Couplage d'un schéma aux résidus distribués à l'analyse isogéométrique : méthode numérique et outils de génération et adaptation de maillage

Algiane Froehly

▶ **To cite this version:**

Algiane Froehly. Couplage d'un schéma aux résidus distribués à l'analyse isogéométrique : méthode numérique et outils de génération et adaptation de maillage. Mécanique des fluides [physics.class-ph]. Université Sciences et Technologies - Bordeaux I, 2012. Français. NNT : 2012BOR14563 . tel-00765918

HAL Id: tel-00765918

https://theses.hal.science/tel-00765918

Submitted on 17 Dec 2012

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.

THÈSE

Pour l'obtention du grade de

DOCTEUR DE L'UNIVERSITÉ DE

BORDEAUX I

ÉCOLE DOCTORALE DE MATHÉMATIQUES ET D'INFORMATIQUE

DOMAINE DE RECHERCHE : MATHÉMATIQUES APPLIQUÉES

Présentée par : **Algiane FRŒHLY**

Couplage d'un schéma aux résidus distribués à l'analyse isogéométrique : méthode numérique et outils de génération et adaptation de maillage

Directeur de thèse : **Rémi ABGRALL**

Co-direction : **Cécile DOBRZYNSKI**

Soutenue le 07/09/2012
Devant la Commission d'Examen

JURY

M. Luc MIEUSSENS	Professeur, U. Bordeaux 1	Président
M. Pascal FREY	Professeur, U. Paris 6	Rapporteur A
M. Jean-François REMACLE	Professeur, U. catholique de Louvain	Rapporteur B
M. Rémi ABGRALL	Professeur, U. Bordeaux 1	Directeur de thèse
Mme. Cécile DOBRZYNSKI	Maître de conférence, U. Bordeaux 1	Directrice de thèse
M. Vincent COUAILLIER	Ingénieur, Onera	Examinateur
M. Adrien LOSEILLE	Chargé de recherche, Inria	Examinateur
M. Mario RICCHIUTO	Chargé de recherche, Inria	Examinateur

Remerciements

Je tiens à remercier en tout premier lieu mes directeurs de thèse, M. Rémi ABGRALL et Mme. Cécile DOBRZYNSKI pour avoir accepté d'encadrer ce travail. Vous m'avez apporté l'un et l'autre une aide complémentaire. Merci Rémi pour ta confiance en m'ayant proposé cette thèse, merci également pour tes conseils avisés. Cécile, merci pour ta disponibilité et ton investissement qui m'ont bien souvent remotivée dans les moments de doutes.

Je remercie bien entendu les membres jury d'avoir bien voulu évaluer ce travail ainsi que M. Pascal FREY et M. Jean-François REMACLE pour avoir accepté d'en être les rapporteurs.

Mes pensées vont également à toutes les personnes ayant contribué de près ou de loin à cette thèse, qu'elles s'en soient rendues compte ou non. Je pense en particulier aux amis des premières années à l'Inria (Christelle, Nico... et tous ceux que j'oublie mais qui se reconnaîtront) : votre aide, votre soutien, et surtout les bons moments passés ensemble tiennent une grande place dans cette thèse (que serais-je devenue sans tes tests de régressions Pascal !).

Un grand merci aussi à ma famille et aux amis de toujours et de tous les jours (Amandine, Lise, Maréva...) auprès desquels il est tellement agréable de bougonner quand on à l'impression que rien ne va. C'est agréable de savoir qu'on a des gens sur qui compter.

Bien sûr, je remercie également tous les membres du labo et de l'équipe Bacchus avec lesquels j'ai pu collaborer.

Je terminerais avec un dernier remerciement mais le plus cher à mes yeux : Xavier, mon p'ti chou, merci pour tout. Pour ton aide lorsque je faisais n'importe quoi avec git, quand je m'arrachais les cheveux sur l'installation de PaStiX, Scotch et j'en passe, que je me rongeais les ongles sur le MPI... (oui, sans toi, je serais aujourd'hui chauve, sans ongles et peut-être même sans doigts).

Merci également d'avoir accepté de relire ma thèse et d'être resté au travail de longues heures lorsque la deadline du manuscrit s'est approchée. Je voudrais te remercier pour tellement d'autres choses (en vrac ta gentillesse, tous les petits plats que tu m'a cuisiné quand je travaillais... et quand je ne travaillais pas, *etc.*) mais comme tu le sais, je ne trouve pas les mots...

Résumé

Lors de simulations numériques d'ordre élevé, la discrétisation sous-paramétrique du domaine de calcul peut générer des erreurs dominant l'erreur liée à la discrétisation des variables. De nombreux travaux proposent d'utiliser l'analyse isogéométrique afin de mieux représenter les géométries et de résoudre ce problème.

Nous présenterons dans ce travail le couplage du schéma aux résidus distribués limité et stabilisé de Lax-Frieirichs avec l'analyse isogéométrique. En particulier, nous construirons une famille de fonctions de base permettant de représenter exactement les coniques et définies tant sur les éléments triangulaires que quadrangulaires : les fonctions de base de Bernstein rationnelles.

Nous nous intéresserons ensuite à la génération de maillages précis pour l'analyse isogéométrique. Notre méthode consiste à créer un maillage courbe à partir d'un maillage linéaire par morceaux de la géométrie. Le maillage obtenu en sortie de notre procédure est non-structuré, conforme et assure la continuité de nos fonctions de base sur tout le domaine.

Pour finir, nous décrirons les différentes méthodes d'adaptation de maillages développées : l'élévation d'ordre et le raffinement isotrope. Bien évidemment, la géométrie exacte du maillage courbe d'entrée est préservée au cours des processus d'adaptation.

Mots-clés :
Équations d'Euler, Schéma aux Résidus Distribués, Ordre Très Élevé, Analyse Isogéométrique, Fonctions de Base $\mathcal{NURBS}$, Fonctions de Base de Bernstein Rationnelles, Génération de Maillages Courbes, Adaptation de Maillages courbes, h-raffinement, p-raffinement

Discipline:
Mathématiques appliquées

Abstract

During high order simulations, the approximation error may be dominated by the errors linked to the sub-parametric discretization used for the geometry representation. Many works propose to use an isogeometric analysis approach to better represent the geometry and hence solve this problem.

In this work, we will present the coupling between the limited stabilized Lax-Friedrichs residual distributed scheme and the isogeometric analysis. Especially, we will build a family of basis functions defined on both triangular and quadrangular elements and allowing the exact representation of conics : the rational Bernstein basis functions.

We will then focus in how to generate accurate meshes for isogeometric analysis. Our idea is to create a curved mesh from a classical piecewise-linear mesh of the geometry. We obtain a conforming unstructured mesh which ensures the continuity of the basis functions over the entire mesh.

Last, we will detail the curved mesh adaptation methods developed : the order elevation and the isotropic mesh refinement. Of course, the adaptation processes preserve the exact geometry of the initial curved mesh.

Keywords:
Euler Equations, Residual Distribution Scheme, Very High Order, Isogeometric Analysis, $\mathcal{NURBS}$, Rational Bernstein Basis Functions, Curved Mesh Generation, Curved Mesh Adaptation, h-refinement, p-refinement

Discipline:
Applied mathematics

Table des matières

Table des figures **xiii**

Nomenclature **xvi**

Introduction **xvii**

1 Problématique xvii

2 État de l'art au début de cette thèse xix

3 Contributions de cette thèse et organisation du mémoire xxii

3.1 Contributions de cette thèse xxii

3.2 Organisation du mémoire xxiii

I Cadre théorique **1**

1 Dynamique des fluides 1

1.1 Équations de conservation 1

1.2 Lois d'état 2

1.3 Lois de comportement 3

1.4 Équations de la mécanique des fluides 4

1.4.1 Équations de Navier-Stokes 4

1.4.2 Équations d'Euler 5

2 Schéma numérique pour la résolution des équations d'Euler 9

2.1 Discrétisation numérique 10

2.2 Schéma aux résidus distribués 12

2.3 Propriétés du schéma 14

2.3.1 Consistance . . . 15
2.3.2 Précision . . . 16
2.3.3 Condition de conservation de la monotonicité . . . 17
2.3.4 Conservation de la linéarité . . . 20
2.4 Écriture d'un schéma aux résidus distribués d'ordre élevé . . . 21
2.4.1 Flux de Lax-Friedrichs . . . 21
2.4.2 Limitation . . . 23
2.4.3 Stabilisation . . . 25
2.5 Discrétisation temporelle . . . 27
3 Schéma numérique pour la résolution des équations de Navier-Stokes . . . 29
3.1 Formulation du problème . . . 29
3.2 Écriture du schéma $\mathcal{RD}$ et problématique . . . 30
3.3 Solution envisagée et calcul du résidu . . . 31

II Analyse isogéométrique **33**
1 Notations et terminologie . . . 33
1.1 Degrés de liberté et points de contrôle . . . 33
1.2 Espaces de travail . . . 35
1.2.1 De l'approximation globale à l'approximation locale sur un élément courbe T . . . 35
1.2.2 D'un élément courbe T de $\mathbb{R}^d$ à sa forme élémentaire dans l'espace paramétrique . . . 36
2 Fonctions de base . . . 39
2.1 Fonctions de base $\mathcal{NURBS}$ . . . 39
2.1.1 Fonctions de base B-Splines . . . 39
2.1.2 Fonctions de base $\mathcal{NURBS}$ . . . 45
2.1.3 Propriétés des B-Splines et $\mathcal{NURBS}$ . . . 46
2.2 Fonctions de base de Bernstein rationnelles . . . 52
2.2.1 Fonctions de base de Bernstein . . . 53
2.2.2 Fonctions de base de Bernstein rationnelles monodimensionnelles . . . 54
2.2.3 Fonctions de base de Bernstein rationnelles multidimensionnelles . . . 59
2.3 Propriétés des polynômes de Bernstein rationnels et non rationnels 63
3 Méthode numérique : Schéma $\mathcal{RD}$ isogéométrique . . . 64

3.1 Calcul du résidu de Lax-Friedrichs limité et stabilisé 65
3.1.1 Résidu global 65
3.1.2 Terme de filtrage 67
3.2 Matrice implicite 70
3.3 Conditions aux limites 72
3.3.1 Condition d'entrée de fluide : $\mathcal{BC}$ de type Dirichlet imposée fortement 72
3.3.2 Condition de paroi glissante imposée faiblement 73

III Génération de maillages courbes d'ordre 3 **77**
1 Pré-requis 78
2 Interpolation des nœuds frontières par une courbe $\mathcal{NURBS}$ 79
2.1 Approximation de la géométrie 79
2.2 Représentation d'un arc de cercle par une courbe $\mathcal{NURBS}$ 81
2.3 Algorithme de génération d'une géométrie courbe 82
3 Vérification et mise en conformité du maillage 85
3.1 Détection des éléments présentant des croisements 85
3.2 Suppression des croisements détectés 89
3.3 Algorithme de vérification et mise en conformité 92

IV Adaptation de maillages courbes d'ordre élevé **95**
1 Convention de numérotation : numérotation fonctionnelle 95
2 p-adaptation de maillages courbes 97
2.1 Élément Triangulaire 97
2.2 Élément Quadrangulaire 98
3 h-adaptation de maillages courbes 99
3.1 Définitions 99
3.2 Adaptation P_1 101
3.2.1 Calcul de la métrique 101
3.2.2 Calcul de la longueur d'arête 104
3.3 Adaptation des maillages courbes d'ordre k 105
3.3.1 Calcul de la métrique 105
3.3.2 Calcul de la longueur d'arête voulue 107
3.3.3 Raffinement d'un élément courbe 108
3.3.4 Algorithme de raffinement 123

3.4 Schéma d'adaptation stationnaire . . . 124
3.5 Validation . . . 125
3.5.1 Raffinement sur une fonction continue . . . 126
3.5.2 Raffinement sur une fonction discontinue . . . 129

V Résultats numériques **143**
1 Mise en œuvre . . . 143
1.1 Parallélisation . . . 143
1.2 Constantes et post-traitement . . . 150
2 Équations d'Euler . . . 151
2.1 Écoulement subsonique autour d'un cylindre . . . 151
2.1.1 Description du cas test . . . 151
2.1.2 Analyse des résultats . . . 152
2.2 Écoulement subsonique sur un bump . . . 156
2.2.1 Solution initiale sur maillage grossier . . . 157
2.2.2 Solution avec adaptation de maillage . . . 161
2.3 Écoulements autour d'un profil d'aile de section NACA0012 . . . 163
2.3.1 Écoulement subsonique . . . 166
2.3.2 Écoulement supersonique . . . 172
3 Équations de Navier-Stokes . . . 174
3.1 Écoulement subsonique autour d'un cylindre . . . 174
3.1.1 Description du cas test . . . 174
3.1.2 Analyse des résultats . . . 174

Conclusion et perspectives **177**

Annexe 1 **181**
1 Matrices Jacobiennes dans $\mathbb{R}^3$ . . . 181
1.1 Forme des matrices . . . 182
1.2 Valeurs et vecteurs propres . . . 182

Annexe 2 **185**
2 Numérotation locale des DoF*s*'un élément . . . 185

Annexe 3 **187**
3 Format de fichier d'un maillage $\mathcal{NURBS}$ . . . 187

Références bibliographiques **189**

Table des figures

.1.2 Précision, ordre et nombre de degrés de liberté xviii

I.2.2 Conformité d'une discrétisation 11
I.2.3 Exemple de discrétisation du domaine de calcul 12
I.2.4 Résidu global et résidus locaux 13
I.2.5 Limitation d'un cas scalaire d'ordre 2 24
I.2.6 Résidus locaux du schéma centré de LLxF et décentrement 26

II.1.2 Degrés de liberté et points de contrôle 34
II.1.3 Mapping d'un segment de l'espace paramétrique sur une courbe de $\mathbb{R}^3$ . . 37
II.1.4 Exemples de formes élémentaires unitaires de l'espace paramétrique. . . . 38
II.2.2 Exemple de fonctions de base B-splines. 40
II.2.3 Mapping d'un patch paramétrique surfacique sur une surface de $\mathbb{R}^3$ 43
II.2.4 Exemple de fonctions de base $\mathcal{NURBS}$ 45
II.2.5 Propriété d'enveloppe convexe 48
II.2.6 Propriété de transformation projective 49
II.2.7 Exemple de fonctions de base de Bernstein 56
II.3.2 Discrétisation courbe du domaine de calcul 65
II.3.3 Mapping d'un tétraèdre courbe sur un tétraèdre de $\mathbb{R}^3$ 66

III.1.2 Frontière courbe et discrétisation 79
III.2.2 Représentation $\mathcal{NURBS}$ d'un arc de cercle 83
III.2.3 Discrétisation linéaire et discrétisation courbe : maillage initial 84
III.2.4 Discrétisation linéaire et discrétisation courbe : maillage courbe obtenu . . 84

III.3.2 Croisement d'arêtes dans un élément courbe 85
III.3.3 Maillage linéaire anisotrope problématique 86
III.3.4 Maillage courbe anisotrope 87
III.3.5 Élément courbe croisé et élément de contrôle associé 88
III.3.6 Décroisement d'un élément courbe 90
III.3.7 Décroisement d'un maillage 91
III.3.8 Maillage de couche limite 93
III.3.9 Destruction de la couche limite du maillage III.3.8.. 94

IV.1.2 Numérotation fonctionnelle d'un triangle 96
IV.1.3 Numérotation fonctionnelle d'un quadrangle 97
IV.3.2 Métriques isotropes et anisotropes. 100
IV.3.3 Maillage courbe d'ordre 5 et maillage droit subdivisé 106
IV.3.4 Erreurs d'interpolation d'ordre 2, 3 et 4 106
IV.3.5 Algorithme linéique de De Casteljau 110
IV.3.6 Motifs de division élémentaire en dimension 2 112
IV.3.7 Subdivision de deux arêtes d'un élément 113
IV.3.8 *découpe* $1-2$ d'un triangle 114
IV.3.9 Motifs de division élémentaire de la *découpe* $1-4$ *spider* 117
IV.3.10 Motifs de division élémentaire de la *découpe* $1-3$ 118
IV.3.11 *découpe* $1-3$ d'un triangle 120
IV.3.12 Motifs de division élémentaire de la *découpe* $1-4$ classique. 122
IV.3.13 Maillage initial 125
IV.3.14 Valeurs de la fonction f_1 sur la discrétisation initiale 126
IV.3.15 Comparaison des raffinements d'ordre 2, 3, 4 et 5 127
IV.3.16 Raffinement d'ordre 5 créant plus de DoF*s* qu'à l'ordre 2. 129
IV.3.17 Maillage d'ordre 2 en sortie de la procédure d'adaptation. 130
IV.3.18 Maillage d'ordre 3 en sortie de la procédure d'adaptation. 131
IV.3.19 Maillage d'ordre 4 en sortie de la procédure d'adaptation. 132
IV.3.20 Maillage d'ordre 5 en sortie de la procédure d'adaptation. 133
IV.3.21 Valeurs de f_2 sur la discrétisation initiale. 134
IV.3.22 Comparaison des raffinements d'ordre 3, 4 et 5 135
IV.3.23 Maillage d'ordre 2 en sortie de la procédure d'adaptation sur f_2 137
IV.3.24 Maillage d'ordre 3 en sortie de la procédure d'adaptation sur f_2 138
IV.3.25 Maillage d'ordre 4 en sortie de la procédure d'adaptation sur f_2 139

IV.3.26 Maillage d'ordre 5 en sortie de la procédure d'adaptation sur f_2 140
IV.3.27 Discrétisations finales de f_2 141

V.1.2 Décomposition d'un domaine d'ordre 4 144
V.1.3 Analyse de la scalabilité parallèle 149
V.1.4 Déséquilibre de charge 150
V.2.2 Discrétisation du domaine de calcul du cylindre 153
V.2.3 Erreur d'entropie des schémas $\mathcal{RD}$ classique et isogéométrique sur le maillage 9×32 154
V.2.4 Erreur d'entropie des schémas $\mathcal{RD}$ classique et isogéométrique sur le maillage 17×64 154
V.2.5 Coefficient de perte de pression et coefficient de pression à la paroi du cylindre 155
V.2.6 Nombre de Mach pour les schémas $\mathcal{RD}$ classique et isogéométrique sur le maillage 9×32 156
V.2.7 Nombre de Mach pour les schémas $\mathcal{RD}$ classique et isogéométrique sur le maillage 17×64 157
V.2.8 Maillage du bump. 158
V.2.9 Erreur d'entropie des schémas classique et isogéométrique d'ordre 3. . . . 159
V.2.10 Erreur d'entropie du schéma isogéométrique d'ordre 4 159
V.2.11 Nombre de Mach des schémas d'ordre élevé classique et isogéométrique. . 160
V.2.12 Nombre de Mach du schéma isogéométrique d'ordre 4 160
V.2.13 Erreur d'entropie des schémas d'ordre élevé classique et isogéométrique. . 162
V.2.14 Nombre de Mach des schémas d'ordre élevé classique et isogéométrique. . 164
V.2.15 Maillages du bump après trois vagues de subdivisions. 165
V.2.16 Profil d'aile à section NACA0012 et notations associées. 166
V.2.17 Maillage du domaine de calcul du NACA0012 167
V.2.18 Erreur d'entropie pour les schémas $\mathcal{RD}$ classique et isogéométrique 168
V.2.19 Erreur d'entropie à la paroi du naca 169
V.2.20 Erreur d'entropie des schémas classique et isogéométrique au bord d'attaque du naca 170
V.2.21 Nombre de Mach pour $\alpha = 2°$ et un nombre de Mach à l'écoulement libre de 0.63 171
V.2.22 Maillage raffiné obtenu après une première évaluation du choc. 173
V.2.23 Capture de choc par le schéma $\mathcal{RD}$ isogéométrique 173

V.3.2 Résultats obtenus par le schéma $\mathcal{RD}$ isogéométrique 176

A2.2 Numérotation locale d'un élément 185

Introduction

Au cours des dernières décennies, la simulation numérique des écoulements fluides s'est imposée dans de nombreux domaines comme un outil crucial de compréhension et donc de contrôle des phénomènes hydrodynamiques. Le développement de schémas d'ordre élevé permettant la modélisation des mouvements des fluides est ainsi devenu l'un des principaux thèmes de recherche de la CFD (Computational Fluid Dynamics). Du fait des géométries complexes des domaines d'étude, les méthodes de discrétisation géométrique sont également au cœur des axes de recherche actuels.

1 Problématique

En raison des besoins industriels toujours croissants, augmenter la précision des méthodes utilisées lors des simulations numériques reste l'un des thèmes centraux de la recherche en dynamique des fluides. L'erreur d'un schéma numérique étant liée à la fois à la discrétisation du domaine de calcul et à l'ordre de représentation des inconnues, deux solutions complémentaires permettent d'améliorer la précision obtenue : raffiner le maillage utilisé lors des simulations (h-adaptation) et élever l'ordre de la méthode utilisée (p-adaptation).

Lors de simulations numériques d'écoulements à l'ordre élevé, un manque de précision dans la discrétisation du domaine de calcul peut générer des erreurs dominant l'erreur liée à la discrétisation des variables. On observe par exemple l'apparition d'une production d'entropie parasite aux parois du domaine lors de la résolution des équations d'Euler sous forme conservative. Afin de résoudre ce problème, on s'intéresse à son analyse isogéométrique : on considère la géométrie du problème comme l'une de ses variables. On représente donc les frontières et les inconnues du problème à l'aide de la même méthode

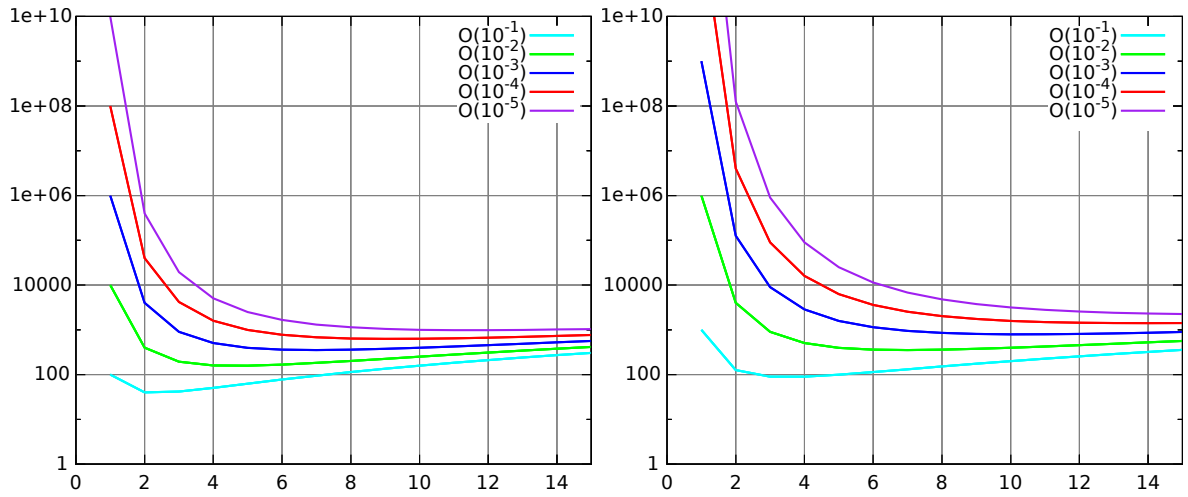
d'approximation. On s'assure ainsi que les erreurs de discrétisation géométrique sont de l'ordre des erreurs de la méthode numérique : elles ne peuvent donc plus engendrer de solutions parasites.

Par ailleurs, bien que l'évolution considérable au cours des dernières décennies de la puissance des machines de calcul permette de lancer des simulations de plus en plus complexes, il reste toujours préférable d'optimiser les rapports entre la précision souhaitée et le nombre de degrés de liberté nécessaires pour l'obtenir.

Pour un calcul de dimension d et d'ordre k, le nombre théorique de DoFs requis pour atteindre la précision ε sur un maillage de taille unitaire satisfait l'inégalité :

$$n_{\text{DoFs}} \leq k^2 \left(\frac{C_k}{\varepsilon}\right)^{\frac{d}{k}}, \text{ avec } C_k \text{ une constante de proportionnalité.}$$

La figure .1.2 représente, pour différents ordres et pour différentes précisions voulues, le nombre maximal théorique de DoFs nécessaires.



(a) dimension 2 (b) dimension 3

Figure .1.2 – Nombre de degrés de liberté n_{DoFs} nécessaires pour atteindre la précision voulue à l'ordre k.

Nous nous intéresserons donc dans ce document à l'ensemble des outils nécessaires à la résolution des équations de la mécanique des fluides par l'analyse isogéométrique, depuis la génération d'un maillage courbe précis pour l'analyse isogéométrique jusqu'au schéma

aux résidus distribués isogéométrique développé en passant par la construction d'outils d'adaptation de maillages courbes préservant la géométrie.

2 État de l'art au début de cette thèse

Cette section vise à donner au lecteur un bref aperçu de l'historique des méthodes isoparamétriques et isogéométriques ainsi des techniques d'adaptation de maillage associées. Cet historique est loin d'être exhaustif, nous nous contentons ici de fournir quelques-unes des références croisées tout au long de notre travail.

En 1968, Ergatroudis, Irons et Zienkiewicz introduisent, dans le cadre de la mécanique des solides, le concept d'*analyse isoparamétrique* [1]. Le principe est simple : la géométrie du problème est considérée comme une de ses variables et les fonctions de base utilisées pour approcher les inconnues sont également utilisées pour l'approximation du maillage. Ainsi, l'approximation du maillage est du même ordre que la méthode numérique employée. Au cours des années suivantes, du fait de leur simplicité et de leur efficacité à représenter les déformations, l'emploi des méthodes isoparamétriques se généralise au sein de la communauté de la mécanique des solides ([2], [3], [4]). Les problèmes considérés sont essentiellement des problèmes de déformation et de rupture de structures simples telles que des poutres et des anneaux. Les maillages utilisés sont, quant à eux, quadrangulaires.

Les méthodes isoparamétriques peinent cependant à s'imposer en mécanique des Fluides. En effet, la génération de maillages courbes représentant les géométries complexes pouvant être rencontrées dans ce domaine demeure un obstacle : l'équation paramétrique des frontières n'est pas nécessairement connue, et, lorsqu'elle l'est, cette équation est souvent rationnelle et d'ordre très élevé.

En 1997, Bassi et Rebay découvrent qu'une représentation linéaire par morceaux des frontières est à l'origine d'une production d'entropie parasite lors de la résolution des équations d'Euler ([5]). Ce constat relance l'intérêt pour l'approximation courbe des géométries et l'analyse isoparamétrique. En particulier, Krivodonova et Berger étudient l'implémentation de conditions aux limites pour des géométries approchées par des suites d'arcs de cercle ([6]).

Le concept d'analyse isogéométrique est introduit en 2005 par Hughes, Cottrell et Brazilevs ([7]). La méthode est proche de celle de l'analyse isoparamétrique, les mêmes fonctions de base permettent d'approcher la géométrie et les variables du problème, mais,

dans leur travail, les fonctions de base sont issues de la CAO du problème. Les calculs sont alors réalisés sous l'hypothèse que la représentation des frontières du domaine fournie par la CAO est exacte. Cette méthode utilise en particulier une famille de fonctions de base permettant la représentation exacte des coniques : les fonctions de bases $\mathcal{NURBS}$ ([8]).

Dès lors, les études sur les méthodes isogéométriques se multiplient : Hughes et ces collaborateurs poursuivent leurs recherches ([9], [10], [11]) de même que Buffa et son équipe ([12], [13], [14]). Leur approche se fonde sur des méthodes numériques de type éléments finis ou Galerkin discontinu et sur l'utilisation de maillages issus de la CAO, donc sur des maillages non conformes constitués de patchs et d'éléments quadrangulaires. Ces maillages ne vérifient pas toujours l'hypothèse de représentation exacte des frontières : il peuvent contenir des trous au niveau desquels il est alors nécessaire de reconstruire la géométrie. Les problèmes étudiés sont divers : mécanique des structures, magnétohydrodynamique, équations de Navier-Stokes...

Bien que moins répandues, quelques recherches s'intéressent également aux méthodes isogéométriques adaptées aux maillages conformes non-structurés : Sevilla présente dans [15] le couplage, sur des maillages triangulaires conformes, de l'analyse isogéométrique et de la méthode des éléments finis tandis que Vymazal *et al.* étudient, toujours sur des maillages triangulaires conformes, l'adaptation à l'analyse isoparamétrique d'un schéma aux résidus distribués de type LDA ([16]).

Dans le premier des deux documents cités, le domaine de calcul est discrétisé par un mailleur linéaire par morceaux classique. Les frontières sont ensuite remplacées par les paramétrisations issues de leurs CAOs. Les fonctions de bases utilisées sont donc les fonctions de base $\mathcal{NURBS}$. Dans le second document, le domaine est discrétisé par un maillage linéaire par morceaux dont les éléments sont assemblés pour former des éléments d'ordre élevé. Les fonctions de bases choisies pour représenter ces éléments d'ordre élevé sont les fonctions lagrangiennes.

À ce jour, le principal challenge de l'utilisation des méthodes isogéométriques sur des maillages non-structurés conformes reste la génération de maillages courbes représentant exactement la géométrie.

Parmi les recherches sur la génération de maillages courbes conformes, nous noterons celles de Talbert et Parkinson ([17]) qui étudient la création de maillages quadrilatéraux dont les frontières sont des courbes de Bézier spécifiées par l'utilisateur ([17]), ainsi que celles, plus récentes, de Georges *et al.* ([18]) qui décrivent une méthode de génération de maillages triangulaires B-Splines d'ordre 3.

L'obtention de résultats numériques précis à moindre coût de calcul nécessite, entre autre, de pouvoir adapter la répartition des degrés de liberté du maillage à la solution du problème considéré. Les méthodes d'adaptation se fondent sur la construction d'un maillage permettant de contrôler l'erreur d'interpolation commise.

Il s'agit donc, en premier lieu, d'estimer cette erreur. Il existe de nombreux estimateurs d'erreur (voir [19] pour un récapitulatif des estimateurs d'erreur existants) parmi lesquels l'estimateur d'erreur d'interpolation qui sera utilisé dans ce travail ([20]). Cet estimateur est anisotrope et fournit une majoration de la norme $\mathcal{L}_\infty$ de l'erreur d'interpolation lagrangienne commise à l'ordre 2. Notons par ailleurs que quelques documents évoquent la construction d'estimateurs d'erreur d'ordre élevé ([21], [22]).

Plusieurs techniques permettent alors l'adaptation de maillage.

- La p-adaptation consiste à élever le degré du maillage.

  Pour des maillages non-conformes constitués de patchs, l'élévation d'ordre est locale au patch ([23]).

  Dans le cas de maillages conformes, cette méthode d'adaptation s'applique à la totalité du domaine de calcul et demande simplement d'élever l'ordre des fonctions de base du schéma numérique utilisé.
- La h-adaptation consiste à adapter la taille et la forme des éléments du maillage.

  Pour des maillages non-conformes constitués de patchs, la h-adaptation se fonde sur le raffinement et le déraffinement de la grille d'éléments constituant le patch. Ainsi, la h-adaptation ne permet pas le raffinement local des éléments, il s'applique au patch. Pour adapter localement un élément, il faut alors introduire de nouvelles fonctions de base : les T-Splines et les T-$\mathcal{NURBS}$ ([24]).

  Dans le cas de maillages conformes, deux techniques de raffinement sont possibles. La première approche consiste à créer un nouveau maillage respectant les tailles et directions d'arêtes fournies par l'estimateur d'erreur ([25]), la seconde se fonde sur la fusion et la subdivision des éléments du maillage initial ([26]). Ces méthodes sont locales à chaque élément.
- La k-adaptation développée par Hughes consiste à combiner p et h-adaptation d'un patch ([23]). Le patch est d'abord élevé à l'ordre voulu (notons le k) puis raffiné par insertion nodale. Les fonctions de bases associées au patch sont alors C^{k-1} continues entre les éléments. Cette technique d'adaptation étant étroitement liée à la notion de patch et de vecteur nodal, elle n'a pas d'équivalent parmi les méthodes classiques d'adaptation.

À l'échelle de l'élément, de nombreux documents traitent des méthodes d'élévation d'ordre et de subdivision d'éléments courbes, qu'ils soient triangulaires ou non ([27], [28]). Les méthodes de h-adaptation de maillages courbes restent cependant limitées par la construction d'un estimateur d'erreur optimal pour les discrétisations d'ordre élevé.

3 Contributions de cette thèse et organisation du mémoire

3.1 Contributions de cette thèse

Nous avons, dans cette thèse, développé tous les outils nécessaires à la résolution d'un problème par l'analyse isogéométrique, de la génération de maillages courbes conformes et non-structurés jusqu'au schéma numérique en passant par les différentes méthodes d'adaptation de maillage. Nous résumerons ici l'essentiel des travaux réalisés. Nous présenterons ceux-ci dans l'ordre des besoins d'une simulation numérique, nous ne respecterons donc ni l'ordre des chapitres de ce manuscrit ni l'ordre réel du travail effectué. Tous les codes de calcul que nous citerons sont développés à l'INRIA Bordeaux Sud-Ouest.

- **Génération de maillages courbes d'ordre** 3 **:** la première étape de toute modélisation numérique consiste à discrétiser le domaine de calcul. Le mailleur développé, nommé « NOMesh », permet, à partir d'un maillage linéaire par morceaux, de générer des maillages courbes conformes d'ordre 3. La frontière du domaine est approchée par une courbe circulaire par morceaux. Il est possible d'imposer des angles et des segments droits lors de la création du maillage. Les frontières seront exactement représentées pour des maillages linéaires ou circulaires par morceaux. Dans ce cas, le maillage généré sera isogéométrique. Le maillage obtenu en sortie garanti la continuité des fonctions de base de Bernstein rationnelles sur la totalité du domaine de discrétisation.

- **Couplage de l'analyse isogéométrique et d'un schéma aux résidus distribués :** le second apport de cette thèse est le couplage du schéma aux résidus distribués limité et stabilisé de Lax-Friedrichs et de l'analyse isogéométrique. Le schéma adapté permet la résolution des équations stationnaires d'Euler et de Navier-Stokes. Nous avons pris soin de conserver la propriété de compacité du schéma initial. Le

schéma développé respecte exactement la géométrie fournie par le maillage. En dimension 2, nous l'avons testé pour des calculs d'ordre 2 à 6 sur des maillages conformes, non structurés et hybrides. En dimension 3, nous avons implémenté les tétraèdres d'ordre 3 et 4.
Nous avons implémenté ce schéma dans le code « FluidBox ».

- **Adaptation isogéométrique de maillages courbes d'ordre élevé :**
  Pour finir, nous avons construit des outils d'élévation d'ordre (p-adaptation) et de raffinement (h-adaptation) pour les maillages courbes d'ordre élevé. Ces outils conservent donc la géométrie du maillage d'entrée.
  - La p-adaptation développée permet, pour des maillages (courbes ou non) bidimensionnels d'ordre 2 à 5, d'élever l'ordre du maillage jusqu'à l'ordre 6. Elle est intrinsèque au code FluidBox.
  - L'outil de raffinement conçu permet le raffinage isotrope des maillages triangulaires (donc bidimensionnels) d'ordre 2 à 6. Bien que développée pour des maillages courbes, la méthode est applicable aux maillages droits d'ordre élevé. Le maillage est d'abord subdivisé en sous-maillage P_1. La métrique P_1 classique est calculée sur chacun des nœuds de ce sous-maillage. On associe cette métrique à notre maillage initial d'ordre élevé et on calcule la longueur d'arête courbe voulue. Les éléments dont au moins l'une des arêtes est trop longue sont alors subdivisés en sous-éléments. Les sous-éléments obtenus par subdivision d'un élément courbe sont également courbes. Le code de calcul développé se nomme « CuMA ».

3.2 Organisation du mémoire

Nous consacrerons le premier chapitre de ce manuscrit à la présentation du cadre théorique dans lequel s'inscrit cette thèse. Nous rappellerons donc l'écriture des équations régissant les mouvements des fluides : les équations de Navier-Stokes. Nous rappellerons également la simplification de ces équations en équations d'Euler dans le cas d'un fluide non visqueux. Par soucis de concision, nous nous focaliserons dans la suite de cette thèse sur les équations d'Euler. Nous terminerons ce premier chapitre par la présentation du schéma numérique utilisé pour résoudre ces équations : le schéma aux résidus distribué limité et stabilisé de Lax-Friedrichs.

Le second chapitre de cette thèse s'intéresse à l'analyse isogéométrique et à son cou-

plage avec le schéma précédemment décrit. Nous introduirons tout d'abord quelques concepts essentiels des méthodes isogéométriques puis nous construirons la famille de fonctions de base choisie pour représenter à la fois notre géométrie et nos variables. Nous conclurons ce chapitre en présentant le schéma aux résidus distribués isogéométrique développé. Nous nous intéresserons plus particulièrement aux différences engendrées par la représentation courbe des éléments.

En troisième partie, nous détaillerons la génération de maillages courbes à partir de maillages linéaire par morceaux « classiques ». Nous expliquerons en premier lieu la reconstruction des frontières adoptée puis nous nous intéresserons à la validité du maillage ainsi généré.

Nous dédirons le quatrième chapitre aux deux méthodes d'adaptation de maillage développées, à savoir l'élévation d'ordre et le raffinement. Ces deux méthodes préserveront la géométrie courbe du domaine.

Nous conclurons ce mémoire par quelques résultats numériques. Nous débuterons par plusieurs cas tests non-visqueux puis nous poursuivrons par un cas test visqueux.

Chapitre I

Cadre théorique

Ce chapitre vise à mettre en place de manière synthétique le cadre général de cette thèse.

En premier lieu nous rappellerons les fondements théoriques permettant de construire les équations de Navier-Stokes. Nous présenterons également, dans le cas d'un fluide parfait adiabatique, leur forme simplifiée : les équations d'Euler. Par la suite, nous nous intéresserons plus particulièrement à ces dernières. Nous étudierons les conditions d'existence et d'unicité d'une solution à ce système d'équations.

Dans une seconde partie, nous détaillerons le schéma numérique utilisé pour résoudre les équations d'Euler ainsi que ses principales propriétés.

Nous terminerons par une brève description du schéma numérique de résolution des équations de Navier-Stokes.

1 Dynamique des fluides

Dans cette section nous nous réfèrerons essentiellement aux documents [29] et [30]. En tant que milieu continu (*c.-à-d.* dont les propriétés caractéristiques sont continues), le comportement d'un fluide est régi par les équations fondamentales de la mécanique des milieux continus. En particulier, il devra satisfaire les *lois de conservations*, *d'état* et *de comportement*.

1.1 Équations de conservation

Soit Ω un sous-domaine ouvert de $\mathbb{R}^d$, on appelle *système de lois de conservation* un système d'équations aux dérivées partielles pouvant s'écrire sous la forme :

$$\frac{\partial \boldsymbol{U}}{\partial t} + \overrightarrow{\nabla}.\overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}) = 0,\ t \geq 0 \tag{I.1}$$

avec $\overrightarrow{\mathcal{F}} = (\mathbf{F}_1, \cdots, \mathbf{F}_n)$ un vecteur-flux de fonctions-flux $\mathbf{F}_i$ de Ω dans $\mathbb{R}^n$ et $\boldsymbol{U} = \boldsymbol{U}(t, \boldsymbol{X})$ une fonction à valeurs vectorielles de $\mathbb{R}^+ \times \mathbb{R}^d$ dans Ω telle que :

$$\boldsymbol{X} \in \mathbb{R}^d \text{ et } \boldsymbol{U}(t, \boldsymbol{X}) = (U_1(t, \boldsymbol{X}), \cdots, U_n(t, \boldsymbol{X}))$$

Un tel système traduit la conservation de la variable $\boldsymbol{U}$ au cours de l'évolution du temps t. Notons $\boldsymbol{X} = (x_i)_{i=1,d}$ les coordonnées de l'espace et considérons un fluide. Ce fluide est caractérisé par sa densité $\rho(t, \boldsymbol{X})$, sa vitesse $\vec{\boldsymbol{u}}(t, \boldsymbol{X})$ et son énergie totale par unité de masse $E(t, \boldsymbol{X})$. Trois systèmes d'équations de conservation régissent l'évolution de ρ, $\vec{\boldsymbol{u}}$ et E au cours du temps :

- l'équation de *conservation de la masse* ou de *continuité* reliant densité et vitesse :

$$\frac{\partial \rho}{\partial t} + \overrightarrow{\nabla}.(\rho \vec{\boldsymbol{u}}) = 0 \tag{I.2}$$

- le système d'équations de *conservation de la quantité de mouvement* reliant la densité et la vitesse au tenseur des contraintes noté $\boldsymbol{\sigma}$:

$$\frac{\partial \rho \vec{\boldsymbol{u}}}{\partial t} + \overrightarrow{\nabla}.(\rho \vec{\boldsymbol{u}} \otimes \vec{\boldsymbol{u}}) = \overrightarrow{\nabla}.\boldsymbol{\sigma} \tag{I.3}$$

- l'équation de *conservation de l'énergie* reliant la densité, la vitesse, l'énergie totale du système (ρE) et le flux de chaleur massique noté $\vec{\boldsymbol{q}} = (q_1, q_2, q_3)$:

$$\frac{\partial (\rho E)}{\partial t} + \overrightarrow{\nabla}.(\rho E \vec{\boldsymbol{u}}) = \overrightarrow{\nabla}.(\boldsymbol{\sigma} \vec{\boldsymbol{u}}) - \overrightarrow{\nabla}.\vec{\boldsymbol{q}} \tag{I.4}$$

Le système formé par ces 5 équations est sous-déterminé (il comporte six inconnues pour le tenseur des contraintes, trois pour les composantes du flux de chaleur, trois autres pour celles de la vitesse, une inconnue pour la densité et enfin une dernière pour l'énergie, soit quatorze inconnues au total). On utilise donc les lois d'état et de comportement du fluide pour clore notre problème.

1.2 Lois d'état

On introduit deux nouvelles variables thermodynamiques, la pression absolue p et la température T. Ces variables dépendent de l'état physique du système et sont donc appelées variables d'état et reliées entre elles par les lois d'état. Dans ce travail, nous

utiliserons la loi des gaz polytropiques ou gaz calorifiquement parfaits :

$$p = \rho R T, \tag{I.5}$$

où R est une constante dépendant de la masse molaire du gaz considéré. Notons C_v la capacité calorifique à volume constant et $e = E - \frac{1}{2}\|\vec{\boldsymbol{u}}\|^2$ l'énergie interne du système, alors $T = \frac{e}{C_v}$. Pour finir, on note C_p la capacité calorifique à pression constante telle que $C_p = R + C_v$ et $\gamma = \frac{C_p}{C_v}$ le rapport des chaleurs spécifiques [1]. L'équation (I.5) peut alors être réécrite sous une forme simplifiée :

$$p = (\gamma - 1)\rho e. \tag{I.6}$$

1.3 Lois de comportement

Les lois de comportements traduisent la réponse d'un milieu continu aux sollicitations extérieures auxquelles il est soumis. Mathématiquement, ces lois sont donc exprimées comme :

- une relation entre le tenseur des contraintes et certaines valeurs mécaniques et thermodynamiques du système à un instant donné,
- une relation (elle aussi instantanée) entre le flux de chaleur et ces mêmes valeurs.

La réaction d'un milieu dépend de sa nature physique, dans cette thèse, nous nous consacrerons uniquement à l'étude des fluides de type *Newtoniens*.

Définition I.1.1 (*Fluide Newtonien*)

Un milieu continu est dit *Newtonien* lorsque :

- le tenseur des contraintes visqueuses $\boldsymbol{\sigma}$ est une fonction linéaire du tenseur des déformations $\mathbf{s} = (s_{ij})_{i,j} = \left(\frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right)_{i,j}$, donc lorsqu'il existe un tenseur du quatrième ordre, $\mathbf{K}$, tel que

$$\boldsymbol{\sigma} = \mathbf{K} \overline{\overline{\otimes}}\, \mathbf{s} + \boldsymbol{\sigma_0},$$

- le milieu est isotrope.

1. γ est alors une constante et vaut 1.4 pour un gaz diatomique tel que l'air

La loi de comportement la plus communément utilisée satisfaisant ces deux conditions est la *loi de Newton* :

$$\boldsymbol{\sigma} = 2\mu\boldsymbol{s} + (\lambda \operatorname{Tr}(\boldsymbol{s}) - p)\,\mathbf{I}. \tag{I.7}$$

μ et λ sont deux coefficients de viscosité appelés respectivement premier et second coefficient de Lamé, $\mathbf{I}$ est la matrice identité et Tr l'opérateur de trace. On adjoint à cette équation la *relation de Stokes* qui relie les valeur de λ et μ :

$$3\lambda + 2\mu = 0. \tag{I.8}$$

Afin de satisfaire le second principe de la thermodynamique, outre la relation I.8, il faut nous assurer que la dissipation thermique de notre système est positive [31]. Cette positivité est garantie si l'on suppose les dissipations de chaleur intrinsèques comme nulles et si l'on satisfait la *loi de conduction de Fourier*. Pour un fluide Newtonien, cette loi s'écrit :

$$\vec{\boldsymbol{q}} = -\lambda_c \overrightarrow{\nabla T}. \tag{I.9}$$

λ_c désigne la viscosité thermodynamique.

1.4 Équations de la mécanique des fluides

1.4.1 Équations de Navier-Stokes

Les lois de comportement et d'état décrites permettent de réécrire les équations de conservation. Sous les hypothèses précédemment admises, on obtient alors les équations de Navier-Stokes :

$$\left\{\begin{array}{rcl} \dfrac{\partial \rho}{\partial t} + \overrightarrow{\nabla}.(\rho\vec{\boldsymbol{u}}) & = & 0, \\ \dfrac{\partial \rho\vec{\boldsymbol{u}}}{\partial t} + \overrightarrow{\nabla}.(\rho\vec{\boldsymbol{u}}\otimes\vec{\boldsymbol{u}} + p\mathbf{I}) & = & (\lambda+\mu)\vec{\nabla}\left(\overrightarrow{\nabla}.\vec{\boldsymbol{u}}\right) + \mu\overrightarrow{\Delta}\vec{\boldsymbol{u}}, \\ \dfrac{\partial \rho E}{\partial t} + \overrightarrow{\nabla}.((\rho E + p)\,\vec{\boldsymbol{u}}) & = & \overrightarrow{\nabla}.\left(\frac{\lambda_c}{C_v}\overrightarrow{\nabla T} + \left(2\mu\boldsymbol{s} + \lambda\overrightarrow{\nabla}.(\vec{\boldsymbol{u}})\,\mathbf{I}\right)\vec{\boldsymbol{u}}\right), \\ T & = & C_v\left(E - \dfrac{1}{2}\|\vec{\boldsymbol{u}}\|^2\right). \end{array}\right. \tag{I.10}$$

Elles permettent de décrire, tant du point de vue macroscopique que microscopique, le mouvement des fluides quels qu'ils soient (phénomènes aérodynamiques, courants océaniques...). À l'heure actuelle, bien que leur formulation soit relativement simple, ces équations demeurent impossible à résoudre dans leur intégralité. En effet, elles comportent des

couplages multi-physiques et multi-échelles ainsi que des termes non-linéaires. Dans ce travail, nous nous intéresserons plus particulièrement à un modèle simplifié des équations de Navier-Stokes, les équations d'Euler.

1.4.2 Équations d'Euler

Dans le cadre d'un *fluide parfait adiabatique* (*c.-à-d.* d'un fluide dans lequel on néglige les effets visqueux ($\lambda = \mu = 0$) et où l'on considère la conductivité thermique comme nulle ($\vec{\boldsymbol{q}} = 0$)), les équations de Navier-Stokes peuvent être réécrite sous une forme plus réduite, les équations d'Euler :

$$\left\{\begin{array}{lcccl} \dfrac{\partial \rho}{\partial t} & + & \nabla.(\rho\vec{\boldsymbol{u}}) & = & 0, \\ \dfrac{\partial \rho\vec{\boldsymbol{u}}}{\partial t} & + & \overrightarrow{\nabla}.(\rho\vec{\boldsymbol{u}} \otimes \vec{\boldsymbol{u}} + p\boldsymbol{I}) & = & 0, \\ \dfrac{\partial \rho E}{\partial t} & + & \nabla.((\rho E + p)\,\vec{\boldsymbol{u}}) & = & 0. \end{array}\right. \tag{I.11}$$

On note $\boldsymbol{U} = \begin{pmatrix} \rho \\ \rho\vec{\boldsymbol{u}} \\ \rho E \end{pmatrix}$ le vecteur des variables conservatives et ${}^{T}\overrightarrow{\boldsymbol{\mathcal{F}}} = (\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3)$ avec $\mathbf{F}_i = \begin{pmatrix} \rho u_i \\ \rho u_i \vec{\boldsymbol{u}} + p\boldsymbol{\delta}_i \\ (\rho E + p)u_i \end{pmatrix}$ le vecteur flux associé. $\vec{\boldsymbol{u}}_i$ et $\boldsymbol{\delta}_i$ désignent respectivement la vitesse et le vecteur unitaire associés à la $i^{\text{ème}}$ dimension.

Le système formé par les équations d'Euler peut être réécrit sous forme d'une équation vectorielle :

$$\frac{\partial \boldsymbol{U}}{\partial t} + \overrightarrow{\nabla}.\left(\overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{U})\right) = 0. \tag{I.12}$$

L'équation (I.12) peut être reformulée comme une équation quasi-linéaire :

$$\frac{\partial \boldsymbol{U}}{\partial t} + \left(\frac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_1(\boldsymbol{U})}{\partial \boldsymbol{U}}, \frac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_2(\boldsymbol{U})}{\partial \boldsymbol{U}}, \frac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_3(\boldsymbol{U})}{\partial \boldsymbol{U}}\right).\overrightarrow{\nabla \boldsymbol{U}} = \frac{\partial \boldsymbol{U}}{\partial t} + \vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla \boldsymbol{U}} = 0 \tag{I.13}$$

où $\vec{\boldsymbol{\lambda}}$ est le vecteur des matrices jacobiennes du flux.

Les équations d'Euler forment un *système hyperbolique* (caractéristique d'un phénomène d'advection) de lois de conservation.

Définition I.1.2 (*système hyperbolique*)

Soit (I.14) un système de lois de conservation d'inconnue $\boldsymbol{W} : \mathbb{R}^d \times \mathbb{R}^+ \mapsto \mathbb{R}^n$.

$$\frac{\partial \boldsymbol{W}}{\partial t} + \overrightarrow{\nabla}.\overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{W}) = \frac{\partial \boldsymbol{W}}{\partial t} + \sum_{i=1}^{d} A_i(\boldsymbol{X}, t)\frac{\partial \boldsymbol{W}}{\partial x_i} = 0, \quad \boldsymbol{X} \in \mathbb{R}^d,\ t \in \mathbb{R}^+. \tag{I.14}$$

Ce système est dit hyperbolique si :
$\forall \boldsymbol{\xi} \in S^{d-1} = \{\boldsymbol{X} \in \mathbb{R}^d, \|\boldsymbol{X}\|_{L^2} = 1\}$, la matrice $\mathcal{A}(\boldsymbol{\xi}) = \sum_{i=1}^{d} A_i \xi_i$ est diagonalisable à valeurs et vecteurs propres réels.

On s'intéresse dans la suite de ce paragraphe à l'existence et à l'unicité d'une solution aux équations d'Euler. Pour une description plus complète des propriétés des systèmes hyperboliques non-linéaires et de leur solutions, le lecteur peut se référer à [32], [33] et [34].
On associe à l'équation (I.12) la donnée initiale :

$$\boldsymbol{U}(t = 0, \boldsymbol{X}) = \boldsymbol{U}_0(\boldsymbol{X}). \tag{I.15}$$

On rappelle la définition d'une *solution classique* d'un problème hyperbolique non-linéaire de lois de conservation.

Définition I.1.3 (*solution classique*)

Soit (I.14) un problème hyperbolique non-linéaire de lois de conservation et soit :

$$\boldsymbol{W}(t = 0, \boldsymbol{X}) = \boldsymbol{W}_0 \text{ tel que } \boldsymbol{W}_0 \in C^0(\mathbb{R}^d) \tag{I.16}$$

la condition initiale associée.
Si :

- $\boldsymbol{W}$ est solution de (I.14,I.16),
- $\boldsymbol{W} \in C^1(\mathbb{R}^d \times \mathbb{R}^+)$,

alors $\boldsymbol{W}$ est solution classique de (I.14,I.16).

La non-linéarité du problème (I.12,I.15) peut être à l'origine de l'apparition et de la propagation de discontinuités, et ce, quelque soit la régularité de la donnée initiale $\boldsymbol{U}_0$. Ce problème n'admet donc pas forcément de solution classique.

La capture des solutions discontinues nécessite donc d'introduire la notion de *solution faible*.

Définition I.1.4 (*solution faible*)

On considère le problème (I.14,I.16). Notons :

- $L^\infty_{loc}(\mathbb{R}^d \times \mathbb{R}^+) = \{f \text{ mesurable}, f_{|K} \in L^\infty(\Omega), \forall K \text{ compact } \subset \Omega\}$,
- $C^1_0(\mathbb{R}^d \times \mathbb{R}^+) = \{f \text{ à support compact dans } \left(\mathbb{R}^d \times \mathbb{R}^+\right), f \in C^1(\mathbb{R}^d \times \mathbb{R}^+)\}$.

Soit $W_0 \in L^\infty_{loc}(\mathbb{R}^d)$. Si :

- $\boldsymbol{W} \in L^\infty_{loc}(\mathbb{R}^d \times \mathbb{R}^+)$.
- $\forall g \in C^1_0(\mathbb{R}^d \times \mathbb{R}^+)$, W vérifie :

$$\int_0^{+\infty} \int_{\mathbb{R}^d} \left(\boldsymbol{W}.\frac{\partial g}{\partial t} + \overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{W}).\overrightarrow{\nabla g} \right) d\boldsymbol{X}\,dt + \int_{\mathbb{R}^d} (\boldsymbol{W}_0(\boldsymbol{X}).g(\boldsymbol{X},0))\,d\boldsymbol{X} = 0 \quad \text{(I.17)}$$

alors $\boldsymbol{W}$ est solution faible de (I.14,I.16).

Bien évidemment toute solution classique du problème (I.14) est aussi solution faible. De même une solution faible à dérivées premières continues est solution classique de ce même problème. De plus, si une solution faible présente des discontinuités, cette solution doit nécessairement satisfaire les relations de *Rankine-Hugoniot* (équation (I.18)) à travers chaque discontinuité. On en déduit les conditions nécessaires et suffisantes données par le théorème I.1.1 pour la construction de solutions faibles.

Théorème I.1.1

Soit $\boldsymbol{W}$ une fonction $C^1(\mathbb{R}^d \times \mathbb{R}^+)$ par morceaux dont les domaines de définitions, dans le plan $(\boldsymbol{X},t)$, sont séparés par des courbes $\Sigma = \left\{\boldsymbol{X} = \boldsymbol{g}(t), t^- \leq t \leq t^+\right\}$ telles que $\boldsymbol{g}(t) \in C^1(\mathbb{R}^+)$. $\boldsymbol{W}$ est une solution faible de (I.14,I.16) si et seulement si :

- $\boldsymbol{W}$ est solution classique sur chaque domaine de définition.
- $\boldsymbol{W}$ vérifie les relations de Rankine-Hugoniot le long des discontinuités Σ :

$$\boldsymbol{\sigma}(t)\left[\boldsymbol{W}^+(t) - \boldsymbol{W}^-(t)\right] = \overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{W}^+(t)\right) - \overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{W}^-(t)\right) \quad \text{(I.18)}$$

avec $\boldsymbol{\sigma}(t) = \overrightarrow{\nabla \boldsymbol{g}(t)}$ la vitesse de la discontinuité et $\boldsymbol{W}^\pm = \lim_{\varepsilon \to 0} \left(\boldsymbol{W}\left(\boldsymbol{\sigma}\left(t \pm \varepsilon\right), t\right)\right)$.

Il n'y a malheureusement pas unicité des solutions faibles. Un contre-exemple très célèbre est fourni par le problème de Burger (problème qui peut être obtenu par reformu-

lation des équations d'Euler quand $\gamma = 3$) :

$$\text{Soit } (x,t) \in \left(\mathbb{R}, \mathbb{R}^+\right), \text{ trouver } W : \mathbb{R} \times \mathbb{R}^+ \mapsto \mathbb{R} \text{ tel que :}$$
$$\begin{cases} \dfrac{\partial W}{\partial t} + \dfrac{\partial}{\partial x}\left(\dfrac{W^2}{2}\right) = 0 & \\ W_0(x) = 0, & \text{si } x < 0 \\ W_0(x) = 1, & \text{si } x > 0 \end{cases} \tag{I.19}$$

Les fonctions :

$$W(x,t) = \begin{cases} 0 & \text{si} \quad x < 0 \\ \dfrac{x}{t} & \text{si} \quad 0 < x < t \\ 0 & \text{si} \quad t < x \end{cases} \quad \text{et} \quad W(x,t) = \begin{cases} 0 & \text{si} \quad 2x < t \\ 1 & \text{si} \quad t < 2x \end{cases}$$

sont toutes deux solutions faibles du problème de Burger pour la même donnée initiale (on vérifie facilement qu'elles satisfont le problème (I.19) et les relations de Rankine-Hugoniot). Nous savons cependant que notre équation représente un phénomène physique, la propagation. Ce problème admet donc, intuitivement, une unique solution faible physiquement acceptable, les autres solutions faibles étant des solutions « parasites ». On sait également que d'un point de vue physique, une solution acceptable est forcément dissipative. Ainsi, afin d'éliminer les solutions non physiques, nous chercherons à distinguer, parmi toutes les solutions faibles, celles dissipant l'entropie. Ceci nécessite de définir la notion de *couple entropie-flux d'entropie*.

Définition I.1.5 (*couple entropie-flux d'entropie*)

Soient $\omega \in C^1(\mathbb{R}^d)$ et $\mathscr{F} \in C^1(\mathbb{R}^d)$, si :

- La fonction $\omega : \boldsymbol{U} \mapsto \omega(\boldsymbol{U})$ est convexe
- $\nabla_{\boldsymbol{U}} \mathscr{F} = \nabla_{\boldsymbol{U}} \omega . \nabla_{\boldsymbol{U}} \overrightarrow{\boldsymbol{\mathcal{F}}}$

alors le couple $(\omega, \mathscr{F})$ est appelé couple entropie-flux d'entropie.

Le caractère dissipatif d'une fonction $\boldsymbol{W}$ se traduit alors mathématiquement par l'équation (I.20) d'où la définition d'une *solution faible entropique*.

Définition I.1.6 (*solution faible entropique*)

On considère le problème (I.14,I.16). Soit $\boldsymbol{W}_0 \in L^\infty(\mathbb{R}^d)$. On appelle solution faible entropique de (I.14,I.16) sur $\left(\mathbb{R}^d \times \mathbb{R}^+\right)$ une solution faible telle que pour tout couple

entropie-flux d'entropie $(\omega, \mathscr{F})$:

$$\frac{\partial \omega(\boldsymbol{W})}{\partial t} + \overrightarrow{\nabla}.\mathscr{F}(\boldsymbol{W}) \leq 0 \text{ au sens des distributions dans } \left(\mathbb{R}^d \times \mathbb{R}^+\right). \tag{I.20}$$

Dans le cas d'un problème scalaire, le théorème I.1.2 démontré par Kruzkhov (voir [35]) donne l'unicité de la solution entropique.

Théorème I.1.2 (*Kruzkhov*)

Soit $\boldsymbol{W}_0 \in L^\infty(\mathbb{R}^d)$ et $\overrightarrow{\boldsymbol{\mathcal{F}}} \in C^1(\mathbb{R}^d)$, le problème (I.14,I.16) admet une et une seule solution entropique $\boldsymbol{W} \in L^\infty(\mathbb{R}^d \times \mathbb{R}^+)$.

Dans le cas qui nous intéresse, celui des systèmes d'équations, l'unicité de cette solution reste cependant à prouver.

Ainsi, la section suivante sera consacrée à la recherche d'une solution faible entropique des équations d'Euler (équation (I.11)) associées à la condition initiale (I.15). Présentons le schéma numérique utilisé pour résoudre notre problème.

2 Schéma numérique pour la résolution des équations d'Euler

Cette section se fonde essentiellement sur les travaux d'A. Larat [36] et M. Ricchiuto [37] [38] ainsi que sur un certain nombre de travaux de R. Abgrall [39] [40] [41]. On trouvera un état de l'art non exhaustif des schémas aux résidus distribués (notés $\mathcal{RDS}$ ou schémas $\mathcal{RD}$) dans [37] (page 20) et [36]. Pour une explication plus détaillée de la construction de ce type de schémas, le lecteur pourra se référer au chapitre 4 de [36]. Les schémas $\mathcal{RD}$, également connus sous le nom de *Fluctuation Splitting methods*, ont été introduits au début des années 80 par P.L. Roe [42]. Le principe de ces schémas est simple, on nomme *résidu* le flux global quittant une cellule puis on répartit ce résidu sur les différents nœuds de la cellule suivant un certain nombre de règles qui garantiront les bonnes propriétés du schéma. De tels schémas permettent la capture de solutions discontinues sans l'ajout de termes supplémentaires, de plus leur stencil très compact les rendent facile à implémenter et à paralléliser. Après un bref rappel du problème étudié et de quelques généralités sur la discrétisation numérique, nous présenterons dans cette section la formulation d'un schéma $\mathcal{RD}$ d'ordre 2 puis, à partir de ce schéma, la construction d'un schéma d'ordre élevé. Nous rappellerons également les principales propriétés du schéma ainsi construit.

2.1 Discrétisation numérique

Soit Ω un ouvert borné de $\mathbb{R}^d$, d désignant la dimension de travail[2]. On associe au problème (I.11,I.15) posé sur $\Omega \times \mathbb{R}^+$ un ensemble de conditions aux bords ($\mathcal{BC}$) définies sur $\Gamma = \partial\Omega$. Par exemple, dans le cas des équations d'Euler, notre fluide n'étant pas visqueux, on suppose qu'il n'y a pas de forces de frottement aux parois. De plus, on suppose que le fluide ne peut franchir la frontière Γ. On peut alors imposer une condition de paroi glissante sur Γ. Nous voulons donc résoudre le système :

$$\begin{cases} \dfrac{\partial \boldsymbol{U}}{\partial t} + \overrightarrow{\nabla}.\overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}) = 0, \quad \forall \boldsymbol{X} \in \Omega \\ (\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}})_{|\Gamma} = 0, \quad \mathcal{BC} \text{ de type paroi glissante sur } \Gamma \\ \boldsymbol{U}(\boldsymbol{X}, t=0) = \boldsymbol{U}_0(\boldsymbol{X}) \end{cases} \tag{I.21}$$

Discrétiser un problème consiste à chercher la solution du problème en un nombre fini de points du domaine Ω. Pour M_i un point du problème discret, on note $\boldsymbol{U}_i$ la valeur de $\boldsymbol{U}$ en M_i. M_i est appelé degré de liberté (noté DoF pour *Degree of Freedom*). L'ensemble des valeurs de $\boldsymbol{U}$ aux degrés de liberté forme la solution discrète de notre problème. On reconstruit ensuite une solution approchée $\boldsymbol{U}^h$ de l'équation (I.21) telle que $\boldsymbol{U}_h$ soit continue sur Ω et interpole la solution discrète (*c.-à-d.* telle que $\boldsymbol{U}_h(\mathrm{M}_i) = \boldsymbol{U}_i$). En général, $\boldsymbol{U}_h(\boldsymbol{X}, t)$ est calculée comme étant l'interpolée de la solution discrète sur un ensemble de fonctions polynomiales formant une base de l'espace discret.

La suite de cette sous-section vise essentiellement à introduire les notations qui seront utilisées dans cette thèse.

On note $\mathbb{P}^k$ l'espace des fonctions polynomiales de degré k. On découpe Ω en éléments T_i tels que l'ensemble des T_i soit une discrétisation conforme (définition I.2.1) de Ω. On note cet ensemble $\mathcal{T}_h$, h étant la longueur caractéristique du maillage (*c.-à-d.* le diamètre du plus grand élément).

Définition I.2.1 (*Discrétisation conforme*)

La discrétisation $\mathcal{T}_h$ est dite conforme si tous les éléments contigus de la discrétisation se raccordent sans recouvrement ni trou et partagent les mêmes nœuds sur leur inter-

2. par soucis de clarté, on prendra $d = 2$ dans la plupart des exemples traités, l'extension à la $3^{\text{ème}}$ dimension étant triviale.

face commune (voir figure I.2.2).
En termes mathématiques, $\mathcal{T}_h = \{T_i\}$ est une discrétisation conforme de Ω si et seulement si :

- $\bigcup_{\mathrm{T}_i \in \Omega} \mathrm{T}_i = \Omega$
- $\mathrm{T}_i \bigcap_{\mathrm{T}_i, \mathrm{T}_j \in \Omega} \mathrm{T}_j$ est soit vide, soit réduite à un sommet commun, soit réduite à une arête commune.

On suppose que $\mathcal{T}_h$ est composé d'éléments triangulaires et quadrangulaires d'aires strictement positives et que ses éléments ne se recouvrent pas. T_i désigne le $i^{\text{ème}}$ élément et M_i le $i^{\text{ème}}$ DoF de $\mathcal{T}_h$ (voir figure I.2.3).

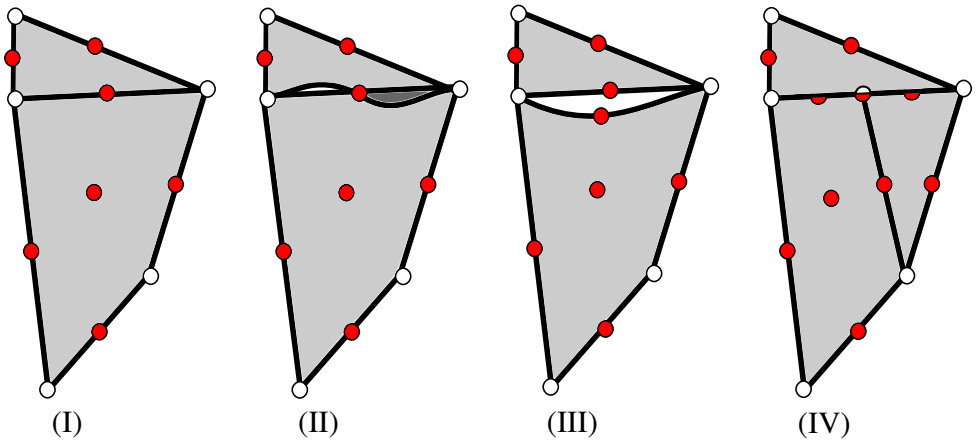


Figure I.2.2 – (I) Discrétisation conforme,
(II) (III) et (IV) Discrétisations non-conformes : recouvrement et trou entre les 2 éléments pour la discrétisation (II), trou pour (III) et nœuds différents à l'interface des éléments pour (IV).

Définition I.2.2 (*Maillage*)

Soit $\mathcal{T}_h$ une discrétisation de degré k. On nomme *maillage* l'ensemble des DoF*s* M_i et des éléments T_i de $\mathcal{T}_h$. On note ce maillage $\mathcal{M}_h\left(\mathbb{P}^k\right)$.

Par soucis de clarté, si il n'y a pas d'ambiguïté, $\mathcal{M}_h\left(\mathbb{P}^k\right)$ sera simplement noté $\mathcal{M}_h$.

Définition I.2.3 (*Maillage conforme*)

Un maillage $\mathcal{M}_h$ est dit conforme si il est associé à une discrétisation conforme.

Nous désignerons par :

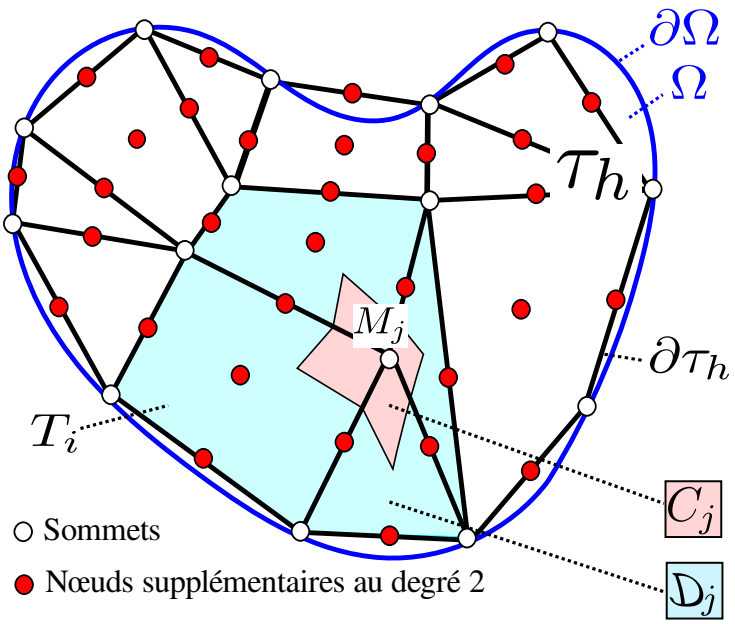


Figure I.2.3 – Exemple de discrétisation de degré 2 du domaine Ω. Au degré 1 seuls les nœuds blancs existent.

- M_{j_i} le $i^{\text{ème}}$ DoF de l'élément T_j,
- $|\mathrm{T}_j|$ l'aire de T_j,
- $\boldsymbol{V}_i$ l'ensemble des sommets reliés par une arête au nœud M_i.
- $\mathcal{D}_i$ le voisinage de M_i (*c.-à-d.* l'ensemble des éléments T tels que M_i soit un DoF de T),
- $\mathcal{D}_h^k$ l'ensemble des volumes duaux associés aux degrés de liberté de $\mathcal{M}_h\left(\mathbb{P}^k\right)$,
- N_{Tot} le nombre total de DoF*s* de $\mathcal{T}_h$.

Enfin, pour n_{DoFs} DoF*s* dans un élément T, on associera au DoF i le volume élémentaire :

$$|C_i| = \frac{\sum_{\mathrm{T}\in\mathcal{D}_i} |T|}{n_{\text{DoFs}}}.$$

2.2 Schéma aux résidus distribués

Considérons le problème (I.21) sous sa forme stationnaire :

$$\vec{\nabla}.\vec{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}_h) = 0. \tag{I.22}$$

On intègre l'équation sur Ω et on reformule l'intégrale de la divergence du flux sur un élément comme une intégrale de contour à l'aide de la formule de Gauss :

$$\int_{\Omega} \vec{\nabla}.\vec{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}_h) = \sum_{T\in\Omega} \int_{\partial \mathrm{T}} \vec{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}_h) d\partial \mathrm{T}. \tag{I.23}$$

On définit Φ^{T}, le *résidu global* (ou *fluctuation*) *associé à l'élément* T, tel que :

$$\Phi^{\mathrm{T}} = \int_{\partial \mathrm{T}} \vec{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}_h) d\partial \mathrm{T}$$

On définit également une quantité appelée *résidu local nodal associé à l'élément* T *et au DoF* i et notée $\Phi_i^{\mathrm{T}}(\boldsymbol{U}_h)$ ou simplement Φ_i^{T} (voir figure I.2.4).

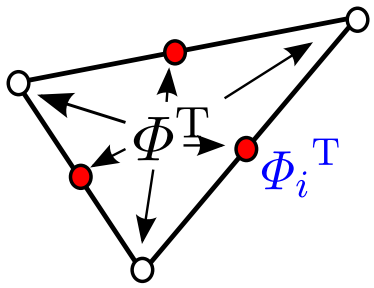


Figure I.2.4 – Résidu global et résidus locaux

Un schéma aux résidus distribués s'écrit :

$$\begin{cases} \forall \mathrm{M}_i, \ \sum\limits_{\mathrm{T}\in\mathcal{D}_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h) = 0. \\ \boldsymbol{U}_i^0 \text{ donné} \end{cases} \tag{I.24}$$

En pratique, on approche la solution du problème (I.24) comme la limite pour $n \mapsto \infty$ du schéma pseudo-itératif :

$$\begin{cases} \forall \mathrm{M}_i, \ |C_i| \dfrac{\boldsymbol{U}_i^{n+1} - \boldsymbol{U}_i^n}{\Delta t} + \sum\limits_{\mathrm{T}\in\mathcal{D}_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h) = 0. \\ \boldsymbol{U}_i^0 \text{ donné} \end{cases} \tag{I.25}$$

Si $\sum\limits_{i\in T} \Phi_i^{\mathrm{T}} = \int_{\partial \mathrm{T}} \vec{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}_h) d\partial \mathrm{T} = \Phi^{\mathrm{T}}$, nous pouvons vérifier que ce schéma est bien conservatif [43]. Il sera explicite si Φ_i^{T} est évalué au temps n et implicite si Φ_i^{T} est calculé au temps $n+1$. La discrétisation temporelle du schéma est détaillée sous-section 2.5.
Il reste maintenant à définir le résidu local nodal Φ_i^{T}. Le choix de la distribution de ce

résidu déterminera le type de schéma utilisé (schéma de Lax-Friedrichs, schéma LDA, schéma SUPG...). Un récapitulatif des différents schémas $\mathcal{RD}$s est disponible section 4.4 de [36]. C'est également de cette définition que découleront les principales propriétés numériques du schéma. À ce stade, il est donc important de se demander sous quelles hypothèses notre schéma converge.

