\partial \eta} \frac{\partial \eta}{\partial u} & \frac{\partial y}{\partial \xi} \frac{\partial \xi}{\partial v} + \frac{\partial y}{\partial \eta} \frac{\partial \eta}{\partial v} \\ \frac{\partial z}{\partial \xi} \frac{\partial \xi}{\partial u} + \frac{\partial z}{\partial \eta} \frac{\partial \eta}{\partial u} & \frac{\partial z}{\partial \xi} \frac{\partial \xi}{\partial v} + \frac{\partial z}{\partial \eta} \frac{\partial \eta}{\partial v} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \\ \frac{\partial z}{\partial u} & \frac{\partial z}{\partial v} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} \end{bmatrix} \begin{bmatrix} \frac{\partial \xi}{\partial u} & \frac{\partial \xi}{\partial v} \\ \frac{\partial \eta}{\partial u} & \frac{\partial \eta}{\partial v} \end{bmatrix} \\
 &= J_{\mathcal{M}} J_{m^{-1}} \\
 &= \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \\ z_2 - z_1 & z_3 - z_1 \end{bmatrix} \frac{1}{(u_2 - u_1)(v_3 - v_1) - (u_3 - u_1)(v_2 - v_1)} \begin{bmatrix} v_3 - v_1 & -(u_3 - u_1) \\ -(v_2 - v_1) & u_2 - u_1 \end{bmatrix}
 \end{aligned} \tag{B.5}$$

## C Spécificités liées à la CFD

### C.1 Critères de qualité du maillage

Les explications et images reportées dans cette annexe proviennent de la documentation [CD- 2014] de STAR-CCM+®.

Les principaux critères sont :

- o Angle d'asymétrie entre cellules :

La mesure d'asymétrie permet de déterminer si les cellules situées de part et d'autre d'une face sont disposées de façon à ce que les quantités diffusées ne deviennent pas illimitées (voir figure C.1).

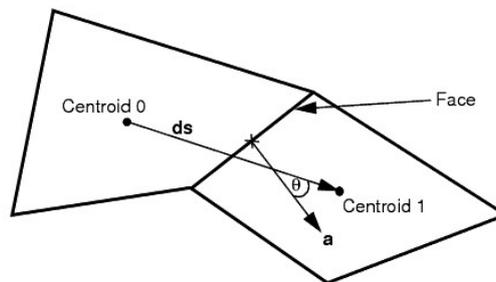


FIGURE C.1 – Illustration du calcul de l'angle d'asymétrie entre cellules

L'angle d'asymétrie  $\theta$  est caractérisé par le vecteur  $\mathbf{a}$ , normal à la face, et le vecteur  $\mathbf{ds}$ , connectant les centroïdes des deux cellules adjacentes. Un angle nul indique que le maillage est parfaitement orthogonal. La présence d'angles supérieurs ou égaux à  $90^\circ$  se traduit typiquement par des problèmes de convergence du solveur. Le problème vient du fait que la formulation du terme de diffusion contient le produit scalaire  $\mathbf{a} \cdot \mathbf{ds}$  au dénominateur pour les variables scalaires transportées. Avec ces valeurs, la précision du calcul et la robustesse sont réduites. Pour réduire cet impact, il est nécessaire d'éviter les angles supérieurs à  $85^\circ$ .

- o Angle d'asymétrie aux frontières :

La mesure d'asymétrie aux frontières peut être aussi importante qu'entre les cellules. Elle correspond à l'angle formé entre le vecteur  $\mathbf{a}$ , normal à la face et le vecteur connectant le centroïde de la cellule au centroïde de la face à la frontière (voir figure C.2). Cette valeur est importante en écoulement laminaire et en transferts thermiques dans les solides car le produit scalaire de ces deux vecteurs apparaît au dénominateur dans le calcul de diffusion aux frontières. De même que pour l'angle d'asymétrie entre cellules, pour garder une robustesse de calcul, il est nécessaire de limiter cet angle à une valeur maximale de  $85^\circ$ .

- o Validité des faces :

La validité des faces repose sur une mesure de la cohérence des normales aux faces par rapport au centroïde de la cellule à laquelle elles se rapportent. Le calcul est pondéré par l'aire des faces.

