



**HAL**  
open science

# Raffinement adaptatif de maillages intersectants, en Volumes Finis d'ordre élevé, pour l'aéropropulsion

Alexandre Limare

► **To cite this version:**

Alexandre Limare. Raffinement adaptatif de maillages intersectants, en Volumes Finis d'ordre élevé, pour l'aéropropulsion. Matériaux et structures en mécanique [physics.class-ph]. Université de Technologie de Troyes, 2017. Français. NNT : 2017TROY0028 . tel-02949471

**HAL Id: tel-02949471**

**<https://theses.hal.science/tel-02949471>**

Submitted on 25 Sep 2020

**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  
de doctorat  
de l'UTT

**Alexandre LIMARE**

**Raffinement adaptatif  
de maillages intersectants,  
en Volumes Finis d'ordre élevé,  
pour l'aéropropulsion**

**Spécialité :**

**Matériaux, Mécanique, Optique et Nanotechnologie**

2017TROY0028

Année 2017



---

---

# THESE

*pour l'obtention du grade de*

## DOCTEUR de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

**Spécialité : MATERIAUX, MECANIQUE, OPTIQUE ET NANOTECHNOLOGIE**

*présentée et soutenue par*

**Alexandre LIMARE**

*le 27 octobre 2017*

---

---

**Raffinement adaptatif de maillages intersectants,  
en Volumes Finis d'ordre élevé, pour l'aéropropulsion**

---

---

### JURY

M. F. HECHT	PROFESSEUR DES UNIVERSITES	Président
M. H. BOROUCAKI	PROFESSEUR DES UNIVERSITES	Directeur de thèse
M. P. BRENNER	DOCTEUR, EXPERT CFD	Examineur
M. P.-L. GEORGE	DIRECTEUR DE RECHERCHE INRIA	Examineur
M. J.-C. JOUHAUD	CHERCHEUR CERFACS- HDR	Rapporteur
M. B. MOHAMMADI	PROFESSEUR DES UNIVERSITES	Examineur
Mme S. PÉRON	INGENIEUR DE RECHERCHE ONERA	Examineur
M. S. POPINET	DIRECTEUR DE RECHERCHE CNRS	Rapporteur

# Remerciements

Un page s'est tournée, un moment fort en émotions, avec des bas très bas et des hauts... très hauts! Beaucoup de code, beaucoup de bugs, quatre contrats, trois colloocations, deux hélicons –pour bien se vider la tête et les poumons–, pas mal de DIPA et de porters, j'ai mis tout ça à mariner pendant 4 ans et par le miracle de la fermentation, un manuscrit et une petite choucroute en sont sortis.

Ce projet est né de la volonté de Jean-François, Bastien et bien évidemment de Pierre (et parfois de sweetie) d'anticiper les besoins des utilisateurs du code de CFD FLUSEPA. Je leur suis grandement reconnaissant d'avoir bien voulu me faire confiance et espère avoir été à la hauteur de leurs attentes –merci à Brice aussi, si tu me lis un jour. Le monde industriel est riche d'expériences et j'ai pris un vrai plaisir à travailler avec des gens comme Thomas, David, Mathieu, Lorris, Greg, Hélène, Jean-Marie, Jean-Marc, Sophie et Clément, Mohamed... Et mon premier bêta-testeur Fabien pour lequel chaque bon jour plein de bugs était un jour qui valait la peine d'être vécu!

Je remercie M. Popinet et M. Jouhaud de bien avoir voulu rapporter mes travaux et des échanges fructueux qui s'en sont suivis, même si je n'avais pas réellement à les convaincre de l'intérêt de l'AMR dans le milieu de la CFD. La soutenance était symboliquement quelque chose de très fort pour moi. Merci à M. Hecht d'avoir bien voulu présider le jury, merci à M. George et Stéphanie Péron pour leur présence. Merci à tous ceux qui ont pu faire le déplacement, ma famille et mes amis. Merci à Lolo, qui n'a de cesse de me surprendre quand il s'agit d'organiser un banquet!

Comment ne pas remercier le SPB avec ces drilles qui ont rythmé de jouages et de manches ma vie et celles de mes proches; Alain, Huafa, Mathilde, Popeye, Anneuh, Victor, Mathieu, Bertrand, et bien d'autres. Ce socle qui m'a permis de mettre en perspective tout ce que je faisais et d'aller toujours plus loin parce que la musique est bien plus que du bruit et des silences. Je pense aussi chaleureusement aux Rillettes.

Je remercie Houman pour ses conseils avisés, ses remarques pertinentes et son soutien tout au long de cette thèse. Je suis reconnaissant envers Pierre de m'avoir laissé carte blanche pour un sujet qui le tenait tant à cœur depuis si longtemps et pour sa compréhension quand ma santé me faisait défaut. Je le remercie aussi pour cette relecture éreintante et tout le temps qu'il m'a consacré. Merci à Lolo et Jacqueline pour leur relecture et leur soutien sans faille!

Je pense à Peffault, à Saint Louis, à toute l'équipe de trèfle 3 que j'ai pu côtoyer et qui risque de m'accompagner pendant encore bien des années.

La liste est encore longue et la digression facile, aussi je me dois d'abréger, merci à l'Asso'coupe, merci à mon Pichon qui m'a récupéré plus d'une fois en morceaux, merci à Estelle aussi d'avoir été là quand il le fallait, je suis heureux de t'avoir vu mûrir durant ces 2 ans. Merci à ma famille, j'aurais tant aimé que Jean et Michou soient encore là. Merci à ma choucroute qui a toujours gravité autour de ce maelström nébuleux, ce capharnaüm et qui m'a prouvé qu'ensemble, on pouvait faire de grandes choses.



« Certains veulent aller plus vite, plus vite qu'autrefois, mais pas trop, et surtout pas n'importe comment : pour beaucoup le galop de la malle-poste paraît largement suffisant, il semble inutile de prendre des risques, d'engager des crédits pour un nouveau moyen de transport, techniquement inachevé. L'arrivée d'un intrus n'est pas accueillie d'emblée comme bénéfique. L'invasion du rail, à l'échelle des rythmes séculaires, ne s'impose pas sans réticence. L'inertie locale au développement des chemins de fer traduit le refus de se laisser ferrer. La vapeur bouscule sans aménité l'ancien mode de vie, perturbe une tranquillité, des jalons stables et sécurisants. Elle contrarie également des habitudes économiques. Il est facile aujourd'hui de critiquer ces réticences, de sourire des incompréhensions, des inquiétudes, des angoisses. La visée rétrospective ici n'est pas la méthode la plus appropriée. Mieux vaut juger l'accélération avec le regard de l'homme de 1830, comment voit-il cette vitesse inédite ? Les thèses traditionnelles sur les résistances à l'introduction et au développement des chemins de fer s'organisent autour de deux thèmes : la peur et les intérêts économiques. »

Christophe Studeny, *L'invention de la vitesse : France, XVIII-XXe siècle.*



# Table des matières

<b>Introduction</b>	<b>1</b>
<b>1 Les méthodes de raffinement de maillages adaptatif</b>	<b>7</b>
1.1 Les différents types de méthodes AMR	8
1.2 Schéma explicite d'avancement en temps	11
1.2.1 La méthode adaptative d'avancement en temps de Berger et Colela	11
1.2.2 Efficacité de l'intégration explicite avec pas de temps par grilles	13
1.2.3 Hybridation de schéma explicite-implicite	14
1.3 Répartition vers une sous-maille	14
1.4 Les critères d'adaptation de maillages	15
1.5 Les méthodes AMR avec maillages chevauchants	16
1.6 Conclusion	17
<b>2 Stratégie numérique</b>	<b>19</b>
2.1 Cadre	19
2.1.1 Les équations de Navier-Stokes	20
2.1.2 La notion de séparation d'échelle	20
2.1.3 La modélisation Reynolds Averaged Navier-Stokes (RANS)	21
2.1.4 La simulation aux grandes échelles (LES)	22
2.1.5 Les modèles Hybrides RANS/LES (HRL)	23
2.2 Les Volumes Finis	24
2.2.1 Propriétés géométriques et description des éléments hexaédriques manipulés	25
2.2.2 Intégration temporelle adaptative	30
2.3 Le chevauchement de maillage	32
2.3.1 Rappels sur les méthodes de chevauchement de maillage	32
2.3.2 La stratégie de chevauchement classique <i>chimère</i>	33
2.3.3 Techniques palliant la non-conservativité des méthodes de chevauchement de maillages classiques	36
2.3.4 La méthode d'intersection de maillage de Brenner	38
2.4 Architectures des versions 6 et 7 de FLUSEPA	43
2.4.1 Structure de données	43
2.4.2 Version 6, parallélisation pour des machines à mémoire partagée	43
2.4.3 Version 7, parallélisation pour des machines à mémoire partagée et distribuée	44
2.5 Conclusion	46
<b>3 Méthode AMR <i>octree</i> par cellule implantée</b>	<b>47</b>
3.1 Architecture logicielle du module d'AMR pour une exécution en MD	47
3.1.1 Schéma d'adaptation de maillages	48
3.1.2 Attribution de numéro d'esclave et équilibrage de la charge	49
3.2 Processus maître	51
3.2.1 Algorithme de modification de maillage	51



3.2.2	Subdivision d'une maille . . . . .	52
3.2.3	Création des nœuds sur les faces lors d'un raffinement . . . . .	52
3.2.4	Structure de données . . . . .	53
3.2.5	Équilibrage . . . . .	54
3.2.6	Traitement des <i>hanging nodes</i> . . . . .	56
3.2.7	Polyèdre conforme par maille . . . . .	57
3.2.8	Création des coques . . . . .	59
3.2.9	Structure de données . . . . .	60
3.2.10	Accès mémoire aux sous-faces . . . . .	60
3.2.11	Espace mémoire occupé par la méthode AMR . . . . .	61
3.3	Processus aérodynamiques (esclaves) . . . . .	61
3.3.1	Structure de données des processus aérodynamiques . . . . .	61
3.3.2	Description algorithmique . . . . .	63
3.3.3	Boucles de parcours des faces pour les maillages non-conformes . . . . .	64
3.4	Conclusion . . . . .	66
<b>4</b>	<b>Redistribution conservative d'ordre élevé des variables</b>	<b>67</b>
4.1	Positionnement du problème . . . . .	67
4.2	Méthode . . . . .	68
4.2.1	Procédé de redistribution conservative de l'erreur . . . . .	70
4.3	Limiteurs associés pour une projection à l'ordre 3 . . . . .	71
4.3.1	Limitation passage bar-tilde sur la Pression . . . . .	71
4.3.2	Limitation sur le voisinage . . . . .	72
4.4	Cas tests de validation . . . . .	72
4.4.1	Tourbillon stationnaire . . . . .	73
4.4.2	Tourbillon en mouvement . . . . .	75
<b>5</b>	<b>Validation de la méthodologie AMR implantée dans FLUSEPA</b>	<b>77</b>
5.1	Marche montante . . . . .	77
5.1.1	Résultats et analyse . . . . .	78
5.2	Marche descendante . . . . .	79
5.2.1	Construction du maillage initial . . . . .	80
5.2.2	Cadre numérique et physique . . . . .	82
5.2.3	Résultats et analyse . . . . .	82
5.3	Simulation d'une onde de souffle . . . . .	87
<b>6</b>	<b>Conclusions et perspectives</b>	<b>93</b>
	<b>Bibliographie</b>	<b>97</b>
<b>A</b>	<b>Critères d'adaptation</b>	<b>105</b>
A.1	Critère seuil de gradient de pression . . . . .	105
A.2	Critère gradient de pression scalaire vitesse . . . . .	105
A.3	Critère basé sur le Hessien de la vitesse . . . . .	105
A.4	Critère basé sur la taille de Von Kármán . . . . .	105
<b>B</b>	<b>Module de compatibilité géométrique d'intersections</b>	<b>107</b>
B.1	Critère de compatibilité simple . . . . .	107
B.2	Compatibilité géométrique pour le passage de structures caractéristiques . . . . .	109

<b>C</b>	<b>Relation entre <math>\tilde{q}</math> et <math>\bar{q}</math></b>	<b>111</b>
C.1	Relation pour la vitesse . . . . .	111
C.2	Relation pour la pression . . . . .	112
C.3	Relation pour la température . . . . .	113



# Table des figures

1	Nombre de mises en orbite par année [STA] . . . . .	1
2	Concept de premier étage réutilisable par SpaceX (source : [Conb]) . . . . .	2
3	Concept de moteur réutilisable par Airbus Safran Launchers (source [Cona]) . . . . .	2
4	Zone pseudo non conforme obtenue après raffinement . . . . .	6
1.1	Simulation du tsunami de 2004 dans l’océan Indien utilisant une méthode d’adaptation <i>quadtree</i> [Pop11]. Colonne de gauche, hauteur de la vague, colonne de droite résolution du maillage . . . . .	7
1.2	Différentes méthodes AMR pour la création de grilles . . . . .	9
1.3	Exemple de grilles hiérarchisées sur 3 niveaux . . . . .	9
1.4	Création d’arbres pour des maillages non structurés (extrait de Kavouklis et Kallinderis [KK10]). Première ligne : raffinement d’un tétraèdre, deuxième ligne : raffinement d’un prisme, troisième ligne : raffinement d’une pyramide, quatrième ligne : raffinement d’un hexaèdre. . . . .	10
1.5	Création d’arbres pour des maillages non structurés [LH08, LH10] . . . . .	10
1.6	Deux grilles incrustées, en pointillés un front de cellules fantômes pour la grille fine . . . . .	11
1.7	Exemple d’utilisation classique de ghost-cells dans la cas d’une méthode explicite. En rouge le flux à l’interface fin-grossier pour la cellule $\Omega_k$ qui utilise la cellule virtuelle $\Omega_{J_i}$ . En noir : le flux calculé pour la cellule $\Omega_J$ . . . . .	12
1.8	Interpolation des gradients tangente aux interfaces inter-niveaux de raffinement [CMG <sup>+</sup> 13] . . . . .	13
1.9	Méthode de remaillage global. En trait pointillés le maillage initial, en traits pleins le maillage après une étape remaillage [LS05] . . . . .	14
1.10	Exemple de méthodes de projection de variables Lagrangienne, ici fusion de la cellule représentée dont le diagramme de Voronoï est dessiné en rouge [RDG17] avec les cellules bleue et verte . . . . .	15
1.11	Maillage <i>near/off body</i> initial après adaptation par <i>octree</i> permettant d’harmoniser la résolution des grilles dans la zone de chevauchement pour une simulation d’hélicoptère [Mea01] . . . . .	17
1.12	Maillage avant et après adaptation utilisés par OVERFLOW2 [HP09] . . . . .	17
2.1	Schéma descriptif des modules du programme FLUSEPA . . . . .	20
2.2	Validité de l’hypothèse d’ergodicité [W <sup>+</sup> 98] . . . . .	22
2.3	Cascade d’énergie turbulente . . . . .	24
2.4	Exemple de reconstructions par cellules et problèmes de Riemann associés. En noir : reconstruction constante, en rouge : reconstruction linéaire, en bleu : reconstruction parabolique. . . . .	25
2.5	Description d’un élément . . . . .	26
2.6	Exemple d’un assemblage de maillage de FLUSEPA . . . . .	33
2.7	Exemple de calculs de séparations effectués avec FLUSEPA . . . . .	33
2.8	Zone d’interpolation <i>chimère</i> dans un cas mono-dimensionnel . . . . .	34
2.9	Schéma de principe de la méthode <i>chimère</i> . . . . .	35
2.10	Exemple d’un chevauchement de deux maillage $\mathcal{M}_1$ et $\mathcal{M}_2$ avec pour chacun une paroi associée. En rouge : cellule qui vont être couvertes par mouvement de $\mathcal{M}_1$ . . . . .	36
2.11	Méthodologie DRAGON (extrait de Chan et Buning [CB95]) . . . . .	37
2.12	Intersection d’un bouddha et d’un lion [Lan17] . . . . .	37
2.13	Comparaison des résultats d’un calcul d’onde de souffle dans une ville (extrait de Alauzet [Ala16]) . . . . .	38

2.14	Chevauchement de deux maillages $\mathcal{M}_1$ et $\mathcal{M}_2$ . . . . .	38
2.15	Cas courant d’une intersection de deux maillages en 2D . . . . .	39
2.16	Exemple de couverture d’une cellule $\mathcal{C}_1$ en multichevauchement . . . . .	40
2.17	Dodécaèdre défini par cellule pour le calcul des intersections . . . . .	41
2.18	Trace polygonale d’une cellule . . . . .	41
2.19	Mise à jour des moments volumiques des cellules après intersection . . . . .	42
2.20	Assemblage de mailles . . . . .	43
2.21	Architecture de FLUSEPA V6 . . . . .	44
2.22	Architecture de FLUSEPA V7 . . . . .	44
2.23	Maillage coloré par numéro d’esclave au cours d’un calcul . . . . .	45
2.24	Index des cellules triées en segments avec les bornes associées . . . . .	46
3.1	Reconstruction du maillage initial après adaptation et répartition en processus esclaves du volume à calculer . . . . .	50
3.2	Raffinement d’une cellule . . . . .	52
3.3	Méthodes possibles pour le raffinement d’une face. En rouge : une découpe correcte, en vert : une découpe utilisant le centre de gravité de la face . . . . .	53
3.4	Forêt d’ <i>octrees</i> à deux arbres, en rouge : les éléments désactivés, en vert : les éléments actifs . . . . .	53
3.5	Exemple de <i>hanging node</i> . . . . .	54
3.6	En rouge <i>hanging nodes</i> double, mauvais équilibrage 2 – 1 . . . . .	55
3.7	Exemple d’une adaptation de maillage avec règle d’équilibrage 2 – 1 pour le nœud $\mathcal{N}_i$ . . . . .	56
3.8	Cas de maillages 2D et 3D non-conformes . . . . .	56
3.9	Arête stockée lors du raffinement d’une cellule. Numérotation locale . . . . .	57
3.10	Echantillon de la librairie de découpe des faces en fonction de la présence de nœuds sur les arêtes ou d’une maille raffinée . . . . .	58
3.11	Polyèdre conforme avec une maille voisine indirecte raffinée . . . . .	58
3.12	Polyèdre par maille après perturbation des nœuds . . . . .	59
3.13	Cas d’un raffinement d’une face qui est un élément constitutif d’un polyèdre d’intersection . . . . .	60
3.14	Voisinage de la cellule $\mathcal{T}_\alpha$ , en vert niveau de raffinement des cellules . . . . .	61
3.15	Schéma du cheminement logique d’un accès mémoire aux sous-faces d’une face divisée . . . . .	62
3.16	Schéma de principe de l’ajout d’un objet dans un tableau contenant des trous . . . . .	62
3.17	Raffinement d’une maille $\mathcal{T}_\alpha$ partiellement couverte par un polyèdre $\mathcal{P}$ , partie grisée . . . . .	64
3.18	Arbre de relation local d’un processus aérodynamique au cours d’une adaptation de maillage . . . . .	64
3.19	Cellule $\Omega_1$ intersectée par un maillage $\mathcal{M}_2$ de priorité supérieure, numérotation locale des faces d’intersection obtenues . . . . .	65
3.20	Exemple de la division de face et numérotation locale . . . . .	66
4.1	Maillage 1D avec fonction de référence et différentes méthodes de reconstruction selon leur ordre . . . . .	68
4.2	Maillage 1D avec fonction de référence. Redistribution d’ordre consistante avec l’ordre de la méthode de reconstruction des variables . . . . .	69
4.3	Partition en deux sous-ensembles $\mathcal{T}_{\alpha_1}$ et $\mathcal{T}_{\alpha_2}$ de la cellule $\mathcal{T}_\alpha$ . . . . .	69
4.4	Résultats tourbillon stationnaire . . . . .	73
4.5	Dérivée première de la vitesse, norme de froebenius . . . . .	74
4.6	Tracé de l’erreur en fonction du pas de discrétisation . . . . .	74
4.7	Champ de température initial et zoom sur la zone de chevauchement de maillage . . . . .	75
4.8	Erreur en fonction du pas de discrétisation pour le tourbillon en mouvement sans intersection . . . . .	75
4.9	Champ de la norme de Froebenius de la matrice hessienne de la vitesse, comparaison des résultats obtenus en présence ou non d’intersection, en fonction de l’ordre de précision de la redistribution des variables . . . . .	76
5.1	Schéma des structures caractéristiques de l’écoulement [Jou97] . . . . .	78
5.2	Comparaison calcul AMR, calcul raffiné sur deux niveaux . . . . .	79

5.3	Résultats obtenus après adaptation de maillage . . . . .	80
5.4	Visualisation des zones de décollement et de rattachement de la couche limite sur Ariane 5 . . . . .	80
5.5	Schéma de la géométrie pour la simulation de la marche descendante . . . . .	81
5.6	Physique de l'écoulement de la marche descendante. En noir : Descriptif de la physique de l'écoulement moyen, en rouge : zone de modélisation LES, en vert : zones RANS . . . . .	81
5.7	Maillage utilisé pour le calcul. En cyan : les grilles de paroi, en magenta : les grilles des fond, en vert : la grille AMR . . . . .	82
5.8	Comparaison entre le maillage fixe et le maillage AMR, coloration par la température . . . . .	83
5.9	Maillage coloré par la température au cours du calcul dans le plan $z = 0$ . . . . .	84
5.10	Champs statistiques . . . . .	84
5.11	Résultat AMR $\frac{\bar{U}}{U_\infty}$ en fonction de $y$ à $x = 0$ , comparaison avec les mesures expérimentales de [MLDB87] . . . . .	85
5.12	Résultat AMR $\frac{U_{\text{rms}}}{U_\infty}$ en fonction de $y$ à $x = 0$ , comparaison avec les mesures expérimentales de [MLDB87] . . . . .	86
5.13	Résultats $\frac{U_{\text{rms}}}{U_\infty}$ en fonction de $y$ à $x = 0$ de Pont [Pon15] pour les modèles HRL de FLUSEPA, comparés aux résultats expérimentaux de Moreau [MLDB87] et à l'étude ZDES de Deck [Dec02] . . . . .	86
5.14	Dispositif expérimental du banc martel . . . . .	87
5.15	Résultats d'une simulation d'onde de souffle par FLUSEPA . . . . .	88
5.16	Maillage initial. $1.10^5$ cellules . . . . .	89
5.17	Maillages au cours du calcul, coloriés par niveau de gradient de pression, avant réflexion de l'onde principale . . . . .	90
5.18	Maillages au cours du calcul, coloriés par niveau de gradient de pression, après réflexion de l'onde principale et retour sur la paroi du booster . . . . .	91
5.19	Signaux temporels de pression. Comparaison des calculs FLUSEPA avec et sans utilisation de l'AMR . . . . .	92
6.1	Exemple d'applications d'une méthode de bi-section . . . . .	95
B.1	Advection d'une structure tourbillonnaire. En noir : grille $\mathcal{G}_1$ , en magenta : grille $\mathcal{G}_2$ . . . . .	107
B.2	Maillage obtenu après application du critère de compatibilité géométrique simple. Grilles colorées par niveau de priorité . . . . .	108
B.3	Propagation de raffinement au niveau d'une intersection . . . . .	109



# Liste des tableaux

3.1	Dénombrement des polyèdres conformes possibles par maille connaissant le nombre de faces entre mailles de niveaux de raffinements différents . . . . .	59
4.1	Détail des calculs effectués . . . . .	73
4.2	Nombre de cellules par grille . . . . .	75
5.1	Données de calcul marche ascendante . . . . .	78
5.2	Comparaison de la résolution initiale du maillage avec le calcul de Pont <a href="#">[Pon15]</a> . . . . .	82
5.3	Comparaison des tailles de la bulle de recirculation . . . . .	85





# Introduction

Après avoir atteint son apogée en 1969 avec le premier pas de l'homme sur la Lune, l'industrie spatiale a perdu bien de son attrait depuis. La mise en orbite de la station MIR puis de l'ISS et plus récemment des missions telles que Rosetta ont pu faire renaître certaines vocations, mais c'est seulement depuis quelques années qu'un regain de vitalité se profile dans l'industrie aérospatiale, le nombre de lancements en augmentation depuis une décennie (*cf.* figure 1) témoigne de cette embellie. En effet, l'arrivée de nouveaux acteurs dynamiques ayant des objectifs ambitieux de baisse des coûts du kilogramme en orbite, a rebattu les cartes. Des compagnies telles que SpaceX, Blue Origin et Virgin Galactic sont apparues sur le marché des lanceurs et ont insufflé un nouvel élan. Ainsi, les concepts de fusée partiellement réutilisable (figures 2, 3), de micro-fusées ou de mission habitée vers Mars voient enfin le jour et amènent aujourd'hui des investisseurs publics et privés à placer à nouveau leurs espoirs dans cette industrie très particulière.