2.3 Propriétés du schéma

Rappelons tout d'abord que des conditions nécessaires pour s'assurer qu'un schéma numérique converge vers la solution de l'équation qu'il modélise, sont les conditions de *consistance* et de *stabilité*. La consistance du schéma exprime tout simplement la capacité de celui-ci à rester suffisamment proche de l'équation au cours de la procédure de calcul. La stabilité quant à elle traduit la capacité du schéma à ne pas amplifier les perturbations apparaissant lors de la résolution du système. À la notion de consistance est souvent associée celle de précision numérique qui exprime la distance mathématique entre la solution approchée trouvée ($\boldsymbol{U}_h$) et la solution exacte ($\boldsymbol{U}^*$) des équations. Cette distance est majorée par le cumul de l'erreur de projection ((1) équation (I.26)) et de l'erreur de troncature ((2) de la même équation) du schéma :

$$\|\boldsymbol{U}^* - \boldsymbol{U}_h\|_E \leq \underbrace{\|\boldsymbol{U}^* - \Pi_h(\boldsymbol{U}^*)\|_E}_{(1)} + \underbrace{\|\Pi_h(\boldsymbol{U}^*) - \boldsymbol{U}_h\|_E}_{(2)} \tag{I.26}$$

E étant l'espace de définition de $\boldsymbol{U}^*$ et $\boldsymbol{U}_h$ et Π_h étant l'opérateur de projection dans l'espace discrétisé. Dans cette sous-section on suppose le problème (I.21) stationnaire, donc on cherche une solution de :

$$\begin{cases} \vec{\nabla}.\vec{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}) = 0, \quad \forall \boldsymbol{X} \in \Omega \\ (\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}})_{|\Gamma} = 0, \quad \mathcal{BC} \text{ de type paroi glissante sur } \Gamma \\ \boldsymbol{U}(\boldsymbol{X}, t=0) = \boldsymbol{U}_0(\boldsymbol{X}) \end{cases} \tag{I.27}$$

Du point de vue du schéma $\mathcal{RD}$ on cherche donc à résoudre :

$$\sum_{\mathrm{T} \in \mathcal{D}_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h) = 0. \tag{I.28}$$

Nous débuterons ce paragraphe avec les principaux résultats concernant la consistance et la précision du schéma. Nous traiterons ensuite des conditions de conservation de la

monotonicité de la linéarité. Pour plus de détails sur les propriétés des schémas $\mathcal{RD}$ on pourra se référer à [39] et [36].

2.3.1 Consistance

Tout d'abord, définissons la consistance.

Définition I.2.4 (*Consistance*)

Un schéma numérique est dit consistant si pour toute solution suffisamment régulière $\boldsymbol{U}^*$, l'erreur de troncature (voir le terme (2) de l'équation (I.26)) du schéma tends vers 0 quand h, la taille caractéristique du maillage et Δt, le pas de temps, tendent vers 0.

Afin d'établir les hypothèses de travail dans lesquelles cette consistance est vérifiée il nous faut introduire quelques définitions.

Définition I.2.5 (*Maillage régulier*)

Le maillage M_h est dit régulier si :

$$\exists C_1, C_2 \in \mathbb{R}^+, \text{ tels que } \forall \mathrm{T} \in \mathcal{M}_h,\ C_1 \leq \frac{h_1}{h_2} \leq C_2, \tag{I.29}$$

où h_1 et h_2 sont les diamètres des cercles circonscrit et inscrit au triangle T.

Un maillage est donc régulier si le rapport de taille entre deux éléments du maillage, quels qu'ils soient, est borné. On note $\mathscr{X}_h^k$ l'ensemble des fonctions v_h de $\mathcal{D}_h^k$ telles que pour tout i, $v_{h|\mathcal{D}_i}$ est constante. On admet que les résidus locaux nodaux de T sont continus vis à vis de la solution discrète aux nœuds de l'élément T (hypothèse I.2.1).

Hypothèse I.2.1 (*continuité des résidus locaux*)

Soit $\mathcal{M}_h$ un maillage conforme et régulier. Quelque soit $C \in \mathbb{R}^+$ et quelque soit $\boldsymbol{U}_h \in \mathscr{X}_h^k$ telle que $\|\boldsymbol{U}_h\|_{L^\infty} \leq C$, il existe C' dépendant uniquement de C et de $\mathcal{M}_h$ tel que :

$$\forall \mathrm{T} \in \mathcal{M}_h, \forall M_i \in \mathrm{T}, \quad |\Phi_i^{\mathrm{T}}| \leq C'(C, \mathcal{M}_h) h \sum_{M_j \in \mathrm{T}} |\boldsymbol{U}_i - \boldsymbol{U}_j|.$$

On suppose également que :

Hypothèse I.2.2 (*régularité du flux approché*)

L'approximation $\overrightarrow{\boldsymbol{\mathcal{F}}}_h$ du flux $\overrightarrow{\boldsymbol{\mathcal{F}}}$ est une fonction continue de $\mathscr{X}_h^k$ dans $\mathscr{X}_h^k$ et quelque soit la suite de solutions $(\boldsymbol{U}_h)_h$, bornée indépendamment de h dans $L^\infty(\mathbb{R}^d)$ et conver-

geant vers la solution exacte $\boldsymbol{U}$ dans $L^2_{\text{loc}}\left(\mathbb{R}^d\right)$, on a :

$$\lim_{h\to 0} \|\overrightarrow{\boldsymbol{\mathcal{F}}_h}(\boldsymbol{U}_h) - \overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{U})\|_{L^1_{loc}(\mathbb{R}^d)} = 0.$$

Cette hypothèse est vérifiée pour toute approximation $\mathbb{P}^k$ de $\overrightarrow{\boldsymbol{\mathcal{F}}}$.

La consistance du schéma $\mathcal{RD}$ est donné par le théorème I.2.1 (voir [44]).

Théorème I.2.1 (*de type Lax-Wendroff*)

Soit $(\boldsymbol{U}_h)_h$ une suite de solutions du problème (I.25) associées aux maillages M_h. On suppose que les maillages sont tous conformes et réguliers et que les hypothèses I.2.1 et I.2.2 sont vérifiées. $\boldsymbol{U} \in L^2(\mathbb{R}^d)$ est solution faible de (I.25) si il existe une constante C dépendant uniquement de C_1 et C_2 telle que :

$$\sup_h \sup_{\boldsymbol{X}\in\Omega} |\boldsymbol{U}_h(\boldsymbol{X})| \leq C.$$

$$\lim_{h\to 0} \|\boldsymbol{U} - \boldsymbol{U}_h\|_{L^2_{loc}(\mathbb{R}^d)} = 0.$$

Comme précédemment expliqué, notre schéma étant consistant et restant, pas après pas, proche de l'équation qu'il modélise, il est intéressant de chercher à quantifier dans quelle mesure notre solution approchée est une bonne approximation de la solution exacte.

2.3.2 Précision

Commençons par définir la notion de précision. On note Π_h^k l'opérateur de projection $\mathbb{P}^k$ dans l'espace discrétisé.

Définition I.2.6 ($\mathcal{RDS}$ *précis à l'ordre* $(k+1)$)

Soit $\boldsymbol{U}^*$ une solution faible du problème (I.27), un schéma est dit précis à l'ordre $(k+1)$ si l'erreur de troncature ε définie par :

$$\varepsilon = \sup_{\substack{\|v\|\leq 1 \\ v\in L^\infty}} \left[\sum_{\mathrm{M}_i\in\mathcal{M}_h} \left(v(\mathrm{M}_i).\left(\sum_{\mathrm{T}\in\mathcal{D}_i} \Phi_i^{\mathrm{T}}\left(\Pi_h^k(\boldsymbol{U}^*)\right)\right)\right)\right],$$

vérifie :

$$\forall v \in C_0^1(\Omega),\ \varepsilon \leq c(\boldsymbol{U}^*, \mathcal{T}_h)\, h^{k+1}.$$

où c est une constante dépendant de la solution $\boldsymbol{U}^*$ et de la discrétisation $\mathcal{T}_h$ du problème.

Cette définition traduit simplement qu'un schéma est d'ordre $(k+1)$ si son erreur de troncature est d'ordre $(k+1)$.

Proposition I.2.1 (*Ordre de précision d'un schéma* $\mathcal{RD}$)

Soit k un entier strictement positif, on suppose que :

- $\mathcal{M}_h$ est conforme et régulier,
- quelque soit $\boldsymbol{U}^*$ solution faible, le flux approché évalué à la projection de la solution, $\overrightarrow{\boldsymbol{\mathcal{F}}}_h\left(\Pi_h^k(\boldsymbol{U}^*)\right)$, est d'ordre $(k+1)$,
- quelque soit l'élément T, les résidus locaux évalués à la projection de la solution, $\Phi_i^{\mathrm{T}}\left(\Pi_h^k(\boldsymbol{U}^*)\right)$, sont d'ordre $(k+d)$,

alors un schéma $\mathcal{RD}$ est d'ordre $(k+1)$.

Ainsi, la précision globale du schéma est contrôlée par la précision de calcul du résidu et, pour un résidu approché sur des fonctions de base $\mathbb{P}^k$, le schéma $\mathcal{RD}$ sera d'ordre $(k+1)$. Par abus de langage, nous parlerons donc souvent dans ce document de fonctions de base d'*ordre* k (ou d'ordre polynomial k) pour désigner les fonctions de base de degré polynomial k.

Jusqu'à présent, nous avons considéré des solutions faibles du problème étudié, cependant, nous avons précédemment expliqué que la solution recherchée est une solution faible entropique (I.20) du problème. Le paragraphe suivant s'intéresse donc aux conditions sous lesquelles notre schéma converge vers cette solution.

2.3.3 Condition de conservation de la monotonicité

Définition I.2.7 (*Schéma monotone*)

On considère un problème scalaire. Un schéma est dit monotone si et seulement si :

$$\forall\, U_i^n, U_j^n \text{ solutions du problème aux nœuds } M_i \text{ et } M_j \text{ et au temps } t^n,$$
$$U_i^n \leq U_j^n \Rightarrow U_i^{n+1} \leq U_j^{n+1}$$

Ainsi, pour une condition initiale monotone, l'utilisation d'un schéma monotone interdira l'apparition d'oscillations parasites. La notion de monotonie ne peut cependant être utilisée dans notre cas, en effet, un schéma monotone est au maximum d'ordre 1 (voir

[45]). On substitue donc à la notion de schéma monotone celle de schéma préservant la monotonie (noté $\mathcal{MP}$ pour *Monotonicity preserving*). Les théorèmes I.2.2 et I.2.3 (voir [46]) donnent les relations entre les différents types de schémas.

Théorème I.2.2

schéma monotone $\Rightarrow$ schéma $\mathcal{MP}$

Théorème I.2.3

Pour un schéma linéaire, schéma $\mathcal{MP} \Leftrightarrow$ schéma monotone

Comme un schéma monotone est d'ordre 1, un schéma $\mathcal{MP}$ d'ordre élevé est nécessairement non-linéaire (*théorème de Godunov*). Commençons par donner sous quelles conditions un schéma $\mathcal{RD}$ est $\mathcal{MP}$. On suppose que la quantité de fluctuation Φ_i^{T} envoyée au $i^{\text{ème}}$ sommet de l'élément T satisfait l'hypothèse I.2.3.

Hypothèse I.2.3 (*forme du résidu local nodal*)

Le résidu local nodal peut s'écrire sous la forme :

$$\Phi_i^{\mathrm{T}} = \sum_{j, M_j \in T} c_{ij}^{T} (\boldsymbol{U}_i - \boldsymbol{U}_j),$$

où les c_{ij}^{T} sont des coefficients indépendants des données dans le cas d'un schéma linéaire ou dépendants de celles-ci dans le cas d'un schéma non linéaire.

En pratique, cette hypothèse est toujours vérifiée. On choisit des coefficients c_{ij} satisfaisant la propriété I.30 ([43]).

Propriété I.2.1 (*Conservation de la monotonicité pour un schéma* $\mathcal{RD}$)

Un schéma $\mathcal{RD}$ est $\mathcal{MP}$ si :

$$\forall M_i \in \mathcal{M}_h,\ \forall \mathrm{T} \in \mathcal{D}_i,\ \forall M_j \in \mathrm{T}, \quad c_{ij}^{\mathrm{T}} \geq 0, \tag{I.30}$$

et, pour un schéma explicite, si la condition de type CFL suivante est satisfaite :

$$\forall M_i \in \mathcal{M}_h, \Delta t \left[\frac{\sum_{\mathrm{T} \in \mathcal{D}_i} \sum_{M_j \in \mathrm{T}} c_{ij}^{T}}{|C_i|} \right] \leq 1. \tag{I.31}$$

Si les c_{ij} sont indépendant de $\boldsymbol{U}^h$, le schéma est linéaire et $\mathcal{MP}$ et donc, il est monotone

et d'ordre 1.
Pour travailler à un ordre supérieur, nous allons chercher à construire, à partir de ce schéma, un schéma non-linéaire $\mathcal{MP}$. Notons :

- $\Phi_i^{T,M}$ le signal d'un schéma monotone et $c_{ij}^{T,M}$ les coefficients permettant d'écrire ce signal sous la forme I.2.3.
- $\Phi_i^{T,H}$ le signal associé à un schéma d'ordre élevé et $c_{ij}^{T,H}$ les coefficients permettant d'écrire ce signal sous la forme I.2.3.

On sait que :

$$\sum_i \Phi_i^{T,H} = \Phi^{\mathrm{T}} = \sum_i \Phi_i^{T,M}. \tag{I.32}$$

De plus, afin de garantir la propriété de conservation de la monotonicité, nous voulons :

$$\Phi_i^{T,M}\Phi_i^{T,H} \geq 0. \tag{I.33}$$

Démonstration : Comme :

$$\begin{aligned} \Phi_i^{T,H} &= \frac{\Phi_i^{T,H}}{\Phi^{\mathrm{T}}}\frac{\Phi^{\mathrm{T}}}{\Phi_i^{T,M}}\Phi_i^{T,M} \\ &= \frac{\Phi_i^{T,H}}{\Phi^{\mathrm{T}}}\frac{\Phi^{\mathrm{T}}}{\Phi_i^{T,M}}\sum_{j,M_j\in T} c_{ij}^{T,M}\left(\boldsymbol{U}_i - \boldsymbol{U}_j\right), \end{aligned} \tag{I.34}$$

on peut écrire le résidu local d'ordre élevé sous la forme :

$$\Phi_i^{T,H} = \sum_{j,M_j\in T} c_{ij}^{T,H}\left(\boldsymbol{U}_i - \boldsymbol{U}_j\right), \quad \text{avec} \quad c_{ij}^{T,H} = \frac{\Phi_i^{T,H}}{\Phi^{\mathrm{T}}}\frac{\Phi^{\mathrm{T}}}{\Phi_i^{T,M}}c_{ij}^{T,M}. \tag{I.35}$$

$c_{ij}^{T,M} \geq 0$, donc :

$$\begin{aligned} c_{ij}^{T,H} \geq 0 &\iff \frac{\Phi_i^{T,H}}{\Phi^{\mathrm{T}}}\frac{\Phi^{\mathrm{T}}}{\Phi_i^{T,M}} \geq 0 \\ &\iff \Phi_i^{T,M}\Phi_i^{T,H} \geq 0 \quad . \end{aligned} \tag{I.36}$$

■

Soient β_i^T et $\beta^{*,\mathrm{T}}$ des coefficients de distribution tels que :

$$\begin{aligned} \beta_i^T &= \frac{\Phi_i^{T,M}}{\Phi^{\mathrm{T}}}, \\ \beta_i^{*,\mathrm{T}} &= \frac{\Phi_i^{T,H}}{\Phi^{\mathrm{T}}}. \end{aligned}$$

Les deux conditions (I.32) et (I.33) peuvent être réécrites :

$$\begin{cases} \sum_{i=1}^{n_{\text{DoFs}}} \beta_i^T = \sum_{i=1,n_{\text{DoFs}}} \beta_i^{*,\text{T}} = 1 & \text{Condition de conservation} \\ \beta_i^T \beta_i^{*,\text{T}} \geq 0 & \text{Condition de conservation de la monotonicité} \end{cases} \tag{I.37}$$

La construction, à partir d'un schéma linéaire monotone, des $\beta_i^{*,\text{T}}$ satisfaisant ces conditions permet d'obtenir des schémas $\mathcal{MP}$ non-linéaires. Il existe de nombreux mappings possibles entre les β_i^T et les $\beta_i^{*,\text{T}}$ d'un élément, ces mappings sont appelés *procédures de limitations*. Il nous faut maintenant vérifier sous quels critères sur les $\beta_i^{*,\text{T}}$ ces nouveaux schémas sont d'ordre $(k+1)$.

2.3.4 Conservation de la linéarité

Rappelons que l'ordre de précision d'un schéma $\mathcal{RD}$ est donné proposition I.2.1. Le schéma doit donc satisfaire la condition nécessaire sur le résidu local ([44]) :

$$\Phi_i^{\text{T}}\left(\Pi_h^k(\boldsymbol{U}^*)\right) = \mathcal{O}(h^{k+d}).$$

Soit $\overrightarrow{\boldsymbol{\mathcal{F}}_h}\left(\Pi_h^k(\boldsymbol{U}^*)\right)$ un flux approché d'ordre $(k+1)$,

$$\begin{aligned} &\text{Si} \quad \Phi^{\text{T}}\left(\Pi_h^k(\boldsymbol{U}^*)\right) = \int_{\partial \text{T}} \left(\overrightarrow{\boldsymbol{\mathcal{F}}_h}\left(\Pi_h^k(\boldsymbol{U}^*)\right)\right).\vec{\boldsymbol{n}}\, d\boldsymbol{X} = \mathcal{O}(k+d), \\ &\text{et si} \quad \Phi_i^{\text{T}}\left(\Pi_h^k(\boldsymbol{U}^*)\right) = \beta_i^{*,\text{T}} \Phi^{\text{T}}\left(\Pi_h^k(\boldsymbol{U}^*)\right), \end{aligned}$$

alors le schéma $\mathcal{RD}$ est précis à l'ordre $(k+1)$ si les $\beta_i^{*,\text{T}}$ sont bornés indépendamment de h. On dit alors que le schéma conserve la linéarité.

Définition I.2.8 (*schéma conservant la linéarité*)

On dit qu'un schéma $\mathcal{RD}$ conserve la linéarité (noté $\mathcal{LP}$ pour *Linearity Preserving*) si ses coefficients de distribution β_i^T sont uniformément bornés par un coefficient C indépendant de la taille du maillage h :

$$\max_{\text{T}\in\mathcal{M}_h} \max_{M_i\in\text{T}} |\beta_i^{\text{T}}| < C < +\infty, \quad \forall \Phi^{\text{T}}, u_h, u_h^0, \tau, \ldots \tag{I.38}$$

Finalement, les coefficients de distribution $\beta_i^{*,\text{T}}$ d'un schéma $\mathcal{MP}$ d'ordre $(k+1)$ doivent donc satisfaire les conditions (I.37) et l'équation (I.38). Nous détaillons la li-

mitation utilisée dans notre schéma paragraphe 2.4.2. Maintenant que nous avons défini les critères que doivent remplir nos résidus locaux, nous pouvons construire un schéma d'ordre élevé.

2.4 Écriture d'un schéma aux résidus distribués d'ordre élevé

Comme expliqué paragraphe 2.3.3, il nous faut d'abord construire un signal linéaire monotone $\Phi_i^{\mathrm{T},M}$, puis, par limitation de celui-ci, nous pourrons construire un signal d'ordre élevé.

2.4.1 Flux de Lax-Friedrichs

Nous ne détaillerons que la construction du schéma utilisé, à savoir le schéma de *Lax-Friedrichs* (noté LxF et aussi connu sous le nom de schéma de *Rusanov*).
Supposons le problème (I.27) monodimensionnel et, pour $\alpha \geq 0$ convenablement choisi, notons :

$$\widehat{\overrightarrow{\boldsymbol{\mathcal{F}}}}_{i+\frac{1}{2}} = \frac{1}{2}\left(\overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_{i+1}\right) + \overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_i\right) - \alpha_{i+\frac{1}{2}}\left(\boldsymbol{U}_{i+1} - \boldsymbol{U}_i\right)\right),$$

Le schéma de Lax-Friedrichs monodimensionnel s'écrit alors :

$$\widehat{\overrightarrow{\boldsymbol{\mathcal{F}}}}_{i+\frac{1}{2}} - \widehat{\overrightarrow{\boldsymbol{\mathcal{F}}}}_{i-\frac{1}{2}} = 0. \tag{I.39}$$

Comme :

$$\begin{array}{rcl}
\widehat{\overrightarrow{\boldsymbol{\mathcal{F}}}}_{i+\frac{1}{2}} - \widehat{\overrightarrow{\boldsymbol{\mathcal{F}}}}_{i-\frac{1}{2}} & = & \left(\widehat{\overrightarrow{\boldsymbol{\mathcal{F}}}}_{i+\frac{1}{2}} - \overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_i\right)\right) + \left(\overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_i\right) - \widehat{\overrightarrow{\boldsymbol{\mathcal{F}}}}_{i-\frac{1}{2}}\right) \\
& = & \left(\frac{1}{2}\left(\overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_{i+1}\right) - \overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_i\right) + \alpha_{i+\frac{1}{2}}\left(\boldsymbol{U}_i - \boldsymbol{U}_{i+1}\right)\right)\right) \\
& + & \left(\frac{1}{2}\left(\overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_i\right) - \overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_{i-1}\right) + \alpha_{i-\frac{1}{2}}\left(\boldsymbol{U}_i - \boldsymbol{U}_{i-1}\right)\right)\right)
\end{array}$$

Posons :

$$\left\{\begin{array}{rcl}
\Phi_i^{i+\frac{1}{2}} & = & \frac{1}{2}\left(\overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_{i+1}\right) - \overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_i\right) + \alpha_{i+\frac{1}{2}}\left(\boldsymbol{U}_i - \boldsymbol{U}_{i+1}\right)\right) \\
\Phi_{i+1}^{i+\frac{1}{2}} & = & \frac{1}{2}\left(\overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_{i+1}\right) - \overrightarrow{\boldsymbol{\mathcal{F}}}\left(\boldsymbol{U}_i\right) + \alpha_{i+\frac{1}{2}}\left(\boldsymbol{U}_{i+1} - \boldsymbol{U}_i\right)\right)
\end{array}\right. \tag{I.40}$$

le schéma de Lax-Friedrichs (I.39) peut se réécrire sous la forme du schéma $\mathcal{RD}$:

$$\Phi_i^{i+\frac{1}{2}} + \Phi_i^{i-\frac{1}{2}} = 0, \text{ ou encore, } \sum_{\mathrm{T}\in\mathcal{D}_i} \Phi_i^{\mathrm{T}} = 0, \text{ T désignant ici un segment.} \tag{I.41}$$

On généralise aux dimensions supérieures la forme du signal donnée par le système (I.40) :

$$\Phi_i^{T,M} = \frac{1}{n_{\text{DoFs}}} \left(\Phi^{\text{T}} + \alpha^{\text{T}} \left[\sum_{M_j \in T} (\boldsymbol{U}_i - \boldsymbol{U}_j) \right] . \right) \tag{I.42}$$

Le signal envoyé ne dépendant que du nombre de degrés de liberté, cette généralisation convient également aux discrétisations d'ordre élevé. Le signal défini satisfait bien la relation de conservation $\sum_{i=1}^{n_{\text{DoFs}}} \Phi_i^{\text{T}} = \Phi^{\text{T}}$.
De plus, considérons le problème vectoriel quasi-linéaire (I.13) sous sa forme stationnaire :

$$\vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla \boldsymbol{U}} = 0 \tag{I.43}$$

et considérons une discrétisation $\mathbb{P}^k$ sur des fonctions de base φ_i^k :

$$\begin{aligned} \Phi^{\text{T}} &= \int_{\text{T}} \vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla \boldsymbol{U}_h} d\boldsymbol{x} \\ &= \sum_{i \in \text{T}} \boldsymbol{U}_i \left(\int_{\text{T}} \vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla \varphi_i^k} d\boldsymbol{x} \right) \end{aligned} \tag{I.44}$$

Posons $\bar{k}_i^{k,T} = \int_{\text{T}} \vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla \varphi_i^k} d\boldsymbol{x}$, l'équation I.44 peut être réécrite sous la forme :

$$\Phi^{\text{T}} = \sum_{i \in \text{T}} \bar{k}_i^{k,T} \boldsymbol{U}_i = - \sum_{j \in \text{T}} \bar{k}_j^{k,T} (\boldsymbol{U}_i - \boldsymbol{U}_j). \tag{I.45}$$

Le résidu de Lax-Friedrichs (I.42) peut être écrit sous la forme de l'hypothèse (I.2.3) avec $c_{ij}^T = \frac{\alpha^{\text{T}} - \bar{k}_j^k}{n_{\text{DoFs}}}$.
La condition de positivité des coefficients c_{ij}^T est donc satisfaite pour des α^{T} tels que :

$$\alpha^{\text{T}} \geq \max_{j \in \text{T}} \rho\left(\bar{k}_j^k\right), \tag{I.46}$$

où ρ désigne le rayon spectral de la matrice $\bar{k}_j^k$. On remarque que dans le cas scalaire, il suffit de s'assurer que :

$$\forall T \in \mathcal{M}_h, \alpha^{\text{T}} \geq \max_{j \in \text{T}} \bar{k}_j^k. \tag{I.47}$$

Comme ce schéma est $\mathcal{MP}$ et linéaire, il est également d'ordre 1. Ce schéma est connu pour être très dissipatif, de plus, rien ne nous garantit que les coefficients de distribution du schéma soient bornés, et, pour finir, c'est un schéma centré, qui a donc le désavantage de ne pas inclure totalement la physique du problème (puisque les signaux envoyés aux

nœuds d'un élément restent toujours symétriques quelque soit la position du nœud par rapport au flux). Il offre cependant l'avantage d'être généralisable à tout type d'éléments (triangles, quadrangles, tétraèdres,...), à tous les ordres, et d'être très simple à coder. La procédure de limitation décrite paragraphe suivant permet d'obtenir le schéma de Lax-Friedrichs limité (noté LLxF pour *Limited Lax-Friedrichs scheme*) précis à l'ordre $(k+1)$.

2.4.2 Limitation

Rappelons tout d'abord que l'on nomme *coefficients de distribution du signal monotone* les termes β_i^T tels que $\beta_i^T = \frac{\Phi_i^{T,M}}{\Phi^{\mathrm{T}}}$. De même, on nomme *coefficients de distribution du signal non-linéaire* les termes $\beta_i^{T,*}$ tels que $\beta_i^{T,*} = \frac{\Phi_i^{T,H}}{\Phi^{\mathrm{T}}}$.

La procédure de limitation peut être vue comme le mapping des coefficients β_i^T sur les coefficients $\beta_i^{T,*}$. Dans ce travail, on utilise le limiteur `PSI` (Positive Streamwise Invariant) de Struijs [47]. Le principe de ce limiteur est simple. Une représentation graphique correspondant à l'explication qui suit dans le cas scalaire et pour $n_{\mathrm{DoFs}} = 3$ est fourni figure I.2.5. On travaille dans un unique élément T, on s'affranchira donc de l'indice T dans les notations. Commençons par présenter le limiteur utilisé dans le cas scalaire. On note $\boldsymbol{\beta}$ le vecteur formé des coefficients de distribution du premier ordre $(\beta_i)_{i=1,n_{\mathrm{DoFs}}}$ et $\boldsymbol{\beta^*}$ le vecteur formé des coefficients de distribution $(\beta_i^*)_{i=1,n_{\mathrm{DoFs}}}$ d'ordre $(k+1)$. On considère ces deux vecteurs comme les coordonnées barycentriques de deux points de l'espace dans un polygone régulier à n_{DoFs} sommets. Nous cherchons des β_i^* bornés, on choisit des β_i^* tels que $0 \leq \beta_i^* \leq 1$. Ceci signifie, en terme de coordonnées barycentriques, que le point de coordonnées $\boldsymbol{\beta^*}$ est dans le polygone. Il peut être sur l'une des arêtes du polygone (une des coordonnées du vecteur est nulle) ou sur l'un des sommets (deux des coordonnées sont nulles). Si les n_{DoFs} coordonnées β_i de $\boldsymbol{\beta}$ sont positives, il suffit de prendre $\boldsymbol{\beta^*} = \boldsymbol{\beta}$. Si $\boldsymbol{\beta}$ est à l'extérieur du polygone, on définit $\boldsymbol{\beta^*}$ comme la projetée par l'application de projection I.2.9 de $\boldsymbol{\beta}$ sur la frontière la plus proche du polygone.

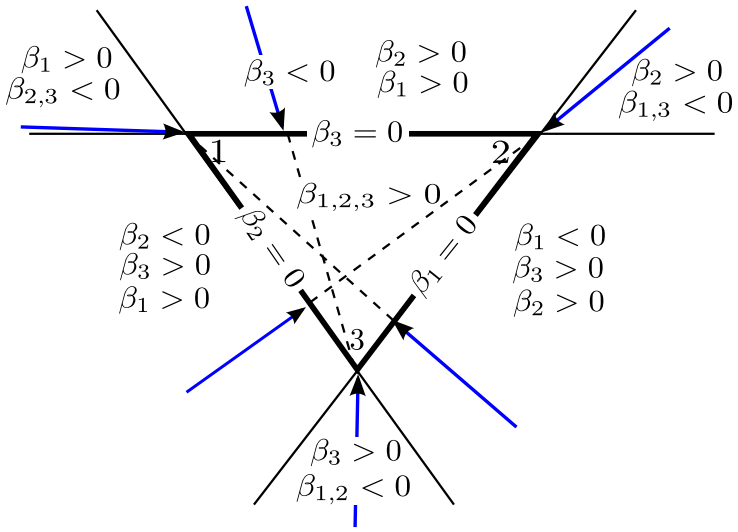


Figure I.2.5 – Limitation d'un cas scalaire d'ordre 2 :
les flèches représentent la projection de $\boldsymbol{\beta}$ sur les bords de l'élément.

Définition I.2.9 (*Projecteur PSI*)

On appelle projecteur *PSI* l'application :

$$\beta_i^* = \frac{\beta_i^+}{\sum_{i=1}^{n_{\text{DoFs}}} \beta_i^+}, \tag{I.48}$$

où $\beta_i^+ = \max(0, \beta_i)$.

Le schéma limité d'ordre $(k+1)$ a bien toutes les propriétés décrites paragraphes 2.3.3 et 2.3.4 (il est conservatif et il conserve la monotonicité et la linéarité). Il nous faut maintenant étendre ce schéma au cas vectoriel. Notons $\bar{G} = \left(\sum_{i=1}^{n_{\text{DoFs}}} G_i\right) / n_{\text{DoFs}}$ l'état moyen d'une variable G sur un élément. Soit $\vec{\boldsymbol{u}}$ le vecteur vitesse de l'écoulement, on définit par $\vec{\boldsymbol{n}}_{\vec{\boldsymbol{u}}} = \frac{\vec{\boldsymbol{u}}}{\|\vec{\boldsymbol{u}}\|}$ la direction de cet écoulement dans un élément. Soit $\mathcal{A}_v = \vec{\boldsymbol{\lambda}}\left(\bar{\boldsymbol{U}}\right).\vec{\boldsymbol{n}}_{\vec{\boldsymbol{u}}}$, $\mathcal{A}_v$ est la matrice jacobienne du flux évalué à l'état moyen. $\mathcal{A}_v$ est diagonalisable, ses valeurs propres représentent les vitesses d'onde de l'écoulement et ses vecteurs propres à gauche les équations caractéristiques associées. On note $\mathcal{L}_i(\Phi_j)$ la projection sur ces vecteurs propres du résidu local de Lax-Friedrichs au nœud j et on définit les coefficients de distribution β_{ij} du premier ordre par :

$$\beta_{ij} = \frac{\mathcal{L}_i(\Phi_j)}{\mathcal{L}_i(\Phi)}. \tag{I.49}$$

On peut alors utiliser le projecteur I.2.9 définit dans le cas scalaire :

$$\beta_{ij}^{*} = \frac{\beta_{ij}^{+}}{\sum_{l=1}^{n_{\text{DoFs}}} \beta_{lj}^{+}}. \tag{I.50}$$

Les β_{ij}^{*} permettent calculer les résidus limités projetés dans la base des vecteurs propres de la matrice $\mathcal{A}_v$. On revient à la base de départ par projection sur la base des vecteurs propres à droite de $\mathcal{A}_v$. Pour résumer, la procédure de limitation (dans le cas scalaire comme vectoriel) se fait en quatre étapes :

- on calcule le résidu de Lax-Friedrich du premier ordre Φ_i^{T}.
- Cela permet le calcul des coefficients de distribution du premier ordre (qui forment un vecteur $\boldsymbol{\beta}^T = \left(\beta_i^T\right)_{(i=1,n_{\text{DoFs}})}$ dans le cas scalaire et une matrice $\boldsymbol{B}^T = \left(B_i^T\right)_{(i=1,n_{\text{DoFs}})}$ dans le cas vectoriel).
- On trouve les coefficients de distribution d'ordre $(k+1)$ $\boldsymbol{\beta}^{T,*}$ ou $\boldsymbol{B}^{T,*}$ par projection `PSI` des coefficients de distribution du premier ordre.
- On calcule le le résidu de Lax-Friedrich limité d'ordre $(k+1)$ $\Phi_i^{T,*}$ tel que :

$$\Phi_i^{T,*} = \beta_i^{\mathrm{T}} \Phi^{\mathrm{T}} \quad \text{dans le cas scalaire} \tag{I.51}$$

$$\Phi_i^{T,*} = B_i^{\mathrm{T}} \Phi^{\mathrm{T}} \quad \text{dans le cas vectoriel} \tag{I.52}$$

Bien sûr, comme nous cherchons à résoudre les équations d'Euler, nous utiliserons la formulation vectorielle de cette limitation.

2.4.3 Stabilisation

Il reste encore un inconvénient majeur au schéma LLxF, en pratique, il ne converge pas complètement car des modes parasites apparaissent dans la partie régulière de l'écoulement. Évidemment, ce phénomène a une cause mathématique, notre problème limité est en fait mal posé. En effet, comme nous l'avons déjà fait remarquer, le schéma LLxF est totalement centré et la distribution du signal aux nœuds ne dépend pas de la physique du problème mais seulement du gradient de la solution sur l'élément. Ainsi, un nœud M_i peut ne recevoir aucune information si pour tout T appartenant à $\mathcal{D}_i$ le coefficient de distribution $B_i^{T,*}$ est nul. Ce phénomène est illustré figure I.2.6. Comme on cherche à résoudre $\sum_{\mathrm{T}\in\mathcal{D}_i} \Phi_i^{T,*}(\boldsymbol{U}_h) = \sum_{\mathrm{T}\in\mathcal{D}_i} B_i^{T,*}\Phi^{\mathrm{T}}(\boldsymbol{U}_h) = 0$ et que tout les $B_i^{T,*}$ sont nuls, cela re-

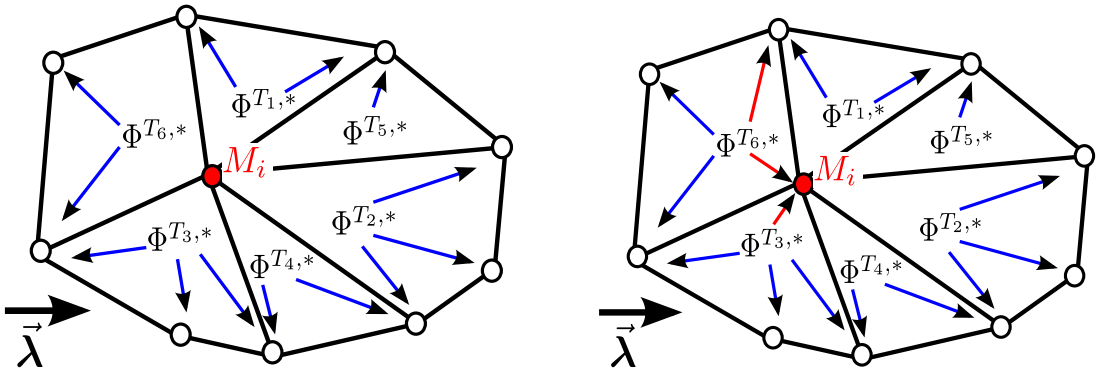


Figure I.2.6 – À gauche : Exemple de mauvaise distribution des signaux du schéma centré LLxF, le nœud M_i ne reçoit aucune information,
À droite : Décentrement amont du schéma par l'ajout d'un terme de dissipation numérique dans le sens de l'advection.

vient à résoudre $\sum_{T\in\mathcal{D}_i} 0 = 0$, d'où l'apparition de solutions parasites au niveau de certains nœuds. La mauvaise distribution des résidus limités aux nœuds étant une conséquence du caractère centré du schéma, une solution naturelle est de décentrer le schéma et de distribuer la fluctuation totale d'un élément dans la direction de l'advection. Quelque soit le DoF M_i intérieur au domaine, il est forcément en aval de l'écoulement pour au moins un élément de $\mathcal{D}_i$. On s'assure ainsi que tous les nœuds reçoivent un signal. On ajoute un terme de type SUPG (*Streamline Upwind Petrov Galerkin*), donc, un terme dissipation numérique dans le sens de l'écoulement, pour décentrer notre schéma. Ce terme, D_i^T, est appelé terme de stabilisation. En notant φ_i la $i^{\text{ème}}$ fonction de base, D_i^{T} est de la forme :

$$D_i^{\mathrm{T}} = \int_{\mathrm{T}} (\vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla\varphi_i})\bar{\boldsymbol{\tau}}(\vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla \boldsymbol{U}_h})\, d\boldsymbol{x} \tag{I.53}$$

$\bar{\boldsymbol{\tau}}$ étant un tenseur d'adimensionalisation. En pratique,

$$\bar{\boldsymbol{\tau}} = \left(\sum_{i\in\mathrm{T}} \bar{k}_i^-\right)^{-1},$$

où $\bar{k}_i^- = \min(\bar{k}_i, 0)$ et les coefficients $\bar{k}_i$ sont définis par l'équation (I.45) tels que :

$$\bar{k}_i = \int_{\mathrm{T}} \vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla\varphi_i^k} d\boldsymbol{x}$$

Pour finir, on ne veut décentrer le schéma que dans les zones où l'écoulement est régulier donc on multiplie D_i par un coefficient θ_{T} tel que θ_{T} vaut 1 si la solution est régulière et h sinon (ainsi, quand h tend vers 0 on tend à ne plus stabiliser le schéma dans les zones de chocs). On choisit de prendre θ_{T} tel que :

$$\theta_{\mathrm{T}} = \max_{i\in \mathrm{T}} \left(\max_{\mathrm{T}\in\mathcal{D}_i} \max_{j\in\mathrm{T}} \frac{\left| \frac{p_j}{\rho_j^\gamma} - \frac{\bar{p}}{\bar{\rho}^\gamma} \right|}{\left| \frac{p_j}{\rho_j^\gamma} \right| + \left| \frac{\bar{p}}{\bar{\rho}^\gamma} \right| + \varepsilon} \right),$$

où ε est un terme proche du zéro machine. En pratique, θ_{T} est très souvent égal à 1.
Nous conclurons ce paragraphe par la forme finale du résidu limité et stabilisé de Lax-Friedrichs :

$$\Phi_i^{T,*} = B_i^{\mathrm{T}} \Phi^{\mathrm{T}} + \theta_{\mathrm{T}} D_i^{\mathrm{T}}. \tag{I.54}$$

2.5 Discrétisation temporelle

Intéressons nous maintenant à la discrétisation en temps du problème :

$$\begin{cases} \forall \mathrm{M}_i,\ |C_i| \dfrac{\boldsymbol{U}_i^{n+1} - \boldsymbol{U}_i^n}{\Delta t} + \displaystyle\sum_{\mathrm{T}\in\mathcal{D}_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h) = 0. \\ \boldsymbol{U}_i^0 \text{ donné} \end{cases} \tag{I.55}$$

On note $\Delta \boldsymbol{U}_i^n = \boldsymbol{U}_i^{n+1} - \boldsymbol{U}_i^n$ et $\omega_i = \frac{\Delta t}{|C_i|}$, $\mathbf{I}$ la matrice identité (de taille $(d+2)$ pour un problème d-dimensionnel), l'équation (I.55) peut être réécrite :

$$\begin{cases} \forall \mathrm{M}_i,\ \dfrac{\mathbf{I}}{\omega_i} \Delta \boldsymbol{U}_i^n = - \displaystyle\sum_{\mathrm{T}\in D_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^\chi) \\ \boldsymbol{U}_i^0 \text{ donné} \end{cases} \tag{I.56}$$

Dans le cas d'un schéma explicite, χ, le paramètre représentant le temps auquel est calculé Φ_i^{T} est égal à n :

$$\begin{cases} \forall \mathrm{M}_i,\ \dfrac{\mathbf{I}}{\omega_i} \Delta \boldsymbol{U}_i^n = - \displaystyle\sum_{\mathrm{T}\in D_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^n) \\ \boldsymbol{U}_i^0 \text{ donné} \end{cases} \tag{I.57}$$

Dans notre cas, du fait de sa stabilité inconditionnelle (voir propriété I.2.2 et la sous-section 5.2.1 de la thèse d'A. Larat [36] pour la preuve), on choisit d'utiliser un schéma

implicite, on a donc $\chi = n+1$ et :

$$\begin{cases} \forall \mathrm{M}_i,\ \dfrac{\mathbf{I}}{\omega_i}\Delta \boldsymbol{U}_i^n = -\displaystyle\sum_{\mathrm{T}\in D_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^{n+1}) \\ \boldsymbol{U}_i^0 \text{ donné} \end{cases} \tag{I.58}$$

Propriété I.2.2 (*Cas scalaire : Stabilité inconditionnelle*)

Quelque soit le pas de temps Δt, le schéma implicite (I.58) vérifie le principe du maximum global discret, pour chaque variable indépendamment.

$$\forall l \in [\![1,m]\!], \forall M_i \in \mathcal{M}_h, \quad \min_{M_j\in\mathcal{M}_h} \boldsymbol{U}_{j,l}^{n+1} \leq \boldsymbol{U}_{i,l}^n \leq \max_{M_j\in\mathcal{M}_h} \boldsymbol{U}_{j,l}^n. \tag{I.59}$$

Malheureusement, bien qu'en pratique le principe du maximum global discret semble également vérifié dans le cas vectoriel, il n'existe à ce jour pas de preuve l'attestant.

On ne connait pas le résidu au temps t^{n+1}, donc, pour l'évaluer, on le linéarise par un développement de Taylor :

$$\Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^{n+1}) \approx \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^n) + \sum_{j\in\mathcal{M}_h} \frac{\partial \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^n)}{\partial \boldsymbol{U}_j}\left(\boldsymbol{U}_j^{n+1} - \boldsymbol{U}_j^n\right).$$

l'équation (I.58) devient :

$$\left(\frac{\mathbf{I}}{\omega_i} + \sum_{\mathrm{T}\in\mathcal{D}_i} \frac{\partial \Phi_i^T(\boldsymbol{U}_h^n)}{\partial \boldsymbol{U}_i}\right)\Delta \boldsymbol{U}_i^n + \sum_{\substack{j\in\mathcal{D}_i \\ j\neq i}} \left(\sum_{\mathrm{T}\in\mathcal{D}_i\cap\mathcal{D}_j} \frac{\partial \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^n)}{\partial \boldsymbol{U}_j}\right)\Delta \boldsymbol{U}_j^n = -\sum_{\mathrm{T}\in\mathcal{D}_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^n), \tag{I.60}$$

ou dans une forme matricielle plus compacte :

$$\mathbf{M}\Delta \boldsymbol{U}^n = -\boldsymbol{\Phi}^n, \tag{I.61}$$

où $\Delta \boldsymbol{U}^n = (\Delta \boldsymbol{U}_i^n)_{i=1,N_{Tot}}$, $\boldsymbol{\Phi}^n = (\sum_{\mathrm{T}\in\mathcal{D}_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^n))_{i=1,N_{Tot}}$ et $\mathbf{M}$ une matrice par bloc telle que $\mathbf{M} = (\mathbf{M}_{i,j})_{i,j=1,N_{Tot}}$. Pour $\boldsymbol{U}_i$ de taille n_{var}, chaque bloc $\mathbf{M}_{i,j}$ est de taille $(n_{\text{var}} \times n_{\text{var}})$ et :

$$\begin{cases} \mathbf{M}_{i,i} = \frac{\mathbf{I}}{\omega_i} + \sum_{\mathrm{T}\in\mathcal{D}_i} \dfrac{\partial \Phi_i^T(\boldsymbol{U}_h^n)}{\partial \boldsymbol{U}_i} \\ \mathbf{M}_{i,j} = \sum_{\mathrm{T}\in\mathcal{D}_i\cap\mathcal{D}_j} \dfrac{\partial \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^n)}{\partial \boldsymbol{U}_j} \quad \text{for } i \neq j \end{cases} \tag{I.62}$$

Les signaux explicites du membre de droite sont les résidus limités et stabilisés de Lax-Friedrichs, par contre, on utilise les résidus non-limités et non-stabilisés pour évaluer la jacobienne des signaux dans la matrice implicite $\mathbf{M}$. Les matrices jacobiennes sont approchées par les jacobiennes du signal d'ordre 1 aux DoFs.

La section suivante vise à présenter en quelques lignes l'extension du précédent schéma aux équations de Navier-Stokes. Pour plus d'informations sur la construction et les propriétés du schéma aux résidus distribués associé aux équations de Navier-Stokes, nous invitons le lecteur à consulter le document [48].

3 Schéma numérique pour la résolution des équations de Navier-Stokes

Les méthodes de discrétisation numérique et temporelle sont identiques à celles présentées dans la précédente section. Nous nous intéresserons donc uniquement à la construction du schéma aux résidus distribués. On conserve les notations précédemment employées.

3.1 Formulation du problème

Rappelons le système d'équations de Navier-Stokes :

$$\left\{\begin{array}{lcl} \dfrac{\partial \rho}{\partial t} + \overrightarrow{\nabla}.(\rho\vec{\boldsymbol{u}}) & = & 0, \\ \dfrac{\partial \rho\vec{\boldsymbol{u}}}{\partial t} + \overrightarrow{\nabla}.(\rho\vec{\boldsymbol{u}}\otimes\vec{\boldsymbol{u}} + p\mathbf{I}) & = & \overrightarrow{\nabla}.\boldsymbol{\sigma}, \\ \dfrac{\partial \rho E}{\partial t} + \overrightarrow{\nabla}.((\rho E + p)\vec{\boldsymbol{u}}) & = & \overrightarrow{\nabla}.(\boldsymbol{\sigma}\vec{\boldsymbol{u}}) - \overrightarrow{\nabla}.\vec{\boldsymbol{q}}. \end{array}\right. \tag{I.63}$$

Dans le cas d'un problème stationnaire, on souhaite donc résoudre le système :

$$\left\{\begin{array}{rcl} \overrightarrow{\nabla}.(\rho\vec{\boldsymbol{u}}) & = & 0, \\ \overrightarrow{\nabla}.(\rho\vec{\boldsymbol{u}}\otimes\vec{\boldsymbol{u}} + p\mathbf{I}) - \overrightarrow{\nabla}.\boldsymbol{\sigma} & = & \mathbf{0}, \\ \overrightarrow{\nabla}.((\rho E + p)\vec{\boldsymbol{u}}) - \overrightarrow{\nabla}.(\boldsymbol{\sigma}\vec{\boldsymbol{u}} - \vec{\boldsymbol{q}}) & = & 0. \end{array}\right. \tag{I.64}$$

Comme expliqué sous-section 1.3, le tenseur des contraintes $\boldsymbol{\sigma}$ et le flux de chaleur massique $\vec{\boldsymbol{q}}$ dépendent des coefficients de viscosité λ et μ. Comme λ et μ dépendent de la température T, $\boldsymbol{\sigma}$ et $\vec{\boldsymbol{q}}$ sont fonctions de la solution $\boldsymbol{U}$.

Par ailleurs, $\boldsymbol{\sigma}$ est une fonction du tenseur des déformations et $\vec{\boldsymbol{q}}$ est une fonction du gradient de température. Ces deux termes sont donc également des fonctions du gradient de la solution $\overrightarrow{\nabla \boldsymbol{U}}$.

Notons $\overrightarrow{\boldsymbol{\mathcal{F}}}_e(\boldsymbol{U})$ et $\overrightarrow{\boldsymbol{\mathcal{F}}}_v(\boldsymbol{U}, \overrightarrow{\nabla \boldsymbol{U}})$ les flux eulérien et visqueux tels que :

$$\overrightarrow{\boldsymbol{\mathcal{F}}}_e(\boldsymbol{U}) = \begin{pmatrix} \rho \vec{\boldsymbol{u}} \\ \rho \vec{\boldsymbol{u}} \otimes \vec{\boldsymbol{u}} + p\mathbf{I} \\ (\rho E + p)\, \vec{\boldsymbol{u}} \end{pmatrix}, \quad \overrightarrow{\boldsymbol{\mathcal{F}}}_v(\boldsymbol{U}, \overrightarrow{\nabla \boldsymbol{U}}) = \begin{pmatrix} 0 \\ \boldsymbol{\sigma} \\ \boldsymbol{\sigma} \vec{\boldsymbol{u}} - \vec{\boldsymbol{q}} \end{pmatrix}. \tag{I.65}$$

Le système I.64 peut se réécrire sous la forme :

$$\overrightarrow{\nabla}. \left(\overrightarrow{\boldsymbol{\mathcal{F}}}_e(\boldsymbol{U}) - \overrightarrow{\boldsymbol{\mathcal{F}}}_v(\boldsymbol{U}, \overrightarrow{\nabla \boldsymbol{U}}) \right) = 0. \tag{I.66}$$

3.2 Écriture du schéma $\mathcal{RD}$ et problématique

On intègre l'équation sur Ω et on reformule l'intégrale de la divergence du flux sur un élément comme une intégrale de contour à l'aide de la formule de Gauss :

$$\sum_{T \in \Omega} \int_{\partial \mathrm{T}} \left(\overrightarrow{\boldsymbol{\mathcal{F}}}_e(\boldsymbol{U}) - \overrightarrow{\boldsymbol{\mathcal{F}}}_v(\boldsymbol{U}, \overrightarrow{\nabla \boldsymbol{U}}) \right) d\partial \mathrm{T} = 0. \tag{I.67}$$

On pose :

$$\sum_{i \in T} \Phi_i^{\mathrm{T}}(\boldsymbol{U}, \overrightarrow{\nabla \boldsymbol{U}}) = \Phi^{\mathrm{T}}(\boldsymbol{U}, \overrightarrow{\nabla \boldsymbol{U}}) = \sum_{T \in \Omega} \int_{\partial \mathrm{T}} \left(\overrightarrow{\boldsymbol{\mathcal{F}}}_e(\boldsymbol{U}) - \overrightarrow{\boldsymbol{\mathcal{F}}}_v(\boldsymbol{U}, \overrightarrow{\nabla \boldsymbol{U}}) \right) d\partial \mathrm{T}. \tag{I.68}$$

Le schéma aux résidus distribués visqueux consiste alors à trouver une solution approchée $\boldsymbol{U}_h$ telle que :

$$\begin{cases} \forall \mathrm{M}_i, \sum\limits_{\mathrm{T} \in \mathcal{D}_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h, \overrightarrow{\nabla \boldsymbol{U}_h}) = 0. \\ \boldsymbol{U}_i^0 \text{ donné} \end{cases} \tag{I.69}$$

Comme dans le cas des équations d'Euler, le schéma est ensuite limité, stabilisé puis résolu par un schéma pseudo-itératif en temps.

Nous avons fourni paragraphe 2.3.1 un théorème de consistance des schémas $\mathcal{RD}$, le théorème I.2.1. Ce théorème stipule qu'un schéma $\mathcal{RD}$ converge vers la solution faible du problème étudié si la solution approchée $\boldsymbol{U}_h$ est continue. Dans le cas d'un modèle visqueux, le gradient de la solution approchée $\overrightarrow{\nabla \boldsymbol{U}_h}$ doit également être continu. Cette

seconde condition n'est *a priori* pas vérifiée : le terme $\overrightarrow{\nabla \boldsymbol{U}_h}$ est discontinu à travers les arêtes du maillage.

3.3 Solution envisagée et calcul du résidu

Notons $\vec{e}$ les arêtes du maillage et E l'ensemble de ces arêtes.

L'idée est de construire un gradient de solution moyen, noté $\overrightarrow{\overline{\nabla \boldsymbol{U}_h}}$, tel que :

- $\forall \vec{e} \in E$, le terme moyenné $\overrightarrow{\overline{\nabla \boldsymbol{U}_h}}$ est continu au travers de $\vec{e}$.
- si $\overrightarrow{\nabla \boldsymbol{U}_h}$ est continu, $\overrightarrow{\overline{\nabla \boldsymbol{U}_h}} = \overrightarrow{\nabla \boldsymbol{U}_h}$.

Il nous reste donc à trouver une définition de $\overrightarrow{\overline{\nabla \boldsymbol{U}_h}}$ satisfaisant les deux conditions précédentes.

Notons $\overrightarrow{\nabla \boldsymbol{U}_{h|T}}$ la restriction du gradient de $\boldsymbol{U}$ à l'élément T et φ_i^T la $i^{\text{ème}}$ fonction de base associée à l'élément T. On calcule $\overrightarrow{\nabla \boldsymbol{U}_{h|T}}$ par la formule I.70.

$$\overrightarrow{\nabla \boldsymbol{U}_{h|T}} = \sum_{i=1}^{n_{\text{DoFs}}} \boldsymbol{U}_i \overrightarrow{\nabla \varphi_i^T} \tag{I.70}$$

Définition I.3.1 (*Gradient moyen de la solution approchée*)

Soit M_i un degré de liberté de voisinage $\mathcal{D}_i$ et n le nombre d'éléments de $\mathcal{D}_i$. On désigne par T_j le $j^{\text{ème}}$ élément de $\mathcal{D}_i$.

Au nœud M_i, le gradient moyen de la solution approchée est défini par :

$$\overrightarrow{\overline{\nabla \boldsymbol{U}_h}}(M_i) = \frac{1}{n} \sum_{T_j \in \mathcal{D}_i} \overrightarrow{\nabla \boldsymbol{U}_{h|T_j}}. \tag{I.71}$$

Intéressons nous maintenant au calcul du résidu.

Du fait de la présence d'un flux visqueux additionnel dans l'équation, le terme de stabilisation précédemment présenté (I.53) est légèrement modifié :

$$D_i^{\mathrm{T}} = \int_{\mathrm{T}} (\vec{\boldsymbol{\lambda}}_e . \overrightarrow{\nabla \varphi_i} - \vec{\boldsymbol{\lambda}}_v . \overrightarrow{\nabla \varphi_i}) \bar{\boldsymbol{\tau}} (\vec{\boldsymbol{\lambda}}_e . \overrightarrow{\nabla \boldsymbol{U}_h} - \vec{\boldsymbol{\lambda}}_v . \overrightarrow{\nabla \boldsymbol{U}_h}) \, d\boldsymbol{x}, \tag{I.72}$$

où $\vec{\boldsymbol{\lambda}}_e$ désigne le vecteur des matrices jacobiennes associées au flux eulérien et $\vec{\boldsymbol{\lambda}}_v$ celui des matrices jacobiennes associées au flux visqueux.

Les autres étapes de calcul demeurent inchangées.

Ceci conclut la mise en place du contexte théorique dans lequel s'inscrit notre travail. Le chapitre suivant est consacré à l'étude du couplage du schéma $\mathcal{RD}$ précédemment décrit et de l'analyse isogéométrique.

Chapitre II

Analyse isogéométrique

Nous allons maintenant nous intéresser à l'analyse isogéométrique et à son intégration dans un schéma $\mathcal{RD}$.

Comme nous l'avons introduit au début de cette thèse, l'analyse isogéométrique se fonde sur une représentation des frontières du même ordre que l'ordre de discrétisation des variables. L'utilisation d'une unique famille de fonctions de base pour décrire à la fois la géométrie et les solutions du problème considéré constitue donc l'une des principales caractéristiques de cette méthode.

Nous débuterons ce chapitre par la mise en place de quelques notations communément utilisées en analyse isogéométrique. Nous détaillerons ensuite la construction d'une famille de fonctions de base adaptée à la représentation exacte des géométries courbes. Nous conclurons par l'implémentation de ces fonctions de base dans le schéma $\mathcal{RD}$ limité et stabilisé de Lax-Friedrichs.

1 Notations et terminologie

Nous dédirons la première partie de cette section à la définition de deux notions cruciales de l'analyse isogéométrique : la notion de *degrés de liberté* et celle de *points de contrôles*. Dans la seconde partie, nous nous intéresserons au passage de l'approximation globale d'une variable sur un maillage à son approximation locale sur un élément de référence.

1.1 Degrés de liberté et points de contrôle

La solution approchée d'un problème, $\boldsymbol{U}_h$, est généralement construite par interpolation de la solution discrète $(\boldsymbol{U}_i)_{i=1,n_{\text{DoFs}}}$ sur un ensemble de fonctions de base polynomiales. Les fonctions de bases sont alors dites *interpolantes*. L'analyse isogéométrique, à

la différence des méthodes classiques, se fonde principalement sur l'utilisation de fonctions de base *extrapolantes*. Notons Ψ_i ces fonctions de base. Ceci signifie qu'une courbe :

$$\boldsymbol{X}_h(t) = \sum_i \tilde{\boldsymbol{X}}_i \Psi_i(t)$$

ne passe pas nécessairement par les valeurs $\tilde{\boldsymbol{X}}_i$. Les $\tilde{\boldsymbol{X}}_i$ sont appelés *points de contrôle* (notés $\mathcal{CP}s$ pour *Control Points*) et sont dits *extrapolés* sur les fonctions de base.

Nous devrons donc distinguer les valeurs « physiques » des variables aux degrés de liberté et les points de contrôle qui, extrapolés sur nos fonctions de base, nous permettrons de trouver une courbe passant par ces valeurs physiques. Nous noterons simplement $\boldsymbol{U}_j = \boldsymbol{U}_h(\mathrm{M}_j)$ les valeurs d'une variable $\boldsymbol{U}$ au degré de liberté M_j et $\tilde{\boldsymbol{U}}$ les valeurs de contrôle permettant d'approcher $\boldsymbol{U}$ par $\boldsymbol{U}_h(\mathrm{M}_j) = \sum_i \tilde{\boldsymbol{U}}_i \Psi_i(\mathrm{M}_j)$. Notons N_{ba} le nombre de fonctions de base du maillage considéré. Comme les $\mathcal{CP}s$ sont extrapolés sur ces fonctions, leur nombre sera égal à N_{ba}. Il est important de noter que le nombre de degrés de liberté du maillage, N_{Tot}, n'est pas nécessairement égal à N_{ba}. La figure II.1.2 donne un exemple de représentation d'une courbe à l'aide de fonctions extrapolantes.

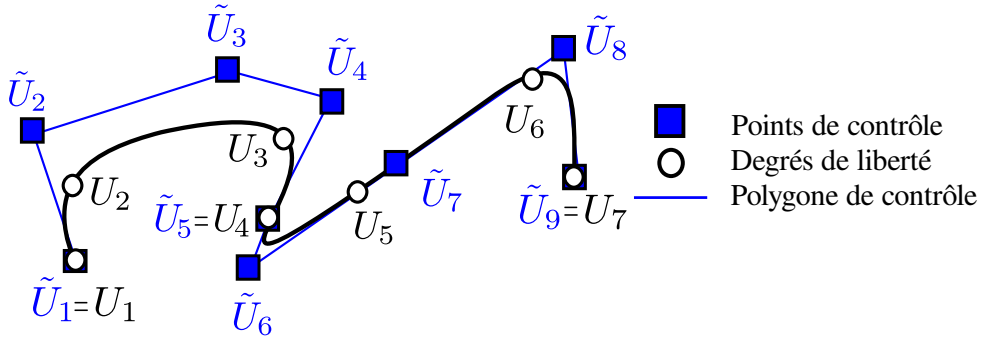


Figure II.1.2 – Degrés de liberté et points de contrôle.

Nous nommerons *polygone de contrôle* (ou *grille de contrôle*) le polygone formé par les points de contrôle (polygone représenté en bleu figure II.1.2).

La base de notre espace discret sera constituée de fonctions rationnelles réelles (*c.-à-d.* des fonctions s'écrivant comme fraction de deux polynômes à coefficients réels). Notons $\mathbb{Q}_{\mathbb{P}}^k$ l'espace des fonctions rationnelles réelles formées de polynômes de degré k et ne contenant pas de pôles dans $\mathbb{R}^d$ (*c.-à-d.* que les fonctions de $\mathbb{Q}_{\mathbb{P}}^k$ sont des fonctions dont le dénominateur est non nul). Notre solution approchée $\boldsymbol{U}_h(\boldsymbol{X}, t)$ sera donc finalement une

extrapolation de valeurs de contrôles $\tilde{\boldsymbol{U}}_i$ sur un ensemble de fonctions de base de $\mathbb{Q}^k_{\mathbb{P}}$. Les valeurs de contrôle doivent être convenablement choisies afin de permettre l'interpolation par $\boldsymbol{U}_h$ de la solution discrète $\boldsymbol{U}_i$.

Comme en analyse isogéométrique la géométrie du problème est considérée comme une variable, les points de la discrétisation $\mathcal{T}_h$ seront obtenus par l'extrapolation de points de contrôle sur les fonctions de base. Nous avons noté $(\mathrm{M}_j)_{j=1,N_{Tot}}$ nos degrés de liberté, nous noterons $(\tilde{\mathrm{M}}_l)_{l=1,N_{\mathrm{ba}}}$ les valeurs de contrôle permettant d'interpoler les degrés de liberté de cette géométrie.

Pour résumer, dans ce travail, une valeur « tildée » désignera toujours une valeur de contrôle et une valeur notée sans tilde désignera une valeur physique. L'approximation d'une variable quelle qu'elle soit sera indicée par h et obtenue par l'extrapolation des points de contrôle sur les fonctions de base :

$$\cdot_h(t) = \sum_{l=1}^{N_{\mathrm{ba}}} \tilde{\cdot}_l \Psi_l(t). \tag{II.1}$$

1.2 Espaces de travail

Cette sous-section vise à décrire le passage de l'approximation globale sur un maillage constitué d'éléments courbes à l'approximation locale sur une élément de référence droit. Dans une première étape nous détaillerons la restriction de l'approximation globale à un unique élément courbe. Dans une seconde étape, nous verrons le passage de cet élément courbe à l'élément de référence.

1.2.1 De l'approximation globale à l'approximation locale sur un élément courbe T

Supposons maintenant que $N_{\mathrm{ba}} = N_{Tot}$. Quelque soit le DoF M_i, la $i^{\text{ème}}$ fonction de base, Ψ_i, est alors à support compact sur le voisinage de M_i, $\mathcal{D}_i$. Si un élément T de $\mathcal{T}_h$ compte n_{DoFs} degrés de liberté, nous aurons n_{DoFs} fonctions de base Ψ non nulles sur T.

On considère une numérotation locale des DoFs sur T. Supposons que M_i est le $j^{\text{ème}}$ DoF de T. On note $(\varphi^{\mathrm{T}}_j)_{j=1,n_{\mathrm{DoFs}}}$ la restriction sur T de Ψ_i. Ainsi, $\Psi_i = \sum_{\mathrm{T}\in\mathcal{D}_i} \varphi^{\mathrm{T}}_j$ et :

$$\boldsymbol{U}_h = \sum_{i=1}^{N_{Tot}} \tilde{\boldsymbol{U}}_i \Psi_i = \sum_{T\in\mathcal{T}_h} \left(\sum_{j=1}^{n_{\mathrm{DoFs}}} \tilde{\boldsymbol{U}}_j \varphi^T_j \right). \tag{II.2}$$

Ainsi, plutôt que de travailler sur $\mathcal{V}_h$, l'espace vectoriel approchant $\mathcal{T}_h$, nous travaillerons sur les sous-espaces vectoriels approchant chaque élément T de $\mathcal{T}_h$. On note $\mathcal{V}_h^{k,T}$ le sous-espace associé à l'élément T pour une discrétisation $\mathbb{Q}_{\mathbb{P}}^k$. $\boldsymbol{U}_h$, la solution approchée complète sur $\mathcal{V}_h$, sera construite à partir des solutions locales $\boldsymbol{U}_h^T$. Une fois de plus, si il n'y a pas d'ambiguïté, nous simplifierons les notations en retirant l'indice T. Le terme d'extrapolation locale, $\boldsymbol{U}_h^T$, sera donc noté $\boldsymbol{U}_h$ lorsque nous travaillerons sur un unique élément. De même, les fonctions de base $\left(\varphi_j^T\right)_{j=1,n_{\text{DoFs}}}$ seront simplement notées $(\varphi_j)_{j=1,n_{\text{DoFs}}}$.

Définition II.1.1 (*Extrapolation locale*)

On appelle *extrapolation locale* le terme $\boldsymbol{U}_h$, obtenu par extrapolation des valeurs de contrôle $\tilde{\boldsymbol{U}}$ associées à l'élément T sur les fonctions de base de T :

$$\boldsymbol{U}_h = \sum_{j=1}^{n_{\text{DoFs}}} \tilde{\boldsymbol{U}}_j \varphi_j. \tag{II.3}$$

1.2.2 D'un élément courbe T de $\mathbb{R}^d$ à sa forme élémentaire dans l'espace paramétrique

k désigne maintenant notre ordre de travail (*c.-à-d.* que les fonctions de base utilisées appartiennent à $\mathbb{P}^{k-1}$ ou $\mathbb{Q}_{\mathbb{P}}^{k-1}$). On travaille dans l'espace réel de dimension d. Notons đ la dimension de l'espace paramétrique et $\boldsymbol{\xi} = (\xi)$, $\boldsymbol{\xi} = (\xi, \eta)$ et $\boldsymbol{\xi} = (\xi, \eta, \zeta)$ les coordonnées dans les espaces paramétriques de dimensions respectives đ = 1, đ = 2 et đ = 3.

Remarque II.1.1 (*dimensions des espaces paramétriques et réels*)

đ, la dimension de l'espace paramétrique n'est pas nécessairement égale à d, la dimension de l'espace réel.

Exemple II.1.1 *(đ n'est pas nécessairement égal à d)* : La figure II.1.3 montre qu'une courbe de $\mathbb{R}^3$ est l'image d'un segment de l'espace paramétrique. Ainsi đ = 1 et $d = 3$.

Par soucis de simplification, nous désignerons simplement par $\boldsymbol{\xi}$ les coordonnées dans l'espace paramétrique, sans préciser la valeur de đ. Implicitement, đ = 1 si l'élément de l'espace réel (quelque soit la dimension de cet espace) est une courbe, đ = 2 si cet élément est une surface (non nécessairement plane) et đ = 3 si c'est un volume.

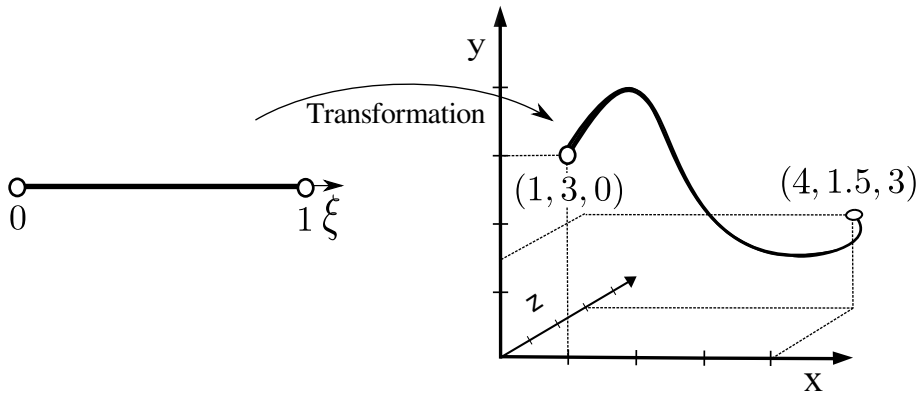


Figure II.1.3 – Mapping d'un segment monodimensionnel de l'espace paramétrique sur une courbe de $\mathbb{R}^3$

Considérons un élément T de $\mathbb{R}^d$. Quelque soit T, il existe une forme élémentaire unitaire [1], notée E, de l'espace paramétrique de dimension đ et une transformation C définie de l'espace paramétrique dans $\mathbb{R}^d$ telle que $C(E) = \mathrm{T}$. Des exemples de formes élémentaires unitaires de l'espace paramétrique sont fournis figure II.1.4.

Soit $(\Upsilon_i)_{i=1,n_{\mathrm{DoFs}}}$ un ensemble de fonctions de base sur l'élément paramétrique E. Soit $P_E = (\boldsymbol{\xi}_E)$ un point de E tel que $C(\boldsymbol{\xi}_E) = \boldsymbol{X}_E$ est un point de l'élément T. On note C^{-1} l'application réciproque de C (*c.-à-d.* la transformation définie sur $\mathbb{R}^d$ dans l'espace paramétrique et qui associe E à T).
$\Upsilon_i(\boldsymbol{\xi}_E) = \Upsilon_i(C^{-1}\boldsymbol{X}_E)$, ainsi, en posant $\varphi_i = \Upsilon_i \circ C^{-1}$, on construit une base de l'élément T.

Remarque II.1.2

La transformation C n'est autre qu'une extrapolation, sur les fonctions de base φ_i choisies, des points de contrôle $\tilde{M}$ associés à l'élément T. Par anglicisme nous parlerons souvent de *mapping* plutôt que de *transformation*.

En pratique, nous ne distinguerons pas les fonctions de base Υ_i des fonctions de base φ_i et, que ce soit dans l'espace réel ou dans l'espace paramétrique, nous noterons φ_i la $i^{\text{ème}}$ fonction de base de notre élément. φ_i sera la $i^{\text{ème}}$ fonction de base associée à l'élément paramétrique E (*c.-à-d.* $\varphi_i = \Upsilon_i$) si son domaine de définition est l'espace paramétrique. Si φ_i est définie sur $\mathbb{R}^d$ alors φ_i est la $i^{\text{ème}}$ fonction de base de T.

1. une forme élémentaire unitaire étant simplement une forme dont đ arêtes sont de longueur unitaire et sont confondues avec les đ axes de l'espace paramétrique.

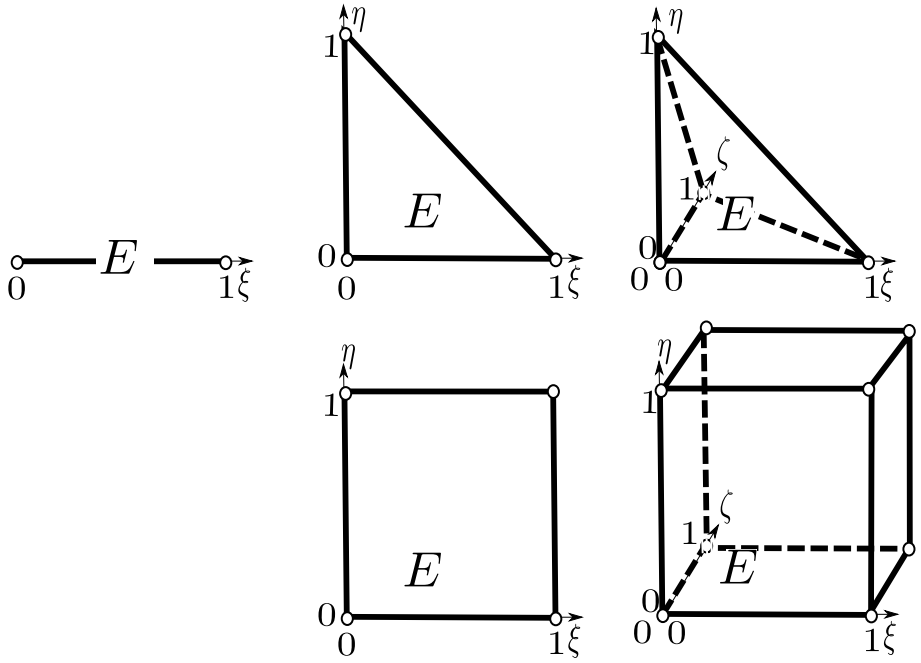


Figure II.1.4 – Quelques formes élémentaires unitaires de l'espace paramétrique de dimension $\text{đ} = 1$ à gauche, $\text{đ} = 2$ au milieu et $\text{đ} = 3$ à droite.

Nous terminerons cette section par une remarque sur la dimension de $\mathcal{V}_h^k$, l'espace vectoriel associé à un élément d'ordre k de $\mathcal{T}_h$. Cette dimension est égale au nombre de degrés de liberté de l'élément, n_{DoFs}, qui est fonction du type d'élément, de l'ordre de cet élément et de la dimension de travail. La propriété II.1.1 rappelle le nombre de DoF*s* des formes élémentaires les plus courantes.