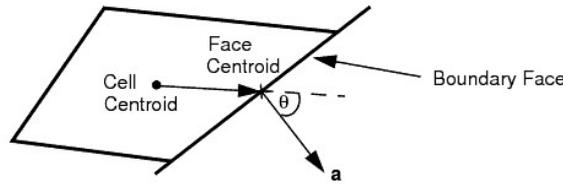
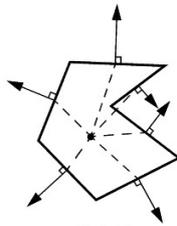
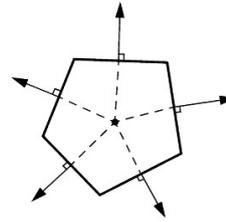


FIGURE C.2 – Illustration du calcul de l’angle d’asymétrie aux frontières



C.3.a – Non-respect du critère



C.3.b – Respect du critère

FIGURE C.3 – Illustration du critère de validité des faces

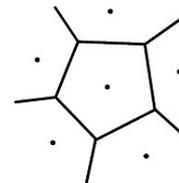
Une cellule avec une mauvaise validité des faces (figure C.3.a) contient une ou plusieurs normales aux faces qui pointent vers l’intérieur, vers le centroïde. Une cellule de bonne qualité (figure C.3.b) a toutes ses normales qui pointent vers l’extérieur, à l’opposé du centroïde de la cellule. Une valeur de validité de 1 indique que toutes les normales sont bien orientées. Les valeurs inférieures à 1 traduisent qu’une partie de ces normales pointent vers l’intérieur indiquant une certaine forme de concavité. Les valeurs inférieures à 0,5 signifient un volume de cellule négatif. Le seuil minimal de validité recommandé est de 0,85.

o Qualité des cellules :

L’algorithme de mesure de qualité de cellules est basé sur une méthode hybride utilisant la méthode de Gauss et celle des moindres carrés pour le calcul des gradients de cellules. Ce n’est pas seulement une fonction pour calculer la distribution géométrique relative du centroïde des cellules adjacentes mais aussi pour connaître l’orientation des faces. Généralement, des cellules plates avec des faces fortement non-orthogonales ont une qualité basse.



C.4.a – Non-respect du critère



C.4.b – Respect du critère

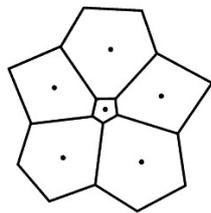
FIGURE C.4 – Illustration du critère de qualité des cellules

Une cellule avec une qualité de 1 est considérée comme parfaite. Une cellule cubique est un exemple de ce type de cellule mais d'autres formes de cellules polyédriques peuvent théoriquement atteindre ce niveau de perfection. Une cellule dégénérée a une valeur de qualité approchant zéro.

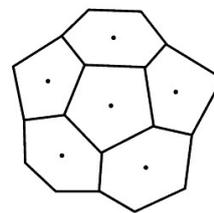
Suivant la physique sélectionnée pour l'analyse, une qualité de cellule basse peut fournir une solution valide. Cependant, cela affecte la robustesse et la précision du calcul. C'est pourquoi l'intervalle de qualité est fixé entre  $1e^{-6}$  et 1.

- o Changement de volume :

La mesure de changement de volume est un ratio entre le volume d'une cellule et sa voisine la plus grande.



C.5.a – Non-respect du critère



C.5.b – Respect du critère

FIGURE C.5 – Illustration du critère de changement de volume

Une valeur de 1 indique que le volume d'une cellule est équivalent à celle de ses voisines directes (figure C.5.b). Comme le volume d'une cellule décroît relativement par rapport à celui de ses voisines, le changement de volume peut être utilisé pour repérer cette cellule. Une grande différence de volume entre deux cellules peut provoquer des imprécisions et des instabilités du solveur. Les valeurs de changement de volume des cellules doivent être compris entre  $1e^{-3}$  et 1 pour limiter les impacts au niveau de la robustesse.