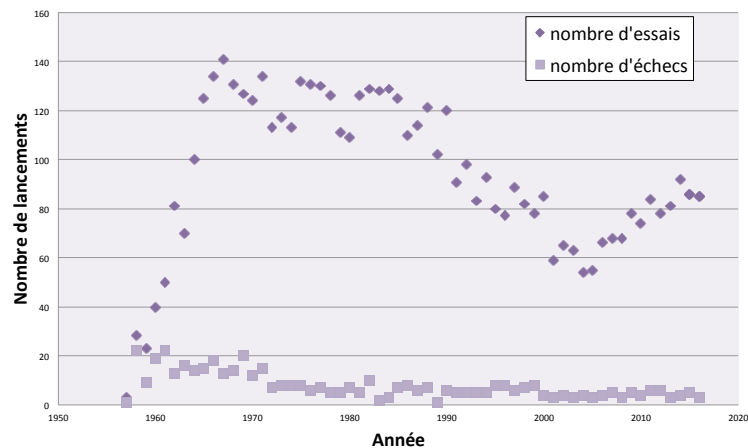


FIGURE 1 – Nombre de mises en orbite par année [STA]

La récupération par SpaceX de ses premiers étages et la réutilisation de ceux-ci en mars 2017 marque un réel tournant. Depuis plusieurs années, de nombreuses solutions ont été étudiées par les différents acteurs du spatial. SpaceX a montré la faisabilité de son concept et a souligné l'importance de fonctionner avec des cycles de développement en adéquation avec un marché en transformation rapide. Ainsi, les contraintes pour les industriels évoluent et les grandes avancées techniques nécessaires à la réalisation de projets aéronautiques novateurs nécessitent d'être clairement identifiées. En particulier, Slotnick *et al.* [SKA<sup>+</sup>14] ont effectué une revue des besoins dans le domaine de la simulation numérique et des recherches associées qui doivent être menées en priorité afin d'engendrer des ruptures technologiques d'ici à 2030. Selon cet article, un des axes majeur d'amélioration dans cette discipline concerne l'automatisation de la chaîne de conception : la création des géométries à étudier, la génération des maillages associés, l'adaptation dynamique des maillages une fois générés, la création de nombreux cas tests de validation, le développement d'outils pour l'extraction et la compréhension des énormes volumes de données produits, tout en étant sûr de pouvoir garder la main sur l'ensemble de ces processus lorsque l'automatisation défaille. Ces grandes voies de recherche permettront à l'ensemble des stratégies de simulation numérique de répondre aux défis à venir. Pour le monde de la recherche et de l'industrie, il sera indispensable

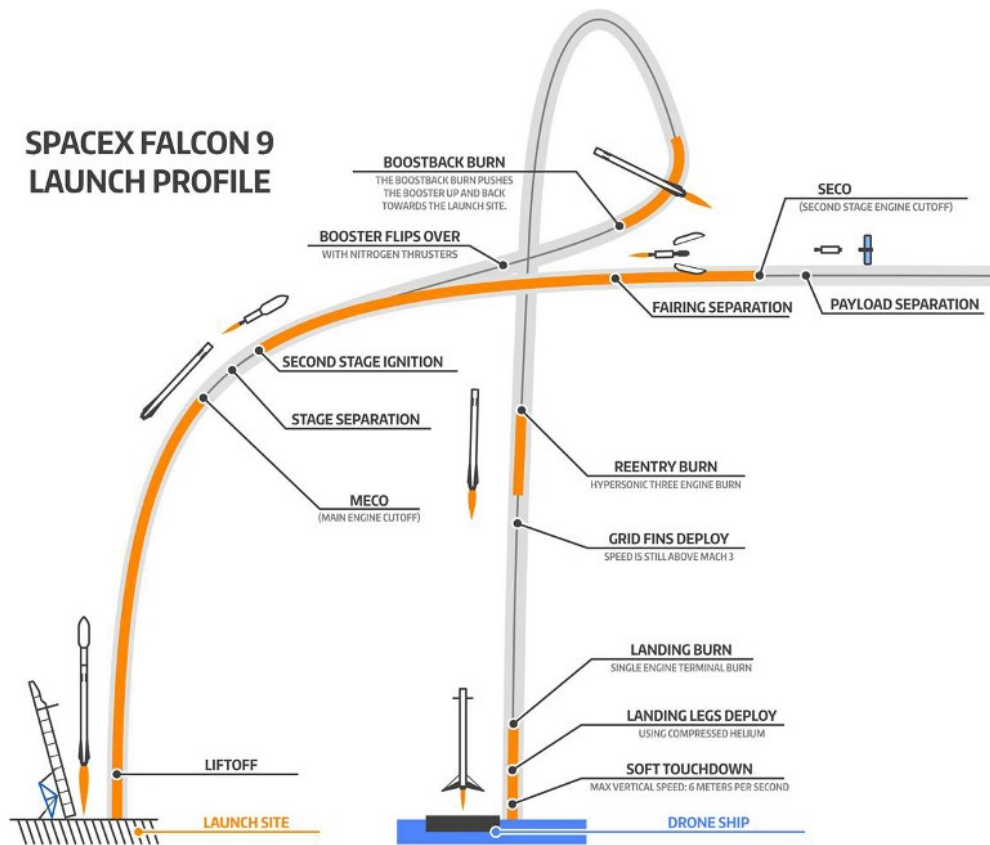


FIGURE 2 – Concept de premier étage réutilisable par SpaceX (source : [Conb])

d'assurer la cohérence de l'ensemble de cette chaîne de conception.

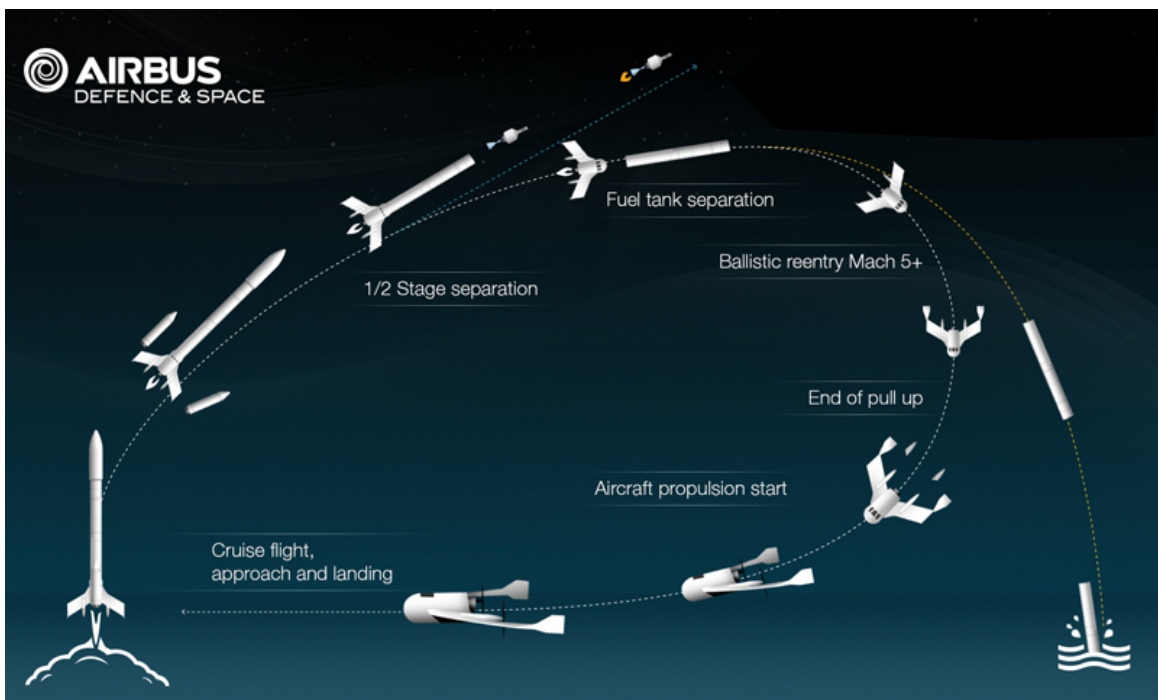


FIGURE 3 – Concept de moteur réutilisable par Airbus Safran Launchers (source [Cona])

La plupart des phénomènes étudiés, liés aux écoulements aéro-propulsifs qui concernent les lanceurs spatiaux, peuvent être modélisés par des systèmes d'équations aux dérivées partielles. Le caractère non linéaire de certains d'entre eux, conjugué à la complexité des géométries étudiées et aux mouvements relatifs des différents corps en présence (séparations d'étages) complexifie énormément leur approche par modélisation. En plus des propriétés énoncées précédemment, ces écoulements possèdent la particularité d'être fortement instationnaires et hautement énergétiques, ce qui implique qu'il est souvent extrêmement difficile de les reproduire précisément expérimentalement. En effet, on imagine difficilement l'allumage d'un moteur fusée dans une soufflerie, au risque d'endommager des installations qui ne sont pas conçues pour supporter de telles conditions, ne parlons pas alors d'effectuer une séparation d'étages à très faible pression. Étant donné que ce type de phase de vol est inhérent au concept même de lanceur-fusée<sup>1</sup>, des méthodes numériques ayant pour but de simuler ces écoulements ont été mises en place progressivement depuis les années 60 et ont permis de modéliser et de reproduire numériquement certaines de leurs propriétés. Cependant, le besoin vital d'accélérer la réactivité et d'augmenter la robustesse en conception, conjuguée à la chute du prix de l'informatique (*flops* et RAM *via* les machines multicœurs distribuées) conduit actuellement à l'explosion des simulations numériques. Conséquemment, plus les simulations effectuées progressent, plus elles se complexifient, notamment de deux façons :

1. la physique que l'on cherche à modéliser est plus complexe et nécessite d'ajuster les modèles utilisés pour qu'ils soient à la fois plus précis et conservent leur robustesse,
2. la mise en place et l'exécution d'un calcul deviennent beaucoup plus contraignantes étant donné la complexité des méthodes numériques requises, la puissance de calcul nécessaire et l'énorme volume de résultats générés à analyser.

D'un point de vue industriel, la question critique se résume à savoir quelle précision peut-on obtenir pour un coût donné et comment garantir cette précision. Pour cela, il est nécessaire d'avoir une idée des paramètres principaux et de leur influence sur les résultats. De même, les utilisateurs doivent être conscients des limites réelles des modèles développés et utilisés dans leur code. Par conséquent, une représentation mathématique correcte des phénomènes étudiés dont les caractéristiques principales sont à la fois leur violence et leur instationnarité, nécessite le choix de méthodes numériques adaptées. Ces méthodes doivent être les plus aptes à supporter des évolutions permettant une amélioration du niveau global de modélisation ou de précision<sup>2</sup>. Les écoulements rencontrés lors des différentes phases de vol d'un lanceur spatial sont caractérisés par leur haute enthalpie et sont traversés par des ondes de choc, dans des milieux souvent confinés (inter-étages de lanceurs) ou quasiment vides. Par conséquent, il est impératif de mettre en œuvre une formulation totalement conservative et les schémas numériques associés doivent absolument assurer la positivité des variables physiques (masse volumique, température, pression, énergie cinétique, fractions massiques, ...) malgré la raideur et l'intensité des phénomènes rencontrés (détentes abruptes, chocs extrêmement forts, cinétique chimique explosive, ...). Pour que le logiciel résultant soit viable, surtout dans le contexte aérospatial, il doit permettre de modéliser les géométries complexes, les corps en mouvement relatif, être suffisamment précis et peu dissipatifs pour pouvoir fournir des résultats pertinents. Enfin, il doit être performant d'un point de vue informatique pour utiliser au mieux les ressources matérielles disponibles. Toutes ces conditions sur l'ensemble des stratégies numériques, leur utilisation, leur développement et leur pérennisation nécessitent un grand degré d'expertise pour être satisfaites.

Les simulations numériques reposent sur des discrétisations de l'espace en volumes élémentaires appelées *maillages* que l'on peut diviser en deux grandes catégories :

- Les maillages structurés qui reposent sur une discrétisation de l'espace *via* une paramétrisation selon 3 coordonnées  $(i, j, k)$ . Ils forment des blocs constitués d'hexaèdres en 3D. On parle de maillages cartésiens lorsque chacune des directions principales du maillage est orthogonale aux deux autres. L'intérêt principal d'un maillage structuré réside dans le fait que la connectivité entre les éléments est implicite ce qui limite la taille mémoire nécessaire au stockage de la topologie. Cependant la génération et la manipulation de telles grilles sont adaptées uniquement aux géométries simples. Dans le cas d'une configuration industrielle, il est nécessaire de décomposer l'objet et l'espace à étudier en sous-domaines élémentaires

1. Qui se doit de posséder une densité énergétique importante afin d'envoyer le maximum de masse hors de l'atmosphère.

2. Par exemple, il est inutile d'implémenter une méthode numérique parfaitement adaptée à la modélisation d'un seul gaz donné et incapable de prendre en compte d'autres types de gaz ou leur mélange.

pouvant être maillés simplement. Avant même que chaque partie soit maillée, il faut effectuer une étape de composition topologique assurant la compatibilité entre les différents maillages dénommée *blocking*. Cette étape peut devenir extrêmement fastidieuse et la recherche est particulièrement active à l’heure actuelle afin de faciliter ce processus, à cet égard on pourra citer les travaux de Peron [Per14], Chan [Cha09b] et Lakshminarayan *et al.*[LSRW16].

- Les maillages non structurés qui utilisent des éléments volumiques définis par leurs nœuds et par leurs faces qui sont reliées par une table de connectivité. Les maillages non structurés sont souvent assimilés à des maillages composés uniquement de simplexes (triangles en 2D et tétraèdres en 3D) mais ils peuvent être hybrides et associer plusieurs types d’éléments; hexaèdres, prismes, pyramides et tétraèdres sont les « éléments canoniques » qui sont principalement utilisés. Dans la suite de cette thèse nous ne manipulerons que des maillages non structurés hexaédriques, cependant le code FLUSEPA<sup>3</sup> développé par Airbus Safran Launchers supporte tous les éléments énoncés précédemment. La génération de maillages non structurés majoritairement composés de tétraèdres est parfaitement adaptée aux configurations complexes que manipulent les industriels, parce qu’elle permet de construire rapidement un maillage. Cependant le coût et la qualité de la simulation en pâtissent souvent. En effet, pour une même densité de mailles, un solveur non structuré utilisant principalement des tétraèdres sera non seulement plus coûteux en nombre d’opérations et en espace de stockage mémoire mais aussi moins précis numériquement pour des raisons d’absence de « compensation naturelle » d’erreur de troncature pour des mailles non régulières<sup>4</sup>. De plus, et contrairement aux tétraèdres, les hexaèdres autorisent l’alignement « naturel » des lignes de faces avec les lignes de courant ce qui permet d’éliminer efficacement toute diffusion numérique.

Récemment, Park *et al.* [PLK<sup>+</sup>16] ont évalué les besoins en terme de maillages non structurés pour les simulations numériques des 20 prochaines années. Ils prédisent qu’il sera indispensable d’ici 5 ans de disposer de méthodes robustes d’adaptation de maillages. Dans un premier temps, la mise en œuvre de ces méthodes devrait permettre de réduire le travail de l’utilisateur dès la phase de génération de maillages. En effet, la construction d’un modèle numérique repose actuellement sur l’estimation *a priori* de la densité de mailles nécessaire en tout point pour simuler correctement l’écoulement dans son ensemble. En utilisant une adaptation dynamique, seules les mailles essentielles à la description précise de la géométrie des objets étudiés et de leur environnement (pas de tir, conteneur, champ proche, veine de soufflerie...) seront initialement nécessaires et donc construites. Ensuite, les contraintes liées à la physique de l’écoulement imposées aux maillages suivant des « critères adéquats » devraient permettre le déroulement de chaque simulation à un coût optimal mais aussi de pouvoir effectuer des simulations qui sont jugées pour l’instant inabordables.

Le présent manuscrit se focalisera sur la mise en œuvre d’une méthode d’adaptation de maillages par enrichissement sur des grilles chevauchantes non structurées. La construction de ce module représente les fondations d’un travail de longue haleine ayant pour but de combiner différentes techniques d’adaptation de maillages afin de tirer le meilleur parti de chacune d’elles pour assurer un suivi ainsi qu’une représentation précise et performante des structures critiques des écoulements aéropulsifs. Depuis quelques années, le code de calcul industriel FLUSEPA a fait l’objet de plusieurs développements dans le cadre d’un programme ambitieux de maîtrise et de modernisation des capacités de calcul CFD<sup>5</sup> d’ArianeGroup (un récapitulatif des capacités de FLUSEPA est présenté figure 2.1). Ce code emploie une technique très spécifique, unique en son genre, de maillages chevauchants s’apparentant aux méthodes *Chimère*. Ces dernières ont été initiées par Volkov dans les années 60 [Vol66, Vol68] et repris par Starius à la fin des années 70 [Sta80, Sta77] pour être enfin mis en œuvre par Steger *et al.* [SDB83]. A la différence des techniques *Chimère* classiques (basées sur une interpolation inter-grilles), la méthode de recouvrement que l’on décrira par la suite, utilise la surface de l’intersection géométrique entre maillages chevauchants pour calculer les flux entre ceux-ci. Il n’y donc aucune interpolation effectuée. L’ajout d’une méthode d’adaptation de maillages par enrichissement dans ce cadre très particulier,

3. Marque déposée en France numéro 134009261.

4. En effet, pour les mailles dont la géométrie et le support sont à symétrie centrale, dans le cas de méthode Volumes Finis de type Godounov [GZI<sup>+</sup>76], on peut démontrer que l’erreur de troncature est du même ordre de grandeur que l’erreur de calcul des flux.

5. Computational Fluid Dynamics.

afin d'automatiser et de permettre à la méthode de chevauchement de maillages d'être plus efficace, est une étape clé pour la suite des développements du code FLUSEPA. Cette nouvelle capacité vise en premier lieu à automatiser les études aéropulsives d'engins spatiaux faites par ArianeGroup.

Ce rapport décrit l'ajout d'une méthode de raffinement de maillages adaptatifs (AMR<sup>6</sup>) pour des grilles chevauchantes. La thèse s'attache plus particulièrement à simplifier la génération du maillage de l'espace et à adapter celui-ci en fonction des phénomènes physiques ayant lieu durant la simulation. Cette adaptation de maillages doit se faire de façon conservative tout en limitant la perte d'informations au cours des différentes phases de raffinement/déaffinement des maillages. Les travaux ainsi menés posent la première pierre d'une stratégie globale de génération visant à terme le couplage de trois méthodes d'adaptation de maillages :

- par création d'arbres de relation *kd-tree* entre cellules,
- par déformation de maillages,
- par suivi de structures cohérentes (par duplication de zone du maillage)<sup>7</sup>.

Le développement relativement ambitieux d'une telle stratégie de maillage repose sur le fait qu'il serait contre productif pour les études futures de mettre en œuvre la seule expertise humaine pour gérer efficacement l'ensemble des tâches liées aux maillages. Ainsi, les techniques d'adaptation présentées au cours de cette étude doivent permettre la détection des zones sur ou sous-résolues afin d'optimiser le coût des simulations effectuées en ajustant en tout point la densité des maillages.

Les travaux menés au cours de cette thèse ont abouti à plusieurs évolutions du code FLUSEPA :

1. adaptation et ajout de nouvelles structures de données permettant la modification dynamique d'un maillage, la création d'arbres de relation entre les éléments,
2. modifications nécessaires pour l'utilisation du raffinement de maillages adaptatif (AMR). Ces améliorations comportent principalement :
  - l'ajout de senseurs pour la détection des problèmes liés à l'inadéquation entre la résolution locale du maillage et les phénomènes étudiés,
  - le développement d'un module permettant le raffinement et le déaffinement des maillages par subdivision ou fusion des mailles concernées et création d'arbres *octree* (arbre de relation entre un élément parent et 8 éléments fils) compatibles avec la notion de chevauchement de maillages
  - le traitement des faces divisées (zones pseudo non conformes, figure 4) en modifiant la définition topologique des éléments dans FLUSEPA,
  - l'ajout d'une méthode de redistribution conservative d'ordre élevé des variables sur les mailles filles lors de la subdivision d'une cellule
3. modification des entrées/sorties du code,
4. création d'un ensemble de cas tests de validation fonctionnelle et des outils de visualisation des données associés afin de faciliter le développement et la validation des techniques liées à l'adaptation dynamique de maillages.

Ce manuscrit décrit le travail qui a été effectué et s'organise comme suit :

- le chapitre 1 fait l'état de l'art des méthodes d'adaptation de maillages par enrichissement pour des problématiques instationnaires,
- le chapitre 2 décrit la stratégie numérique et l'architecture du code FLUSEPA,
- le chapitre 3 présente la méthode AMR *octree* et les différentes spécificités algorithmiques associées,
- le chapitre 4 traite de la méthode de redistribution conservative d'ordre élevé après subdivision d'une cellule,

---

6. Pour *Adaptive Mesh Refinement*.

7. La méthode d'adaptation par suivi de structures cohérentes est d'autant plus intéressante qu'elle permet de suivre une structure avec un maillage qui se déplace avec la même vitesse de groupe que celle-ci. Cette propriété permet de limiter les erreurs liées aux schémas numériques utilisés.

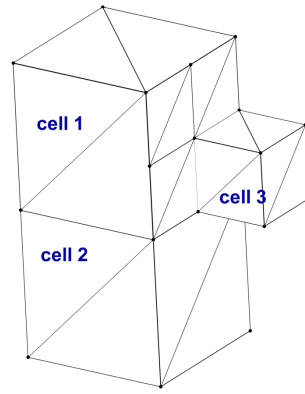


FIGURE 4 – Zone pseudo non conforme obtenue après raffinement

- le chapitre 5 décrit les simulations effectuées avec le démonstrateur ainsi obtenu,
- le chapitre 6 rappelle l'ensemble des résultats obtenus et décrit les prochains grands axes de développement de FLUSEPA, liés à la thématique de l'adaptation de maillages.

# Chapitre 1

## Les méthodes de raffinement de maillages adaptatif

### Introduction

Les méthodes de raffinement de maillages adaptatif (AMR) ont été développées initialement afin de permettre la simulation de système d'équations aux dérivées partielles hyperboliques (EDPH) discrétisées. Ces méthodes adaptent en subdivisant de façon dynamique les mailles contenues dans un maillage en fonction de critères et augmentent localement la précision des calculs effectués. Les EDPH sont caractérisées par la présence d'ondes dont la vitesse de propagation est finie. Ainsi, en appliquant une méthode d'adaptation de maillages suffisamment fréquemment, il est possible de capturer des structures qui ne peuvent pas s'échapper de la zone « raffinée »<sup>1</sup>.

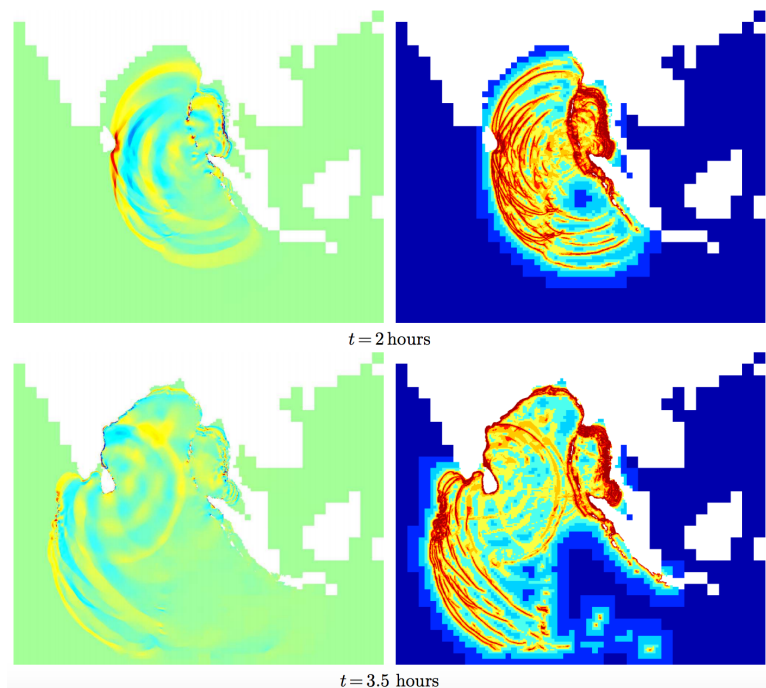


FIGURE 1.1 – Simulation du tsunami de 2004 dans l'océan Indien utilisant une méthode d'adaptation *quadtree* [Pop11]. Colonne de gauche, hauteur de la vague, colonne de droite résolution du maillage

1. A noter que cette propriété est conservée si les équations contiennent des termes visqueux faibles et/ou si la solution est stationnaire. Ainsi, l'adaptation peut aussi être utilisée avec des équations paraboliques.