Propriété II.1.1 (*nombre de DoFs d'un élément*)

Soit T un élément, à l'ordre k, le nombre de degrés de liberté n_{DoFs} de cet élément est tel que :

- $n_{\text{DoFs}} = \left(\frac{\prod_{l=0}^{d-1} (d+l)}{d!} \right)$, si T est un triangle ou un tétraèdre (ou encore un segment)
- $n_{\text{DoFs}} = k^d$, si T est un quadrangle ou un pavé (ou encore un segment)

En résumé, nous avons montré que la construction de fonctions de base de l'espace vectoriel $\mathcal{V}_h$ peut se ramener à la construction de fonctions de base sur des formes élé-

mentaires de l'espace paramétrique.
La section suivante se consacre à la construction de familles de fonctions de base définies sur nos formes élémentaires et permettant la représentation d'éléments courbes.

2 Fonctions de base

Dans les sous-sections suivantes nous présentons la construction de différents types de fonctions de base élémentaires, en partant des fonctions de base $\mathcal{NURBS}$, généralement utilisées en analyse isogéométrique, jusqu'aux fonctions choisies dans notre cas : les fonctions de base de Bernstein rationnelles.

2.1 Fonctions de base $\mathcal{NURBS}$

Les fonctions de base *B-Splines non uniformes rationnelles* (dites $\mathcal{NURBS}$ pour *N*on *U*niform *R*ational *B*-Splines) sont une généralisation des fonctions *B*-Splines. Nous débuterons donc cette sous-section par la présentation de ces dernières. Nous décrirons ensuite la construction des fonctions $\mathcal{NURBS}$. Pour finir nous présenterons les principales propriétés des $\mathcal{NURBS}$.

2.1.1 Fonctions de base *B*-Splines

Parmi les multiples définitions des *B*-Splines, nous donnerons ici la définition de *Cox* [49] et *de Boor* [50], qui, de par sa forme récursive, permet une implémentation directe.

2.1.1.1 Fonctions de base *B*-Splines monodimensionnelles

On suppose pour l'instant que $\mathrm{đ} = 1$.
On considère des fonctions définies sur une partition Ξ, d'un sous-intervalle I de $\mathbb{R}^{\mathrm{đ}} = \mathbb{R}$. Cette partition est telle que :

$$\Xi = \{\xi_i\}_{i=1}^{m}; \quad \xi_i \leq \xi_{i+1}, \quad \forall i \in [\![1, m-1]\!] \text{ et } I = (\xi_1, \xi_m).$$

Ξ est appelé *vecteur nodal* et les réels ξ_i qui le composent sont appelés *nœuds* du vecteur nodal. Un vecteur nodal Ξ de m nœuds engendre $n = m - k$ fonctions de base *B*-Splines d'ordre k, et donc de degré $(k-1)$, définies récursivement à partir des *B*-Splines d'ordre

$(k-1)$. La définition II.2.1 donne la formule permettant de calculer les fonctions de base B-Splines. On fournit également un exemple de fonctions de base figure II.2.2.

Définition II.2.1 (*Fonctions de base B-Splines pour đ$=1$*)

Soit k l'ordre polynomial voulu et $\Xi = \{\xi_1, \xi_2 ..., \xi_{n+k}\}$ un vecteur nodal.
Notons N_i^k la $i^{\text{ème}}$ fonction de base B-Spline d'ordre k. Les n fonctions de base $(N_i^k)_{i=1,n}$ sont définies par :

$$\begin{cases} N_i^1(\xi) &= \begin{cases} 1, \text{ si } \xi_i \leq \xi < \xi_{i+1} \\ 0, \text{ sinon} \end{cases} \\ N_i^k(\xi) &= \dfrac{\xi - \xi_i}{\xi_{i+k-1} - \xi_i} N_i^{k-1}(\xi) + \dfrac{\xi_{i+k} - \xi}{\xi_{i+k} - \xi_{i+1}} N_{i+1}^{k-1}(\xi) \end{cases} \tag{II.4}$$

où une fraction à dénominateur nul est définie comme nulle.

Exemple II.2.1 *(Fonctions de base B-splines)* :

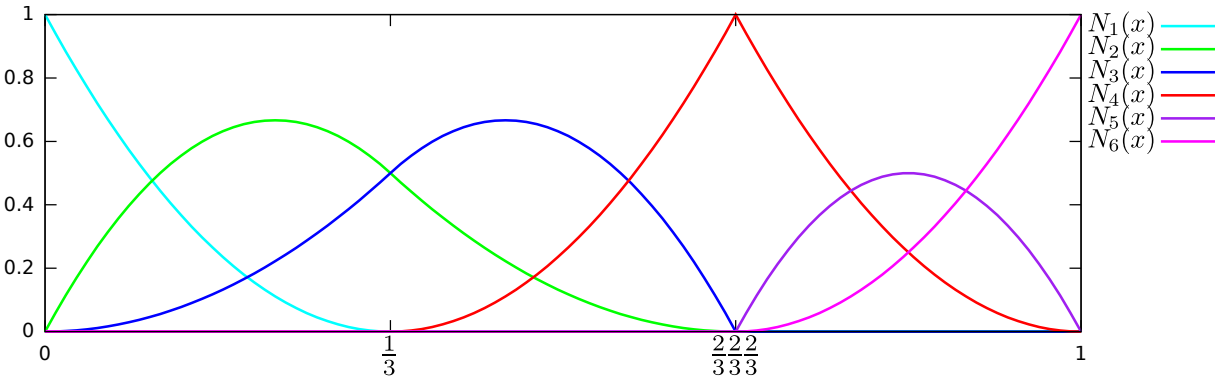


Figure II.2.2 – Fonctions de base B-splines d'ordre 3 associées au vecteur nodal ouvert $\left[0,0,0,\frac{1}{3},\frac{2}{3},\frac{2}{3},1,1,1\right]$

Remarque II.2.1 (*Multiplication d'un vecteur nodal par une constante c*)

Soient $\xi = \{\xi_1, \xi_2, ..., \xi_{n+k}\}$ et $\bar{\xi} = c\xi = \{c\xi_1, c\xi_2 ..., c\xi_{n+k}\}$, $c \in \mathbb{R}$, deux vecteurs nodaux, alors les fonctions de base engendrées par ξ et $\bar{\xi}$ sont identiques.

2.1.1.2 Fonctions de base B-Splines multidimensionnelles

Supposons maintenant đ > 1, par exemple đ $= 2$ (l'extension aux dimensions supérieures est directe et ne sera donc pas présentée ici). I est maintenant un sous-intervalle de $\mathbb{R}^2$. Ξ

est toujours une partition de I. On note $\Xi_j = \left\{\xi_{i_j}\right\}_{i_j=1}^{m_j}$ le vecteur nodal associé à la $j^{\text{ème}}$ dimension (*c.-à-d.* chaque vecteur nodal représente une direction de $\mathbb{R}^{\text{đ}}$.

Ξ est alors simplement défini comme le produit tensoriel des vecteurs nodaux associés à chaque dimension ($\Xi = \Xi_1 \otimes \Xi_2$) et les fonctions de base B-splines bidimensionnelles sont obtenues à partir des fonctions de base monodimensionnelles associées à Ξ_1 et Ξ_2 (voir définition II.2.2).

Définition II.2.2 (*Fonctions de base B-Splines pour đ$= 2$*)

Soient $\xi = \{\xi_1, \xi_2 ..., \xi_{n_1+k_1}\}$ et $\eta = \{\eta_1, \eta_2 ..., \eta_{n_2+k_2}\}$ deux vecteurs nodaux et soient k_1 et k_2 les ordres polynomiaux voulus respectivement en ξ et η.
Les $(n_1 \times n_2)$ fonctions de base B-Splines (d'ordre k_1 en ξ et k_2 en η) définies de $\Xi = \xi \otimes \eta$ dans $\mathbb{R}$ sont données par :

$$\varphi_{ij}^{k_1 k_2}(\xi, \eta) = N_i^{k_1}(\xi) \times N_j^{k_2}(\eta). \tag{II.5}$$

Remarque II.2.2 (*Numérotation des fonctions de base et des $\mathcal{CP}s$*)

Ici, nous avons utilisée une numérotation à 2 indices (généralement employée pour décrire les B-Splines) de nos fonctions de base. Le premier et le second indice étant associé, respectivement, à la première et seconde direction. Il nous arrivera (souvent) d'utiliser un unique indice pour repérer nos fonctions, par exemple $\varphi_l = \varphi_{ij}$ pour $l = (j-1)n_1 + i$. Quelque soit la numérotation utilisée, le point de contrôle associé à la fonction considérée sera toujours indicé comme cette fonction. Ainsi, si on considère un fonction de base φ_{ij}, le $\mathcal{CP}$ correspondant est noté $\tilde{P}_{ij}$ et si on considère la même fonction de base mais indicée φ_l, le $\mathcal{CP}$ est noté $\tilde{P}_l$.

On peut remarquer que, pour connaître les fonctions de base B-Splines sur Ξ il suffit de se donner :

- leur ordre ou leur degré polynomial suivant chaque dimension,
- đ vecteurs nodaux Ξ_j formés de nœuds de coordonnées croissantes (non nécessairement distinctes).

La notion de vecteur nodal est donc essentielle. Elle est associée à une autre notion cruciale que nous allons présenter : celle de *patch*. Le concept de *patch* doit être distingué celui d'*élément* généralement utilisé en éléments finis classique. L'espace Ξ définit par les vecteurs nodaux fournit un quadrillage de l'intervalle I de notre espace paramétrique. On nomme *patch paramétrique* ce quadrillage (*c.-à-d.* le produit tensoriel des Ξ_j) et *éléments*

paramétriques les éléments de ce quadrillage. Un patch est donc généralement constitué d'un ensemble d'éléments. Dans le cas particulier d'un patch contenant un unique élément, les termes de *patch* et *élément* désigneront le même objet géométrique, on pourra alors les confondre.

Nous nommerons *patch B-Spline* (ou, en anticipant sur le paragraphe suivant, *patch* $\mathcal{NURBS}$) l'image du patch paramétrique par un mapping B-Spline (*resp.* par un mapping $\mathcal{NURBS}$).

Exemple II.2.2 ***(Patchs, éléments et mappings)*** **:** La figure II.2.3 illustre le lien entre *élément* et *patch*.

- Le patch représenté contient 12 éléments.
- Le vecteur Ξ_1 est un vecteur à 10 nœuds. À l'ordre 3 est est donc associé à 7 fonctions de base.
- Le vecteur Ξ_2 est un vecteur à 9 nœuds. À l'ordre 3 est est donc associé à 6 fonctions de base.
- Le patch compte donc 42 $\mathcal{CP}s$ notés $\tilde{P}_{ij}$, avec $i \in [\![1,7]\!]$ et $j \in [\![1,6]\!]$.

Deux nœuds distincts d'un vecteur nodal délimitent une arête d'un élément de l'espace paramétrique (si $\text{đ} = 1$, ils délimitent l'élément lui-même). Dans notre exemple, les nœuds $\Xi_{1,6} = 0.5$ et $\Xi_{1,7} = 0.9$ du vecteur nodal Ξ_1 délimitent 4 arêtes :

- l'arête $([0.5, 0.9], 0)$,
- l'arête $([0.5, 0.9], 0.25)$,
- l'arête $([0.5, 0.9], 0.6)$,
- et l'arête $([0.5, 0.9], 1)$.

L'image d'une arête d'un élément paramétrique par le mapping B-Spline ou $\mathcal{NURBS}$ est une arête de l'élément physique. De même, un élément physique du patch B-Spline ou $\mathcal{NURBS}$ est l'image d'un élément paramétrique du patch paramétrique.

Remarque II.2.3 (*Vecteurs nodaux homothétiques*)

Comme deux vecteurs nodaux égaux à une constante près génèrent des fonctions de base linéiques identiques, l'image de l'un des vecteurs par une transformation $\mathcal{NURBS}$ ou B-Spline sera identique à l'image de l'autre vecteur par la même transformation. On pourra donc indifféremment travailler sur un vecteur nodal Ξ ou sur une homothétie de celui-ci. En pratique, on se ramènera donc toujours à un vecteur nodal représentant le segment $[0, 1]$ (*c.-à-d.* tel que $\xi_1 = 0$ et $\xi_{n+k} = 1$).

Jusqu'à présent, nous avons présenté très succinctement les vecteurs nodaux générant

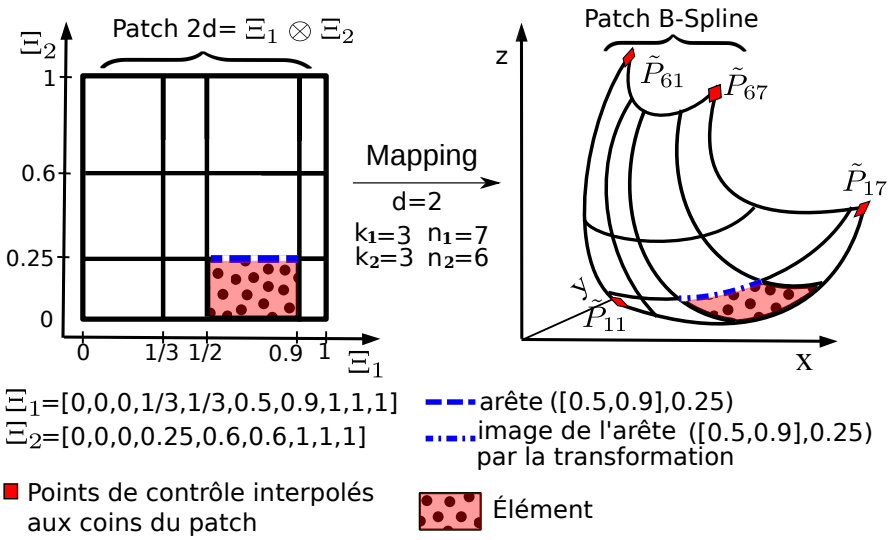


Figure II.2.3 – Transformation d'un patch paramétrique à 12 éléments (à gauche) en patch B-Spline (ou $\mathcal{NURBS}$). Une arête d'un élément est l'image d'un intervalle du vecteur nodal dans la direction paramétrique de ce vecteur pour une valeur nodale constante dans les autres directions.

le patch paramétrique. C'est pourtant leur forme qui déterminera les principales propriétés des fonctions de base. Il nous faut donc introduire quelques définitions supplémentaires concernant ces vecteurs.

Définition II.2.3 (*Vecteurs nodaux uniformes et non-uniformes*)

Un vecteur nodal est dit uniforme si les pas d'espace entre les nœuds du vecteur sont constants. Dans le cas contraire, le vecteur nodal est dit non-uniforme.

Définition II.2.4 (*Multiplicité nodale*)

Le nombre de répétition d'une même valeur nodale est appelé *degré de multiplicité* du nœud.

Définition II.2.5 (*Vecteur nodal ouvert*)

On appelle *vecteur nodal ouvert* un vecteur dont les premiers et derniers nœuds sont k fois répétés à l'ordre k (donc, dont les premiers et derniers nœuds sont de degré de multiplicité k). Par abus de langage, on nommera *patch ouvert*, le patch paramétrique associé à des vecteurs nodaux ouverts.

Évidemment, compte tenu des définitions précédentes, un vecteur nodal ouvert est nécessairement non uniforme. Les vecteurs Ξ_1 et Ξ_2 de la figure II.2.3 sont, par exemple,

deux vecteurs dont les premiers et derniers nœuds sont de degré de multiplicité 3. Comme on travaille à l'ordre 3 (dans les 2 directions paramétriques), ils sont donc non uniformes et ouverts.

Nous pouvons maintenant nous intéresser au lien entre le vecteur nodal et les caractéristiques des fonctions de base.

Propriété II.2.1 (*vecteur nodal et régularité des fonctions de base*)

Les fonctions de base B-Splines d'ordre k sont C^{k-1} continues entre les nœuds du vecteur nodal et C^{k-p-1} continues en un nœud de degré multiplicité p.
En particulier :

- les fonctions de base sont C^0 en un nœud de multiplicité $(k-1)$. Elles sont alors interpolantes en ce nœud.
- les fonctions de base sont discontinues en un nœud de multiplicité k. Le patch est alors séparé en deux patchs distincts.

On peut déduire de la propriété II.2.1 que les fonctions de base sont interpolantes aux extrémités d'un vecteur nodal ouvert. Une courbe, une surface ou un volume B-Spline partage, bien sûr, les propriétés de régularité des fonctions de base permettant de la (ou le) représenter.

Exemple II.2.3 ***(Effet du degré de multiplicité d'un nœud)*** **:** On peut voir figure II.2.2 l'effet du degré de multiplicité des nœud sur les fonctions de base. Le nœud $\frac{2}{3}$ est d'ordre de multiplicité 2, la régularité des fonctions de base est donc plus faible en ce point. De plus, comme on travaille à l'ordre 3, ce nœud est d'ordre de multiplicité $(k-1)$, les fonctions de base sont donc interpolantes en ce nœud. Mathématiquement, ceci ce traduit par le fait qu'une seule fonction de base est non-nulle en $\frac{2}{3}$.

Exemple II.2.4 ***(Effet du degré de multiplicité d'un nœud)*** **:** Le patch paramétrique de la figure II.2.3 est issu de deux vecteurs nodaux ouverts, les fonctions de base associées à ce patch sont donc interpolantes aux coins du patch, et la surface B-Spline obtenue interpole les quatre points de contrôle des coins du patch.

Dans la suite de ce travail, nous ne travaillerons que sur des vecteurs nodaux non uniformes ouverts. Les fonctions de base associées à de tels vecteurs nodaux sont alors appelées *B-Splines non-uniformes*. Nous supposerons enfin, que quelle que soit la dimension, notre vecteur nodal représente le segment paramétrique $[0,1]$ donc, que $\xi_1 = 0$ et

$\xi_m = 1$ (ceci n'est pas une supposition très contraignante car tout segment peut être ramené au segment $[0,1]$ par translation et changement d'échelle).

2.1.2 Fonctions de base $\mathcal{NURBS}$

Les fonctions de base $NURBS$ ou B-Splines non uniformes rationnelles sont construites comme une moyenne pondérée par une série de poids $(W_i)_{i=1,N_{\text{ba}}}$ des fonctions de base B-Splines (voir définitions II.2.6 et II.2.7 pour $đ = 1$ et $đ = 2$ respectivement).

Afin d'éviter les singularités pouvant apparaître en présence de poids négatifs ([51]) nous n'utiliserons dans ce travail que des poids strictement positifs.

Définition II.2.6 (*fonctions de base* $\mathcal{NURBS}$ *pour* $đ = 1$)

Soit k l'ordre polynomial voulu et $\Xi = \{\xi_1, \xi_2 ..., \xi_{n+k}\}$ un vecteur nodal. On note N_i^k la $i^{\text{ème}}$ fonction de base B-Spline définie sur Ξ et d'ordre k.
Soit $(W_j)_{j=1,n}$ un ensemble de poids tels que,

$$\forall j,\ W_j \in]0,1]\,.$$

Les n fonctions de base $\mathcal{NURBS}$, notées $(M_i^k)_{i=1,n}$, sont alors définies telles que :

$$M_i^k(\xi) = \frac{W_i N_i^k(\xi)}{\sum_{l=1,n} W_l N_l^k(\xi)}\,. \tag{II.6}$$

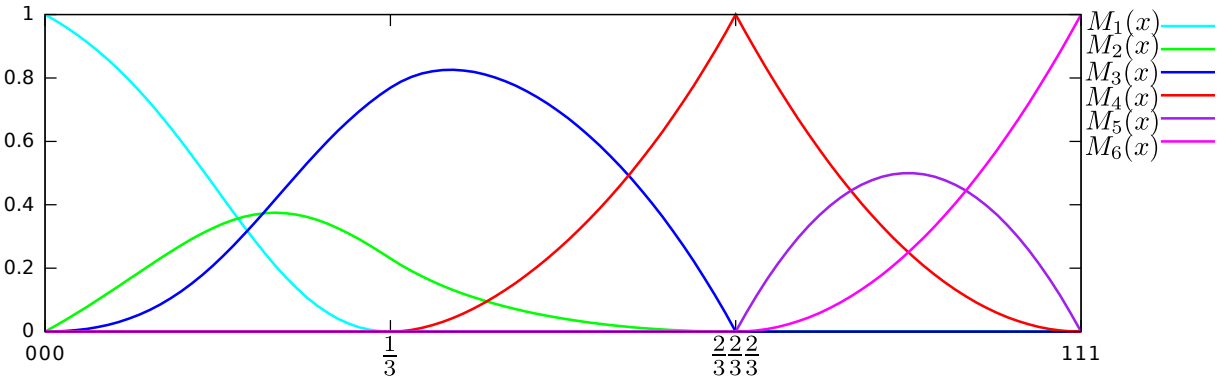


Figure II.2.4 – Fonctions de base $\mathcal{NURBS}$ d'ordre 3 associées au vecteur nodal ouvert $\left[0,0,0,\frac{1}{3},\frac{2}{3},\frac{2}{3},1,1,1\right]$ et aux poids $W_2 = 0.3$ et $W_1 = W_3 = W_4 = W_5 = W_6 = 1$

Définition II.2.7 (*fonctions de base* $\mathcal{NURBS}$ *pour* $\mathbb{d} = 2$)

Soient $\xi = \{\xi_1, \xi_2 ..., \xi_{n_1+k_1}\}$ et $\eta = \{\eta_1, \eta_2 ..., \eta_{n_2+k_2}\}$ deux vecteurs nodaux et soient k_1 et k_2 les ordres polynomiaux voulus respectivement en ξ et η. On note $\varphi_{ij}^{k_1 k_2}$ les fonctions de base B-splines définies sur $\Xi = \xi \otimes \eta$ et d'ordre k_1 en ξ et k_2 en η.
Soit $(W_{ij})_{\substack{i=1,n_1 \\ j=1,n_2}}$ un ensemble de $n \times m$ poids tels que,

$$\forall j,\, W_j \in]0, 1]\,.$$

Les $(n_1 \times n_2)$ fonctions de base $\mathcal{NURBS}$, notées $\Psi_{ij}^{k_1 k_2}$, sont alors définies telles que :

$$\Psi_{ij}^{k_1 k_2}(\xi, \eta) = \frac{W_{ij} \varphi_{ij}^{k_1 k_2}(\xi, \eta)}{\sum\limits_{\substack{l_1=1,n_1 \\ l_2=1,n_2}} W_{l_1 l_2} \varphi_{l_1 l_2}^{k_1 k_2}(\xi, \eta)}$$

.

Pour résumer, il nous suffit de connaître le ou les vecteurs nodaux générant Ξ, l'ordre recherché dans chaque direction paramétrique et les poids W associés à chaque fonction de base pour pouvoir calculer nos fonctions de base $\mathcal{NURBS}$.

Exemple II.2.5 ***(Fonctions de base*** $\mathcal{NURBS}$***)*** **:** La figure II.2.4 donne un exemple de fonctions $\mathcal{NURBS}$. Le vecteur nodal utilisé est le même que dans l'exemple pour les B-Splines ce qui permet d'observer l'effet du poids W_2. On constate que, M_2, la fonction de base associée à W_2, est de maximum plus faible lorsque la valeur du poids W_2 est plus faible. Le point de contrôle extrapolé sur M_2 aura donc une influence plus faible sur la courbe. Ainsi, plus un poids sera élevé, plus le point de contrôle associé attirera la courbe vers lui.

2.1.3 Propriétés des B-Splines et $\mathcal{NURBS}$

Maintenant que nous avons présenté les fonctions de base $\mathcal{NURBS}$, nous allons nous intéresser à leurs propriétés. Ces propriétés proviennent des B-Splines utilisées pour construire nos $\mathcal{NURBS}$:

1. Les fonctions de base (B-Splines ou $\mathcal{NURBS}$) associées à un vecteur nodal ouvert

constituent une partition de l'unité :

$$\forall k,\ \forall \boldsymbol{\xi} \sum_{i=0}^{k} \varphi_i^k(\boldsymbol{\xi}) = 1.$$

2. Une fonction de base (B-Splines ou $\mathcal{NURBS}$) φ_i^k est à support compact sur $[\xi_i, \xi_{i+k}]$:

$$\begin{cases} \varphi_i^k(\boldsymbol{\xi}) > 0 & \text{si} \quad \xi_i \leq \boldsymbol{\xi} < \xi_{i+k} \\ \varphi_i^k(\boldsymbol{\xi}) = 0 & \text{sinon} \end{cases}$$

3. Les fonctions de base sont positives ou nulles sur la totalité de leur domaine de définition, le patch Ξ.
4. Les propriétés de continuité des $\mathcal{NURBS}$ sont les même que celles des B-Splines (voir propriété II.2.1)
5. Les fonctions de base $\mathcal{NURBS}$ permettent de représenter exactement les coniques.

Remarque II.2.4 (*Les fonctions de base* $\mathcal{NURBS}$ *ne contiennent pas de pôles*)

Comme avec toute fonction rationnelle, il faut s'intéresser à la présence ou non de pôles dans nos fonctions $\mathcal{NURBS}$. Notons $(\varphi_j^k)_{j=1,N_{\text{ba}}}$ les fonctions de base B-Splines nous permettant de définir nos fonctions de base $\mathcal{NURBS}$.
On considère la $i^{\text{ème}}$ fonction de bas $\mathcal{NURBS}$.
Étudions le terme $\sum_{j=1,N_{\text{ba}}} W_j \varphi_j^k(\boldsymbol{\xi})$:

$$\begin{aligned} \sum_{j=1,N_{\text{ba}}} W_j \varphi_j^k(\boldsymbol{\xi}) &\geq \min_{j \in N_{\text{ba}}} (W_j) \sum_{j=1,N_{\text{ba}}} \varphi_j^k(\boldsymbol{\xi}) \\ &\geq \min_{j \in N_{\text{ba}}} (W_j) \end{aligned}$$

Comme les poids W_j sont tous strictement positifs, $\sum_{j=1,N_{\text{ba}}} W_j \varphi_j^k(\boldsymbol{\xi}) \geq 0$.

On en déduit que le terme $\sum_{j=1,N_{\text{ba}}} W_j \varphi_j^k(\boldsymbol{\xi})$ est strictement positif et donc, que les fonctions de base $\mathcal{NURBS}$ ne contiennent pas de pôles sur leur patch de définition

Remarque II.2.5 (*Une* B*-Spline est une* $\mathcal{NURBS}$)

Les fonctions de base B-Splines sont en fait des fonctions $\mathcal{NURBS}$ dont tous les poids W sont égaux.

Démonstration : Soit Ξ un patch paramétrique et φ_i la $i^{\text{ème}}$ fonction de base B-Spline d'ordre k associée. On se donne N_{ba} poids W tels que $\forall j,\ \forall l,\ W_j = W_l$. Notons Ψ_i la

$i^{\text{ème}}$ fonction de base $\mathcal{NURBS}$ d'ordre k associée à Ξ et aux W_j. Par définition :

$$\Psi_i(\boldsymbol{\xi}) = \frac{W_i \varphi_i(\boldsymbol{\xi})}{\sum_{j=1}^{N_{\text{ba}}} W_j \varphi_j(\boldsymbol{\xi})}.$$

Or, $\forall j, W_j = W_i$ donc,

$$\Psi_i(\boldsymbol{\xi}) = \frac{W_i \varphi_i(\boldsymbol{\xi})}{W_i \sum_{j=1}^{N_{\text{ba}}} \varphi_j(\boldsymbol{\xi})} = \frac{\varphi_i(\boldsymbol{\xi})}{\sum_{j=1}^{N_{\text{ba}}} \varphi_j(\boldsymbol{\xi})}.$$

Comme on l'a dit, Ξ est un patch généré par des vecteurs nodaux ouverts, or, pour de tels vecteurs, les fonctions de base B-Splines forment une partition de l'unité donc, $\sum_{j=1}^{N_{\text{ba}}} \varphi_j(\boldsymbol{\xi}) = 1$ et donc, $\Psi_i(\boldsymbol{\xi}) = \varphi_i(\boldsymbol{\xi})$. ■

Les courbes, surfaces et volumes $\mathcal{NURBS}$ possèdent également quelques propriétés qu'il est intéressant de citer. Soit $(\tilde{P}_i)_{i=1,N_{\text{ba}}}$ un ensemble de points de contrôle et soit $S = \sum_{i=1}^{N_{\text{ba}}} \tilde{P}_i \varphi_i$ la courbe, la surface ou le volume obtenu par transformation $\mathcal{NURBS}$, alors :

- S est inclus dans l'enveloppe convexe de son polygone de contrôle (voir figure II.2.5).

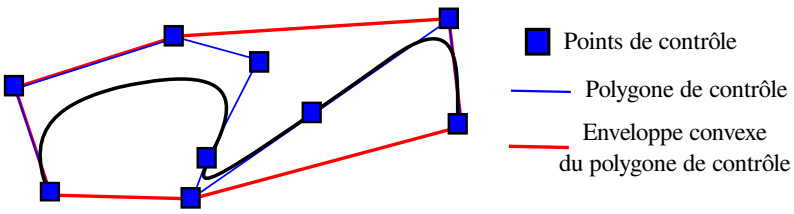


Figure II.2.5 – Propriété d'enveloppe convexe

- Si S est l'image d'un patch paramétrique ouvert, alors S est tangente en ses extrémités à son polygone de contrôle (également visualisable figure II.2.5).
- Soit g une transformation affine, alors $g(S) = \sum_{l=i}^{N_{\text{ba}}} g\left(\tilde{P}_i\right) \varphi_i$, ou, autrement dit, l'image par g de S a pour points de contrôle l'image par g des points de contrôle de S. C'est la propriété d'*invariance affine*.
- Une géométrie $\mathcal{NURBS}$, notée S_r, de dimension d peut-être vue comme la projection dans $\mathbb{R}^d$ d'une géométrie B-Spline définie dans l'hyperespace $\mathbb{R}^{d+1}$. Notons $(\tilde{P}_l)_{l=1,n}$, $\tilde{P}_l \in \mathbb{R}^d$, les points de contrôle et $(W_l)_{l=1,n}$ les poids associés à S_r, alors S_r

est la projection dans $\mathbb{R}^d$ de la géométrie B-Spline associée aux points de contrôle de coordonnées homogènes $\tilde{P}_{\mathrm{h}l}$ dans $\mathbb{R}^{d+1}$ (voir figure II.2.6). La définition des coordonnées homogènes est fournie définition II.2.8. Nous nommerons cette propriété la propriété de *transformation projective*.

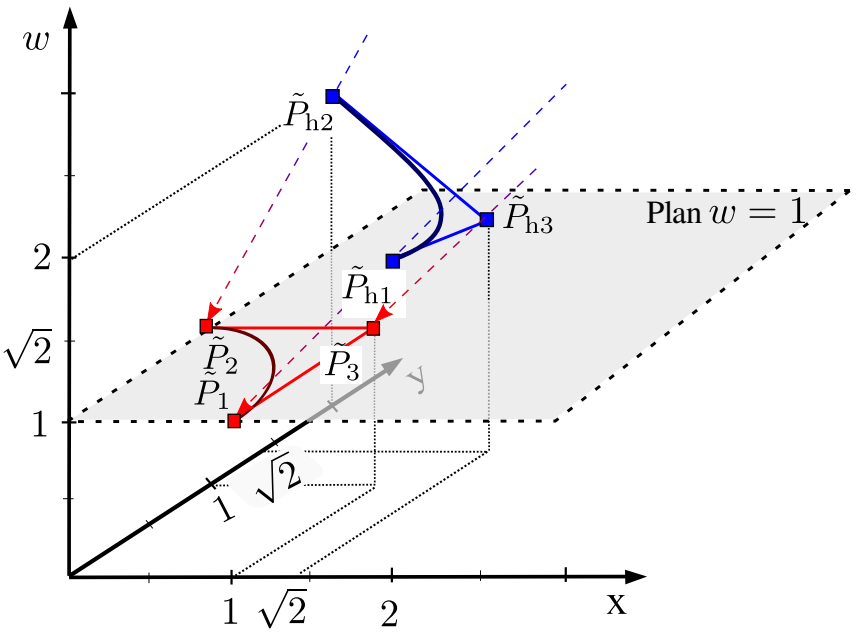


Figure II.2.6 – Propriété de transformation projective :
La courbe $\mathcal{NURBS}$ de $\mathbb{R}^2$ et son polygone de contrôle (en rouge) sont obtenus par projection sur le plan $\{w = 1\}$ de la courbe B-Spline de l'hyperespace $\mathbb{R}^3$ et du polygone de ses points de contrôle homogènes.

Définition II.2.8 (*Coordonnées homogènes*)

Soient $(\tilde{P}_l)_{l=1,n}$ une suite de points de $\mathbb{R}^d$ de coordonnées $(X_i^l)_{i=1,d}$ et $(W_l)_{l=1,n}$ une suite de poids. On note $\tilde{P}_{\mathrm{h}l}$ les points de l'hyperespace $\mathbb{R}^{d+1}$ de coordonnées $(X_{\mathrm{h}i}^l)_{i=1,d+1}$ tels que :

$$\begin{cases} X_{\mathrm{h}i}^l = W_l X_i^l, & \text{si } i \in [\![1, d]\!] \\ X_{\mathrm{h}d}^l = W_l \end{cases}$$

les d premières coordonnées de $\tilde{P}_{\mathrm{h}l}$ sont donc obtenues en multipliant par W_l chaque coordonnées du point de contrôle $\tilde{P}_l$ et en prenant le poids W_l comme $(d+1)^{\text{ème}}$ coordonnée. Ce poids représente alors la distance de $\tilde{P}_{\mathrm{h}l}$ à l'espace $\mathbb{R}^d$.

On peut noter que si, pour tout l, $W_l = 1$, alors tous les points homogènes appartiennent au sous-espace R^d de R^{d+1} et la projection de la géométrie B-Spline est égale à elle-même, c'est donc une B-Spline. Ceci confirme la remarque II.2.5.

Remarque II.2.6 (*Les* $\mathcal{NURBS}$ *représentent exactement les coniques*)

La propriété de transformation projective est intéressante car elle permet de comprendre la capacité des $\mathcal{NURBS}$ à représenter exactement les coniques : une conique est la projection d'un polynôme de degré 2 de l'hyperplan

Exemple II.2.6 ***(Transformation projective d'une parabole)*** **:** La figure II.2.6 fournit un exemple de la propriété de transformation projective et de la représentation d'une conique par une $\mathcal{NURBS}$. Les points de contrôles homogènes ont pour coordonnées :

$$\begin{cases} \tilde{P}_{\mathrm{h}1} &= (2,0,2) \\ \tilde{P}_{\mathrm{h}3} &= (\sqrt{2},\sqrt{2},\sqrt{2}) \\ \tilde{P}_{\mathrm{h}2} &= (0,2,2) \end{cases}$$

Les points de contrôles non homogènes ont pour poids et coordonnées :

$$\begin{cases} \tilde{P}_1 = (1,0,1), & W_1 = 2 \\ \tilde{P}_3 = (1,1,1), & W_3 = \sqrt{2} \\ \tilde{P}_2 = (0,1,1), & W_2 = 2 \end{cases}$$

Aux propriétés précédemment citées s'ajoute la propriété d'approximation II.2.2 dont nous allons détailler les notations. On considère un élément T d'un patch. Notons h_1 le rayon du plus petit des cercles contenant tous les sommets de T et h_2 le rayon du plus grand des cercles inclus dans T. Nous nommerons h_1 le *rayon maximal* de T et h_2 son *rayon minimal*.

On désigne par L le polynôme interpolant de Lagrange et par I l'opérateur de projection sur l'espace vectoriel des fonctions de base $\mathcal{NURBS}$ tel que, pour une fonction f :

$$I(f)(x) = \frac{L(Wf)}{W}, \tag{II.7}$$

où W est la fonction poids définie par $W = \sum_{i=1}^{N_{\text{ba}}} W_i \varphi_i$.

Propriété II.2.2 (*Erreur d'interpolation* $\mathcal{NURBS}$)

Si f est de classe $C^p(T)$, alors il existe une constante $C'(f,p)$ dépendant de la régularité de f telle que :

$$\|f - I(f)\|_{H^m} \leq C'(f,p) \frac{h_1^k}{h_2^m}, \quad m \leq p, \tag{II.8}$$

avec h_1 et h_2 les rayons de T, k l'ordre des fonctions de base $\mathcal{NURBS}$ et $\|.\|_{H^m}$ la norme standard de Sobolev définie par $\|.\|_{H^m} = \sqrt{\sum_{|\alpha| \leq m} \|D^\alpha . \|_{L^2}^2}$.

Démonstration : L est le projecteur de Lagrange sur T, donc, à l'ordre k, $L(wf) \in \mathbb{P}^{k-1}(T)$.

Si f est une fonction de classe C^p, alors Wf est aussi une fonction de classe C^p, et donc :

$$\|Wf - L(Wf)\|_{H^m} \leq C(f,p) \frac{h_1^k}{h_2^m}, \quad m \leq p,$$

où C est une constante dépendant de la régularité de f (voir [52]).

Comme W est une fonction strictement positive, on en déduit que :

$$\|f - \frac{L(Wf)}{W}\|_{H^m} \leq C'(f,p) \frac{h_1^k}{h_2^m}, \quad m \leq p.$$

■

La manipulation géométrique des patchs de l'espace réel (rotations, transformations, représentation d'une forme particulière, *etc.*) se fait à travers la manipulation des poids et points de contrôle. La manipulation fondamentale des patchs, *c.-à-d.* la manipulation des caractéristiques numériques des patchs (ordre et nombre d'éléments), est réalisée via l'insertion et la suppression de nœuds dans les vecteurs nodaux générant le patch paramétrique. De par la construction tensorielle de l'espace paramétrique sur lequel sont définies les fonctions de base B-Splines et $\mathcal{NURBS}$ multidimensionnelles, cet espace paramétrique est nécessairement quadrangulaire. Ces fonctions de base ne peuvent donc pas être utilisées sur des triangles ou tétraèdres paramétriques. La sous-section suivante s'intéresse à la construction d'une famille de fonctions de base définie sur les éléments non-tensoriels et permettant de représenter des frontières courbes (et plus particulièrement coniques).

2.2 Fonctions de base de Bernstein rationnelles

Nous souhaitons représenter nos frontières par une géométrie $\mathcal{NURBS}$. Les arêtes de nos éléments (qu'ils soient triangulaires ou tétraédriques) seront donc représentées par des courbes $\mathcal{NURBS}$. La représentation de la géométrie nous imposera ainsi le choix des poids W_i. Nous cherchons donc à construire des fonctions de base multidimensionnelles qui, le long des arêtes de notre élément paramétrique, soient égales aux fonctions de base $\mathcal{NURBS}$ monodimensionnelles. Plus simplement, la restriction de nos fonctions de base multidimensionnelles à la dimension 1, doit être égale aux fonctions de base $\mathcal{NURBS}$ monodimensionnelles.

Débutons cette sous-section par quelques simplifications numériques ainsi que leurs conséquences sur nos fonctions de base $\mathcal{NURBS}$ $1D$.

1. Nous associerons, par soucis de simplicité, un vecteur nodal à chaque arête du maillage et, afin de ne pas stocker ces vecteurs nodaux, nous utiliserons un unique vecteur nodal pour toutes les arêtes. Ce vecteur formera notre patch paramétrique linéique unitaire donc son premier et son dernier nœud seront respectivement égaux à 0 et 1.
2. Nous choisissons d'utiliser un vecteur nodal ouvert (ses premiers et derniers nœuds seront donc de degré de multiplicité k). Ceci nous garantit que les extrémités du patch seront interpolantes et donc que les extrémités de nos arêtes physiques (*c.-à-d.* les sommets de nos éléments physiques) seront interpolés. En outre, nos fonctions de base formeront une partition de l'unité.
3. Pour finir, nous souhaitons conserver la compacité du stencil de notre schéma. Nous voulons donc, à chaque pas de temps, pouvoir actualiser la valeur d'un nœud M_i en utilisant uniquement ses voisins directs (*c.-à-d.* les nœuds partageant un élément avec M_i). Notons ξ_l les nœuds de notre vecteur nodal. La $l^{\text{ème}}$ fonction de base $\mathcal{NURBS}$ φ_l est à support compact sur $]\xi_l, \xi_{l+k}]$. Rappelons qu'un élément du patch paramétrique est délimité par deux nœuds distincts du vecteur nodal, la valeur de contrôle associée à φ_l (la $l^{\text{ème}}$ valeur de contrôle) aura donc une influence sur les éléments de l'intervalle $]\xi_l, \xi_{l+k}]$.
   Pour conserver la compacité, il faut que nous puissions calculer la valeur de M_i à partir des points de contrôle associés aux seuls éléments auxquels appartient M_i. Notre intervalle $]\xi_l, \xi_{l+k}]$ doit donc contenir un unique élément, c'est à dire seulement deux nœuds distincts. Ceci revient à n'avoir que des nœuds de multiplicité k dans

notre vecteur nodal.

Comme un patch est partagé en deux patchs distincts aux nœuds de multiplicité k, cette dernière condition conduit finalement à utiliser le vecteur nodal à $2k$ composantes tel que :

$$\Xi = \left[\underbrace{0,0,\cdots}_{k \text{ fois}},\underbrace{\cdots,1,1}_{k \text{ fois}}\right].$$

Nous avons un unique élément par patch, donc, comme dans le cas des éléments finis classiques, les deux notions pourront être confondues. En contrepartie, du fait de la forme de notre vecteur nodal, nos fonctions de base seront C^0 à l'interface de deux éléments.

Nous pouvons remarquer que les fonctions de base B-Splines associées à ce vecteur nodal sont les fonctions de base de *Bézier* ou de *Bernstein* (puisque dans le cas monodimensionnel, ces deux types de fonctions de base sont confondus).

Les fonctions de base de Bernstein étant définies à la fois sur les éléments quadrangulaires et triangulaires, elles satisfont toutes les conditions que nous requerrons. Afin de représenter les coniques, nous nous inspirerons de la construction des fonctions $\mathcal{NURBS}$ à partir des fonctions B-Splines pour construire, à notre tour, une famille de fonctions représentant exactement les coniques à partir des fonctions de Bernstein. Nous nommerons ces fonctions les *fonctions de base de Bernstein rationnelles*. Ces fonctions seront donc une moyenne pondérée par des poids des fonctions de base de Bernstein classiques.
Présentons tout d'abord les fonctions de base de Bernstein classiques.

2.2.1 Fonctions de base de Bernstein

L'idée des polynômes de Bernstein pour construire des fonctions de degré $k-1$ est de partir du développement multinomial :

$$1 = \left(\sum_{i=1}^{n_b} \lambda_i\right)^{k-1}, \tag{II.9}$$

où λ_i désigne la $i^{\text{ème}}$ coordonnée barycentrique associée à la forme élémentaire considérée et où n_b est le nombre de coordonnées barycentriques sur cette forme.

On considère une forme élémentaire unitaire quelconque de l'espace paramétrique. On note $\lambda = (\lambda_i)_{i=1,n_b}$ les coordonnées dans le repère barycentrique associé à cette forme. On

a donc $\sum_{i=1}^{n_b} \lambda_i = 1$, et donc $\left(\sum_{i=1}^{n_b} \lambda_i\right)^{k-1} = 1$. On obtient une somme de n_{DoFs} fonctions qui formeront notre base.

Plus formellement, quelque soit la dimension et le type d'élément considéré, les fonctions de base de Bernstein seront donnés par l'équation (II.10).

Définition II.2.9 (*Fonctions de base de Bernstein d'ordre* k)

Soient $i_1, i_2, ...i_{n_b}$ des entiers de l'intervalle $[\![0, k-1]\!]$, l'ensemble $\left\{\varphi_i^k\right\}_{i=1,...,n_{\text{DoFs}}}$ également noté, en numérotation dite *barycentrique*, $\left\{\varphi_{i_1 i_2 ... i_{n_b}}\right\}$ et tel que :

$$\left\{\varphi_{i_1 i_2 ... i_{n_b}}\right\} = \left\{ \binom{k-1}{i_1, i_2, \ldots, i_{n_b}} \lambda_1^{i_1} \lambda_2^{i_2} \ldots \lambda_{n_b}^{i_{n_b}}, i_1 + i_2 + \ldots + i_{n_b} = k-1 \right\} \tag{II.10}$$

forme la famille des fonctions de base de Bernstein.

Comme précédemment expliqué, les fonctions de base de Bernstein rationnelles seront construites par moyenne pondérée des fonctions de base de Bernstein classiques. Notons $\{M_i^k\}_{i=1,n_{\text{DoFs}}}$ notre famille de fonctions rationnelles d'ordre k. Soit $\{W_i\}_{i=1,n_{\text{DoFs}}}$ un ensemble de poids, alors la $i^{\text{ème}}$ fonction de base de Bernstein rationnelle est telle que :

$$M_i^k(\boldsymbol{\xi}) = \frac{W_i \varphi_i^k(\boldsymbol{\xi})}{\sum_{l=1}^{n_{\text{DoFs}}} W_l \varphi_l^k(\boldsymbol{\xi})} . \tag{II.11}$$

L'expression générique des fonctions de base de Bernstein que nous avons fourni définition II.2.9 étant très abstraite, nous nous intéresserons dans les paragraphes suivants à l'étude de nos fonctions de base pour quelques-unes des formes élémentaires unitaires de l'espace paramétrique.

2.2.2 Fonctions de base de Bernstein rationnelles monodimensionnelles

2.2.2.1 Fonctions de base

En dimension 1, les fonctions de Bernstein sont égales aux fonctions B-Splines associées au vecteur nodal $[\![0_k, 1_k]\!]$. Les fonctions de Bernstein rationnelles monodimensionnelles sont donc égales aux fonctions $\mathcal{NURBS}$ associées à ce même vecteur nodal. Vérifions que ces fonctions forment effectivement une base de notre espace.

Démonstration : Commençons par prouver que quelque soit l'ordre k considéré, la famille

des fonctions de Bernstein non rationnelles forme bien une base de notre espace.
À l'ordre k, le segment paramétrique unitaire compte k degrés de liberté, il génère donc un espace de dimension k. Pour n_b coordonnées barycentriques, le développement multinomial de l'expression (II.9) est une somme de $C^{k-1}_{n_b+k-2}$ termes (loi des tirages non ordonnés avec répétition). Pour un segment, $n_b = 2$, la famille des $\{\varphi_i\}$ est donc composée de k fonctions. Si ces fonctions sont linéairement indépendantes, elles formeront une base de notre segment.
D'après l'équation (II.10) nos fonctions de base sont de la forme $\varphi_i(\xi) = c_i(1-\xi)^{k-1-i}\xi^i$, où c_i est un coefficient indépendant de ξ. On prouve qu'elles sont linéairement indépendantes par récurrence. Soit $(\alpha_i)_{i=0,k-1}$ une famille de coefficients à valeurs réelles. On considère la proposition $\mathcal{P}(k)$.

$$\mathcal{P}(k) \ : \text{ Si } \sum_{i=0}^{k-1} \alpha_i (1-\xi)^{k-1-i}\xi^i = 0, \forall \xi \in [0;1], \text{ alors tous les coefficients } \alpha_i \text{ sont nuls.}$$

Pour $k = 2$, il est trivial de montrer que $\mathcal{P}(k)$ est vérifiée.
On suppose que $\mathcal{P}(l)$ est vérifiée. Pour $k = l+1$, on cherche les $(\alpha_i)_{i=0,k-1}$ tels que :

$$\sum_{i=0}^{k-1} \alpha_i (1-\xi)^{k-1-i}\xi^i = \sum_{i=0}^{l} \alpha_i (1-\xi)^{l-i}\xi^i = 0 \qquad \forall \xi \in [0;1].$$

Comme $\xi = 1 \Rightarrow \alpha_l = 0$, on a :

$$\sum_{i=0}^{l-1} \alpha_i (1-\xi)^{l-i}\xi^i = (1-\xi)\sum_{i=0}^{l-1} \alpha_i (1-\xi)^{l-1-i}\xi^i = 0 \qquad \forall \xi \in [0;1].$$

On en déduit, d'après l'hypothèse de récurrence $\mathcal{P}(l)$, que les coefficients α_i sont nuls pour $i = 1, l-1$. De plus, $\alpha_l = 0$ donc tous les coefficients α_i sont nuls et $\mathcal{P}(k+1)$ est vraie.
Par récurrence, $\mathcal{P}(k)$ est donc vraie quelque soit k. Les polynômes de Bernstein sont donc indépendants et forment une base de notre espace.
Les k fonctions de base de Bernstein nous permettent de définir k fonctions de base rationnelles. Elles formeront donc une base de notre espace de dimension k si et seulement si elles sont linéairement indépendantes.

$$\sum_{i=1}^{k} \alpha_i M_i = 0 \Leftrightarrow \sum_{i=1}^{k} \alpha_i W_i \varphi_i = 0. \tag{II.12}$$

Comme les fonctions de base de Bernstein sont linéairement indépendantes, ceci implique que tout les coefficients α_i sont nuls.

■

La figure II.2.7 représente les fonctions de base de Bernstein monodimensionnelles d'ordre 3 et les fonctions de base de Bernstein rationnelles monodimensionnelles du même ordre pour les poids $W_1 = W_3 = 1$ et $W_2 = 0.3$.

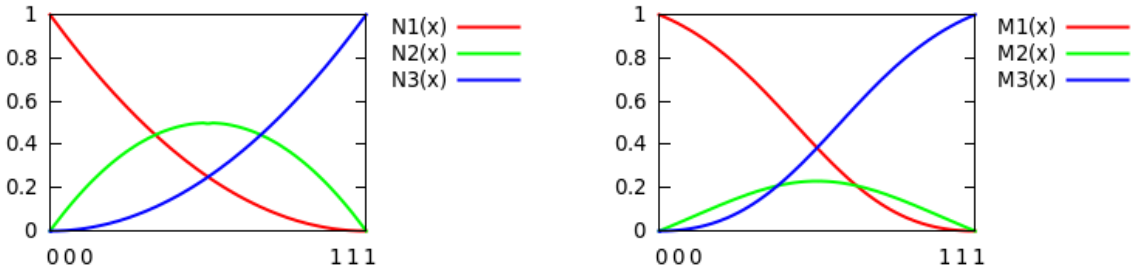


Figure II.2.7 – Fonctions de base de Bernstein d'ordre 3 :
Fonctions non rationnelles à gauche et fonctions rationnelles pour $W_2 = 0.3$, et $W_1 = W_3 = 1$ à droite.

2.2.2.2 Patchs de Bernstein monodimensionnels

Commençons par rappeler la définition d'un patch de Bernstein.

Définition II.2.10 (*Patch de Bernstein*)

Soit Ξ une forme paramétrique unitaire et soit $(P_i)_{i=1,N_{\text{ba}}}$ un ensemble de points de contrôle.

On nomme *patch de Bernstein* l'extrapolation des $\mathcal{CP}s$ sur les fonctions de bases de Bernstein associées à Ξ.

Nous allons maintenant nous intéresser à la modification des patchs. De même que les patchs $\mathcal{NURBS}$, les patchs de Bernstein peuvent être manipulés géométriquement ou fondamentalement. Dans notre cas, la géométrie de nos patchs physiques est imposée par la géométrie du domaine sur lequel on souhaite résoudre notre problème, nous ne manipulerons donc pas géométriquement nos patchs.
En revanche, nous aurons recours à la manipulation fondamentale des éléments de notre espace paramétrique. Nous décrirons donc dans la suite de ce paragraphe les méthodes de subdivision et d'élévation d'ordre d'un segment paramétrique.

Considérons tout d'abord l'élévation d'ordre d'un patch :

- À l'ordre k, notre segment paramétrique est le segment $[0_k, 1_k]$, où k désigne le degré de multiplicité du nœud indicé, *c.-à-d.* $\alpha_k = \{\underbrace{\alpha, \alpha, \cdots}_{k \text{ fois}}\}$.
- À l'ordre $(k+1)$, notre segment paramétrique est le segment $[0_{k+1}, 1_{k+1}]$.

Le passage de l'ordre k à l'ordre $(k+1)$, du point de vue du segment paramétrique, se fait donc simplement par l'augmentation de 1 du degré de multiplicité de chaque nœud.

Considérons maintenant la subdivision d'un patch en deux patchs identiques :

- Notre segment paramétrique initial est le segment $[0_k, 1_k]$.
- Les deux sous-segments égaux que l'on souhaite obtenir sont les sous-segments $\left[0_k, \frac{1}{2}_k\right]$ et $\left[\frac{1}{2}_k, 1_k\right]$.

Cette opération est donc fondée sur l'insertion de nœuds dans notre vecteur. Comme, en conséquence des propriétés de régularité des fonctions B-Splines (propriétés II.2.1), un nœud de multiplicité k du vecteur nodal partage un patch paramétrique en deux patchs, il suffit, pour partager notre élément paramétrique, d'insérer k fois le nœud $\frac{1}{2}$ dans notre vecteur nodal.

La manipulation fondamentale de nos patchs repose donc entièrement sur l'insertion nodale. La géométrie représentée par notre patch physique ne doit pas être modifiée par cette ou ces insertions. Comme les nœuds d'un patch paramétriques influent sur les fonctions de base associées à celui-ci, les $\mathcal{CP}s$ (qui, rappelons-le, une fois extrapolés sur nos fonctions de base, nous permettent de représenter la géométrie physique) seront également modifiés et devront être recalculés à chaque insertion. Par ailleurs, comme un vecteur nodal à $(n+k)$ nœuds est associé à n fonctions de base et à n points de contrôle, si l'ordre n'est pas modifié, on aura un $\mathcal{CP}$ supplémentaire à chaque insertion. Si l'ordre est modifié, pour un vecteur nodal ouvert à un élément, nous aurons 2 insertions (le nœud 0 et le nœud 1) et le nombre de $\mathcal{CP}s$ sera incrémenté de 1. Les formules d'insertion de nœuds sans modification géométrique de la courbe représentée sont fournies propriétés II.2.3 et II.2.4 (voir [53]).

Propriété II.2.3 (*Insertion nodale à l'ordre k et conservation de la géométrie*)

Soit $\xi = [\xi_1, \xi_2, \cdots, \xi_{n+k}]$ un vecteur nodal et soit $\delta \in [\xi_1, \xi_{n+k}]$ la coordonnée du nœud que l'on souhaite insérer. On note l l'indice tel que $\xi_l \leq \delta < \xi_{l+1}$.

Le nouveau vecteur nodal à $(n+k+1)$ nœuds, $\bar{\xi}$, est tel que :

$$\bar{\xi} = [\xi_1, \xi_2, \cdots, \xi_l, \delta, \xi_{l+1}, \cdots \xi_{n+k}].$$

Soient $\left\{\tilde{P}_1, \cdots, \tilde{P}_n\right\}$ nos points de contrôle initiaux en coordonnées homogènes.
Notre géométrie initiale sera conservée si l'on associe aux $(n+1)$ fonctions de base B-splines d'ordre k issues de $\bar{\xi}$ (équation (II.2.1)) les $(n+1)$ nouveaux points de contrôle homogènes $\left\{\tilde{Q}_1, \cdots, \tilde{Q}_{n+1}\right\}$ calculés tels que :

$$\begin{cases} \tilde{Q}_i = & \tilde{P}_i & \text{si} & 1 & \leq i \leq l-k+1 \\ \tilde{Q}_i = & \frac{\delta-\xi_i}{\xi_{i+k-1}-\xi_i}\tilde{P}_i + \frac{\xi_{i+k-1}-\delta}{\xi_{i+k-1}-\xi_i}\tilde{P}_{i-1} & \text{si} & l-k+2 & \leq i \leq l \\ \tilde{Q}_i = & \tilde{P}_{i-1} & \text{si} & l+1 & \leq i \leq n+1 \end{cases}$$

Propriété II.2.4 (*Élévation d'ordre et conservation de la géométrie*)

On note k notre ordre de travail initial. On souhaite élever l'ordre à l'ordre $(k+1)$.
Soit $\xi = [0_k, 1_k]$ notre vecteur nodal.
Le nouveau vecteur nodal à $(n+1+(k+1))$ nœuds, $\bar{\xi}$, est tel que :

$$\bar{\xi} = [0_{k+1}, 1_{k+1}].$$

Soient $\left\{\tilde{P}_1, \cdots, \tilde{P}_n\right\}$ nos points de contrôle initiaux en coordonnées homogènes.
Notre géométrie initiale sera conservée si l'on associe aux $(n+1)$ fonctions de base B-splines d'ordre $(k+1)$ issues de $\bar{\xi}$ (équation (II.2.1)) les $(n+1)$ nouveaux points de contrôle homogènes $\left\{\tilde{Q}_1, \cdots, \tilde{Q}_{n+1}\right\}$ calculés tels que :

$$\begin{cases} \tilde{Q}_1 = & \tilde{P}_1 & & & \\ \tilde{Q}_i = & \frac{k-i+1}{k}\tilde{P}_i + \frac{i-1}{k}\tilde{P}_{i-1} & \text{si} & 2 & \leq i \leq n \\ \tilde{Q}_{n+1} = & \tilde{P}_n & & & \end{cases}$$

Les deux propriétés précédentes sont des propriétés classiques des B-Splines de dimension 1. Nous les appliquerons donc au vecteur nodal que nous avons choisi d'utiliser et aux B-Splines qu'il génère, les fonctions de Bernstein.

Dans le cas des fonctions rationnelles, les calculs nous permettant d'évaluer les points de contrôle sont appliqués aux coordonnées homogènes des $\mathcal{CP}$. Ainsi, que l'on travaille

avec des fonctions de base $\mathcal{NURBS}$ ou des fonctions de base B-Splines, une unique formule nous permet d'obtenir nos nouveaux points. Pour les $\mathcal{NURBS}$, nos calculs seront précédés d'une étape de passage en coordonnées homogènes, et suivis, une fois nos nouveaux points de contrôle homogènes évalués, du retour aux coordonnées réelles.

2.2.3 Fonctions de base de Bernstein rationnelles multidimensionnelles

Présentons maintenant quelques exemples de fonctions de base de Bernstein rationnelles multidimensionnelles. Par soucis de clarté nous nous restreindrons à des exemples en dimension 2. L'extension à la dimension supérieure est, cependant, directe. Les deux formes élémentaires bidimensionnelles traitées sont le triangle et le quadrangle.

2.2.3.1 Cas du triangle On rappelle les coordonnées barycentriques sur un triangle élémentaire :

$$\begin{aligned} \lambda_1(\xi,\eta) &= (1-\xi-\eta), \\ \lambda_2(\xi,\eta) &= \xi, \\ \lambda_3(\xi,\eta) &= \eta. \end{aligned}$$

Comme nous l'avons expliqué précédemment, les fonctions de base polynomiales d'ordre k sont obtenues en développant le terme : $((1-\xi-\eta)+\xi+\eta)^{k-1}$.
Les $\dfrac{k(k+1)}{2}$ fonctions de base sont donc données par la formule (II.13) de la définition II.2.11.

Définition II.2.11 (*Fonctions de base de Bernstein sur un triangle*)

Soit un triplet $(l_i, m_i, n_i) \in [\![0, k-1]\!]^3$ tel que $l_i + m_i + n_i = k-1$,
alors la $i^{\text{ème}}$ fonction de base de Bernstein est telle que :

$$\varphi_i = \varphi_{l_i m_i n_i}(\xi,\eta) = \frac{(k-1)!}{l_i!\, m_i!\, n_i!}\,(1-\xi-\eta)^{l_i}\,\xi^{m_i}\eta^{n_i}. \tag{II.13}$$

Exemple II.2.7 ***(Fonctions de base d'ordre*** 3***)*** **:**

$$\begin{array}{llllll}
l_1=2, & m_1=0, & n_1=0 & \rightarrow & \varphi_1(\xi,\eta)=\varphi_{200}(\xi,\eta) & = & (1-\xi-\eta)^2, \\
l_2=0, & m_2=2, & n_2=0 & \rightarrow & \varphi_2(\xi,\eta)=\varphi_{020}(\xi,\eta) & = & \xi^2, \\
l_3=0, & m_3=0, & n_3=2 & \rightarrow & \varphi_3(\xi,\eta)=\varphi_{002}(\xi,\eta) & = & \eta^2, \\
l_4=1, & m_4=1, & n_4=0 & \rightarrow & \varphi_4(\xi,\eta)=\varphi_{110}(\xi,\eta) & = & 2\,(1-\xi-\eta)\,\xi, \\
l_5=0, & m_5=1, & n_5=0 & \rightarrow & \varphi_5(\xi,\eta)=\varphi_{011}(\xi,\eta) & = & 2\xi\eta, \\
l_6=1, & m_6=0, & n_6=1 & \rightarrow & \varphi_6(\xi,\eta)=\varphi_{101}(\xi,\eta) & = & 2\,(1-\xi-\eta)\,\eta.
\end{array}$$

Les fonctions de base de Bernstein obtenues nous permettent alors de définir les fonctions de Bernstein rationnelles multidimensionnelles (que, par abus de langage, nous nommerons souvent *fonctions* $\mathcal{NURBS}$) sur le triangle :

Définition II.2.12 (*Fonctions de base de Bernstein rationnelles*)

$$\Psi_i(\xi,\eta)=\frac{W_i\varphi_i(\xi,\eta)}{\sum_{j=1}^{k(k+1)/2} W_j\varphi_j(\xi,\eta)}. \tag{II.14}$$

On peut noter que la remarque II.2.5 reste vraie, les polynômes de Bernstein peuvent être considérés comme des polynômes de Bernstein rationnels dont tous les poids sont identiques.

Vérifions que nos fonctions forment bien une base de notre triangle.

Démonstration : À l'ordre k, un triangle comptera $\frac{k(k+1)}{2}$ degrés de liberté. Comme on a trois coordonnées barycentriques sur notre triangle, le développement multinomial de l'équation (II.9) nous fournira $C_{k+1}^{k-1}=\frac{k(k+1)}{2}$ fonctions. Comme la relation de l'équation (II.12) est valable pour toutes les dimensions, il nous suffit de prouver l'indépendance linéaire des fonctions de Bernstein pour prouver prouver également celles des $\mathcal{NURBS}$ et démontrer que ces deux familles forment des base de notre élément.

On considère les $\varphi_{l_im_in_i}(\xi,\eta)$. On cherche une famille de coefficients $(\alpha_{l_im_in_i})$ tels que $l_i+m_i+n_i=k-1$ et $(l_i,m_i,n_i)\in[\![0,k-1]\!]^3$ et tels que :

$$\sum_{l_im_in_i}\alpha_{l_im_in_i}\,(1-\xi-\eta)^{l_i}\,\xi^{m_i}\eta^{n_i}=0 \qquad \forall(\xi,\eta)\in[0,1]^2.$$

Pour $\xi=0$, $\sum_{n_i=0}^{k-1}\alpha_{(k-1-n_i)n_i}\,(1-\eta)^{k-1-n_i}\,\eta^{n_i}=0$, on reconnait la forme linéaire des fonctions de base de Bernstein. Nous avons montré précédemment qu'elles étaient linéaire-

ment indépendantes. On en déduit que $\alpha_{l_i m_i n_i} = 0$ pour $m_i = 0$. En prenant successivement $\eta = 0$ et $(\xi, \eta) = (\frac{1}{2}, \frac{1}{2})$ et en appliquant la même technique, on montre que les $\alpha_{l_i m_i n_i}$ sont également nuls pour $l_i = 0$ ou $n_i = 0$. Il nous suffit donc de montrer que pour (l_i, m_i, n_i) non nuls :

$$\sum_{\substack{l_i m_i n_i \\ l_i, m_i, n_i \neq 0 \\ l_i + m_i + n_i = k-1}} \alpha_{l_i m_i n_i} \lambda_1^{l_i}(\boldsymbol{\xi}) \lambda_2^{m_i}(\boldsymbol{\xi}) \lambda_3^{n_i}(\boldsymbol{\xi}) = 0, \forall \boldsymbol{\xi} \Rightarrow \alpha_{l_i m_i n_i} = 0, \forall (l_i, m_i, n_i).$$

On considère la proposition $\mathcal{P}(k)$:

Si, $\forall \boldsymbol{\xi}$, $\sum_{\substack{lmn \\ l,m,n \neq 0 \\ l+m+n=k-1}} \alpha_{lmn} \lambda_1^l(\boldsymbol{\xi}) \lambda_2^m(\boldsymbol{\xi}) \lambda_3^n(\boldsymbol{\xi}) = 0$, alors tous les coefficients α_{lmn} sont nuls.

Il est évident que $\mathcal{P}(3)$ est vérifiée. On suppose que $\mathcal{P}(k)$ est vraie.
Comme l, m, et n sont non nuls, à l'ordre $(k+1)$,

$$\sum_{\substack{lmn \\ l,m,n \neq 0 \\ l+m+n=k}} \alpha_{lmn} \lambda_1^l \lambda_2^m \lambda_3^n = (\lambda_1 + \lambda_2 + \lambda_3) \left(\sum_{\substack{lmn \\ l,m,n \neq 0 \\ l+m+n=k-1}} \alpha_{lmn} \lambda_1^l \lambda_2^m \lambda_3^n \right) = 0.$$

$\sum_{i=1}^{3} \lambda_i = 1 \Rightarrow \sum_{\substack{lmn \\ l,m,n \neq 0 \\ l+m+n=k-1}} \alpha_{lmn} \lambda_1^l \lambda_2^m \lambda_3^n = 0$. Par hypothèse de récurrence, tout les coefficients α_{lmn} sont nuls donc $\mathcal{P}(k) \Rightarrow \mathcal{P}(k+1)$ et $\mathcal{P}(k)$ est donc vraie quelque soit k. ■

2.2.3.2 Cas du quadrangle

Appliquons maintenant la définition des fonctions de Bernstein à un élément quadrangulaire. Comme pour le triangle, on utilise les coordonnées barycentriques du quadrangle pour obtenir les k^2 fonctions de base cherchées. On rappelle ces coordonnées barycentriques :

$$\begin{aligned} \lambda_1(\xi, \eta) &= (1-\xi)(1-\eta), & \lambda_3 &= \xi\eta, \\ \lambda_2(\xi, \eta) &= \xi(1-\eta), & \lambda_4 &= (1-\xi)\eta. \end{aligned}$$

Les k^2 fonctions de base sont donc données par la formule (II.15) de la définition II.2.13.

Définition II.2.13 (*Fonctions de base de Bernstein sur un quadrangle*)

Soit un quadruplet $(l_i, m_i, n_i, p_i) \in [\![0, k-1]\!]^4$ tel que $l_i + m_i + n_i + p_i = k-1$,
Si k est paire et $i \leq k$ ou si k est impaire et $i < k$, la $i^{\text{ème}}$ fonction de base de Bernstein est telle que :

$$\begin{aligned} \varphi_i &= \varphi_{l_i m_i n_i p_i}(\xi, \eta) \\ &= \frac{(k-1)!}{l_i!\, m_i!\, n_i!\, p_i} \left((1-\xi)(1-\eta)\right)^{l_i} \left(\xi(1-\eta)\right)^{m_i} \left(\xi\eta\right)^{n_i} \left((1-\xi)\eta\right)^{p_i}. \end{aligned} \tag{II.15}$$

Si k est un nombre impaire, la dernière fonction de base est donné par la formule :

$$\varphi_{k^2} = \varphi_{\frac{k-1}{2}0\frac{k-1}{2}0}(\xi, \eta) + \varphi_{0\frac{k-1}{2}0\frac{k-1}{2}}(\xi, \eta) \tag{II.16}$$

Exemple II.2.8 ***(Fonctions de base d'ordre*** 3***)*** **:** On cherche neuf fonctions de base. Les quatre fonctions de base associées aux sommets principaux du quadrangle seront les fonctions :

$$\begin{aligned} (l_1, m_1, n_1, p_1) = (2,0,0,0) &\rightarrow \varphi_1(\xi, \eta) = \varphi_{2000}(\xi, \eta) = (1-\xi)^2 (1-\eta)^2, \\ (l_2, m_2, n_2, p_2) = (0,2,0,0) &\rightarrow \varphi_2(\xi, \eta) = \varphi_{0200}(\xi, \eta) = \xi^2 (1-\eta)^2, \\ (l_3, m_3, n_3, p_3) = (0,0,2,0) &\rightarrow \varphi_3(\xi, \eta) = \varphi_{0020}(\xi, \eta) = \xi^2 \eta^2, \\ (l_4, m_4, n_4, p_4) = (0,0,0,2) &\rightarrow \varphi_4(\xi, \eta) = \varphi_{0002}(\xi, \eta) = (1-\xi)^2 \eta^2. \end{aligned}$$

Les quatre fonctions de base associées aux milieux des arêtes du quadrangle seront les fonctions :

$$\begin{aligned} (l_1, m_1, n_1, p_1) = (1,1,0,0) &\rightarrow \varphi_5(\xi, \eta) = \varphi_{1100}(\xi, \eta) = 2(1-\xi)\xi(1-\eta)^2, \\ (l_2, m_2, n_2, p_2) = (0,1,1,0) &\rightarrow \varphi_6(\xi, \eta) = \varphi_{0110}(\xi, \eta) = 2\xi^2 (1-\eta)\eta, \\ (l_3, m_3, n_3, p_3) = (0,0,1,1) &\rightarrow \varphi_7(\xi, \eta) = \varphi_{0011}(\xi, \eta) = 2\xi(1-\xi)\eta^2, \\ (l_4, m_4, n_4, p_4) = (1,0,0,1) &\rightarrow \varphi_8(\xi, \eta) = \varphi_{1001}(\xi, \eta) = 2(1-\xi)^2 \eta(1-\eta). \end{aligned}$$

Les deux derniers termes du développement multinomial nous donnent notre dernière fonction de base :

$$\varphi_9(\xi, \eta) = \varphi_{1010}(\xi, \eta) + \varphi_{0101}(\xi, \eta) = 4(1-\xi)\xi(1-\eta)\eta.$$

Remarque II.2.7 (*Fonctions* $\mathcal{NURBS}$ *et fonctions de Bernstein rationnelles*)

Considérons le patch paramétrique bidimensionnel $\Xi = \Xi_1 \times \Xi_2 = [0_k, 1_k] \times [0_k, 1_k]$.
À l'ordre k, les deux vecteurs nodaux générant notre patch contiennent un unique segment, notre patch est donc constitué d'un unique élément, le quadrangle unitaire. Les k^2 fonctions de base B-Splines d'ordre k générées par notre patch (voir définition II.2.2) coïncident alors avec les fonctions de Bernstein associées au quadrangle unitaire. Nous pourrons donc, lorsque nous travaillerons sur des espaces pouvant être représentés par un produit tensoriel, utiliser nos fonctions de base soit sous la forme des fonctions de Bernstein, soit sous la forme des fonctions B-Splines.

La démonstration permettant de prouver que nos fonctions forment une base du quadrangle étant similaire à celles précédemment présentées, nous la laisserons à la charge du lecteur.

2.3 Propriétés des polynômes de Bernstein rationnels et non rationnels

Intéressons-nous maintenant aux propriétés des polynômes rationnels de Bernstein sur leur espace de définition (la forme unitaire de l'espace paramétrique). On note $\boldsymbol{\xi}$ les coordonnées d'un point R de l'élément paramétrique et on note $\varphi_i^k = \varphi_{i_1,i_2,\cdots,i_{n_b}}^k$ les fonctions de base d'ordre k dont l'expression est donnée définition II.2.9. Nos fonctions de base :

- forment une partition de l'unité.
- sont positives ou nulles sur l'élément : $\forall k$, $\forall i$, $\forall \boldsymbol{\xi}$, $\varphi_i^k(\boldsymbol{\xi}) \geq 0$.
- sont à support compact sur l'élément.
- peuvent être définis récursivement :

$$\varphi_{i_1,i_2,\cdots,i_{n_b}} = \lambda_1 \varphi_{i_1-1,i_2,\cdots,i_{n_b}} + \lambda_2 \varphi_{i_1,i_2-1,\cdots,i_{n_b}} + \cdots + \lambda_{n_b} \varphi_{i_1,i_2,\ldots,i_{n_b}-1}.$$

- interpolent les points de contrôle au coin de l'élément paramétrique.
- sont C^{k-1} à l'intérieur de l'élément et au moins continues le long des arêtes.
- atteignent exactement un maximum sur l'élément.
- permettent de représenter exactement les coniques si elles sont rationnelles.

Il est important de noter également que les propriétés des courbes, surfaces et volumes B-Splines et $\mathcal{NURBS}$ restent valables :

- La courbe, la surface ou le volume de Bernstein rationnel sont inclus dans l'enveloppe convexe du polygone de contrôle.
- On a invariance par transformation affine.
- Une géométrie de Bernstein rationnelle de dimension d peut être vue comme la projection de la géométrie de Bernstein non rationnelle de l'hyperespace de dimension $(d+1)$ associée aux coordonnées homogènes (définition II.2.8).
- L'erreur d'interpolation d'une fonction f est majorée par la relation II.8 que l'on rappelle :

$$\|f - I(f)\|_{H^m} \leq C'(f,p)\frac{h_1^k}{h_2^m}, \quad m \leq p,$$

avec h_1 le rayon maximal de l'élément considéré, h_2 son rayon minimal, k l'ordre des fonctions de base $\mathcal{NURBS}$, $\|.\|_{H^m}$ la norme standard de Sobolev, p la régularité de f, I l'opérateur de projection sur l'espace des fonctions de Bernstein rationnelles et C' une constante dépendant de p.