## C.2 Types de maillage

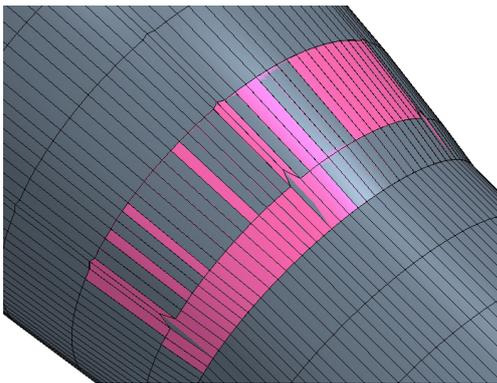
Les critères de qualité données en annexe C.1 permettent d'évaluer le maillage. En plus du respect de ces critères, le choix du type de maillage est primordial car ils sont plus ou moins adaptés en fonction de l'application. Notre intérêt s'est porté sur deux maillages proposés par le logiciel STAR-CCM+<sup>®</sup> et qui répondent aux spécifications du solveur adjoint :

- o *Maillage polyédrique* ;
- o *Generalized Cylinder*, basé sur un maillage polyédrique extrudé dans la longueur de la pièce qui se doit d'être de forme cylindrique.

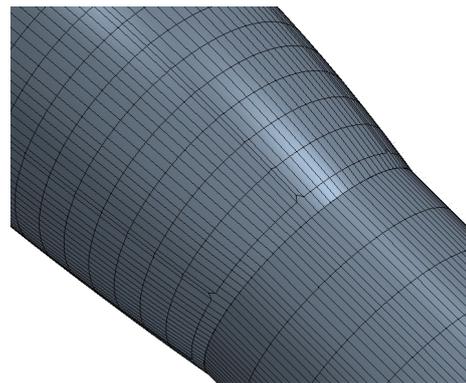
Les études que nous avons menées se limitent pour l'instant à des simulations en aérodynamique interne. Les formes sont donc le plus souvent assimilables à des tubulures avec des sections plus ou moins cylindriques. Le maillage de type *Generalized Cylinder* est le meilleur candidat vis-à-vis de la définition de ces formes car il génère un maillage par extrusion suivant la longueur de la partie considérée, réduisant ainsi le nombre de cellules et facilitant la convergence dans certains cas. Néanmoins, l'utilisation de ce maillage présente quelques limites. Tout d'abord sa mise

en place fait appel à de nombreux paramètres comme le nombre de couches, les différents ratio ou encore le profil (symétrique ou hyperbolique) qui sont à fixer à priori en fonction de la forme. Dans notre cas, la forme va évoluer au fur et à mesure des itérations de la boucle d'optimisation. Il est nécessaire de faire évoluer ces paramètres au fil des itérations de boucle pour conserver un maillage de qualité. Cette contrainte n'est pas une chose aisée : leur ajustement nécessite d'avoir accès à des caractéristiques géométriques de la pièce comme les rayons de courbure interne et externe du tube. Cela peut se faire par exemple dans l'environnement CAO mais a le gros défaut de venir alourdir la méthode.

Comme l'illustre la figure C.6.a, le Field Function Bad Cell Indicator, utilisé lors du Cell Quality Remediation, met en évidence des cellules qui ne respectent pas certains critères de qualité. Il faut donc adapter l'épaisseur des couches dans les extrusions afin de rester cohérent et éviter les ratio trop important (figure C.6.b). De plus, pour permettre une discrétisation plus fine aux extrémités et un transfert satisfaisant aux jonctions coude-extrusion, on utilise le profil *symmetric hyperbolic* avec 140 couches et un stretching à 4 pour la partie principale et 80 couches et un stretching à 4 pour les extrusions.



C.6.a – Maillage avec des cellules de mauvaise qualité



C.6.b – Maillage avec des cellules corrigées

FIGURE C.6 – Visualisation de l'indicateur de mauvaise cellule

Le maillage polyédrique, nécessitant plus de cellules pour obtenir un résultat équivalent à celui obtenu avec un maillage *Generalized Cylinder*, n'implique pas de modification des réglages au fur et à mesure de l'évolution de la forme, ce qui est très intéressant pour l'automatisation et la simplification de la boucle d'optimisation.

### C.3 Remarque sur le nombre d'itérations et la convergence du calcul CFD

Afin de minimiser le nombre d'itérations nécessaires, il serait intéressant d'utiliser des critères de convergence sur les résidus. Ces derniers décrivent l'erreur numérique dans les équations de conservation. L'équation (C.1) présente le calcul d'un résidu en moyenne quadratique (*root mean*

square) pour l'ensemble des cellules.

$$R_{RMS} = \sqrt{\frac{1}{n} \sum_{i=1}^n r^2} \quad (C.1)$$

avec  $r$  : l'erreur absolue dans la solution d'une équation de conservation d'une cellule  
 $n$  : le nombre de cellules