L'utilisation et le développement de méthodes AMR a été démocratisé au début des années 80 par la thèse de Berger [Ber82], puis par les travaux de Berger et Olinger [BO84] et ceux de Berger et Colella [BC89]. Depuis, le champ d'application s'est largement étendu et les domaines scientifiques nécessitant la simulation d'échelles très variables en constituent les champs d'application les plus intéressants. En effet, si la disparité d'échelles à simuler est très grande et si les petites échelles sont très localisées et se déplacent à travers l'intégralité du maillage, il est quasiment impossible d'effectuer un calcul complet de façon performante, sans adaptation dynamique de maillages. Par conséquent, l'AMR est un élément-clé de la simulation de certains domaines, notamment pour les simulations de tsunamis, pour lesquelles il est crucial de pouvoir effectuer une simulation rapide afin de pouvoir prévenir les populations à temps des risques qu'elles encourent, figure 1.1.

Aujourd'hui, les méthodes AMR ont tendance à se démocratiser et de simples recherches internet ou même certains actes de communication de grandes industries répertorient un nombre grandissant de secteurs scientifiques impactés par l'AMR [Conc]. Parfois, le noyau du code de simulation contient directement le module AMR qui impose des contraintes fortes sur l'architecture logicielle et la stratégie numérique de simulation. Cette étude bibliographique a volontairement été restreinte au champ d'application très particulier qui nous intéresse, à savoir, les méthodes de raffinement de maillages recouvrants adaptatifs pour des écoulements instationnaires avec utilisation d'un schéma temporel explicite. Cet état de l'art attache une importance particulière à l'étude de la conservativité, à l'ordre des méthodes décrites et la cohérence globale de la stratégie de simulation.

Ce chapitre s'articule comme suit ; tout d'abord nous rappelons les grandes catégories des méthodes AMR pour des maillages structurés ou non-structurés, ce qui nous permet de positionner les travaux que nous allons exposer dans le contexte des méthodes d'adaptation. L'AMR contraint les méthodes numériques qui doivent supporter de fortes disparités locales de taille de mailles. Sans traitements particuliers, les performances des codes s'effondrent, c'est pourquoi nous présentons des méthodes efficaces d'avancement en temps explicite adaptatives, telles qu'elles sont utilisées dans la plupart des codes contenant un module AMR. Nous étudions ensuite rapidement les méthodes de projection de variables lors de la subdivision d'une cellule ayant pour but d'obtenir une subdivision conservative, « précise »<sup>2</sup> et bornée. Le repérage des cellules devant être raffinées est critique quand à la précision du calcul et repose sur des senseurs de raffinement permettant la résolution en temps des structures caractéristiques d'un écoulement. Nous analysons quelques senseurs issus de la littérature. Enfin, nous comparons quelques unes des stratégies d'adaptation AMR pour des maillages chevauchants qui ont été développées dans certains codes tant industriels que de recherche.

## 1.1 Les différents types de méthodes AMR

On distingue deux grandes familles de méthodes AMR, dont une divisée en deux sous-catégories (figure 1.2) :

1. les méthodes construisant des arbres de relation entre grilles. Ces méthodes incrustent de nouvelles grilles dans les grilles existantes, on parle alors de système de grilles hiérarchisées (figure 1.3). Les méthodes construisant des grilles hiérarchisées peuvent elles-mêmes être divisées en deux sous-catégories distinctes :
  - celles utilisant un raffinement par bloc (*block-based* AMR). Le maillage est découpé en blocs, si une cellule d'un bloc doit être raffinée alors l'ensemble du bloc est raffiné. De nouvelles grilles sont créées et superposées aux grilles précédentes. Afin de raffiner et manipuler les blocs, des arbres de relation entre les différentes grilles<sup>3</sup> sont créés. Les méthodes de raffinement par bloc entraînent souvent un sur-raffinement des zones critiques de l'écoulement. En outre, il faut bien noter que les grilles raffinées sont superposées aux grilles précédentes donc les grilles qui ont été raffinées sont toujours calculées mais uniquement par interpolation. Une analyse des méthodes AMR par bloc a été menée par Dubey *et al.* [DAB<sup>+</sup>14] et donne le détails des propriétés des différents grand modules disponibles aujourd'hui.

2. En supposant que les variables soient suffisamment régulières, il est possible de faire un développement de Taylor et de définir un ordre de précision représentant l'ordre des termes modélisés, jusqu'à la troncature.

3. Les modules de référence pour l'AMR par blocs : SAMRAI, Chombo

- celles effectuant un raffinement par patch (*patch-based*). Les zones raffinées sont optimisées en étant regroupées et certaines parties superflues peuvent être déraffinées. Berger et Olinger, puis Berger et Collela ont été les pionniers de ce genre de méthodes<sup>4</sup>.
2. les méthodes qui construisent des arbres de relation par cellule (*cell-based*) autorisent chaque cellule à être raffinée indépendamment. Ce type de méthode limite le plus possible la création de cellules pour capter une structure. L'AMR *cell-based* est à la fois utilisée sur des maillages structurés mais aussi non structurés. En particulier, pour des maillages non-structurés, les subdivisions possibles peuvent être nombreuses et souvent les méthodes de raffinement se restreignent à un certain nombre de subdivisions possibles. On présente sur la figure 1.4 les subdivisions retenues par Kalinderis et Kavouklis [KK05, KK10]. Leicht et Hartmann [LH10, HHL11] introduisent dans leur travaux une méthode AMR *cell-based* non-structurée pour des hexaèdres qui traite les cas d'anisotropie par construction d'arbres *di-*, *quad-* et *oct-tree* (figure 1.5<sup>5</sup>).

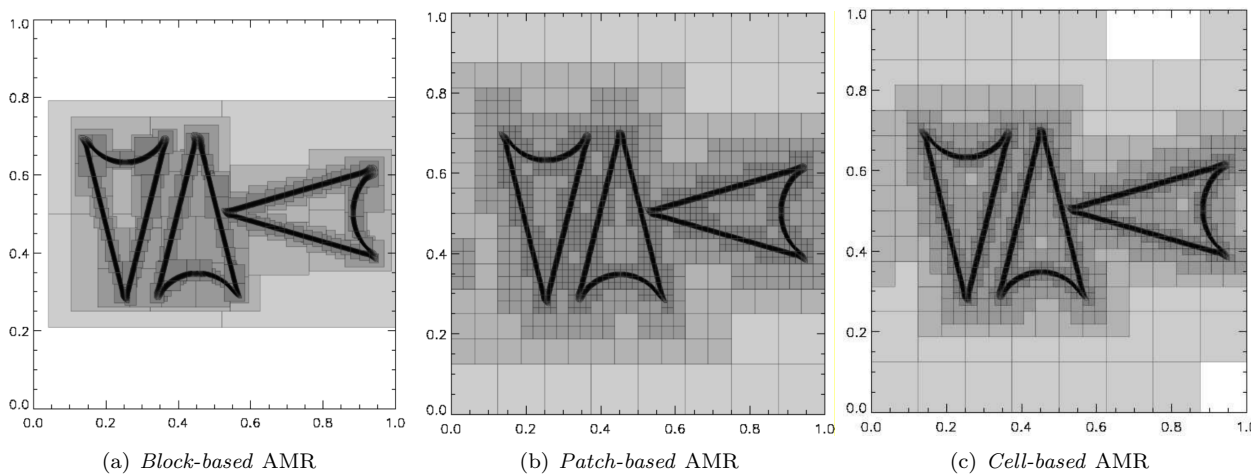


FIGURE 1.2 – Différentes méthodes AMR pour la création de grilles

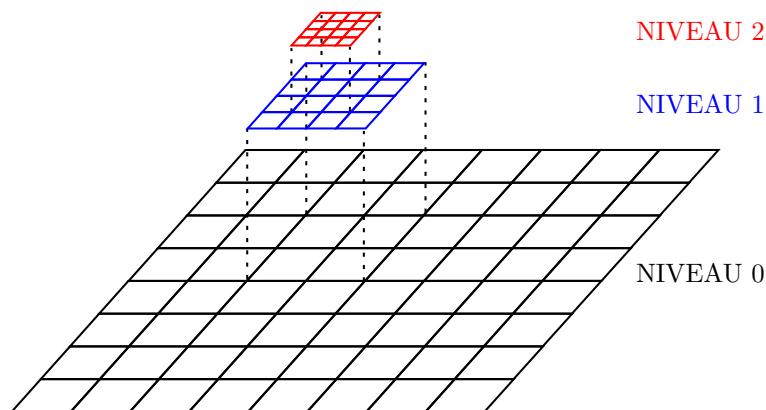


FIGURE 1.3 – Exemple de grilles hiérarchisées sur 3 niveaux

4. Le module de référence sera donc celui de Berger, dont le développement a été fait conjointement avec Levêque à savoir CLAWPACK [LB<sup>+</sup>11].

5. A titre d'exemple, on peut aussi citer les codes RAMSES [Tey02], HERA [Jou05].

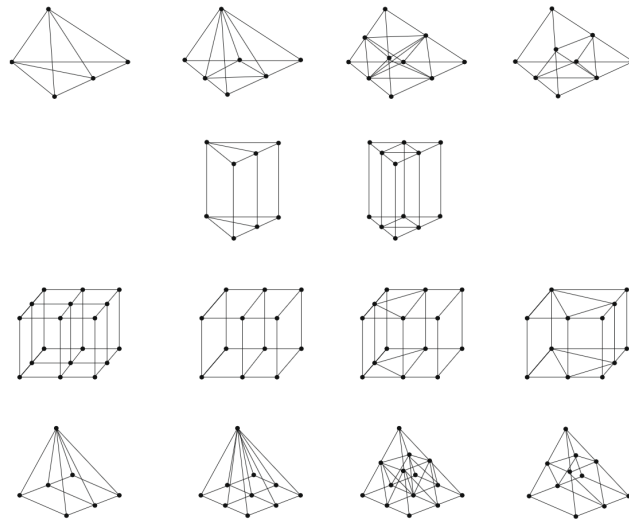


FIGURE 1.4 – Création d’arbres pour des maillages non structurés (extrait de Kavouklis et Kallinderis [KK10]). Première ligne : raffinement d’un tétraèdre, deuxième ligne : raffinement d’un prisme, troisième ligne : raffinement d’une pyramide, quatrième ligne : raffinement d’un hexaèdre.

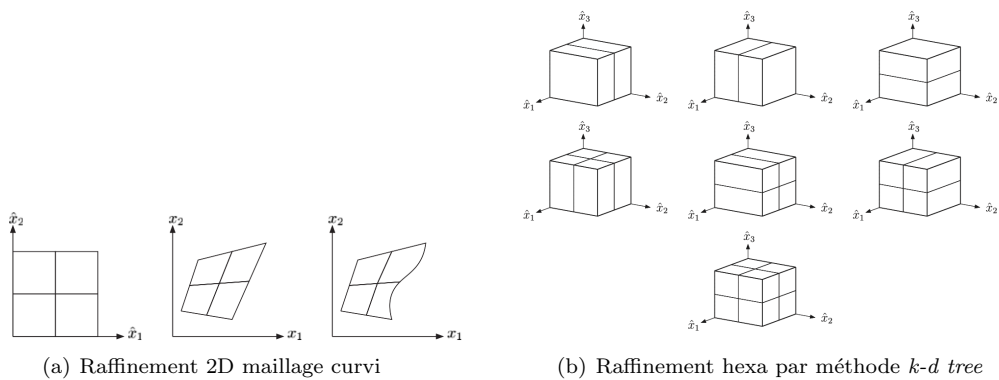


FIGURE 1.5 – Création d’arbres pour des maillages non structurés [LH08, LH10]

## 1.2 Schéma explicite d'avancement en temps

Nous étudions ici quelques unes des méthodes d'avancement en temps efficaces dans le cadre des méthodes AMR utilisant des grilles hiérarchisées. Le raffinement de maillages introduit des transitions brusques en termes de taille de mailles qui se répercutent sur les pas de temps physiques caractéristiques à chacune d'entre elles. Pour simuler correctement les phénomènes instationnaires « raides », il est nécessaire d'utiliser un schéma d'intégration explicite. Différentes méthodes permettant un avancement en temps ont été développées afin de coupler ce type de schémas avec les méthodes AMR.

### 1.2.1 La méthode adaptative d'avancement en temps de Berger et Colella

Berger et Olinger [BO84] puis Berger et Colella [BC89] ont développé des schémas explicites permettant d'optimiser l'avancement en temps des différentes grilles. Ces schémas dits « adaptatifs » font avancer les différentes grilles avec des pas de temps qui leur sont propres en utilisant une condition de Courant-Friedrichs-Lewy. Pour chaque nouvelle grille construite par raffinement, un front de cellules fictives, dites cellules fantômes (*ghost cells*), est construit. Ce front sert de conditions aux limites pour les grilles raffinées (figure 1.6). Si ces conditions aux limites particulières sont situées dans une grille plus grossière, alors elles sont remplies en utilisant les informations des grilles de niveau inférieur (figure 1.6 : l'intégralité des cellules fantômes est située dans la grossière), sinon ce sont des conditions limites simples ou des interfaces simples avec d'autres grilles de niveau équivalent.

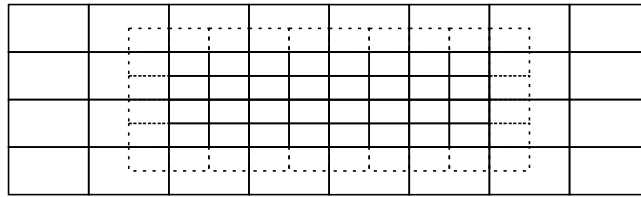


FIGURE 1.6 – Deux grilles incrustées, en pointillés un front de cellules fantômes pour la grille fine

L'intérêt de ce schéma réside dans le fait qu'ils soit simple à mettre en place. En effet, le schéma numérique est presque partout identique à schéma explicite à pas de temps global. La procédure peut être vue comme un algorithme itératif qui pour faire avancer en temps une grille de niveau  $l$ ,  $0 \leq l \leq l_{max}$  effectue les actions suivantes :

1. Avancer le niveau  $l$  d'un pas de temps  $\Delta t^l$  comme si c'était la seule grille. Si  $l > 0$ , récupérer les informations de bords en utilisant les flux interpolés des grilles de niveau  $l - 1$  et les conditions aux limites.
2. Synchroniser les niveaux  $l$  et  $l + 1$  et interpoler les corrections vers les grilles plus fines (de niveau  $[l + 2, \dots, l_{max}]$ ).

Le schéma temporel adaptatif ainsi décrit effectue l'avancement en temps de chaque grille de façon indépendante (sauf pour les conditions limites) et ensuite calcule une correction permettant de synchroniser les différents niveaux. De façon simplifiée, l'objectif de cette synchronisation est d'apporter une correction à la grille de niveau  $l$  pour garantir la consistance du schéma dans les conditions aux limites. Cependant, du fait des interpolations et des synchronisations, l'ordre de précision et la conservativité lors du calcul des flux sont plus difficiles à garantir. Des erreurs sont commises à l'interface entre niveaux de raffinement, dans les zones où la projection en temps (ou interpolation) est utilisée. Lors du calcul des flux sur les faces inter-niveaux avec présence de cellules fantômes, ceux-ci ne sont pas définis de façon univoque, comme présenté schématiquement sur la figure 1.7 qui comprend deux cellules  $\Omega_J$  et  $\Omega_K$ , de niveau de raffinement  $l$  et  $l + 1$  respectivement. La cellule  $\Omega_{J_i}$ , dont le centre de gravité est présenté en rouge sur la figure 1.7, est une cellule fantôme. Comme évoqué précédemment, cette cellule permet de ne pas modifier les schémas numériques classiques en calculant un flux  $\mathcal{F}_{J_i K}$  pour la cellule  $\Omega_K$ . Cependant,  $\mathcal{F}_{J_i K} \neq -\mathcal{F}_{K J}$ , il y a incompatibilité des flux avec les schémas numériques spatiaux

classiques conçus pour des maillages structurés conformes (*cf.* définition 9). Le fait que les flux ne soient pas univoques conduit à une perte de précision au niveau des interfaces inter-niveaux de raffinement. Colella *et al.* expliquent ainsi dans le document de référence de Chombo [CGJ+09] qu'il existe une perte de précision dans les interfaces inter-niveaux de raffinement mais que cette perte de précision est, selon eux, acceptable. En effet, pour des problèmes elliptiques et paraboliques, une erreur de troncature de  $\mathcal{O}(h^{p-1})$  dans un ensemble de codimension 1<sup>6</sup> induit une erreur  $\mathcal{O}(h^p)$  sur la solution. Pour des problèmes hyperboliques, une erreur de troncature d'ordre  $\mathcal{O}(h^{p-1})$  sur un sous-ensemble de codimension 1 induit une erreur  $\mathcal{O}(h^p)$  pour la norme  $L_1$  (et  $L_\infty$  si les conditions aux limites ne comprennent pas d'interface inter-niveaux).

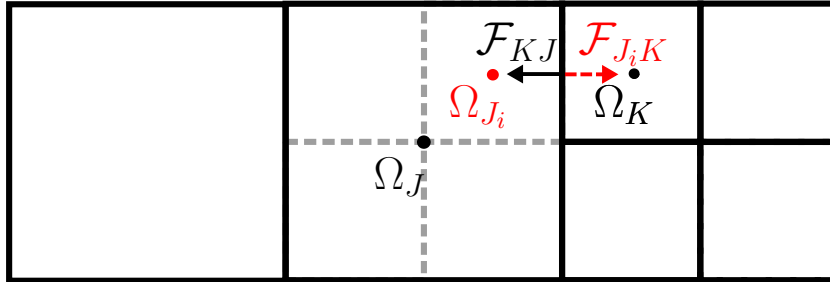


FIGURE 1.7 – Exemple d'utilisation classique de ghost-cells dans la cas d'une méthode explicite. En rouge le flux à l'interface fin-grossier pour la cellule  $\Omega_k$  qui utilise la cellule virtuelle  $\Omega_{J_i}$ . En noir : le flux calculé pour la cellule  $\Omega_J$

Quelques méthodes permettent de limiter l'impact de cette perte de précision locale. Bell [Bel05] applique une correction qui peut s'apparenter à une condition de Dirichlet ce qui induit une inconsistance dans les flux inter-niveaux de raffinement. Cette condition de Dirichlet ajoute un terme d'erreur sur la moyenne qui doit ensuite être compensé par résolution d'un système linéaire. Martin et Collela [MC00] décrivent plusieurs problèmes, notamment de conservativité, liés à l'utilisation d'un schéma temporel adaptatif lors des projections en temps de la méthode introduite par Bell, Colella et Glaz [BCG87]. Ils introduisent une correction des champs liée à un scalaire passif<sup>7</sup>. Lors de la projection en temps s'il y a déviation par rapport à la valeur théorique du scalaire passif, alors l'erreur de conservativité est injectée dans une équation de Poisson pour obtenir un champ potentiel. Ce champ potentiel permet alors de corriger la vitesse d'advection sans pour autant garantir la conservativité. Une étape supplémentaire de synchronisation des flux est donc nécessaire. Les auteurs précisent que ces corrections ne réduisent pas les erreurs  $L_1, L_2$  et  $L_\infty$  sur les variables de l'écoulement, si celles-ci varient beaucoup. Cependant dans le cas de champs quasi-constants, l'utilisation de cette correction accélère grandement la convergence des simulations.

Pour calculer les gradients et affecter une valeur aux cellules fantômes, Martin *et al.* [MC00, MCG08] utilisent une méthode de double projection, comme présenté sur la figure 1.8. Leur algorithme est le suivant :

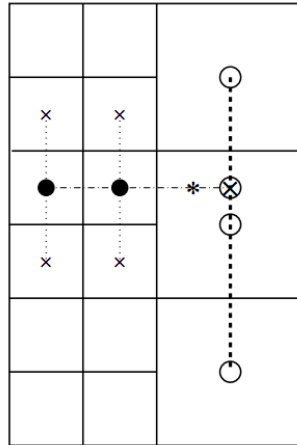
1. Calculer les gradients des cellules raffinées, en utilisant les valeurs notées "X" pour obtenir les valeurs centrées aux cellules marquées de cercles pleins
2. Calculer les gradients inter-niveaux par interpolation sur les cellules notées d'un cercle avec une croix en utilisant une interpolation quadratique utilisant 3 points de la grille la moins raffinée (cercles non pleins)
3. Calculer le gradient tangentiel dans les *ghost cells* marquées d'une astérisque par interpolation quadratique des gradients tangentiels des cellules raffinées et non raffinées.

6. En dimension finie, la codimension est la dimension du supplémentaire d'un sous-espace F vectoriel dans un espace vectoriel E.

7. Un scalaire passif  $\Lambda$  peut être ajouté dans un écoulement en résolvant l'EDP suivante :

$$\begin{aligned} \frac{\partial \Lambda}{\partial t} + \nabla \cdot \mathbf{u} \Lambda &= 0 \\ \Lambda(\mathbf{x}, \mathbf{t} = \mathbf{0}) &= 1 \end{aligned}$$

cela permet en général la visualisation de l'écoulement et peut être assimilé aux particules ou au colorants que l'on ajoute expérimentalement en mécanique des fluides pour mettre en exergue la structure d'un écoulement.

FIGURE 1.8 – Interpolation des gradients tangente aux interfaces inter-niveaux de raffinement [CMG<sup>+</sup>13]

Ces méthodes sont décentrées lorsque les cellules à interpoler sont situées dans un coin (plus d'une interface inter-niveaux pour une cellule). Martin et Collela précisent qu'ils se limitent à une interpolation d'ordre 2. Ils notent qu'une interpolation d'ordre 3 est trop sensible et nécessite de sur-raffiner les zones autour des structures détectées (donc de diminuer les seuils des critères), ce qui dégrade l'efficacité de l'AMR en augmentant le nombre de mailles.

### 1.2.2 Efficacité de l'intégration explicite avec pas de temps par grilles

Dursi et Zingale [DZ05] étudient le gain obtenu en n'effectuant que des interpolations dans les grilles grossières et concluent logiquement que plus la proportion de petites mailles dans le maillage est importante moins le schéma explicite avec des sous-cycles d'intégration est efficace. Par conséquent, tant que les zones raffinées peuvent être assimilées à des droites ou des surfaces contenues dans le volume à simuler, le schéma est efficace. Cependant, si la zone à raffiner est assimilable à un volume inclus dans le domaine, alors les gains sont relativement faibles. Il est donc nécessaire de trouver des compromis dans le cas de maillages AMR contenant proportionnellement un grand nombre de petites mailles.

Des travaux plus récents de Löffler *et al.* [LCBD16] s'intéressent à des cas où les maillages sont essentiellement constitués de mailles très raffinées pour lesquels les interpolations dans les *ghost cells* peuvent devenir extrêmement coûteuses au point de faire perdre tout intérêt aux schémas explicites adaptatifs<sup>8</sup>. Même si plusieurs pas de temps sont admissibles (un pour chaque grille cartésienne), il y a nécessité dans les grilles les moins raffinées d'effectuer des intégrations temporelles intermédiaires par rapport au pas de temps qui leur est propre, pour repositionner les variables et les flux associés. Cette opération de projection temporelle possède un coût important et rend la méthode non-concurrentielle par rapport à un schéma explicite à pas de temps global (NSAMR pour *Non-Subcycling AMR*). C'est pour cette raison que certaines bibliothèques de routines AMR permettent de ne pas utiliser d'intégration temporelle adaptative. A titre d'exemple, Boxlib [Pag], Cactus [GAL<sup>+</sup>03] et Chombo [CGJ<sup>+</sup>09] proposent cette option. La méthode CSAMR (pour *Concurrent Structured AMR*) de Löffler *et al.* fait rentrer en concurrence les deux méthodes d'intégration en temps, celle comprenant des pas de temps intermédiaires et celle à pas de temps global, pour optimiser encore plus l'intégration en temps, quand cela est nécessaire.

8. De fait, les maillages manipulés sont tellement optimaux que la proportion des mailles de niveau inférieur au niveau de raffinement maximal présentes dans le maillage est négligeable. Les maillages sont donc quasiment uniquement constitués de mailles très raffinées.

### 1.2.3 Hybridation de schéma explicite-implicite

Plutôt que d'utiliser un pas de temps local lors de l'utilisation d'un module AMR, Tóth *et al.* [TDZGP06, TDZM<sup>+</sup>08] se sont focalisés sur l'utilisation de méthodologies d'intégration couplées implicite/explicite combinées à l'AMR hiérarchisée par bloc pour gagner en efficacité. L'idée est assez simple, le schéma temporel peut être défini différemment pour chaque bloc et dépend d'un pas de temps imposé par l'utilisateur. L'ensemble des grilles avance ainsi avec le même pas de temps. Le schéma est donc explicite dans les blocs où la condition locale de stabilité permet d'effectuer une intégration explicite, et implicite dans les blocs où un schéma explicite est instable *i.e.* s'il ne satisfait pas la condition de CFL. Ce schéma a été implémenté dans le code de Magnéto-Hydro-Dynamique (MHD) BATSURUS [TSG<sup>+</sup>05] qui sert à la simulation de magnétosphère de planètes. Une autre méthode a été introduite récemment par May et Berger [MB15] dans le cadre de stratégies de chevauchement de type *cut-cells*<sup>9</sup> pour des problèmes de pas de temps caractéristiques très faibles dans les cellules proches de la paroi parce que ces cellules sont coupées.