3 Méthode numérique : Schéma $\mathcal{RD}$ isogéométrique

Le rôle de cette section est de présenter le couplage entre le schéma $\mathcal{RD}$ décrit section 2 et les fonctions de base de Bernstein rationnelles précédemment introduites. Il convient de noter que, bien que nous travaillions maintenant dans $\mathbb{Q}_{\mathbb{P}}^k$, ceci ne modifie pas les propriétés du schéma présentées sous-section 2.1 dans le cas d'un espace de fonctions polynomiales. Nous détaillerons d'abord le calcul du terme explicite du système. Nous nous intéresserons ensuite à la partie implicite de notre système. Pour finir nous donnerons deux exemples pertinents de conditions aux limites.

Du fait de notre géométrie courbe, la discrétisation linéaire par morceaux présentée figure I.2.3 de la sous-section 2.1 est modifiée :
les arêtes du domaine peuvent être courbes ou non. On représente un exemple de discrétisation courbe figure II.3.2. Notons que les bords du domaine ne sont pas nécessairement exactement représentés (approximation détaillée chapitre III et plus particulièrement section 2).

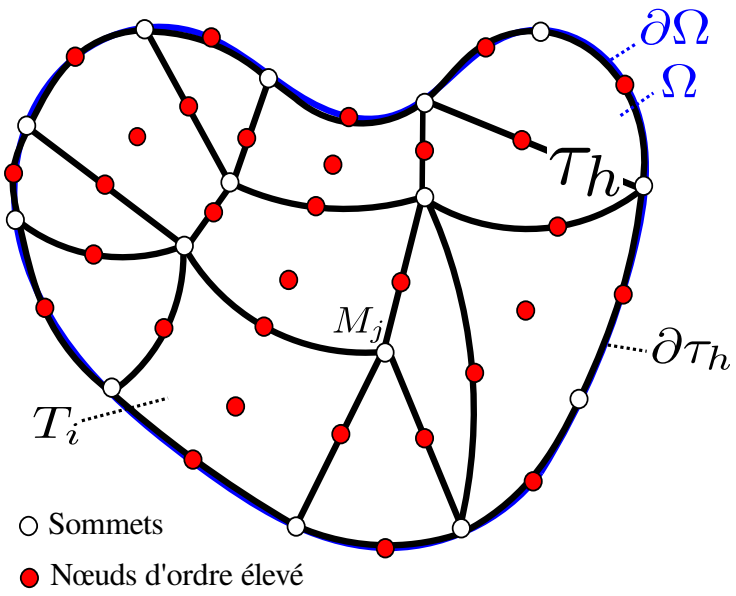


Figure II.3.2 – Discrétisation courbe du domaine Ω à l'ordre 3.

3.1 Calcul du résidu de Lax-Friedrichs limité et stabilisé

Rappelons d'abord que notre problème implicite s'écrit :

$$\mathbf{M}\Delta\boldsymbol{U}^n = -\boldsymbol{\Phi}^n \tag{II.17}$$

Avec $\boldsymbol{\Phi}^n = (\sum_{\mathrm{T}\in\mathcal{D}_i} \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^n))_{i=1,N_{Tot}}$. Il suffit donc de calculer le résidu de Lax-Friedrichs limité et stabilisé associé à chaque élément du voisinage de i pour connaître la $i^{\text{ème}}$ composante du second membre. La construction de ce résidu est détaillée sous-section 2.4 et plus particulièrement équation (I.54). Il est de la forme :

$$\Phi_i^{T,*} = \underbrace{B_i^{\mathrm{T}}\Phi^{\mathrm{T}}}_{(I)} + \underbrace{\theta_{\mathrm{T}} D_i^{\mathrm{T}}}_{(II)}. \tag{II.18}$$

La première étape est donc de calculer le résidu global, Φ^{T}.

3.1.1 Résidu global

Notons n_{edge} le nombre d'arêtes (en dimension 2) ou de faces (en dimension 3) de bord de l'élément considéré (*par ex.* $n_{\text{edge}} = 4$ pour un quadrangle et pour un tétraèdre). Par abus de langage, on parlera d'arêtes de bord, que l'on soit en dimension 2 ou 3. Γ_e désigne la $i_e{}^{\text{ème}}$ arête de l'élément T et $\vec{\boldsymbol{n}}_{i_e}$ la normale unitaire à Γ_e. Ces notations sont reprises

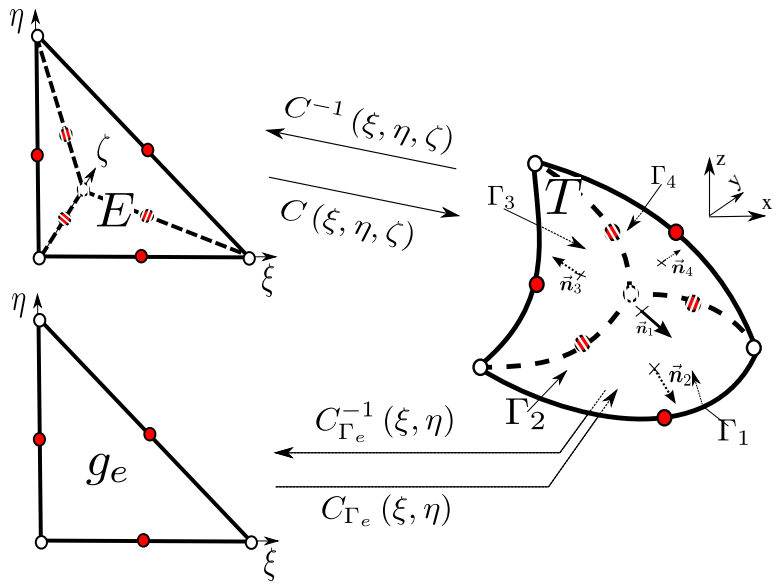


Figure II.3.3 – Tétraèdre courbe d'ordre 3 et mappings associés depuis l'espace paramétrique

à titre d'exemple figure II.3.3 à droite. Φ^{T} est la quantité de fluctuation traversant un élément donc :

$$\Phi^{\mathrm{T}} = \int_{\partial \mathrm{T}} \overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}_h) d\partial \mathrm{T} = \sum_{i_e=1}^{n_{\mathrm{edge}}} \left(\int_{\Gamma_e} \overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}_h).\vec{\boldsymbol{n}}_{i_e} d\boldsymbol{s} \right). \tag{II.19}$$

On note $C^{\mathrm{T}}(\boldsymbol{\xi})$ l'application qui, à l'élément de référence E de l'espace paramétrique, associe l'élément T.

De manière équivalente, on note $C^{\mathrm{T}}_{\Gamma_e}(\boldsymbol{\xi})$ l'application qui, à l'élément de référence de l'hyperplan de l'espace paramétrique, g_e, associe l'arête Γ_e (voir figure II.3.3).

Exemple II.3.1 ***(Notations des applications de transformation)*** **:**

$$\begin{array}{lllllllll} \text{Si} & C^{\mathrm{T}}(\boldsymbol{\xi}) & = & C^{\mathrm{T}}(\xi,\eta), & \text{alors} & C^{\mathrm{T}}_{\Gamma_e}(\boldsymbol{\xi}) & = & C^{\mathrm{T}}_{\Gamma_e}(\xi) \\ \text{Si} & C^{\mathrm{T}}(\boldsymbol{\xi}) & = & C^{\mathrm{T}}(\xi,\eta,\zeta), & \text{alors} & C^{\mathrm{T}}_{\Gamma_e}(\boldsymbol{\xi}) & = & C^{\mathrm{T}}_{\Gamma_e}(\xi,\eta) \end{array}$$

On allègera souvent les notations en omettant l'indice T.

Chaque intégrale est ramenée par changement de variable à une intégrale dans l'espace

paramétrique :

$$\int_{\Gamma_e} \overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}_h).\vec{\boldsymbol{n}}_{i_e} d\boldsymbol{s} = \int_{g_e} \overrightarrow{\boldsymbol{\mathcal{F}}}\Big(\boldsymbol{U}_h\big(C_{\Gamma_e}(\boldsymbol{\xi})\big)\Big).\vec{\boldsymbol{n}}_{i_e}\big(C_{\Gamma_e}(\boldsymbol{\xi})\big)\left|J_{C_{\Gamma_e}}(\boldsymbol{\xi})\right| d\boldsymbol{\xi} \tag{II.20}$$

$\left|J_{C_{\Gamma_e}}\right|$ désigne la jacobienne de C_{Γ_e} dans le cas particulier où $\text{đ} = d$. Dans le cas général :

$$\left|J_{C_{\Gamma_e}}\right| = \begin{cases} \|\partial_\xi \boldsymbol{X}\| & \text{si } \text{đ} = 1 \\ \|\partial_\xi \boldsymbol{X} \wedge \partial_\eta \boldsymbol{X}\| & \text{si } \text{đ} = 2 \end{cases} \tag{II.21}$$

On peut remarquer que ce terme correspond à la norme de la normale non unitaire à l'arête Γ_e donc $\left[\vec{\boldsymbol{n}}_{i_e}\big(C_{\Gamma_e}(\boldsymbol{\xi})\big)\left|J_{C_{\Gamma_e}}(\boldsymbol{\xi})\right|\right]$ est simplement la normale non unitaire à l'arête Γ_e. On la notera $\vec{\boldsymbol{n}}_{\mathrm{n}_{i_e}}$. Comme les éléments sont courbes, ces normales non unitaires ne sont pas constantes et doivent être calculées localement. À la différence des éléments droits, on ne peut donc plus sortir ce terme de l'intégrale. Le terme que l'on intègre équation (II.20) est d'ordre polynomial $\Big((k-1)+\text{đ}\,(k-2)\Big)$ (puisque $\vec{\boldsymbol{n}}_{\mathrm{n}_{i_e}}$ est d'ordre $\text{đ}(k-2)$). Dans le cas d'éléments droits, ce terme est simplement d'ordre $(k-1)$. Cette remarque est importante car on approche les intégrales par une formule de quadrature de Gauss, le nombre de points de quadrature devra donc être soigneusement choisi. On note n_{quad} le nombre de points de quadrature, $\boldsymbol{\xi}_{\mathrm{q}}$ et ω_{q} les coordonnées et poids des points de quadrature dans l'espace paramétrique. L'équation (II.19) peut être réécrite :

$$\Phi^{\mathrm{T}} = \sum_{i_e=1}^{n_{\text{edge}}} \left(\sum_{q=1}^{n_{quad}} \omega_{\mathrm{q}} \left[\overrightarrow{\boldsymbol{\mathcal{F}}}\Big(\boldsymbol{U}_h\Big).\vec{\boldsymbol{n}}_{\mathrm{n}_{i_e}} \right] \right) \tag{II.22}$$

Le calcul de Φ^{T} permet d'évaluer les résidus de LxF d'ordre 1 et, par la procédure détaillée à la fin du paragraphe 2.4.2, de calculer le terme de limitation B_i^{T}.

Étudions le calcul du terme de stabilisation ou terme de filtrage.

3.1.2 Terme de filtrage

On s'intéresse ici au terme (II) de l'équation II.18 : le terme $\theta_{\mathrm{T}} D_i^{\mathrm{T}}$.

Le coefficient d'adimensionalisation θ_{T} ne dépend pas du type d'élément considéré donc nous nous intéresserons seulement au terme de dissipation numérique :

$$D_i^{\mathrm{T}} = \int_{\mathrm{T}} (\vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla\varphi_i})\bar{\boldsymbol{\tau}}(\vec{\boldsymbol{\lambda}}.\overrightarrow{\nabla \boldsymbol{U}_h})\, d\boldsymbol{x} \tag{II.23}$$

Comme pour le résidu global, la différence essentielle entre le calcul de la stabilisation sur

un élément droit et sur un élément courbe réside dans l'évaluation de l'intégrale. Cette fois-ci on intègre sur l'élément complet, et non sur son contour. À nouveau, on calcule notre intégrale dans l'espace paramétrique grâce à un changement de variable :

$$D_i^{\mathrm{T}} = \int_E \left(\vec{\boldsymbol{\lambda}}(\boldsymbol{U}_h).\overrightarrow{\nabla\varphi_i(C(\boldsymbol{\xi}))}\right)\bar{\boldsymbol{\tau}}\left(\vec{\boldsymbol{\lambda}}(\boldsymbol{U}_h).\overrightarrow{\nabla \boldsymbol{U}_h}\right)|J_C(\boldsymbol{\xi})|\, d\boldsymbol{\xi}. \tag{II.24}$$

Comme dans le paragraphe précédent :

$$|J_C| = \begin{cases} \|\partial_\xi \boldsymbol{X} \wedge \partial_\eta \boldsymbol{X}\| & \text{si } đ = 2 \\ \|\partial_\xi \boldsymbol{X} \wedge \partial_\eta \boldsymbol{X} \wedge \partial_\zeta \boldsymbol{X}\| & \text{si } đ = 3 \end{cases} \tag{II.25}$$

ce qui coïncide avec le déterminant de la matrice jacobienne de la transformation C si $đ = d$. Comme $\overrightarrow{\nabla \boldsymbol{U}_h} = \vec{\nabla}\left(\sum_{j=1}^{n_{\mathrm{DoFs}}} \tilde{\boldsymbol{U}}_j \varphi_j\right) = \sum\left(\tilde{\boldsymbol{U}}_j \overrightarrow{\nabla\varphi_j}\right)$, l'équation (II.24) peut être réécrite :

$$D_i^{\mathrm{T}} = \int_E \sum_{j=1}^{n_{\mathrm{DoFs}}} \left(\tilde{\boldsymbol{U}}_j\left(\vec{\boldsymbol{\lambda}}(\boldsymbol{U}_h).\overrightarrow{\nabla\varphi_i(\boldsymbol{\xi})}\right)\bar{\boldsymbol{\tau}}\left(\vec{\boldsymbol{\lambda}}(\boldsymbol{U}_h).\overrightarrow{\nabla\varphi_j(\boldsymbol{\xi})}\right)\right)|J_C(\boldsymbol{\xi})|\, d\boldsymbol{\xi} \tag{II.26}$$

Une fois de plus, le terme de changement de variable, $|J_C(\boldsymbol{\xi})|$ n 'est constant que si l'élément T est droit. On voit que ce terme est d'ordre $đ(k-2)$ pour un élément courbe. $\bar{\boldsymbol{\tau}}$ étant un paramètre d'adimensionalisation, il sera pris constant sur tout l'élément. Les matrices jacobiennes du flux, $\vec{\boldsymbol{\lambda}}$ doivent être évaluées en $\boldsymbol{U}_h(C(\boldsymbol{\xi}))$. Leur forme est fournie annexe 3.1.2. On souhaite donc intégrer un terme d'ordre $(đ+2)(k-2)$. Notre intégrale est évalué par une formule de quadrature de Gauss à n_{quad} points, n_{quad} étant choisi judicieusement. Détaillons notre choix.

Le terme de filtrage jouera son rôle sous 3 conditions :

1. Tous les termes de D_i sont au moins précis à l'ordre des résidus nodaux.
2. La forme bilinéaire D_i doit être positive.
3. La forme bilinéaire D_i doit être définie.

On cherche donc à évaluer le nombre minimum de points de quadratures n_{quad} nécessaires à la vérification de ces trois conditions.

La première des trois conditions citées est satisfaite dès lors que les termes $\overrightarrow{\nabla\varphi_i}$ et $|J_C|$ sont calculés à l'ordre des résidus nodaux.

Pour des poids et points de quadratures strictement positifs, nos fonctions rationnelles vérifient, évidemment, la seconde condition.

Intéressons nous enfin à la dernière des trois conditions que doit remplir le terme de stabilisation et regardons à quelle condition le terme D_i est défini.

On rappelle que le terme $\vec{\boldsymbol{\lambda}}$ est constant et que le paramètre d'adimensionalisation $\bar{\boldsymbol{\tau}}$ est positif.
Comme on choisit des poids de quadratures strictement positifs,

$$\sum_{q=1,n_{\text{quad}}} \omega_{\text{q}} \ \left(\vec{\boldsymbol{\lambda}}\overrightarrow{\nabla \boldsymbol{U}_h}(\boldsymbol{\xi}_q)\right) \bar{\boldsymbol{\tau}} \left(\vec{\boldsymbol{\lambda}}\overrightarrow{\nabla \boldsymbol{U}_h}(\boldsymbol{\xi}_q)\right) = 0 \Longleftrightarrow \vec{\boldsymbol{\lambda}}\overrightarrow{\nabla \boldsymbol{U}_h}(\boldsymbol{\xi}_q) = 0, \forall q \in [\![1, q]\!].$$

Ainsi, la forme bilinéaire D_i est définie si la relation II.27 est vérifiée.

$$\vec{\boldsymbol{\lambda}}\overrightarrow{\nabla \boldsymbol{U}_h}(\boldsymbol{\xi}_q) = 0, \forall q \in [\![1, q]\!] \quad \Longrightarrow \quad \vec{\boldsymbol{\lambda}}\overrightarrow{\nabla \boldsymbol{U}_h} \equiv 0. \tag{II.27}$$

On cherche donc une condition suffisante pour satisfaire II.27.

Plaçons-nous dans le cas d'un élément bidimensionnel triangulaire et notons $(\varphi_i)_{i=1,\frac{k(k+1)}{2}}$ les polynômes de Bernstein d'ordre k et $(\psi_i)_{i=1,\frac{k(k+1)}{2}}$ les fonctions de base de Bernstein rationnelles. On désigne par W la fonction poids telle que :

$$W = \sum_{j}^{\frac{k(k+1)}{2}} W_j \varphi_j.$$

Comme $\psi_i = \dfrac{\varphi_i}{W}$,

$$\overrightarrow{\nabla \psi_i} = \frac{W\overrightarrow{\nabla \varphi_i} - \varphi_i \overrightarrow{\nabla W}}{W^2}.$$

Quelque soit q, $\vec{\boldsymbol{\lambda}}\overrightarrow{\nabla \boldsymbol{U}_h}(\boldsymbol{\xi}_q) = 0$, donc, quelque soit q :

$$\sum_{i=1}^{\frac{k(k+1)}{2}} \tilde{\boldsymbol{U}}_i \left(\vec{\boldsymbol{\lambda}} W(\boldsymbol{\xi}_q)\overrightarrow{\nabla \varphi_i}(\boldsymbol{\xi}_q) - \varphi_i(\boldsymbol{\xi}_q)\overrightarrow{\nabla W}(\boldsymbol{\xi}_q)\right) = 0. \tag{II.28}$$

On peut réécrire II.28 sous la forme :

$$\mathcal{P}(\boldsymbol{\xi}_q) = W(\boldsymbol{\xi}_q) \underbrace{\left(\sum_{i=1}^{\frac{k(k+1)}{2}} \tilde{\boldsymbol{U}}_i \vec{\boldsymbol{\lambda}} \overrightarrow{\nabla \varphi_i}(\boldsymbol{\xi}_q) \right)}_{\substack{(I) \\ \text{ordre } k-1}} - \underbrace{\left(\sum_{i=1}^{\frac{k(k+1)}{2}} \tilde{\boldsymbol{U}}_i \vec{\boldsymbol{\lambda}} \varphi_i(\boldsymbol{\xi}_q) \right)}_{\substack{(II) \\ \text{ordre } k}} \overrightarrow{\nabla W}(\boldsymbol{\xi}_q) = 0. \tag{II.29}$$

Comme la fonction W est strictement positive, chercher sous quelle condition le polynôme $\mathcal{P}(\boldsymbol{\xi}_q)$ est uniformément nul revient à chercher :

- si $W \equiv 1$, sous quelle condition le polynôme (I) d'ordre $k-1$ est uniformément nul. On se réfère dans ce cas à l'article [41].
- si $W \not\equiv 1$, sous quelle condition le polynôme (I) d'ordre $k-1$ et le polynôme (II) d'ordre k sont uniformément nuls. Cette condition est satisfaite pour $\frac{k(k+1)}{2}$ points de quadratures.

Pour conclure, le terme D_i sera dissipatif si :

- Les poids de quadratures ω_{q} sont tous strictement positifs.
- La formule de quadrature est une formule à $\frac{k(k+1)}{2}$ points.

3.2 Matrice implicite

Il nous reste à étudier le calcul de la matrice implicite $\mathbf{M}$ où $\mathbf{M}$ est une matrice par blocs telle que :

$$\begin{cases} \mathbf{M}_{i,i} = \frac{\mathbf{I}}{\omega_i} + \sum_{\mathrm{T} \in \mathcal{D}_i} \dfrac{\partial \Phi_i^T(\boldsymbol{U}_h^n)}{\partial \boldsymbol{U}_i} \\ \mathbf{M}_{i,j} = \sum_{\mathrm{T} \in \mathcal{D}_i \cap \mathcal{D}_j} \dfrac{\partial \Phi_i^{\mathrm{T}}(\boldsymbol{U}_h^n)}{\partial \boldsymbol{U}_j} \quad \text{pour } i \neq j \end{cases} \tag{II.30}$$

On s'intéresse donc au terme de dérivée du résidu local de LxF (équation (I.42)) au nœud M_i par rapport au vecteur des inconnues en M_j :

$$\frac{\partial \Phi_i^T(\boldsymbol{U}_h^n)}{\partial \boldsymbol{U}_j} = \frac{1}{n_{\mathrm{DoFs}}} \left(\underbrace{\frac{\partial \Phi^{\mathrm{T}}}{\partial \boldsymbol{U}_j}}_{(I)} + \alpha^{\mathrm{T}} \underbrace{\frac{\partial}{\partial \boldsymbol{U}_j} \left[\sum_{j \in T} (\boldsymbol{U}_i - \boldsymbol{U}_j) \right]}_{(II)} \right). \tag{II.31}$$

Concentrons-nous tout d'abord sur le terme (I). En calculant le résidu global comme l'intégrale du flux approché $\overrightarrow{\boldsymbol{\mathcal{F}}}_h$ sur le contour de l'élément T, on trouve :

$$\frac{\partial \Phi^{\mathrm{T}}}{\partial \boldsymbol{U}_j} = \frac{\partial}{\partial \boldsymbol{U}_j}\left(\int_{\partial \mathrm{T}} \overrightarrow{\boldsymbol{\mathcal{F}}}_h d\partial \mathrm{T}\right). \tag{II.32}$$

On a déjà montré équation (II.22) que notre intégrale peut-être réécrite dans l'espace paramétrique et approchée par une formule de quadrature :

$$\begin{aligned} \frac{\partial \Phi^{\mathrm{T}}}{\partial \boldsymbol{U}_j} &= \frac{\partial}{\partial \boldsymbol{U}_j} \sum_{i_e=1}^{n_{\mathrm{edge}}} \left(\sum_{q=1}^{n_{quad}} \omega_{\mathrm{q}} \left[\overrightarrow{\boldsymbol{\mathcal{F}}}_h\left(\boldsymbol{\xi}_q\right) . \vec{\boldsymbol{n}}_{\mathrm{n}_{i_e}}\left(\boldsymbol{\xi}_q\right) \right] \right) \\ &= \sum_{i_e=1}^{n_{\mathrm{edge}}} \left(\sum_{q=1}^{n_{quad}} \omega_{\mathrm{q}} \left[\frac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_h\left(\boldsymbol{\xi}_q\right)}{\partial \boldsymbol{U}_j} . \vec{\boldsymbol{n}}_{\mathrm{n}_{i_e}}\left(\boldsymbol{\xi}_q\right) \right] \right). \end{aligned} \tag{II.33}$$

$\boldsymbol{\xi}_q$ et w_q étant, respectivement, les points et poids de quadrature. Le flux approché $\overrightarrow{\boldsymbol{\mathcal{F}}_h}$ est tel que $\overrightarrow{\boldsymbol{\mathcal{F}}}_h\left(\boldsymbol{\xi}_q\right) = \sum_{l=1}^{n_{\mathrm{DoFs}}} \overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}_l \varphi_l\left(\boldsymbol{\xi}_q\right)$, donc :

$$\frac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_h\left(\boldsymbol{\xi}_q\right)}{\partial \boldsymbol{U}_j} = \sum_{l=1}^{n_{\mathrm{DoFs}}} \frac{\partial \overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}_l}{\partial \boldsymbol{U}_j} \varphi_l\left(\boldsymbol{\xi}_q\right).$$

Il nous faut donc calculer, pour chaque DoF l, la dérivée du $l^{\text{ème}}$ flux de contrôle par rapport à la variable $\boldsymbol{U}_j$.

On rappelle que quelque soit le DoF l de coordonnées $\boldsymbol{\xi}_l$, $\overrightarrow{\boldsymbol{\mathcal{F}}}_l = \sum_{j=1}^{n_{\mathrm{DoFs}}} \varphi_j(\boldsymbol{\xi}_l) \overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}_j$.

Soient $\overrightarrow{\boldsymbol{\mathcal{F}}} = {}^T\left(\overrightarrow{\boldsymbol{\mathcal{F}}}_1, \cdots, \overrightarrow{\boldsymbol{\mathcal{F}}}_{n_{\mathrm{DoFs}}}\right)$ le vecteur des valeurs prises par la fonction $\overrightarrow{\boldsymbol{\mathcal{F}}}$ aux DoFs de l'élément T et $\overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}} = {}^T\left(\overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}_1, \cdots, \overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}_{n_{\mathrm{DoFs}}}\right)$ le vecteur des valeurs de contrôle associées à $\overrightarrow{\boldsymbol{\mathcal{F}}}$. On note $\mathcal{B} = (b)_{ij} = (\varphi_j(\boldsymbol{\xi}_i))_{ij}$ la matrice $(n_{\mathrm{DoFs}} \times n_{\mathrm{DoFs}})$ des fonctions de base $\mathcal{NURBS}$ évaluées aux DoFs de T. Quelque soit l'élément T, $\overrightarrow{\boldsymbol{\mathcal{F}}} = \mathcal{B}\overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}$. La matrice $\mathcal{B}$ est inversible donc $\overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}} = \mathcal{B}^{-1}\overrightarrow{\boldsymbol{\mathcal{F}}}$, et donc $\dfrac{\partial \overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}}{\partial \boldsymbol{U}_j} = \mathcal{B}^{-1}\dfrac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}}{\partial \boldsymbol{U}_j}$.

Comme :

$$\frac{\partial \overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}_i}{\partial \boldsymbol{U}_j} = \begin{cases} \vec{\boldsymbol{\lambda}}(\boldsymbol{U}_j), & \text{si } i = j \\ 0, & \text{si } i \neq j \end{cases},$$

alors :

$$\frac{\partial \overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}}{\partial \boldsymbol{U}_j} = {}^T\left(b_{1j}^{-1}, \cdots, b_{n_{\mathrm{DoFs}}j}^{-1}\right) \vec{\boldsymbol{\lambda}}(\boldsymbol{U}_j).$$

Et donc :

$$\frac{\partial \overrightarrow{\mathcal{F}}_h\left(\boldsymbol{\xi}_q\right)}{\partial \boldsymbol{U}_j} = \left(\sum_{l=1}^{n_{\text{DoFs}}} b_{lj}^{-1}\varphi_l\left(\boldsymbol{\xi}_q\right)\right)\vec{\boldsymbol{\lambda}}(\boldsymbol{U}_j). \tag{II.34}$$

Finalement, le terme (I) de l'équation (II.31) est approché par la relation :

$$\frac{\partial \Phi^{\mathrm{T}}}{\partial \boldsymbol{U}_j} = \sum_{i_e=1}^{n_{\text{edge}}}\left(\sum_{q=1}^{n_{quad}}\left[\omega_{\mathrm{q}}\sum_{l=1}^{n_{\text{DoFs}}}\left(b_{lj}^{-1}\varphi_l\left(\boldsymbol{\xi}_q\right)\right)\vec{\boldsymbol{n}}_{\mathrm{n}_{i_e}}\left(\boldsymbol{\xi}_q\right)\right]\right)\vec{\boldsymbol{\lambda}}(\boldsymbol{U}_j). \tag{II.35}$$

Considérons maintenant le terme (II). $\dfrac{\partial \boldsymbol{U}_i}{\partial \boldsymbol{U}_j}$ est une matrice de taille $(n_{\text{var}} \times n_{\text{var}})$ telle que $\dfrac{\partial \boldsymbol{U}_i}{\partial \boldsymbol{U}_j} = \text{diag}(\delta_{ij})$ avec δ_{ij} le symbole de Kronecker. On a donc :

$$\frac{\partial}{\partial \boldsymbol{U}_j}\left[\sum_{j\in T}(\boldsymbol{U}_i - \boldsymbol{U}_j)\right] = \begin{cases} \text{diag}(n_{\text{DoFs}}-1), & \text{si } i = j \\ \text{diag}(-1), & \text{si } i \neq j \end{cases} \tag{II.36}$$

Pour conclure, les blocs complets de notre matrice implicite sont de la forme :

$$\begin{cases} \mathbf{M}_{i,i} = & \text{diag}\Big(\frac{\mathbf{I}}{\omega_i} + \displaystyle\sum_{\mathrm{T}\in\mathcal{D}_i}\frac{n_{\text{DoFs}}-1}{n_{\text{DoFs}}}\Big) + \\ & \displaystyle\sum_{\mathrm{T}\in\mathcal{D}_i}\left(\frac{1}{n_{\text{DoFs}}}\sum_{i_e=1}^{n_{\text{edge}}}\sum_{q=1}^{n_{quad}}\omega_{\mathrm{q}}\sum_{l=1}^{n_{\text{DoFs}}}\left(b_{li}^{-1}\varphi_l\left(\boldsymbol{\xi}_q\right)\right)\vec{\boldsymbol{n}}_{\mathrm{n}_{i_e}}\left(\boldsymbol{\xi}_q\right)\vec{\boldsymbol{\lambda}}(\boldsymbol{U}_i)\right) \\ \mathbf{M}_{i,j} = & \text{diag}\Big(-\displaystyle\sum_{\mathrm{T}\in\mathcal{D}_i}\frac{1}{n_{\text{DoFs}}}\Big) + \\ & \displaystyle\sum_{\mathrm{T}\in\mathcal{D}_i\cap\mathcal{D}_j}\left(\frac{1}{n_{\text{DoFs}}}\sum_{i_e=1}^{n_{\text{edge}}}\sum_{q=1}^{n_{quad}}\omega_{\mathrm{q}}\sum_{l=1}^{n_{\text{DoFs}}}\left(b_{lj}^{-1}\varphi_l\left(\boldsymbol{\xi}_q\right)\right)\vec{\boldsymbol{n}}_{\mathrm{n}_{i_e}}\left(\boldsymbol{\xi}_q\right)\vec{\boldsymbol{\lambda}}(\boldsymbol{U}_j)\right) \end{cases} \tag{II.37}$$

3.3 Conditions aux limites

Comme il existe de nombreux types de conditions aux bords, nous donnerons ici simplement deux exemples de $\mathcal{BC}$ typiques. Dans ces exemples, on suppose que la condition est imposée sur la frontière $\Gamma = \partial\mathrm{T} \cap \partial\Omega$.

3.3.1 Condition d'entrée de fluide : $\mathcal{BC}$ de type Dirichlet imposée fortement

Cette condition au bord impose la vitesse du fluide à la frontière : $\vec{\boldsymbol{u}}(x,y)_{|\Gamma} = g(x,y)$, avec g une fonction connue. Pour tout nœud de M_i de Γ et pour tout temps t^n, la valeur de la vitesse, $\vec{\boldsymbol{u}}_i^n = g(M_i)$, est donc donnée. Afin d'avoir un problème bien posé, il nous

faut également imposer la densité et la pression.

Comme les valeurs nodales de toutes les composantes de l'inconnue $\boldsymbol{U}_i^n$ sont constantes en temps le long de Γ, on sait que $\Delta \boldsymbol{U}_i^n = 0$. Pour le second membre de notre système (II.18), ceci ce traduit par un résidu nodal Φ_i^{T} nul quelque soit le nœud frontière M_i et pour tout T appartenant au voisinage de M_i. On pose donc :

$$\Phi_i^{\mathrm{T}} = 0 \ \forall T \in D_i. \tag{II.38}$$

Comme :

$$U_i^{n+1} = \boldsymbol{U}_i^n = g(M_i), \tag{II.39}$$

$\boldsymbol{U}_i^{n+1}$ est déjà connu. Par conséquent, le $i^{\text{ème}}$ bloc de la diagonale de la matrice $\mathbf{M}$ est remplacé par la matrice identité et tous les autres blocs de la $i^{\text{ème}}$ ligne sont remplacés par des matrices nulles :

$$\begin{cases} \mathbf{M}_{i,i} &= \mathbf{I}_{(d+2)^2} \\ \mathbf{M}_{i,j} &= \mathbf{0}_{(d+2)^2} \quad \text{pour } i \neq j \end{cases} \tag{II.40}$$

3.3.2 Condition de paroi glissante imposée faiblement

On souhaite imposer une condition de paroi glissante. Cette condition aux bords se traduit par l'annulation de la composante normale à la paroi de la vitesse :

$$\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}} = 0 \text{ sur } \Gamma. \tag{II.41}$$

On impose faiblement les $\mathcal{BC}$ de type « paroi glissante ».
On définit le flux eulérien à la paroi :

$$\overrightarrow{\boldsymbol{\mathcal{F}}}_w = \overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}).\vec{\boldsymbol{n}} = \begin{pmatrix} \rho\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}} \\ (\rho\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}})\, u_1 + p n_1 \\ (\rho\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}})\, u_2 + p n_2 \\ (\rho\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}) \left(E + \frac{p}{\rho}\right) \end{pmatrix} = \begin{pmatrix} 0 \\ p n_1 \\ p n_2 \\ 0 \end{pmatrix}.$$

On voit donc que pour un DoF M_i appartenant à Γ, il nous faut ajouter au résidu nodal explicite Φ_i^{T} un terme additionnel $\Phi_{b,i,\Gamma}$ tel que :

$$\Phi_{b,i,\Gamma} = \int_\Gamma \varphi_i^T \overrightarrow{\boldsymbol{\mathcal{F}}}_{\text{slip}}(\boldsymbol{U}_h, \vec{\boldsymbol{n}}) ds, \tag{II.42}$$

avec :

$$\overrightarrow{\boldsymbol{\mathcal{F}}}_{\text{slip}}(\boldsymbol{U}_h,\vec{\boldsymbol{n}}) = \left(\overrightarrow{\boldsymbol{\mathcal{F}}}_w - \overrightarrow{\boldsymbol{\mathcal{F}}}(\boldsymbol{U}_h)\right).\vec{\boldsymbol{n}} = \begin{pmatrix} -\rho\vec{\boldsymbol{u}} \\ -\rho u_1\vec{\boldsymbol{u}} \\ -\rho u_2\vec{\boldsymbol{u}} \\ -\rho\left(E+p/\rho\right)\vec{\boldsymbol{u}} \end{pmatrix}.\vec{\boldsymbol{n}}. \tag{II.43}$$

Notre arête de bord étant une arête courbe, il faut à nouveau porter une attention particulière au calcul de la normale et à l'évaluation de notre intégrale. L'intégrale de $\Phi_{b,i,\Gamma}$ est approchée, après changement de variable pour se placer dans l'espace paramétrique, par une formule de quadrature d'ordre $(đ(k-2)+2(k-1))$:

$$\Phi_{b,i,\Gamma} = \sum_{q=1}^{n_{quad}} \left(\omega_{\text{q}}\varphi_i^T(\boldsymbol{\xi}_q)\overrightarrow{\boldsymbol{\mathcal{F}}}_{\text{slip}}(\boldsymbol{U}_h(\boldsymbol{\xi}_q),\vec{\boldsymbol{n}}_{\text{n}}(\boldsymbol{\xi}_q))\right). \tag{II.44}$$

Comme dans le paragraphe 3.1.1, $\vec{\boldsymbol{n}}_{\text{n}}(\boldsymbol{\xi}_q)$ est la normale non unitaire à Γ en $C_{\Gamma_e}^{-1}(\boldsymbol{\xi}_q)$ telle que $\vec{\boldsymbol{n}}_{\text{n}}(\boldsymbol{\xi}_q) = \|\Gamma\|\vec{\boldsymbol{n}}(\boldsymbol{\xi}_q)$. La condition de paroi glissante doit également être imposée dans la matrice implicite du schéma. Il faut donc ajouter à la matrice $\mathbf{M}_{i,j}$ la matrice associée au flux frontière du DoF M_i, $J_{b,ij} = \dfrac{\partial\Phi_{b,i,\Gamma}}{\partial\boldsymbol{U}_j}$. Comme sous-section 3.2 lors du calcul de la matrice implicite, on approche le flux $\overrightarrow{\boldsymbol{\mathcal{F}}}_{\text{slip}}(\boldsymbol{U}(\boldsymbol{\xi}_q),\vec{\boldsymbol{n}}_{\text{n}}(\boldsymbol{\xi}_q))$ comme l'extrapolée des flux de contrôle $(\overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}_{\text{slip},l})_{l=1,n_{\text{DoFs}}}$ sur les fonctions de base $\mathcal{NURBS}$ $(\varphi_l)_{l=1,n_{\text{DoFs}}}$ évaluées en $\boldsymbol{\xi}_q$ donc :

$$\begin{aligned} J_{b,ij} &= \sum_{q=1}^{n_{quad}} \left[\omega_{\text{q}}\varphi_i^T(\boldsymbol{\xi}_q)\sum_{l=1}^{n_{\text{DoFs}}}\frac{\partial\overrightarrow{\tilde{\boldsymbol{\mathcal{F}}}}_{\text{slip},l}}{\partial\boldsymbol{U}_j}\varphi_l(\boldsymbol{\xi}_q)\right] \\ &= \sum_{q=1}^{n_{quad}} \left[\omega_{\text{q}}\varphi_i^T(\boldsymbol{\xi}_q)\sum_{l=1}^{n_{\text{DoFs}}}\left(b_{lj}^{-1}\varphi_l(\boldsymbol{\xi}_q)\right)\right]\frac{\partial\overrightarrow{\boldsymbol{\mathcal{F}}}_{\text{slip},j}}{\partial\boldsymbol{U}_j} \\ &= \sum_{q=1}^{n_{quad}} \left[\omega_{\text{q}}\varphi_i^T(\boldsymbol{\xi}_q)\sum_{l=1}^{n_{\text{DoFs}}}\left(b_{lj}^{-1}\varphi_l(\boldsymbol{\xi}_q)\right)\right]\vec{\boldsymbol{\lambda}}_{\text{s}}(\boldsymbol{U}_j), \end{aligned} \tag{II.45}$$

où $\vec{\boldsymbol{\lambda}}_{\text{s}}(\boldsymbol{U}_j)$ est la matrice jacobienne associée au flux de paroi glissante en $\boldsymbol{U}_j$:
Notons $\gamma_1 = \gamma - 1$ et $\mathcal{H} = E + \frac{p}{\rho}$ ($\mathcal{H}$ est l'enthalpie par unité de volume),

$$\vec{\boldsymbol{\lambda}}_{\text{s}}(\boldsymbol{U}_j) = \begin{pmatrix} 0 & n_{\text{n}1} & n_{\text{n}2} & 0 \\ -u_1\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}_{\text{n}} & 2u_1n_{\text{n}1}+u_2n_{\text{n}2} & u_1n_{\text{n}2} & 0 \\ -u_2\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}_{\text{n}} & u_2n_{\text{n}1} & 2u_2n_{\text{n}1}+u_1n_{\text{n}2} & 0 \\ -\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}_{\text{n}}\left(\mathcal{H}-\gamma_1\|\vec{\boldsymbol{u}}\|\right) & \mathcal{H}n_1-\gamma_1u_1\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}_{\text{n}} & \mathcal{H}n_{\text{n}2}-\gamma_1u_2\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}_{\text{n}} & \gamma_1\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}_{\text{n}} \end{pmatrix}, \tag{II.46}$$

où les valeurs de $(\rho, \vec{\boldsymbol{u}}, E)$ et des normales sont calculés au DoF M_j.

Finalement, on impose les conditions aux bords de type paroi glissante :

- en ajoutant le terme $\Phi_{b,i,\Gamma}$ au résidu nodal explicite Φ_i^{T}.
- en ajoutant la matrice additionnelle $J_{b,ij}$ à $\mathbf{M_{ij}}$.

Ceci conclut cette section consacrée à la présentation de la résolution des équations d'Euler par un schéma aux résidus distribué isogéométrique. Nous ne présenterons pas l'adaptation de la résolution des équations de Navier-Stokes à ce même schéma : le travail réalisé est similaire à celui que nous avons présenté ici.

Chapitre III

Génération de maillages courbes d'ordre 3

L'utilisation effective du schéma précédemment décrit nécessite d'avoir une discrétisation courbe de notre domaine de calcul. La création de maillages conformes et non structurés représentant exactement la géométrie d'un problème reste, à ce jour, l'un des principaux challenges de l'analyse isogéométrique. Intuitivement, nous souhaiterions utiliser un maillage issu de la C.A.O. (Conception Assistée par Ordinateur) [1]. En effet, ces maillages représentent la géométrie exacte du domaine. Utilisés dans le cadre de l'ingénierie, certaines de leurs caractéristiques deviennent cependant problématiques :

- Les maillages sont constitués de patchs qui ne sont pas nécessairement du même ordre.
- Ces patchs sont généralement d'ordre beaucoup plus élevé que celui nécessaire pour le calcul numérique et il est souvent impossible de diminuer l'ordre d'un patch tout en conservant la même géométrie.
- Un patch peut recouvrir un autre patch.
- Le maillage peut contenir des trous à l'interface de deux patchs.

En conséquence, les maillages issus de la C.A.O. sont généralement non conformes. Comme certaines notions nécessaires à l'utilisation des schémas $\mathcal{RD}$ ne sont pas encore clairement définies pour des maillages non-conformes (par exemple la définition du voisinage d'un sommet ou celle de volume dual), ce type de schéma n'est, pour l'instant, pas développé sur de telles discrétisations. Il nous faut donc trouver une méthode pour approcher notre domaine et générer des maillages isogéométriques. La solution envisagée, que nous détaillerons dans ce chapitre, consiste à courber un maillage *classique* linéaire par morceaux de notre domaine afin d'approcher au mieux ses frontières. Il est important de noter que,

1. C.A.O. [54] : La conception assistée par ordinateur comprend l'ensemble des logiciels et des techniques de modélisation géométrique permettant de concevoir, de tester virtuellement à l'aide d'un ordinateur et des techniques de simulation numérique et de réaliser des produits manufacturés et les outils pour les fabriquer.

dans ce cas, nous obtenons uniquement une approximation de la géométrie. La géométrie créée nous permettra, par contre, de faciliter l'adaptation de maillage.

1 Pré-requis

Comme nous l'avons expliqué en introduction, notre maillage courbe sera généré à partir d'un maillage linéaire par morceaux, donc d'ordre 2. En effet, il existe de nombreux générateurs très performants de maillages conformes d'ordre 2, et ce, quel que soit le type de discrétisation recherché (discrétisation structurée ou non structurée, hybride ou non). Nous pourrons donc toujours accéder à ce type de maillages. Nous ne détaillerons pas, dans ce travail, l'étape classique de création du maillage linéaire par morceaux et nous considérerons tout simplement que nous avons ce maillage en main. Notre frontière est donc une courbe linéaire par morceaux et notre maillage contient des éléments droits.

Définition III.1.1 (*Maillage* $P1$ *ou droit*)

Nous désignerons souvent par les termes de *maillages* $P1$ et *maillages droits* les maillages linéaires par morceaux d'ordre 2.

Définition III.1.2 (*Points* $P1$)

Quel que soit l'ordre considéré, les sommets des éléments de notre maillage seront souvent nommés *points* $P1$. En effet, ils correspondent aux DoF*s* d'un maillage $P1$.

Le maillage initial doit satisfaire un unique critère, celui de représenter suffisamment bien notre frontière. Ceci signifie simplement qu'il doit être suffisamment fin pour représenter les principales variations de courbures de la géométrie réelle.

Exemple III.1.1 ***(Mauvaise approximation linéaire)*** **:** On fournit, figure III.1.2, un exemple de mauvaise approximation linéaire : les changements de courbures de la frontière sont perdus durant l'étape de discrétisation $P1$.

Tous les maillages linéaires utilisés par d'autres méthodes numériques satisfont, évidemment, ce pré-requis et pourront donc être utilisés comme précurseurs de nos maillages courbes.

Figure III.1.2 – Frontière courbe (en bleu) et maillages $P1$ associés (en noir) : maillage trop large (à gauche) et maillage suffisamment fin (à droite)

2 Interpolation des nœuds frontières par une courbe $\mathcal{NURBS}$

2.1 Approximation de la géométrie

Nous voulons approcher les frontières de notre domaine par des courbes $\mathcal{NURBS}$. On suppose que l'on ne connait pas les équations paramétriques des bords de notre domaine et que nous avons seulement en main la discrétisation $P1$ de ce dernier. Comme tous les nœuds de bord de notre maillage droit appartiennent à la géométrie réelle de notre frontière, on cherche une courbe interpolant chacun de ces points.

Définition III.2.1 (*Vecteur tangent*)

On considère une arête P_iP_{i+1}. Le vecteur tangent à cette arête, $\vec{\mathbf{t}}_i$, est défini comme le vecteur unitaire de sens et direction $P_i\vec{P}_{i+1}$:

$$\vec{\mathbf{t}}_i = \frac{P_{i+1} - P_i}{|P_{i+1} - P_i|}.$$

Définition III.2.2 (*Courbure*)

Soit $\kappa(P_i)$ la courbure en un point P_i d'une courbe paramétrée $\Gamma = C(\xi)$. Notons $\vec{\mathbf{t}}_P$ la tangente locale en un point P et $\angle\vec{\mathbf{t}}_{P_i}\vec{\mathbf{t}}_{P_j}$ l'angle formé par les tangentes locales en P_i et P_j et $l(P_iP_j)$ la longueur de la courbe entre les points P_i et P_j.

On définit la courbure $\kappa(P_i)$ par :

$$\kappa(P_i) = \lim_{P_j \to P_i} \left(\frac{\angle\vec{\mathbf{t}}_{P_i}\vec{\mathbf{t}}_{P_j}}{l(P_iP_j)} \right)$$

Définition III.2.3 (*Cercle osculateur*)

Un cercle $\mathscr{C}$ est dit *osculateur* à une courbe Γ en un point P si sa tangente $\vec{\mathbf{t}}_{\mathscr{C}P}$ et sa courbure $\kappa_{\mathscr{C}}(P)$ en P sont identiques à la tangente $\vec{\mathbf{t}}_{\Gamma P}$ et à la courbure $\kappa_\Gamma(P)$ de Γ en P. Le rayon r_P du cercle osculateur est tel que $r_P = \kappa(P)^{-1}$. Son centre C_P est tel que $C_P = P + r_P \times \vec{\boldsymbol{n}}_P$, où $\vec{\boldsymbol{n}}_P$ est la normale unitaire à Γ en P.

Propriété III.2.1 (*Approximation locale d'une courbe par le cercle osculateur*)

Le cercle osculateur en un point d'une courbe approche localement cette courbe à l'ordre 2 [55].

Nous souhaitons approcher localement notre frontière par un cercle osculateur. En effet, d'après la proposition III.2.1, une telle approximation est d'ordre 2. De plus, les fonctions de base $\mathcal{NURBS}$ permettent de représenter exactement les coniques. Ainsi, nous devons calculer ce cercle en tout point de bord de notre domaine. Comme nous ne connaissons que des valeurs ponctuelles de la courbe Γ, la définition III.2.2 ne peut pas être utilisée. On définit donc la *courbure discrète ponctuelle*.

Définition III.2.4 (*Courbure discrète ponctuelle*)

Notons $\angle\vec{\mathbf{t}}_{i-1}\vec{\mathbf{t}}_i$ l'angle entre $\vec{\mathbf{t}}_i$ et $\vec{\mathbf{t}}_{i-1}$. La courbure discrète ponctuelle en un point P_i, κ_i, est définie comme le rapport entre le défaut angulaire des segments d'extrémités P_i et la moyenne de leurs longueurs ([56]) :

$$\kappa_i = \frac{\angle\vec{\mathbf{t}}_{i-1}\vec{\mathbf{t}}_i}{\frac{|P_{i+1}-P_i|}{2} + \frac{|P_i-P_{i-1}|}{2}}$$

Considérons une arête de bord Γ d'extrémités P_i, P_j et notons $\bar{\kappa} = \frac{\kappa_i+\kappa_j}{2}$ la moyenne des courbures aux extrémités de Γ. Nous approcherons Γ par l'arc de cercle de courbure $\bar{\kappa}$ passant par P_i et P_j. Comme nos arêtes sont approchées par des portions de cercles, nous parlerons d'*approximation circulaire*.

Il est important de noter que les propriétés de continuité de notre approximation ne sont pas modifiées et que notre courbe reste définie par morceaux, même si ce n'est plus linéairement.

Dans la pratique, l'interpolation circulaire d'une arête de bord Γ, est obtenue par un algorithme à quatre étapes :

- On trouve la courbure en l'extrémité P_i de Γ.
- On trouve la courbure en l'extrémité P_j de Γ.

- On calcule la courbure moyenne sur l'arête $\bar{\kappa}$.
- On trouve le rayon et le centre du cercle de courbure $\bar{\kappa}$ permettant d'interpoler P_i et P_j.

Remarque III.2.1 (*Arêtes droites*)

La géométrie pouvant contenir des arêtes droites, on permet la représentation de coins et de segments linéaires en marquant les nœuds concernés.

Maintenant que nous avons totalement défini la courbe approchant chacune de nos arêtes, nous pouvons calculer les poids et les points de contrôle nécessaires à la représentation de chaque arête par une courbe $\mathcal{NURBS}$.

2.2 Représentation d'un arc de cercle par une courbe $\mathcal{NURBS}$

Puisque nous pourrons toujours, *a posteriori*, élever l'ordre de notre courbe, nous cherchons une $\mathcal{NURBS}$ d'ordre le plus faible possible. Notre arête étant une portion de cercle, l'ordre minimal permettant de la représenter est l'ordre 3. On peut noter que, connaissant le centre et le rayon du cercle $\mathscr{C}$, nous connaissons les coordonnées de notre arête courbe en tout point. Comme nous l'avons expliqué sous-section 2.2, à l'ordre 3, notre vecteur nodal est le vecteur ouvert $\Xi = [0,0,0,1,1,1]$. Soient δ_i nos degrés de liberté dans l'espace paramétrique. On équidistribue nos trois DoF*s* sur le segment paramétrique que représente Ξ donc $\delta_1 = 0$, $\delta_2 = \frac{1}{2}$ et $\delta_3 = 1$. Notons $\tilde{P}_i$ et W_i le $i^{\text{ème}}$ point et le $i^{\text{ème}}$ poids de contrôle associés à notre courbe et notons M_i la $i^{\text{ème}}$ fonction de base $\mathcal{NURBS}$ d'ordre 3 associée à notre vecteur nodal. Notre courbe $\mathscr{C}$ est paramétrée par l'équation :

$$\mathscr{C}(\xi) \quad = \quad \tilde{P}_1 M_1(\xi) + \tilde{P}_2 M_2(\xi) + \tilde{P}_3 M_3(\xi). \tag{III.1}$$

On note N_i la $i^{\text{ème}}$ fonction de base B-Spline d'ordre 3 associée à Ξ et on réécrit notre paramétrisation en fonction des fonctions de base B-Splines (voir équation (II.2.6)) :

$$\mathscr{C}(\xi) \quad = \quad \frac{\tilde{P}_1 W_1 N_1(\xi) + \tilde{P}_2 W_2 N_2(\xi) + \tilde{P}_3 W_3 N_3(\xi)}{W_1 N_1(\xi) + W_2 N_2(\xi) + W_3 N_3(\xi)}. \tag{III.2}$$

Soient $P_i = \mathscr{C}(\delta_i)$ nos degrés de liberté dans l'espace réel. Comme $P_1 = \mathscr{C}(0)$ et $P_3 = \mathscr{C}(1)$, alors :

$$\begin{cases} P_1 = \mathscr{C}(0) = \sum_{i=1}^{3} \tilde{P}_i M_i(0) = \tilde{P}_1 \\ P_3 = \mathscr{C}(1) = \sum_{i=1}^{3} \tilde{P}_i M_i(1) = \tilde{P}_3 \end{cases}$$

On sait que si l'ensemble de poids $\{W_1, W_2, W_3\}$ est solution de notre problème de représentation, alors quelque soit $c \in \mathbb{R}$, l'ensemble $\{cW_1, cW_2, cW_3\}$ est également solution. De plus, comme on cherche un arc de cercle, on peut poser $W_1 = W_3$ et donc imposer $W_1 = W_3 = 1$. Comme les poids W_1 et W_3 sont les poids associés aux deux extrémités de notre arête, l'hypothèse précédente nous garantit l'unicité du poids associé à un point partagé par deux arêtes. De plus, comme $\tilde{P}_1$ et $\tilde{P}_3$ sont des points de contrôle interpolés, nous aurons également unicité des points de contrôle partagés par deux arêtes. Comme $\tilde{P}_1 = P_1$, $\tilde{P}_3 = P_3$ et comme $W_1 = W_3 = 1$, on peut réécrire l'équation (III.2) sous la forme :

$$\mathscr{C}(\xi) = \frac{P_1 (1-\xi)^2 + 2\tilde{P}_2 W_2 \xi (1-\xi) + P_3 \xi^2}{(1-\xi)^2 + 2W_2 \xi (1-\xi) + \xi^2}. \tag{III.3}$$

Il nous suffit donc de trouver le point de contrôle $\tilde{P}_2$ et le poids associé, W_2, tels que notre courbe $\mathcal{NURBS}$ soit égale en tout point à notre arc de cercle.

Soit Θ l'angle au centre associé à notre arc, on suppose que notre point $P_2 = \mathscr{C}(\frac{1}{2})$ est le point du cercle en $\frac{\Theta}{2}$. Ceci revient simplement à supposer que, lors du parcours de notre intervalle paramétrique $[0,1]$, on parcourt notre arc de cercle à vitesse constante. On obtient une équation à deux inconnues :

$$P_2 = \frac{P_1 + P_3 + 2W_2 \tilde{P}_2}{2 + 2W_2}. \tag{III.4}$$

Rappelons que notre courbe $\mathcal{NURBS}$ est tangente en ses extrémités à son polygone de contrôle (voir paragraphe 2.1.3). $\tilde{P}_2$ est donc le point d'intersection des tangentes aux extrémités de Γ à notre arc de cercle (voir figure III.2.2). Un fois $\tilde{P}_2$ évalué, le calcul de W_2 est trivial.

2.3 Algorithme de génération d'une géométrie courbe

Il nous reste à définir comment calculer les arêtes internes du maillage.

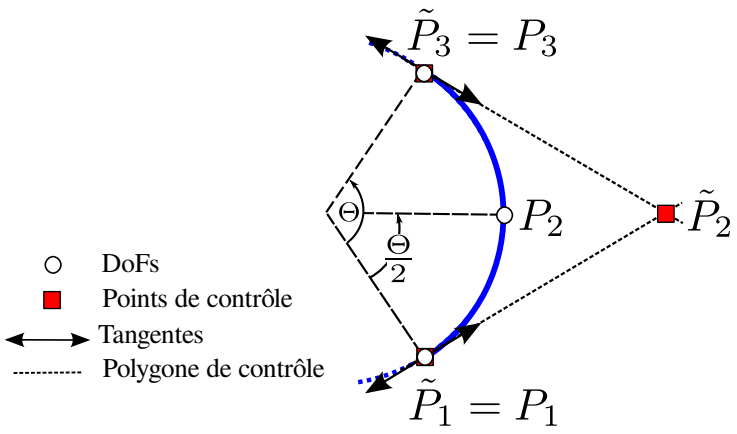


Figure III.2.2 – Représentation $\mathcal{NURBS}$ d'un arc de cercle

Comme pour les poids associés aux extrémités de nos arêtes courbes, on peut poser que tous les poids associés aux points $P1$ internes au domaine sont égaux à 1. A priori, les arêtes n'appartenant pas à la frontière sont droites. Ainsi, en supposant que l'on parcourt toujours notre arête à vitesse constante, le second degré de liberté, P_2, est au milieu de l'arête, d'où $P_2 = \frac{P_1+P_3}{2}$. On déduit donc de l'équation (III.4) que $\tilde{P}_2 = P_2$. Notre maillage courbe d'ordre 3 est maintenant totalement défini.

Pour conclure, un algorithme à quatre étapes nous permet d'évaluer les poids et points de contrôle nécessaires à la représentation de notre géométrie.

Algorithme III.2.1: Génération d'une géométrie courbe

- Tri des arêtes : Chaque frontière fermée est repérée et on ordonne ses arêtes de bord.
- Calcul de la courbure discrète de chaque nœud frontière.
- Pour chaque arête de bord : Calcul du cercle de courbure moyenne et interpolant les extrémités de l'arête puis calcul de la $\mathcal{NURBS}$ permettant de représenter le cercle trouvé à l'étape précédente entre les extrémités de l'arête.
- Pour chaque arête interne : Calcul de la $\mathcal{NURBS}$ associée à l'arête.

La figure III.2.3 donne un exemple de discrétisation $P1$. Le maillage courbe obtenu par interpolation circulaire des nœuds de bord $P1$ est fourni figure III.2.4.

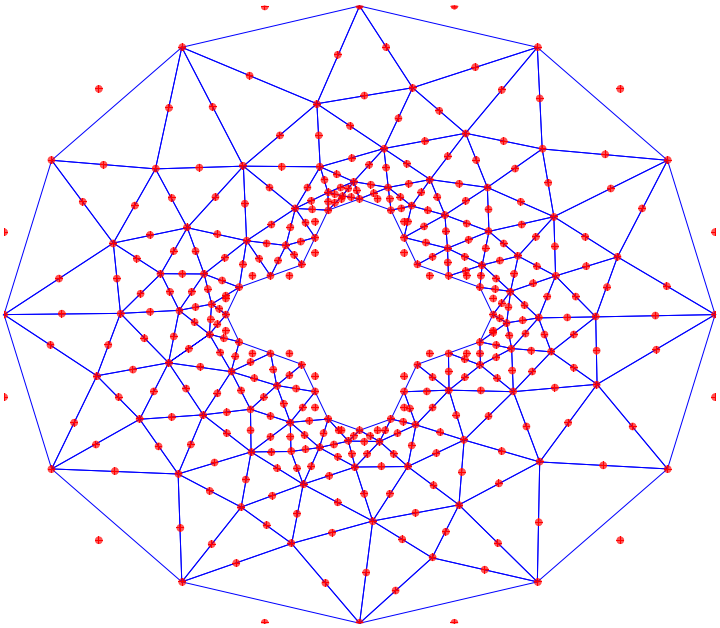


Figure III.2.3 – Discrétisation linéaire d'un domaine. On représente en rouge les points de contrôle calculés à partir de sa géométrie.

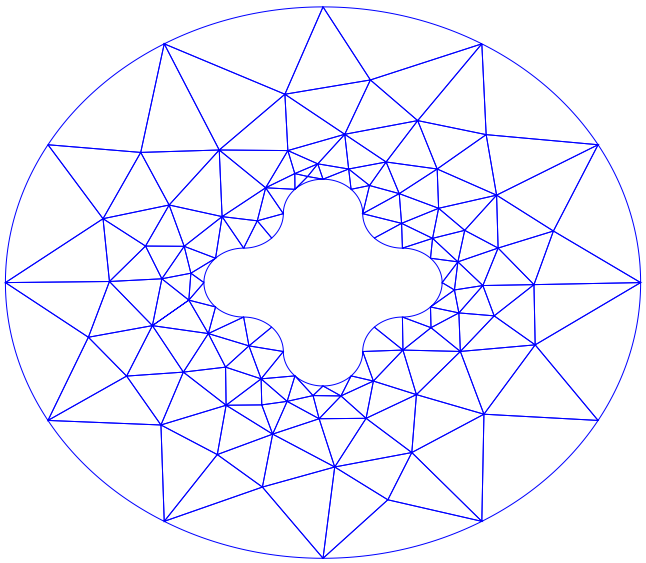


Figure III.2.4 – Maillage courbe issu du maillage III.2.3.

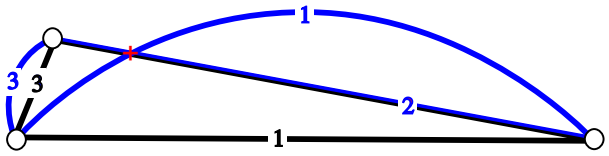


Figure III.3.2 – Croisement d'arêtes dans un élément courbe :
En noir est représenté l'élément droit initial.
L'arête 1 est une arête de bord, elle est donc courbée.
On obtient l'élément courbe représenté en bleu.
Le point de croisement de cet élément est indiqué en rouge.

3 Vérification et mise en conformité du maillage

Nos arêtes de bords, initialement droites, sont maintenant courbes. Il est donc possible que nous ayons créé des intersections d'arêtes dans notre maillage courbe (typiquement en présence de maillages contenant des couches limites ou de maillages trop grossièrement maillés vis à vis de leur courbure).

Exemple III.3.1 ***(Croisement dans un élément courbe)*** **:** La figure III.3.2 fournit un exemple de l'apparition d'un croisement d'arêtes dans un élément.

Exemple III.3.2 ***(Croisements dans un maillage courbe)*** **:** Les figures III.3.3 et III.3.4 donnent un exemple de maillage linéaire et du maillage courbe trouvé. Quatorze croisements apparaissent le long de la frontière interne de ce maillage.

Comme de tels maillages ne peuvent être utilisés, il nous faut établir un moyen de détecter les éléments posant problème et de les « décroiser ».

3.1 Détection des éléments présentant des croisements

Nous souhaitons établir un critère de détection de croisements dans un maillage $\mathcal{NURBS}$. On s'assurera ainsi de n'utiliser que des maillages parfaitement conformes.

Notre procédé de vérification se fonde sur la propriété d'enveloppe convexe du polygone de contrôle d'une $\mathcal{NURBS}$ (un procédé similaire dans le cas des courbes de Bézier est décrit dans [18]).

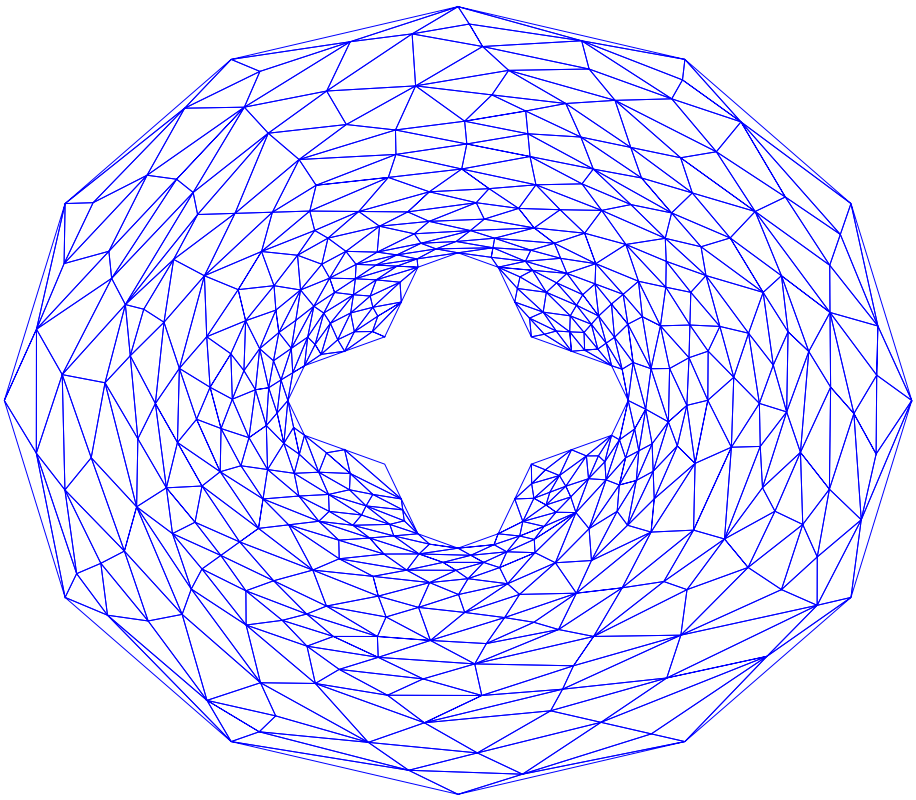


Figure III.3.3 – Maillage linéaire problématique :
Les mailles proches de la frontière sont trop anisotropes dans les zones de fortes courbures.

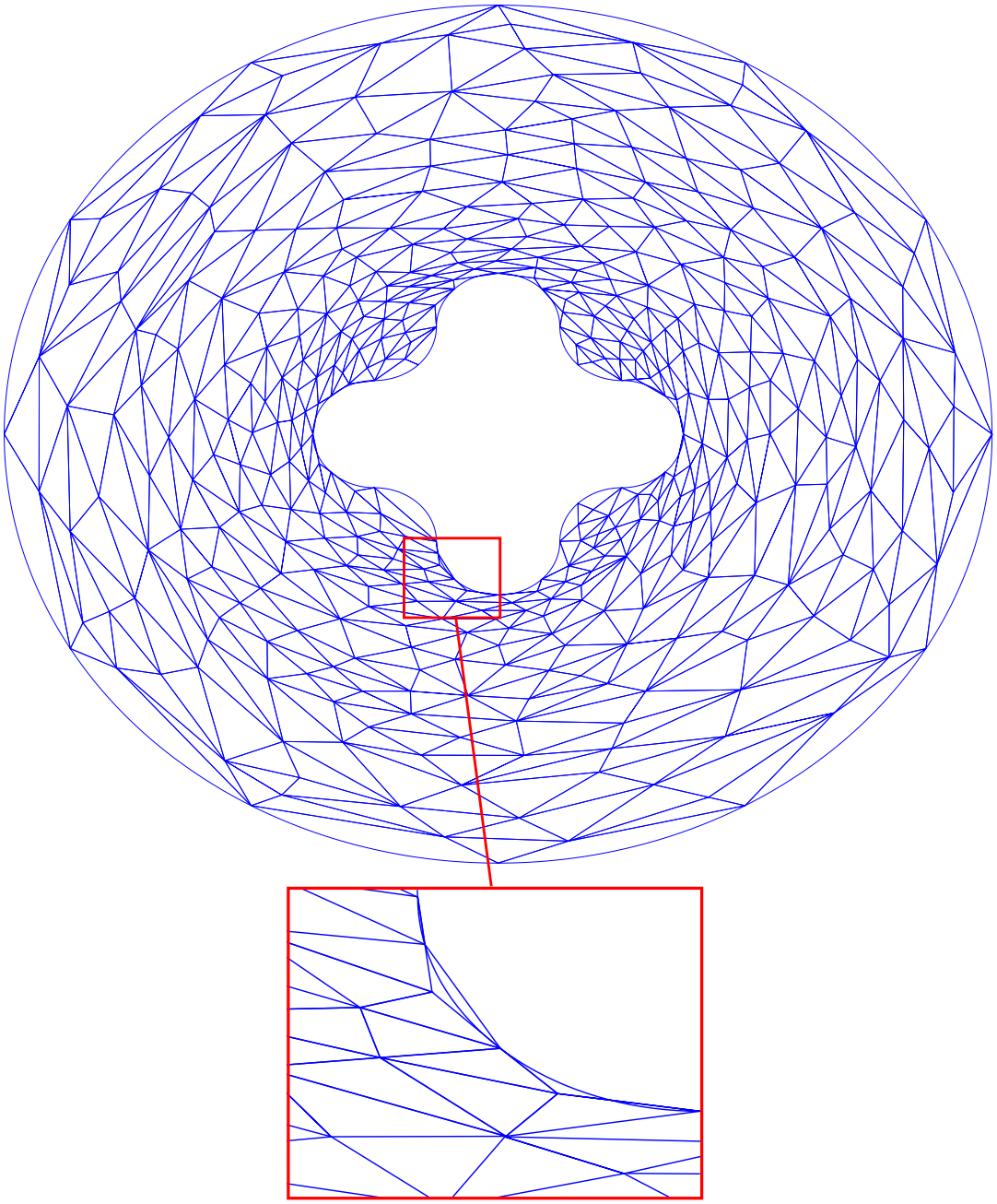


Figure III.3.4 – Maillage courbe anisotrope :
Le maillage courbe issu de III.3.3 présente des croisements le long de sa frontière intérieure (voir l'encadré rouge).

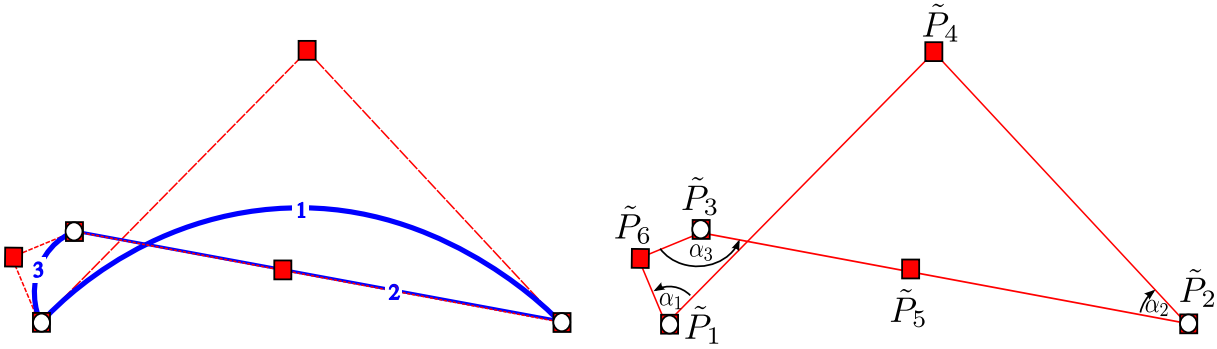


Figure III.3.5 – Notations et conventions de numérotation d'un élément courbe et de l'élément de contrôle associé :
À gauche on représente l'élément courbe et le polygone de contrôle associé à chacune de ses arêtes, à droite l'élément de contrôle et la numérotation des angles et points de contrôle.
Les angles α_1 et α_3 sont positifs, l'angle α_2 est négatif.

Définition III.3.1 (*élément de contrôle*)

On note *élément de contrôle* le polygone formé des polygones de contrôle associés à chaque arête de cet élément.

Remarque III.3.1 (*Points et polygone de contrôle d'une arête droite*)

Le polygone de contrôle associé à une arête droite est confondu avec cette arête. De plus, les points de contrôle du polygone sont confondus avec les DoF*s* de cette arête.

Considérons un élément courbe. Cet élément courbe est associé à un *élément de contrôle*. La figure III.3.5 donne un exemple d'élément de contrôle dans le cas d'un élément croisé.

Commençons par indicer notre élément de contrôle. On le parcourt dans le sens direct, on numérote les trois points de contrôle interpolés par les sommets de notre triangle courbe. Ces points de contrôle sont confondus avec les points $P1$ de l'élément donc on les nommera *points de contrôle* $P1$. On parcourt ensuite une seconde fois l'élément dans le sens direct en numérotant les points de contrôle restants (indiçage représenté figure III.3.5).

Définition III.3.2 (*Angles principaux d'un élément de contrôle*)

Dans un élément de contrôle, on nomme *angles principaux* les angles orientés dont le sommet est un point de contrôle $P1$.

On détermine ensuite à l'aide des trois angles principaux de notre élément de contrôle (définition III.3.2) si ce dernier est positif ou négatif (les définitions de la positivité et de la négativité d'un élément sont fournies définition III.3.3).

Définition III.3.3 (*Positivité et négativité d'un élément de contrôle*)

Un élément de contrôle est dit *négatif* si au moins l'un de ses angles principaux est négatif ou nul dans le sens direct. Sinon, il est dit *positif*.

Comme une courbe $\mathcal{NURBS}$ est incluse dans l'enveloppe convexe de son polygone de contrôle et est tangente en ses extrémités à ce dernier, alors le croisement entre deux des arêtes d'un élément courbe se traduit par un élément de contrôle négatif. Avec notre convention de numérotation, $\tilde{P}_1$, $\tilde{P}_2$ et $\tilde{P}_3$ sont les trois points de contrôle $P1$ de notre élément de contrôle. Son signe sera donc déterminé par les angles $\alpha_1 = \angle\tilde{P}_4\tilde{P}_1\tilde{P}_6$, $\alpha_2 = \angle\tilde{P}_5\tilde{P}_2\tilde{P}_4$ et $\alpha_3 = \angle\tilde{P}_6\tilde{P}_3\tilde{P}_5$.

Exemple III.3.3 ***(Élément de contrôle négatif)*** **:** Revenons à l'exemple fourni figure III.3.5, α_1 et α_3 sont positifs mais α_2 est négatif. On en conclut que l'élément de contrôle est négatif et donc que les deux arêtes partageant l'extrémité $\tilde{P}_2$ se croisent.

3.2 Suppression des croisements détectés

Lors de la création de notre maillage courbe, nous nous sommes assurés de conserver des arêtes droites à l'intérieur de notre domaine. Ainsi, dans la mesure où notre maillage $P1$ ne présentait pas de croisements, les seules intersections que l'on peut rencontrer sont localisées au niveau des éléments de bord de notre maillage et, plus particulièrement, entre l'arête $\mathcal{NURBS}$ et l'une des arêtes droites d'un élément ou entre deux arêtes de bord d'un élément. Si les deux arêtes croisées sont des arêtes de bord, cela signifie que deux des arêtes d'un même élément appartiennent au bord du domaine. Nous considèrerons ce type de maillage comme non-valide. Par contre, si une arête de bord intersecte une arête interne, nous devons les décroiser.