Usuellement, les valeurs RMS de chacun des résidus sont normalisés afin de pouvoir les comparer entre eux.

$$R_{Pres} = \frac{R_{RMS}}{R_{norm}} \quad (C.2)$$

avec  $R_{norm} = \max(|R_1|, |R_2|, \dots, |R_m|)$   
 et  $\left\{ \begin{array}{l} R_m : \text{les résidus de chaque itération} \\ m : \text{le nombre d'itérations sur lesquelles est cherché le maximum} \end{array} \right.$

En pratique, on prend le maximum sur 5 itérations ( $m = 5$ ). Si l'on utilise une rampe sur le débit par exemple, il est intéressant de modifier cette valeur. En effet, le terme  $R_{Pres}$  est trop dépendant du terme  $R_{norm}$  car, par exemple, si l'on réinitialise le calcul, les résidus repartent à 1.

En pratique, le processus itératif est arrêté lorsque l'estimateur du résidu atteint une valeur admissible. Cela ne garantit pas que la solution trouvée corresponde à la réalité. Il faudrait s'attarder à suivre la grandeur voulue comme la perte de charge et vérifier que la valeur reste stable d'une itération à l'autre. Toutefois, ce n'est pas un critère suffisant de convergence car cela signifie seulement que le solveur ne peut pas trouver de meilleure solution et non pas que la solution est correcte. Il faudrait, à l'avenir, définir un critère quantitatif sur les résidus (et non qualitatif comme actuellement).

## Bibliographie des annexes

[CD- 2014] CD-adapco. *STAR-CCM+ Documentation*, 9.06 édition, October 2014. (cité pages 107 et 139)

# Bibliographie globale

- [Adelman 1986] Howard M. Adelman et Raphael T. Haftka. *Sensitivity Analysis of Discrete Structural Systems*. AIAA Journal, vol. 24, no. 5, pages 823–832, May 1986. (cité page 18)
- [Albertelli 2015] Marc Albertelli, Thomas Abballe, Grégoire Allaire, Axelle Caron, Philippe Conraux, Luca Dall’Olio, Cécile Dobrzynski Charles Dapogny, Benoit Jeannin, Damien Lachouette François Jouve, Thierry Le Sommer, Kevin Maquin, Georgios Michailidis, Maximilien Siavelis et Vassili Srithammavanh. *RODIN project, Topology Optimization 2.0 ?* SIA, Congrès Simulation, 2015. (cité page 11)
- [Allaire 2004] Grégoire Allaire, François Jouve et Anca-Maria Toader. *Structural Optimization Using Sensitivity Analysis and a Level-set Method*. J. Comput. Phys., vol. 194, no. 1, pages 363–393, February 2004. (cité page 11)
- [Allaire 2007] Grégoire Allaire. *Conception optimale de structures*, volume 58. Springer Berlin Heidelberg New York, 2007. (cité pages 11 et 14)
- [Anderson 1999] W.K. Anderson et V. Venkatakrishnan. *Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation*. Computers and Fluids, vol. 28, page 443–480, 1999. (cité page 19)
- [Armijo 1966] Larry Armijo. *Minimization of functions having Lipschitz continuous first partial derivatives*. Pacific J. Math., vol. 16, no. 1, pages 1–3, 1966. (cité page 100)
- [Armstrong 2007] C.G. Armstrong, T.T. Robinson, H Ou et C. Othmer. *Linking adjoint sensitivity maps with CAD parameters*. Evolutionary and Deterministic Methods for Design, Optimization and Control, vol. Eurogen 2007, pages 234–239, 2007. (cité page 23)
- [Bendsøe 2003] Martin P. Bendsøe et Ole Sigmund. *Topology optimization : Theory, methods and applications*. Berlin Heidelberg, 2003. (cité page 11)
- [Benoist 1994] D. Benoist, Y. Tourbier et S. Germain-Tourbier. *Plans d’expériences : construction et analyse*. Technique & Documentation-Lavoisier, 1994. (cité pages 8, 9 et 116)
- [Binois 2015] Mickaël Binois. *Uncertainty quantification on pareto fronts and high-dimensional strategies in bayesian optimization, with applications in multi-objective automotive design*. Theses, Ecole Nationale Supérieure des Mines de Saint-Etienne, December 2015. (cité page 8)
- [Boole 1860] G. Boole. *A treatise on the calculus of finite differences*. Macmillan, 1860. (cité page 18)
- [Borrvall 2003] Thomas Borrvall et Joakim Petersson. *Topology optimization of fluids in Stokes flow*. International Journal for Numerical Methods in Fluids, vol. 41, no. 1, pages 77–107, 2003. (cité page 12)
- [Cacuci 1981] Dan G. Cacuci. *Sensitivity theory for nonlinear systems. I. Nonlinear functional analysis approach*. Journal of Mathematical Physics, vol. 22, no. 12, pages 2794–2802, 1981. (cité page 19)