## 1.3 Technique de projection/répartition de variables d'une maille vers une sous-maille

Lors du raffinement d'une cellule, il est nécessaire d'affecter une valeur aux différents champs dans les cellules créées. Plus généralement, la problématique d'affecter une valeur à des sous-ensembles  $\Omega_i$  inclus dans un volume  $\Omega$  est intrinsèque à toute découpe de mailles. À ce titre, la plupart des travaux d'intérêt sur le sujet sont liés aux méthodes utilisant une formulation Arbitrary Lagrangian Eulerian (ALE) et non aux méthodes AMR. En effet, dans le cadre des méthodes AMR, une revue bibliographique montre que la méthode choisie se limite souvent à une interpolation trilineaire des champs des cellules grossières vers les cellules créées en utilisant quelques limiteurs pour préserver la monotonie et empêcher l'apparition d'extrema locaux. Nous citerons uniquement l'étude de Calhoun et Levêque [CL05] car elle traite en particulier le cas où les mailles créées ne sont pas contenues dans la maille initiale. La valeur  $\kappa_{ij}$ , définie comme étant le ratio entre le volume théorique de la cellule et le volume réellement occupé dans l'espace de calcul, permet alors de corriger l'erreur commise sur le volume et les variables conservatives. Cependant, la méthode est peu précise sauf dans quelques cas, notamment celui où la somme des  $\kappa_{ij}$  des mailles filles après raffinement est égal au  $\kappa_{ij}$  de la maille mère dont elles sont issues.

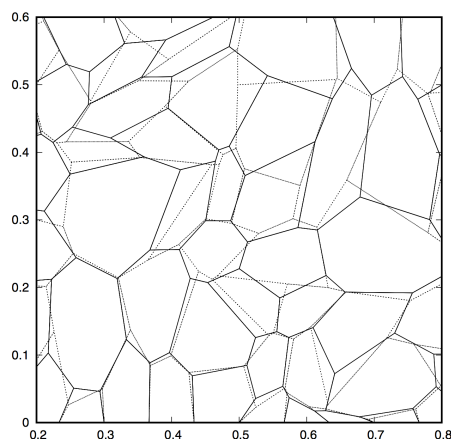


FIGURE 1.9 – Méthode de remaillage global. En trait pointillés le maillage initial, en traits pleins le maillage après une étape remaillage [LS05]

Dans le cadre des méthodes ALE, Loubere et Shashkov [LS05] ont introduit une projection 3D linéaire et conservative d'un maillage initial sur un maillage remaillé (voir figure 1.9). Récemment leurs travaux ont été

9. Ce type de méthode revient à éviter un maillage cartésien servant de grille de fond par des surfaces et à couper les cellules du maillage par les différentes facettes constitutives de la surface chevauchante.

repris par Velechovsky *et al.* [VLS13] qui ont mis en place une reconstruction parabolique par morceaux conservative. Cependant, les travaux de Velechovsky ne semblent pas être suffisamment matures pour des applications industrielles, les applications numériques étant limitées à des cas analytiques.

Des travaux récents de Re *et al.* [RDG17] effectuent une projection en réduisant une maille en un point en déplaçant les faces. Ce déplacement des faces induisant un flux sur chaque maille, ce flux est conservé. Les nouvelles mailles sont construites par gonflement des faces à partir du point en lequel on a réduit la maille et en calculant des flux associés. Les valeurs dans les deux nouvelles cellules peuvent alors être déduites. La procédure inverse de fusion de cellules est présentée sur la figure 1.10, les faces représentées sont celles du diagramme de Voronoï.

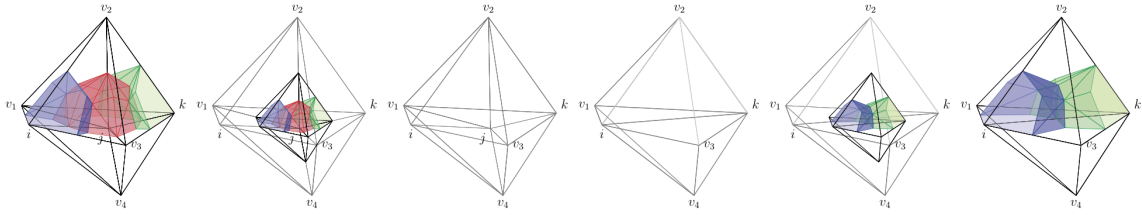


FIGURE 1.10 – Exemple de méthodes de projection de variables Lagrangienne, ici fusion de la cellule représentée dont le diagramme de Voronoï est dessiné en rouge [RDG17] avec les cellules bleue et verte

## 1.4 Les critères d'adaptation de maillages

Ce paragraphe s'intéresse à la notion de *senseur*<sup>10</sup> pour le raffinement de maillages pour des écoulements instationnaires. Le principe des méthodes AMR est de réduire ou d'augmenter la résolution locale du maillage afin d'équi-répartir l'erreur sur la solution calculée. Ainsi l'adaptation de maillages est toujours couplée à un *senseur* qui définit la résolution nécessaire pour le suivi de structures caractéristiques de l'écoulement. Une analyse bibliographique détaillée des senseurs pour l'adaptation sort du cadre de cette thèse. En effet, il existe presque autant de senseurs que de méthodes d'adaptation de maillages, chacun de ces senseurs ayant de bonnes propriétés pour certains domaines d'application et/ou certains codes de calcul. Cependant, deux grandes catégories se démarquent, les senseurs utilisant une estimation de l'erreur et les senseurs fondés sur la physique de l'écoulement.

Dans le contexte de la simulation industrielle intégrant une stratégie AMR, il est nécessaire de disposer d'une méthode à la fois souple et robuste ne nécessitant que peu d'interventions de l'utilisateur. Il est donc complexe de définir des senseurs qui évaluent la résolution d'un maillage et la comparent à des seuils tolérables de sur- ou sous-résolution et qui soient à la fois applicables pour tous types d'écoulements et simples à manipuler pour un utilisateur non formé. A ce titre, les travaux de Kamkar *et al.* [KJWS10], puis de Kamkar [Kam11] montrent la voie à suivre, telle qu'elle a été implémentée dans la plateforme HELIOS [WKP<sup>+</sup>10]. Ces travaux ont formalisé un grand nombre de critères physiques adimensionnés et invariants très simples à mettre en place, par exemple en s'appuyant sur un critère  $Q$ <sup>11</sup>, ou la seconde valeur propre de la matrice des dérivées premières de la vitesse  $\lambda_2$ . Cependant, ces critères s'appliquent principalement à des sillages tourbillonnaires et permettent simplement une évaluation *a posteriori* de la physique de l'écoulement, autrement dit, ils ne peuvent anticiper le mouvement des structures.

10. Ici, anglicisme pour « capteur » ou « sonde ».

11. Le critère permet d'identifier les structures cohérentes tourbillonnaires d'un écoulement, il est défini de la façon suivante :  $Q = \frac{1}{2} \Omega_{ij} \Omega_{ij} - S_{ij} S_{ij}$  avec  $\Omega$  : le terme rotation du tenseur de gradient de vitesse et  $S$  : le terme de cisaillement du tenseur de gradient de vitesse.



Pour traiter correctement l'instationnarité, Kamkar reprend aussi les travaux de Berger [BO84] en utilisant une méthode d'extrapolation de Richardson. L'idée est d'effectuer le même calcul sur un maillage grossier et un maillage fin. La différence entre la solution calculée sur les deux maillages peut être mise sous la forme d'une série :

$$\frac{u_g^{n+1} - u_f^{n+1}}{\Delta t} = \mathcal{O}(\Delta x_g^p) \sum_{i=0}^{\infty} \mathcal{O}(\Delta t^i) \quad (1.1)$$

avec  $u_g^{n+1}$  la solution calculée sur le maillage grossier et  $u_f^{n+1}$  la solution calculée sur le maillage à l'instant  $n + 1$ ,  $\Delta t$  le pas de temps utilisé,  $p$  l'ordre du schéma spatial. Ce type d'estimateur a l'avantage de pouvoir prédire l'évolution en temps des structures, dans des cas instationnaires.

Alauzet *et al.* [AGM<sup>+</sup>03, AFGM07] ont introduit un critère utilisant un point fixe transitoire qui tient compte de l'évolution temporelle des structures caractéristiques de l'écoulement. La méthode fonctionne comme suit : en partant d'un instant  $t$ , on applique un schéma d'avancement en temps classique sur un premier maillage  $\mathcal{M}_1$  pour atteindre l'instant  $t + \Delta t$ . Le maillage est alors adapté à l'instant  $t + \Delta t$  et ce nouveau maillage  $\mathcal{M}_2$  est utilisé pour ré-effectuer le calcul à partir de l'instant  $t$ . La solution initiale de l'instant  $t$  est préalablement interpolée depuis le maillage  $\mathcal{M}_1$  sur le maillage  $\mathcal{M}_2$ . Le schéma temporel est à nouveau appliqué pour atteindre l'instant  $t + \Delta t$ . Une seconde adaptation de maillages est opérée pour un obtenir un maillage  $\mathcal{M}_3$ . Cet algorithme est itéré  $n$  fois jusqu'à convergence. Le maillage  $\mathcal{M}_n$  à  $t + \Delta t$  devient la nouvelle condition initiale. Les études d'Alauzet *et al.* utilisant cette méthode point fixe ont principalement été menées sur des maillages non-structurés avec des méthodes d'adaptation fortement anisotropes pour des ondes de choc propagées dans des géométries complexes. La méthode permet de mieux capter les phénomènes transitoires de ce type, elle est cependant relativement onéreuse, à nombre de nœuds équivalents, l'adaptation avec résolution d'un problème de point fixe transitoire est au moins deux fois plus chère qu'une méthode d'adaptation classique.

## 1.5 Les méthodes AMR avec maillages chevauchants

Nous utilisons la stratégie introduite par Meakin [Mea98, Mea00] comme référence pour les méthodologies AMR en maillages chevauchants. Le principe du chevauchement de maillages sera introduit plus en détail dans la section 2.3. On ne s'attarde ici que sur les méthodologies d'adaptation de maillages dans un contexte utilisant plusieurs maillages avec un recouvrement non-nul. Plus particulièrement, nous nous plaçons dans un cadre comprenant un maillage proche des corps à simuler et des maillages loin des corps ou paradigme *near-body/off-body* pour la génération de maillages. De façon générale, les maillages proches paroi sont des maillages curvilignes adaptés aux couches limites et les maillages *off-body* sont cartésiens.

L'adaptation introduite par Meakin dépend :

- de la distance à la paroi des différentes grilles, ainsi les maillages en début de calcul possèdent de bonnes propriétés et peuvent être générés totalement automatiquement en définissant des boîtes dans lesquels sont créés les maillages cartésiens (figure 1.11). Un ensemble de modules permet de regrouper les différentes zones raffinées et de les assembler en parallélépipèdes par coalescence, ceci limite le nombre de grilles cartésiennes. Cette technique de génération permet de simplifier la génération de maillages pour une large gamme d'écoulements. Cette méthodologie a été récemment reprise et étendue par Péron et Benoit [PB13],
- des variables de l'écoulement (gradients, ou autre senseurs de sous-résolution), en appliquant des senseurs sur les cellules, les faces, les arêtes ou les nœuds, il est alors possible de définir des zones à raffiner dans les maillages manipulés.

Nous citons ici quelques codes CFD matures couplant un paradigme *near-body/off-body* donc comprenant un chevauchement de maillages avec un module AMR. Souvent, uniquement le solveur *off-body* contient un noyau AMR permettant l'adaptation de maillages, c'est le cas de la plateforme Helios [WH00] qui utilise le

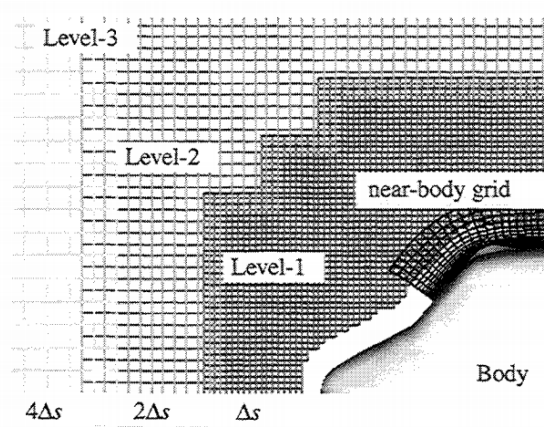


FIGURE 1.11 – Maillage *near/off body* initial après adaptation par *octree* permettant d’harmoniser la résolution des grilles dans la zone de chevauchement pour une simulation d’hélicoptère [Mea01]

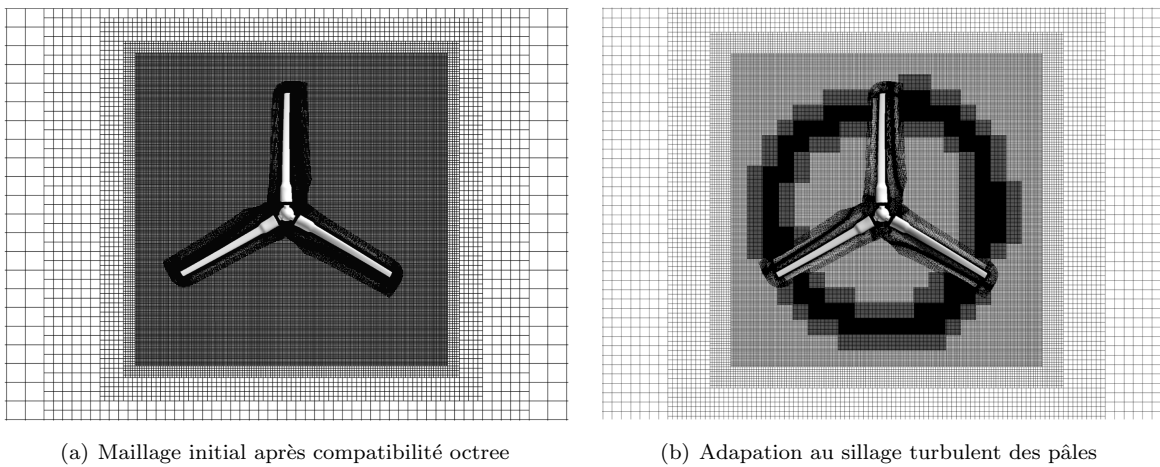


FIGURE 1.12 – Maillage avant et après adaptation utilisés par OVERFLOW2 [HP09]

solveur SAMRAI. Le code OVERFLOW [HP09] effectue aussi uniquement une adaptation sur la partie *off-body* des maillages comme présenté sur la figure 1.12. L’utilisation d’un module AMR dans OVERFLOW permet de faire varier les paramètres des simulations en ne modifiant que relativement peu les maillages utilisés, ceux-ci étant adaptés dynamiquement à l’écoulement. Ainsi, des bases aérodynamiques de séparations des boosters du lanceur Space Launcher System [CPWJ+16], un lanceur spatial actuellement en phase de développement par la NASA, permettent d’étudier les incertitudes des simulations<sup>12</sup>. Le code elsA [els] développé par l’ONERA est aussi couplé à une stratégie d’adaptation AMR pour des maillages chevauchants, présentée par Hue *et al.* [HPW+15] qui permet à partir d’une solution obtenue convergée, d’appliquer un critère de raffinement, de modifier le maillage puis de relancer le calcul et d’itérer le processus jusqu’à convergence. Le code OVERTURE [HS06] utilise des maillages structurés *near-body*, l’AMR est donc appliqué sur des maillages curvilignes proche paroi, ce qui en fait un cas particulier par rapport aux codes précédemment cités.

## 1.6 Conclusion

Cette étude bibliographique a rappelé les grandes catégories de méthodes de raffinement de maillages adaptatifs utilisées à l’heure actuelle : par bloc, par patch, par cellule. Les seules méthodes AMR applicables à des

<sup>12</sup>. Il est à noter que dans un effort de fusion des différents codes de calcul de la NASA, OVERFLOW a été intégré au code Launch Ascent and Vehicle Aerodynamics (LAVA) qui dispose donc du même module d’adaptation de maillages.

maillages non-structurés sont celles construisant des arbres de relation par cellule étant donné l'absence de paramétrisation possible des relations de connectivité. Nous avons décrit diverses méthodes classiques d'intégration adaptative explicite et souligné les problèmes de consistance et de précision dans les interfaces inter-niveaux de raffinement. Nous expliciterons par la suite la méthode d'intégration adaptative déjà incluse dans FLUSEPA, qui est quant à elle, par nature, parfaitement consistante et conservative étant donné l'absence d'une double définition des flux pour les interfaces inter-niveaux. Une rapide étude des techniques de redistribution des variables lors de la subdivision d'un volume a montré que garantir l'ordre de précision d'une redistribution conservative est encore aujourd'hui une question ouverte en 3D. Pour simuler correctement les écoulements instationnaires fortement énergétiques, il est primordial de respecter le principe du maximum, de garantir la conservativité et l'ordre de précision, de sorte que lors du raffinement des cellules, l'adaptation de maillages reste totalement cohérente l'ensemble des schémas d'intégration numériques mis en place. C'est pourquoi nous apporterons une réponse à cette problématique au cours de cette thèse. Ne rentrant pas dans le cadre de cette thèse, la mise en place de senseurs d'adaptation a été peu détaillée. Ainsi, nous avons choisi de n'utiliser que des critères simples et peu sélectifs pour les simulations réalisées. Ces critères sont présentés à l'annexe A. Enfin nous avons positionné les travaux de cette thèse par rapport aux codes CFD existants qui utilisent une méthode de chevauchement de maillages et des méthodes AMR d'adaptation de maillages. Nous avons pu observer qu'il n'existe pas, à notre connaissance, de méthodes AMR pour des maillages chevauchants non-structurés. Les travaux engagés dans cette voie, propres à l'approche et à la logique de FLUSEPA, sont donc novateurs et proposent une nouvelle méthodologie d'adaptation.

## Chapitre 2

# Stratégie numérique pour la simulation d'écoulements instationnaires

### Introduction

Le développement d'un code de mécanique des fluides nécessite d'avoir au préalable réalisé des choix de stratégie numérique. En effet, il n'existe pas aujourd'hui de méthode de simulation qui soit à la fois universelle et performante pour résoudre les équations de la mécanique des fluides. Les pré-requis de la stratégie à mettre en place sont dictés par les caractéristiques des écoulements que l'on souhaite simuler. Ils doivent assurer une cohérence à l'ensemble des différentes parties d'un code de calcul. FLUSEPA présente des caractéristiques particulières, qui garantissent une performance optimale pour certains types de simulations. On comprendra aisément que l'intégration d'un module d'adaptation de maillage dans FLUSEPA impacte plusieurs pans de la stratégie de simulation déjà en place. Ce code a été originellement développé pour pouvoir simuler les séparations d'étages de lanceurs spatiaux. Il devait donc être capable de simuler des corps en mouvement relatif, ainsi que des écoulements compressibles et fortement énergétiques. Aujourd'hui, la version de production de FLUSEPA s'exécute sur quelques centaines d'unités de calcul (CPUs)<sup>1</sup> en utilisant une parallélisation à mémoire partagée et à mémoire distribuée. Avant d'exposer l'élaboration de la méthode d'adaptation de maillage dont cette thèse fait l'objet, il est nécessaire de présenter les principaux modules de FLUSEPA, avec lesquels le module AMR aura une interaction forte. La première partie de ce chapitre s'attache à décrire une partie de la physique modélisée au travers des équations de Navier-Stokes filtrées en temps ou en espace et le principe de la séparation d'échelles. Dans un second temps, nous effectuons une description de la méthode de résolution du type Volumes Finis utilisée et de la reconstruction d'ordre élevé des variables primitives, nécessaire à la construction des schémas numériques d'ordre élevé. Ensuite, nous présentons la méthode de chevauchement de maillage par intersection de FLUSEPA. Nous décrivons alors la partie la plus impactée par l'introduction d'une technique d'adaptation de maillage, à savoir la représentation des propriétés géométriques des faces et des cellules. Nous exposons par la suite la méthode d'intégration temporelle explicite adaptative et sa technique de répartition en classes temporelles pour l'avancement en temps. Enfin, nous définissons l'architecture logicielle de FLUSEPA pour une exécution sur des machines MPMD, ce qui permettra par la suite de bien situer l'intégration des différents modules de l'AMR.

### 2.1 Cadre

Flusepa a été développé pour la simulation des écoulements compressibles, 3D, instationnaires, visqueux et réactifs liés aux lanceurs spatiaux aéropulsés. Afin d'effectuer ces simulations, de nombreux modèles physiques ont été rendus disponibles et permettent par exemple, la simulation d'écoulements hypersoniques de gaz brûlés issus de propergols à poudre à la fois chargés en particules et avec réactions chimiques. Un schéma

---

1. Ce nombre de processeurs est celui utilisé pour les simulations « raides » avec un schéma temporel adaptatif explicite. La scalabilité du solveur implicite étant beaucoup plus grande.

récapitulatif des différents modules est présenté sur la figure 2.1. Toutefois pour simplifier la suite de l'étude nous ne modéliserons pas de réactions chimiques ni de composition particulière des gaz, sauf mention explicite.

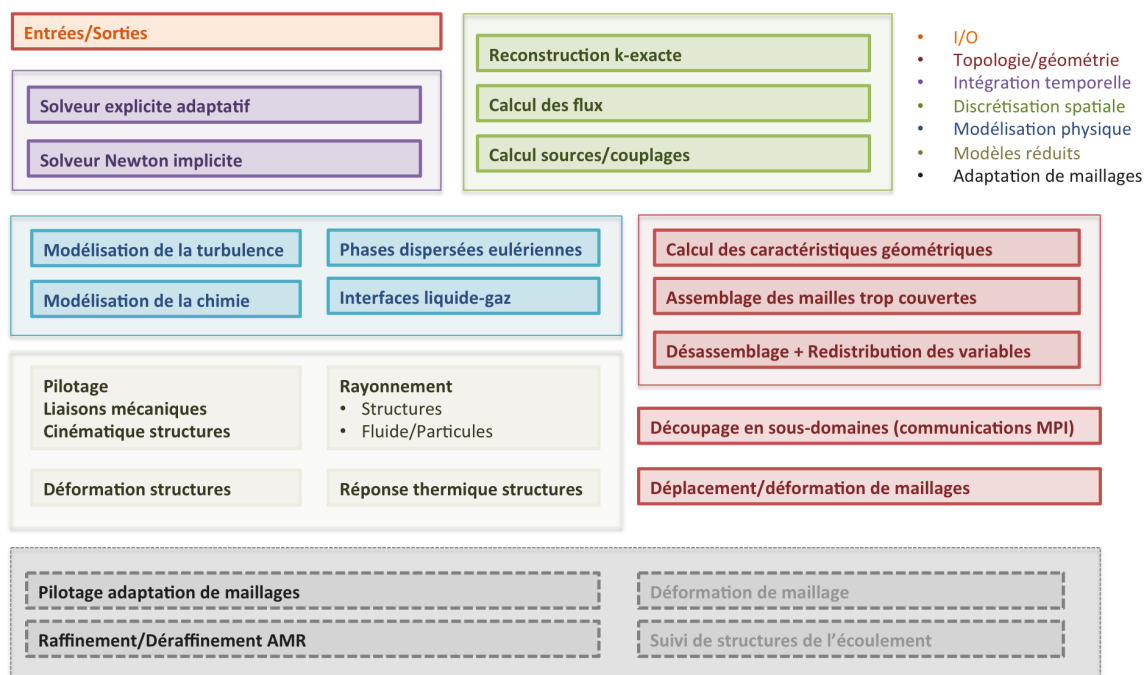


FIGURE 2.1 – Schéma descriptif des modules du programme FLUSEPA

### 2.1.1 Les équations de Navier-Stokes

Dans le cadre de cette thèse, nous nous limiterons aux équations de Navier-Stokes qui décrivent l'évolution d'un fluide visqueux, compressible, newtonien<sup>2</sup>, en l'absence de force extérieure. Sous forme conservative et en utilisant la convention de sommation des indices répétés d'Einstein, elles se mettent sous la forme :

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j} (\rho u_j) = 0 \quad (2.1)$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_j} (\rho u_i u_j) = -\frac{\partial P}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j}, \quad i = 1, 2, 3 \quad (2.2)$$

$$\frac{\partial \rho E}{\partial t} + \frac{\partial (\rho E + P) u_j}{\partial x_j} = -\frac{\partial q_j}{\partial x_j} + \frac{\partial \tau_{ij} u_i}{\partial x_j} \quad (2.3)$$

Ces équations traduisent respectivement la conservation de la masse, de la quantité de mouvement et de l'énergie. On note  $P$  la pression,  $E$  l'énergie totale,  $\mathbf{u} = (u_i)$ ,  $i=1,2,3$  le vecteur vitesse,  $\rho$  la masse volumique,  $\mathbf{q}$  le vecteur flux de chaleur et  $\tau$  représente le tenseur des contraintes visqueuses.