Nous ne pouvons pas modifier notre arête courbe puisqu'elle représente notre interpolation circulaire des points de bord et donc notre géométrie. La méthode la plus simple, dans le cas d'un maillage isotrope, consiste alors simplement à insérer un point permettant de créer un triangle valide, *c.-à-d.* un triangle garantissant la stricte positivité de tous les angles principaux de notre élément de contrôle. On crée ce nouvel élément par un forçage des frontières similaire à celui réalisé pour les maillages de Delaunay (voir [57] pour une

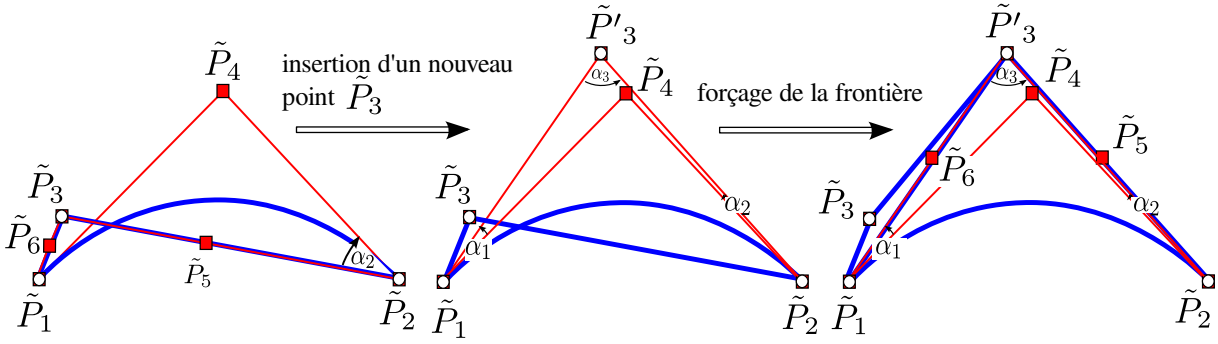


Figure III.3.6 – Décroisement d'un élément par insertion d'un sommet et forçage de la frontière.

présentation des maillages de Delaunay et de la méthode de forçage utilisée). Comme on a modifié les deux arêtes droites de l'élément, il nous faut recalculer leurs points de contrôle non $P1$.

Exemple III.3.4 ***(Procédure de décroisement)*** **:** On donne un exemple de décroisement figure III.3.6. On représente l'élément problématique à gauche de la figure. On insère un sommet $\tilde{P}'_3$ dans notre maillage tel que ce sommet, associé aux extrémités $\tilde{P}_1$ et $\tilde{P}_2$ de notre arête courbe, forme un élément valide. Ce nœud doit donc être positionné de sorte à avoir α_1, α_2, mais également α_3, positifs. Une fois l'élément créé, on calcule les nouveaux points $\tilde{P}_5$ et $\tilde{P}_6$. Notre élément de contrôle est maintenant positif, l'élément courbe associé ne comporte donc plus de croisement.

Comme nous forçons la création d'un nouvel élément pour décroiser les éléments de bord non conformes, les nœuds initialement localisés sur le lieu géométrique du nouvel élément ne seront plus utilisés. Nous terminerons donc notre procédure par la suppression de tous les nœuds isolés du maillage.

Exemple III.3.5 ***(Décroisement de la figure III.3.4)*** **:** On représente figure III.3.7 le décroisement du maillage représenté figure III.3.4. On constate que les éléments problématiques ont été modifiés et que tous les problèmes sont résolus. Le maillage obtenu est conforme.

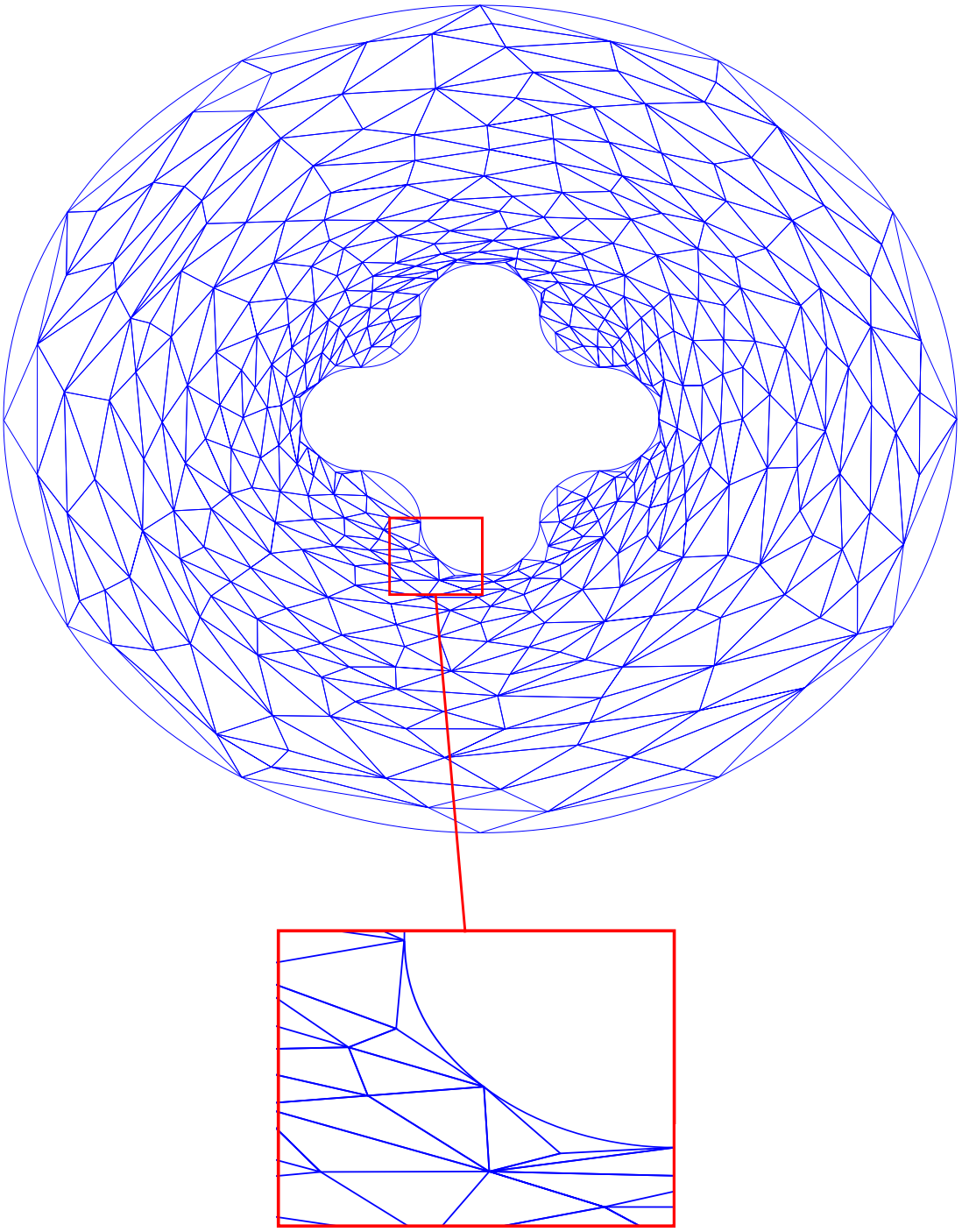


Figure III.3.7 – Décroisement du maillage représenté figure III.3.4.

3.3 Algorithme de vérification et mise en conformité

Pour récapituler, le maillage courbe créé à la section 2 peut contenir des croisements, un algorithme à 6 étapes permet de vérifier la présence ou non d'éléments croisés et, le cas échéant, de rendre notre maillage conforme tout préservant la géométrie de nos frontières.

Algorithme III.3.1: Mise en conformité du maillage

- Pour chaque élément :
  - Calcul des angles principaux de l'élément de contrôle considéré.
  - Détermination du signe de l'élément de contrôle. Si il est positif, il ne contient pas de croisement. On passe à l'élément suivant. Si il est négatif :
    - Comptage du nombre d'arêtes de bord. Si l'élément en contient plus d'une, il n'y a rien à faire, on arrête la procédure. Sinon, on passe à l'étape suivante.
    - Insertion d'un point $\tilde{P'}_3$ permettant de créer un élément valide.
    - Création de l'élément par forçage des arêtes $\tilde{P}_1\tilde{P'}_3$ et $\tilde{P}_2\tilde{P'}_3$, et passage à l'élément suivant.
- Suppression des sommets qui ne sont plus utilisés.

Remarque III.3.2 (*Limitation*)

Si le procédé de création, vérification et rectification permettant de générer des maillages courbes est particulièrement robuste et nous garantit des maillages valides, il souffre d'un inconvénient majeur : il détruit les couches limites que le maillage $P1$ peut contenir.

Exemple III.3.6 ***(Destruction des couches limites)*** **:** On représente figure III.3.8 un maillage de couche limite. Le maillage obtenu après courbure de la géométrie et procédure de mise en conformité est fourni figure III.3.9. En effet, il est évident qu'un élément de couche limite, dont les arêtes intérieures sont droites et l'arête de bord est courbe, présentera des croisements. Les éléments de la couche limite seront donc détruits par notre méthode.

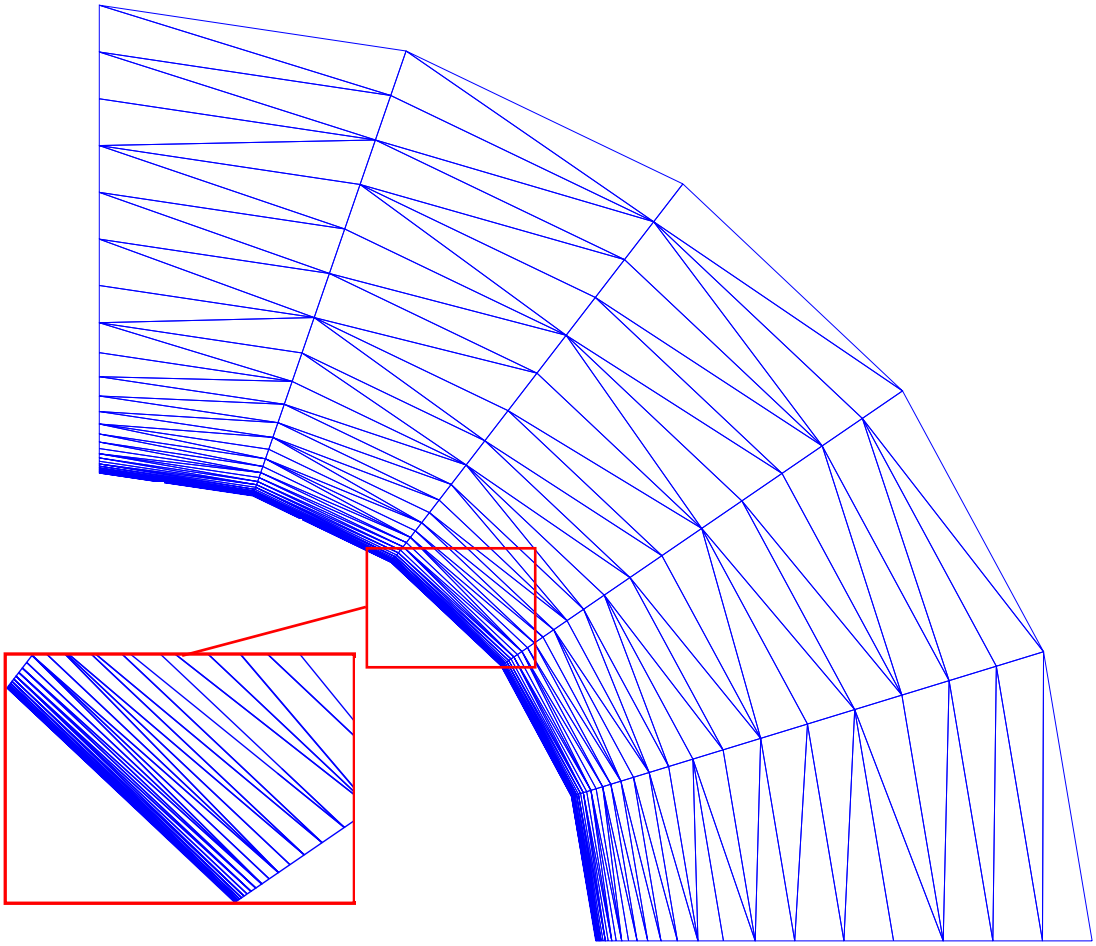


Figure III.3.8 – Maillage de couche limite..

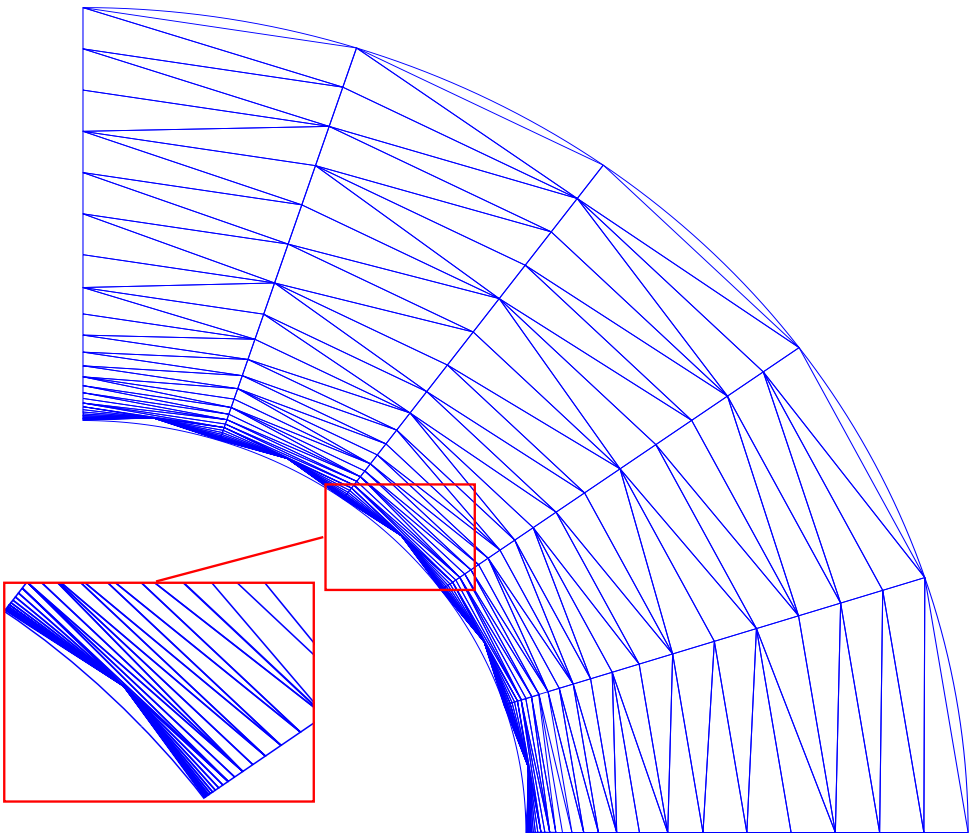


Figure III.3.9 – Destruction de la couche limite du maillage III.3.8..

Chapitre IV

Adaptation de maillages courbes d'ordre élevé

Ce chapitre vise à décrire les techniques d'adaptation de maillage utilisées. Le point de départ de cette adaptation étant un maillage courbe, on souhaitera préserver sa géométrie.

Très sommairement, l'adaptation de maillage consiste à élever le nombre de degrés de liberté dans les zones de forts gradients (donc, dans les zones de localisation des phénomènes physiques). On vise ainsi à distribuer nos nœuds de façon cohérente et donc à obtenir, à nombre de DoF*s* fixé, des résultats numériques plus précis. L'optimisation d'un maillage peut se faire à travers diverses techniques parmi lesquelles la p-adaptation, que nous évoquerons section 2, et le h-raffinement, que nous allons traiter section 3. La p-adaptation augmente le nombre de DoF*s* via l'élévation du degré des fonctions de forme tandis que la h-adaptation consiste à adapter la taille élémentaire des mailles par l'ajout et la suppression de nœuds. Nous décrirons ces méthodes tant d'un point de vue global qu'élémentaire.

Au préalable, il nous faut détailler les conventions d'indiçage que nous allons utiliser. Comme, dans notre cas, le nombre de points de contrôle associés à un élément est égal à son nombre de degrés de liberté, nous utiliserons toujours la même convention de numérotation pour décrire les DoF*s* de l'élément et pour décrire sa grille de contrôle.

1 Convention de numérotation : numérotation fonctionnelle

La numérotation principalement utilisée dans ce document, que nous nommerons *numérotation fonctionnelle*, consiste à adopter la même convention d'indiçage pour nos DoF*s* (et notre grille de contrôle) que pour les fonctions de base associées.
Nous avons présenté plusieurs types de fonctions de base et plusieurs formes de numéro-

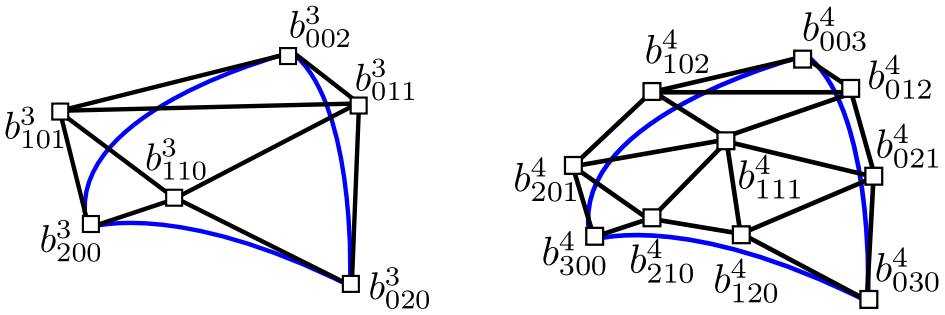


Figure IV.1.2 – Numérotation fonctionnelle des points de contrôle : grilles de contrôle permettant de représenter le triangle de bézier bleu à l'ordre 3 (à gauche) et à l'ordre 4 (à droite).

tations. Nous choisirons donc la plus pratique.

Pour un triangle, nous avons choisi d'utiliser une numérotation barycentrique des fonctions de base, DoF*s* et $\mathcal{CP}$*s*. On note donc $b_{l_i m_i n_i}$ le point de contrôle associé à la fonction de base $\Psi_{l_i m_i n_i}$ d'ordre k (où $\Psi_{l_i m_i n_i}$ est la $i^{\text{ème}}$ fonction de Bernstein rationnelle obtenue à partir de la $i^{\text{ème}}$ fonction de base de Bernstein $\varphi_{l_i m_i n_i}$ définie équation (II.13)). En règle générale, nous allégerons la notation en supprimant l'indice i. Notons Δ_o l'ensemble des triplets $(l,m,n) \in [\![0,o]\!]^3$ tels que $l+m+n=o$. Notre grille de contrôle est donc formée des $(b_{lmn})_{(lmn)\in\Delta_{k-1}}$. Pour simplifier notre notation, on pose$(b^k_{lmn}) = (b_{lmn})_{(lmn)\in\Delta_{k-1}}$. Nos fonctions de base d'ordre k sont donc simplement notées b^k_{lmn}. La figure IV.1.2 donne deux exemples de patchs triangulaires et des grilles de contrôle qui leur sont associés.

Pour les quadrangles, la numérotation la plus simple est la numérotation tensorielle. On considère notre élément comme le produit tensoriel de deux vecteurs nodaux ouverts. On note b_{lm} le point de contrôle associé à la fonction de base Ψ_{lm} (où Ψ_{lm} est la fonction de base définie équation (II.2.7)). La figure IV.1.3 donne deux exemples de patchs tensoriels et des grilles de contrôle associées.

Quelque soit l'élément considéré, nous désignerons par DoF*s externes* l'ensemble des sommets et des DoF*s* des faces de l'élément et par DoF*s internes* l'ensemble des DoF*s* situés à l'intérieur de l'élément. Par analogie, les points de contrôle de l'élément associés à un DoF externe seront nommés $\mathcal{CP}$*s externes* et ceux associés à un DoF interne seront nommés $\mathcal{CP}$*s internes*.

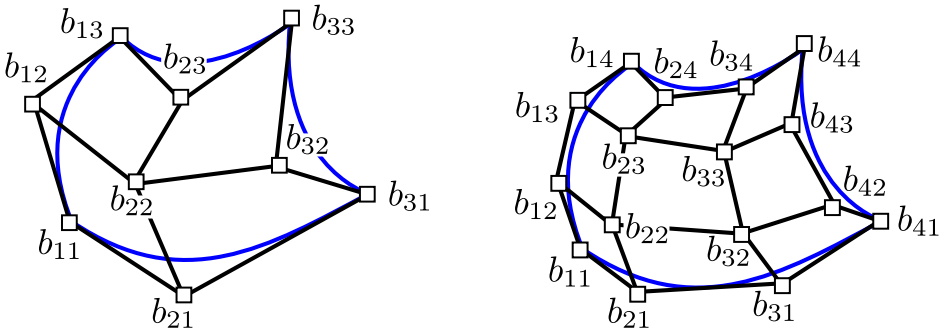


Figure IV.1.3 – Numérotation fonctionnelle des points de contrôle : grilles de contrôle permettant de représenter le quadrangle de Bézier bleu à l'ordre 3 (à gauche) et à l'ordre 4 (à droite).

Remarque IV.1.1 (*Points de contrôles externes d'un triangle*)

les points de contrôle externes d'un triangle sont les points (b^k_{lmn}) tels qu'au moins l'un des indices l, m ou n soit nul.

2 p-adaptation de maillages courbes

Afin de pouvoir lancer des calculs d'ordres très élevés [1] à partir des maillages courbes précédemment décrits, il nous faut pouvoir élever l'ordre du maillage lu. De par leur définition récursive, les fonctions de Bernstein se prêtent parfaitement à cette opération. Elles permettent également d'élever localement l'ordre des éléments et ainsi, de raffiner le maillage, c'est le *p-raffinement*. Cependant, afin de garantir la continuité C^0 de nos fonctions de base à l'interface des éléments, les points de contrôle de cette interface doivent être identiques, quelque soit l'élément depuis lequel on les considère. Ceci nous contraint à élever uniformément l'ordre sur les éléments du maillage. Par souci de simplicité, nous utiliserons les coordonnées homogènes des points de contrôle. Ainsi, nous ne travaillerons que sur des surfaces de Bézier. Nous présenterons donc dans cette section l'élévation d'ordre des deux formes élémentaires unitaires de la dimension 2.

2.1 Élément Triangulaire

Pour élever l'ordre d'un élément on utilise la formule d'élévation d'ordre de De Casteljau adaptée aux patchs de Bézier triangulaires. Cette formule est inspirée de l'algorithme

1. la notion d'ordre très élevé est, bien sûr, relative au domaine d'étude. Dans notre cas, on considèrera l'ordre 3 comme de l'ordre élevé, les ordres supérieurs étant dits *très élevés*.

d'élévation d'ordre de De Casteljau du document [27].

Propriété IV.2.1 (*Élévation d'ordre sur les patchs triangulaires*)

Soit $\{\Psi_{lmn}, (l,m,n) \in \Delta_{k-1}\}$ la famille des $\frac{k(k+1)}{2}$ fonctions de base de Bernstein non rationnelles d'ordre k et soit b^k le polygone de contrôle homogène associé à notre élément.

Notre géométrie initiale sera conservée si l'on associe aux $\frac{(k+1)(k+2)}{2}$ fonctions de base de Bernstein d'ordre $(k+1)$ les $\frac{(k+1)(k+2)}{2}$ nouveaux points de coordonnées homogènes : $\bar{b}^{k+1}_{lmn} = (l-1)\, b^k_{(l-1)mn} + (m-1)\, b^k_{l(m-1)n} + (n-1)\, b^k_{lm(n-1)}$

La figure IV.1.2 présentée section 1 illustre l'élévation de l'ordre 3 à l'ordre 4 d'un triangle courbe.

2.2 Élément Quadrangulaire

À nouveau, on élèvera l'ordre de notre quadrangle en s'inspirant du document [27].

Propriété IV.2.2 (*Élévation d'ordre sur les patchs quadrangulaires*)

Soit $\{\Psi^k_{ij}\}$ la famille des k^2 fonctions de base de Bernstein non rationnelles d'ordre k sur le quadrangle et soit b^k le polygone de contrôle homogène associé à notre élément.

Notre géométrie initiale sera conservée si l'on associe aux $(k+1)^2$ fonctions de base de Bernstein d'ordre $(k+1)$ les $(k+1)^2$ nouveaux points de coordonnées homogènes :

$$\bar{b}^{k+1}_{l,m} = \frac{1}{k^2}\Big(\begin{array}{ll} (l-1)(m-1)\, b^k_{(l-1)(m-1)} & +(k-l+1)(m-1)\, b^k_{l(m-1)} + \\ (l-1)(k-m+1)\, b^k_{(l-1)m} & +(k-l+1)(k-m+1)\, b^k_{lm} \end{array} \Big)$$

La figure IV.1.3 présentée section 1 illustre l'élévation de l'ordre 3 à l'ordre 4 d'un quadrangle courbe.

Remarque IV.2.1 (*Alternative*)

On peut noter qu'une autre méthode d'élévation d'ordre des surfaces quadrangulaires consiste à les considérer comme le produit tensoriel de deux vecteurs nodaux. On élève alors successivement l'ordre de chacun des vecteurs générant notre patch paramétrique. Par soucis de cohérence avec la technique de p-adaptation employée pour les éléments triangulaires, nous n'emploierons pas cette autre méthode.

Nous terminerons cette section par le récapitulatif de l'algorithme de p-adaptation. On suppose que notre maillage est initialement d'ordre k_1 et qu'on souhaite travailler à l'ordre k_2.

Algorithme IV.2.1: Algorithme de p-adaptation

- Calcul des coordonnées homogènes des nœuds du maillage.
- Pour chaque élément du maillage :
  - Tant que k_1 est strictement inférieur à k_2 :
    - Calcul des coordonnées homogènes permettant de représenter l'élément à l'ordre supérieur.
    - Incrémentation de k_1
- Calcul des poids et coordonnées réelles des nœuds du maillage.

3 h-adaptation de maillages courbes

Nous pouvons maintenant nous consacrer à l'adaptation de maillages courbes en fonction de la physique des problèmes traités. Nous nous concentrerons dans cette section sur l'adaptation de maillage dans le cas d'un problème stationnaire. On cherche à équirépartir les erreurs de notre solution sur le maillage, c'est-à-dire que l'on cherche un point fixe pour le couple *maillage-solution*. Pour plus de précisions sur l'adaptation de maillage, le lecteur pourra se référer à la thèse de C. Dobrzynski [58], en particulier à la section **1.3**.

Nous rappellerons tout d'abord quelques définitions nécessaires à la compréhension de ce chapitre. Nous nous intéresserons ensuite à l'adaptation d'un maillage classique P_1 et à son extension aux maillages courbes d'ordre k. Pour conclure, nous rappellerons le schéma d'adaptation stationnaire et validerons notre méthode.

3.1 Définitions

Cette sous-section vise à introduire les notions de *métrique*, de *produit scalaire associée à une métrique* et de *longueur d'arête dans une métrique*.

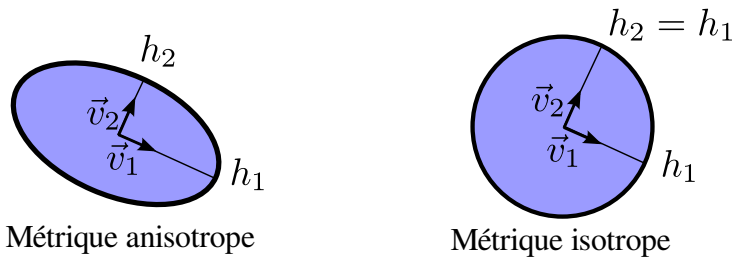


Figure IV.3.2 – Métriques isotropes et anisotropes

Définition IV.3.1 (*Métrique*)

Une métrique $\mathcal{M}$ est un tenseur symétrique défini positif ($\mathcal{SDP}$). Comme il est $\mathcal{SDP}$, le tenseur $\mathcal{M}$ est diagonalisable et peut s'écrire sous la forme :

$$\mathcal{M} = V\,\Lambda\,V^{-1} = \begin{pmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \begin{pmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{pmatrix}^{-1}$$

où $\vec{\mathbf{v}}_1 = {}^T(v_{11}, v_{21})$ et $\vec{\mathbf{v}}_2 = {}^T(v_{12}, v_{22})$ sont les vecteurs propres de la métrique et λ_1 et λ_2 ses valeurs propres.

Posons $h_1 = \frac{1}{\sqrt{\lambda_1}}$ et $h_2 = \frac{1}{\sqrt{\lambda_2}}$, la métrique $\mathcal{M}$ définie au point O peut être représentée graphiquement par sa boule unité de centre O, *c.-à-d.* par une ellipse de centre O dont les rayons ont pour direction les vecteurs $\vec{\mathbf{v}}_1$ et $\vec{\mathbf{v}}_2$ et pour longueurs respectives h_1 et h_2.

Remarque IV.3.1 (*Métriques isotropes et anisotropes*)

Une métrique est dite anisotrope si $\lambda_1 \neq \lambda_2$ et isotrope si $\lambda_1 = \lambda_2$ (voir figure IV.3.2).

Définition IV.3.2 (*Produit scalaire associé à une métrique*)

Soit $\mathcal{M}$ une métrique et $\vec{\mathbf{u}}$ et $\vec{\mathbf{v}}$ deux vecteurs de $\mathbb{R}^d$. Le produit scalaire de $\vec{\mathbf{u}}$ et $\vec{\mathbf{v}}$ dans la métrique $\mathcal{M}$ est défini de $\mathbb{R}^d$ dans $\mathbb{R}$ par :

$$\langle \vec{\mathbf{u}}, \vec{\mathbf{v}} \rangle_{\mathcal{M}} = {}^T\vec{\mathbf{u}}\,\mathcal{M}\,\vec{\mathbf{v}}. \tag{IV.1}$$

Définition IV.3.3 (*Norme dans une métrique*)

La norme euclidienne d'un vecteur $\vec{\mathbf{u}}$ dans la métrique $\mathcal{M}$ est définie par :

$$\|\vec{\mathbf{u}}\|_{\mathcal{M}} = \sqrt{\langle \vec{\mathbf{u}}, \vec{\mathbf{u}} \rangle_{\mathcal{M}}} = \sqrt{{}^T\vec{\mathbf{u}}\,\mathcal{M}\,\vec{\mathbf{u}}}. \tag{IV.2}$$

Définition IV.3.4 (*Longueur et distance dans une métrique*)

Notons $d_{\mathcal{M}}(A, B)$ la distance de A à B dans la métrique $\mathcal{M}$ et $l_{\mathcal{M}}(\vec{AB})$ la longueur, toujours dans dans la métrique $\mathcal{M}$, du vecteur $\vec{AB}$. On a alors les relations suivantes :

$$d_{\mathcal{M}}(A, B) = l_{\mathcal{M}}(\vec{AB}) = \|\vec{AB}\|_{\mathcal{M}}. \tag{IV.3}$$

Dans notre cas, nous associerons une métrique à chaque nœud du maillage. Les vecteurs propres de la métrique nous donneront donc les directions de raffinement, *c.-à-d.* l'orientation de nos arêtes. Les valeurs propres de la métrique nous fourniront les longueurs d'arêtes voulues. On peut noter qu'une métrique isotrope ne privilégiera donc aucune direction d'arête et aura tendance à créer des mailles équilatérales.

Intéressons nous maintenant au cœur de cette section, le raffinement de maillage.

3.2 Adaptation P_1

Cette sous-section a pour but de rappeler le calcul, dans le cas d'une discrétisation P_1, de la métrique $\mathcal{M}$ associée à un maillage $\mathcal{T}$. Nous détaillerons également le calcul de la longueur des arêtes du maillage dans la métrique $\mathcal{M}$.

3.2.1 Calcul de la métrique

Notons ε le seuil d'erreur maximale admis sur notre maillage. On cherche donc une solution approchée satisfaisant $\|\boldsymbol{U} - \boldsymbol{U}_h\| \leq \varepsilon$. En réalité, nous souhaitons contrôler uniquement l'une des variables de la solution (par exemple, la densité ρ). Quelle qu'elle soit, nous nommerons cette variable u. Il nous faut donc à estimer l'erreur $\|u - u_h\|$. Afin de contrôler notre maillage, l'estimateur d'erreur exhibé doit nous donner des informations sur la taille et la direction des arêtes du maillage.

Un exemple d'estimateur d'erreur classique est l'estimateur d'erreur d'interpolation. Cet estimateur se fonde sur la majoration en norme infinie de l'erreur d'interpolation

d'ordre 2 sur un élément T (voir [20]).

Propriété IV.3.1 (*Estimateur d'erreur d'interpolation* P_1)

On considère un maillage $\mathcal{T}$, une variable u et son approximation u_h sur les fonctions de base P_1. Notons E_T l'ensemble des arêtes de l'élément T, $\vec{e}$ les arêtes du maillage et $\mathcal{M}_\varepsilon(T)$ la métrique associée à l'élément T et à l'erreur ε (définie équation IV.14).

L'erreur d'interpolation P_1 est majorée par ε si les arêtes de l'élément T sont de longueur inférieure ou égale à 1 dans la métrique $\mathcal{M}_\varepsilon(T)$:

$$\mathcal{P}_e(x) = (u - \Pi_h(u)) \leq \varepsilon \Leftrightarrow \langle \vec{e}, \mathcal{M}_\varepsilon(T)\vec{e} \rangle \leq 1 \; \forall \vec{e} \in E_T, \tag{IV.4}$$

où $\Pi_h(u)$ est l'interpolée linéaire de u sur T.

Démonstration : Notons $\Pi_h(u)$ l'interpolée linéaire de u sur T, $H_u(x)$ la matrice hessienne de u au point x et $\langle \cdot, \cdot \rangle$ le produit scalaire de $\mathbb{R}^2$. Pour simplifier les formules, l'erreur d'interpolation $(u - \Pi_h(u))(x)$ sera notée $\mathcal{P}_e(x)$. Pour obtenir la majoration voulue, on écrit l'erreur d'interpolation en un sommet a de T comme son développement de Taylor avec reste intégral par rapport à un point x quelconque de T :

$$\mathcal{P}_e(a) = \mathcal{P}_e(x) + \langle x\vec{a}, \nabla \mathcal{P}_e(x) \rangle + \int_0^1 (1-t) \langle a\vec{x}, H_u(x + t\, a\vec{x})\, a\vec{x} \rangle dt. \tag{IV.5}$$

On suppose que nos fonctions de base sont interpolantes aux sommets de T (ceci est vrai pour les fonctions de base lagrangiennes comme pour les fonctions de base $\mathcal{NURBS}$), on a alors $\mathcal{P}_e(a) = 0$ d'où :

$$\mathcal{P}_e(x) = \langle a\vec{x}, \nabla \mathcal{P}_e(x) \rangle - \int_0^1 (1-t) \langle a\vec{x}, H_u(x + t\, a\vec{x})\, a\vec{x} \rangle dt. \tag{IV.6}$$

Pour la suite de cette démonstration, plaçons-nous dans le cas d'un maillage triangulaire bidimensionnel, l'extension aux autres cas étant directe. Comme on cherche à majorer notre erreur, on cherche un maximum de $\mathcal{P}_e(x)$:

- Si ce maximum est atteint en un point x situé strictement dans notre élément T, alors $\nabla \mathcal{P}_e(x) = 0$ et :

$$|\mathcal{P}_e(x)| = |\int_0^1 (1-t) \langle a\vec{x}, H_u(x + t\, a\vec{x})\, a\vec{x} \rangle dt|. \tag{IV.7}$$

On suppose que x est plus proche du sommet a que des autres sommets de l'élément que l'on nomme b et c. Soit a' le point d'intersection de la droite (ax) avec l'arête

bc, alors il existe un scalaire λ tel que $\vec{ax} = \lambda \vec{aa'}$ et $0 \leq \lambda \leq \frac{2}{3}$. On en déduit que :

$$\begin{array}{rcll} |\mathcal{P}_e(x)| & \leq & \lambda^2 & |\int_0^1 (1-t) \quad \langle \vec{aa'}, H_u\,(x+t\,\vec{ax})\;\vec{aa'}\rangle dt| \\ & \leq & \frac{4}{9} & |\int_0^1 (1-t)| \quad \max\limits_{t\in[0,1]} |\langle \vec{aa'}, H_u\,(x+t\,\vec{ax})\;\vec{aa'}\rangle dt|. \end{array} \tag{IV.8}$$

D'où, en notant $\vec{v}$ les vecteurs de T :

$$|\mathcal{P}_e(x)| \leq \frac{2}{9} \max_{y\in \mathrm{T}} \max_{\vec{v}\in \mathrm{T}} |\langle \vec{v}, H_u\,(y)\;\vec{v}\rangle|.$$

- Supposons maintenant que ce maximum est atteint en un point x situé sur une arête $\vec{ab}$ de T, comme $\mathcal{P}_e(x)$ est maximum le long de $\vec{ab}$, son gradient est orthogonal à $\vec{ab}$. De plus $\vec{xa}$ est colinéaire à $\vec{ab}$ donc $\langle \vec{ax}, \nabla \mathcal{P}_e\,(x)\rangle = 0$. En sommant les développements de Taylor de $\mathcal{P}_e(a)$ et $\mathcal{P}_e(b)$ par rapport à x on trouve alors :

$$2\,\mathcal{P}_e(x) = -\int_0^1 (1-t)\langle \vec{ax}, H_u\,(x+t\,\vec{ax})\;\vec{ax} + (1-t)\langle \vec{bx}, H_u\left(x+t\,\vec{bx}\right)\;\vec{bx}\rangle dt. \tag{IV.9}$$

Comme x est un point de l'arête ab, il existe un scalaire λ tel que $\vec{ax} = \lambda\,\vec{ab}$, $\vec{bx} = (\lambda - 1)\,\vec{ab}$ et $0 \leq \lambda \leq 1$.
On a donc :

$$|\mathcal{P}_e(x)| \quad \leq \quad \tfrac{1}{4} \quad \max_{\lambda\in[0,1]} \left(\lambda^2 + (1-\lambda)^2\right) \max_{y\in ab} |\langle \vec{ab}, H_u\,(y)\;\vec{ab}\rangle|. \tag{IV.10}$$

Finalement, on trouve :

$$|\mathcal{P}_e(x)| \leq \frac{1}{8} \max_{y\in ab} \max_{\vec{v}\in \mathrm{T}} |\langle \vec{v}, H_u\,(y)\;\vec{v}\rangle| \leq \frac{2}{9} \max_{y\in \mathrm{T}} \max_{\vec{v}\in \mathrm{T}} |\langle \vec{v}, H_u\,(y)\;\vec{v}\rangle|.$$

On peut alors majorer l'erreur d'interpolation réalisée sur le triangle T en norme $\mathcal{L}^\infty$:

$$\|u - \Pi_h\,(u)\|_{\infty,T} \leq \frac{2}{9} \max_{y\in T} \max_{\vec{v}\in T} \langle \vec{v}, |H_u(y)|\vec{v}\rangle. \tag{IV.11}$$

Dans le cas général (quadrangles, tétraèdres...), une démonstration analogue permet de montrer que l'on obtient la majoration :

$$\|u - \Pi_h\,(u)\|_{\infty,T} \leq c \max_{y\in T} \max_{\vec{v}\in T} \langle \vec{v}, |H_u(y)|\vec{v}\rangle, \tag{IV.12}$$

où c est une constante dépendant du type d'élément considéré. Il n'est malheureusement pas possible d'évaluer numériquement les maximas de cette majoration, nous allons donc chercher à les majorer.

Tout vecteur $\vec{v}$ peut être réécrit comme combinaison linéaire à coefficients dans $[0,1]$ des arêtes $\vec{e}$ de T (voir annexe 2 de [20]). De plus le Hessien de u peut être majoré sur T par une métrique $\bar{\mathcal{M}}$. On a donc :

$$\|u - \Pi_h(u)\|_{\infty,T} \leq c \max_{\vec{e} \in E_T} \langle \vec{e}, \bar{\mathcal{M}}(T)\vec{e} \rangle. \tag{IV.13}$$

Posons :

$$\varepsilon_c = c \max_{\vec{e} \in E_T} \langle \vec{e}, \bar{\mathcal{M}}(T)\vec{e} \rangle.$$

Le terme ε_c nous fournit une carte des erreurs commises sur notre maillage et nous savons que $\|u - u_h\| \leq \varepsilon$ pour $\varepsilon_c \leq \varepsilon$.
Posons :

$$\mathcal{M}_\varepsilon(T) = \frac{c}{\varepsilon} \bar{\mathcal{M}}(T), \tag{IV.14}$$

alors :

$$\|u - u_h\| \leq \varepsilon \Leftrightarrow \langle \vec{e}, \mathcal{M}_\varepsilon(T)\vec{e} \rangle \leq 1 \; \forall \vec{e} \in E_T. \tag{IV.15}$$ ■

Notre erreur d'interpolation d'ordre 2 sur T sera inférieure à ε si, pour toute arête du maillage, le terme $\langle \vec{e}, \mathcal{M}_\varepsilon(T)\vec{e} \rangle$ est inférieur à 1 (*c.-à-d.* si toute arête $\vec{e}$ est de taille inférieure à 1 dans la métrique $\mathcal{M}_\varepsilon(T)$). Ce terme nous permet donc d'estimer la qualité de notre élément T. En pratique, la métrique $\mathcal{M}_\varepsilon$ sera évaluée, non pas sur les éléments du maillage, mais sur les sommets de celui-ci.
Dans la suite, nous noterons simplement $\mathcal{M}$ la métrique $\mathcal{M}_\varepsilon$.

3.2.2 Calcul de la longueur d'arête

Soit T un élément de sommets P_1, P_2, et P_3, et $\mathcal{M}(P_i)$ la métrique associée. On souhaite que chaque arête de T soit de longueur unitaire dans les métriques qui lui sont associées. Considérons, par exemple, l'arête P_1P_2. Cette arête doit être de longueur 1 pour la métrique $\mathcal{M}(P_1)$ ainsi que pour la métrique $\mathcal{M}(P_2)$. Il nous faut donc évaluer la longueur moyenne pour les métriques des deux extrémités de notre arête. Soit $l_{\mathcal{M}(P_1,P_2)}$ la longueur moyenne pour les métriques $\mathcal{M}(P_1)$ et $\mathcal{M}(P_2)$. Cette longueur moyenne est définie telle que :

$$l_{\mathcal{M}(P_1,P_2)}(\overrightarrow{P_1P_2}) = \int_0^1 \sqrt{{}^T\overrightarrow{P_1P_2} \left[\mathcal{M}(P_1) + t\Big(\mathcal{M}(P_2) - \mathcal{M}(P_1)\Big)\right] \overrightarrow{P_1P_2}}\, dt. \tag{IV.16}$$

Ceci correspond simplement à l'intégration sur notre arête de sa longueur pour la métrique approchée $\mathcal{M}_h$, où $\mathcal{M}_h$ est définie en tout point de l'arête comme l'interpolation des valeurs de $\mathcal{M}$ aux degrés de liberté de l'arête sur les fonctions de base de Lagrange d'ordre 2.

Le calcul de la longueur moyenne de chaque arête à partir des métriques des sommets du maillage permet ainsi d'obtenir la carte de la taille des arêtes du maillage et de décider des arêtes trop longues qu'il faudra couper.

3.3 Adaptation des maillages courbes d'ordre k

Dans notre, cas, comme nous travaillons à l'ordre k, nous avons des informations supplémentaires que nous souhaitons utiliser. Il nous faut donc trouver une méthode similaire à celle décrite mais adaptée à l'ordre k.

3.3.1 Calcul de la métrique

L'estimateur d'erreur que nous avons précédemment présenté, n'est valable, a priori, que pour une interpolation d'ordre 2.

La solution envisagée pour contourner ce problème consiste à décomposer le couple *maillage courbe-solution* d'ordre k en le subdivisant en un couple *maillage droit-solution* d'ordre 2. Chaque élément d'ordre k est donc décomposé, en dimension d, en k^d éléments d'ordre 2. Ainsi tous les degrés de liberté de notre maillage courbe deviennent des sommets du maillage droit (voir figure IV.3.3).

On utilise alors l'estimateur d'erreur $P1$ sur notre couple subdivisé. On obtient ainsi la métrique en chaque degré de liberté de notre maillage initial d'ordre élevé. Comme on sait que l'erreur commise, à même nombre de DoFs, sur un maillage Pk, est inférieure à celle commise sur un maillage $P1$, l'erreur $P1$ estimée majore bien l'erreur que l'on commet.

Le graphique IV.3.4 représente l'erreur commise en norme $\mathcal{L}_\infty$ sur l'approximation de la fonction cosinus à l'ordre 3 (courbe rouge) et à l'ordre 4 (courbe bleue). L'erreur d'interpolation linéaire calculée sur les maillages droits associés est représentée en vert. On remarque que la courbe représentant l'erreur $P1$ majore effectivement les autres courbes. La pente des courbes d'erreur étant liée à l'ordre du calcul, cette majoration restera vraie quelque soit le nombre de DoFs du maillage.

On constate également que notre erreur sera surestimée, nous aurons donc tendance à trop raffiner notre maillage. Bien que notre estimateur ne soit pas optimal, il reste néanmoins valide.

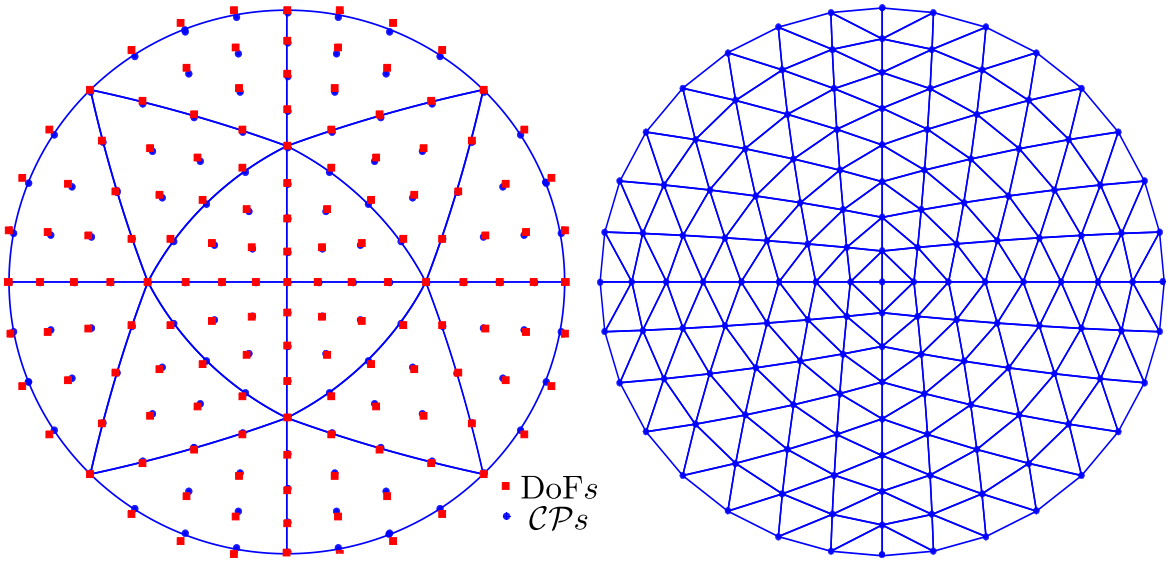


Figure IV.3.3 – Maillage courbe d'ordre 5 (à gauche) et maillage droit subdivisé (à droite) sur lequel sera calculée la métrique.

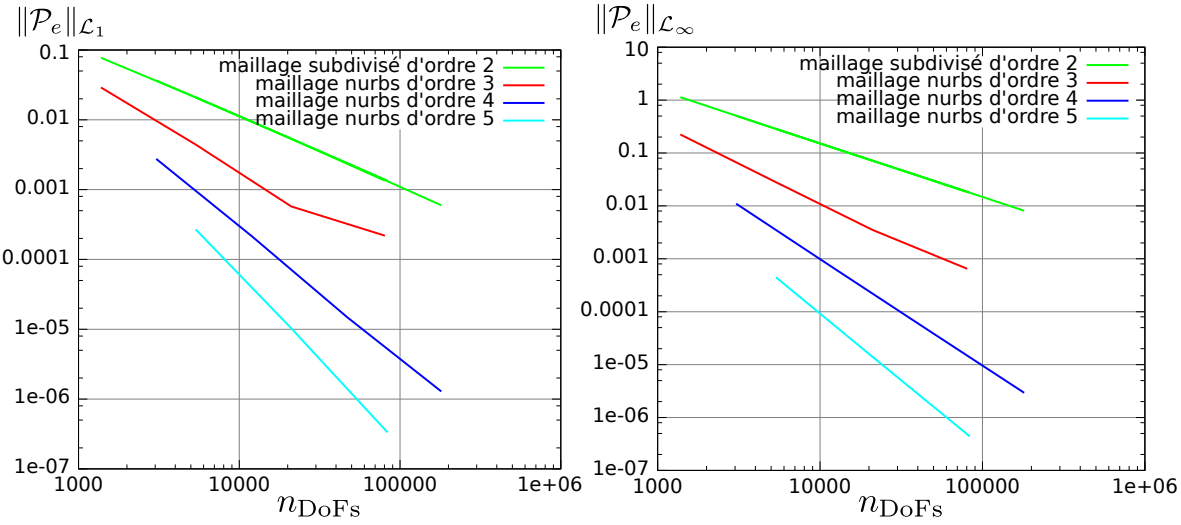


Figure IV.3.4 – Comparaison de l'erreur d'interpolation lagrangienne d'ordre 2, et des erreurs d'extrapolations $\mathcal{NURBS}$ d'ordre 3, 4 et 5 en norme $\mathcal{L}_1$ (à gauche) et $\mathcal{L}_\infty$ (à droite).

3.3.2 Calcul de la longueur d'arête voulue

À nouveau, considérons un segment et nommons $(\mathcal{M}_i)_{i=1,k}$ les métriques associées aux k DoFs de ce segment. Notons $(\varphi_i)_{i=1,k}$ les fonctions de base $\mathcal{NURBS}$ linéiques d'ordre k et $(\varphi_i')_{i=1,k}$ leur dérivées. Notre arête est maintenant courbe et sa longueur moyenne, $l_{\mathcal{M}}$, est calculée à l'ordre élevé :

$$l_{\mathcal{M}}(\overrightarrow{P_1P_2}) = \int_0^1 \sqrt{{}^T\vec{\mathbf{t}}_{P_1P_2}(t) \left[\sum_{i=1}^{k} \tilde{\mathcal{M}}_i\varphi_i\right] \vec{\mathbf{t}}_{P_1P_2}(t)}\, dt, \tag{IV.17}$$

où $\vec{\mathbf{t}}_{P_1P_2}(t)$ désigne la tangente (non normalisée) locale au vecteur courbe orienté $\overrightarrow{P_1P_2}$ et évaluée au point paramétrique t. Le terme $\tilde{\mathcal{M}}_i$ désigne la $i^{\text{ème}}$ métrique de contrôle et, comme toujours, les k métriques de contrôle sont calculées pour que leur extrapolation sur les fonctions de base $\mathcal{NURBS}$ interpole les métriques physiques $(\mathcal{M}_i)_{i=1,k}$ des DoFs.

Remarque IV.3.2 (*Métriques de contrôle*)

Dans le cas anisotrope, les métriques de contrôle n'ont pas forcément de sens physique, nous nous restreindrons donc aux cas isotropes. La métrique physique est alors un tenseur diagonal et il nous suffit de calculer les valeurs de contrôle associées aux valeurs physiques de sa diagonale.

On calcule la tangente à notre arête en t comme l'extrapolation des points de contrôle $\tilde{P}_i = (\tilde{X}_i, \tilde{Y}_i)$ de l'arête sur la dérivée évaluée à la coordonnée paramétrique t des fonctions de base de Bernstein rationnelles. Finalement, on réécrit notre longueur sous la forme :

$$l_{\mathcal{M}}(\overrightarrow{P_1P_2}) = \int_0^1 \sqrt{\left(\sum_{i=1}^{k} \tilde{X}_i\varphi_i'(t) \quad \sum_{i=1}^{k} \tilde{Y}_i\varphi_i'(t)\right) \left[\sum_{i=1}^{k} \tilde{\mathcal{M}}_i\varphi_i\right] \begin{pmatrix} \sum_{i=1}^{k} \tilde{X}_i\varphi_i'(t) \\ \sum_{i=1}^{k} \tilde{Y}_i\varphi_i'(t) \end{pmatrix}}\, dt, \tag{IV.18}$$

Les longueurs d'arêtes obtenues nous permettent de déterminer les éléments devant être raffinés. Le paragraphe suivant s'intéresse à la découpe d'un élément en sous-éléments.

Remarque IV.3.3 (*Estimateur d'erreur d'ordre élevé*)

Il serait, bien entendu, plus intéressant d'utiliser un estimateur d'erreur d'ordre élevé. Supposons que l'on travaille à l'ordre k et tentons d'étendre la définition de l'estimateur d'erreur présenté paragraphe 3.2.1. On note D^k l'opérateur différentiel d'ordre k.

Par développement de Taylor avec reste intégral d'ordre $k-1$, on majore l'erreur

d'interpolation d'ordre k :

$$|\mathcal{P}_e(x)| \leq |\int_0^1 \frac{(1-t)}{(k-1)!}^{(k-1)} \langle \vec{ax}, \underbrace{D^k u\,(x+t\,\vec{ax})}_{(I)}\, \vec{ax}\rangle dt|. \tag{IV.19}$$

On constate que la dérivée $k^{\text{ème}}$ de la solution (terme (I)) contient des dérivées d'ordre élevé, la construction d'une métrique majorant (I) n'est donc pas triviale et reste à étudier.

3.3.3 Raffinement d'un élément courbe

Comme dans la section 2, les algorithmes présentés s'appliquent à des surfaces de Bernstein classiques. Nous utiliserons donc à nouveau la propriété de transformation projective si les fonctions de base considérées sont rationnelles. Nous présenterons d'abord la méthode de subdivision mono-dimensionnelle ($đ = 1$) employée puis, les méthodes bi-dimensionnelles ($đ = 2$). Bien que le raffinement d'un maillage soit plus complexe en dimension 3 que la h-adaptation bidimensionnelle que nous présentons ici, l'extension au cas tridimensionnel de la subdivision d'un unique élément est directe. Nous la laisserons donc à la charge du lecteur. Nous avons vu précédemment que notre critère de raffinement porte sur les arêtes, ainsi, quelle que soit la dimension considérée, nous subdiviserons en réalité les arêtes de nos éléments et, par conséquence, nos éléments. Nous imposerons deux contraintes à ces subdivisions. D'une part, un segment sera subdivisé en son milieu au sens paramétrique. D'autre part, les éléments issus d'un raffinement seront tous de même type que leur parent (*c.-à-d.* qu'un triangle est découpé en sous-triangles, un quadrangle en sous-quadrangles, *etc.*).

3.3.3.1 Cas mono-dimensionnel : subdivision d'un segment courbe

Soit k l'ordre de notre segment de Bézier et b le polygone de contrôle qui lui est associé. On numérote b_1 le premier point de contrôle et b_n le dernier point de contrôle du polygone de contrôle. Les points de contrôle intermédiaires sont numérotés en parcourant le polygone de contrôle du point b_1 au point b_n. On note $\widehat{b}$ le polygone de contrôle qui sera associé au premier sous-segment enfant de la subdivision et $\bar{b}$ celui qui sera associé au second. Les polygones de chaque sous-segment seront formés de k points de contrôle. Notre segment courbe initial, notons-le (S), est l'image du vecteur nodal ouvert $[0_k, 1_k]$. Ce segment représente l'intervalle paramétrique $[0,1]$. On souhaite partager (S) en son milieu paramétrique donc les segments enfants (S_1) et (S_2) seront les images respectives

des intervalles paramétriques $\left[0,\frac{1}{2}\right]$ et $\left[\frac{1}{2},1\right]$. Comme expliqué paragraphe 2.2.2, notre élément paramétrique est séparé si l'on y insère k fois le nœud $\frac{1}{2}$. En conséquence, le patch physique (S) est également subdivisé. L'algorithme IV.3.1 (en pseudo-code), illustré figure IV.3.5, permet de calculer les points de contrôle tels que :

$$\begin{cases} S(\xi) = S_1(2\xi) = \sum_{i=1}^{k} \widehat{b}_i \varphi_i(2\xi), & \text{si } \xi \in \left[0,\frac{1}{2}\right] \\ S(\xi) = S_2(2\xi-1) = \sum_{i=1}^{k} \overline{b}_i \varphi_i(2\xi), & \text{si } \xi \in \left[\frac{1}{2},1\right] \end{cases}$$

Algorithme IV.3.1: de Casteljau : subdivision d'une courbe de Bézier en $\xi = \frac{1}{2}$

Données : b : **tableau de rang** 1 **et de profil** (k), k : **entier**;
Sorties : $\widehat{b}, \overline{b}$: **tableaux de rang** 1 **et de profil** (k);

Fonction DeCasteljau (b,k)
- **Pour** l ***de*** 1 ***à*** $k-1$ **Faire**
  - **Pour** i ***de*** 1 ***à*** l **Faire**
    - $\text{tmp}_i \leftarrow b_i$; $\text{tmp}_{(k+l+1-i)} \leftarrow b_{k+l-i}$;
  - **Pour** i ***de*** $1+l$ ***à*** $l+k-1$ **Faire**
    - $\text{tmp}_i \leftarrow \frac{1}{2}\left(b_i + b_{(i-1)}\right)$;
  - $b \leftarrow \text{tmp}$;
- **Pour** l ***de*** 1 ***à*** k **Faire**
  - $\widehat{b}_l \leftarrow b_l$; $\widehat{b}_l \leftarrow b_{k+l-1}$;
- **Résultat** : $(\widehat{b}, \overline{b})$;

On peut noter que la version que nous donnons de l'algorithme de De Casteljau est une version simplifiée pour $\xi = \frac{1}{2}$. Par ailleurs, la figure IV.3.5 montre que chaque étape de cet algorithme revient, en fait, à former un nouveau polygone de contrôle à partir des barycentres des côtés du polygone de contrôle initial et de ses extrémités.

Détaillons le procédé. Nous souhaitons subdiviser le polygone de contrôle b. On sauvegarde les extrémités de notre polygone b, donc les points de contrôle b_1 et b_k et on calcule les points de coordonnées barycentriques ξ pour chacun des sous-segments $[b_l, b_{l+1}]$. L'ensemble barycentres calculés forme un nouveau polygone de contrôle, comptant maintenant $(k-1)$ points.

Nos deux extrémités ont des indices différents, on dénotera par *extrémité gauche* l'ex-

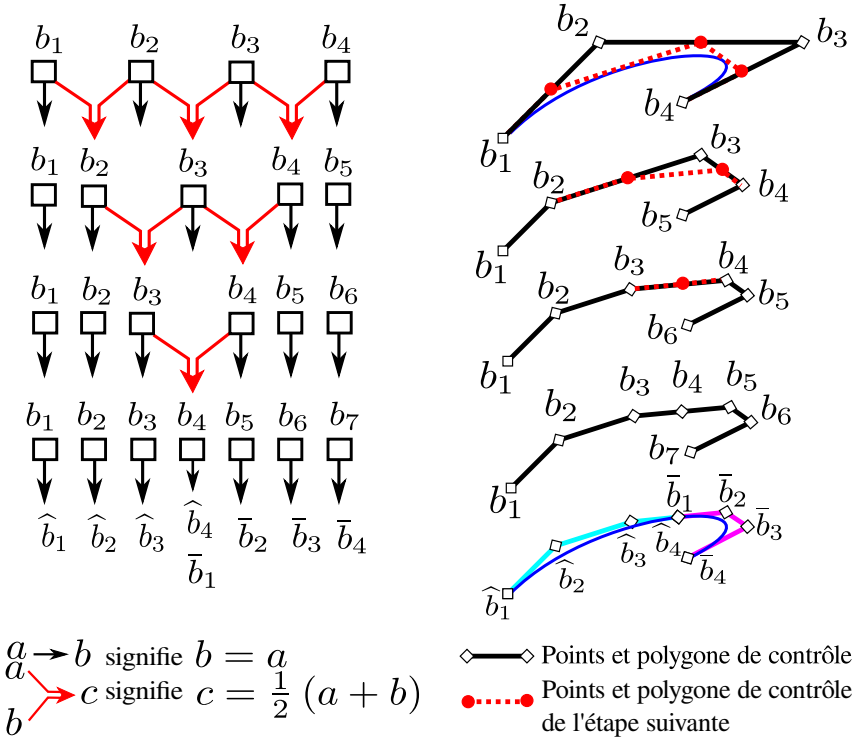


Figure IV.3.5 – Algorithme linéique de De Casteljau pour $k = 4$ (à gauche). Polynômes et points de contrôle trouvés à chaque étape (à droite).

trémité d'indice minimal et par *extrémité droite* l'extrémité d'indice maximal.

On réitère l'opération précédente, *c.-à-d.* que l'on sauvegarde les extrémités du nouveau polygone et que l'on calcule les points de coordonnées barycentriques ξ pour chacun de ses sous-segments. Le polygone de contrôle obtenu est alors composé de $(k-2)$ points. En réitérant le processus $k-1$ fois, on obtient un polygone à un point. En associant ce point aux extrémités gauches des $(k-1)$ polygones de contrôle utilisés, on trouve le polygone de contrôle $\widehat{b}$ et en l'associant aux extrémités droites, on trouve le polygone de contrôle $\bar{b}$.

Par ailleurs, comme nous l'avons expliqué paragraphe 2.1.1, seuls les rapports entre les longueurs des intervalles nodaux influent sur un patch (les coordonnées nodales n'ont, en elles-mêmes, aucune influence, ni sur le calcul des fonctions de base, ni sur le mapping $\mathcal{NURBS}$). Ainsi, n'importe quel vecteur nodal ouvert à un élément (*c.-à-d.* n'importe quel segment paramétrique) pourra être réécrit sous forme du vecteur nodal $[0_k, 1_k]$ et subdivisé en deux sous-segments enfants de mêmes longueurs par l'algorithme de De Casteljau précédemment décrit.

On peut également remarquer que l'algorithme de De Casteljau, utilisé ici pour subdiviser un segment courbe, a initialement pour but d'évaluer la valeur d'une courbe de Bézier en un point ξ (ici, $\xi = \frac{1}{2}$) du segment paramétrique. Effectivement, le dernier point de contrôle obtenu, correspond, après subdivision de (S), respectivement au dernier et au premier point de contrôle de (S_1) et (S_2) (*c.-à-d.* $\widehat{b}_k$ et $\bar{b}_1$). Or nos segments sont associés à des vecteurs nodaux ouverts, les extrémités de leur polygone de contrôle sont donc interpolées. Ce qui signifie que, pour $\xi = \frac{1}{2}$, $S(\xi) = \widehat{b}_k = \bar{b}_1$.

Si notre courbe est une courbe rationnelle, on précède bien sûr, l'algorithme d'une étape de passage en coordonnées homogènes (coordonnées stockées dans les b_i) et on le fait suivre d'une autre étape, appliquée aux $\widehat{b}_i$ et $\bar{b}_i$, de retour aux coordonnées réelles.

C'est à partir de cette méthode de subdivision de segment que seront subdivisés les éléments de dimension supérieure.

3.3.3.2 Cas bidimensionnel : subdivision d'une surface courbe

Nous souhaitons diviser nos éléments en sous-éléments de mêmes types.

Comme le partage d'un élément est une conséquence du partage d'une ou plusieurs des arêtes de l'élément, on souhaite également que la ou les arêtes découpées le soient en leur milieu paramétrique et que les sous-éléments paramétriques obtenus soient superposables. Le nombre de configurations possibles est donc limité.

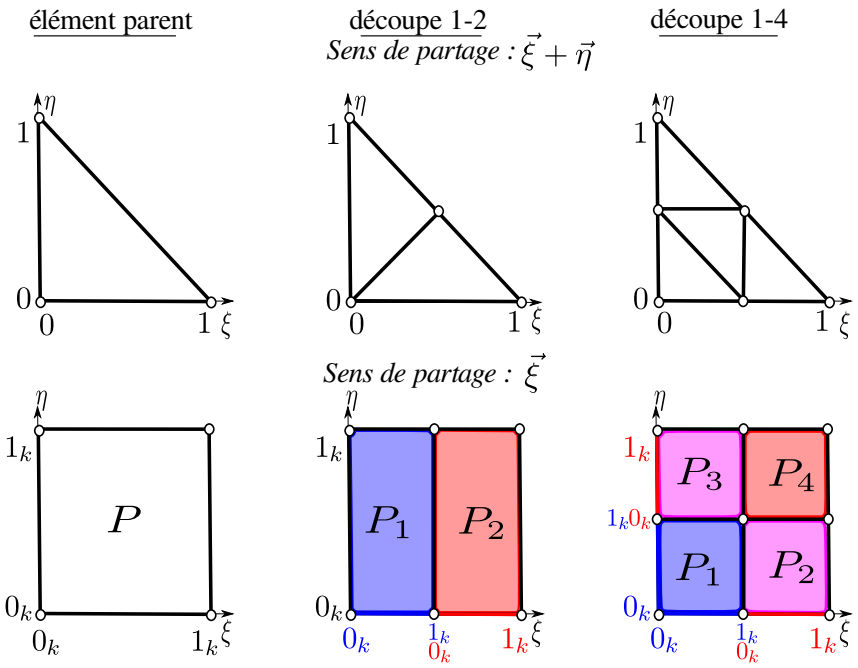


Figure IV.3.6 – Motifs de division élémentaire. Pour le quadrangle, les segments $[0,1]$ sont vus comme les vecteurs nodaux ouverts $[0_k, 1_k]$.

En fait, en dimension 2, quelque soit l'élément, seulement deux types de découpes satisferont nos contraintes. La première, nommée *découpe* $1-2$, consiste à subdiviser l'élément en deux éléments enfants (figure IV.3.6, colonne du milieu). La seconde découpe possible, notée *découpe* $1-4$, permet d'obtenir quatre éléments enfants (figure IV.3.6, colonne de droite).

Remarque IV.3.4 (*Partage de deux des arêtes d'un élément*)

Il est également possible de subdiviser deux des arêtes d'un élément (voir figure IV.3.7). Nous n'utiliserons pas ce type de subdivision dans notre travail. D'un part les éléments créés ne satisfont pas notre critère de superposabilité, d'autre part ils ont tendance à être de forme très allongés et pour finir, deux configurations de subdivisions différentes sont possibles sans critère de choix (figure IV.3.7 colonnes du milieu et de droite).

3.3.3.2.1 Subdivision des quadrangles

Nous passerons en quelques mots sur la subdivision d'un quadrangle en sous-quadrangles car c'est une opération très simple si l'on considère celui-ci comme le produit tensoriel de deux vecteurs nodaux (ici, on suppose que ces deux vecteurs contiennent un unique intervalle).

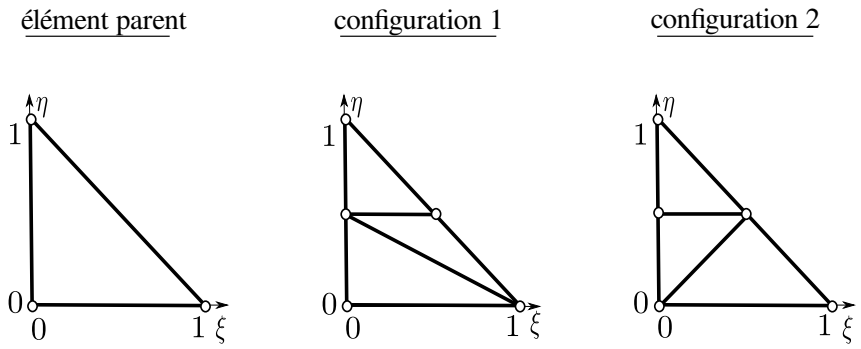


Figure IV.3.7 – Subdivision de deux arêtes d'un élément.

Une *découpe* $1-2$ se réalise en subdivisant le vecteur nodal tangent au sens de partage (si l'on considère le sens de partage comme celui dans lequel on a plus d'éléments qu'initialement). Cette découpe est présentée figure IV.3.6, ligne du bas, seconde colonne. On suppose que notre vecteur nodal, ξ, contient n éléments et que le vecteur nodal suivant la seconde direction paramétrique, η, contient m éléments. On utilise une numérotation fonctionnelle donc on note b_{ij} les $n \times m$ points de contrôle du patch. On insère k fois le nœud $\frac{1}{2}$ dans le vecteur nodal $\xi = [0_k, 1_k]$. Comme l'insertion à lieu dans la direction ξ, on aura, à chaque insertion, $((n+1) \times m)$ nouveaux $\mathcal{CP}s$ notés $\bar{b}_{ij}$. On calcule ces $\mathcal{CP}s$ en appliquant la formule d'insertion de nœud (équation (II.2.3)) aux b_{ij} avec j fixé. On applique cette formule pour j allant de 1 à m, donc m fois. Notre patch (P) est alors partagé en deux nouveaux patchs (P_1) et (P_2) tels que $(P_1) \cup (P_2) = (P)$ et $(P_1) \cap (P_2) = \emptyset$.

La *découpe* $1-4$ suit le même principe mais cette fois, on subdivise consécutivement les deux vecteurs nodaux générant le patch.

3.3.3.2.2 Subdivision d'un triangle en deux sous-triangles

Consacrons-nous maintenant au principal objectif de ce paragraphe, la description de la méthode de subdivision des triangles courbes.

Détaillons tout d'abord la *découpe* $1-2$. Les explications fournies sont illustrées figure IV.3.8. Dans ce type du subdivision, seule l'une des arêtes du triangle est coupée.

Soit (ABC) notre triangle et soit $\Gamma = AB$ l'arête que l'on souhaite couper en deux arêtes, Γ_1 et Γ_2, de mêmes longueurs paramétriques.

Considérons notre polygone de contrôle surfacique complet sous forme de *couches parallèles* de polygones de contrôle linéiques (voir figure IV.3.8). Tout d'abord, expliquons ce que nous entendons par le terme de *couches parallèles*. Si l'on définit la distance entre deux points de contrôle d'une grille comme le nombre minimum de sous-segments de la

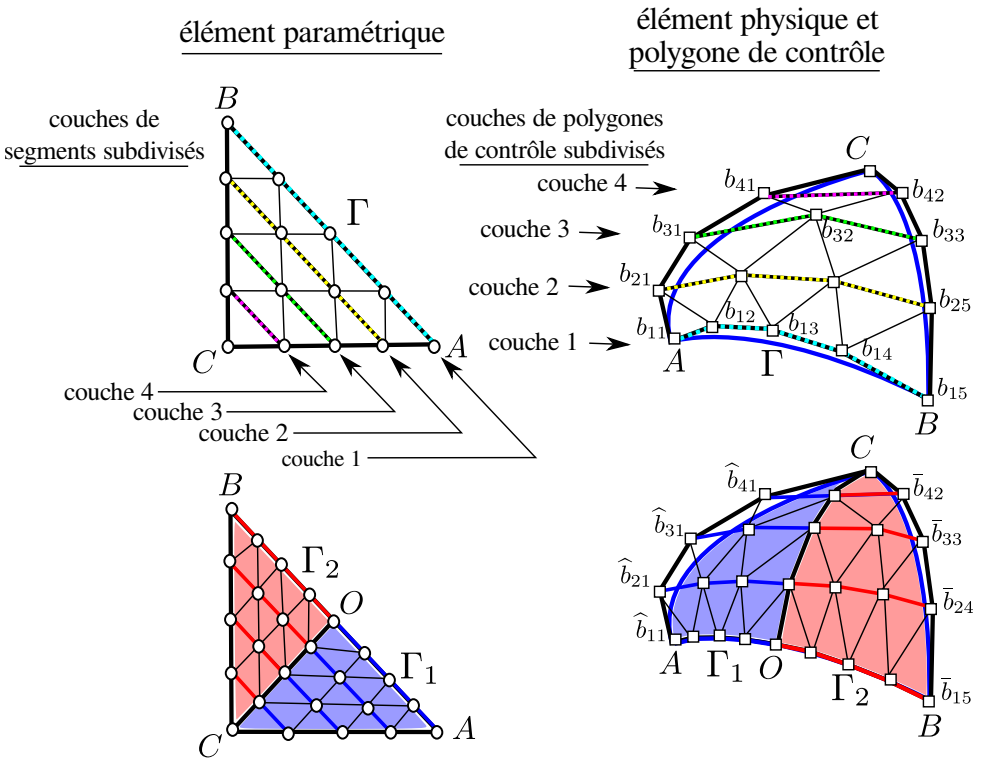


Figure IV.3.8 – Exemple de *découpe* $1-2$ à l'ordre $k=5$ sur $\Gamma = AB$.
Subdivision du point de vue de l'espace paramétrique à gauche.
Subdivision du polygone de contrôle et de l'élément physique à droite.

grille de contrôle à parcourir pour aller d'un point de contrôle à l'autre, alors une couche est dite parallèle au polygone de contrôle de Γ si la distance, évaluée entre chacun des points de contrôle de la couche et le point de contrôle le plus proche appartenant au polygone de contrôle de Γ, est constante.