- [CD- 2014] CD-adapco. *STAR-CCM+ Documentation*, 9.06 édition, October 2014. (cité pages 107 et 139)
- [Céa 1986] Jean Céa. *Conception optimale ou identification de formes, calcul rapide de la dérivée directionnelle de la fonction coût*. Modélisation Mathématique et Analyse Numérique, vol. 20, no. 3, pages 371–402, 1986. (cité page 19)
- [Chen 1997] S. Chen et D.A. Tortorelli. *Three-dimensional shape optimization with variational geometry*. Structural optimization, vol. 13, no. 2-3, pages 81–94, 1997. (cité page 24)
- [Choi 1994] Kyung K. Choi et Kuang-Hua Chang. *A study of design velocity field computation for shape optimal design*. Finite Elements in Analysis and Design, vol. 15, no. 4, pages 317 – 341, 1994. (cité page 25)
- [Choi 2005] Kyung K. Choi et Nam-Ho Kim. *Structural sensitivity analysis and optimization 1*. Springer, 2005. (cité page 25)
- [Culioli 1994] J.C. Culioli. *Introduction à l’optimisation*. Ellipses, 1994. (cité pages 7 et 99)
- [Dannenhoffer III 2015] J.F. Dannenhoffer III et R. Haimes. *Design Sensitivity Calculations Directly on CAD-based Geometry*. In 53rd AIAA Aerospace Sciences Meeting, page 1370, 2015. (cité pages 23, 65 et 66)
- [Dapogny 2013] C. Dapogny. *Shape optimization, level set methods on unstructured meshes and mesh evolution*. PhD thesis, UPMC, 2013. (cité pages 11 et 13)
- [Darcy 1857] Henry Darcy. *Recherches expérimentales relatives au mouvement de l’eau dans les tuyaux*, volume 1. Mallet-Bachelier, 1857. (cité page 12)
- [Delaunay 1934] Boris Delaunay. *Sur la sphere vide*. Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk, vol. 7, no. 793-800, pages 1–2, 1934. (cité page 56)
- [du Cauzé de Nazelle 2013] P. du Cauzé de Nazelle. *Paramétrage de formes surfaciques pour l’optimisation*. PhD thesis, École Centrale de Lyon, 2013. (cité page 14)
- [Floater 2002] Michael S. Floater et Kai Hormann. *Surface Parameterization : a Tutorial and Survey*. Article, 2002. (cité page 34)
- [Foretnik 2010] Jan Foretnik. Nurbs demo. [geometrie.foretnik.net/files/NURBS-en.swf](http://geometrie.foretnik.net/files/NURBS-en.swf), 2010. (cité pages 70, 71, 72 et 74)
- [Froment 2014] Pierre Froment. *Optimisation de formes paramétriques en grande dimension*. PhD thesis, École Centrale de Lyon, 2014. (cité pages 10, 12, 15, 27, 28, 29, 34, 45, 47 et 97)
- [Geuzaine 2009] C. Geuzaine et J.F. Remacle. *Gmsh : a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities*. International Journal for Numerical Methods in Engineering, vol. 79, no. 11, pages 1309–1331, 2009. (cité page 27)
- [Goldstein 1965] A. A. Goldstein. *On Steepest Descent*. Journal of the Society for Industrial and Applied Mathematics Series A Control, vol. 3, no. 1, pages 147–151, 1965. (cité page 100)
- [Goupy 2013] Jacques Goupy. *Introduction aux plans d’expériences*. Dunod/L’usine nouvelle, 5th édition, 2013. (cité page 9)
- [Gu 2008] X.D. Gu et S.T. Yau. *Computational conformal geometry*, volume 3 of *Advanced lectures in mathematics*. International Press, 2008. (cité pages 27, 34, 54 et 133)
- [Hamilton 1982] Richard S. Hamilton. *Three-manifolds with positive Ricci curvature*. J. Differential Geom., vol. 17, no. 2, pages 255–306, 1982. (cité page 54)