### 2.1.2 La notion de séparation d'échelle

Les équations de Navier-Stokes ont intrinsèquement une forte non-linéarité qui se traduit, lorsqu'un fluide est soumis à des contraintes de cisaillement, par l'apparition de grosses structures tourbillonnaires. Ces grosses structures vont ensuite transférer leur énergie selon un mécanisme de cascade énergétique à des structures plus petites qui elles-mêmes transféreront leur énergie à de plus petites échelles jusqu'à atteindre l'échelle de longueur

2. Un fluide est dit newtonien si sa loi contrainte – vitesse de déformation est linéaire. La constante de proportionnalité est appelée viscosité. On a alors en 1D une relation de la forme :  $\tau = \mu \frac{\partial u}{\partial y}$  avec  $\mu$  la viscosité dynamique du fluide,  $\tau$  la contrainte de cisaillement exercée sur le fluide,  $\frac{\partial u}{\partial y}$  le gradient de vitesse dans la direction perpendiculaire au cisaillement en  $s^{-1}$ .

caractéristique de Kolmogorov  $\eta$ . A partir de l'échelle de Kolmogorov, les tourbillons sont dissipés par viscosité sous forme de chaleur. En notant  $L$  la taille caractéristique du domaine à discrétiser on a, pour un écoulement turbulent, homogène, isotrope :

$$\frac{L}{\eta} = \mathcal{O}\left(\frac{\rho|\mathbf{u}|L}{\mu}\right)^{\frac{3}{4}} = \mathcal{O}\left(Re^{\frac{3}{4}}\right) \quad (2.4)$$

où  $Re$  est le nombre de Reynolds de l'écoulement considéré<sup>3</sup> et  $\mu$  est la viscosité dynamique du fluide. La complexité du problème 3D à résoudre devient, si le pas de discrétisation du domaine est de l'ordre de la longueur de Kolmogorov, de l'ordre de  $Re^{\frac{9}{4}}$ . Or, il est courant d'avoir pour des lanceurs spatiaux des nombres de Reynolds de l'ordre de  $10^8$ <sup>4</sup>. Ainsi, le coût d'une simulation qui résoudrait donc toutes les échelles de la turbulence jusqu'à celle de Kolmogorov est actuellement prohibitif et reste encore inaccessible. Afin de pouvoir simuler les écoulements rencontrés en aéronautique, différents types de modélisation ont été mises en place et permettent de prendre en compte les effets de la turbulence sans pour autant résoudre toutes les échelles qui la caractérisent.

Ces modèles ont pour principe de séparer différentes échelles de la turbulence et utilisent des filtres  $\bar{\bullet}$  qui permettent de décomposer le champ simulé sous la forme :

$$f(x, t) = \bar{f}(x, t) + f'(x, t) \quad (2.5)$$

où  $\bar{f}(x, t)$  est la partie résolue de l'écoulement et  $f'(x, t)$  est la partie modélisée.

En reprenant les équations de Navier-Stokes, on obtient pour la conservation de la masse :

$$\frac{\partial \bar{p}}{\partial t} + \frac{\partial}{\partial x_i}(\bar{\rho} \tilde{u}_i) = \mathcal{A} \quad (2.6)$$

où  $\tilde{\bullet}$  est la moyenne de Favre :

$$\tilde{f} = \frac{\bar{\rho} f}{\bar{\rho}}. \quad (2.7)$$

La conservation de la quantité de mouvement devient :

$$\frac{\partial \bar{\rho} \tilde{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\bar{\rho} \tilde{u}_i \tilde{u}_j) = -\frac{\partial \bar{p}}{\partial x_i} - \frac{\partial}{\partial x_i} \tilde{\tau}_{ij} + \frac{\partial}{\partial x_j} \widetilde{\tau}_{SGS} + \mathcal{B}, \quad (2.8)$$

où  $\bar{p}$  est la pression résolue et  $\tilde{\tau}_{ij}$  est le tenseur des contraintes visqueuses tel que :

$$\tilde{\tau}_{ij} = \bar{\mu} \left( \frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \frac{\partial \tilde{u}_k}{\partial x_k} \right), \quad (2.9)$$

et  $\tau_{SGS}$  est le tenseur des contraintes de sous-maille dont l'expression est :

$$\tau_{SGS} = \bar{\rho} (\widetilde{u_i u_j} - \tilde{u}_i \tilde{u}_j) \quad (2.10)$$

où  $\bar{\mu}$  est la viscosité dynamique filtrée,  $\delta_{ij}$  est le symbole de Kronecker. Les quantités  $\mathcal{A}$  et  $\mathcal{B}$  sont liées à la non-commutativité des opérateurs de dérivations et de filtrage. On notera bien que dans le cas général, il n'y a pas commutativité des opérateurs de dérivation et de filtrage.

### 2.1.3 La modélisation Reynolds Averaged Navier-Stokes (RANS)

En ingénierie, il suffit souvent d'obtenir les valeurs moyennes d'un écoulement (la portance, la trainée, la répartition de pression moyenne) pour concevoir un système. La méthode courante consiste à utiliser un

3. Le nombre de Reynolds est un nombre sans dimension, utilisé en mécanique des fluides qui permet de caractériser la nature du régime d'un écoulement.

4. En considérant les ordres de grandeurs suivant :  $L = 2\text{m}$ ,  $\rho = 1\text{kg.m}^{-3}$ ,  $|\mathbf{u}| = 100\text{m.s}^{-1}$ ,  $\mu = 2.10^{-5} \text{kg.m}^{-1}.\text{s}^{-1}$ . On obtient  $Re \simeq 1.10^8$ .

séparateur d'échelle basé sur ces valeurs moyennes. La moyenne d'ensemble est un opérateur général qui est basé sur l'hypothèse d'un nombre infini de simulations d'un même écoulement :

$$\bar{f}_0 = \lim_{N \rightarrow +\infty} \left[ \frac{1}{N} \sum_{n=0}^N f(x, t) \right]. \quad (2.11)$$

En pratique, effectuer un nombre infini de simulations est techniquement infaisable, il serait trop onéreux de simuler un grand nombre de fois le même écoulement. Cependant, pour un écoulement stationnaire il est possible d'utiliser le théorème d'ergodicité démontré par Birkhoff [Bir31] qui permet de passer d'une moyenne d'ensemble à une moyenne en temps, on a alors :

$$\bar{f}_T = \lim_{T \rightarrow +\infty} \left[ \frac{1}{T} \int_t^{t+T} f(x, t) \right]. \quad (2.12)$$

Pour que ce théorème soit applicable, la période  $T_1$  des oscillations doit être très faible devant la période d'intégration  $T$ , ce qui est évidemment vérifié pour un écoulement stationnaire comme présenté sur la figure 2.2(a). Dans le cas d'un écoulement instationnaire présenté sur la figure 2.2(b), on est amené à considérer plusieurs échelles de temps, on appelle toujours  $T_1$  l'échelle des oscillations rapides et  $T_2$  la période des oscillations lentes, la période d'intégration  $T$  doit alors vérifier :  $T_1 \ll T \ll T_2$ . Si l'hypothèse est valide, alors il est possible d'utiliser un modèle dit URANS<sup>5</sup> qui supporte un certain degré d'instationnarité dans l'écoulement.

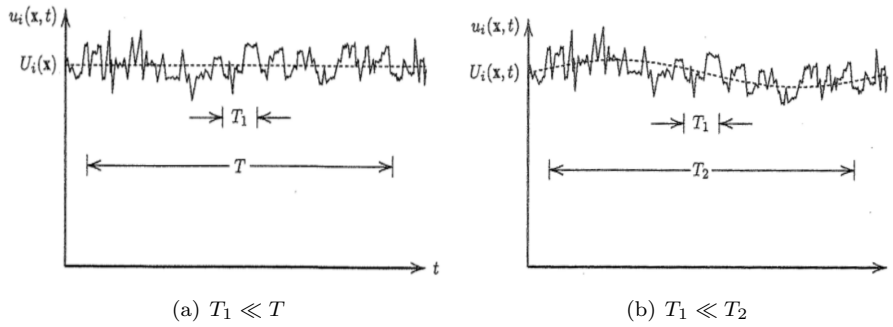


FIGURE 2.2 – Validité de l'hypothèse d'ergodicité [W<sup>+</sup>98]

Il est important de souligner que l'utilisation d'un modèle RANS n'introduit pas de termes croisés liés à la non-commutativité des opérateurs de filtrage et de dérivées, étant donné que ces opérateurs commutent dans ce cas précis. Par conséquent les termes  $\mathcal{A}$  et  $\mathcal{B}$  des équations 2.6 et 2.8 sont nuls.

#### 2.1.4 La simulation aux grandes échelles (LES)

La simulation aux grandes échelles est un bon compromis entre coût de calcul et précision en résolvant uniquement les grandes échelles de la turbulence et en modélisant les plus petites. Cette modélisation repose sur l'application, *via* un produit de convolution, d'un filtre  $G$  en espace que l'on définit de la façon suivante :

$$\bar{f}(x, t) = (G * f) = \int_0^{+\infty} \int_{\Omega} G(\bar{\Delta}(x, t), x - \xi, t - t') \cdot f(\xi, t') d\xi dt' \quad (2.13)$$

où  $\Omega \in \mathbb{R}^3$ . Dans le cas particulier où le filtre ne dépend pas du temps et est isotrope,  $\bar{\Delta}(x, t) = \bar{\Delta}$ , l'expression devient :

$$\bar{f}(x, t) = (G * f)(x, t) = \int_{\Omega} G(\bar{\Delta}, x - \xi) \cdot f(\xi, t) d\xi. \quad (2.14)$$

5. Pour *Unsteady Reynolds Averaged Navier-Stokes*

L'utilisation explicite d'un tel filtre est faite dans certains cas et plusieurs études ont été menées pour comparer l'utilisation d'un filtre implicite et l'application d'un filtre explicite [GC03]. Toutefois dans le cadre de cette thèse nous n'utiliserons que des filtres implicites autrement dit, on considérera que la limite de résolution est liée au maillage et au schéma numérique utilisés. En supposant que l'on ait un maillage isotrope de pas  $\Delta x$ , on résout au mieux des structures de tailles  $2 \times \Delta x$ <sup>6</sup>. Cependant l'influence du schéma numérique peut aussi s'apparenter à un filtrage de taille de coupure  $\Delta$ , et des études de résolubilité [Pon15] ont montré que le schéma numérique est prépondérant en terme de taille de coupure pour les modèles de turbulence. En pratique, on retiendra donc :  $\Delta > \max(\overline{\Delta}, \Delta x)$ .

Par différenciation de l'équation 2.14, on obtient :

$$\begin{aligned} \frac{\partial \overline{f}}{\partial x_i} &= \int_{\Omega} \frac{\partial G(\overline{\Delta}, x - \xi)}{\partial x_i} f(\xi, t) d\xi + \int_{\Omega} G(\overline{\Delta}, x - \xi) \frac{f(\xi, t)}{\partial x_i} d\xi \\ &= \int_{\Omega} \frac{\partial G(\overline{\Delta}, x - \xi)}{\partial x_i} f(\xi, t) d\xi + \frac{\partial \overline{f}}{\partial x_i}, \end{aligned}$$

il y a donc non commutativité du filtre sur l'opérateur de dérivée spatiale et l'on retiendra l'expression :

$$\frac{\overline{\partial f}}{\partial x_i} = \frac{\partial \overline{f}}{\partial x_i} - \frac{\partial \Delta}{\partial x_i} \int_{\Omega} \frac{\partial G(\overline{\Delta}, x - \xi)}{\partial \Delta} f(\xi, t) d\xi. \quad (2.15)$$

Pour pouvoir résoudre les équations 2.6 et 2.8, on suppose alors que l'on a :  $\frac{\partial \Delta}{\partial x_i} \simeq 0$  dans les zones où le pas du maillage varie de façon lente, ce qui permet de négliger les termes  $\mathcal{A}$  et  $\mathcal{B}$  de non-commutativité des opérateurs de filtrage et de dérivation, contenus dans les équations filtrées.

### 2.1.5 Les modèles Hybrides RANS/LES (HRL)

Une stratégie alternative aux deux modélisations présentées précédemment suppose de passer de façon continue d'une approche RANS à une approche LES par l'évaluation d'une échelle de longueur de coupure hybride. On présente sur la figure 2.3 la courbe type représentant la cascade d'énergie turbulente en fonction de la longueur d'onde des structures tourbillonnaires de l'écoulement. On peut distinguer trois grandes parties sur cette courbe en fonction du nombre d'onde des structures turbulentes :

- la zone de production qui fait apparaître les structures tourbillonnaires,
- la zone inertielle dans laquelle les grosses structures turbulentes transfèrent leur énergie à de plus petites structures,
- la zone de dissipation dans laquelle les structures sont dissipées sous forme de chaleur par la viscosité.

Les modèles hybrides RANS/LES s'adaptent aux propriétés intrinsèques tant de l'écoulement que du maillage. Dans une approche RANS/LES, le modèle RANS est appliqué sur tout le maillage : dans les zones peu résolues, le modèle reste en mode RANS et modélise toutes les échelles de la turbulence, par contre dans les parties résolues du maillage, le modèle bascule en modèle de sous-maille. Si l'on applique ce type de modèle de turbulence, toutes les structures tourbillonnaires ayant une fréquence inférieure à la fréquence de coupure du schéma numérique utilisé seront résolues et les autres structures seront correctement représentées par le modèle RANS qui se transforme ainsi en modèle de sous-maille. Ainsi, les modèles HRL représentent une solution viable d'un point de vue industriel, puisqu'ils permettent d'optimiser fortement le coût d'une simulation en appliquant uniquement une stratégie de type LES dans les zones où le maillage possède une résolution suffisante et des modèles de type RANS dans les zones peu raffinées. Des modèles de turbulence auto-adaptatifs permettent de compenser à la fois les défauts des modélisations RANS qui sont trop dissipatives mais aussi les défauts des modèles LES qui, pour être utilisés, nécessitent de résoudre jusqu'aux échelles dissipatives (isotropes) de Kolmogorov. En effet, les modèles de sous-maille utilisés pour la LES sont basés sur une taille caractéristique isotrope (racine cubique du volume pour le modèle de Smagorinski par exemple).

6. On utilise ici le théorème d'échantillonnage ou théorème de Nyquist-Shannon.



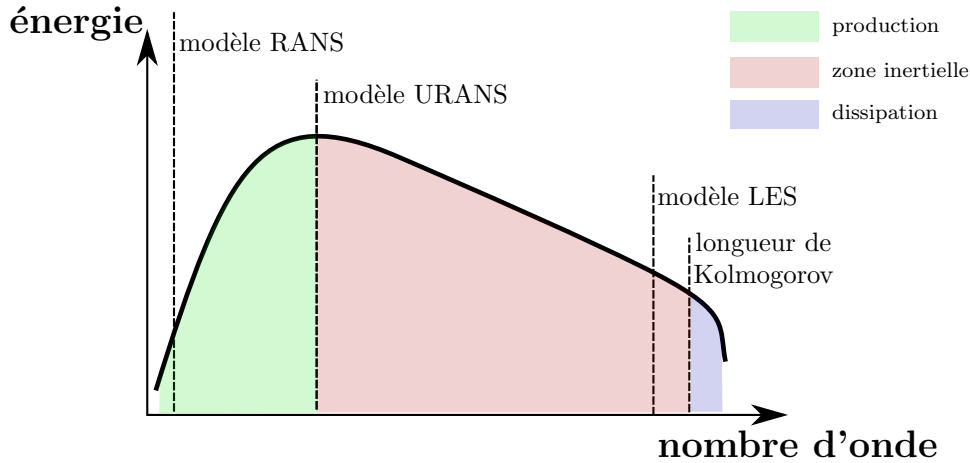


FIGURE 2.3 – Cascade d'énergie turbulente

Dans les simulations effectuées au cours de cette thèse, on utilise uniquement le modèle hybride RANS/LES *Delayed Detached Eddy Simulation* (DDES) [SDS<sup>+</sup>06, SSST08]. Il est largement utilisé pour les calculs de production avec FLUSEPA sur des configurations d'écoulements massivement décollés. Ce modèle est issu du modèle *Detached Eddy Simulation* (DES) introduit par Spalart *et al.* [SJS<sup>+</sup>97]. Le développement de ce modèle a été décrit dans la thèse de Pont [Pon15]. Ce modèle est similaire au modèle Zonal Detached Eddy Simulation mode II de Deck [Dec12].

## 2.2 Les Volumes Finis

Sous une forme intégrale, pour un domaine  $\Omega_i$ , de bord  $\Sigma_i$  fermé, les équations de Navier-Stokes se mettent sous la forme :

$$\frac{\partial}{\partial t} \iiint_{\Omega_i} \mathbf{w} dV = - \iint_{\Sigma_i} \mathbf{F} \cdot \mathbf{n}_i d\sigma_i, \forall \Omega_i \subset \Omega \quad (2.16)$$

où  $\mathbf{n}$  est la normale orientée vers l'extérieur du domaine,  $\mathbf{F}$  est le vecteur densité surfacique de flux et  $\mathbf{w} = (\rho, \rho \mathbf{u}, \rho E)$  est le vecteur des variables conservatives. La méthode des Volumes Finis repose sur le principe d'utiliser un ensemble fini de  $N$  volumes de contrôle  $\Omega_j$  dont l'intersection est soit une surface  $\mathcal{A}_{jk}$ , soit nulle, pour obtenir un domaine  $\Omega = \cup_{j=1}^N \Omega_j$  tel que :

$$\forall \{\Omega_j, \Omega_k\} / j \neq k, \Omega_j \cap \Omega_k = \begin{cases} 0 & , \text{ si } \Omega_j \text{ et } \Omega_k \text{ ne sont pas voisines} \\ \mathcal{A}_{jk} & , \text{ si } \Omega_j \text{ et } \Omega_k \text{ sont voisines} \end{cases} . \quad (2.17)$$

L'équation 2.16 est appliquée à une discrétisation en volumes de contrôle tous inclus dans  $\Omega$ . La solution du problème posé est obtenue en effectuant un bilan de flux sur l'ensemble des faces  $\{\mathcal{A}_{JK}, \forall J, K / J \neq K\}$ . La formulation des Volumes Finis est par essence conservative, ce qui signifie que toute masse, quantité de mouvement ou énergie sortant d'un volume de contrôle  $\Omega_I$  est reçue par un des autres volumes de contrôle ayant une face en commun avec lui (ou par une condition aux limites, s'il possède des faces de ce type).

La formulation Volumes Finis utilisée dans FLUSEPA repose sur l'approche de Godounov [God59, God76, GZI<sup>+</sup>76] pour la résolution des équations ainsi posées. Supposons l'existence d'un ensemble de  $N$  volumes de contrôle  $\{\Omega_I, I \in \llbracket 1, N \rrbracket\}$  qui forment une partition du domaine  $\Omega$ . Dans la formulation Volumes Finis les seules quantités connues dans chaque élément  $\Omega_I$  sont les valeurs intégrales des variables conservatives (masse, quantité de mouvement, énergie...). En supposant l'existence d'une fonction de répartition (appelée reconstruction)

propre à chaque variable considérée<sup>7</sup>, on peut obtenir au sein des mailles  $\Omega_I$  une valeur en tout point  $P \in \Omega_I$  des variables. En particulier, on peut obtenir une valeur sur les faces  $\mathcal{A}_{JK}$  des volumes de contrôle. On représente sur la figure 2.4 la situation rencontrée dans un cas mono-dimensionnel pour différents ordres de reconstruction des variables dans les cellule  $\Omega_J$  et  $\Omega_K$ .

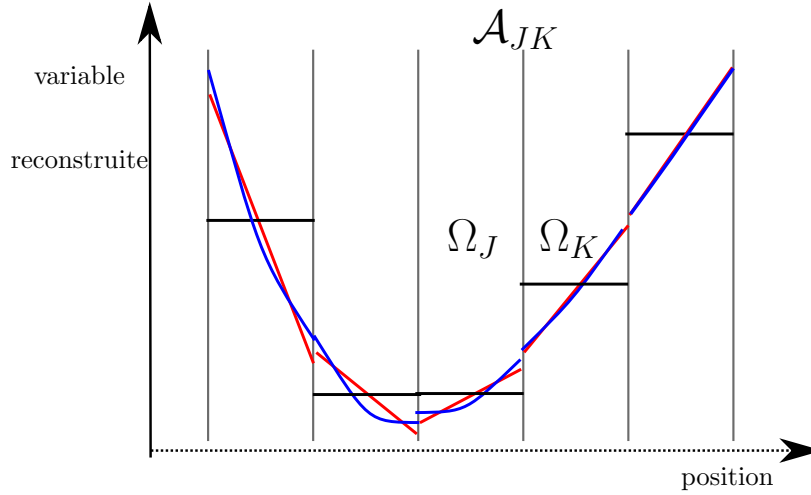


FIGURE 2.4 – Exemple de reconstructions par cellules et problèmes de Riemann associés. En noir : reconstruction constante, en rouge : reconstruction linéaire, en bleu : reconstruction parabolique.

L'utilisation d'une fonction de reconstruction introduit une discontinuité des champs sur chacune des faces du maillage. Pour une reconstruction constante par élément, on obtient un problème de Riemann<sup>8</sup> sur chaque face. La méthode de Godounov consiste à résoudre sur l'ensemble des faces tous les problèmes de Riemann posés en traitant tous les cas possibles *via* les relations de Rankine-Hugoniot pour les chocs et les invariants de Riemann pour les détentes.

Le détail de la méthode de reconstruction de FLUSEPA est donné au paragraphe 2.2.1.5. FLUSEPA utilise une formulation centrée aux éléments<sup>9</sup> et effectue une reconstruction affine d'ordre 2 (ou parabolique d'ordre 3) par élément puis utilise un solveur de Riemann sur les points d'intégration des flux des faces du maillage.

## 2.2.1 Propriétés géométriques et description des éléments hexaédriques manipulés

### 2.2.1.1 Définition d'un hexaèdre

On se place dans le contexte où les mailles sont toutes des hexaèdres, réguliers ou non, et pour plus de simplicité les schémas explicatifs s'appuieront sur des cubes dans cette partie. De façon classique, pour un maillage non structuré chaque élément est défini par un ensemble de nœuds et un ensemble de faces. On présente donc les différentes structures telles qu'elles sont définies dans FLUSEPA sur la figure 2.5.

Tout élément est défini par un ensemble de 8 nœuds présenté sur la figure 2.5(a) en rouge. À partir de cette numérotation locale des nœuds par élément, on définit les faces de l'élément en bleu sur la figure 2.5(a). Chaque face est définie par 4 nœuds et possède une numérotation locale, présentée en vert sur la figure 2.5(b). Pour calculer les propriétés géométriques des faces, chacune est divisée en deux triangles que l'on appelle « facette ». La découpe effectuée par face est unique, *ie* à chaque fois que les propriétés de cette face doivent être mises à jour (par intersection, déplacement, déformation), cette découpe est faite de la même façon avec à chaque fois

7. Dans la formulation originale de Godounov elles sont constantes dans chaque cellule.

8. Bien posé pour un problème mono-dimensionnel.

9. Autrement dit, les volumes de contrôles sont les mailles du maillage.

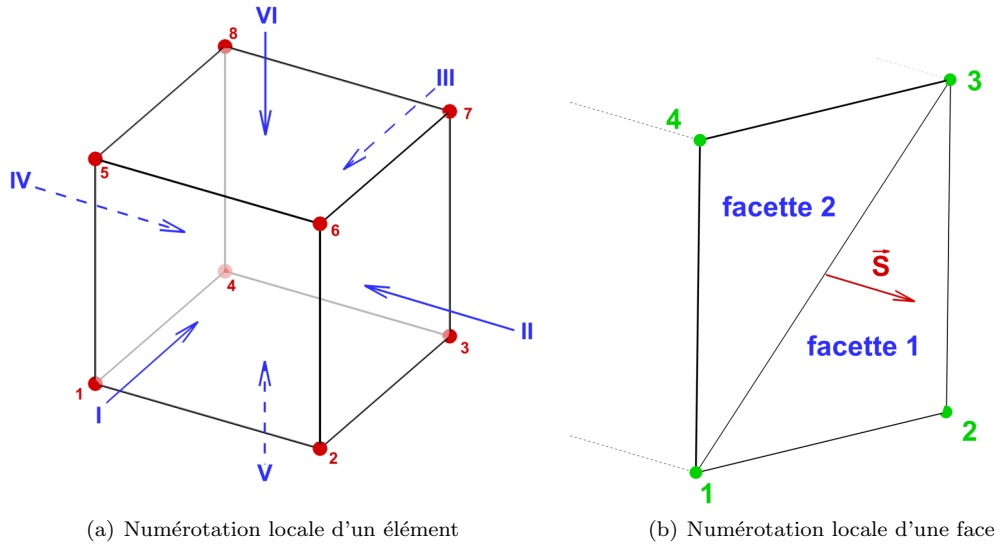


FIGURE 2.5 – Description d'un élément

le même nœud servant de référence. Ici, le nœud de référence est le nœud local 1, la découpe est présentée sur la figure 2.5(b) ; la facette 1 est composée des nœuds 1, 2 et 3 et la facette 2 des nœuds 1, 3 et 4. On précise l'orientation du vecteur surface  $\vec{S}$  en rouge.