Comme les polygones de contrôle des arêtes AC et AB comptent k points de contrôle, on aura, en comptant une couche pour l'arête Γ, $(k-1)$ couches de points de contrôle (le dernier point de contrôle des deux arêtes étant le sommet C, la $k^{\text{ème}}$ couche est réduite à un point qui n'a pas besoin d'être recalculé et sera donc ignorée). De plus, comme nous travaillons sur un triangle, chaque couche contiendra un point de contrôle de moins que celle qui la précède. Ainsi, en numérotant Γ comme étant la couche 1, la $l^{\text{ème}}$ couche de points de contrôle (*c.-à-d.* le $l^{\text{ème}}$ polygone de contrôle) contiendra $(k-(l-1))$ points.

Chacune de ces couches de points de contrôle est associée à un segment du triangle paramétrique parallèle à Γ dans cet espace. On a donc également $(k-1)$ couches de segments paramétriques et l'image du $l^{\text{ème}}$ segment associé au $l^{\text{ème}}$ polygone de contrôle est une courbe de la surface $\mathcal{NURBS}$ que l'on souhaite partager.

Notre triangle physique est découpé en deux triangles enfants si le triangle paramétrique associé est découpé en deux triangles paramétriques enfants. Dans ce cas, chacune des $(k-1)$ couches de segments paramétriques est découpée en son milieu en deux sous-segments paramétriques. On applique donc l'algorithme IV.3.1 de De Casteljau à chaque couche de points de contrôle.

Rappelons que pour n points de contrôle, un vecteur nodal contient, à l'ordre k, $(n+k)$ nœuds, de plus, comme nous travaillons uniquement avec des vecteurs nodaux ouverts à un intervalle, le nombre de nœuds de nos vecteurs est aussi égal à $2k$. Ceci implique que, à nombre de points de contrôle imposé, notre ordre de travail, $\tilde{k}$ est également imposé , à savoir, $\tilde{k}=n$. Dans notre cas, comme le polygone de contrôle associé au $l^{\text{ème}}$ segment contient $(k-(l-1))$ points de contrôle, alors l'ordre du segment sera égal à $(k-l+1)$.

On note b_l le $l^{\text{ème}}$ polygone de contrôle (avant subdivision) et, de manière analogue au polygone associé à un segment de Bézier (voir sous-paragraphe 3.3.3.1), on note $(b_{lj})_{j=1,k-l+1}$ ses points de contrôle. Après subdivision, on note $(\widehat{b}_{lj})_{j=1,k-l+1}$ les points du $l^{\text{ème}}$ polygone de contrôle de gauche (*c.-à-d.* les points du polygone associé au $l^{\text{ème}}$ sous-segment de gauche) et $(\bar{b}_{lj})_{j=1,k-l+1}$ ceux associés au $l^{\text{ème}}$ polygone de droite. Le polygone de contrôle surfacique associé, après subdivision, au sous-triangle physique de gauche, est formé des $(k-1)$ couches de polygones $(\widehat{b}_l)$ et du sommet opposé à Γ, C. De même, le sous-triangle physique de droite, est formé des $(k-1)$ couches de polygones $(\bar{b}_l)$ et de C. L'algorithme IV.3.2 résume la *découpe* $1-2$ que nous venons de décrire.

Algorithme IV.3.2: *découpe* $1-2$: partage d'un triangle en 2 sous-triangles

Données : b : **tableau de rang** 2 **et de profil** (k,k), k : **un entier**;
Sorties : $\widehat{b},\bar{b}$: **tableaux de rang** 2 **et de profil** (k,k);

Fonction DECOUPE1-2 (b,k)
  Pour l ***de*** 1 ***à*** $k-1$ **Faire**
    $(\widehat{b}_{l*},\bar{b}_{l*}) \leftarrow$ DECASTELJAU $(b_{l*},k+l-1)$;
  Résultat : $(\widehat{b},\bar{b})$;

Remarque IV.3.5 (*Notations*)

Soit (ABC) un triangle. On désignera respectivement comme le *premier côté*, le *second côté* et le *troisième côté* de (ABC) les côtés AB, BC et CA. Évidemment, l'algorithme précédemment présenté permet de subdiviser indifféremment, la première, deuxième ou troisième arête de notre triangle. Nous considèrerons, quand nous parlerons de *découpe* $1-2$ de (ABC), que l'arête subdivisée est la première du triangle. Une simple rotation sur les sommets du triangle nous permettra de désigner la subdivision des autres côtés (par exemple, si nous souhaitons partager (ABC) sur sa troisième arête, nous parlerons de *découpe* $1-2$ de (CAB). Nous noterons également $b_{(ABC)}$ le polygone de contrôle associé au triangle (ABC) et tel que b_{1j} est le $j^{\text{ème}}$ point de contrôle du polygone de contrôle associé à l'arête que l'on coupe, AB. $b_{(ABC)} = b_{(ABC)ij}$ est donc le polygone de contrôle surfacique de (ABC), numéroté de telle sorte que, pour i croissant, on parcourt les couches de polygones linéiques parallèles à AB.
Avec cette notation, DECOUPE1-2 $(b_{(ABC)},k) = (b_{(AOC)},b_{(OBC)})$.

3.3.3.2.3 Subdivision d'un triangle en 4 sous-triangles

Nous souhaitons maintenant découper notre triangle en quatre sous-triangles.

Chaque arête doit être subdivisée. Si l'on met de côté le fait que l'on souhaite avoir des éléments enfants superposables dans l'espace paramétrique, une première solution, intuitive, consiste à découper notre triangle (ABC) en deux sous-triangles par *découpe* $1-2$ puis à subdiviser, à nouveau par *découpe* $1-2$, chacun des deux sous-triangles.

L'algorithme correspondant à ce type de découpe, que nous nommerons *découpe* $1-4$ *spider* en raison de la forme particulière des éléments obtenus, est fourni algorithme IV.3.3 et illustré figure IV.3.9.

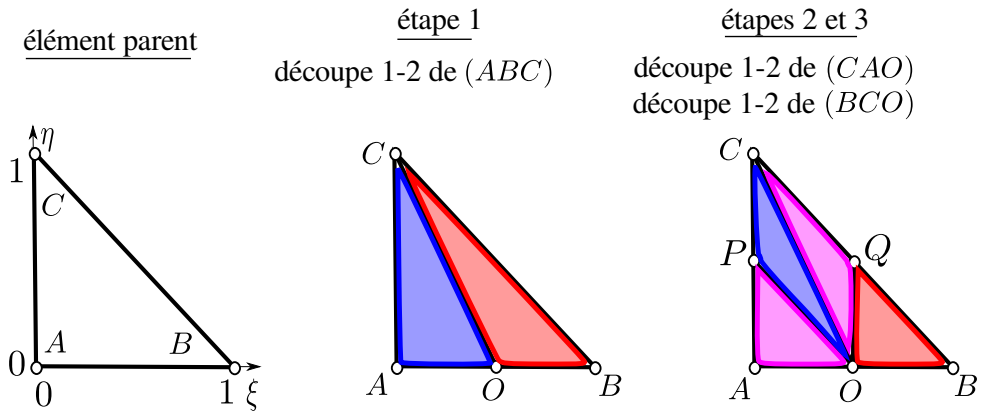


Figure IV.3.9 – Motifs de division élémentaire de la *découpe* $1-4$ *spider*. On obtient les 4 triangles (CPO), (PAO), (BQO) et (QCO) après 3 étapes de *découpe* $1-2$.

Algorithme IV.3.3: *découpe* $1-4$ *spider* : partage d'un triangle en 4

Données : $b_{(ABC)}$: **tableau de rang** 2 **et profil** (k,k),
k : **un entier**;
Sorties : $b_{(CPO)}, b_{(OBQ)}, b_{(PAO)}, b_{(OQC)}$: **tableaux de rang** 2 **et profil** (k,k);

Fonction DECOUPE1-4SPIDER (b,k)

$(b_{(AOC)}, b_{(OBC)}) \leftarrow$ DECOUPE1-2 $(b_{(ABC)})$;
$(b_{(CPO)}, b_{(PAO)}) \leftarrow$ DECOUPE1-2 $(b_{(AOC)})$;
$(b_{(OBQ)}, b_{(OQC)}) \leftarrow$ DECOUPE1-2 $(b_{(OBC)})$;
Résultat : $(b_{(CPO)}, b_{(OBQ)}, b_{(PAO)}, b_{(OQC)})$;

Remarque IV.3.6 (*Forme des éléments issus d'une découpe 1-4 spider*)

La nouvelle arête créée, CO, est de longueur $\frac{CA+CB}{2}$, *c.-à-d.* qu'elle est plus longue que la plus petite des deux arêtes CA et CB. Elle est donc également deux fois plus longue que la plus courte des arêtes CP et CQ. Ceci explique que cette découpe crée des éléments allongés, en particulier après plusieurs vagues de subdivisions.

Remarque IV.3.7 (*Faiblesses de la découpe 1-4 spider*)

Dans le cas particulier où CA est de l'ordre de grandeur de CB et où ces deux arêtes sont plus longues que AC, notre nouvelle arête est de l'ordre de grandeur de la plus longue arête de notre triangle parent. La découpe réalisée aura donc été inutile. Ce phénomène est visible dans l'exemple fourni figure IV.3.9.

Dans un cas plus général, le fait qu'il ait fallu découper notre triangle en quatre enfants implique qu'au moins deux des arêtes de notre triangle ont été estimées trop longues, il est donc fort probable que notre nouvelle arête CO soit elle aussi trop longue et qu'elle doive être découpée dans la prochaine vague de raffinement.

Nous ne pouvons donc pas nous contenter de ce type de découpe et il nous faut définir la *découpe* $1-4$ classique présentée en introduction de ce paragraphe (figure IV.3.6). Cette découpe à l'avantage de créer des triangles enfants superposables dans l'espace paramétrique et, dont l'arête la plus longue, dans l'espace physique, est, dans le pire des cas, de l'ordre de grandeur de la moitié de l'arête la plus longue de l'élément parent.

Le document [28] donne une méthode permettant d'obtenir la *découpe* $1-4$ classique (que nous nommerons tout simplement *découpe* $1-4$) à partir de la *découpe* $1-4$ *spider*. Cette méthode nécessite de présenter un autre type de subdivision que nous nommerons *découpe* $1-3$. Soit (ABC) notre élément initial, la *découpe* $1-3$ consiste tout simplement à insérer un nouveau point, O, ce qui crée trois nouveaux triangles, chacun formé d'une des arêtes de l'élément parent et ayant pour troisième sommet O. les arêtes de l'élément parent ne sont donc pas modifiées. Cette découpe est illustrée figure IV.3.10.

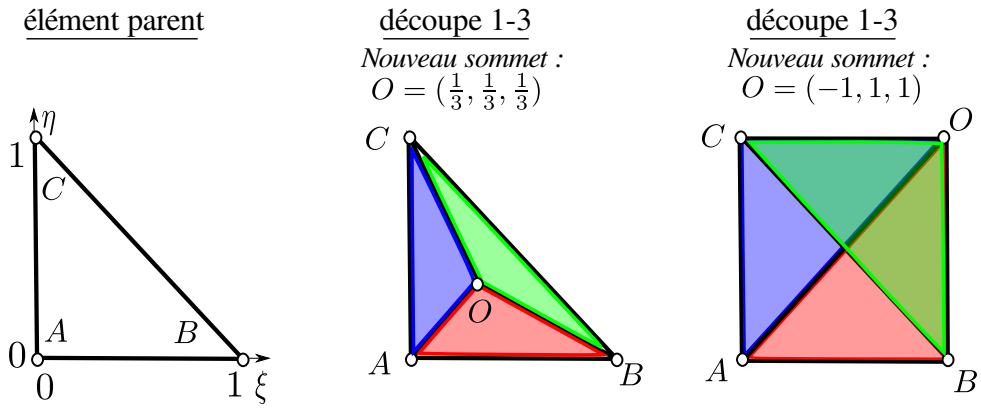


Figure IV.3.10 – Motifs de division élémentaire de la *découpe* $1-3$ pour le nouveau sommet O. On obtient les 3 triangles (ABO), (BCO) et (CAO). Exemples pour $O = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$ au milieu et $O = (-1, 1, 1)$ à droite.

L'algorithme de De Casteljau pour les triangles de Bernstein (voir algorithme IV.3.4) permet de calculer les polygones de contrôle associés à chacun des nouveaux sous-triangles. Le principe de cet algorithme est très proche de celui de l'algorithme de De Casteljau pour

les courbes de Bézier.

Notons $\beta = (\beta_1, \beta_2, \beta_3)$ les cordonnées dans le repère barycentrique lié à (ABC), du sommet que l'on souhaite insérer, O. On sauvegarde les points de contrôle externes de l'élément, c'est-à dire les points (b^k_{lmn}) tels qu'au moins l'un des indices l, m ou n soit nul. On considère alors les sous triangles de la grille de contrôle de notre élément et, pour chaque sous-triangle de sommets b^k_{lmn}, $b^k_{(l-1)(m+1)n}$ et $b^k_{(l-1)m(n+1)}$, on calcule le point de coordonnées barycentriques β dans le repère lié au sous-triangle. Les $\frac{(k-1)k}{2}$ barycentres calculés forment une nouvelle grille de contrôle triangulaire. On numérote ces points de contrôle (b^{k-1}_{lmn}).

Le calcul des barycentres des sous-triangles est montré figure IV.3.11, étape 1. On numérote la nouvelle grille de contrôle lors de son utilisation, à l'étape 2.

Chacun des trois sommets de cette nouvelle grille, donc des points (b^{k-1}_{lmn}) tels que deux des indices l, m et n soient nuls, appartiendra au polygone de contrôle de l'une des nouvelles arêtes AO, BO et CO.

On réitère le processus, *c.-à-d.* que l'on considère notre nouvelle grille de contrôle, on sauvegarde ses points de contrôle externes et on calcule les points de coordonnées barycentriques β dans les sous-triangles $\left(b^{k-1}_{lmn} b^{k-1}_{(l-1)(m+1)n} b^{k-1}_{(l-1)m(n+1)}\right)$. On applique $(k-1)$ itérations on obtient un unique barycentre noté b^1_{lmn} et confondu à notre nouveau sommet, O.

La grille de contrôle formée de ce barycentre et de tous les points externes précédemment sauvegardés peut alors être partagée en trois grilles de contrôle. La première grille de contrôle, associée au triangle (ABO), la seconde grille de contrôle, associée au triangle (BCO) et enfin, la dernière grille de contrôle, associée au triangle (CAO), auront respectivement pour points de contrôle externes les points :

$$
\begin{array}{lllll}
b^{k-1}_{lm0} & \cup & (b^i_{l00})_{i=1,k-1} & \cup & (b^i_{0m0})_{i=1,k-1} \\
b^{k-1}_{0mn} & \cup & (b^i_{00n})_{i=1,k-1} & \cup & (b^i_{0m0})_{i=1,k-1} \\
b^{k-1}_{l0n} & \cup & (b^i_{l00})_{i=1,k-1} & \cup & (b^i_{00n})_{i=1,k-1}
\end{array}
$$

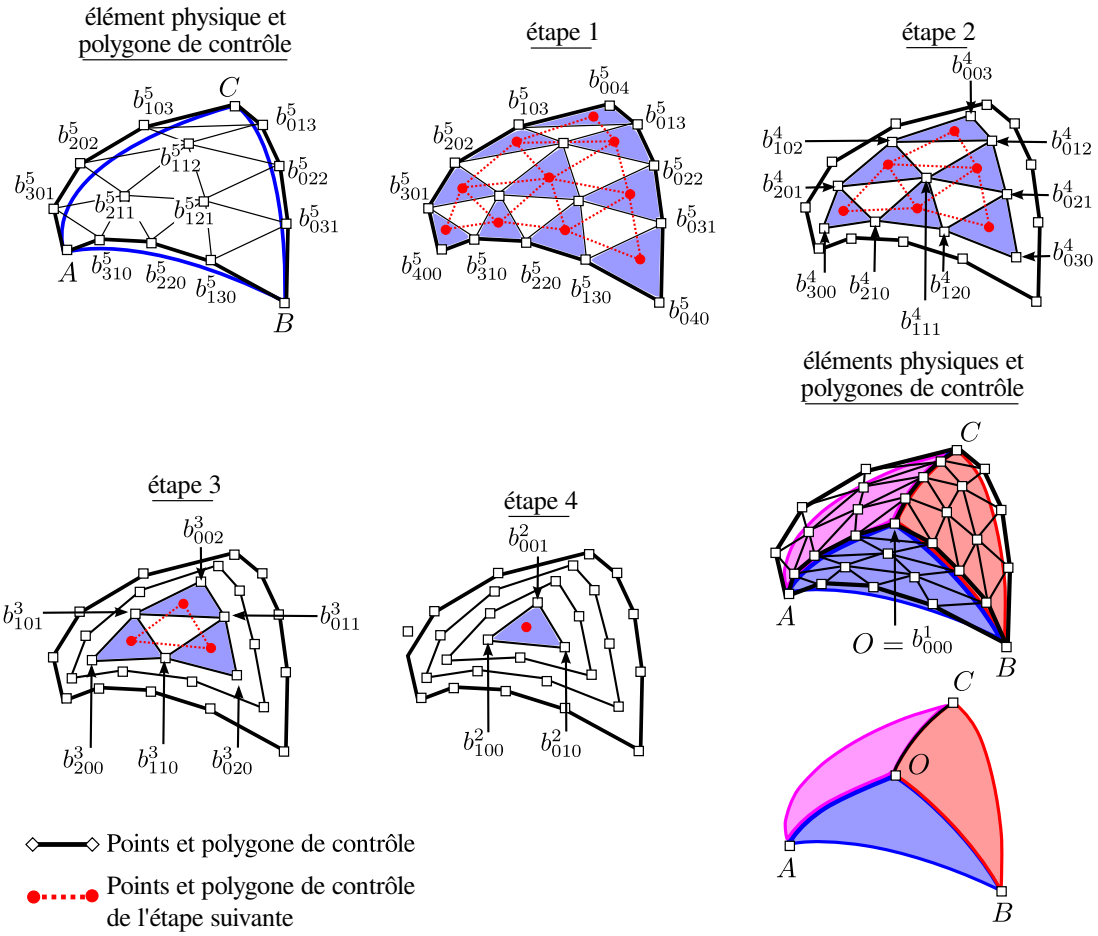


Figure IV.3.11 – Exemple de *découpe* $1-3$ à l'ordre $k=5$ pour $\beta=\left(\frac{1}{3},\frac{1}{3},\frac{1}{3}\right)$.
À chaque étape on représente en noir le polygone de contrôle considéré, en bleu les sous-triangles de ce polygone et en rouge les points de coordonnées barycentriques β dans chaque sous-triangle et le polygone de contrôle de l'étape suivante.

Algorithme IV.3.4: *découpe* $1-3$: partage d'un triangle en 3

Données : $b^k_{(ABC)}$: **tableau de rang** 3 **et profil** (k,k,k),
β : **tableau de** 3 **flottants**;
k : **un entier**;

Sorties : $b_{(ABO)}, b_{(BCO)}, b_{(CAO)}$: **tableaux de rang** 2 **et profil** (k,k);

Fonction DECOUPE1-3 (b,k)

Pour i ***de*** $k-1$ ***à*** 1 **Faire**

Pour $(lmn) \in \Delta_i$ **Faire**

$b^i_{lmn} \leftarrow \beta_1\, b^{i+1}_{(l+1)mn} + \beta_2\, b^{i+1}_{l(m+1)n} + \beta_3\, b^{i+1}_{lm(n+1)}$;

/* $\widehat{b}$ stocke les points de contrôle de $b_{(ABO)}$ */

/* $\bar{b}$ stocke les points de contrôle de $b_{(BCO)}$ */

/* $\check{b}$ stocke les points de contrôle de $b_{(CAO)}$ */

$\widehat{b}^k_{(k-1)00} \leftarrow b^k_{(k-1)00}$; $\widehat{b}^k_{0(k-1)0} \leftarrow b^k_{0(k-1)0}$; $\widehat{b}^k_{00(k-1)} \leftarrow b^1_{000}$;
$\bar{b}^k_{(k-1)00} \leftarrow b^k_{0(k-1)0}$; $\bar{b}^k_{0(k-1)0} \leftarrow b^k_{00(k-1)}$; $\bar{b}^k_{00(k-1)} \leftarrow b^1_{000}$;
$\check{b}^k_{(k-1)00} \leftarrow b^k_{00(k-1)}$; $\check{b}^k_{0(k-1)0} \leftarrow b^k_{(k-1)00}$; $\check{b}^k_{00(k-1)} \leftarrow b^1_{000}$;

Pour i ***de*** 1 ***à*** $k-2$ **Faire** /* les arêtes connues */

$\widehat{b}^k_{i(k-1-i)0} \leftarrow b^k_{i(k-1-i)0}$;
$\bar{b}^k_{i(k-1-i)0} \leftarrow b^k_{0i(k-1-i)}$;
$\check{b}^k_{i(k-1-i)0} \leftarrow b^k_{i0(k-1-i)}$;

Pour i ***de*** $k-2$ ***à*** 1 **Faire** /* les nouvelles arêtes */

$\widehat{b}^k_{0i(k-1-i)} \leftarrow b^{i+1}_{0i0}$; $\widehat{b}^k_{(k-1-i)0i} \leftarrow b^{k-i}_{(k-1-i)00}$;
$\bar{b}^k_{0i(k-1-i)} \leftarrow b^{i+1}_{00i}$; $\bar{b}^k_{(k-1-i)0i} \leftarrow b^{k-i}_{0(k-1-i)0}$;
$\check{b}^k_{0i(k-1-i)} \leftarrow b^{i+1}_{i00}$; $\check{b}^k_{(k-1-i)0i} \leftarrow b^{k-i}_{00(k-1-i)}$;

Pour i ***de*** $k-1$ ***à*** 2 **Faire** /* l'intérieur des triangles */

Si $(lmn) \in \Delta_{k-1}$ ***Et*** $\widehat{b}_{lmn}$ *non rempli* **Alors**

$\widehat{b}^k_{lmn} \leftarrow b^i_{i(k-1-i)0}$;
$\bar{b}^k_{lmn} \leftarrow b^i_{0i(k-1-i)}$;
$\check{b}^k_{lmn} \leftarrow b^i_{(k-1-i)0i}$;

$b_{(ABO)} \leftarrow \widehat{b}^k$; $b_{(BCO)} \leftarrow \bar{b}^k$; $b_{(CAO)} \leftarrow \check{b}^k$;

Résultat : $(b_{(ABO)}, b_{(BCO)}, b_{(CAO)})$;

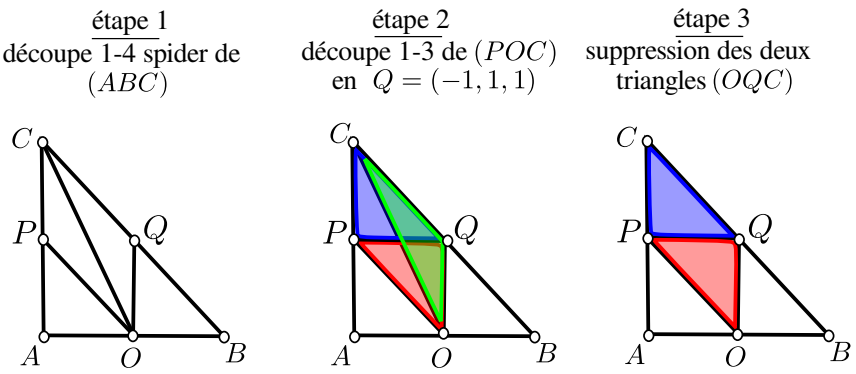


Figure IV.3.12 – Motifs de division élémentaire de la *découpe* $1-4$ classique.

Maintenant que nous possédons tous les outils pour réaliser notre *découpe* $1-4$, nous pouvons la détailler en quelques mots.

Les trois étapes de cette découpe sont représentées figure IV.3.12. L'algorithme correspondant est l'algorithme IV.3.5

L'élément parent (ABC) est découpé en 4 sous-éléments par *découpe* $1-4$ *spider*. On considère ensuite l'un des deux triangles ayant pour côté la nouvelle arête CO (donc de (CPO) ou (QCO)). On découpe cet élément par une *découpe* $1-3$ dont le nouveau sommet est le sommet opposé à CO dans l'autre des deux triangles (*resp.* Q de coordonnées barycentriques β_Q dans (CPO) ou P de coordonnées barycentriques β_P dans (QCO)). On obtient 3 nouveaux triangles, (CPQ), (POQ) et (COQ). Pour finir on supprime tous les triangles ayant CO pour côté (donc le triangle (COQ) issu de la *découpe* $1-4$ *spider* et le même triangle, (COQ), issu de la *découpe* $1-3$. On obtient les 4 triangles (PAO), (BQO), (CPQ) et (POQ).

Algorithme IV.3.5: *découpe* $1-4$: partage d'un triangle en 4

Données : $b_{(ABC)}$: **tableau de rang** 2 **et profil** (k,k),
k : **un entier**;
Sorties : $b_{(PAO)}, b_{(OBQ)}, b_{(CPQ)}, b_{(POQ)}$: **tableaux de rang** 2 **et profil** (k,k);
Fonction DECOUPE1-4 (b,k)

$(b_{(CPO)}, b_{(OBQ)}, b_{(PAO)}, b_{(OQC)}) \leftarrow$ DECOUPE1-4SPIDER $(b_{(ABC)})$;
$\beta_Q \leftarrow$ Coordonnées de Q dans le repère barycentrique lié à (CPO);
Supprime : $b_{(OQC)}$;
$(b_{(CPQ)}, b_{(POQ)}, b_{(OCQ)}) \leftarrow$ DECOUPE1-3 $(b_{(CPO)}, \beta_Q, k)$;
Supprime : $b_{(OCQ)}$, β_Q;
Résultat : $(b_{(PAO)}, b_{(OBQ)}, b_{(CPQ)}, b_{(POQ)})$;

3.3.4 Algorithme de raffinement

Nous terminerons cette sous-section dédiée au raffinement d'un maillage courbe d'ordre élevé par l'algorithme de raffinement d'un maillage. On considère qu'en entrée de l'algorithme, on fournit le maillage courbe d'ordre élevé et la métrique physique associée.

Algorithme IV.3.6: raffinement d'un maillage

- Calcul des métriques de contrôle associées aux nœuds du maillage courbe.
- Calcul à l'ordre élevé de la longueur des arêtes du maillage courbe par extrapolation des métriques de contrôle sur les polynômes de Bernstein rationnels. On obtient une carte des arêtes trop longues.
- Tant qu'il reste des éléments dont seules deux des arêtes doivent être subdivisées :
  - parcours de ces éléments :
    - marquage de la dernière arête comme devant être également subdivisée.
    - mise à jour de la carte des arêtes.
- Parcours de tous les éléments du maillage :
  - Si une seule des arêtes de l'élément doit être subdivisée, on applique l'algorithme de *découpe* 1-2.
  - Si toutes les arêtes de l'élément doivent être subdivisées, on applique l'algorithme de *découpe* 1-4.

À la sortie, on obtient notre maillage raffiné en accord avec la taille de maille imposée.

Remarque IV.3.8 (*Éléments issus du raffinement d'un élément courbe*)

Les enfants d'un élément courbe subdivisé sont très probablement également courbes. Ainsi, après raffinage, certaines arêtes internes du maillage peuvent être courbes, en particulier les arêtes proches des bords courbes de la géométrie.

3.4 Schéma d'adaptation stationnaire

En conclusion de la partie théorique de cette section, nous présenterons la boucle d'adaptation stationnaire.

On entre dans l'algorithme au temps $i = 0$ avec un maillage initial $\mathcal{T}_0$.

Algorithme IV.3.7: Boucle d'adaptation stationnaire

1. Calcul de la solution $\mathcal{S}_i$ sur le maillage $\mathcal{T}_i$.
2. Test de convergence : si l'erreur d'interpolation de la solution $\mathcal{S}_i$ est majorée par ε sur $\mathcal{T}_i$ on sort de la boucle, sinon on passe à l'étape suivante.
3. Calcul de la métrique $\mathcal{M}_i$:
   - calcul, par un estimateur d'erreur d'interpolation classique, de la métrique aux nœuds du sous-maillage droit associé au maillage $\mathcal{T}_i$.
   - couplage de cette métrique au maillage d'ordre élevé $\mathcal{T}_i$.
4. Génération du maillage $\mathcal{T}_{i+1}$ par l'algorithme de raffinement IV.3.6.
5. Interpolation de la solution $\mathcal{S}_i$ sur le maillage $\mathcal{T}_{i+1}$: le maillage $\mathcal{T}_{i+1}$ est un maillage $\mathcal{NURBS}$ d'ordre élevé donc on utilise une interpolation d'ordre élevé sur les fonctions de base $\mathcal{NURBS}$ pour obtenir les valeurs de $\mathcal{S}_i$ aux nœuds de $\mathcal{T}_{i+1}$.
6. $i = i + 1$ et retour à l'étape 1.

Décrivons cette boucle.

Pour équi-répartir notre erreur, on se donne un seuil d'erreur maximal ε, et un maillage initial $\mathcal{T}_0$. On lance un premier calcul sur ce maillage. Après convergence, on obtient une solution $\mathcal{S}_0$. L'erreur commise est évaluée en chaque nœud. Si notre erreur est supérieure à ε en certains nœuds, le couple $(\mathcal{T}_0, \mathcal{S}_0)$ n'a pas convergé. On évalue donc pour chaque arête la métrique voulue en fonction de l'erreur commise, de ε et de la taille de l'arête actuelle. Cette métrique permet d'adapter notre maillage et de générer le maillage $\mathcal{T}_1$. On calcule la nouvelle solution $\mathcal{S}_1$ sur $\mathcal{T}_1$. Si l'erreur de cette solution est inférieure à ε en tout nœud, le couple $(\mathcal{T}_1, \mathcal{S}_1)$ a atteint son point fixe donc on stoppe le calcul. Sinon, on

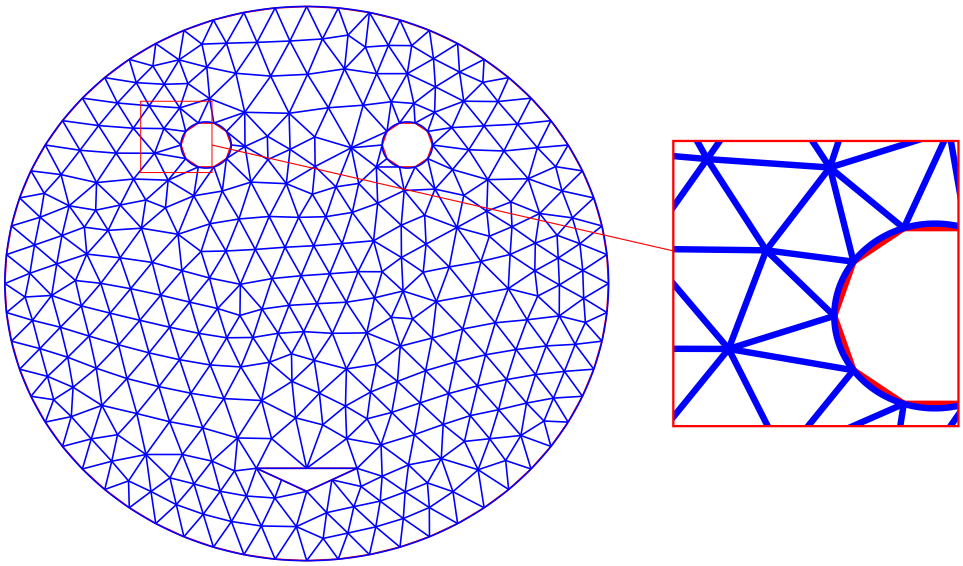


Figure IV.3.13 – Maillage initial : pour l'ordre 2, les contours du domaines sont linéaires par morceaux (segments rouges), pour des ordres supérieurs les contours du domaine sont courbes (représentés en bleu).

continue la boucle car notre couple maillage-solution n'a toujours pas convergé.

Afin de garantir la validité de notre méthode de h-adaptation, il convient d'étudier le comportement de notre remailleur vis à vis d'une fonction analytique.

3.5 Validation

Nous avons testé notre procédure de raffinement sur deux fonctions définies sur $[0,1]^2$, la première, continue, et la seconde, discontinue. Le domaine de calcul est un disque contenant trois « trous ». Deux des trous sont circulaires, la frontière du dernier est linéique par morceaux. Le maillage initial est représenté figure IV.3.13. Il est composé de :

- 648 triangles,
- 365 sommets,
- 365 DoFs à l'ordre 2, 1380 DoFs à l'ordre 3, 3043 DoFs à l'ordre 4 et 5354 DoFs à l'ordre 5.

On note n_T le nombre de triangles du maillage d'ordre élevé.

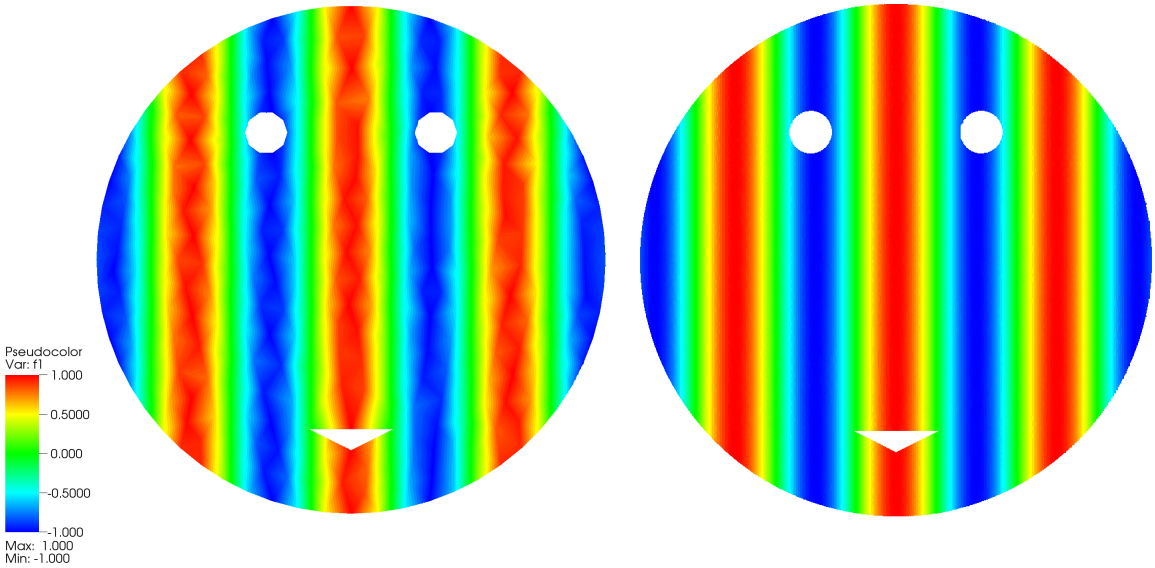


Figure IV.3.14 – Valeurs de la fonction f_1 aux ordres 2 (à gauche) et 5 (à droite) sur la discrétisation initiale.

3.5.1 Raffinement sur une fonction continue

3.5.1.1 Description du cas test

Les paramètres de notre cas tests sont les suivants :

- On étudie la fonction :

$$f_1(x) = \cos(10\,x). \tag{IV.20}$$

- Notre critère d'arrêt portera sur la norme $\mathcal{L}_\infty$ de l'erreur d'interpolation P_1 commise sur le maillage subdivisé.
- On fixe notre seuil d'erreur maximale ε à 0.001.

Les approximations initiales de f_1 aux ordres 2 et 5 sont fournies figure IV.3.14.

3.5.1.2 Résultats

On étudie les maillages obtenus à convergence de l'algorithme. La figure IV.3.15 représente, pour les ordres 2 à 5, le nombre de degrés de liberté et de triangles des maillages (graphique du haut), la norme $\mathcal{L}_\infty$ de l'erreur d'interpolation sur ces maillages et la norme $\mathcal{L}_\infty$ de l'erreur d'interpolation obtenue sur les maillages subdivisés associés (graphique du bas).

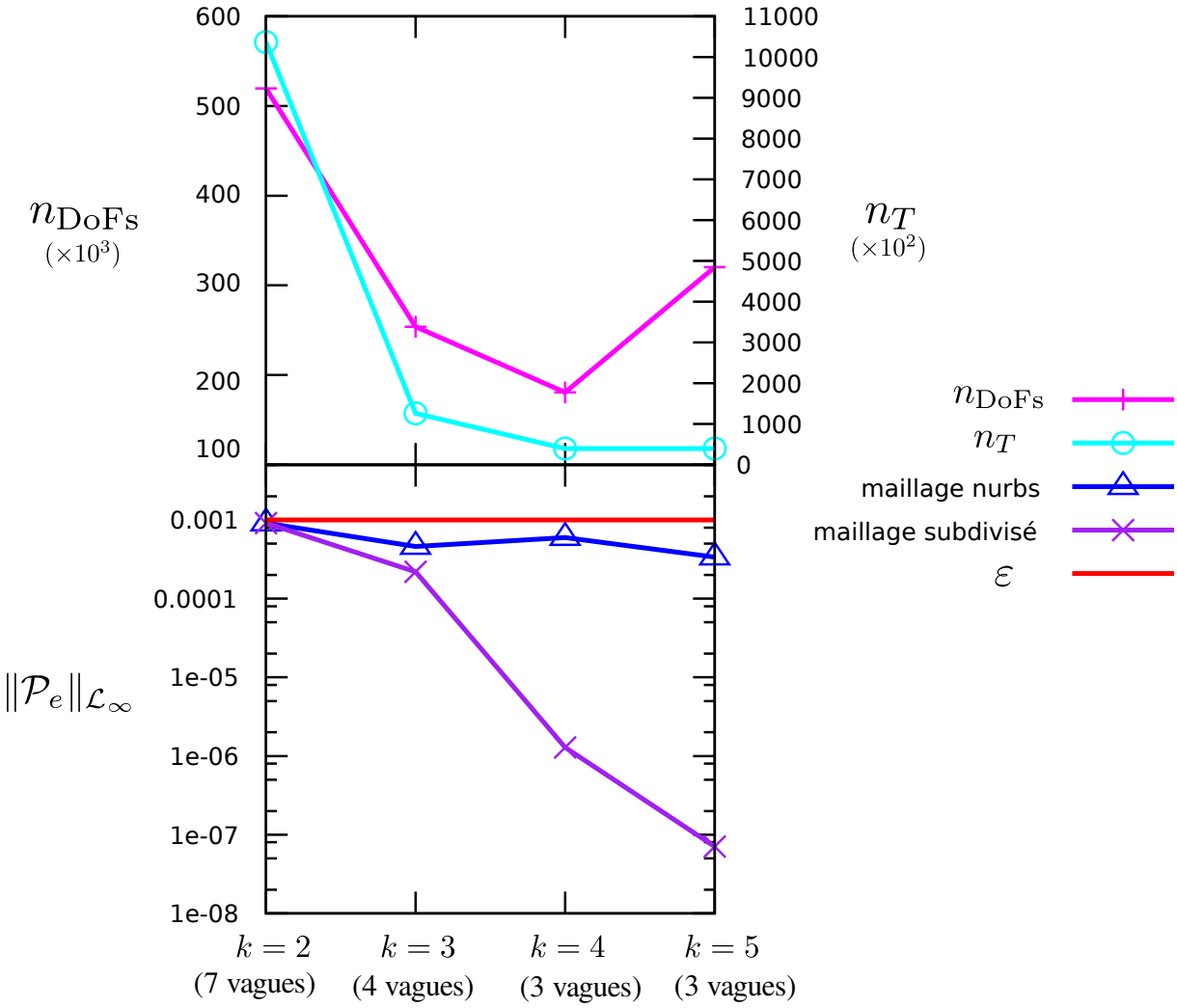


Figure IV.3.15 – Comparaison des raffinements d'ordre 2, 3, 4 et 5 à convergence de l'algorithme d'adaptation : le graphique du haut représente l'évolution du nombre de DoFs et du nombre de triangles en fonction de l'ordre de travail. Celui du bas représente la norme $\mathcal{L}_\infty$ de l'erreur d'interpolation commise sur les maillages obtenus et sur leur version subdivisée.

On constate que jusqu'à l'ordre 4, notre modèle semble se comporter comme nous nous l'attendions. Plus le maillage est d'ordre élevé :

- plus le nombre final de DoFs est faible,
- plus le nombre final de triangles est faible,
- plus le nombre de vagues de subdivisions nécessaires pour atteindre la convergence est faible.

On remarque cependant qu'à l'ordre 5, si le maillage final contient bien moins de DoFs que celui obtenu à l'ordre 2, il en contient, en revanche, plus que ceux d'ordre 3 et 4.

Ce dernier phénomène s'explique simplement par le fait que plus l'ordre est élevé, plus la subdivision d'un élément créera de DoFs : à l'ordre k, la subdivision d'un triangle crée $\left(\left(\frac{k+1}{2}-1\right)3k\right)$ nouveaux DoFs. Donc, la croissance du nombre de DoFs à chaque vague de subdivision est d'autant plus importante que l'ordre de travail est élevé et n'est pas nécessairement contrebalancée par le nombre plus faible d'éléments à raffiner. L'exemple qui suit illustre ce comportement.

> **Exemple IV.3.1** ***(DoFs créés aux ordres*** 2 ***et*** 5***)*** **:** l'exemple décrit est illustré figure IV.3.16.
>
> On compare le raffinement d'un élément à l'ordre 2 et à l'ordre 5. Supposons qu'à l'ordre 2 comme à l'ordre 5 l'erreur d'interpolation commise sur notre élément est supérieure à notre seuil. L'élément, quelque soit l'ordre, doit être subdivisé en 4 sous-éléments. On représente en rose les éléments devant être subdivisés sur notre figure.
>
> On obtient 4 éléments pour chacun des ordres à la vague de raffinement numéro 1. On atteint la convergence pour l'ordre 5. À l'ordre 2, un élément doit encore être raffiné. À cet ordre, la convergence est atteinte à la seconde vague de raffinement. On constate qu'on obtient alors plus d'éléments à l'ordre 2 mais moins de DoFs. On constate également que plus on raffine à un ordre élevé, moins le raffinage est local.

On peut noter que l'erreur d'interpolation sur le maillage subdivisé associé au maillage d'ordre 5 est de l'ordre de 3.10^{-4}, donc très inférieure à notre ε. On en conclut que le raffinement du maillage d'ordre 5 fournit une solution trop précise.

On représente figures IV.3.17, IV.3.18, IV.3.19 et IV.3.20 les maillages obtenus en sortie de la procédure d'adaptation.

Leur aspect confirme nos remarques préalables : les maillages obtenus aux ordres 4 et 5 sont raffinés quasiment uniformément et sont presque identiques.

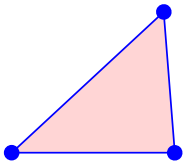
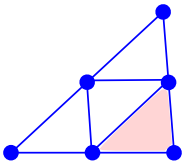
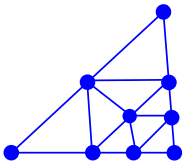
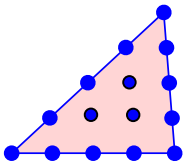
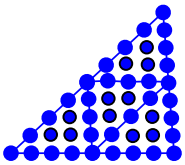
	Vague 0	Vague 1	Vague 2	n_{DoFs}	n_T
$k=2$				9	8
$k=5$				30	4

Figure IV.3.16 – Raffinement d'ordre 5 créant plus de DoF*s* qu'à l'ordre 2.

3.5.2 Raffinement sur une fonction discontinue

3.5.2.1 Description du cas test

Les paramètres du cas test sont les suivants :

- On étudie la fonction :

$$f_2(x) = \begin{cases} u_{\sqrt{\Pi/2}}(x,y), & \text{si } x \leq \frac{1}{2}\cos(\Pi y) \\ u_{-\sqrt{\Pi/2}}(x,y) + \cos(2\Pi y), & \text{si } x > \frac{1}{2}\cos(\Pi y) \end{cases}, \tag{IV.21}$$

  où, pour $r = y + \tan(-\Phi)\,x$:

$$u_\Phi(x,y) = \begin{cases} -r\sin\left(\frac{\Pi}{2}r^2\right), & \text{si } r \leq -\frac{1}{3} \\ 2r - 1 + \frac{1}{6}\sin(3\Pi r), & \text{si } r \geq \frac{1}{3} \\ |\sin(2\Pi r)|, & \text{si } |r| < \frac{1}{3} \end{cases}$$

- Du fait de la présence d'un choc dans la fonction étudiée, l'algorithme de raffinement ne peut pas converger en norme $\mathcal{L}_\infty$, notre critère d'arrêt porte donc sur la norme $\mathcal{L}_1$ de l'erreur d'interpolation commise sur le maillage subdivisé.
- On fixe le seuil d'erreur maximale ε à $\frac{0.2}{36\Pi}$.

Les interpolées de f_2 sur la discrétisation initiale aux ordres 2 et 5 sont fournies figure IV.3.21.

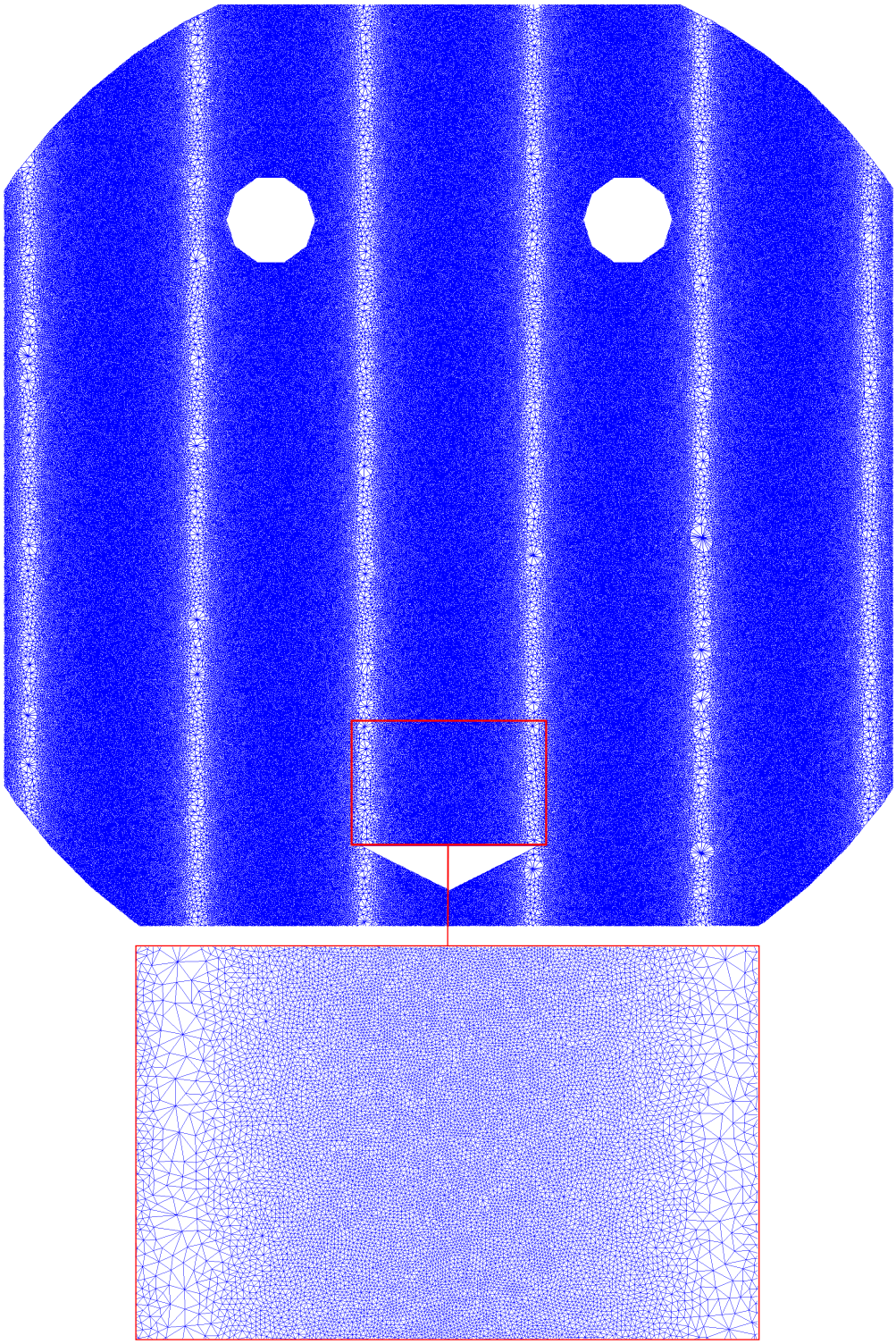


Figure IV.3.17 – Maillage obtenu en sortie de la procédure d'adaptation à l'ordre 2 : le maillage compte 1036618 triangles.

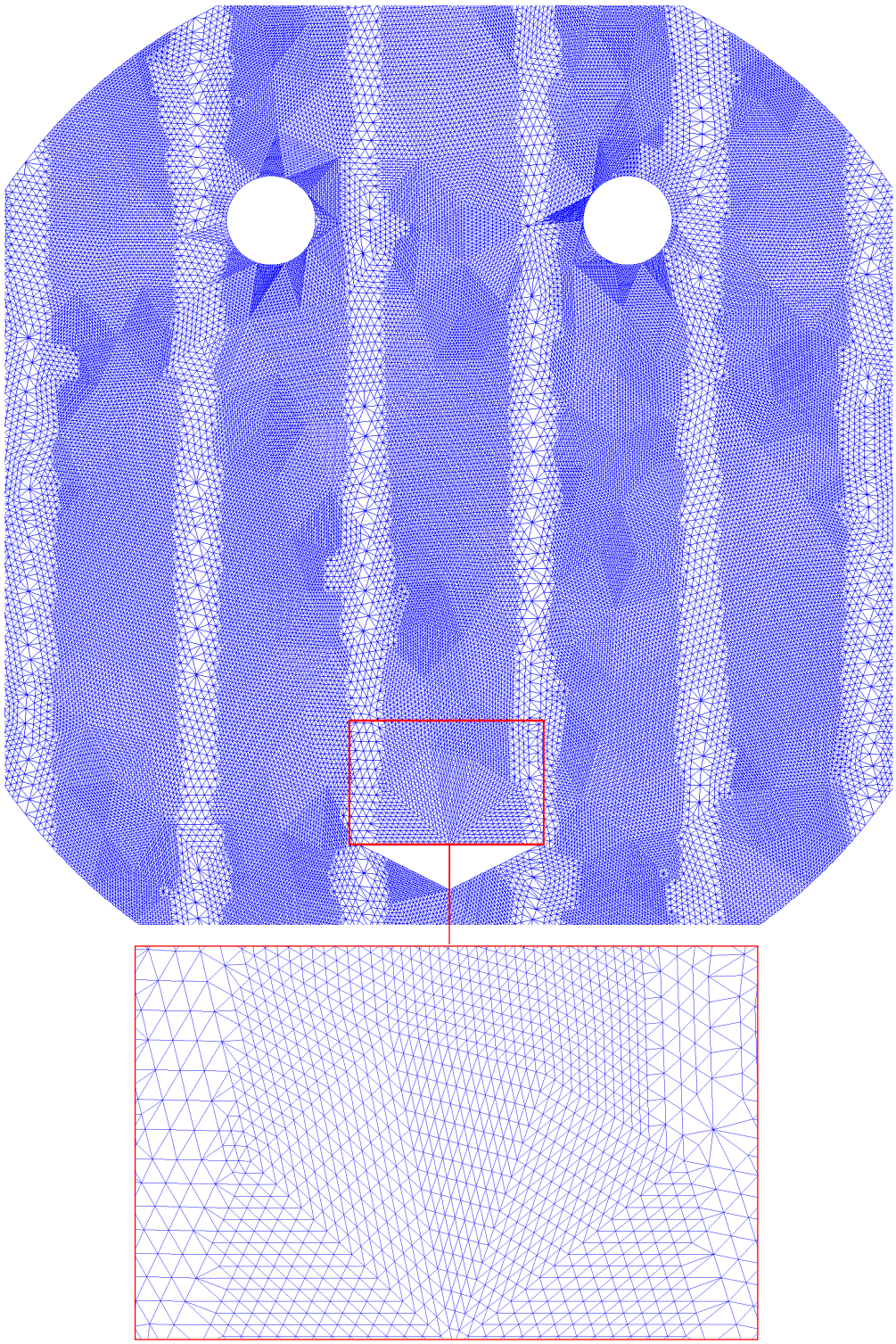


Figure IV.3.18 – Maillage obtenu en sortie de la procédure d'adaptation à l'ordre 3 : le maillage compte 126290 triangles.

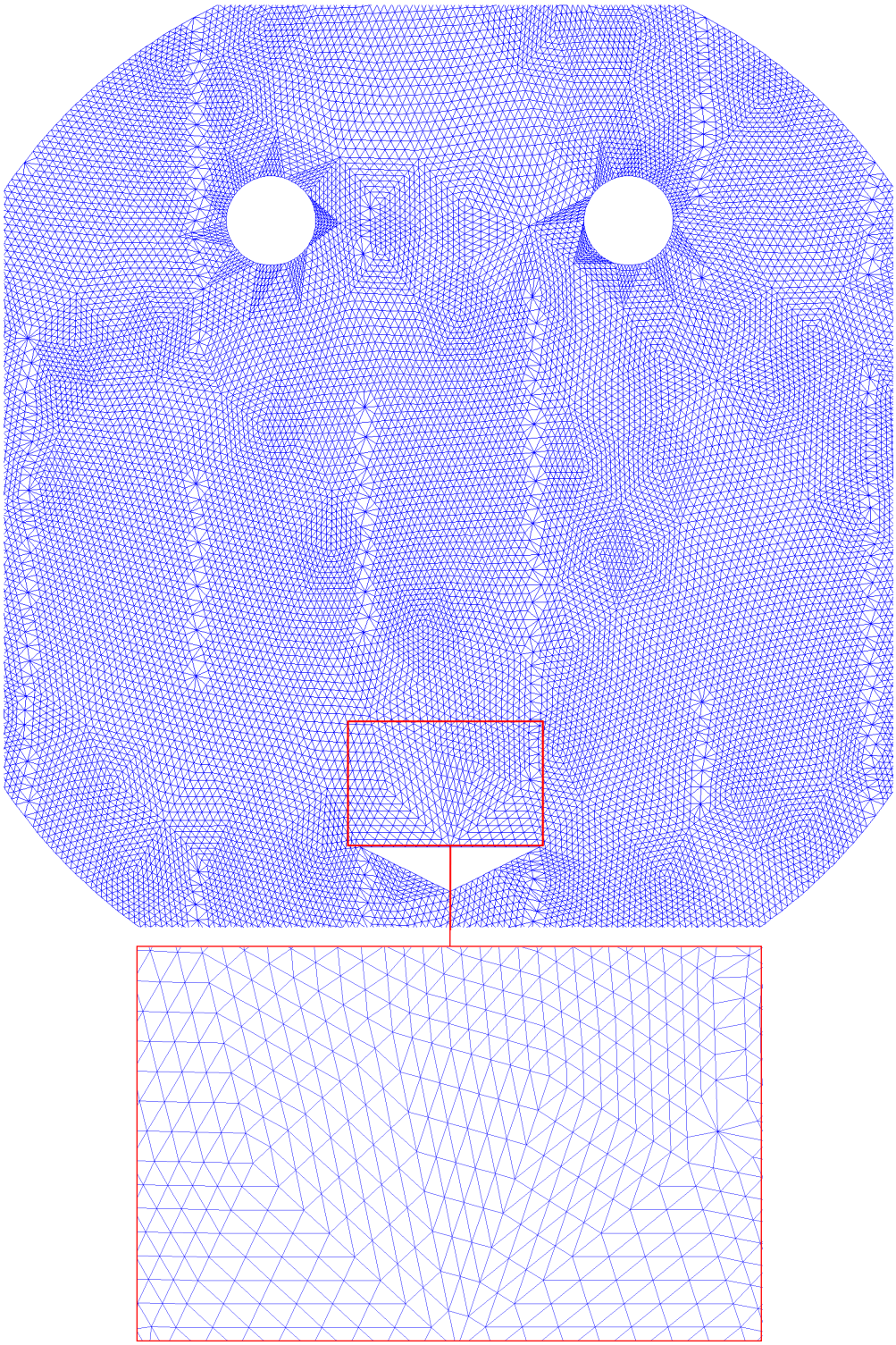


Figure IV.3.19 – Maillage obtenu en sortie de la procédure d'adaptation à l'ordre 4 : le maillage compte 39934 triangles.

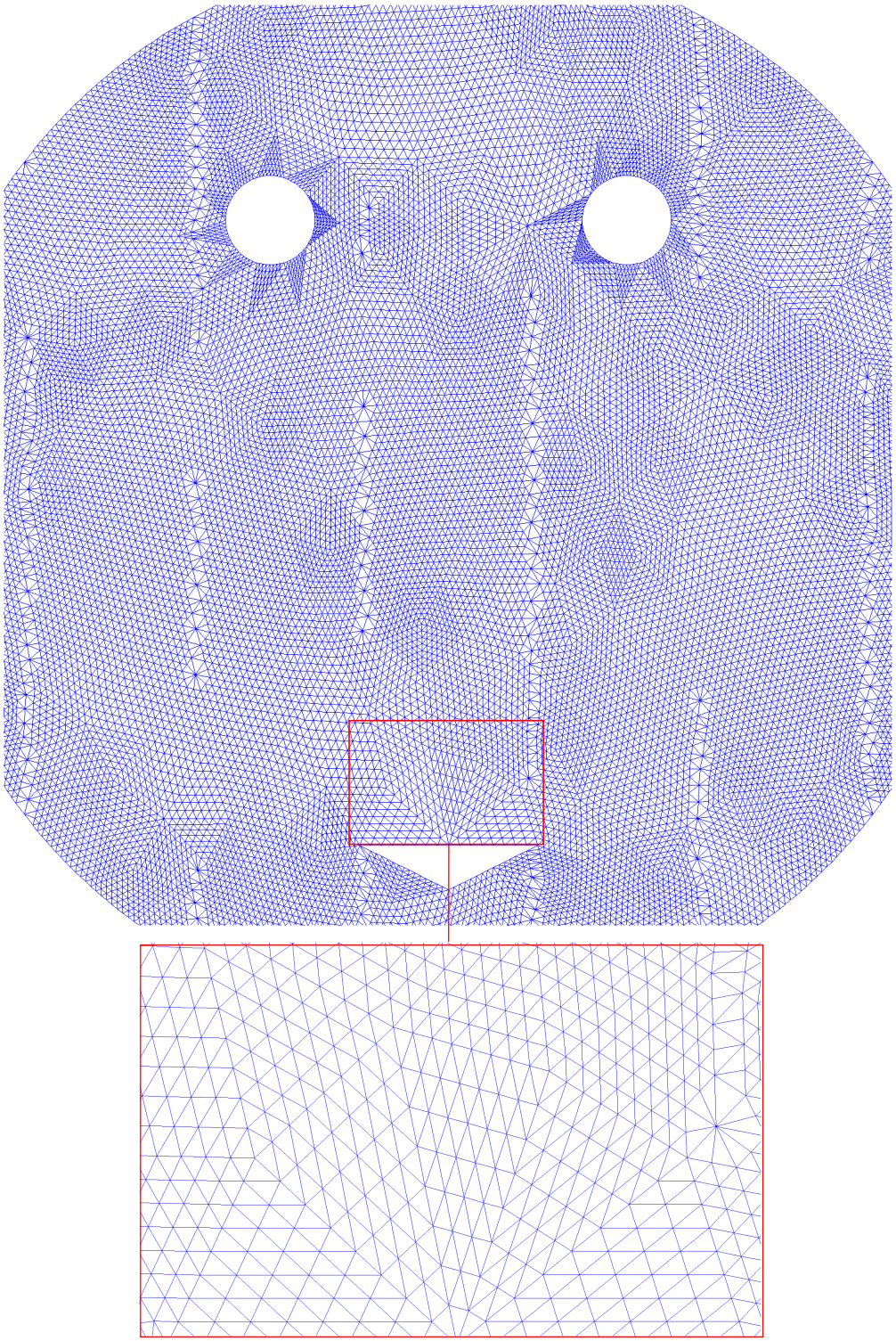


Figure IV.3.20 – Maillage obtenu en sortie de la procédure d'adaptation à l'ordre 5 : le maillage compte 39876 triangles.

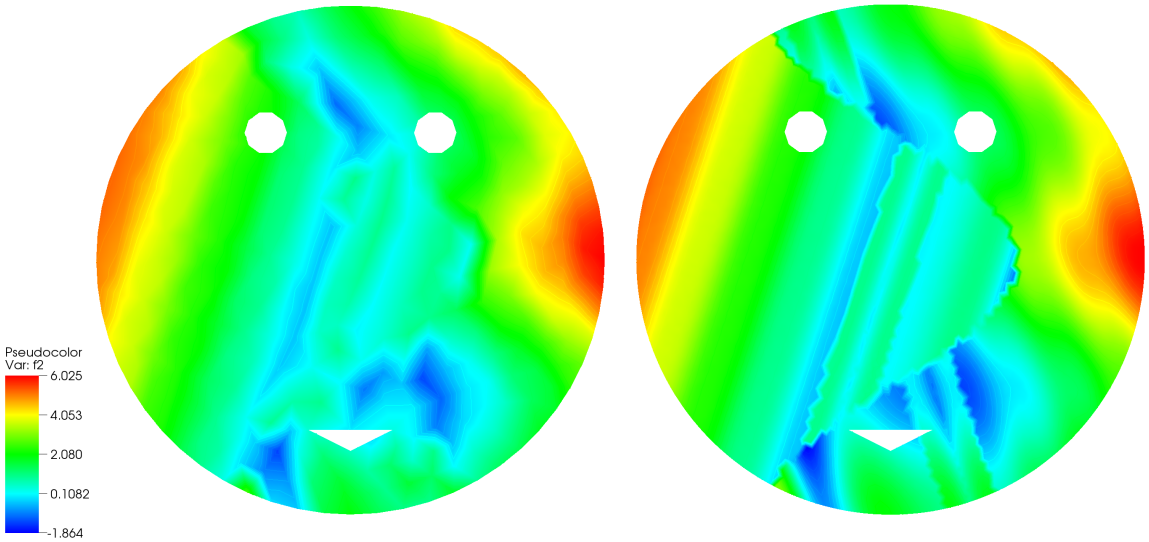


Figure IV.3.21 – Valeurs de f_2 aux ordres 2 (à gauche) et 5 (à droite) sur la discrétisation initiale.

3.5.2.2 Résultats

On étudie les maillages obtenus à convergence de l'algorithme. La figure IV.3.22 représente, pour les ordres 3 à 5, le nombre de degrés de liberté et de triangles des maillages, la norme $\mathcal{L}_1$ de l'erreur d'interpolation sur les maillages d'ordre élevé et la norme $\mathcal{L}_1$ de l'erreur d'interpolation obtenue sur les maillages subdivisés associés. Nous remarquons que :

- comme précédemment, le nombre de triangles que comptent les maillages diminue quand on élève l'ordre de ceux-ci.
- le nombre de vagues de subdivisions diminue également.
- par contre le nombre de DoF*s* augmente avec l'ordre du maillage.

L'élévation du nombre de degrés de liberté s'explique par le même argument que dans le paragraphe précédent, le phénomène est juste beaucoup plus important du fait de la présence d'un choc dans la solution : la métrique calculée sur les sous-maillages majore la norme $\mathcal{L}_\infty$ de notre erreur d'interpolation, or, dans cette norme, l'erreur commise sur les éléments situés dans la zone du choc ne diminue pas. Chacune des vagues de raffinement subdivisera donc les éléments situés sur le choc. Comme, à chacune de ces vagues, on crée $\left(\left(\frac{k+1}{2}-1\right)3k\right)$ nouveaux DoF*s* par élément subdivisé, plus l'ordre de travail est élevé,

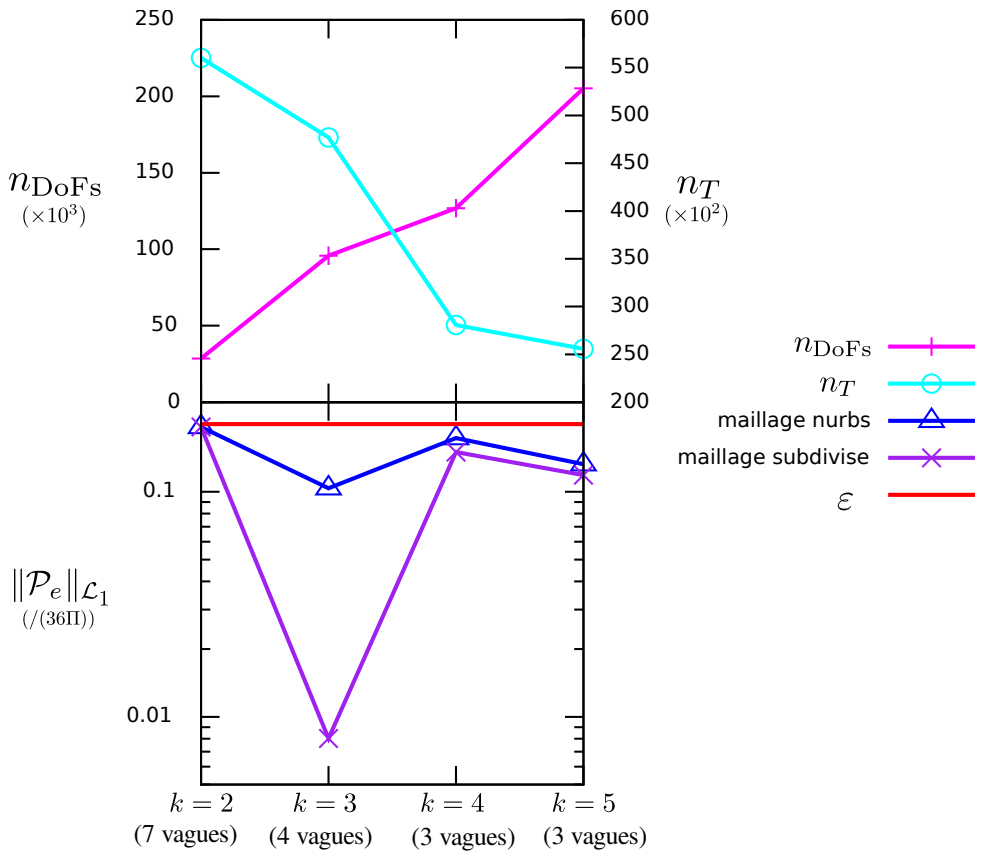


Figure IV.3.22 – Comparaison des raffinements d'ordre 2, 3, 4 et 5 à convergence de l'algorithme d'adaptation.

plus on incrémente de façon drastique notre nombre de degrés de liberté.

On représente figures IV.3.23, IV.3.24, IV.3.25 et IV.3.26 les maillages obtenus en sortie de la procédure d'adaptation. Ces maillages confirment que notre procédure de raffinement respecte la forme de la solution.

La figure IV.3.27 représente l'interpolée de f_2 sur les maillages d'ordre 2 et 5. La solution est maintenant très bien représentée. La représentation des frontières est cependant dégradée à l'ordre 2 puisque les raffinements successifs n'ont pas modifié la discrétisation initiale des frontières.

Nous disposons maintenant de tous les outils nécessaires au lancement de quelques cas tests.

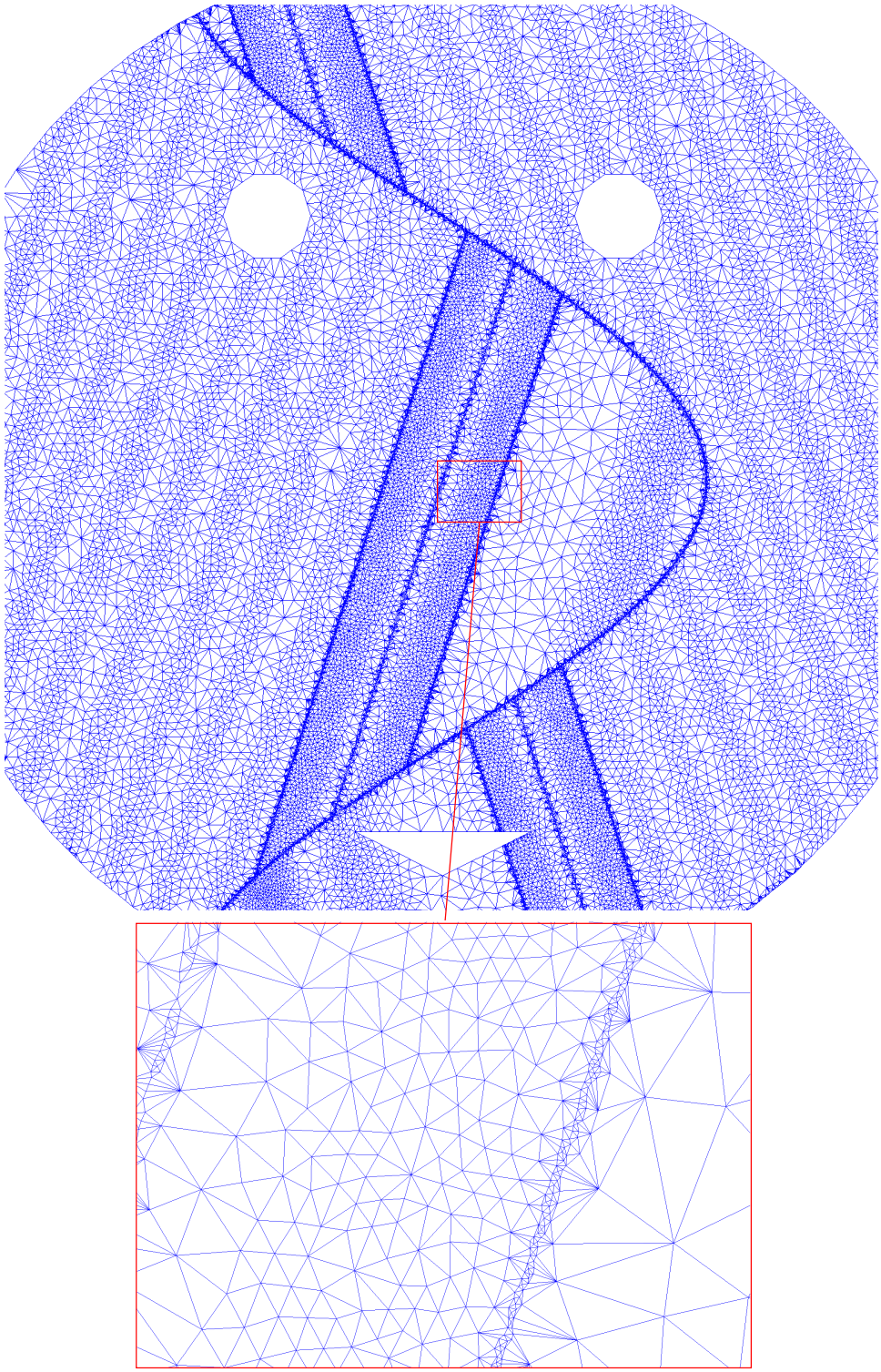


Figure IV.3.23 – Maillage obtenu en sortie de la procédure d'adaptation à l'ordre 2 : le maillage compte 56019 triangles.