- [Hastie 2001] Trevor Hastie, Robert Tibshirani et Jerome Friedman. The elements of statistical learning. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001. (cité page 117)
- [Hughes 2005] T.J.R. Hughes, J.A. Cottrell et Y. Bazilevs. *Isogeometric analysis : CAD, finite elements, NURBS, exact geometry and mesh refinement*. Computer Methods in Applied Mechanics and Engineering, vol. 194, no. 39-41, pages 4135–4195, 2005. (cité page 13)
- [Johnson 1990] M.E. Johnson, L.M. Moore et D. Ylvisaker. *Minimax and maximin distance designs*. Journal of Statistical Planning and Inference, vol. 26, no. 2, pages 131–148, 1990. (cité page 117)
- [Jones 2001] Eric Jones, Travis Oliphant, Pearu Peterson *et al.* *SciPy : Open source scientific tools for Python*, 2001. [Online ; accessed 2016-06-21]. (cité page 118)
- [Julisson 2016] Sarah Julisson. *Optimisation de formes de coques minces pour des géométries complexes*. PhD thesis, UVSQ, Décembre 2016. (cité page 14)
- [Keller 1976] J. B. Keller. *Inverse Problems*. The American Mathematical Monthly, vol. 83, no. 2, pages 107–118, 1976. (cité page 4)
- [Kelley 1999] Carl T Kelley. Iterative methods for optimization, volume 18. Siam, 1999. (cité page 7)
- [Kern 2016] Michel Kern. Méthodes numériques pour les problèmes inverses. Collection Mathématiques et statistiques. ISTE Éditions, Mars 2016. (cité pages 5 et 17)
- [Kripac 1997] Jiri Kripac. *A mechanism for persistently naming topological entities in history-based parametric solid models*. Computer-Aided Design, vol. 29, no. 2, pages 113 – 122, 1997. (cité pages 24, 34 et 76)
- [Lévy 2000] Bruno Lévy et J-L Mallet. *Paramétrisation des surfaces triangulées*. Revue internationale de CFAO et d’informatique graphique, vol. 15, no. 1, pages 25–42, 2000. (cité page 34)
- [Lindby 1999] T. Lindby et J.L.T. Santos. *Shape optimization of three-dimensional shell structures with the shape parametrization of a CAD system*. Structural optimization, vol. 18, no. 2-3, pages 126–133, 1999. (cité page 25)
- [Marler 2004] R.T. Marler et J.S. Arora. *Survey of multi-objective optimization methods for engineering*. Structural and Multidisciplinary Optimization, vol. 26, pages 369–395, 2004. (cité page 106)
- [Michailidis 2014] Georgios Michailidis. *Manufacturing Constraints and Multi-Phase Shape and Topology Optimization via a Level-Set Method*. Theses, Ecole Polytechnique X, January 2014. (cité pages 11 et 12)
- [Moos 2004] O. Moos, F. Klimetzek et R. Rossmann. *Bionic optimization of air-guiding systems*. SAE Technical Paper, 2004. (cité page 12)
- [Nocedal 2006] J. Nocedal et S. J. Wright. Numerical optimization. Springer, New York, 2nd édition, 2006. (cité page 118)
- [Osher 1988] Stanley Osher et James A. Sethian. *Fronts Propagating with Curvature Dependent Speed : Algorithms Based on Hamilton-Jacobi Formulations*. Journal of Computational Physics, vol. 79, no. 1, pages 12–49, 1988. (cité page 11)