Pour chaque face, on indique la correspondance entre la numérotation des nœuds des éléments représentés en rouge sur la figure 2.5(a) et la numérotation locale des nœuds des faces présentés en vert sur la figure 2.5(b) :

- face I : {1, 2, 6, 5}
- face II : {2, 3, 7, 6}
- face III : {4, 8, 7, 3}
- face IV : {1, 5, 8, 4}
- face V : {1, 4, 3, 2}
- face VI : {5, 6, 7, 8}

### 2.2.1.2 Notions mathématiques

Ce paragraphe présente les notions mathématiques de géométrie qui permettent de définir les propriétés des cellules et des faces. Nous nous appuyons ici sur les notations introduites par Brenner [Bre08] puis reprises Haider [Hai09] et Pont [Pon15]. On note indifféremment en gras les vecteurs. On travaille ici dans l'espace vectoriel  $\mathbb{R}^d$  muni de sa base canonique  $\{\mathbf{e}_1, \dots, \mathbf{e}_d\}$  et les coordonnées cartésiennes des différents objets manipulés sont notées  $\{x_1, \dots, x_d\}$ .

**Définition 1** (Tenseur). Soit  $\mathbf{a}^{(k)}$  un tenseur d'ordre  $k \geq 1$ , c'est une application linéaire d'ordre  $k$  telle que :

$$\mathbf{a}^{(k)} : \begin{cases} (\mathbb{R}^d)^k & \longrightarrow \mathbb{R} \\ (\mathbf{x}_1, \dots, \mathbf{x}_k) & \longmapsto \mathbf{a}^{(k)}(\mathbf{x}_1, \dots, \mathbf{x}_k) \end{cases} \quad (2.18)$$

et le tenseur  $\mathbf{a}^{(k)}$  sera défini par ses composantes :

$$a_{i_1 \dots i_k} = \mathbf{a}^{(k)}(\mathbf{e}_{i_1}, \dots, \mathbf{e}_{i_k}), \quad 1 \leq i_l \leq d, \quad 1 \leq l \leq k. \quad (2.19)$$

**Définition 2** (Produit tensoriel). Soit  $\mathbf{a}^{(k)}$  un tenseur d'ordre  $k$ , avec  $a_{i_1 \dots i_k}$  ses composantes. Soit  $\mathbf{b}^{(m)}$  un autre tenseur d'ordre  $m$  avec  $b_{i_1 \dots i_m}$  ses composantes, on définit leur produit tensoriel  $\mathbf{c}^{(m+k)} = \mathbf{a}^{(k)} \otimes \mathbf{b}^{(m)}$  par

ses composantes de la façon suivante :

$$c_{i_1 \dots i_{k+m}} = a_{i_1 \dots i_k} b_{i_{k+1} \dots i_{k+m}}. \quad (2.20)$$

Ainsi, le produit tensoriel de deux vecteurs est une matrice de composantes :

$$(\mathbf{x} \otimes \mathbf{y})_{ij} = x_i y_j. \quad (2.21)$$

Si  $\mathbf{a}$  est un vecteur, alors on note :

$$\mathbf{a}^{\otimes n} = \overbrace{\mathbf{a} \otimes \dots \otimes \mathbf{a}}^{n \text{ fois}},$$

qui est un tenseur symétrique d'ordre  $n$  dont les composantes sont :

$$\left( (\mathbf{a})^{\otimes n} \right)_{i_1 \dots i_n} = a_{i_1} \dots a_{i_n}. \quad (2.22)$$

**Définition 3** (Moment surfacique d'ordre  $k$ ). *On appelle moment surfacique d'ordre  $k$  de la surface  $\mathcal{S}_{\alpha\beta}$  au point  $\Gamma$ , le tenseur  $\mathcal{S}_{\alpha\beta}^{(k)} \Big|_{\Gamma}$  tel que :*

$$\mathcal{S}_{\alpha\beta}^{(k)} \Big|_{\Gamma} = \int_{\mathcal{S}_{\alpha\beta}} (\mathbf{x} - \mathbf{x}_{\Gamma})^{\otimes k} d\mathcal{S} = \int_{\mathcal{S}_{\alpha\beta}} \overbrace{(\mathbf{x} - \mathbf{x}_{\Gamma}) \otimes \dots \otimes (\mathbf{x} - \mathbf{x}_{\Gamma})}^{k \text{ fois}} d\mathcal{S} \quad (2.23)$$

où  $\Gamma$  est un point quelconque de la surface,  $d\mathcal{S}$  l'élément de surface pour l'intégration et :

$$\mathcal{S}_{\alpha\beta, i_1 \dots i_k}^{(k)} \Big|_{\Gamma} = \int_{\mathcal{S}_{\alpha\beta}} (x_{i_1} - x_{\Gamma, i_1}) \dots (x_{i_k} - x_{\Gamma, i_k}) d\mathcal{S}. \quad (2.24)$$

est un vecteur.<sup>10</sup>

**Définition 4** (Moment normalisé surfacique d'ordre  $k$ ). *On appelle moment normalisé surfacique d'ordre  $k$  de la surface  $\mathcal{S}_{\alpha\beta}$  au point  $\Gamma$ , le tenseur  $s_{\alpha\beta}^{(k)} \Big|_{\Gamma}$  tel que :*

$$s_{\alpha\beta}^{(k)} \Big|_{\Gamma} = \frac{\mathcal{S}_{\alpha\beta}^{(k)}}{|\mathcal{S}_{\alpha\beta}|} = \frac{\mathcal{S}_{\alpha\beta}^{(k)}}{|\mathcal{S}_{\alpha\beta}^{(0)}|} \quad (2.25)$$

où  $|\mathcal{S}_{\alpha\beta}|$  est la surface de  $\mathcal{S}_{\alpha\beta}$  et  $\mathcal{S}_{\alpha\beta}^{(0)}$  est le vecteur surface.

De la même façon, on définit :

**Définition 5** (Moment volumique d'ordre  $k$ ). *On appelle moment volumique d'ordre  $k$  de la cellule  $\mathcal{T}_{\alpha}$  au point  $\Gamma$ , le tenseur  $\mathcal{V}_{\alpha}^{(k)} \Big|_{\Gamma}$  tel que :*

$$\mathcal{V}_{\alpha}^{(k)} \Big|_{\Gamma} = \int_{\mathcal{T}_{\alpha}} (\mathbf{x} - \mathbf{x}_{\Gamma})^{\otimes k} d\mathcal{V} = \int_{\mathcal{T}_{\alpha}} (\mathbf{x} - \mathbf{x}_{\Gamma}) \otimes \dots \otimes (\mathbf{x} - \mathbf{x}_{\Gamma}) d\mathcal{V} \quad (2.26)$$

avec  $d\mathcal{V}$  l'élément de volume pour l'intégration et :

$$\mathcal{V}_{\alpha, i_1 \dots i_k}^{(k)} \Big|_{\Gamma} = \int_{\mathcal{T}_{\alpha}} (x_{i_1} - x_{\Gamma, i_1}) \dots (x_{i_k} - x_{\Gamma, i_k}) d\mathcal{V} \quad (2.27)$$

qui est un scalaire.

**Définition 6** (Moment volumique normalisé d'ordre  $k$ ). *On appelle moment volumique d'ordre  $k$  de la cellule*

10. En effet,  $d\mathcal{S} = n dS$  avec respectivement  $\mathbf{n}$  et  $dS$  le vecteur normal et la surface élémentaire au point  $\mathbf{x} \in \mathcal{S}_{\alpha\beta}$ .

$\mathcal{T}_\alpha$  au point  $\Gamma$ , le tenseur  $v_\alpha^{(k)} \Big|_\Gamma$  tel que :

$$v_\alpha^{(k)} \Big|_\Gamma = \frac{\mathcal{V}_\alpha^{(k)}}{|\mathcal{T}_\alpha|} = \frac{\mathcal{V}_\alpha^{(k)}}{\mathcal{V}_\alpha^{(0)}} \quad (2.28)$$

où  $|\mathcal{T}_\alpha| = \mathcal{V}_\alpha^{(0)}$  est le volume de  $\mathcal{T}_\alpha$

On utilisera le produit contracté noté " : " qui pour deux matrices  $A = (a_{ij})$  et  $B = (b_{ij})$  de même taille  $m \times n$  donne :

$$A : B = \sum_{i=1}^m \sum_{j=1}^n a_{ij} b_{ij}. \quad (2.29)$$

Dans le cas particulier où  $\mathbf{A} = (a_i)$  et  $\mathbf{B} = (b_i)$ ,  $i \in \llbracket 1, d \rrbracket$  sont des vecteurs, on utilisera le produit scalaire classique " . " :

$$\mathbf{A} \cdot \mathbf{B} = \sum_{i=1}^d a_i b_i. \quad (2.30)$$

### 2.2.1.3 Cas d'un assemblage des cellules

Il existe des situations où il est nécessaire d'assembler des cellules<sup>11</sup>, pour cela on introduit les relations sur les propriétés géométriques d'un agrégat de cellules  $\mathcal{T}_\alpha = \bigcup_{i=1}^N \mathcal{T}_{\alpha_i}$ . On a alors les relations suivantes :

$$|\mathcal{V}_\alpha| = \sum_{i=1}^N |\mathcal{V}_{\alpha_i}| \quad (2.31)$$

$$|\mathcal{V}_\alpha| (\mathbf{x}_{G_\alpha} - \mathbf{x}_{N_i}) = \sum_{i=1}^N |\mathcal{V}_{\alpha_i}| (\mathbf{x}_{G_{\alpha_i}} - \mathbf{x}_{N_i}) \quad (2.32)$$

$$\begin{aligned} \mathcal{V}_{\alpha, jk}^{(2)} \Big|_{N_1} &= \sum_{i=1}^N \mathcal{V}_{\alpha_i, jk}^{(2)} \Big|_{N_i} + \left( (\mathbf{x}_{N_i} - \mathbf{x}_{N_1})_j \mathcal{V}_{\alpha_i, k}^{(1)} \Big|_{N_i} + (\mathbf{x}_{N_i} - \mathbf{x}_{N_1})_k \mathcal{V}_{\alpha_i, j}^{(1)} \Big|_{N_i} \right) \\ &+ (\mathbf{x}_{N_i} - \mathbf{x}_{N_1})_j (\mathbf{x}_{N_i} - \mathbf{x}_{N_1})_k |\mathcal{V}_{\alpha_i}| \end{aligned} \quad (2.33)$$

### 2.2.1.4 Utilisation pratique des moments dans le cadre d'une stratégie numérique

Tous les objets géométriques introduits ici servent de socle à l'ensemble des méthodes numériques de FLU-SEPA, ils permettent de :

- calculer les propriétés géométriques des cellules après calcul des intersections,
- effectuer la reconstruction d'ordre élevé d'une variable dans une cellule comme cela doit être fait pour des méthodes de type *MUSCL*<sup>12</sup> et d'intégrer les flux à l'ordre élevé sur les faces,
- disposer de toutes les dérivées nécessaires par intégration des développements limités qui font apparaître ces moments sur les volumes et les faces des mailles manipulées.

### 2.2.1.5 Reconstruction d'ordre élevé

En utilisant les moments volumiques  $v_\alpha^{(k)} \Big|_{N_1}$  tels qu'ils sont définis dans la partie 2.2.1, définition 5, pour la cellule  $\mathcal{T}_\alpha$ , avec  $N_1$  un point de référence de cette cellule, on définit  $h$  comme étant une taille caractéristique de la maille considérée :

$$h = \sqrt{12 \times \max_{i \in \llbracket 1, 3 \rrbracket} \left( v_{\alpha, ii}^{(2)} \Big|_{N_1} \right)}. \quad (2.34)$$

11. Cf. paragraphe 2.3.4.

12. Pour *Monotonic Upstream-Centered Scheme for Conservation Laws*. Ces méthodes ont été introduites par Van Leer [VL79], ce sont des méthodes de reconstruction d'ordre élevé pour les Volumes Finis.

On rappelle les relations qui lient les variables primitives  $q$  et les variables conservatives  $w$  :

$$q = (P, T, \mathbf{u})^T = f(w), \quad (2.35)$$

soit pour un gaz parfait polytropique :

$$\begin{cases} P &= (\gamma - 1) \left( (\rho E) - \frac{1}{2} \frac{|\rho \mathbf{u}|^2}{\rho} \right) \\ T &= \frac{P}{\rho R} \\ \mathbf{u} &= \frac{\rho \mathbf{u}}{\rho} \end{cases} \quad (2.36)$$

avec les deux premières équations valables pour un gaz parfait,  $R$  la constante qui leur est associée et  $\gamma = \frac{C_p}{C_v}$  le coefficient de Laplace. On note la moyenne volumique :

$$\bar{q}_\alpha = \frac{1}{|\mathcal{T}_\alpha|} \iiint_{\mathcal{T}_\alpha} q dV. \quad (2.37)$$

Dans chaque cellule  $\mathcal{T}_\alpha$  on définit alors les variables tildées :

$$\tilde{q}_\alpha = f(\tilde{w}_\alpha), \quad \begin{cases} \tilde{P}_\alpha &= (\gamma - 1) \left( (\overline{\rho E})_\alpha - \frac{1}{2} \frac{|\overline{\rho \mathbf{u}}|_\alpha^2}{\bar{\rho}_\alpha} \right) \\ \tilde{T}_\alpha &= \frac{\tilde{P}_\alpha}{R \bar{\rho}_\alpha} \\ \tilde{\mathbf{u}}_\alpha &= \frac{(\overline{\rho \mathbf{u}})_\alpha}{\bar{\rho}_\alpha} \end{cases}. \quad (2.38)$$

Afin de pouvoir appliquer une méthode de type MUSCL, il est nécessaire d'établir une fonction de reconstruction  $\mathcal{R}$  telle que :

$$q|_{N \in \mathcal{T}_\alpha} = \mathcal{R}(\tilde{q}_\alpha) + \mathcal{O}(h^p). \quad (2.39)$$

ce qui permet d'avoir une évaluation en tout point du maillage des champs manipulés. L'ordre de la méthode est directement lié à l'ordre de précision  $p$  de la fonction de reconstruction.

Brenner dans [Bre08] décrit l'implémentation d'une méthode de reconstruction polynomiale d'ordre  $k$  dite «  $k$ -exacte » qui a ensuite été revue par Pont [Pon15] et par Limare *et al.* [LBB16]. Nous utilisons les opérateurs dérivée  $D^{(k)}$  qui sont des tenseurs d'ordre  $k$ . On suppose ces tenseurs comme étant précis à un ordre  $n$ . Si  $q$  est  $j$  fois continûment différentiable,  $D^{(j)}q$  est un tenseur symétrique d'ordre  $j$  de composantes :

$$\left( D^{(j)}q \right)_{i_1 \dots i_j} = \frac{\partial^j q}{\partial x_{i_1} \partial x_{i_2} \dots \partial x_{i_j}} \quad (2.40)$$

On obtient alors que le développement de Taylor de la fonction  $q$  à l'ordre  $k$  en  $\mathbf{x}_0$  est :

$$q(\mathbf{x}) = \sum_{j=0}^{k-1} \frac{1}{j!} D^{(j)}q \Big|_{\mathbf{x}_0} : (\mathbf{x} - \mathbf{x}_0)^{\otimes j} + \mathcal{O}(h^k), \quad (2.41)$$

avec la convention :

$$q(\mathbf{x}_0) = D^{(0)}q \Big|_{\mathbf{x}_0}. \quad (2.42)$$

Par intégration sur la cellule  $\mathcal{T}_\alpha$  de la relation 2.41 et en l'évaluant au centre de gravité  $G$  de la cellule, on obtient :

$$q(\mathbf{x}_G) = \bar{q}_\alpha - \sum_{m=1}^{n-1} \frac{1}{m!} v_\alpha^{(m)} \Big|_{\mathbf{x}_G} : D^{(m)}q \Big|_{\mathbf{x}_G} + \mathcal{O}(h^n). \quad (2.43)$$

Enfin, on cherche une relation de la forme :

$$\bar{q}_\alpha = \widetilde{q}_\alpha + \Delta q + \mathcal{O}(h^n), \quad (2.44)$$

que l'on obtient avec :

$$\widetilde{q}_\alpha = f(\bar{w}_\alpha) = f\left(w_\alpha + \sum_{m=2}^{n-1} \frac{1}{m!} v_\alpha^{(m)} \Big|_G : D^{(m)} w \Big|_G + \mathcal{O}(h^n)\right), \quad (2.45)$$

le calcul des  $\Delta q_i$  a été mené dans la thèse de Pont [Pon15]. En combinant les relations 2.41, 2.43 et 2.45, on obtient la relation cherchée. Ainsi, à partir des variables tildées, il est possible de reconstruire la valeur des grandeurs primitives en un point à l'ordre  $n$  souhaité pour obtenir le flux à l'ordre  $n$ . Le détail des relations pour l'ensemble des variables est donné dans l'annexe C du manuscrit. La reconstruction  $\mathcal{R}$  permet de définir un schéma spatial d'ordre  $n$  qui a été décrit par Pont [Pon15] puis Pont *et al.* [PBCR17].

## 2.2.2 Intégration temporelle adaptative

### 2.2.2.1 Descriptif

FLUSEPA utilise deux schémas temporels pour l'avancement en temps, l'un explicite et l'autre implicite. Le schéma implicite n'est pas décrit dans cette thèse qui utilise uniquement le schéma temporel explicite. Cette partie se focalise sur le schéma d'intégration temporelle explicite adaptatif qui s'apparente au schéma introduit par Kleb *et al.* [KBW92]. L'idée d'utiliser un pas de temps local pour chaque maille n'est pas récente, elle a été introduite notamment par Berger et Olinger [BO84] lors des premiers développements de l'AMR, pour ensuite être reprise par Osher et Sanders [OS83]. Plus récemment Müller *et al.* [MS07] ont introduit une méthode qui combine pas de temps local avec étape de remaillage dans la boucle d'intégration temporelle.

Chaque maille possède son propre pas de temps caractéristique pour un schéma numérique lié à la condition de stabilité de Courant Friedrichs Levy (CFL). Cette condition impose un pas de temps maximal admissible qui est directement proportionnel à la taille caractéristique de cette même maille pour les phénomènes de convection. Pour garantir la stabilité d'un schéma explicite, la condition  $CFL < 1$  est souvent utilisée sur l'ensemble des mailles du maillage. Cependant cette condition est plus restrictive pour certaines mailles que pour d'autres. Ainsi, si l'on impose le même pas de temps à toutes les mailles d'un maillage, donc celui lié à la plus petite cellule du maillage, on contraint très fortement certaines mailles de grandes tailles qui supporteraient *a priori* un plus grand pas de temps. Pour que le pas de temps effectif appliqué à chaque maille se rapproche du pas de temps que leur imposerait leur condition de CFL, l'idée est de définir localement différents pas de temps. On regroupe alors les mailles par tranche temporelle  $\theta_i$  en fonction du plus petit pas de temps  $\Delta t_{min}$  et de leur propre pas de temps admissible  $\Delta t_i$  tel que  $2^{\theta_i} \times \Delta t_{min} \leq \Delta t_i \leq 2^{\theta_i+1} \times \Delta t_{min}$ . Ces différents pas de temps sont alors de la forme  $\Delta 2_i^\theta \times t_{min}$ . Ainsi le CFL effectif est compris entre 0.5 et 1 pour chaque maille. Une description complète de l'algorithme du schéma temporel est donnée dans la thèse de Couteyen [CC16], nous en rappelons, ci-dessous, dans l'algorithme 1 simplement les grands principes.

En regroupant les mailles en  $\theta + 1$  niveaux temporels, numérotés entre 0 et  $\theta$ , on a alors les mailles du niveau  $\theta_i$  qui ont pour pas de temps  $2^{\theta_i} \times \Delta t_{min}$  où  $\Delta t_{min}$  est le plus petit pas de temps imposé par une maille sur tout le maillage. L'algorithme décrit le schéma d'avancement en temps utilisé dans FLUSEPA. Une itération de l'algorithme est composée de plusieurs sous-itérations.

A la différence des méthodes d'intégration temporelle étudiées dans le chapitre précédent, le schéma numérique de FLUSEPA, ne nécessite pas l'introduction de *ghost-cells* pour l'interpolation en temps, le schéma étant applicable pour des polyèdres généraux. Ainsi, les flux calculés sont consistants même lors des étapes d'interpolation et la conservativité de la méthode est préservée.

---

**Algorithme 1** Intégration temporelle adaptative de FLUSEPA

---

```

1: Calcul du pas de temps
2: Classification des cellules dans un niveau temporel
3: Boucle d'intégration temporelle adaptative :
4: pour sous-itération = 1 à  $\leq 2^\theta$  faire
5:    $\tau = 0$ 
6:   pour  $tmp = 1$  à  $\theta$  faire
7:     si ( $mod(\text{sous-itération}-1, 2^{tmp}) == 0$ ) alors
8:        $\tau = tmp$ 
9:     fin si
10:    fin pour
11:    si sous-itération  $\geq 1$  alors
12:      Correction des variables intensives ( $0 \dots \tau$ )
13:      Interpolation des variables intensives ( $\tau + 1$ )
14:    fin si
15:    Prédicteur
16:    Calcul des gradients ( $0 \dots \tau$ )
17:    Limitation et reconstruction des flux ( $0 \dots \tau$ )
18:    Repositionnement des flux ( $\tau + 1$ )
19:    Solveur de Riemann ( $0 \dots \tau$ )
20:    Sommation des flux sur les cellules ( $0 \dots \tau$ )
21:    si  $\tau \neq \theta$  alors
22:      Repositionnement des variables intensives ( $\tau + 1$ )
23:    fin si
24:    pour  $\tau' = \tau$  à 0 faire
25:      Fin du prédicteur pour  $\tau'$  :
26:      Prédiction des variables extensives ( $\tau'$ )
27:      Prédiction des variables intensives ( $\tau'$ )
28:      Correcteur :
29:      Calcul des gradients ( $\tau'$ )
30:      Limitation et reconstruction des flux ( $\tau'$ )
31:      Interpolation des flux ( $\tau'$ )
32:      Solveur de Riemann ( $\tau'$ )
33:      Sommation des flux sur les cellules ( $\tau'$ )
34:      Correction des variables extensives ( $\tau'$ )
35:      Correction des variables intensives ( $\tau'$ )
36:    fin pour
37: fin pour

```

---



### 2.2.2.2 Efficacité de la méthode et coût du calcul par maille

Une fois la répartition en niveaux temporels effectuée, on peut estimer le gain  $\mathcal{G}$  lié à l'utilisation d'une intégration temporelle adaptative. Dans un premier temps, on peut négliger les opérations d'interpolation inter-niveaux temporels. Si l'on note  $\Omega(\tau)$  l'ensemble des cellules d'un niveau temporel  $\tau$ ,  $|\Omega(\tau)|$  le nombre de cellules de ce niveau et  $\Omega$  l'ensemble des cellules du domaine, une bonne approximation du coût calculatoire d'un niveau temporel est :

$$C(\tau) \simeq 2^{\theta-\tau} \times |\Omega(\tau)|. \quad (2.46)$$

Le coût d'une itération complète du solveur est alors :

$$C(\Omega) \simeq \sum_{\tau=0}^{\theta} C(\tau), \quad (2.47)$$

On peut alors établir que le gain par rapport à une intégration temporelle avec un pas de temps uniforme, dont le coût serait  $2^\theta |\Omega|$ , est :

$$\mathcal{G} = \frac{2^\theta |\Omega|}{\sum_{\tau=0}^{\theta} C(\tau)}. \quad (2.48)$$

Cette estimation est une borne supérieure du gain. En pratique, il est nécessaire d'effectuer des interpolations et des repositionnements pour chacune des variables.

## 2.3 Le chevauchement de maillage

Nous détaillons longuement dans cette section les méthodologies de chevauchement classiques déjà exposées brièvement dans le chapitre 1 pour lesquelles il existe un très grand nombre de publications. Cette étude nous permet de correctement présenter la méthodologie de chevauchement de maillages de FLUSEPA et de bien distinguer les spécificités de celle-ci.