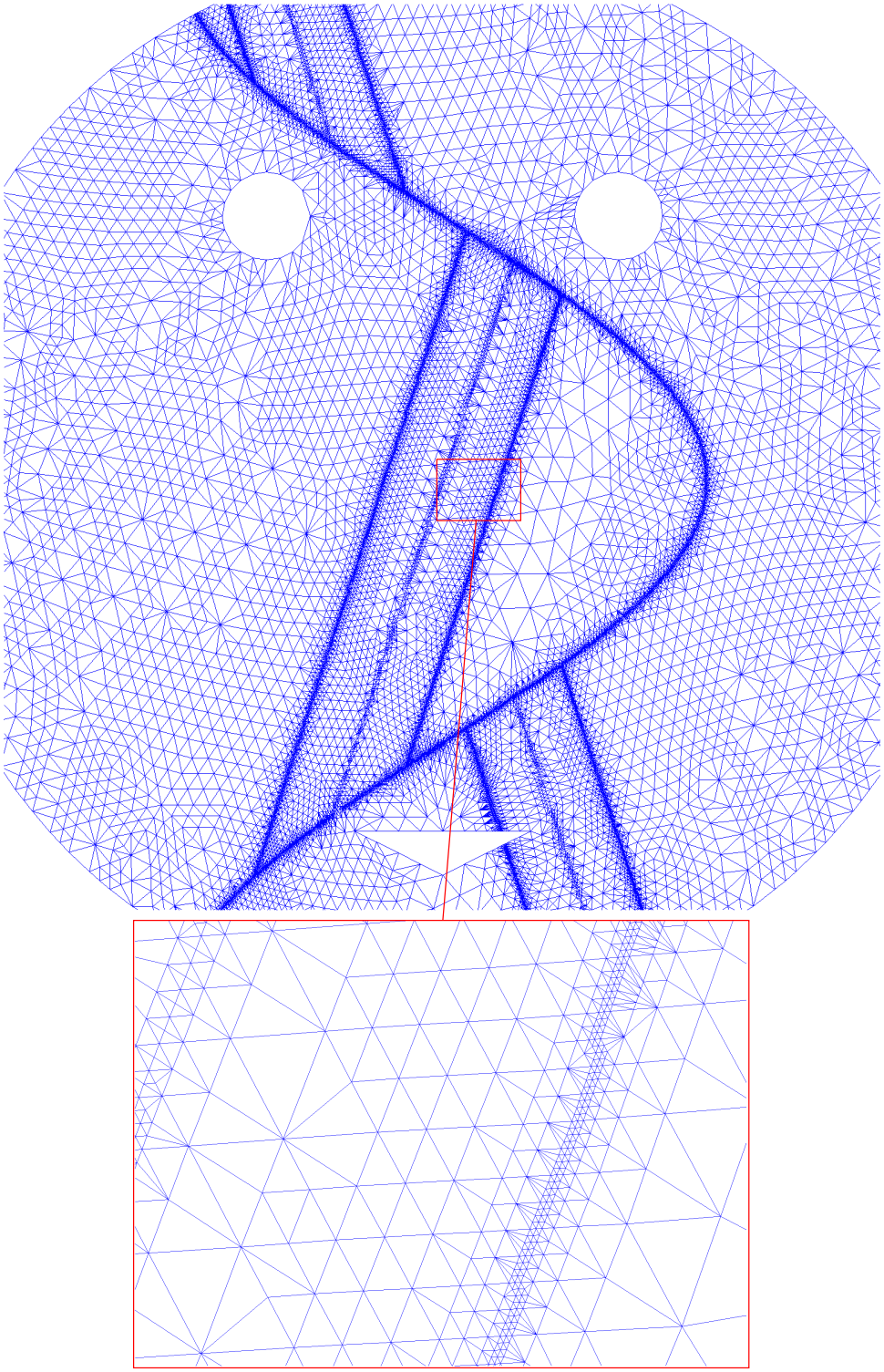


Figure IV.3.24 – Maillage obtenu en sortie de la procédure d'adaptation à l'ordre 3 : le maillage compte 47691 triangles.

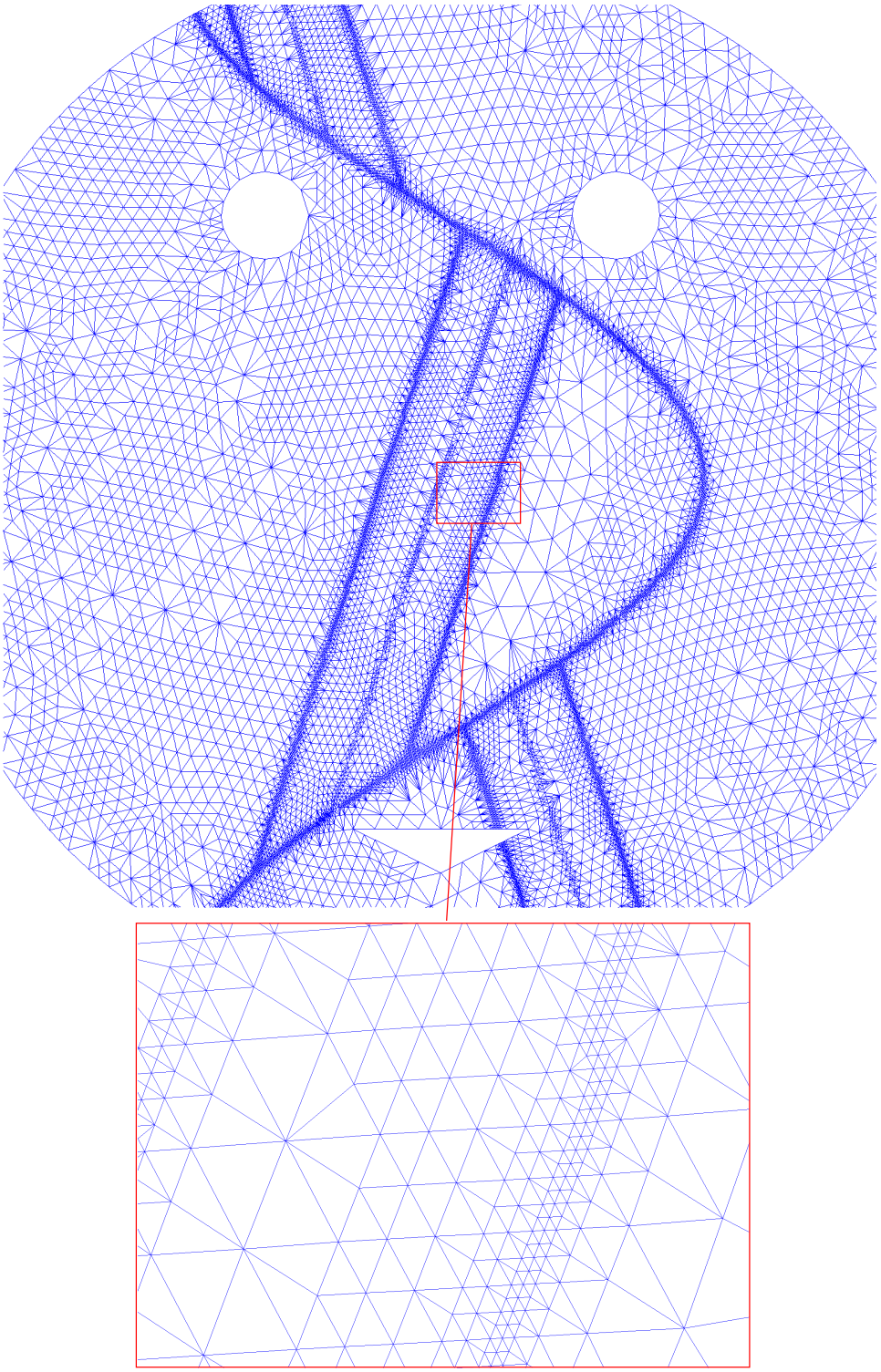


Figure IV.3.25 – Maillage obtenu en sortie de la procédure d'adaptation à l'ordre 4 : le maillage compte 28077 triangles.

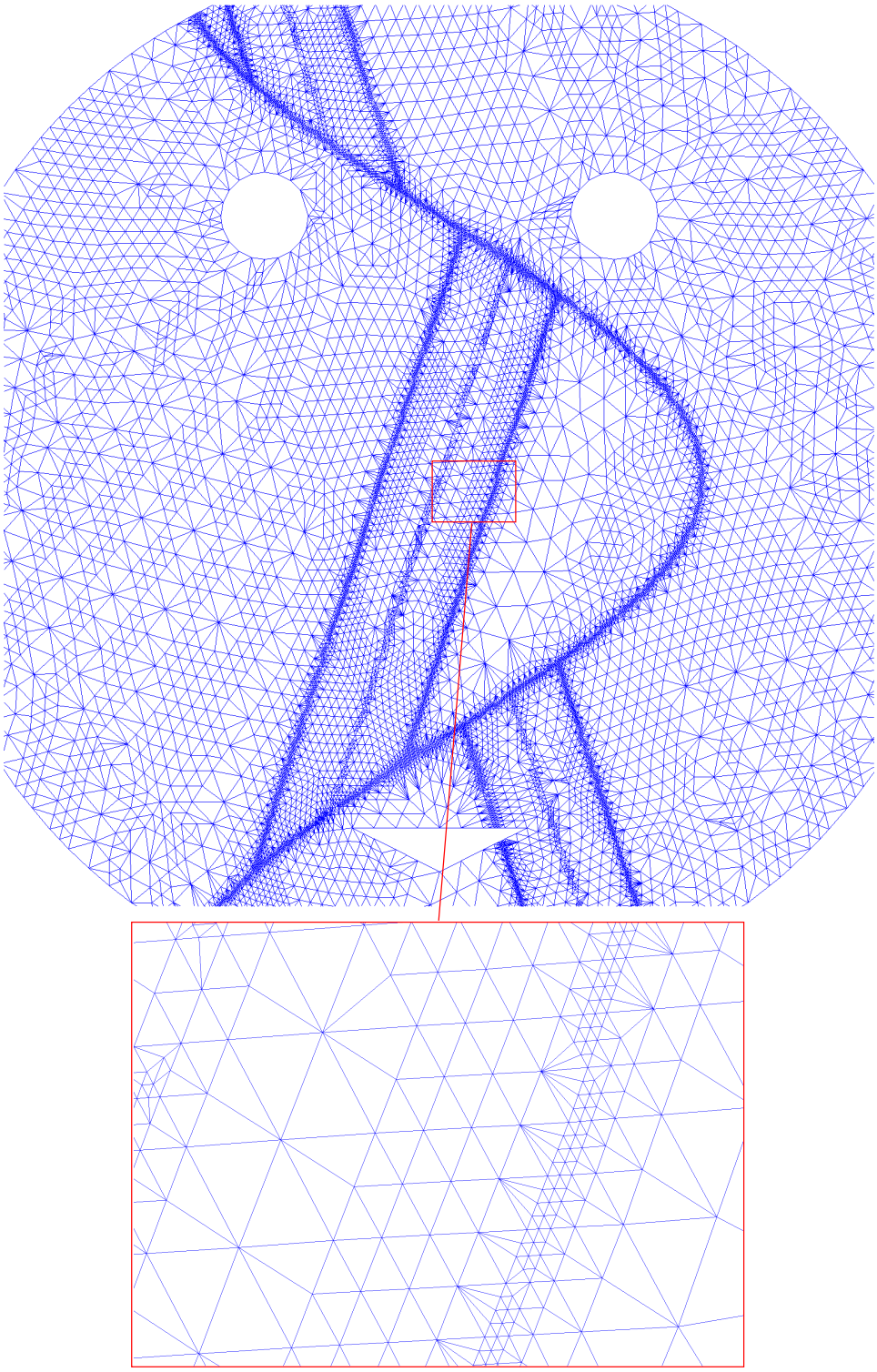


Figure IV.3.26 – Maillage obtenu en sortie de la procédure d'adaptation à l'ordre 5 : le maillage compte 25577 triangles.

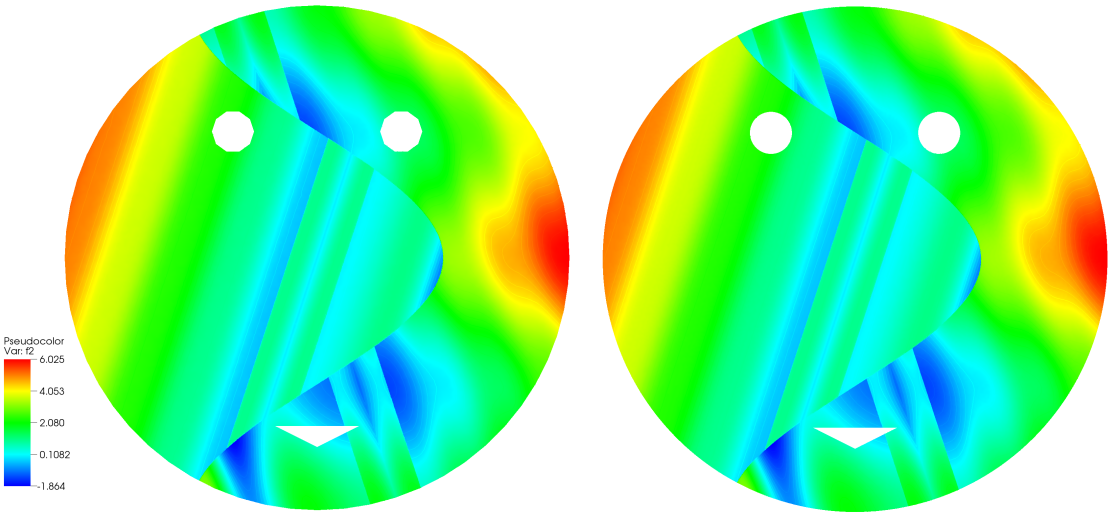


Figure IV.3.27 – Discrétisations finales de f_2 aux ordres 2 (à gauche) et 5 (à droite). On représente les valeurs de la fonction ainsi que 40 isolignes allant de -1.864 à 6.025.

Chapitre V

Résultats numériques

Nous avons construit, dans les chapitres précédents, tous les outils nécessaires à la résolution des équations d'Euler (et de Navier-Stokes) par une méthode isogéométrique. Nous souhaitons maintenant tester et valider nos développements. Nous présenterons donc ici quelques cas tests classiques. Comme nous travaillons à l'ordre élevé voir très élevé et que la nature courbe de nos éléments entraine des calculs plus coûteux que ceux nécessaires pour un schéma $\mathcal{RD}$ classique, la parallélisation de notre code est cruciale. Nous débuterons donc ce chapitre par la description de notre méthode de parallélisation.

1 Mise en œuvre

1.1 Parallélisation

Le principe de la parallélisation est simple, il consiste à partager un problème en plusieurs sous-problèmes de plus petites tailles. Les sous-problèmes sont alors résolus simultanément, à raison d'un problème par processeur. On reconstitue la solution de notre problème initial à partir des solutions de chaque sous-problème.

Notre méthode de parallélisation se fonde sur le travail précédemment réalisé par A. Larat et détaillé dans sa thèse [36]. Dans son travail, il développe une méthode parallèle adaptée aux maillages d'ordre 3 linéaires par morceaux. Nous adapterons donc cette méthode à notre schéma $\mathcal{RD}$ isogéométrique ainsi qu'aux ordres très élevés.

L'essentiel de la parallélisation d'un code réside dans le calcul des solutions aux nœuds proches de l'interface de deux sous-problèmes. En effet, la mise à jour au temps t^{n+1} de la solution en un DoF demande de connaître la solution au temps t^n d'autres degrés de liberté qui devront donc appartenir au sous-problème. La décomposition du domaine de calcul en sous-domaines devra donc tenir compte des dépendances entre les nœuds.

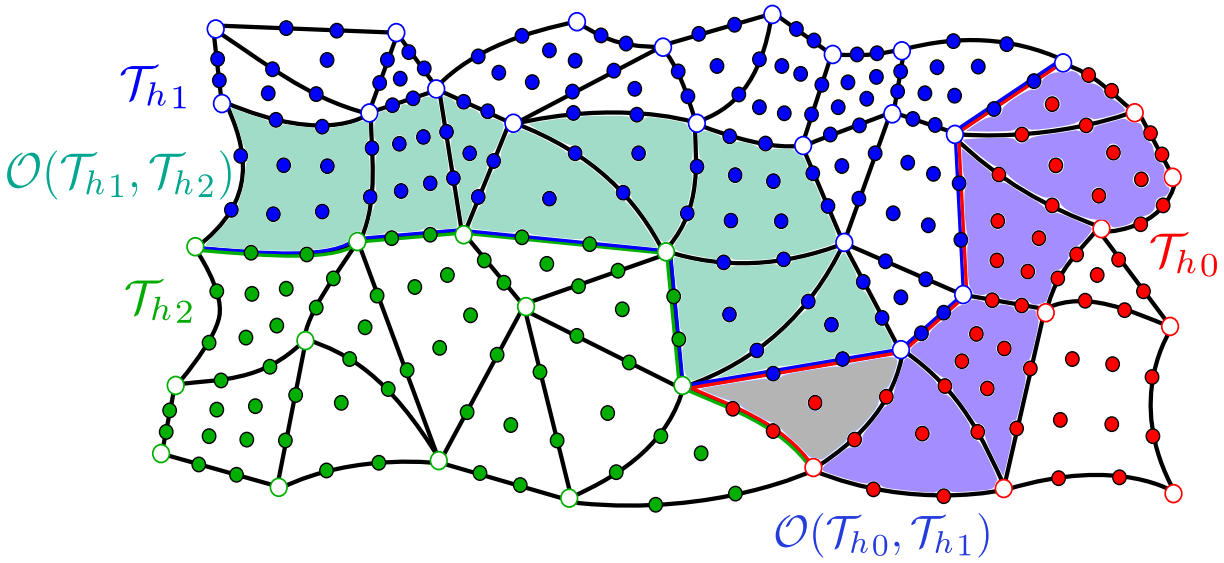


Figure V.1.2 – Décomposition d'un domaine d'ordre 4 en trois sous-domaines. L'overlap entre les sous-domaines ne contient qu'une couche d'éléments. L'élément en gris appartient à la fois à $\mathcal{O}(\mathcal{T}_{h0}, \mathcal{T}_{h1})$, $\mathcal{O}(\mathcal{T}_{h1}, \mathcal{T}_{h2})$ et $\mathcal{O}(\mathcal{T}_{h0}, \mathcal{T}_{h2})$.

Lors du développement de notre schéma isogéométrique, nous avons tenu à conserver la compacité du schéma $\mathcal{RD}$ initial. Ainsi, la mise à jour de la solution en un DoF M_i de notre maillage nécessite seulement la connaissance des DoF*s* partageant un élément avec M_i. Tous les nœuds internes aux sous-domaines ou appartenant à l'une des frontières physique du problème peuvent donc être calculés et, finalement, pour calculer tous les DoF*s* de notre discrétisation initiale, il nous suffit de décomposer notre domaine en sous-domaines dont les cellules à l'interface se recouvrent, *c.-à-d.* en sous-domaines ayant un overlap d'épaisseur 1 (voir définition V.1.1 et figure V.1.2).

Définition V.1.1 (*Overlap*)

Soient $\mathcal{T}_{hi}$ et $\mathcal{T}_{hj}$ deux sous-domaines d'un domaine initial $\mathcal{T}_h$. On nomme *overlap de* $\mathcal{T}_{hi}$ *et* $\mathcal{T}_{hj}$ l'intersection de $\mathcal{T}_{hi}$ et $\mathcal{T}_{hj}$. On note cet overlap $\mathcal{O}(\mathcal{T}_{hi}, \mathcal{T}_{hj})$.

Notre discrétisation initiale est distribuée en sous-domaines par le partitionneur SCOTCH ([59]) qui nous fournit également une renumérotation locale à chaque sous-domaine de nos éléments et des nœuds principaux. Lorsque les extrémités d'une arête se situent sur deux processeurs différents, on choisit de faire appartenir les nœuds d'ordre élevé de l'arête au processeur de plus faible numéro. On applique le même principe pour

les nœuds internes aux éléments : si les sommets d'un élément appartiennent à des processeur différents, alors les points internes à l'élément seront calculés par le processeur de plus faible numéro. Le calcul des DoFs des éléments de l'overlap nécessite des échanges de données entre les processeurs.

> **Exemple V.1.1** ***(Décomposition de domaine)*** **:** Les nœuds représentés en rouge figure V.1.2 sont calculé par le premier processeur, ceux en bleu par le second et ceux en vert par le troisième. Les DoFs d'ordre élevé d'une arête dont l'une des extrémité appartient à $\mathcal{T}_{h1}$ et l'autre à $\mathcal{T}_{h2}$ appartiendront à $\mathcal{T}_{h1}$. Les éléments colorés sont les éléments des overlaps entre processeurs.
>
> Prenons l'exemple de l'élément représenté en gris et appartenant aux deux overlaps. On cherche à calculer la valeur des nœuds au temps t^{n+1}. Chacun des processeurs connait la valeur de tous les DoFs de l'élément gris au temps t^n et peut donc calculer la valeur au temps t^{n+1} des nœuds lui appartenant. Il les envoie alors aux autres processeurs de l'overlap et reçoit les valeurs des DoFs de l'élément ne lui appartenant pas.

On peut noter que, puisque l'on travaille avec des éléments courbes, il nous faudra sauvegarder le poids associé à un point de contrôle dans le sous-domaine auquel appartient ce $\mathcal{CP}$. Pour conclure, pour chaque sous-domaine, il nous faudra connaître :

- Le nombre de sous-domaines voisins de notre domaine.
- Pour chaque voisin, on liste les numéros des DoFs pour lesquels on devra communiquer une valeur à ce voisin ainsi que la liste des DoFs dont on recevra une valeur du voisin.
- La liste des DoFs du domaine, les coordonnées de leur point de contrôle, les valeurs des poids associés et, éventuellement, la valeur des solutions aux nœuds. On peut noter que la numérotation des nœuds, et plus particulièrement des DoFs d'ordre élevé doit permettre les communications entre processeurs tout en interdisant les permutations entre les points d'un processeur à l'autre.
- La liste des éléments du domaine n'appartenant pas aux overlaps et de leurs degrés de libertés ainsi que la liste des éléments des overlaps et de leur DoFs.
- La liste des segments n'appartenant pas aux overlaps et de leurs degrés de libertés ainsi que la liste des segments des overlaps et de leur DoFs.

Finalement, trois fichiers complémentaires permettent de définir chacun des sous-domaines. Le premier de ces fichiers (d'extension *pmsh*) décrit le sous-domaine et les

nombre de sous-domaines voisins au notre ; nombre total de sous-domaines
nombre maximum d'émetteurs pour un voisin ; nombre maximum de récepteurs

pour chaque sous-domaine voisin au notre
numéro du sous-domaine
nombre d'émetteurs ; nombre de récepteurs
liste des sommets que l'on envoie au voisin
liste des sommets que l'on reçoit du voisin
dimension de travail ; type de maillage (courbe ou non) ;
nombre de sommets du maillage ; nombre d'éléments ;
nombre de faces frontières ; nombre maximal de sommets d'un élément ;
nombre maximal de sommets d'une face frontière

pour chaque point de notre sous-domaine
coordonnées du point ; poids associé au point

pour chaque élément de notre sous-domaine
nombre de sommets de l'élément ; numéro logique de l'élément
liste des sommets de l'élément ;

pour chaque face frontière de notre sous-domaine
nombre de sommets de la face ; numéro logique
liste des sommets de la face

Tableau V.1 – format de fichier *pmsh*

communications à effectuer à l'ordre 2. On fournit le format de ce fichier tableau V.1. Les données ajoutées pour l'analyse isogéométrique sont écrites en bleu, les commentaires en italique.

Le second fichier, d'extension *pseg*, décrit les communications d'ordre élevé de l'overlap. On fournit le format de ce fichier tableau V.2.

Le dernier fichier (d'extension *pmshp2*) est utilisé uniquement en analyse isogéométrique. Il complète le fichier *pmsh* par les informations sur les DoF*s* d'ordre élevé du sous-domaine. Dans le cas d'un maillage droit, les DoF*s* peuvent être recalculés, ce fichier est donc inutile.
On fournit le format *pmshp2* tableau V.3.

La figure V.1.3 analyse la scalabilité du code parallèle. On représente l'évolution du

nombre de sous-domaines voisins au notre ;
nombre maximum d'émetteurs pour un voisin ; nombre maximum de récepteurs

pour chaque sous-domaine voisin au notre
numéro du sous-domaine
nombre de récepteurs ; nombre d'émetteurs

pour chaque élément de l'overlap
pour chaque face de l'élément frontière de l'overlap
numéro de l'élément ;
numéros des sommets de la face
pour chaque DoF d'ordre élevé de la face
coordonnées et poids du DoF

pour chaque élément de l'overlap
pour chaque face de l'élément non-frontière de l'overlap
numéro de l'élément ;
numéros des sommets de la face

Tableau V.2 – format de fichier *pseg*

nombre de segments du domaine n'étant pas frontières de l'overlap ;

pour chaque élément du domaine
pour chaque face de l'élément n'étant pas frontière de l'overlap
numéros des sommets de la face
pour chaque DoF d'ordre élevé de la face
coordonnées et poids du DoF

nombre total d'éléments du domaine ; nombre de triangles ; nombre de quadrangles...

pour chaque élément
nombre de DoF*s* internes
numéros des sommets de l'élément
pour chaque DoF interne de l'élément
coordonnées et poids du DoF

Tableau V.3 – format de fichier *pseg*

Un nœud fourmi	
Processeur	2 Quad-core Nehalem Intel®Xeon®X5550
Fréquence	2,66 GHz
RAM	24Go (DDR3 1333MHz)
Infiniband QDR	40Gb/s

Tableau V.4 – Configuration d'un nœud fourmi

temps de calcul en fonction du nombre de processeurs sous-figure V.1.3(a) et l'accélération parallèle en fonction du nombre de processeurs sous-figure V.1.3(b). Rappelons que l'accélération $\mathcal{A}$ est donnée par la formule :

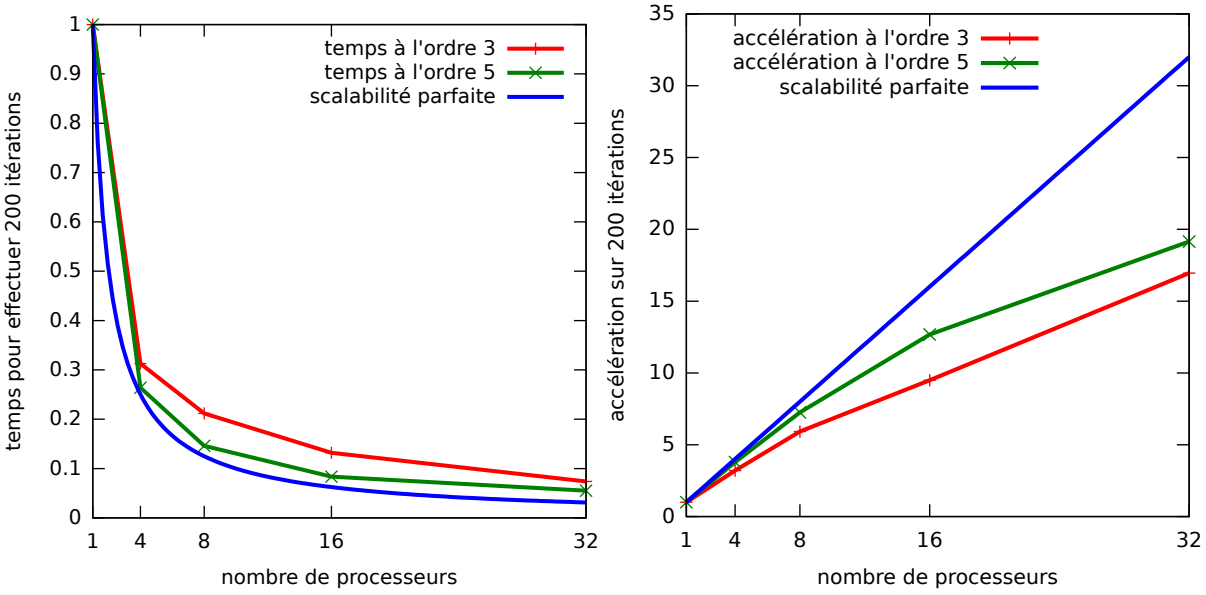
$$\mathcal{A}(n_p) = \frac{t(1)}{t(n_p)},$$

où n_p désigne le nombre de processeurs du calcul, $t(1)$ le temps de calcul sur 1 processeur et $t(n_p)$ le temps de calcul sur n_p processeurs.

À titre d'information, le cas test utilisé est celui de l'écoulement subsonique sur un bump (présenté sous-section 2.2 de la section 2). On le stoppe au bout de 200 itérations. Les calculs ont été effectués sur les nœuds fourmi de la machine plafrim de l'INRIA dont les caractéristiques sont fournies tableau V.4.

On constate que si notre code parallèle n'est pas tout à fait scalable, il permet toutefois un gain de temps non négligeable. Cette perte de scalabilité peut être en grande partie expliquée par le déséquilibre de charge entre processeurs : les processeurs de bas numéros (*par ex.* le processeur numéro 0) devront résoudre le problème sur un plus grand nombre de nœuds. Cet effet dépendra donc du rapport entre le nombre de DoF*s* intérieurs au sous-domaine et le nombre de DoF*s* appartenant à des segments frontières de ce sous-domaine. Comme en élevant l'ordre du calcul on ajoute des nœuds à l'intérieur des éléments, ceci aura tendance à rééquilibrer les charges entre domaines.

La figure V.1.4 représente la décomposition sur 8 processeurs du maillage d'ordre 5 du bump.



(a) Temps de calcul parallèle

(b) Accélération parallèle

Figure V.1.3 – Analyse de la scalabilité parallèle pour des calculs d'ordre 3 et 5. L'échelle des temps de la sous-figure (a) est normalisée par rapport au temps de calcul séquentiel.

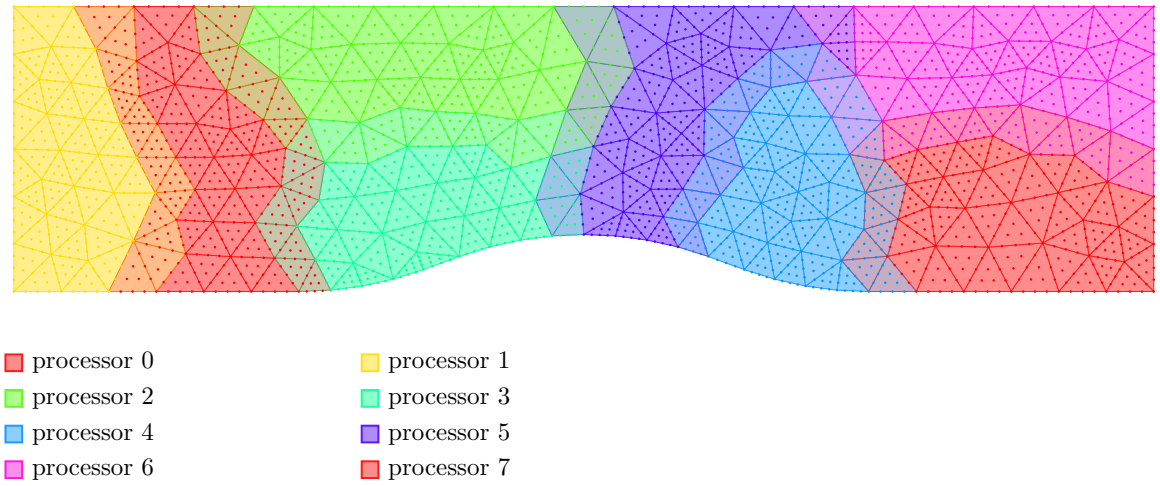


Figure V.1.4 – Équilibrage de charge entre les domaines de calculs des 8 processeurs à l'ordre 5.

1.2 Constantes et post-traitement

Cette sous-section vise à définir quelques-unes des valeurs utilisées pour étudier nos résultats. Dans tous les cas tests présentés le milieu considéré est l'air à la densité atmosphérique de $1.225\,kg.m^{-1}$. Quelque soit la variable ou constante v, v_∞ désignera la valeur de v à l'infini.

Notons :

- $C = \sqrt{\dfrac{\gamma p}{\rho}}$, la vitesse du son dans l'air.
- $M\mathrm{a} = \dfrac{\|\vec{\boldsymbol{u}}\|}{C}$, le nombre de Mach.
- $P_\mathrm{c} = \dfrac{2}{\gamma.M\mathrm{a}_\infty^2}.\left(\dfrac{p}{p_\infty} - 1\right)$, le coefficient de pression.
- $\varepsilon_\mathrm{ent} = \dfrac{p}{p_\infty}\left(\dfrac{\rho_\infty}{\rho}\right)^\gamma - 1$, l'erreur d'entropie.
- $P_\mathrm{l} = \dfrac{P_t}{P_{t\infty}}$, le coefficient de perte de pression avec $P_t = p.\left(1 + \dfrac{\gamma-1}{2}.M\mathrm{a}^2\right)^{\gamma.(\gamma-1)}$.
- $r_r = \max\limits_{i=1,n_\mathrm{var}} \dfrac{|\boldsymbol{U}^{n+1}(i) - \boldsymbol{U}^n(i)|_{\mathcal{L}_1}}{|\boldsymbol{U}^1(i) - \boldsymbol{U}^0(i)|_{\mathcal{L}_1}}$, le résidu de la méthode itérative où n_var désigne le nombre de variables du problème et $\boldsymbol{U}^n(i)$ la valeur de la $i^{\text{ème}}$ variable au temps

t^{n+1}.

2 Équations d'Euler

Nous validerons tout d'abord notre modèle sur quelques cas tests non visqueux puis nous poursuivrons par un cas test visqueux. Dans ce travail, l'ordre du maillage courbe et l'ordre de la méthode seront toujours identiques. Nous désignerons par *schéma* $\mathcal{RD}$ *classique* le schéma aux résidus distribués utilisé sur un maillage linéaire par morceaux.

2.1 Écoulement subsonique autour d'un cylindre

Le premier test numérique que nous décrirons consiste en un écoulement subsonique autour d'un cylindre. La géométrie du cylindre est exactement représentée. Nous souhaitons vérifier sur ce cas test typique de l'analyse isogéométrique que la représentation exacte des frontières permet bien d'améliorer les résultats obtenus par le schéma $\mathcal{RD}$. Comme il n'y a pas de production d'entropie dans le cas d'un écoulement d'Euler subsonique, nous utiliserons l'erreur d'entropie ε_{ent} pour évaluer nos résultats.

2.1.1 Description du cas test

On plonge le cylindre autour duquel on souhaite étudier l'écoulement dans un cylindre concentrique de rayon quarante fois plus grand. Notre domaine de calcul est la zone située entre les deux cylindres.

On s'inspire de [6] pour générer notre maillage. Le cylindre central est de rayon 0.5. Le maillage sera constitué de couches de cylindres concentriques. Notons n_c le nombre de cylindres du maillage et c le rapport $\frac{32}{n_c}$. Les rayons des cylindres successifs sont tels que, pour j allant de c à 32 par pas de c,

$$r_j = r_0 \left(1 + \frac{2\pi}{128}\left(\sum_{k=0}^{j-1} \alpha^k\right)\right), \text{ où } \alpha = 1.1648336. \tag{V.1}$$

Nous testerons deux maillages de tailles différentes, contenant respectivement 9×32 sommets et 17×64 sommets. Le premier chiffre désigne le nombre de couches de cylindres constituant les maillages, le second chiffre désigne le nombre de sommets le long de ces cylindres. Les maillages courbes sont représentés figure V.2.2. Les maillages linéaires par morceaux utilisés sont identiques à ceux fournis sauf pour les arêtes de bords qui seront

représentées par des segments. Le maillage V.2(a) compte 288 sommets, 1088 DoF*s* et 512 triangles. Le maillage V.2(b) compte 1088 sommets, 4224 DoF*s* et 2048 triangles.

Les simulations sont faites à l'ordre 3 et on impose comme paramètres de notre cas test :

- Le nombre de Mach à l'écoulement libre : $Ma_\infty = 0.38$.
- Une condition de paroi glissante sur la paroi du cylindre central (paroi turquoise de la figure V.2.2).
- Une condition d'entrée de fluide sur la paroi gauche du cylindre externe (paroi rouge) : $\rho = 1$, $\vec{\boldsymbol{u}} = (0.38, 0.)$, et $p = 0.71428571$.
- Une condition de sortie subsonique sur la paroi droite du cylindre externe (paroi verte).
- On stoppe le calcul pour un résidu itératif r_r inférieur à 10^{-10}.

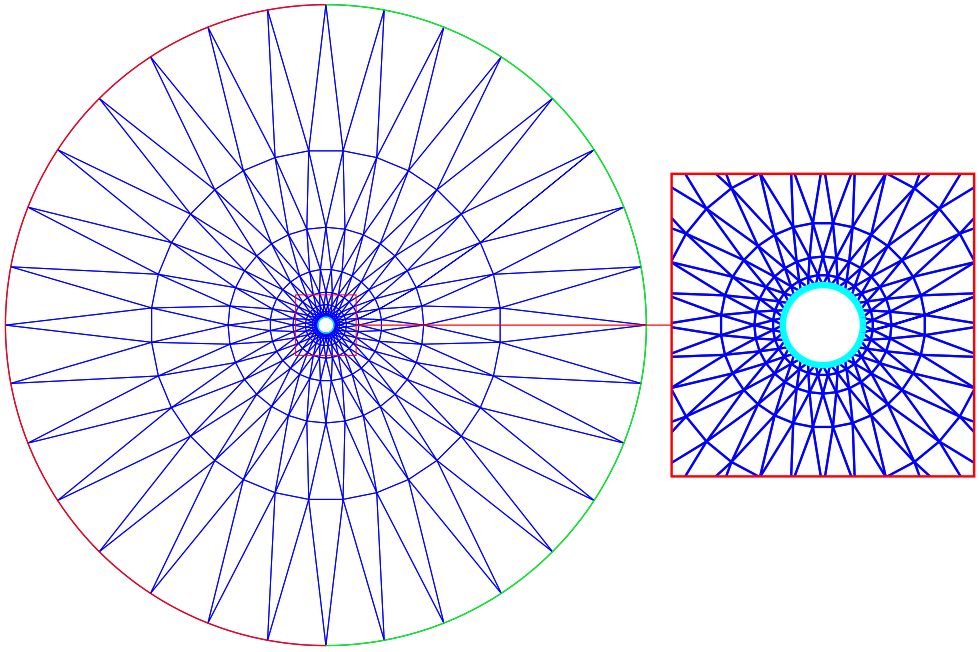
2.1.2 Analyse des résultats

Comparons les résultats obtenus par le modèle isogéométrique d'ordre 3 et le modèle d'ordre élevé classique (également d'ordre 3). Les figures V.2.3 et V.2.4 représentent l'erreur d'entropie commise respectivement sur les maillages 9×32 et 17×64. Nous dessinons également sur ces figures 20 des isolignes de ε_{ent} pour des valeurs comprise entre 0 et 3.5×10^{-3} sur le maillage 9×32 et entre 0 et 1.8×10^{-3} sur le maillage 17×64.

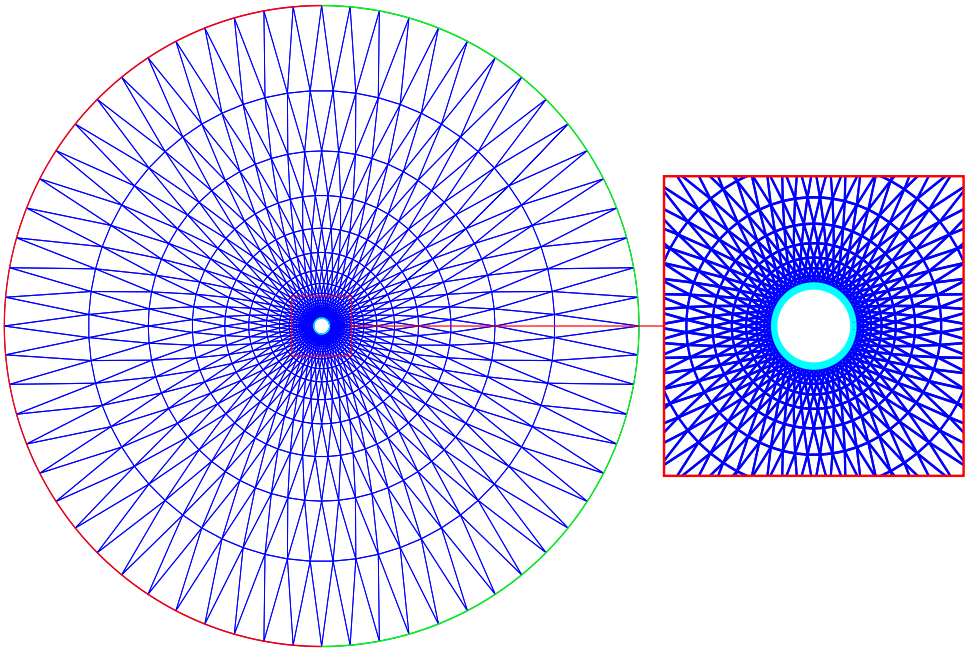
On constate une diminution de l'erreur lorsqu'on utilise la méthode isogéométrique, en particulier à la paroi du cylindre. Pour 9×32 sommets, l'erreur maximale est de l'ordre de 3.6×10^{-3} sur le maillage droit et de 2.6×10^{-3} sur le maillage courbe. Pour 17×64 sommets, cette erreur est de l'ordre de 1.8×10^{-3} sur le maillage droit et de 4.3×10^{-4} sur le maillage courbe. La valeur maximale de l'erreur commise est donc réduite de 27% sur le maillage courbe 9×32 et de 76% sur le maillage 17×64. Ces résultats corroborent ceux de Bassi et Rebay [5] : une mauvaise représentation des frontières limite la convergence en maillage du schéma.

Le tableau V.5 récapitule les valeurs en norme $\mathcal{L}_2$ de l'erreur d'entropie commise sur le domaine complet des différents maillages. L'erreur présentée est normalisée par la surface du domaine. Les résultats confirment l'amélioration de la précision de la méthode par l'utilisation de l'analyse isogéométrique.

Étudions maintenant l'évolution du coefficient de perte de pression et du coefficient de pression des différents modèles.



(a) Maillage de 9×32 sommets



(b) Maillage de 17×64 sommets

Figure V.2.2 – Discrétisation du domaine de calcul du cylindre.

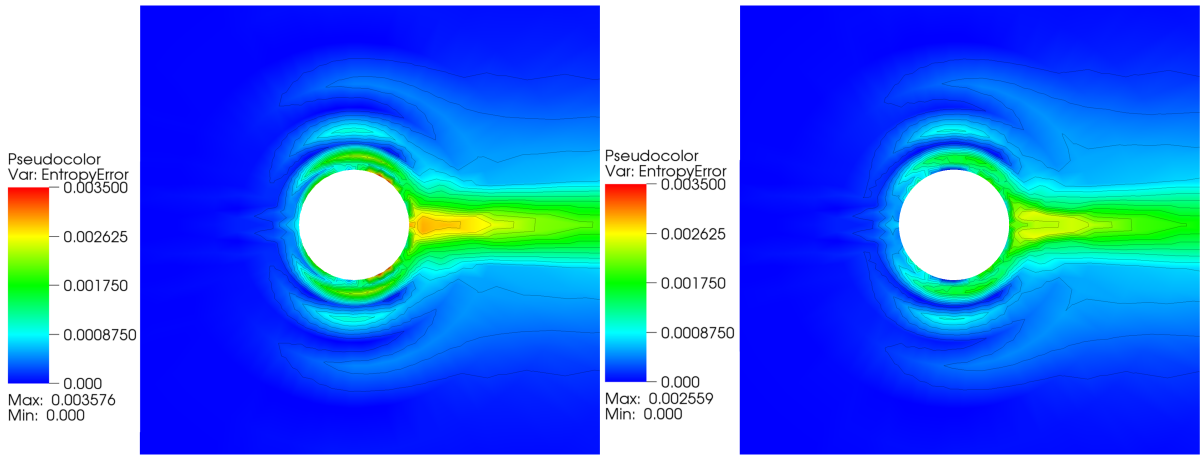


Figure V.2.3 – Erreur d'entropie commise par le schéma $\mathcal{RD}$ classique (à gauche) et le schéma isogéométrique (à droite) sur le maillage 9×32.

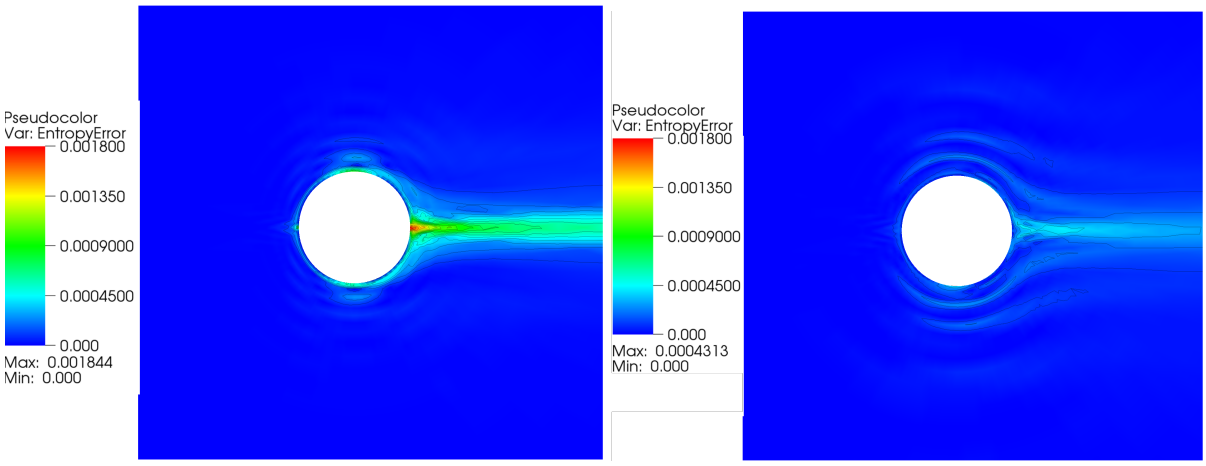
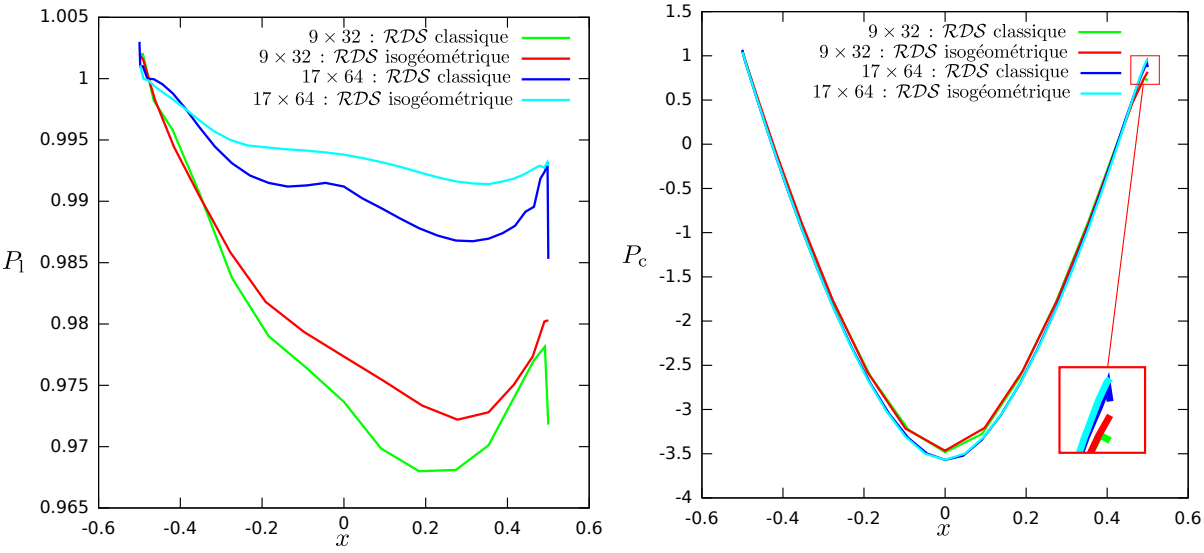


Figure V.2.4 – Erreur d'entropie commise par le schéma $\mathcal{RD}$ classique (à gauche) et le schéma isogéométrique (à droite) sur le maillage 17×64.

Maillage	Schéma $\mathcal{RD}$ classique	Schéma $\mathcal{RD}$ isogéométrique
9×32	4.877735×10^{-06}	4.604823×10^{-06}
17×64	8.797774×10^{-07}	6.028510×10^{-07}

Tableau V.5 – Norme $\mathcal{L}_2$ (normalisée par la surface du domaine) de l'erreur d'entropie commise sur les différents maillages.



(a) Coefficient de perte de pression P_l

(b) Coefficient de pression P_c

Figure V.2.5 – Coefficient de perte de pression et coefficient de pression à la paroi du cylindre pour les différents tests lancés.

La figure V.5(b) représente l'évolution, en fonction de l'abscisse x, du coefficient de pression le long de la paroi du cylindre central. Le coefficient de pression trouvé par la méthode classique est très proche de celui trouvé par la méthode isogéométrique. Dans l'idéal, comme l'énergie du système est conservée, les valeurs trouvées doivent être symétriques par rapport à l'axe $x = 0$. En particulier, la valeur du coefficient de pression trouvée au point de coordonnée $x = 0.5$, donc au point critique situé à l'arrière de cylindre, doit être identique à la valeur trouvée au point $x = -0.5$, *c.-à-d.* qu'elle doit être égale à 1.

On remarque que, en $x = 0.5$, P_c est plus proche de 1 lorsqu'on utilise le modèle isogéométrique.

Considérons maintenant le coefficient de perte de pression du modèle (représenté fi-

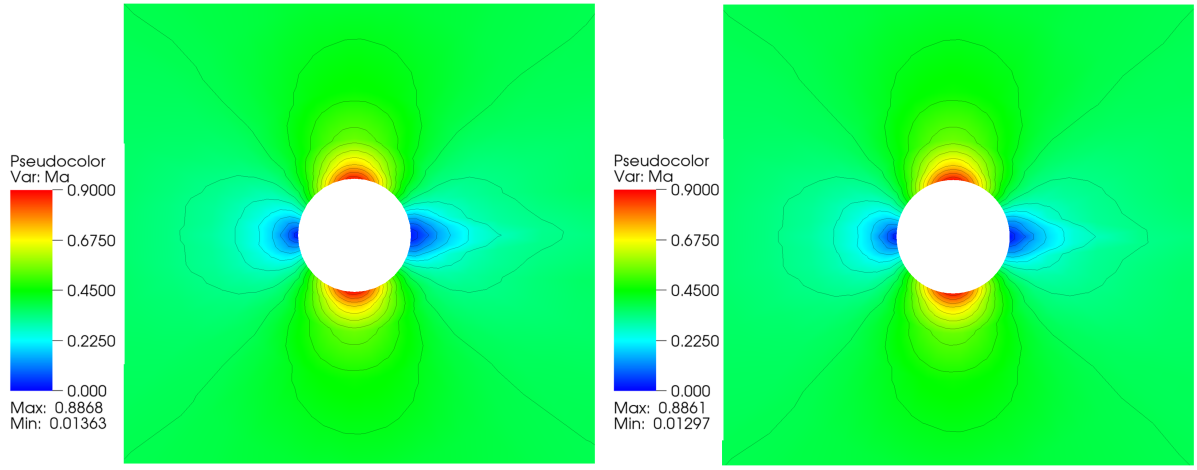


Figure V.2.6 – Nombre de Mach pour le schéma $\mathcal{RD}$ classique (à gauche) et le schéma isogéométrique (à droite) sur le maillage 9×32.

gure V.5(a)). Il ne devrait pas y avoir de perte de pression dans ce cas test, le coefficient associé devrait donc rester égal à 1 le long de la paroi du cylindre central. La perte de pression du schéma est clairement inférieure lorsque la géométrie du problème est exactement représentée. On constate par ailleurs que dans le cas d'une discrétisation linéaire par morceaux, la perte de pression est particulièrement importante au point critique $x = 0.5$.

Pour finir, intéressons-nous au nombre de Mach trouvé sur les différents maillages. La figure V.2.6 fournit les résultats trouvés pour les maillages droit et courbe à 9×32 sommets et la figure V.2.7 donne ceux obtenus sur les maillages à 17×64 sommets. La plage des isovaleurs représentées s'étend de 0 à 0.9. On représente également sur ces deux figures 20 isolignes du nombre de Mach. On sait que les isolignes attendues doivent être symétriques à gauche et à droite du cylindre mais que l'erreur d'entropie du schéma est à l'origine d'une perte de symétrie des résultats. On remarque que, conformément aux erreurs d'entropies évaluées, la perte de symétrie des isolignes de Ma est réduite par l'utilisation de l'analyse isogéométrique.

2.2 Écoulement subsonique dans un canal avec bump circulaire de pente 26.9% ($\alpha = 15°$)

Vérifions maintenant l'efficacité de l'adaptation de maillage sur un cas test physique. On étudie un écoulement subsonique dans un canal avec un bump circulaire de pente 26.9%.

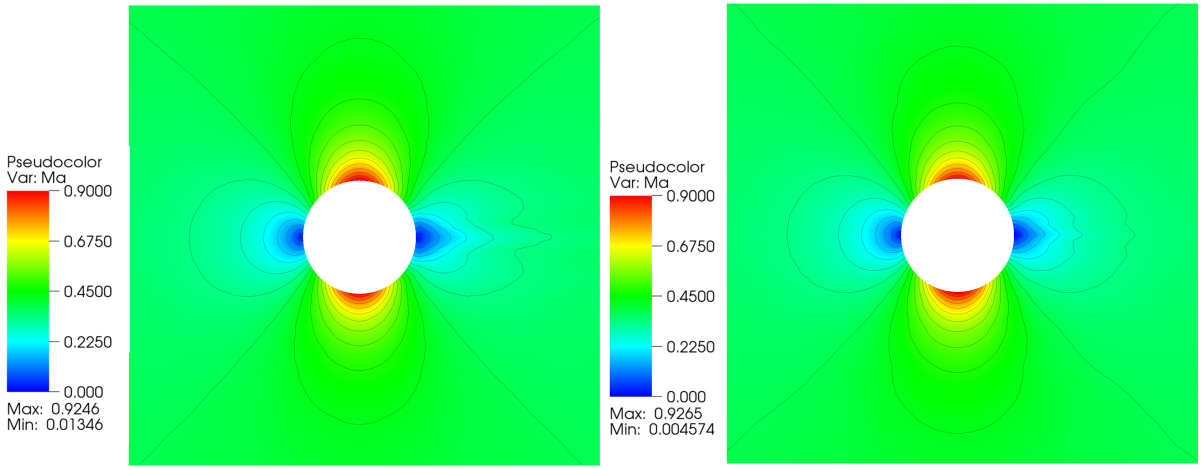


Figure V.2.7 – Nombre de Mach pour le schéma $\mathcal{RD}$ classique (à gauche) et le schéma isogéométrique (à droite) sur le maillage 17×64.

Dans un premier temps, nous comparerons les valeurs obtenues à l'ordre 3 par analyse isogéométrique et par analyse classique sur des maillages grossiers. Nous étudierons par la même occasion les résultats isogéométriques d'ordre 4. Dans une seconde partie, nous raffinerons le maillage courbe et le maillage droit initialement utilisés et nous étudierons les nouveaux résultats trouvés.

2.2.1 Solution initiale sur maillage grossier

2.2.1.1 Description du cas test

On représente la discrétisation courbe du domaine de calcul considéré figure V.2.8. Le maillage isogéométrique représente exactement la géométrie du problème. La discrétisation linéaire par morceaux est identique aux arêtes frontières près. Le sommet du bump est centré en $x = 0$ et le bump est symétrique par rapport l'axe $x = 0$. Le maillage compte 240 sommets et 417 triangles ainsi que 896 DoFs à l'ordre 3 et 1969 DoFs à l'ordre 4.

Présentons les paramètres de notre cas test. On impose :

- Le nombre de Mach à l'écoulement libre : $Ma = 0.5$.
- Une condition de paroi glissante sur les parois horizontales (parois rose et turquoise de la figure V.2.8).
- Une condition d'entrée de fluide sur la paroi gauche du bump (paroi verte) : $\rho = 1$, $\vec{\boldsymbol{u}} = (0.5, 0.)$, et $p = 0.71428571$.

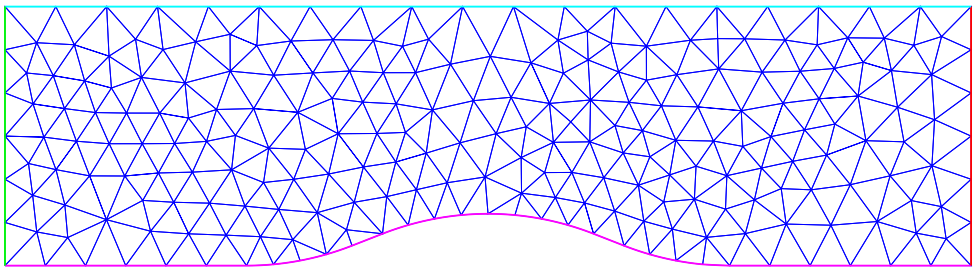


Figure V.2.8 – Maillage du bump circulaire de pente 26.9%.

- Une condition de sortie subsonique sur la paroi droite du bump (paroi rouge).
- On stoppe le calcul pour un résidu itératif r_r inférieur à 10^{-10}.

2.2.1.2 Analyse des résultats

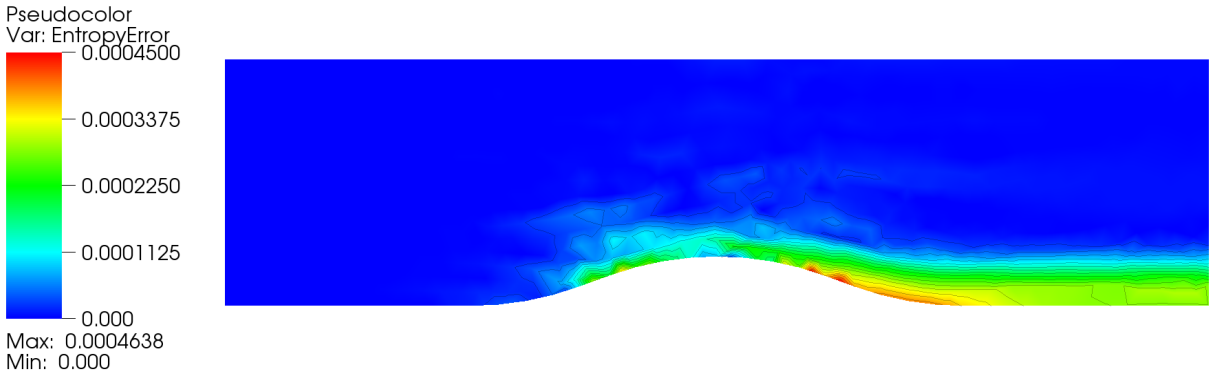
Comme le cas test considéré est subsonique, on ne doit pas avoir de production d'entropie au cours de la résolution de notre problème. La première valeur que nous étudierons est donc l'erreur d'entropie commise. La figure V.2.9 représente les valeurs de l'erreur réalisée à l'ordre 3 par les deux schémas (classique et isogéométrique) ainsi que 20 isolignes comprises entre 0 et 4.5×10^{-4}. La figure V.2.10 fournit les mêmes isovaleurs et isolignes pour le schéma isogéométrique d'ordre 4.

On constate que le schéma isogéométrique d'ordre 3 donne une erreur plus faible que le schéma classique du même ordre. En particulier, l'erreur maximale commise sur le maillage droit est de 4.6×10^{-4} (sous-figure V.2.9(a)) contre une erreur maximale de 3.5×10^{-4} sur le maillage courbe (V.2.9(b)). L'analyse isogéométrique permet donc de réduire l'erreur maximale de 24%. À l'ordre 4, l'erreur maximale commise n'est plus que de 3.3×10^{-4}. De plus, la trainée d'entropie est fortement diminuée.

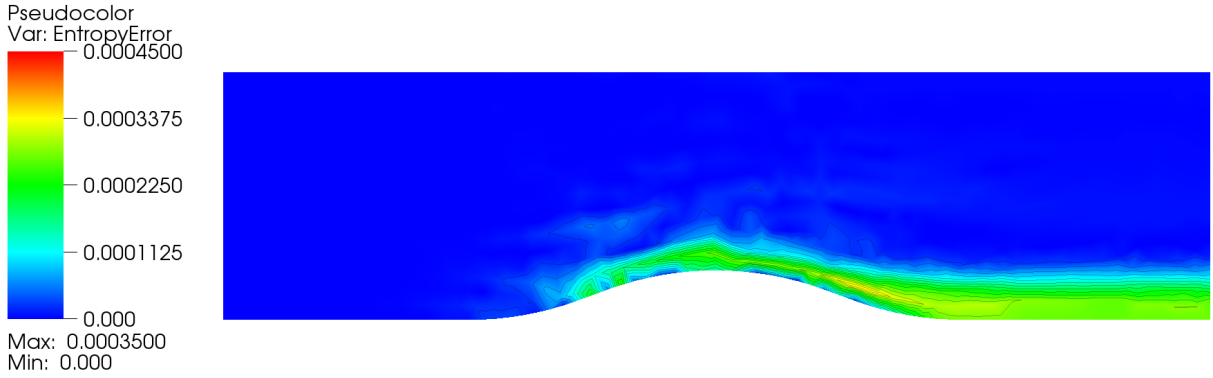
Le tableau V.6 reporte, pour chacune des simulations étudiées, la norme $\mathcal{L}_2$ de l'erreur d'entropie commise sur le domaine de calcul. L'erreur est normalisée par l'aire du domaine. À l'ordre 3, cette erreur est plus faible lorsque le schéma $\mathcal{RD}$ utilisé est isogéométrique. L'erreur est encore réduite par l'utilisation du schéma isogéométrique d'ordre 4.

Regardons maintenant le nombre de Mach aux DoFs de notre maillage. On représente les isovaleurs de celui-ci et 20 isolignes comprises entre 0.4 et 0.9 (figure V.2.11 pour les schémas d'ordre 3 et figure V.2.12 pour le schéma isogéométrique d'ordre 4).

On voit que le nombre de Mach, qui devrait être symétrique par rapport à l'axe $x = 0$, ne l'est pas. Cette perte de symétrie est une conséquence de l'erreur d'entropie commise.



(a) Schéma classique



(b) Schéma isogéométrique

Figure V.2.9 – Erreur d'entropie des schémas classique et isogéométrique d'ordre 3.

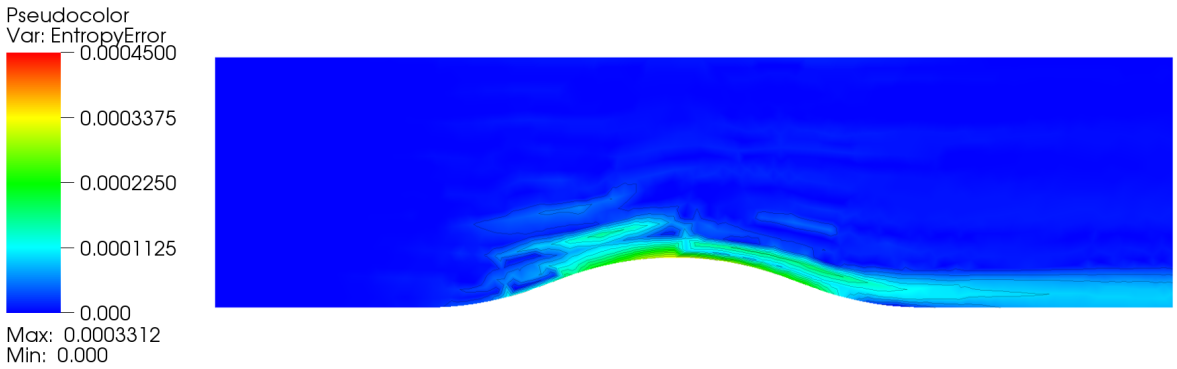
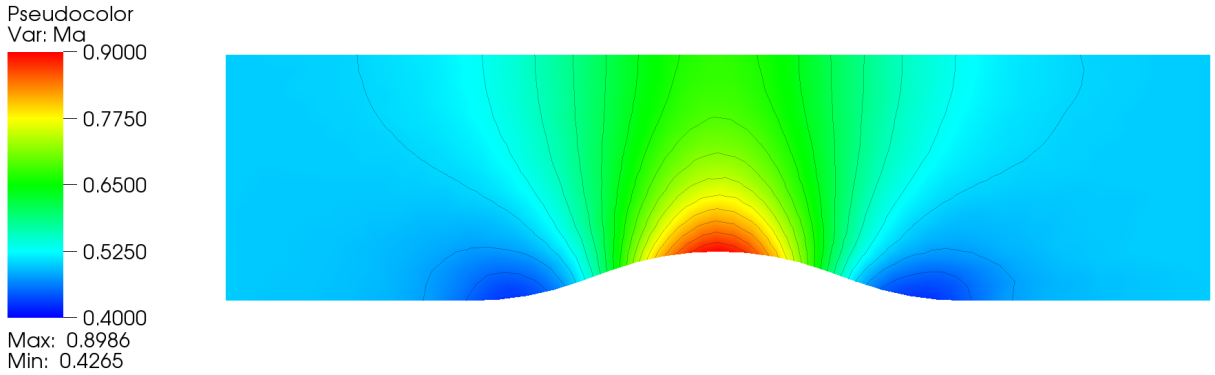
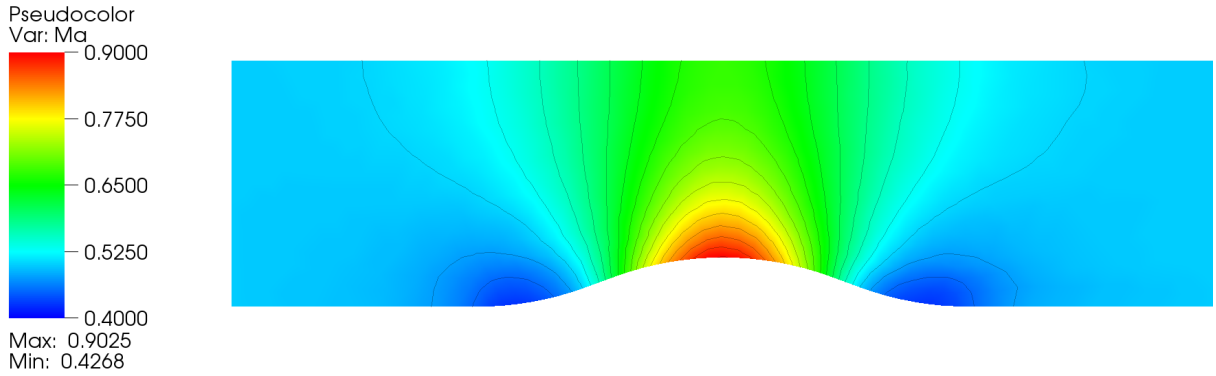


Figure V.2.10 – Erreur d'entropie du schéma isogéométrique d'ordre 4.



(a) Schéma classique



(b) Schéma isogéométrique

Figure V.2.11 – Nombre de Mach des schémas d'ordre élevé classique et isogéométrique.

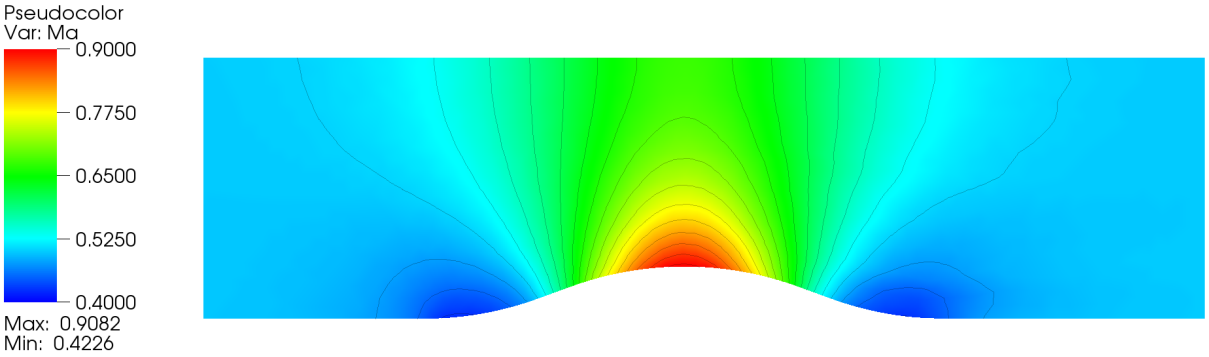


Figure V.2.12 – Nombre de Mach du schéma isogéométrique d'ordre 4.

	Schéma $\mathcal{RD}$ classique	Schéma $\mathcal{RD}$ isogéométrique
Ordre 3	2.106755×10^{-05}	2.032411×10^{-05}
Ordre 4		8.653369×10^{-06}

Tableau V.6 – Norme $\mathcal{L}_2$ (normalisée par l'aire du domaine) de l'erreur d'entropie réalisée par les différents modèles.

Afin de diminuer l'erreur d'entropie des calculs, nous allons raffiner nos maillages initiaux. La norme $\mathcal{L}_2$ de l'erreur d'entropie totale du schéma d'ordre 4 étant suffisamment faible, nous nous intéresserons uniquement aux schémas classique et isogéométrique d'ordre 3.

2.2.2 Solution avec adaptation de maillage

2.2.2.1 Description du cas test

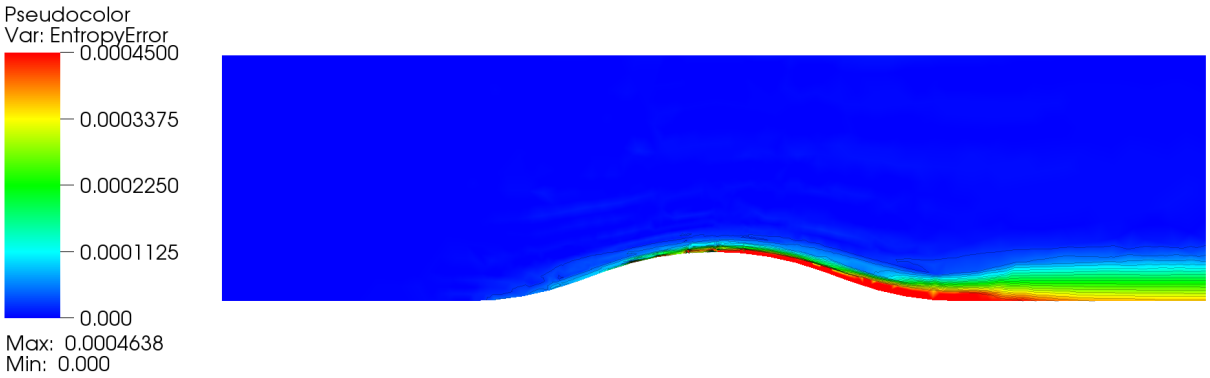
On utilise la procédure de raffinement d'ordre élevé détaillée section 3 pour raffiner, et notre maillage courbe, et notre maillage droit. Le critère de raffinement porte sur la norme $\mathcal{L}_\infty$ de l'erreur d'interpolation commise sur la densité. Le seuil d'erreur maximale voulue sur les sous-maillages associés à nos maillages initiaux est de 0.001, ce seuil sera donc notre critère d'arrêt. La procédure de raffinement est stoppée sur les deux maillages dès que ce critère est atteint l'un d'eux.

On conserve les paramètres du précédent cas test. Tous les calculs sont d'ordre 3.

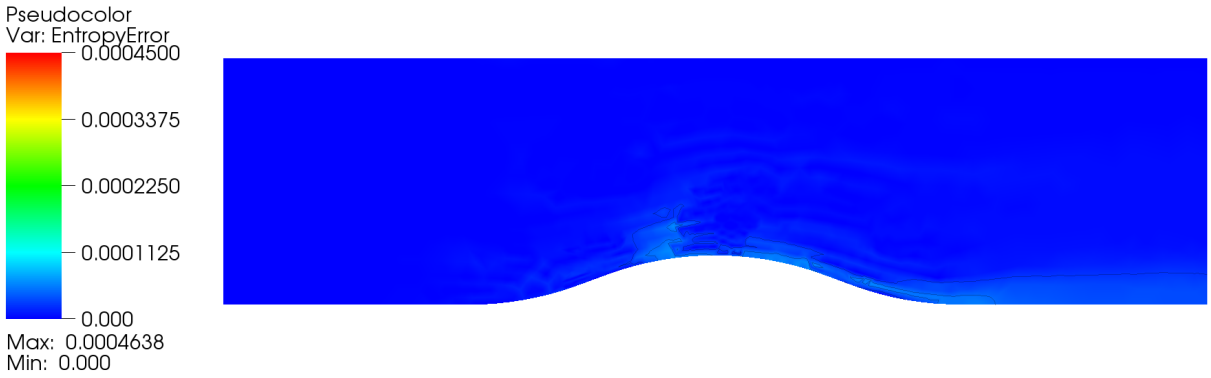
2.2.2.2 Analyse des résultats

On satisfait le critère d'arrêt sur le maillage courbe, après trois vagues de subdivisions. Rappelons que la géométrie des frontières du domaine initial est préservée durant le procédé de h-adaptation. Notre frontière courbe reste donc exactement représentée.

On s'intéresse à nouveau à l'erreur d'entropie de nos modèles. La figure V.2.13 représente la même palette d'isovaleurs et les mêmes isolignes que dans le cas précédent. On constate que l'erreur maximale est maintenant de 1.577×10^{-3} sur le maillage droit et de 6.252×10^{-5} sur le maillage courbe. Si l'erreur d'entropie du modèle isogéométrique est nettement inférieure à celle commise sur le maillage courbe grossier, l'erreur réalisée sur



(a) Schéma classique



(b) Schéma isogéométrique

Figure V.2.13 – Erreur d'entropie des schémas d'ordre élevé classique et isogéométrique.

le maillage droit est beaucoup plus forte. En effet, notre procédure de raffinement respecte la discrétisation initiale de la frontière. Dans le cas d'un maillage droit, raffiner le maillage ne permet pas de mieux représenter la géométrie du domaine et donc, de réduire la production d'entropie.

Le tableau V.7 représente la norme $\mathcal{L}_2$ de l'erreur d'entropie commise sur le domaine de calcul dans le cas d'une analyse classique et dans le cas d'une analyse isogéométrique. On constate que la procédure de raffinement augmente l'erreur totale commise sur le maillage linéaire par morceaux tandis qu'elle la réduit fortement sur le maillage isogéométrique.