- [Othmer 2005] C. Othmer et Th. Grahns. *Approaches to fluid dynamic optimization in the car development process*. EUROGEN, 2005. (cité page 17)
- [Pedregosa 2011] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot et E. Duchesnay. *Scikit-learn : Machine Learning in Python*. Journal of Machine Learning Research, vol. 12, pages 2825–2830, 2011. (cité page 117)
- [Peter 2006] J. Peter, S. Burguburu et M. Marcelet. Introduction à l’optimisation de forme en aérodynamique et quelques exemples d’application. ONERA, 2006. (cité page 19)
- [Piegl 1995] L.A. Piegl et W. Tiller. *The nurbs book*. Monographs in visual communication. Springer, 1995. (cité page 70)
- [Remacle 2010] J.-F. Remacle, C. Geuzaine, G. Compère et E. Marchandise. *High-quality surface remeshing using harmonic maps*. International journal for numerical methods in engineering, vol. 83, pages 403 – 425, 2010. (cité pages 27, 28 et 34)
- [Robinson 2012] Trevor T. Robinson, Cecil G. Armstrong, Hung Soon Chua, Carsten Othmer et Thorsten Grahns. *Optimizing Parameterized CAD Geometries Using Sensitivities Based on Adjoint Functions*. Computer-Aided Design and Applications, vol. 9, no. 3, pages 253–268, 2012. (cité page 25)
- [Rondepierre 2013] Aude Rondepierre et Pierre Weiss. Méthodes standards en optimisation non linéaire déterministe. 2013. (cité page 103)
- [Saitou 2005] Kazuhiro Saitou, Kazuhiro Izui, Shinji Nishiwaki et Panos Papalambros. *A Survey of Structural Optimization in Mechanical Product Development*. Journal of Computing and Information Science in Engineering, vol. 5, no. 3, pages 214–226, 2005. (cité page 11)
- [Shepard 1968] Donald Shepard. *A Two-dimensional Interpolation Function for Irregularly-spaced Data*. In Proceedings of the 1968 23rd ACM National Conference, ACM ’68, pages 517–524, New York, NY, USA, 1968. ACM. (cité page 61)
- [Sokolowski 1992] Jan Sokolowski et Jean-Paul Zolesio. Introduction to shape optimization : shape sensitivity analysis. Springer series in computational mathematics. Springer, New York, Heidelberg, Paris, 1992. (cité page 66)
- [Staten 2011] M.L. Staten, S.J. Owen, S.M. Shontz, A.G. Salinger et T.S. Coffey. *A comparison of mesh morphing methods for 3D shape optimization*. Proceedings of the 20<sup>th</sup> international meshing round-table, pages 293–311, 2011. (cité page 20)
- [Talbi 2009] El-Ghazali Talbi. Metaheuristics : from design to implementation, volume 74. John Wiley & Sons, 2009. (cité page 8)
- [Toivanen 2006] Jukka I. Toivanen et Janne Martikainen. *A new method for creating sparse design velocity fields*. Computer Methods in Applied Mechanics and Engineering, vol. 196, pages 528 – 537, 2006. (cité page 25)
- [Vasilopoulos 2016] I Vasilopoulos, D Agarwal, M Meyer, TT Robinson et CG Armstrong. *Linking Parametric CAD with Adjoint Surface Sensitivities*. 2016. (cité page 26)
- [Wolfe 1971] Philip Wolfe. *Convergence Conditions for Ascent Methods. II : Some Corrections*. SIAM Review, vol. 13, no. 2, pages 185–188, 1971. (cité page 100)
- [Xu 2014] Shenren Xu, Wolfram Jahn et Jens-Dominik Müller. *CAD-based shape optimisation with CFD using a discrete adjoint*. International Journal for Numerical Methods in Fluids, vol. 74, no. 3, pages 153–168, 2014. (cité page 23)

- [Yang 1992] R.J. Yang, A. Lee et D.T. McGeen. *Application of basis function concept to practical shape optimization problems*. Structural optimization, vol. 5, pages 55–63, 1992. (cité page 15)



# Liste des acronymes et abréviations

---

CAO	conception assistée par ordinateur.
CATIA <sup>®</sup>	conception assistée tridimensionnelle interactive appliquée.
CCA	continuity convergence accelerator.
CFD	computational fluid dynamics.
CFL	Courant–Friedrichs–Lewy.
ESI	engineering systems international.
FEM	finite element method.
FUI	fonds unique interministériel.
HPC	high performance computing.
IGES	initial graphics exchange specification.
LHS	latin hypercube sampling.
LOOCV	leave-one-out cross-validation.
MARS	multivariate adaptative regression spline.
NURBS	non-uniform rational basis splines.
NVH	noise, vibration and harshness.
ONERA	office national d'études et de recherches aérospatiales.
PTC	parametric technology corporation.
RANS	Reynolds-averaged Navier-Stokes.
RMS	root mean square.
RODIN <sup>®</sup>	robust structural optimization for design in industry.
Scilab <sup>®</sup>	scientific laboratory.
SIMP	solid isotropic material with penalization.
SLSQP	Sequential Least Squares Programming.