### 2.3.1 Rappels sur les méthodes de chevauchement de maillage

Les premiers développements d'une méthode de chevauchement de maillage dans le cadre de la CFD ont été mis en œuvre par Steger *et al.* [SDB83] pour la prise en compte de corps en mouvements relatifs. Du fait de la simplicité de la construction du maillage initial, ce type de méthode permet de simplifier la modélisation des géométries industrielles complexes. Elle permet par ailleurs d'aborder aisément, toujours dans un contexte industriel, la simulation de corps en mouvement relatifs comme par exemple les séparations d'étages de lanceur ou des simulations de voilures tournantes. On présente sur la figure 2.6(a) une vue éclatée des maillages utilisés pour la simulation d'un lanceur et sur la figure 2.6(b) une vue après assemblage de ces maillages. Les figures 2.7(a) et 2.7(b) présentent une simulation effectuée avec des maillages chevauchants pour différents types de séparations. Même si une méthode de chevauchement de maillage n'est utilisée que pour une configuration stationnaire, le chevauchement simplifie grandement la génération du maillage. Ainsi, le développement de ces méthodes est relativement actif depuis une trentaine d'années, à titre d'exemple on peut citer les travaux de Meakin [MS89, Mea93, Mea00], Lee [Lee08], Fidkowski [Fid07], Shenoy [SSP14] et Péron *et al.* [PBG<sup>+</sup>16]. Ces méthodes sont appliquées à la fois sur des maillages structurés et non structurés.

Les méthodes de chevauchement de maillage permettent de faire abstraction des contraintes classiques de construction de maillage en rendant indépendantes d'un point de vue géométrique des mailles qui sont voisines spatialement. Pour un maillage, le fait de raffiner les zones proches de la paroi peut entraîner un surcoût ailleurs, en général dans les zones où l'advection est prépondérante. Il existe plusieurs méthodologies permettant de construire des maillages exploitant le chevauchement de maillage. La méthode définie par Meakin [Mea93] est une des méthodes les plus utilisées. Le principe de la méthode de Meakin est d'avoir un maillage proche paroi (ou *near-body*) qui permet de résoudre les couches limites et d'un maillage loin de la paroi (*off-body*) qui assure le transport des grosses structures turbulentes de l'écoulement. On illustre sur les figures 2.6(a) et 2.6(b)

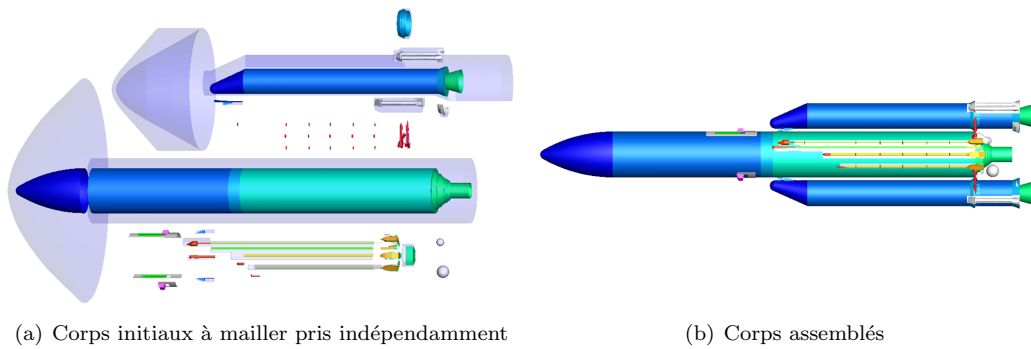


FIGURE 2.6 – Exemple d'un assemblage de maillage de FLUSEPA

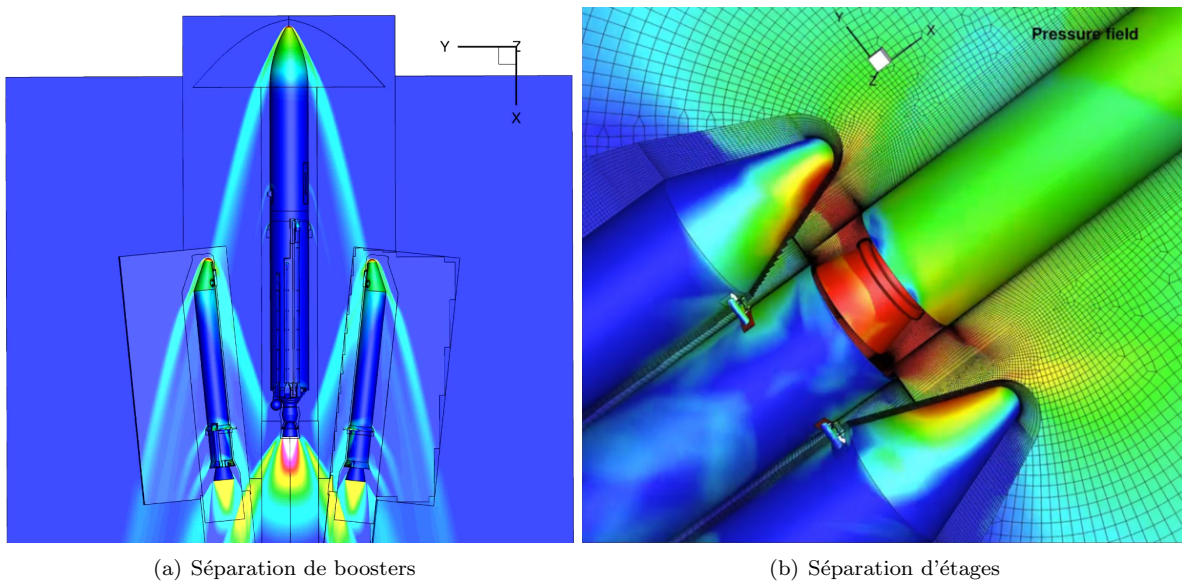


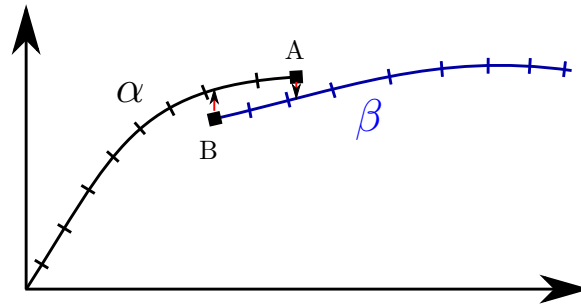
FIGURE 2.7 – Exemple de calculs de séparations effectués avec FLUSEPA

la technique de construction de maillage par assemblage des maillages liés à chacun des corps. La figure 2.7a présente un calcul avec les bords en noir des différentes grilles et la figure 2.7(b) montre un autre calcul avec cette fois les cellules utiles au calcul clairement représentées. Il est à noter que les zones proches des parois sont fortement raffinées dans la direction de la normale à la paroi. A partir d'une certaine distance de la paroi, la technique de chevauchement permet donc d'utiliser un maillage moins raffiné, ainsi il existe une discontinuité franche de résolution au niveau des intersections de grilles.

### 2.3.2 La stratégie de chevauchement classique *chimère*

La méthode *chimère* dite « classique » repose sur le principe d'un transfert d'information entre plusieurs intervalles ayant une intersection non nulle en utilisant des interpolations pour les cellules de bord. On présente un cas mono-dimensionnel sur la figure 2.8. Soit  $\alpha$  d'une part et  $\beta$  d'autre part, deux zones discrétisées dont le recouvrement spatial est non nul. Afin de faire circuler l'information d'une zone à l'autre, on fixe les valeurs des conditions aux limites de chaque zone en utilisant la valeur des champs sur un ensemble de points de l'autre zone appelée molécule. Dans notre cas A, est la cellule de bord de  $\alpha$  qui utilise les valeurs de points de  $\beta$  les plus proches de l'abscisse de A et B interpole les valeurs de points de  $\alpha$  dans les plus proches de l'abscisse de B. En pratique, un prétraitement doit être effectué pour chaque zone afin d'optimiser le recouvrement.

La figure 2.9 présente de façon très schématique, un cas bi-dimensionnel décrivant deux blocs discrétisés

FIGURE 2.8 – Zone d’interpolation *chimère* dans un cas mono-dimensionnel

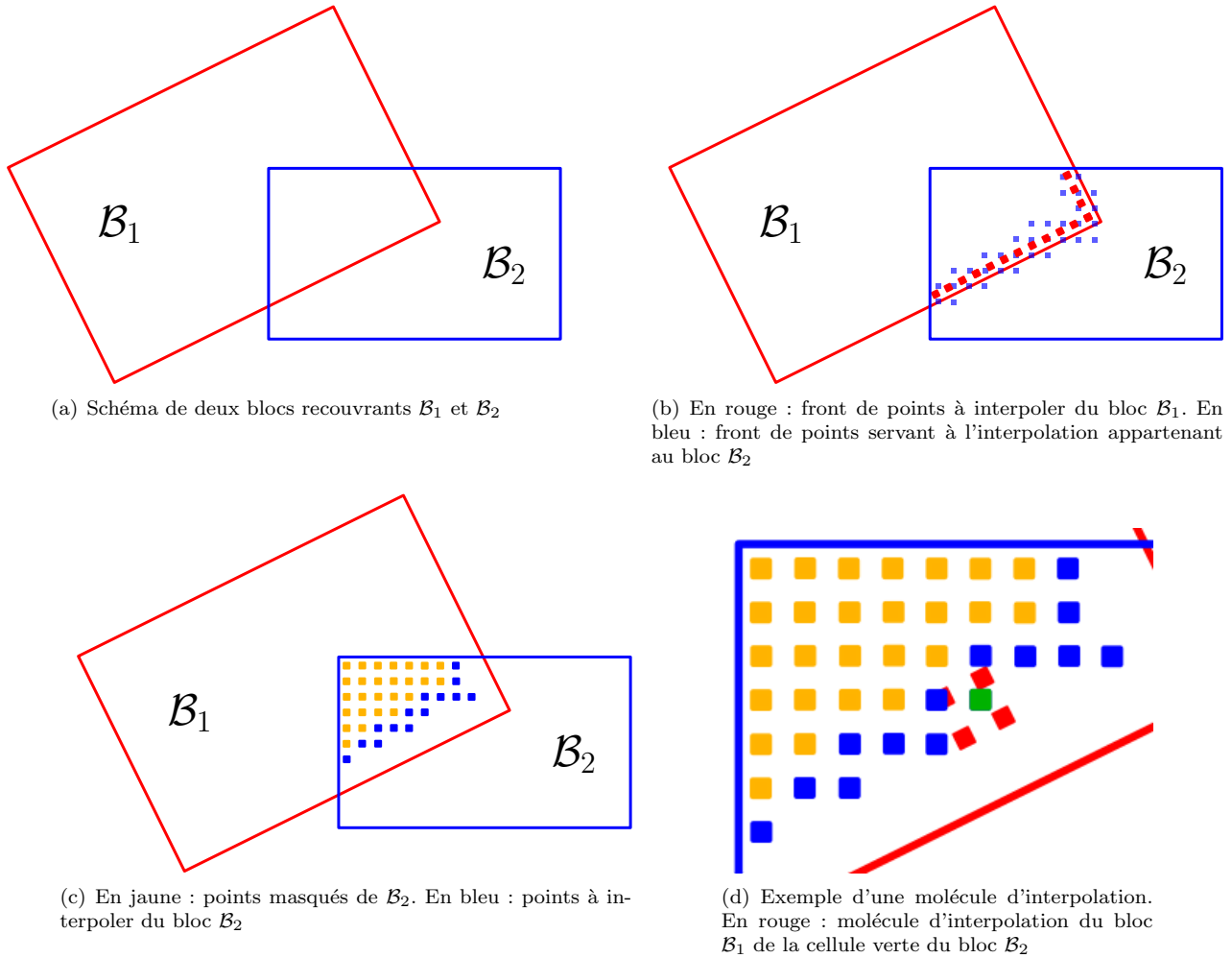
$\mathcal{B}_1$  et  $\mathcal{B}_2$  possédant un recouvrement non nul. On décrit dans chaque sous-figure uniquement certains sous-ensembles de cellules. On suppose que le bloc  $\mathcal{B}_1$  est prioritaire sur le bloc  $\mathcal{B}_2$ . Les cellules de bord de  $\mathcal{B}_1$  qui sont entièrement plongées dans  $\mathcal{B}_2$  sont alors repérées, et les cellules du bloc  $\mathcal{B}_2$  qui leur sont spatialement voisines vont servir pour effectuer l’interpolation qui assurera le transit d’information du bloc  $\mathcal{B}_2$  vers  $\mathcal{B}_1$ , ces cellules sont représentées en rouge sur la figure 2.9(b) et utilisent les informations des cellules en bleu. Une fois cette étape réalisée, on repère en jaune sur la figure 2.9(c) les cellules du bloc  $\mathcal{B}_2$  qui sont entièrement couvertes par le bloc  $\mathcal{B}_1$  et qui ne servent pas aux molécules d’interpolation des cellules de  $\mathcal{B}_1$ . Dans ces cellules complètement couvertes du bloc  $\mathcal{B}_2$ , un front de cellules, figure 2.9(c) en bleu est alors repéré, ce front doit utiliser les informations des cellules de  $\mathcal{B}_1$ . Un masquage des cellules complètement recouvertes qui ne sont pas dans ce front est alors effectué ainsi, elles ne serviront plus au calcul. Enfin on présente sur la figure 2.9(d) une molécule en rouge servant à l’interpolation d’une cellule appartenant à  $\mathcal{B}_2$  que l’on a représentée en vert, on notera que cette molécule ne comprend aucune des cellules devant être interpolées dans le bloc  $\mathcal{B}_1$ .

Les cellules contenues dans les maillages doivent être marquées et repérées selon leur type. En définissant un système de priorité tel que les cellules de plus haute priorité masquent les cellules de priorité moindre, on peut regrouper en 3 catégories les cellules du maillage :

- Zone 0 : les cellules qui ne recouvrent aucune autre cellule
- Zone 1 : les cellules qui sont totalement couvertes et qui ne participent plus au calcul (en bleu sur la figure 2.9(c))
- Zone 2 : zone de couplage interdomaine (cellules rouge sur la figure 2.9(b) et bleu sur les figures 2.9(c) et 2.9(d))

La méthode *chimère* repose principalement sur l’utilisation d’une zone de couplage interdomaine. Autour des cellules masquées de la zone 1, un front de points interpolés est défini et permet d’effectuer le couplage entre les différentes zones du maillage. Les cellules interpolées sont couramment appelées *fringe cells*. La zone 2 contient les cellules qui forment le  $n^{\text{ème}}$  front de cellules depuis la surface d’intersection. Ces cellules permettent à l’information de circuler entre les différents maillages. On associe à chacune des cellules devant être interpolée une molécule de cellules dans un des autres maillage qui lui sont voisines spatialement. Cette molécule est en général fournie par un module externe qui dialogue avec le programme de calcul (les plus connues sont PEGASUS [RSD03], SUGGAR++ [NBKC09], OVERTURE [BCHQ97], pour une liste plus complète, se référer à la thèse de Lee [Lee08]). Le choix de la technique d’interpolation reste aujourd’hui une question ouverte, à ce titre, on peut citer les travaux de Lee [Lee08], de Chicheportiche et Gloerfeldt [CG12], de Legouez [LG16] et de Ramirez *et al.* [RNO<sup>+</sup>17]. L’interpolation est intrinsèquement non conservative, toutefois des stratégies permettant de minimiser l’erreur sont mises en place pour les écoulements dans lesquels la conservation de la masse est importante. En pratique, cela se traduit par des méthodes de génération de maillages assez strictes de sorte qu’il y ait continuité de la résolution au niveau des zones d’interpolation. Cette restriction est la principale contrainte d’un point de vue industriel pour la génération de maillage.

La méthode d’interpolation impose par nature d’avoir une quasi-continuité de la résolution spatiale au ni-

FIGURE 2.9 – Schéma de principe de la méthode *chimère*

veau de la zone de couplage. En effet, les schémas numériques utilisés nécessitent que chaque cellule ait une molécule d'interpolation suffisante dans les zones de recouvrement pour que les valeurs des différents champs puissent être calculés. Dans le cas où il existe une sous-résolution d'un maillage pour un autre, des cellules qui doivent être interpolées mais qui ne peuvent pas l'être, apparaissent. On parle de cellule « orpheline » lorsque l'on ne peut pas définir de cellules d'une autre grille pour l'interpolation d'une cellule qui doit être interpolée. La recherche de ce type de cellules et les traitements effectués amènent localement à abaisser l'ordre de précision de la méthode globale. Ainsi, les méthodes par interpolation deviennent complexes voire impossibles dans le cas où deux frontières solides entrent en contact, il y a alors présence de nombreux points orphelins. Un exemple est donné figure 2.10, deux maillages  $\mathcal{M}_1$  (en bleu) et  $\mathcal{M}_2$  (en noir) se chevauchent et possèdent chacun une paroi. Le maillage  $\mathcal{M}_1$  est en mouvement relatif par rapport à  $\mathcal{M}_2$ . Les cellules en rouge de  $\mathcal{M}_2$  vont être recouvertes par la paroi de  $\mathcal{M}_2$ , il n'y aura alors plus de cellules utilisables pour les *fringe cells* de  $\mathcal{M}_1$  et création de cellules orphelines dans  $\mathcal{M}_1$ . Différentes méthodes existent et permettent de prendre en compte ces situations, cependant cela ne rentre pas dans le périmètre de cette thèse.

Un autre défaut de la méthode *chimère* apparaît pour les écoulements contenant des discontinuités, l'interpolation dans les *fringe cells* peut s'avérer complexe et conduire à de graves pertes de conservativité qui impactent la précision de la méthode<sup>13</sup>.

13. Malgré ces problèmes, la méthode *chimère* est utilisée pour des simulations des lanceurs spatiaux, comme le montrent les travaux de l'équipe du code Launch, Ascent and Vehicle Aerodynamics (LAVA) [KHB<sup>+</sup>16, BBHK15, BHK16, SBK14, MYBS<sup>+</sup>12].

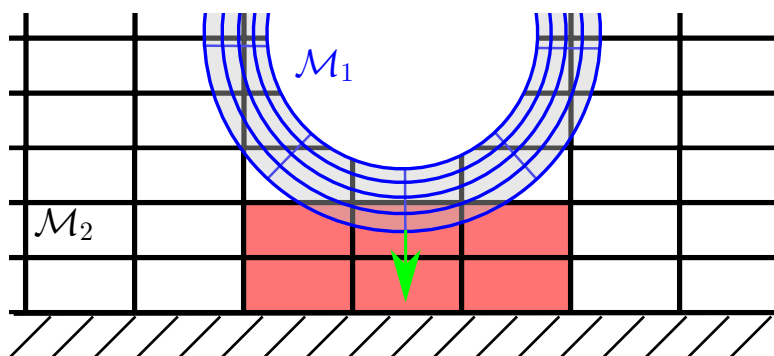


FIGURE 2.10 – Exemple d’un chevauchement de deux maillages  $\mathcal{M}_1$  et  $\mathcal{M}_2$  avec pour chacun une paroi associée. En rouge : cellule qui vont être couvertes par mouvement de  $\mathcal{M}_1$ .

### 2.3.3 Techniques palliant la non-conservativité des méthodes de chevauchement de maillages classiques

Dans le cadre des méthodes *chimère* classiques, du fait du chevauchement, certaines surfaces sont comptées en double lors de l’intégration de la pression. Des méthodes dites *zipper grids* ou encore *DRAGON*<sup>14</sup> de remaillage ont été développées afin de garantir la conservativité et ainsi pouvoir calculer correctement les forces et les moments sur les différents corps [CB95, Cha09a]. Contrairement à des méthodes de chevauchement classiques, le remaillage *DRAGON* assure un recouvrement nul entre les grilles (les mailles chevauchantes sont retirées du calcul) à partir d’une triangulation inter-maillage. Les nouvelles faces construites permettent d’effectuer un calcul des flux (voir figure 2.11(c)). Cependant, la qualité des maillages utilisés dans les zones remaillées est souvent mauvaise et, si les maillages sont en mouvement relatif, alors une interpolation (non-conservative) est nécessaire pour affecter des valeurs aux nouvelles mailles.

Un module permettant d’effectuer des opérations booléennes (union, intersection, ...) sur des maillages a été développé par Landier [Lan15, Lan17]. Cette méthodologie effectue une triangulation dans les zones d’intersection de grilles, cependant elle ne semble pas être applicable pour l’instant dans un contexte industriel avec mouvement de maillage. Par exemple, pour le cas présenté sur la figure 2.12 le temps de calcul de l’intersection est de l’ordre de 30s pour un nombre de  $1 \times 10^5$  triangles intersectés avec simplement deux niveaux de priorité, ce qui est au moins un ordre de grandeur trop grand pour nos applications qui nécessitent de recalculer l’intersection des maillages à chaque itération.

Il existe aujourd’hui quelques méthodes *chimère* pour lesquelles l’interpolation peut être conservative ; on peut citer [ML89, Rai86] pour des formulations conservatives en 2D. De façon plus récente en 3D, Alauzet a obtenu une formulation  $P_1$ -conservative pour des tétraèdres [Ala16] par intersection de maillages. Sa méthode se rapproche de celle de Landier et ne considère que les intersections tétraèdre-tétraèdre en définissant une librairie des cas possibles par arêtes. Les polyèdres convexes obtenus par intersection sont alors définis comme de nouveaux éléments pouvant être utilisés par le solveur. Une attention particulière a été portée à ce que les algorithmes d’intersection définis dans ce papier soient déterministes. Les cas de polyèdres dégénérés possibles (faces ou arêtes coincidentes) sont traités par perturbation de la position des nœuds. La méthode de projection est conservative sur la masse, n’introduit pas de maxima et est 1-exacte dans le sens où si la variable à projeter est une fonction affine alors l’opérateur de projection renvoie la fonction affine. Cette méthode de projection a été comparée à une méthode projection 1-exacte non conservative, sans intersection dans un cas d’onde de souffle sphérique dans une ville. Les résultats du calcul en comparaison avec un calcul très raffiné de référence sont présentés sur la figure 2.13. Le coût du calcul avec conservation de la masse est doublé cependant l’erreur finale du calcul gagne un ordre de grandeur. Il existe donc un réel intérêt à utiliser des méthodes conservatives pour le chevauchement de maillage.

14. On peut aussi citer les méthodes *arlequin* [DR05] utilisées en France pour la mécanique des solides. Ces méthodes sont fortement apparentées aux stratégies *zipper grids* et établissent une triangulation des intersections géométriques.