Maillage	Schéma $\mathcal{RD}$ classique	Schéma $\mathcal{RD}$ isogéométrique
Maillage initial	2.106755×10^{-05}	2.032411×10^{-05}
Maillage raffiné	2.811759×10^{-05}	2.648264×10^{-06}

Tableau V.7 – Norme $\mathcal{L}_2$ (normalisée par l'aire du domaine) de l'erreur d'entropie réalisée par les différents modèles.

La figure V.2.14 représente les nombres de Mach obtenus. Intéressons nous tout d'abord à la sous-figure V.2.14(a). On constate la présence d'oscillations au sommet du bump. La qualité de notre solution est clairement dégradée par le raffinement réalisé.

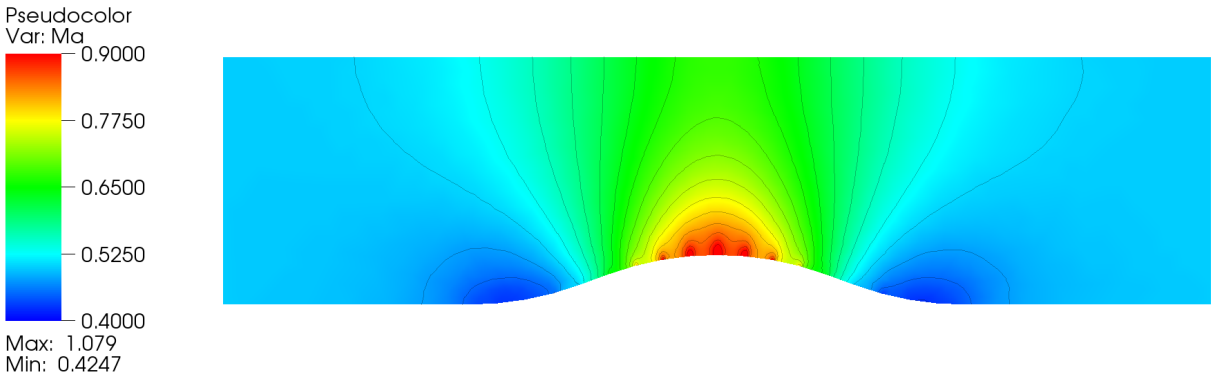
Dans le cas d'une étude isogéométrique, le nombre de Mach représenté sous-figure V.2.14(b) est, par contre, presque symétrique.

Les maillages obtenus après trois raffinements des maillages initiaux sont fournis figure V.2.15. Le maillage droit, représenté sous-figure V.2.15(a), compte 1121 sommets, 4305 DoF*s* et 2064 triangles. Le maillage courbe, représenté sous-figure V.2.15(b), compte 496 sommets, 1889 DoF*s* et 898 triangles. On remarque que le maillage droit est nettement plus raffiné que le maillage courbe. Par ailleurs, l'algorithme de raffinement ne converge pas sur un tel maillage : les zones proches des sommets du maillage initial sont indéfiniment raffinées. Ces zones correspondent aux zones d'oscillation du nombre de Mach.

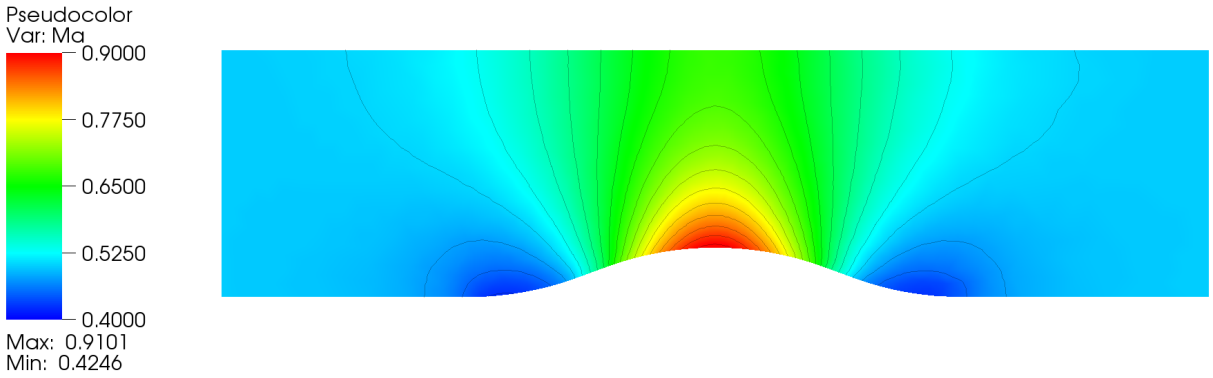
Nous conclurons en notant que créer des maillages grossiers isogéométriques puis les raffiner permet d'obtenir des résultats précis sur des maillages comptant peu de DoF*s*.

2.3 Écoulements autour d'un profil d'aile de section NACA0012

Étudions maintenant le comportement de notre modèle lorsque la géométrie du problème est approchée.

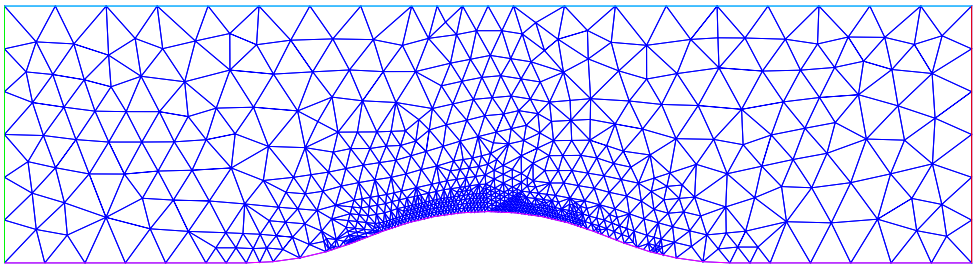


(a) Schéma classique

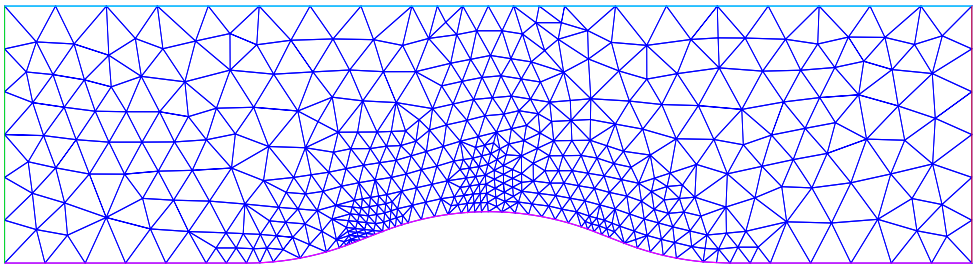


(b) Schéma isogéométrique

Figure V.2.14 – Nombre de Mach des schémas d'ordre élevé classique et isogéométrique.



(a) Maillage droit raffiné



(b) Maillage courbe raffiné

Figure V.2.15 – Maillages du bump après 3 vagues de subdivisions.

On s'intéresse à un profil d'aile de section NACA0012 (l'aile est donc symétrique par rapport à l'axe $y = 0$). Nous considèrerons deux types d'écoulements autour de ce profil, un écoulement subsonique et un écoulement supersonique.

Avant tout il nous faut introduire quelques notations (que nous illustrons figure V.2.16). Soit :

- Γ la frontière du naca et n_Γ la normale locale à la frontière,
- α l'angle d'incidence de l'écoulement,
- $\vec{\mathbf{d}}_{\alpha,x} = (\cos(\alpha), \sin(\alpha))$ et $\vec{\mathbf{d}}_{\alpha,l} = (-\sin(\alpha), \cos(\alpha))$.

Nous utiliserons les coefficients aérodynamiques suivants :

- le coefficient de trainée C_x tel que $C_x = \left(2 \frac{\int_\Gamma p.\vec{\boldsymbol{n}}_\Gamma d\Gamma}{\rho_\infty \|\vec{\boldsymbol{u}}_\infty\|^2} \right) .\vec{\mathbf{d}}_{\alpha,x}$.
- le coefficient de portance C_l défini par $C_l = \left(2 \frac{\int_\Gamma p.\vec{\boldsymbol{n}}_\Gamma d\Gamma}{\rho_\infty \|\vec{\boldsymbol{u}}_\infty\|^2} \right) .\vec{\mathbf{d}}_{\alpha,l}$.

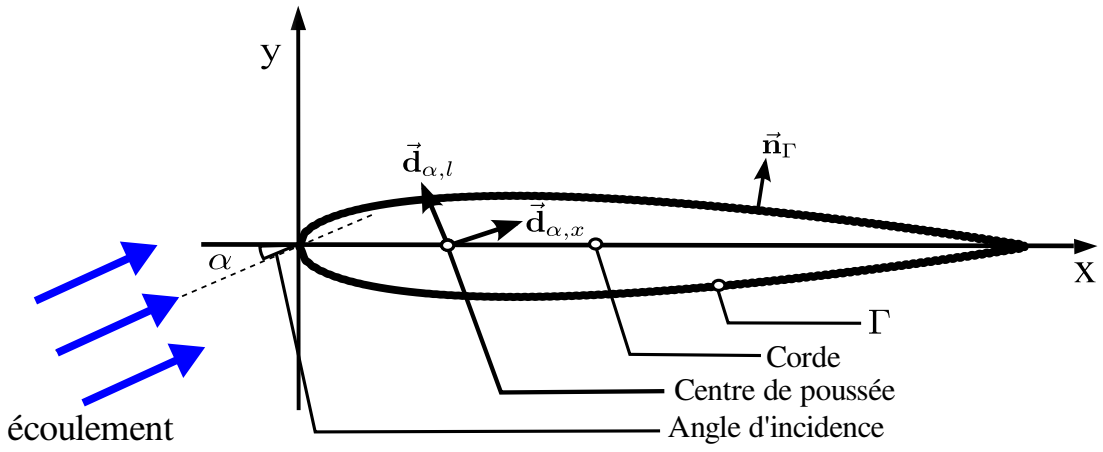


Figure V.2.16 – Profil d'aile à section NACA0012 et notations associées.

2.3.1 Écoulement subsonique

2.3.1.1 Description du cas test

La discrétisation du domaine de calcul est fournie figure V.2.17. Le profil d'aile est plongé dans un cylindre. La taille des mailles est maximale à la paroi du cylindre et décroît régulièrement lorsque l'on s'approche de la paroi du naca. Le maillage est plus particulièrement raffiné au niveau du bord d'attaque et du bord de fuite du naca. Il compte 2355 sommets, 9247 DoFs et 4537 triangles.

On travaille à l'ordre 3. On impose :

- L'angle d'incidence de l'écoulement : $\alpha = 2°$.
- Le nombre de Mach à l'écoulement libre : $Ma = 0.63$.
- Une condition de paroi glissante sur la paroi du naca.
- Une condition de Dirichlet : $\rho = 1$, $\vec{\boldsymbol{u}} = (0.62961622, 0.02198668)$ et $p = 0.71428571$ sur la paroi extérieure au domaine.
- On stoppe le calcul pour un résidu itératif r_r inférieur à 10^{-10}.

2.3.1.2 Analyse des résultats

Les figures V.18(a) et V.18(b) représentent l'erreur d'entropie commise par, respec-

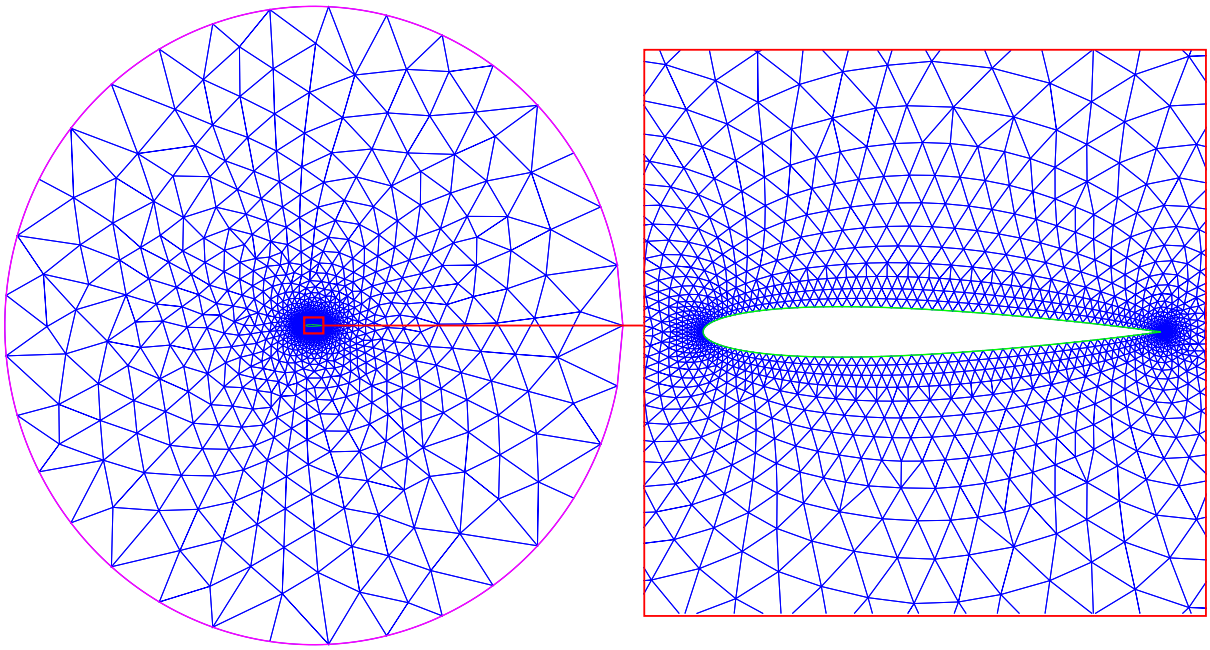
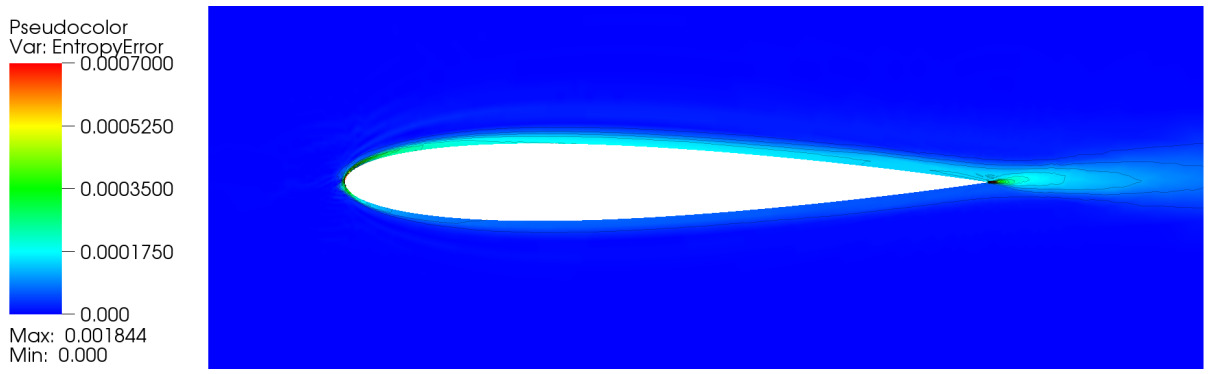


Figure V.2.17 – Maillage du domaine de calcul du NACA0012 (à gauche) et zoom au centre du maillage (à droite).

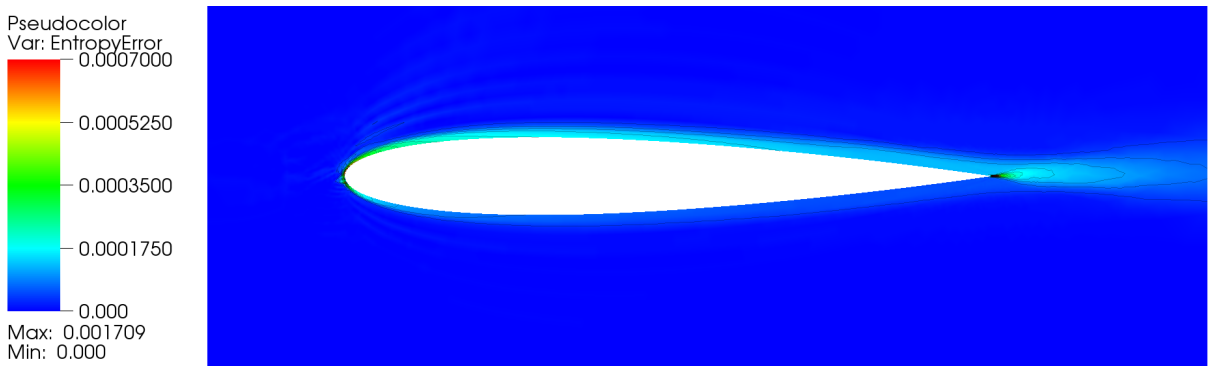
tivement, le schéma aux résidus distribués classique et le schéma aux résidus distribués isogéométrique. Dans les deux cas, on trace 20 isolignes de l'erreur, les isolignes étant comprises entre 0 et 7×10^{-4}. Comme, une fois de plus, le cas présenté est subsonique, on souhaite que l'erreur soit la plus faible possible. Visuellement, l'erreur d'entropie des deux méthodes semble presque identique.

On note cependant un maximum d'erreur de 1.844×10^{-3} sur le maillage droit contre un maximum de 1.709×10^{-3} sur le maillage courbe, soit une réduction d'erreur de 7.3% sur ce dernier. La norme $\mathcal{L}_2$ de l'erreur commise sur la totalité du maillage est de 4.34×10^{-05} pour les deux types de schémas. Cette égalité s'explique facilement : On ne commet une erreur que dans des zones extrêmement proches du naca alors que le diamètre du cylindre délimitant le domaine de calcul est quarante fois plus grand que la longueur du naca. De plus, les erreurs commises par les deux méthodes sont très faibles, leurs différences sont donc insignifiantes à l'échelle du domaine de calcul complet.

Afin de mieux estimer l'erreur commise nous allons étudier ses valeurs le long de la paroi du naca. On représente l'évolution de l'erreur d'entropie à la paroi en fonction de x figure V.2.19. La sous-figure V.2.19(a) parcourt la totalité du naca tandis que les sous-figures V.2.19(b) et V.2.19(c) s'intéressent plus particulièrement au bord d'attaque et au bord de fuite du naca.

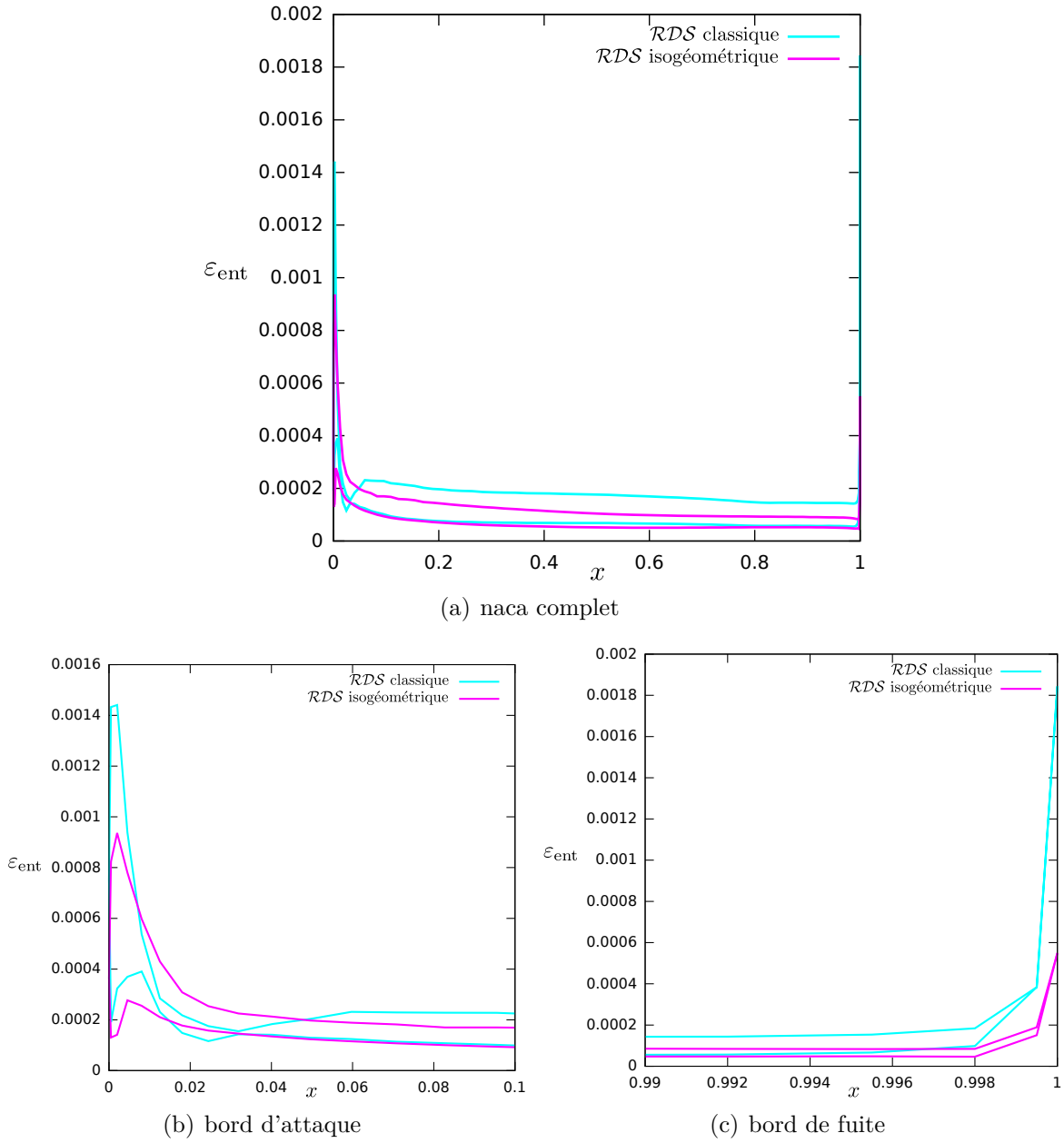


(a) Maillage droit



(b) Maillage courbe

Figure V.2.18 – Erreur d'entropie pour les schémas $\mathcal{RD}$ classique et isogéométrique.

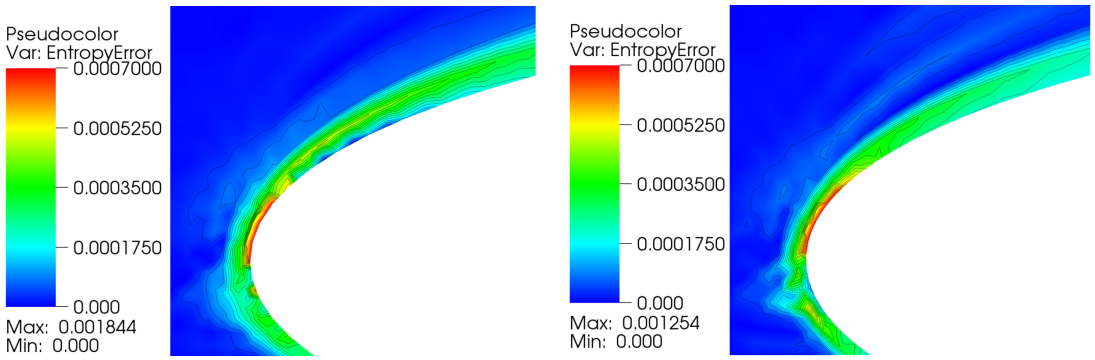


(a) naca complet

(b) bord d'attaque

(c) bord de fuite

Figure V.2.19 – Erreur d'entropie à la paroi du naca.



(a) $\mathcal{RDS}$ classique (b) $\mathcal{RDS}$ isogéométrique

Figure V.2.20 – Erreur d'entropie des schémas classique et isogéométrique au bord d'attaque du naca.

On constate que, globalement, l'erreur d'entropie commise à la paroi est plus faible pour un maillage courbe que pour un maillage droit. On remarque par ailleurs des pics d'erreur au niveau du bord d'attaque et du bord de fuite du naca. Le maximum de ces pics est nettement réduit par l'utilisation de l'analyse isogéométrique. On voit également sous-figure V.2.19(b) que l'erreur d'entropie du schéma isogéométrique domine localement celle du schéma classique (pour x compris entre 0.01 et 0.05 sur l'extrados et x compris entre 0.015 et 0.03 sur l'intrados). Un zoom sur l'erreur d'entropie trouvée dans cette zone (figure V.2.20) permet de constater que l'erreur du schéma isogéométrique est adjacente à la paroi tandis que l'erreur du schéma classique est légèrement décollée de la paroi. C'est pourquoi, dans le cas droit, bien que l'erreur ne diminue pas, elle n'apparait plus sur notre graphe. Le schéma $\mathcal{RD}$ isogéométrique permet donc bien de réduire la production d'erreur d'entropie à la paroi, et ce, même dans le cas d'une géométrie approchée.

Intéressons-nous au nombre de Mach évalué par les différentes méthodes. La figure V.2.21 représente les isolignes de Ma pour la méthode isogéométrique ainsi que pour le schéma $\mathcal{RD}$ classique. Les isolignes tracées sont équiréparties entre 0 et 1. On constate que les isolignes des deux méthodes sont parfaitement confondues. On explique la similitude des résultats par la faiblesse de l'erreur commise.

Pour conclure, l'approximation de la géométrie d'un domaine par une courbe circulaire par morceaux permet de réduire l'erreur commise sur une discrétisation linéaire par

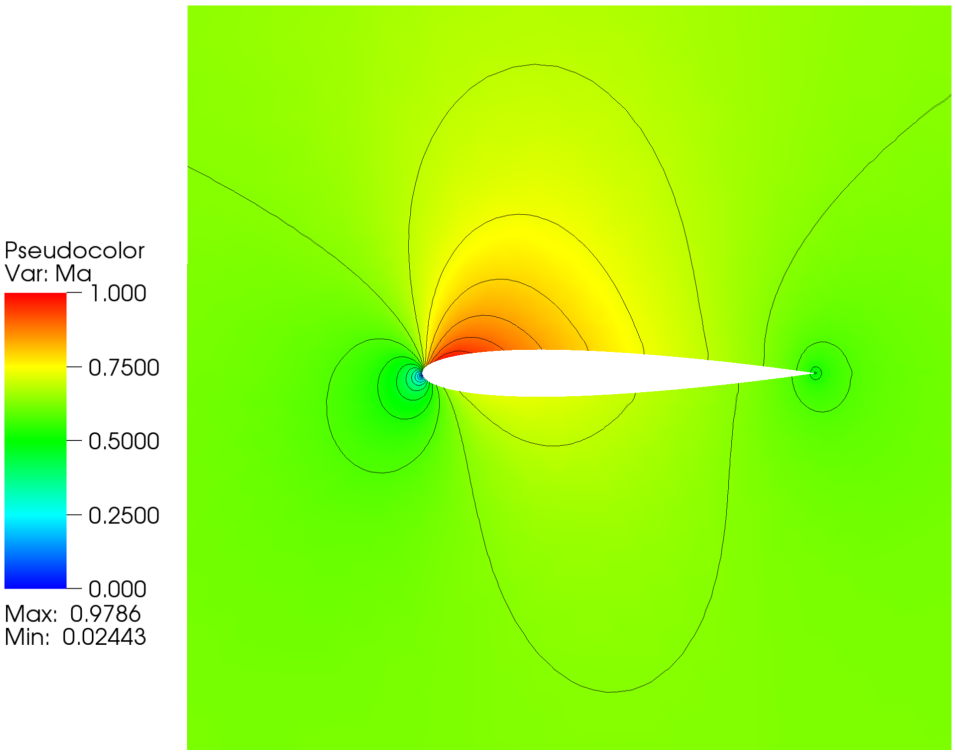


Figure V.2.21 – Nombre de Mach pour $\alpha = 2°$ et pour un nombre de Mach à l'écoulement libre de 0.63.

morceaux. Cette réduction est cependant moins importante que lorsque la géométrie est exactement représentée.

2.3.2 Écoulement supersonique

Ce cas test vise à vérifier que notre schéma permet bien la capture des chocs.

2.3.2.1 Description du cas test

Le maillage de départ est celui que nous avons présenté figure V.2.17. La simulation est réalisée à l'ordre 3. On impose :

- L'angle d'incidence de l'écoulement : $\alpha = 0°$.
- Le nombre de Mach à l'écoulement libre : $Ma = 1.2$
- Une condition de paroi glissante sur la paroi du naca.
- Une condition de Dirichlet : $\rho = 1$, $\vec{\boldsymbol{u}} = (1.2, 0.)$ et $p = 0.71428571$ sur la paroi extérieure au domaine.

On lance un premier calcul que l'on stoppe une fois le résidu itératif inférieur à 10^{-4}. Ce calcul préalable nous permet de capturer grossièrement le choc détaché qui se forme devant le bord d'attaque du naca et donc, de raffiner notre maillage sur ce choc.

On calcule alors la solution sur le maillage raffiné, le critère d'arrêt étant un résidu itératif inférieur à 10^{-10}. Visuellement, le choc semble suffisamment précis pour stopper la procédure de raffinement.

2.3.2.2 Analyse des résultats

On représente figure V.2.22 le maillage raffiné obtenu ainsi que, figures V.23(a) et V.23(b), le nombre de Mach trouvé sur le maillage initial et sur le maillage raffiné. Les 20 isolignes représentées sont comprises entre 0 et 1.5.

On constate que le choc est capturé sur les deux maillages, il est cependant beaucoup plus précis sur le maillage raffiné, les isolignes obtenues sur ce maillage présentent également moins d'oscillations.

Comme le profil d'aile de section NACA0012 est symétrique par rapport à l'axe $y = 0$ et que l'angle d'incidence de l'écoulement est nul, le coefficient de portance est théoriquement nul. Afin d'évaluer les résultats obtenus par notre méthode, nous regarderons

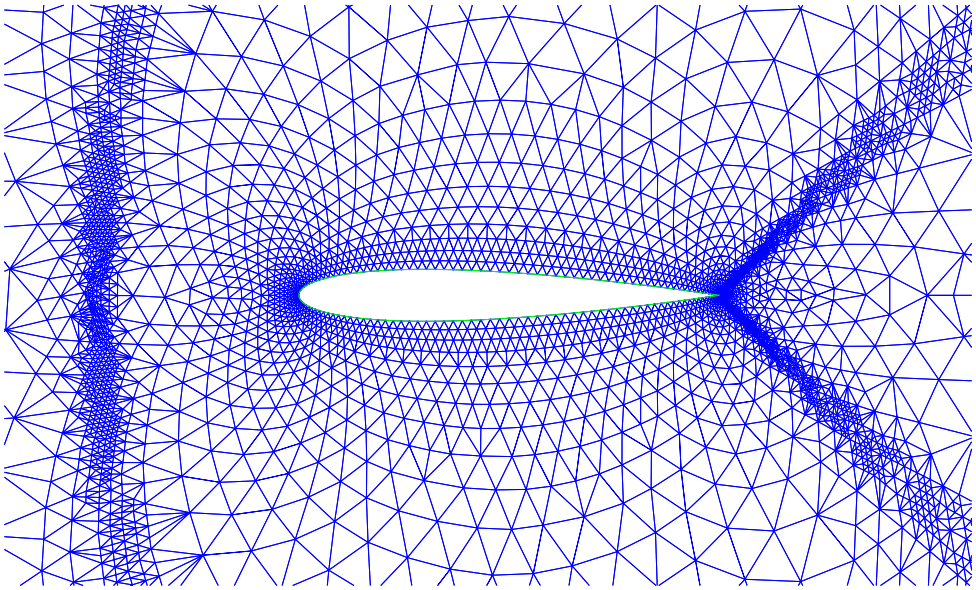
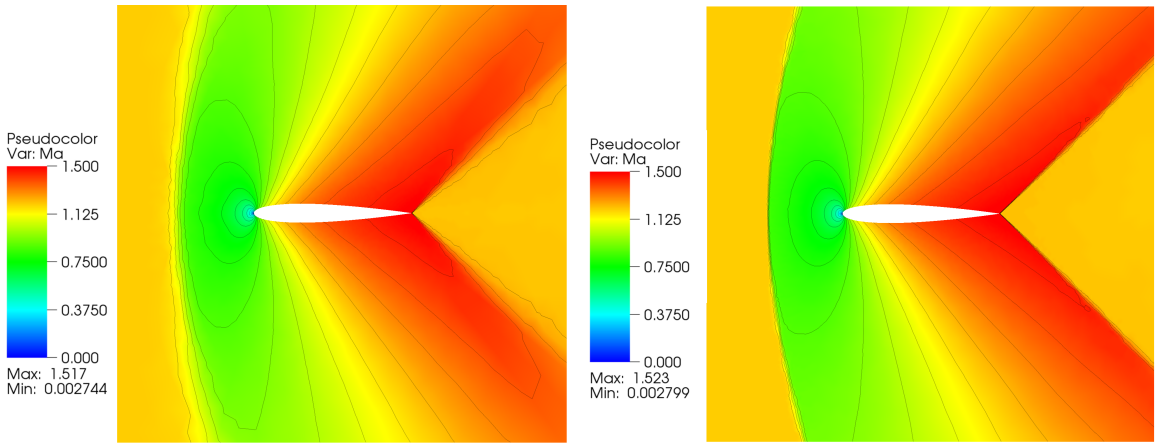


Figure V.2.22 – Maillage raffiné obtenu après une première évaluation du choc.



(a) Maillage initial (b) Maillage raffiné

Figure V.2.23 – Capture de choc par le schéma $\mathcal{RD}$ isogéométrique :
La figure (a) fournit le nombre de Mach obtenu sur le maillage initial.
La figure (b) fournit le nombre de Mach obtenu sur le maillage raffiné.

donc la valeur de C_l de notre schéma. Pour le maillage initial, le coefficient de portance trouvé est de $-6.348674 \times 10^{-04}$. Pour le maillage final, on obtient $C_\mathrm{l} = 5.574759 \times 10^{-05}$. Notre méthode est donc relativement précise. À titre de comparaison, le coefficient de portance trouvé par le schéma aux résidus distribués classique sur ce dernier maillage est de 8.817235×10^{-05}. L'utilisation de l'analyse isogéométrique permet donc une réduction d'erreur de 36.8% sur ce terme. Le coefficient de trainée associé est égal à 9.680553×10^{-02} sur le maillage droit final et 9.686984×10^{-02} sur le maillage courbe.

3 Équations de Navier-Stokes

Nous allons maintenant tester notre code sur un cas test typique des écoulements visqueux.

3.1 Écoulement subsonique autour d'un cylindre

Comme sous-section 2.1 on considère un écoulement subsonique autour d'un cylindre mais, ici, on ne néglige plus la viscosité.

3.1.1 Description du cas test

Le maillage du domaine de calcul est le maillage 9×32 précédemment présenté (voir figure V.2(a)), le cylindre central est donc de rayon 0.5.

On travaille à l'ordre 3. Le cas test considéré est le suivant :

- Le nombre de Mach à l'écoulement libre est tel que : $Ma = 0.3$.
- L'écoulement étudié est laminaire à nombre de Reynolds $Re = 40$.
- On impose une condition de paroi adhérente sur la paroi du cylindre central : $\vec{\boldsymbol{u}} = 0$.
- On impose une condition d'entrée de fluide sur la paroi gauche du cylindre externe : $\rho = 1$, $\vec{\boldsymbol{u}} = (0.3, 0.)$, et $p = 0.71428571$.
- On impose une condition de sortie subsonique sur la paroi droite du cylindre externe.
- On stoppe le calcul pour un résidu itératif r_r inférieur à 10^{-10}.

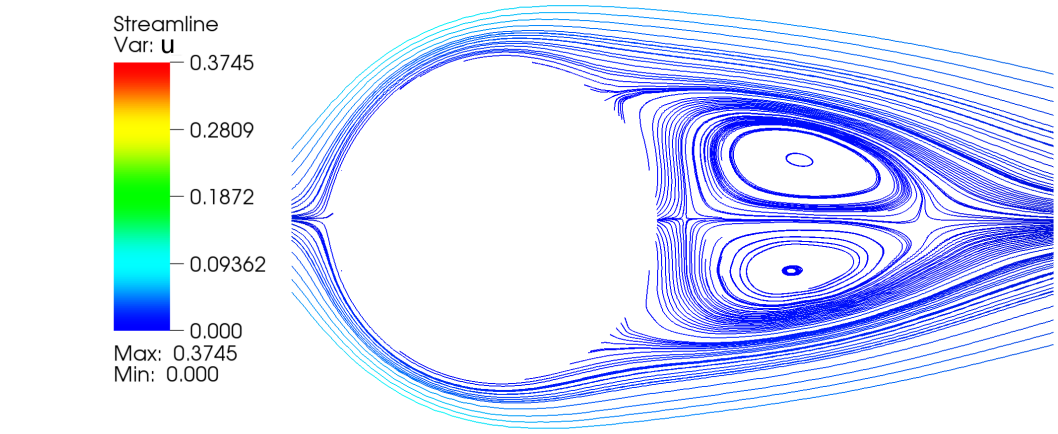
3.1.2 Analyse des résultats

La figure V.3.2 représente quelques-unes des lignes de courant (sous-figure V.3.2(a)) obtenues ainsi que les isovaleurs et vingt isolignes du nombre de Mach (sous-figure

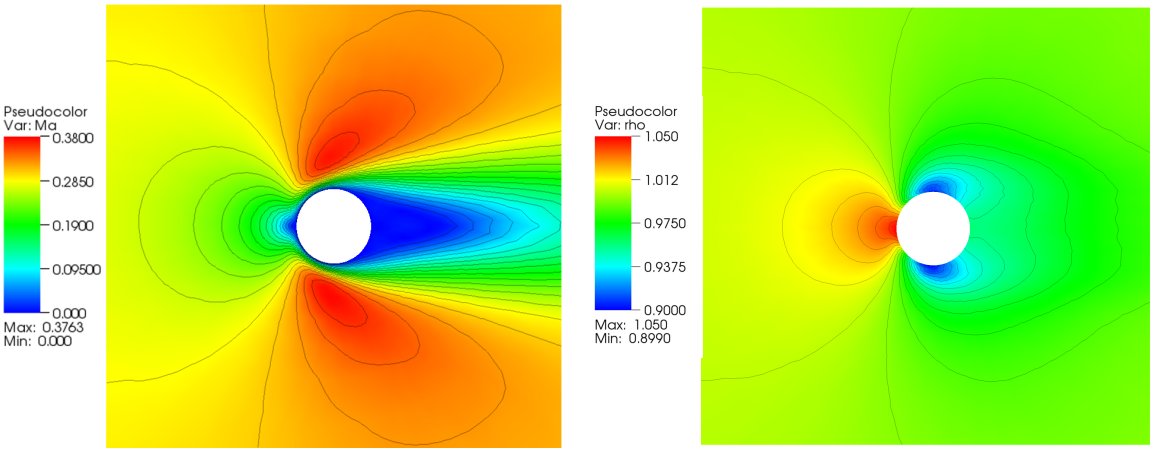
V.3.2(b)) et de la densité volumique (sous-figure V.3.2(c)) à la fin de la simulation. Les isolignes du nombre de Mach sont équiréparties entre 0 et 0.38. Celles de la densité s'étendent de 0.9 à 1.05.

Les résultats sont de la forme attendue ([60]) : les lignes de courant de l'écoulement montrent un décollement à l'arrière du cylindre ainsi que deux zones tourbillonnaires antisymétriques de rayons inférieurs à celui du cylindre. Du fait de la viscosité du fluide, ce dernier adhère à la paroi, le nombre de Mach est donc localement nul à la paroi du cylindre. Il est également très faible derrière celui-ci. Il augmente rapidement sur les cotés du cylindre lorsque l'on s'éloigne de sa couche limite.

La densité volumique est également bien approchée, elle augmente à l'amont du cylindre et présente deux zones dépressionnaires sur les côtés de ce dernier.



(a) Lignes de courant



(b) Nombre de Mach

(c) densité volumique

Figure V.3.2 – Résultats obtenus par le schéma $\mathcal{RD}$ isogéométrique.

Conclusion et perspectives

Cette thèse avait pour objectif d'adapter un schéma aux résidus distribués à l'analyse isogéométrique. Nous présenterons ici le bilan du travail effectué ainsi qu'une liste non exhaustive des futures recherches et améliorations possibles. Nous tenterons de respecter l'ordre dans lequel nous avons présenté notre travail.

Schéma aux résidus distribués isogéométrique

Intéresserons nous tout d'abord au couplage du schéma aux résidus distribués et de l'analyse isogéométrique présenté chapitre II. Le couplage réalisé permet la résolution, dans le cas stationnaire, des équations d'Euler et de Navier-Stokes. En dimension 2 les éléments triangulaires et quadrangulaires d'ordre 2 à 6 sont disponibles. Les tétraèdres d'ordre 3 et 4 sont codés pour les calculs tridimensionnels. Le travail réalisé a été parallélisé.

Perspectives :

Le code isogéométrique obtenu reste relativement couteux en temps comparé au schéma $\mathcal{RD}$ classique. Le temps de calcul additionnel s'explique en grande partie par le nombre de points de quadrature nécessaires à l'évaluation des intégrales de contour du schéma.

D'un point de vue informatique, traiter des éléments dont tous les bords sont droits comme des éléments classiques et leur appliquer les méthodes de quadrature associées permettrait un gain de temps non négligeable.

L'utilisation des points de quadrature de Taylor pour l'évaluation des intégrales de

Flux semble permettre une nette amélioration des résultats, tant pour un schéma de type Galerkin discontinu ([61]) que pour un schéma de type éléments finis ([15]). Tester l'influence de ces points de quadratures sur une schéma aux résidus distribués pourrait donc être intéressant.

Le schéma présenté dans ce mémoire est adapté aux problèmes stationnaires laminaires. La forme instationnaire du schéma vient d'être construite par R. Abgrall. Outre l'implémentation de nouveaux éléments (*par ex.* les hexaèdres d'ordre 3 et 4) pouvant être utiles, l'adaptation d'un modèle de turbulence à l'analyse isogéométrique pourrait être intéressante.

Génération de maillages courbes d'ordre 3

Le mailleur développé chapitre III génère, à partir d'un maillage P_1, un maillage à bords courbes. Les coins et les arêtes droites peuvent être spécifiés dans la géométrie initiale. Une arête de bord est approchée par un arc de cercle évalué à partir du cercle osculateur associé à chacune des extrémités de l'arête. La validité du maillage généré est vérifiée. Si des éléments présentant des arêtes croisées sont détectés, on résout le problème par une procédure d'insertion de nœud et de forçage d'arêtes. Le maillage obtenu est conforme, il garantit la continuité de nos fonctions de base sur la totalité du domaine de calcul et constitue une bonne approximation de nos frontières. Seules ses arêtes de bord sont courbes. Dans le cas de géométries linéaires ou circulaires par morceaux, le maillage obtenu est un maillage exact.

Perspectives :

Les maillages générés fournissent une bonne approximation de la géométrie du domaine dans le cas où celle-ci n'est pas connue. Cependant, lorsque l'équation régissant la géométrie de notre domaine est connue, nous n'utilisons pas cette information et nous n'obtenons qu'une approximation de nos frontières.

Nous souhaitons donc implémenter prochainement la possibilité d'intégrer l'équation exacte d'une géométrie dans notre mailleur.

Par ailleurs, bien qu'ayant l'avantage d'être extrêmement robuste, la procédure de vérification et mise en conformité du maillage, détruit les couches limites que celui-ci peut contenir. Intuitivement, notre maillage ne peut contenir des couches limites que si

celles-ci sont constituées d'éléments courbes. Une piste intéressante, actuellement en cours d'étude, consiste à générer des couches limites courbes par la résolution de l'équation de l'élasticité sur le maillage courbe obtenu après vérification.

Les frontières courbes générées par notre mailleur sont $\mathcal{G}_0$ par morceaux.

- D'un point de vue numérique, il pourrait être intéressant de tester l'effet d'une continuité plus élevée entre les éléments. Ceci nécessiterait donc d'adapter notre mailleur à la génération de frontières plus régulières. Les documents [62], [63] et [64] fournissent des pistes sur la génération de patchs triangulaires à plusieurs éléments. La continuité entre les éléments d'un même patch est alors préservée.
- Pour finir, il serait intéressant d'étendre l'algorithme de génération de maillages courbes à la dimension 3.

Adaptation isogéométrique de maillages courbes d'ordre élevé

Nous terminerons ce bilan par les méthodes d'adaptation développées. La p-adaptation implémentée est intrinsèque au code de calcul, elle permet de passer d'un maillage courbe hybride bidimensionnel d'ordre 2 à 5 à un maillage d'ordre maximal 6. La h-adaptation est, quant à elle, implémentée pour des maillages triangulaires bidimensionnels et pour une métrique isotrope. Notre méthode de raffinement nécessite le maillage d'ordre élevé initial ainsi que sa version P_1 subdivisée. Un estimateur d'erreur P_1 permet d'évaluer la métrique en chaque DoF du sous-maillage. On associe cette métrique au maillage d'ordre élevé initial. Ceci nous permet d'évaluer à l'ordre élevé la longueur de chacune des arêtes du maillage. Les fonctions de base choisies pour ce calcul sont les fonctions de base de Bernstein rationnelles imposées par la géométrie. La carte des longueurs d'arêtes obtenue nous permet de créer une carte des éléments à subdiviser. Deux types de découpes sont disponibles, la *découpe* 1-2 et la *découpe* 1-4. Le maillage de sortie est un maillage courbe d'ordre et de géométrie identiques à celui d'entrée. Il peut contenir des éléments internes courbes.

Perspectives :

- Nous souhaitons, à terme, remplacer notre estimateur d'erreur d'interpolation $P1$ par un estimateur d'erreur d'extrapolation d'ordre élevé sur les fonctions de base $\mathcal{NURBS}$. Les documents [21] (page 30) et [22] (annexe A.1.2) fournissent des pistes intéressantes quand à la construction d'un estimateur d'erreur d'interpolation la-

grangienne d'ordre élevé. Ceci reste à étudier dans le cadre de futurs travaux.

- Il serait utile d'adapter les opérateurs de remaillage aux maillages courbes. Il faudrait chercher, par exemple, des méthodes de retournement d'arêtes et de suppression d'éléments permettant de préserver la géométrie.
- L'estimateur d'erreur que nous utilisons étant anisotrope il serait intéressant d'étudier la construction de maillages anisotropes. Cette construction nécessite les opérateurs de remaillage précédemment évoqués.
- Bien entendu, l'extension aux cas tridimensionnels des méthodes d'adaptation est également un axe de recherche intéressant.

Annexe 1

1 Matrices Jacobiennes dans $\mathbb{R}^3$

À de nombreuses reprises dans ce document, nous avons utilisé la formulation quasi-linéaire des équations d'Euler. Cette forme nécessite le calcul des matrices jacobiennes du flux advectif. le vecteur formé par ces matrices est noté :

$$\vec{\boldsymbol{\lambda}} = \begin{pmatrix} \dfrac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_1(\boldsymbol{U})}{\partial \boldsymbol{U}} \\ \dfrac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_2(\boldsymbol{U})}{\partial \boldsymbol{U}} \\ \dfrac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_3(\boldsymbol{U})}{\partial \boldsymbol{U}} \end{pmatrix}$$

De plus, certaines valeurs utilisées dans le schéma $\mathcal{RD}$ (*par ex.* le calcul des coefficients de distribution $B_i^{T,*}$ ou celui de θ, le coefficient associé à la dissipation), sont obtenues par projection dans la base des vecteurs propres de la matrice $\boldsymbol{K} = \vec{\boldsymbol{\lambda}}.\vec{\boldsymbol{n}}$. $\vec{\boldsymbol{n}} = {}^T(n_x, n_y, n_z)$ étant un vecteur de $\mathbb{R}^3$. Nous donnons donc dans cette annexe les matrices jacobiennes du flux ainsi que les valeurs propres et les vecteurs propres de $\boldsymbol{K}$. On garde dans cette annexe les notations adoptées tout le long de ce manuscrit sauf pour les composantes du vecteur vitesse que l'on notera $\vec{\boldsymbol{u}} = (u, v, w)$.

1.1 Forme des matrices

Nous donnons ici les jacobiennes $3D$. Les jacobiennes $2D$ peuvent être retrouvées en supprimant les lignes et colonnes correspondant à la $3^{\text{ème}}$ variable d'espace (donc les $4^{\text{ème}}$ lignes et colonnes) et en remplaçant w par 0.
On note $\mathscr{E}_c = \frac{\|\vec{\boldsymbol{u}}\|^2}{2}$ l'énergie cinétique, $\mathcal{H} = E + \frac{p}{\rho}$ l'enthalpie et on pose $\gamma_1 = \gamma - 1$

$$\frac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_1(\boldsymbol{U})}{\partial \boldsymbol{U}} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ \gamma_1 \mathscr{E}_c - u^2 & (2-\gamma_1)\,u & -\gamma_1 v & -\gamma_1 w & \gamma_1 \\ -uv & v & u & 0 & 0 \\ -uw & w & 0 & u & 0 \\ u\,(\gamma_1 \mathscr{E}_c - \mathcal{H}) & -\gamma_1 u^2 + \mathcal{H} & -\gamma_1 uv & -\gamma_1 uw & \gamma u \end{pmatrix}$$

$$\frac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_2(\boldsymbol{U})}{\partial \boldsymbol{U}} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ -uv & v & u & 0 & 0 \\ \gamma_1 \mathscr{E}_c - v^2 & -\gamma_1 u & (2-\gamma_1)\,v & -\gamma_1 w & \gamma_1 \\ -vw & 0 & w & v & 0 \\ v\,(\gamma_1 \mathscr{E}_c - \mathcal{H}) & -\gamma_1 uv & -\gamma_1 v^2 + \mathcal{H} & -\gamma_1 vw & \gamma v \end{pmatrix}$$

$$\frac{\partial \overrightarrow{\boldsymbol{\mathcal{F}}}_3(\boldsymbol{U})}{\partial \boldsymbol{U}} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ -uw & w & 0 & u & 0 \\ -vw & 0 & w & v & 0 \\ \gamma_1 \mathscr{E}_c - w^2 & -\gamma_1 u & -\gamma_1 v & (2-\gamma_1)\,w & \gamma_1 \\ w\,(\gamma_1 \mathscr{E}_c - \mathcal{H}) & -\gamma_1 uw & -\gamma_1 vw & -\gamma_1 w^2 + \mathcal{H} & \gamma w \end{pmatrix}$$

1.2 Valeurs et vecteurs propres

On désigne par $c = \sqrt{\frac{\gamma p}{\rho}}$ la vitesse du son. La matrice $\boldsymbol{K}$ est diagonalisable donc $\boldsymbol{K} = \mathcal{R}\boldsymbol{\Lambda}\mathcal{L}$ avec :

$$\boldsymbol{\Lambda} = \mathrm{diag}(\vec{\boldsymbol{u}}.\vec{\boldsymbol{n}} - c, \vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}, \vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}, \vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}, \vec{\boldsymbol{u}}.\vec{\boldsymbol{n}} + c)$$

$$\mathcal{R} = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ u - c.n_x & u & n_y & n_z & u + c.n_x \\ v - c.n_y & v & -n_x & 0 & v + c.n_y \\ w - c.n_z & w & 0 & -n_x & w + c.n_z \\ \mathcal{H} - \vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}.c & \mathscr{E}_c & u.n_y - v.n_x & u.n_z - w.n_x & \mathcal{H} + \vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}.c \end{pmatrix}$$

$$\mathcal{L} = \begin{pmatrix} \frac{1}{2c}\left(\frac{\gamma_1}{c}\mathscr{E}_c + \vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}\right) & -\frac{1}{2c}\left(\frac{\gamma_1}{c}u + n_x\right) & -\frac{1}{2c}\left(\frac{\gamma_1}{c}v + n_y\right) & -\frac{1}{2c}\left(\frac{\gamma_1}{c}w + n_z\right) & \frac{\gamma_1}{2c^2} \\ 1 - \frac{\gamma_1}{c^2}\mathscr{E}_c & \frac{\gamma_1}{c^2}u & \frac{\gamma_1}{c^2}v & \frac{\gamma_1}{c^2}w & -\frac{\gamma_1}{c^2} \\ v.n_x - u.n_y & n_y & -n_x & 0 & 0 \\ w.n_x - u.n_z & n_z & 0 & -n_x & 0 \\ \frac{1}{2c}\left(\frac{\gamma_1}{c}\mathscr{E}_c - \vec{\boldsymbol{u}}.\vec{\boldsymbol{n}}\right) & -\frac{1}{2c}\left(\frac{\gamma_1}{c}u - n_x\right) & -\frac{1}{2c}\left(\frac{\gamma_1}{c}v - n_y\right) & -\frac{1}{2c}\left(\frac{\gamma_1}{c}w - n_z\right) & \frac{\gamma_1}{2c^2} \end{pmatrix}$$

Annexe 1

Annexe 2

2 Numérotation locale des DoFs'un élément

Nous allons décrire ici la numérotation des degrés de liberté d'ordre élevé dans la méthode des résidus distribués.

On se réfèrera pour les explications qui suivent à l'exemple de numérotation fourni figure A2.2.

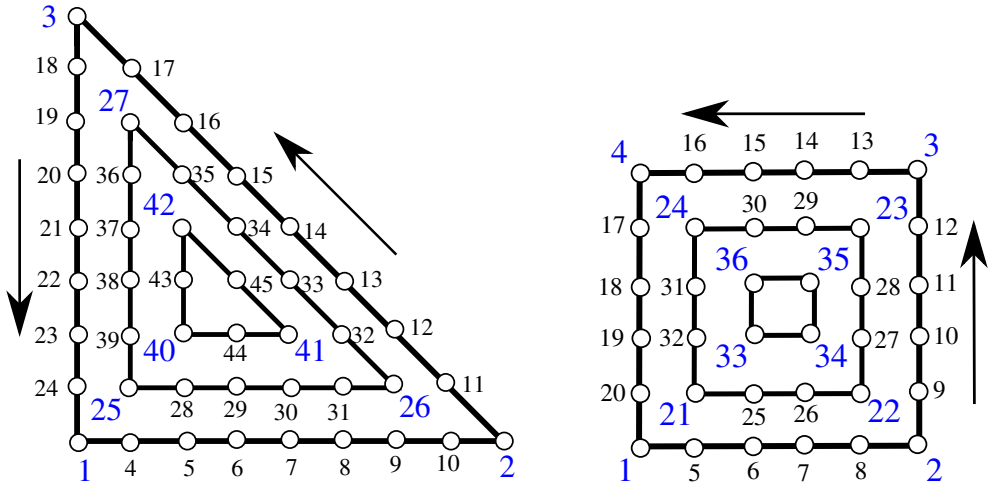


Figure A2.2 – Numérotation locale dans l'espace paramétrique :
à gauche, triangle d'ordre 8. À droite, quadrangle d'ordre 6.

Dans un schéma $\mathcal{RD}$, les sommets des éléments, les points des arêtes et les points internes aux éléments jouent des rôles différents, ils devront donc être distingués. C'est

pourquoi nous listerons en premier lieu les sommets de l'élément, puis les degrés de liberté situés sur ses faces, puis, pour finir, nous numéroterons les degrés de liberté internes aux éléments.

Du point de vue de notre schéma, les degrés de liberté internes jouent tous un rôle identique, nous pouvons donc les ordonner comme nous le souhaitons. Nous avons choisi de les organiser comme s'organisent les degrés de liberté externes d'un élément d'ordre plus faible. Ceci nous permet de donner une forme récursive à notre numérotation. Nous présenterons ici la numérotation utilisée en dimension 2. À l'ordre k un triangle contient $\frac{k(k+1)}{2}$ degrés de liberté formant $n_i = \lfloor \frac{k+1}{3} \rfloor$ triangles, d'ordres de plus en plus faibles, inclus les uns dans les autres. Un quadrangle contient, quant à lui, k^2 degrés de liberté formant $n_i = \lfloor \frac{k}{2} \rfloor$ quadrangles, d'ordres de plus en plus faibles, inclus les uns dans les autres. On parcourt successivement, de l'extérieur vers l'intérieur, les sous-éléments inclus et on les numérote, dans le sens de parcourt, de 1 à n_i. On ne s'intéressera qu'aux DoFs externes de nos inclusions. La première inclusion est donc notre élément complet et on ne considèrera que les sommets de cet élément et les DoFs de ses arêtes. Si notre élément est triangulaire, le $i^{\text{ème}}$ triangle inclus est d'ordre $k_{loc} = (k - 3i)$. Dans le cas d'un quadrangle, la $i^{\text{ème}}$ inclusion est d'ordre $k_{loc} = (k - 2i)$. Dans les deux cas, chacune des arêtes du $i^{\text{ème}}$ élément inclus comptera donc $(k_{loc} - 2)$ DoFs. Ainsi, nos DoFs sont numérotés en parcourant les inclusions, de la première à la $n_i^{\text{ème}}$, et, pour chaque inclusion, on numérote les n_s sommets de l'inclusion (où $n_s = 3$ pour un triangle et 4 pour un quadrangle) puis on parcourt dans le sens direct les n_a arêtes de l'inclusion (en dimension 2, $n_a = n_s$) et on numérote, dans le sens de parcourt, les $n_a\,(k_{loc} - 2)$ DoFs des arêtes.

Annexe 3

3 Format de fichier d'un maillage $\mathcal{NURBS}$

L'utilisation des maillages courbes, en particulier des maillages $\mathcal{NURBS}$, nécessite la lecture de toute les informations nécessaires à la connaissance des éléments courbes composant le maillage. Il nous a donc fallu définir un format de maillage adapté à nos besoins. Comme nous l'avons expliqué dans la section 2 de cette thèse, un élément $\mathcal{NURBS}$ (qu'il soit linéique, surfacique ou volumique) est totalement défini par le triplet *(ordre, points de contrôle, poids)*. Notre maillage devra donc contenir, l'ordre de travail (que l'on note k), la liste des points de contrôle du maillage et des poids associés, ainsi que, pour chaque élément du maillage (pour les éléments internes comme pour les éléments de bord), la liste des numéros des points de contrôle du polygone de contrôle de l'élément.
Nous avons comme format de fichier de base le format mesh. L'intérêt de ce format est que les outils qui lui sont associé (visualiseur, remailleur...) ont été développés pour ne reconnaitre que les champs qui lui sont connus et pour ignorer les autres. Chaque champ est identifié par un mot clé suivi du nombre d'entités du champ. Ceci nous permet donc d'ajouter de champs associés à de nouveaux mots clés et adaptés à nos besoins.
La page suivante résume le format employé, les mots clés sont écrits en bleu, les éventuelles explications sont écrites à droite. Dans l'exemple fourni, par soucis de place, nous avons supposé le maillage entièrement composé de triangles mais, évidemment, le maillage peut contenir des quadrangles (on aura alors le nouveau champ *PkQuadrangles*), des tétraèdres en dimension 3 (associés au champ *PkTetrahedra*), *etc.*

```
MeshVersionFormatted 2
Dimension
d

Order
k

NumberP1Dofs
np

Vertices
 nDoFs
 x1          y1          (z1)          ref1
 ⋮           ⋮           ⋮             ⋮
 xnDoFs      ynDoFs      (znDoFs)      refnDoFs

Weights
nDoFs
w1
⋮
wnDoFs

Triangles
nt
 v1^1   v1^2   v1^3   ref1
 ⋮      ⋮      ⋮      ⋮
 vnt^1  vnt^2  vnt^3  refnt

PkTriangles
nt
 v1^1   v1^2   ⋯   v1^{k(k+1)/2}   ref1
 ⋮      ⋮          ⋮              ⋮
 vnt^1  vnt^2  ⋯   vnt^{k(k+1)/2}  refnt

Edges
ne
 e1^1   e1^2   ref1
 ⋮      ⋮      ⋮
 ene^1  ene^2  refne

PkEdges
ne
 e1^1   e1^2   ⋯   e1^k   ref1
 ⋮      ⋮          ⋮     ⋮
 ene^1  ene^2  ⋯   ene^k  refne

End
```

Entête du format classique

Nouveau champ : ordre du maillage

Nouveau champ : Nombre de sommets (points $P1$)

Nombre total de points de contrôle
Liste des points de contrôle

Nouveau champ : Nombre de poids= n_{DoFs}
Liste des poids dans le même ordre que sont listés les DoFs

Nombre de triangles
Sommets des triangles

Nouveau champ : Nombre de triangles
DoFs des triangles dans l'ordre de la numérotation locale (figure A2.2)

Nombre d'arêtes
Extrémités des arêtes

Nouveau champ : Nombre d'arêtes
DoFs des arêtes dans l'ordre de la numérotation locale

Références bibliographiques

[1] I. Ergatoudis, B.M. Irons, and O.C. Zienkiewicz. *Curved, isoparametric, "quadrilateral" elements for finite element analysis.* International Journal of Solids and Structures **4**(1), 31–42 (1968). xix

[2] R.S. Barsoum. *On the use of isoparametric finite elements in linear fracture mechanics.* International Journal for Numerical Methods in Engineering **10**(1), 25–37 (1976). xix

[3] C.F. Shih, H.G. Lorenzi, and M.D. German. *Crack extension modeling with singular quadratic isoparametric elements.* International Journal of Fracture **12**(4), 647–651 (1976). xix

[4] G. Bfer. *An isoparametric joint/interface element for finite element analysis.* International journal for numerical methods in engineering **21**(4), 585–600 (1985). xix

[5] F. Bassi and S. Rebay. *High-order accurate discontinuous finite element solution of the 2d euler equations 1.* Journal of Computational Physics **138**(2), 251–285 (1997). xix, 152

[6] L. Krivodonova and M. Berger. *High-order accurate implementation of solid wall boundary conditions in curved geometries.* Journal of computational physics **211**(2), 492–512 (2006). xix, 151

[7] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs. *Isogeometric analysis : Cad, finite elements, nurbs, exact geometry and mesh refinement.* Computer methods in applied mechanics and engineering **194**(39-41), 4135–4195 (2005). xix

[8] L.A. Piegl and W. Tiller. *The NURBS book.* Springer Verlag (1997). xx

[9] JA Cottrell, A. Reali, Y. Bazilevs, and TJR Hughes. *Isogeometric analysis of structural vibrations.* Computer methods in applied mechanics and engineering **195**(41), 5257–5296 (2006). xx

[10] Y. Bazilevs, L.B. Da Veiga, J.A. Cottrell, T.J.R. Hughes, and G. Sangalli. *Isogeometric analysis : approximation, stability and error estimates for h-refined meshes.* Mathematical Models and Methods in Applied Sciences **16**(7), 1031 (2006). xx

[11] J.A. Cottrell, T.J.R. Hughes, and Y. Bazilevs. *Isogeometric analysis : toward integration of CAD and FEA.* John Wiley & Sons Inc (2009). xx

[12] A. Buffa, G. Sangalli, and R. Vázquez. *Isogeometric analysis in electromagnetics : B-splines approximation.* Computer Methods in Applied Mechanics and Engineering **199**(17-20), 1143–1152 (2010). xx

[13] F. Auricchio, L.B. da Veiga, A. Buffa, C. Lovadina, A. Reali, and G. Sangalli. *A fully "locking-free" isogeometric approach for plane linear elasticity problems : a stream function formulation.* Computer methods in applied mechanics and engineering **197**(1-4), 160–172 (2007). xx

[14] A. Buffa, C. de Falco, and G. Sangalli. *Isogeometric analysis : stable elements for the 2d stokes equation.* International Journal for Numerical Methods in Fluids **65**(11-12), 1407–1422 (2011). xx

[15] R. Sevilla. *NURBS-enhanced finite element method (NEFEM).* Thèse de Doctorat, Universitat Politècnica de Catalunya July (2009). xx, 178

[16] M. Vymazal, T. Quintino, N. Villedieu, and H. Deconinck. *High-order upwind residual distribution schemes on isoparametric curved elements.* Journal of Computational Physics **230**(4), 890–906 (2011). xx

[17] J.A. Talbert and A.R. Parkinson. *Development of an automatic, two-dimensional finite element mesh generator using quadrilateral elements and bezier curve boundary definition.* International Journal for Numerical Methods in Engineering **29**(7), 1551–1567 (1990). xx

[18] P.L. George and H. Borouchaki. *Construction of tetrahedral meshes of degree two.* International Journal for Numerical Methods in Engineering (2012). xx, 85

[19] R. Verfürth. *A review of a posteriori error estimation and adaptive mesh-refinement techniques. Vol. 1.* Wiley-Teubner New York, Stuttgart (1996). xxi

[20] Frédéric Alauzet and Pascal J. Frey. Estimateur d'erreur géométrique et métriques anisotropes pour l'adaptation de maillage. Partie I : aspects théoriques. Rapport de recherche RR-4759 INRIA (2003). xxi, 102, 104

[21] F. Hecht. *Mesh generation and error indicator.* Summer school : More efficiency in finite element methods (2008). http://math.univ-lille1.fr/~creuse/FORMATION_ESTIMATEURS/Transparents_Intervenants/Hecht-Ecole-Valencienne-2008.pdf. xxi, 179

[22] A. Loseille. *Adaptation de maillage anisotrope 3D multi-échelles et ciblée à une fonctionnelle pour la mécanique des fluides. Application à la prédiction haute-fidélité du bang sonique.* Thèse de Doctorat, Université Pierre et Marie Curie, Paris VI (2008). xxi, 179

[23] J.A. Cottrell, T.J.R. Hughes, and A. Reali. *Studies of refinement and continuity in isogeometric structural analysis.* Computer methods in applied mechanics and engineering **196**(41-44), 4160–4183 (2007). xxi

[24] M.R. Dörfel, B. Jüttler, and B. Simeon. *Adaptive isogeometric analysis by local h-refinement with t-splines.* Computer methods in applied mechanics and engineering **199**(5-8), 264–275 (2010). xxi

[25] J.F. Remacle, X. Li, N. Chevaugeon, and M.S. Shephard. *Transient mesh adaptation using conforming and non conforming mesh modifications.* 11h International Meshing Roundtable pages 261–271 (2002). xxi

[26] M. Dindar, M.S. Shephard, J.E. Flaherty, and K. Jansen. *Adaptive cfd analysis for rotorcraft aerodynamics.* Computer methods in applied mechanics and engineering **189**(4), 1055–1076 (2000). xxi

[27] G. Greiner. *Geometric Modelling.* Notes de cours. (2001). http://atrey.karlin.mff.cuni.cz/projekty/vrr/doc/grafika/geometricmodelling.pdf. xxii, 98

[28] J. Gallier. *On the efficiency of strategies for subdividing polynomial triangular surface patches.* Arxiv preprint cs/0606061 (2006). xxii, 118

[29] G.P. Galdi. *An Introduction to the Mathematical Theory of the Navier-Stokes Equations.* Springer Verlag (2009). 1

[30] R. Temam and A. Chorin. *Navier stokes equations : Theory and numerical analysis.* Journal of Applied Mechanics **45**, 456 (1978). 1

[31] J. Besson, G. Cailletaud, and S. Forest. *Lois de comportement non linéaires des matériaux.* Notes de cours. http://www2.mat.ensmp.fr/Pages/sf/teaching/masterecomadis.pdf. 4

[32] R.J. LeVeque. *Numerical Methods for Conservation Laws.* Birkhäuser Verlag AG (1994). 6

[33] P.D. Lax. *Weak solutions of nonlinear hyperbolic equations and their numerical computation.* Selected Papers Volume I pages 198–232 (2005). 6

[34] E. Godlewski and P.A. Raviart. *Numerical approximation of hyperbolic systems of conservation laws.* Springer (1996). 6

[35] P. Bénilan and S. Kružkov. *Conservation laws with continuous flux functions.* NoDEA : Nonlinear Differential Equations and Applications **3**(4), 395–419 (1996). 9

[36] L. Adam. *Conception et analyse de schémas d'ordre très élevé distribuant le résidu : application à la mécanique des fluides.* Thèse de Doctorat, INRIA Bordeaux Sud-Ouest (2009). 9, 14, 15, 27, 143

[37] M. Ricchiuto. *Construction and Analysis of Compact Residual Discretizations for Consevation Laws on Unstructured Meshes.* Thèse de Doctorat, Von Karman Institute for Fluid Dynamics (2005). 9

[38] M. Ricchiuto. *Contributions to the development of residual discretizations for hyperbolic conservation laws with application to shallow water flows.* Habilitation à diriger des recherches, INRIA Bordeaux Sud-Ouest (2011). 9

[39] R. Abgrall. *Essentially non-oscillatory residual distribution schemes for hyperbolic problems.* Journal of Computational Physics **214**(2), 773–808 (2006). 9, 15

[40] R. Abgrall, A. Larat, M. Ricchiuto, and C. Tave. *A simple construction of very high order non-oscillatory compact schemes on unstructured meshes.* Computers and Fluids **38**(7), 1314–1323 (2009). 9

[41] R. Abgrall, A. Larat, M. Ricchiuto, and C. Tave. *A simple construction of very high order non-oscillatory compact schemes on unstructured meshes.* Computers & Fluids **38**(7), 1314–1323 AUG (2009). 9, 70

[42] P.L. Roe. *Fluctuations and signals, a framework for numerical evolution problems.* Numerical Methods for Fluid Dynamics **11** (1982). 9

[43] R. Abgrall and P.L. Roe. *High order fluctuation schemes on triangular meshes.* Journal of Scientific Computing **19**(1), 3–36 (2003). 13, 18

[44] R. Abgrall and P.L. Roe. *High order fluctuation schemes on triangular meshes.* J. Sci. Comput. **19**, 3–36 December (2003). 16, 20

[45] A. Harten, J.M. Hyman, and P.D. Lax. *On finite-difference approximations and entropy conditions for shocks.* Communications on Pure and Applied Mathematics **29**(3), 297–322 (1976). 18

[46] P. Wesseling. *Principles of computational fluid dynamics. Vol. 29.* Springer Verlag (2010). 18

[47] R. Struijs. Flux vector distribution, (2010). http://math.utexas.edu/mp_arc/c/10/10-5.pdf. 23

[48] R. Abgrall and D. De Santis. High order residual distribution scheme for Navier-Stokes equations. In AIAA, editor, *20th AIAA Computational Fluid Dynamics Conference*, Honolulu, Hawaii, United States June (2011). AIAA, AIAA. 29

[49] M.G. Cox. *The numerical evaluation of b-splines*. IMA Journal of Applied Mathematics **10**(2), 134 (1972). 39

[50] C de Boor. *On calculating with b-splines*. Journal of Approximation Theory **6**(1), 50–62 (1972). 39

[51] G.E. Farin. *Curves and Surfaces for Computer-Aided Geometric Design : A Practical Code*. Academic Press, Inc. (1996). 45

[52] P.G. Ciarlet and P.A. Raviart. *Interpolation theory over curved elements, with applications to finite element methods*. Computer Methods in Applied Mechanics and Engineering **1**(2), 217–249 (1972). 51

[53] D. F. Rogers. *An introduction to NURBS : with historical perspective*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001). 57

[54] Conception assistée par ordinateur. http://fr.wikipedia.org/w/index.php?title=Conception_assistée_par_ordinateur&oldid=78304001. 77

[55] P.J. Frey and H. Borouchaki. *Geometric evaluation of finite element surface meshes*. Finite elements in analysis and design **31**(1), 33–53 (1998). 80

[56] F. Orgeret. *Sur l'approximation discrète des courbures des courbes planes et des surfaces lisses de l'espace euclidien de dimension 3*. Thèse de Doctorat, Université de Lyon (2007). 80

[57] P. Frey and F. Hecht. *Cours Maillage 2D, surfacique et 3D*. Notes de cours. (2005). http://www.ljll.math.upmc.fr/~hecht/ftp/old/IST/NoteCours-v1.pdf. 89

[58] C. Dobrzynski. *Adaptation de maillage anisotrope 3d et applicationa l'aérothermique des bâtiments*. These UPMC (2005). 99

[59] F. Pellegrini. *Distillating knowledge about scotch*. Combinatorial Scientific Computing (2009). 144

[60] S. Bensmina, C. Tenaud, Y. Fraigneau, and V. Daru. Une méthode de frontière immergée pour la simulation d'écoulements visqueux compressibles. In *proceedings of 19ème Congrés Français De Mécanique [CFM2009]*. AFM, Maison de la Mécanique, 39/41 rue Louis Blanc-92400 Courbevoie (2011). 175

[61] J.J.W. Vegt, H. Ven, HA Mang, and FG Rammenstorfer. Slip flow boundary conditions in discontinuous galerkin discretizations of the euler equations of gas dynamics. In *proceedings of the fifth World Congress on Computational Mechanics, Vienna-Austria*, NLR-TP, pages 1–16. National Aerospace Laboratory, NLR (2002). 178

[62] H.P. Seidel. *Symmetric recursive algorithms for surfaces : B-patchs and the de boor algorithm for polynomials over triangles.* Constructive Approximation **7**(1), 257–279 (1991). 179

[63] W. Dahmen, C.A. Micchelli, and H.P. Seidel. *Blossoming begets ß-spline bases built better by ß-patches.* Mathematics of computation **59**(199), 97–115 (1992). 179

[64] H.P. Seidel. *An introduction to polar forms.* Computer Graphics and Applications, IEEE **13**(1), 38–46 (1993). 179