## Liste des acronymes et abréviations

---

<i>s.t.</i>	<i>subject to.</i>
STAR-CCM+ <sup>®</sup>	star-computational continuum mechanics+.
STEP	standard for the exchange of product model data.
SVD	singular value decomposition.
UPMC	Université Pierre et Marie Curie (Paris 6).



**Titre :** Calcul de gradient sur des paramètres CAO pour l'optimisation de forme

**Mots clés :** Paramètres CAO, optimisation de forme, gradient, solveur adjoint

**Résumé :** Dans ce manuscrit, nous présentons une méthode d'optimisation de forme qui se base sur des paramètres géométriques comme des longueurs, des angles, etc. Nous nous appuyons sur des techniques d'optimisation basées sur un gradient. La sensibilité de la fonction objectif par rapport à la position des nœuds du maillage nous est fournie par un solveur adjoint que l'on considère comme une boîte noire. Afin d'optimiser par rapport aux paramètres CAO, nous nous concentrons sur l'évaluation de la sensibilité de la position des nœuds par rapport à ces paramètres. Ainsi, nous proposons deux approches par différences finies.

La première méthode s'appuie sur une projection harmonique afin de comparer dans un même espace le maillage initial et celui obtenu suite à la variation d'un paramètre CAO. Les développements présentés dans ce manuscrit permettent d'étendre l'application aux formes ayant plusieurs frontières comme les collecteurs d'échappement. Nous avons développé une méthode d'interpolation adaptée à cette comparaison. L'ensemble du processus a été automatisé et nous en montrons l'entière efficacité sur des applications industrielles en aérodynamique interne.

La deuxième méthode se base directement sur les géométries CAO pour évaluer cette sensibilité. Nous utilisons la définition intrinsèque des patches dans l'espace paramétrique  $(u,v)$  pour effectuer cette comparaison. Grâce à l'utilisation des coordonnées exactes en tout point de la surface fournies par la CAO, nous évitons d'avoir recours à une interpolation afin d'avoir la meilleure précision de calcul possible. Cependant, contrairement à la première méthode, elle requiert d'identifier les correspondances entre les patches d'une forme à l'autre. Une application sur un cas académique a été faite en aérodynamique externe.

La pertinence de la première méthode a été démontrée sur des cas représentatifs et multiobjectifs, ce qui permettrait de faciliter son déploiement et son utilisation dans un cadre industriel. Quant à la deuxième méthode, nous avons montré son fort potentiel. Cependant, des développements supplémentaires seraient nécessaires pour une application plus poussée. Du fait qu'elles sont indépendantes des solveurs mécaniques et du nombre de paramètres, ces méthodes réduisent considérablement les temps de développement des produits, notamment en permettant l'optimisation multiphysique en grande dimension.

**Title :** Gradient-based methods for shape optimization on CAD parameters

**Keywords :** CAD parameters, shape optimization, gradient, adjoint solver

**Abstract :** In this manuscript, we present a shape optimization method based on CAD parameters such as lengths, angles, etc. We rely on gradient-based optimization techniques. The sensitivity of the objective function, with respect to the mesh nodes position, is provided by an adjoint solver considered here as a black box. To optimize with respect to CAD parameters, we focus on computing the sensitivity of the nodes positions with respect to these parameters. Thus, we propose two approaches based on finite differences.

The first method uses a harmonic projection to compare in the same space the initial mesh and the one obtained after a change of the set of CAD parameters. The developments presented in this manuscript open up new doors like the application to shapes with multiple borders such as exhaust manifolds. We also developed an interpolation method suitable for this comparison. The entire process is automated, and we demonstrate the entire effectiveness on internal aerodynamics industrial applications.

The second method is directly based on the CAD geometries to assess this sensitivity. To perform this comparison, we use the intrinsic definition of the patches in the parametric space  $(u,v)$ . Through the use of the exact coordinates at any point on the surface provided by the CAD, we avoid using an interpolation to get the best calculation accuracy possible. However, unlike the first method, it requires to identify the correspondence between patches from one shape to another. An application on an external aerodynamics academic case was made.

The relevance of the first method is demonstrated on a representative multi-objective case, which facilitate its deployment use in an industrial environment. Regarding the second method, we showed its great potential. However, further developments are needed to handle more advanced cases. Because they are independent of the mechanical solver and the number of parameters, these methods significantly reduce product development time, particularly by allowing large and multiphysics optimization.