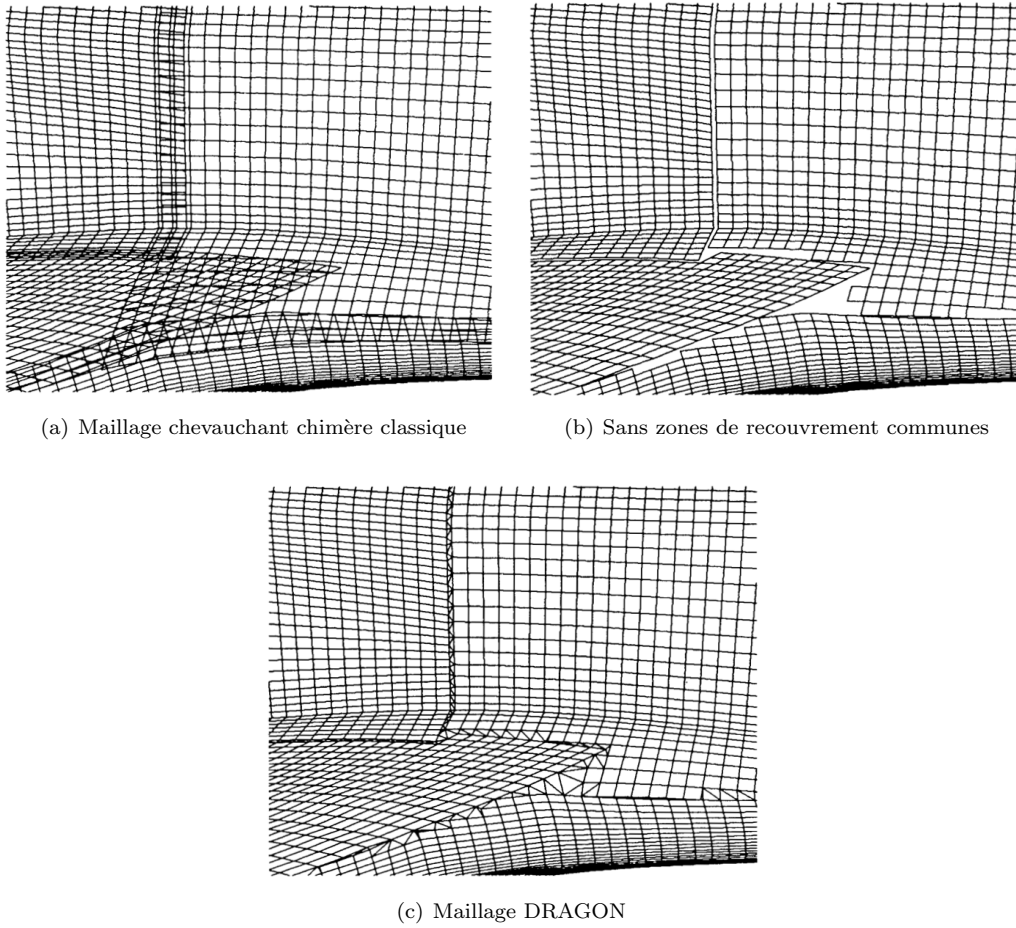
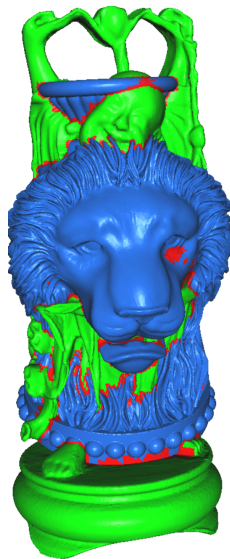
FIGURE 2.11 – Méthodologie *DRAGON* (extrait de Chan et Buning [CB95])

FIGURE 2.12 – Intersection d'un bouddha et d'un lion [Lan17]

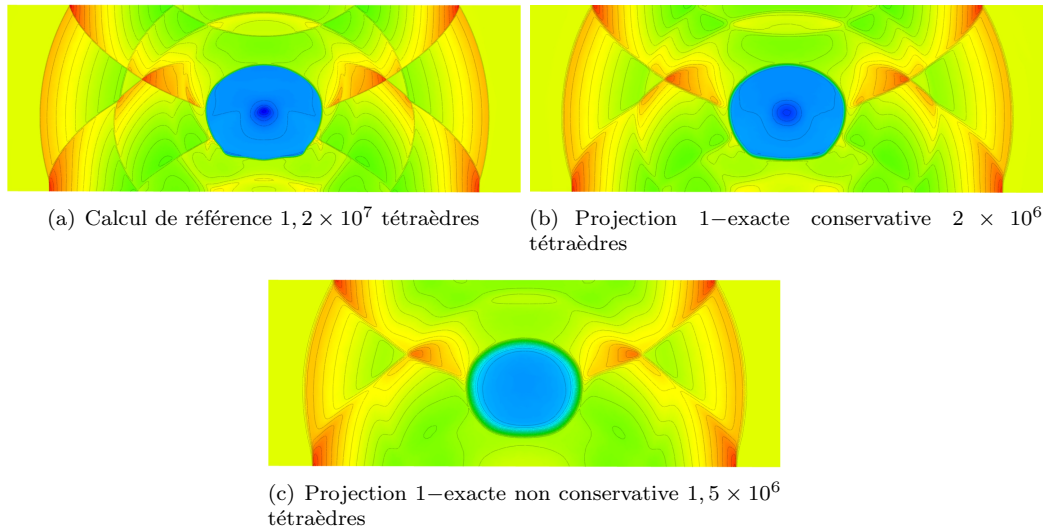


FIGURE 2.13 – Comparaison des résultats d’un calcul d’onde de souffle dans une ville (extrait de Alauzet [Ala16])

### 2.3.4 La méthode d’intersection de maillage de Brenner

Partant du constat que l’interpolation est problématique pour la conservation des variables dans les méthodes classiques, Pollet et Brenner [PB89] puis Brenner [Bre91, Bre96] ont développé une méthode de chevauchement de maillage pour les calculs de corps en mouvements relatifs lors des séparations d’étages. La communication inter-maillages s’appuie sur la surface d’intersection géométrique exacte des différents maillages.

#### 2.3.4.1 Cas du chevauchement de deux maillages

Avant de présenter le cas général des intersections, présentons un cas simple (figure 2.14) : considérons qu’un maillage  $\mathcal{M}_2$  de priorité  $P_2$  se comporte comme un masque qui se déplace et recouvre partiellement un autre maillage  $\mathcal{M}_1$  de priorité  $P_1$ , avec  $P_1 < P_2$ . L’interface entre les maillages est créée naturellement : il s’agit de la surface d’intersection formée par la bordure de l’évidement dans le maillage  $\mathcal{M}_1$  du volume occupé par le masque  $\mathcal{M}_2$ .

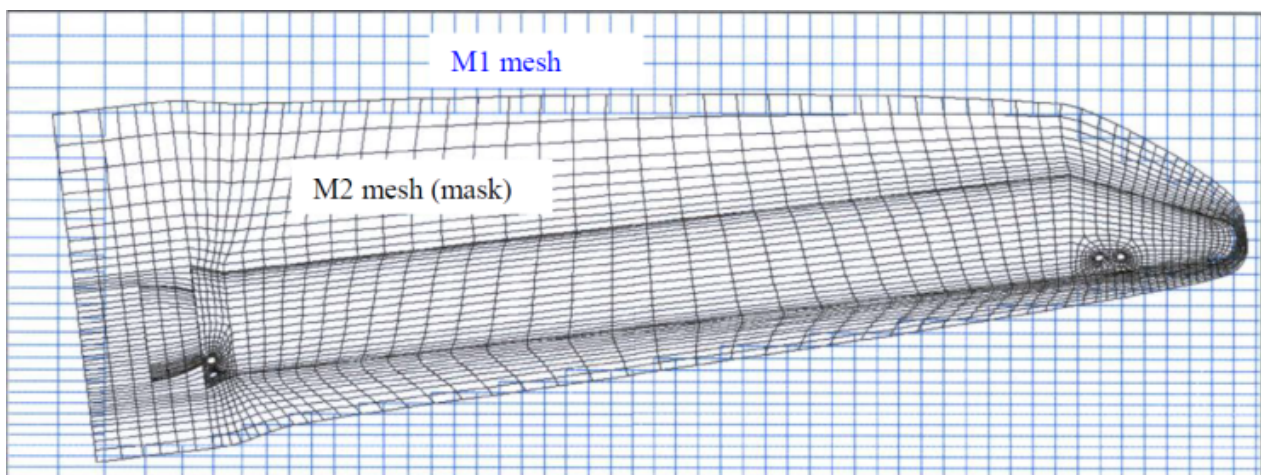


FIGURE 2.14 – Chevauchement de deux maillages  $\mathcal{M}_1$  et  $\mathcal{M}_2$

Les mailles de  $\mathcal{M}_2$  ne subissent donc aucune modification. Par contre, dans le maillage  $\mathcal{M}_1$  trois types de mailles caractérisées par leur niveau de couverture (figure 2.15) sont présentes :

- les mailles totalement couvertes
- les mailles partiellement couvertes
- les mailles totalement découvertes

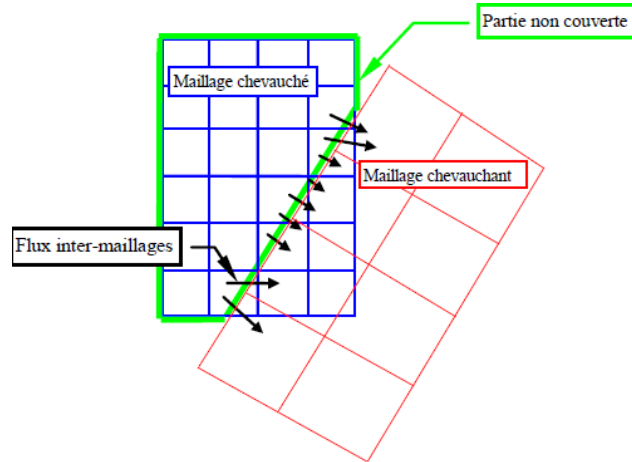


FIGURE 2.15 – Cas courant d’une intersection de deux maillages en 2D

Les mailles de la deuxième catégorie sont donc topologiquement modifiées car une partie des faces qui les constituent est couverte et de nouvelles faces correspondant à la limite externe du masque sont créées et participent à leur fermeture. Ces nouvelles faces forment l’interface entre les mailles de  $\mathcal{M}_1$  coupées et les mailles de  $\mathcal{M}_2$  non modifiées.

### 2.3.4.2 Cadre général

FLUSEPA manipule plusieurs maillages de priorités différentes qui se recouvrent les uns les autres, on parle de multi-chevauchement. Chaque maillage est en général lié à un corps à étudier et se déplace avec celui-ci. Des maillages supplémentaires sont utilisés comme grilles de fond auxquels sont affectés des priorités faibles de sorte qu’ils soient masqués par tous les autres maillages. Cette technique permet l’étude de corps mobiles en mouvements relatifs. L’intersection géométrique en trois dimensions est calculée pour assurer l’insertion et l’incrustation des maillages de priorité élevée dans ceux de priorité inférieure. Ce calcul d’intersection est effectué via des polyèdres fermés<sup>15</sup> qui s’appuient sur les maillages surfaciques et/ou volumiques fournis à FLUSEPA.

Le concept de multi-chevauchement de maillage nécessite d’avoir un ensemble de surfaces fermées auxquelles on a affecté une priorité. Ces surfaces sont :

1. soit liées à des mailles volumiques et dans ce cas elles peuvent être de trois types :
  - les conditions limites fluides intersectantes : ce sont les éléments constitutifs des polyèdres d’intersection servant au calcul des intersections ;
  - les faces internes à une grille dans laquelle coexistent des mailles de priorités différentes ;
  - les parois de priorité différentes de celles des mailles auxquelles elles sont liées,
2. soit ce sont des surfaces qui ne sont liées à aucune maille : les « coques ». Dans ce cas, la méthode s’apparente dans son principe à une méthode de type *cut-cell* telle qu’elle est définie par exemple, dans les références suivantes [PB79, BC89, YCI00, CKCM01, Fid07] .

La méthode d’intersection évide un maillage volumique, les propriétés géométriques des cellules coupées sont alors mises à jour pour prendre en compte la présence de la surface créée par l’ensemble des coques. La surface ainsi définie est indépendante d’un maillage volumique.

À l’aide de ces outils, FLUSEPA effectue des calculs d’intersection avec les règles suivantes :

15. En effet, chaque polyèdre utilisé pour le calcul d’intersection doit vérifier la relation de fermeture  $\iint_S \mathbf{n} \cdot \mathbf{dS} = \mathbf{0}$ .



- chaque maille de moindre priorité peut être intersectée par les « polyèdres intersectants » de priorité supérieure ;
- chaque maille partiellement couverte (donc coupée) est fermée par la fraction des polyèdres intersectants de priorité supérieure qui doit être incluse dans cette maille, tout en étant non couverte (par des polyèdres de priorité encore supérieure). Cette partie de polyèdre deviendra soit une condition aux limites pariétales (parois prioritaires et coques), soit une interface entre mailles de priorités différentes (*conditions aux limites fluides intersectantes* si les mailles appartiennent à deux maillages différents).

La figure 2.16 schématise un cas d'intersection entre 3 cellules  $\mathcal{C}_1$ ,  $\mathcal{C}_2$  et  $\mathcal{C}_3$  respectivement de priorités  $\mathcal{P}_1$ ,  $\mathcal{P}_2$  et  $\mathcal{P}_3$  et une surface  $\mathcal{S}$  de priorité  $\mathcal{P}_4$  tel que  $\mathcal{P}_4 > \mathcal{P}_3 > \mathcal{P}_2 > \mathcal{P}_1$ . La cellule  $\mathcal{C}_2$  de priorité  $\mathcal{P}_2$  coupe  $\mathcal{C}_1$ , mais  $\mathcal{C}_2$  est elle-même couverte par la cellule  $\mathcal{C}_3$  de priorité  $\mathcal{P}_3$ . Enfin la cellule  $\mathcal{C}_3$  est elle-même couverte par le polyèdre formé par la surface  $\mathcal{S}$  de priorité  $\mathcal{P}_4$ . La surface de  $\mathcal{C}_1$  est formée alors par l'union de ces différentes surfaces d'intersection. La surface extérieure résultante de la cellule  $\mathcal{C}_1$  est représentée en rouge.

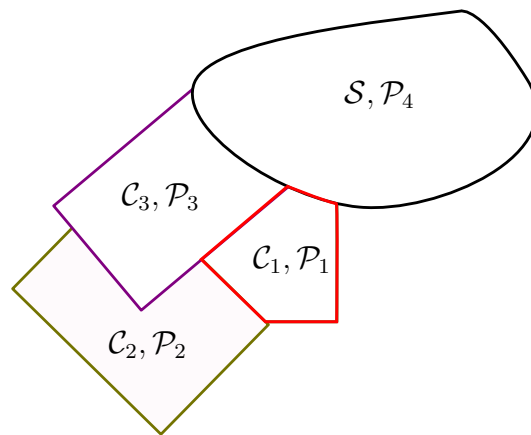


FIGURE 2.16 – Exemple de couverture d'une cellule  $\mathcal{C}_1$  en multichevauchement

L'incrustation géométrique des maillages de priorité supérieure dans ceux de moindre priorité nécessite impérativement que les polyèdres d'intersection forment des surfaces fermées ayant une orientation intérieure/extérieure *univoque*, autrement dit, les différentes faces constituant un même polyèdre ne doivent pas s'interpénétrer.

### 2.3.4.3 Calcul des intersections

Nous présentons dans ce paragraphe les grandes étapes du calcul d'intersection. Pour simplifier le problème géométrique, on considère toutes les faces des éléments comme étant planes. Les faces quadrangulaires sont traitées comme deux faces triangulaires, un hexaèdre devient alors un dodécaèdre comme présenté sur la figure 2.17. On appellera « facettes » les demi-faces ainsi formées.

L'algorithme de calcul se résume à deux étapes :

1. Déterminer la surface couverte de chaque facette d'un maillage par le maillage des polyèdres de priorité supérieure.
2. Déterminer la partie de chaque polyèdre intersectant de priorité supérieure qui ferme chacune des mailles coupées du maillage de priorité inférieure et assure le transfert des flux inter-maillage ou bien forme de nouvelles parois.

Ces deux étapes sont en fait identiques d'un point de vue algorithmique, elles consistent à déterminer la surface d'une face contenue dans un polyèdre<sup>16</sup>. Pour calculer l'intersection, on détermine la trace polygonale

16. L'ensemble des faces quadrilatères manipulé dans FLUSEPA est coupé pour ne plus manipuler que des triangles.

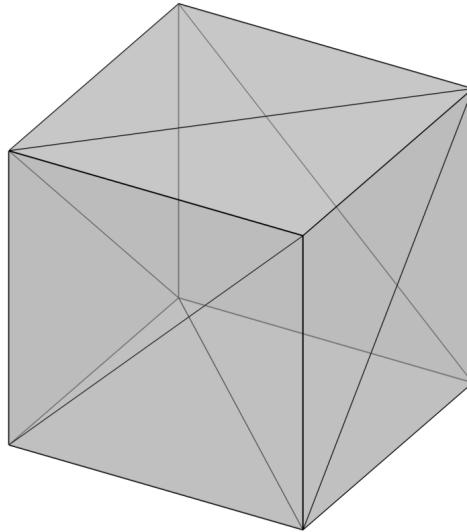
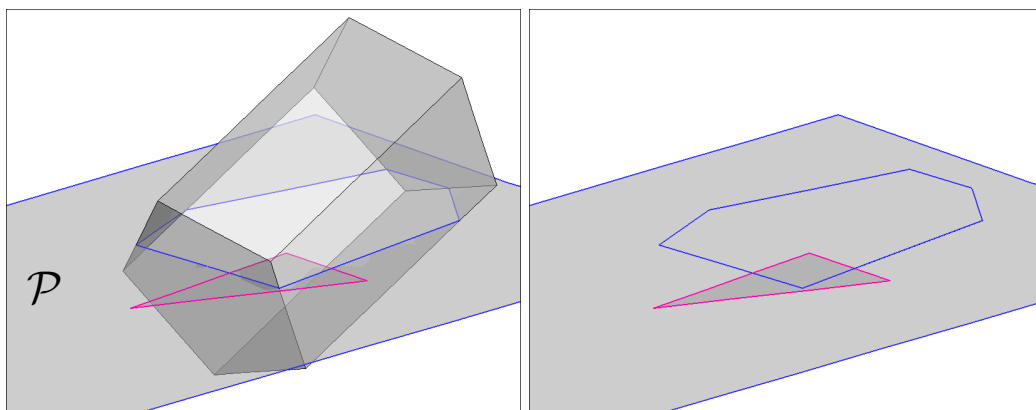


FIGURE 2.17 – Dodécaèdre défini par cellule pour le calcul des intersections

du polyèdre sur le plan  $\mathcal{P}$  de chaque facette triangulaire, puis la partie commune au triangle et à ce polygone comme présenté sur les figures 2.18(a) et 2.18(b) respectivement. On décompose alors la surface commune en une somme de sous surfaces algébriques pour le calcul.

(a) Projection sur un plan  $\mathcal{P}$  des différents polyèdres

(b) Utilisation de la trace du polyèdre

FIGURE 2.18 – Trace polygonale d'une cellule

Une fois la première étape effectuée, les moments surfaciques de chaque facette partiellement couverte ainsi que ceux des morceaux de facette du maillage de priorité supérieure qui ferment les cellules coupées doivent être recalculés. Finalement, les volumes, les moments volumiques et donc les centres de gravité des mailles coupées de  $\mathcal{M}_2$  sont calculés en utilisant les formules de *Green* tel que présenté sur la figure 2.19.

#### 2.3.4.4 Calcul des flux

Les flux à la frontière entre maillages sont calculés de la même façon que les flux entre deux mailles appartenant au même maillage. La topologie importe peu, puisqu'une interface entre des mailles sera traitée de la même manière, que ces mailles appartiennent au même maillage ou que cette face ait été créée par intersection. En effet, les propriétés géométriques ont été réactualisées de façon à utiliser les caractéristiques géométriques des faces nouvellement créées par intersection. On pourra remarquer que la conservativité du schéma global est préservée puisque le transit d'information est assuré par les flux calculés au niveau de la surface exacte

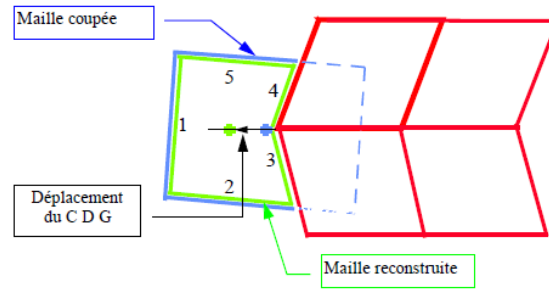


FIGURE 2.19 – Mise à jour des moments volumiques des cellules après intersection

d'intersection qui est, de fait, maillée de manière conforme.

### 2.3.4.5 Assemblage

Si le volume d'une cellule après calcul des intersections est trop faible, alors le pas de temps « caractéristique » de cette maille peut être pénalisant pour le calcul. Ce cas est assez classiquement rencontré dans les méthode *cut-cell*. Etant donné que le pas de temps convectif est proportionnel au rapport du volume sur la surface externe (et à son carré pour la diffusion), certains cas d'intersection de maillage peuvent devenir très pénalisants pour le calcul. On considère qu'une maille  $\mathcal{T}_\alpha$  est trop couverte dans les deux cas suivants :

$$|\mathcal{T}_\alpha| < \begin{cases} c_{\text{assemb}} |\mathcal{T}_{\alpha, \text{init}}| \\ \frac{3c_{\text{assemb}}}{1 + 2c_{\text{assemb}}} \sqrt{\mathcal{S}_{\text{min}}} |\mathcal{S}_{\text{tot}}| \end{cases} \quad (2.49)$$

où l'indice  $_{\text{init}}$  signifie que l'on utilise les propriétés avant calcul des intersections et :

$$\begin{aligned} c_{\text{assemb}} &= 0.3 \\ \mathcal{S}_{\alpha, \text{tot}} &= \sum_{i=1}^N |\mathcal{S}_i| \text{ est la somme des surfaces des } N \text{ faces de } \mathcal{T}_\alpha, \\ \mathcal{S}_{\text{min}} &= \left( \frac{|\mathcal{T}_{\text{init}}|}{\mathcal{S}_{\text{tot, init}}} \right)^2, \text{ une surface caractéristique} \end{aligned}$$

La solution consiste alors à assembler les mailles trop couvertes avec des mailles voisines suffisamment découvertes. De cette façon, l'ensemble ainsi formé constitue une « macro-maille » (figure 2.20) dont la taille caractéristique est assez grande pour ne plus pénaliser l'intégration temporelle. L'assemblage peut s'effectuer entre des mailles d'un même maillage mais aussi, lorsque cela n'est plus possible entre des mailles provenant de maillages différents. L'agrégat est ensuite construit en faisant la somme des volumes, des moments volumiques et des variables conservatives des sous-mailles qui le constituent : l'assemblage est une opération conservative vis à vis des grandeurs du vecteur  $w$  (paragraphe 2.2).

Dans le cas où les corps sont en mouvement, la couverture des mailles varie à chaque itération de sorte qu'un assemblage valide à un instant peut ne plus l'être à l'itération suivante. De façon pratique, il est nécessaire de pouvoir désassembler les cellules assemblées, les variables conservatives sont alors redistribuées à chacune des mailles formant l'agrégat. Nous détaillerons l'algorithmique, dans un contexte de parallélisme à Mémoire Distribuée, des mécanismes d'assemblage et désassemblage au paragraphe 2.4 qui traite de l'architecture logicielle globale de FLUSEPA.

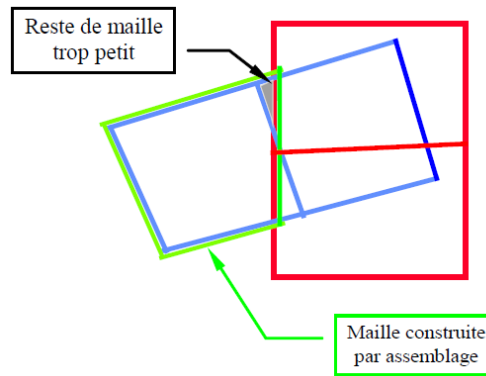


FIGURE 2.20 – Assemblage de mailles

### 2.3.4.6 La formulation *Euler-Lagrange Arbitraire*

FLUSEPA met en œuvre une formulation *Arbitrary Lagrangian-Eulerian* (ALE en anglais) pour les cellules en mouvement (*cf.* Hirt [HAC74, HAC97]). Une revue exhaustive sur l'utilisation de telles méthodes dans le cadre de la CFD (*ie* sans prendre en compte les applications pour la mécanique du solide) a été faite récemment par Barlow [BMR<sup>+</sup>16]. Dans cette formulation, chaque nœud du maillage peut se déplacer selon un vecteur qui lui est propre et qui est lié en pratique au déplacement du corps auquel il est rattaché. Ce déplacement est déduit de l'intégration des efforts sur chacun des corps. Chaque grille se voit affectée un numéro de corps, ainsi, plusieurs grilles peuvent avoir le même numéro de corps. Chaque nœud du maillage peut se déplacer en s'appuyant, par exemple, sur une formulation classique de déplacement des corps rigides, en utilisant le vecteur translation et rotation du corps auquel il est attaché<sup>17</sup>.

## 2.4 Architectures des versions 6 et 7 de FLUSEPA

### 2.4.1 Structure de données

FLUSEPA est un code actuellement développé en FORTRAN90 avec quelques interfaces en C. Historiquement la structure est héritée des premiers développements dans les années 90 écrits en FORTRAN77 et ressemble donc à la technique classique de *structure of arrays* qui permet d'avoir une bonne contiguïté des données pour la partie qui est intensive en terme de calcul à savoir le solveur aérodynamique (voir section 2.4.3). FLUSEPA n'est donc pas écrit en mode orienté objet, utilisant des types dérivés comme cela est fait couramment.

### 2.4.2 Version 6, parallélisation pour des machines à mémoire partagée

Le développement du module de maillage a été effectué sur deux versions du code FLUSEPA, la première version est celle développée pour des machines à mémoire partagée via l'utilisation de l'interface de programmation (API) OpenMP [DM98]. Cette version possède deux modules exécutés de façon séquentielle qui sont tous deux inclus dans la boucle d'intégration temporelle :

1. le solveur aérodynamique qui effectue l'intégration en temps,
2. le module de calcul des propriétés topologiques des cellules qui effectue les déplacements des différents maillages et le calcul des intersections.

L'ordinogramme de la version 6 de FLUSEPA tel qu'il est présenté sur la figure 2.21 est relativement simple et ne nécessite pas de développement particulier.

17. Le cas d'un maillage déformable est aussi prévu et utilisé en aéro-élasticité voir schéma 2.1.

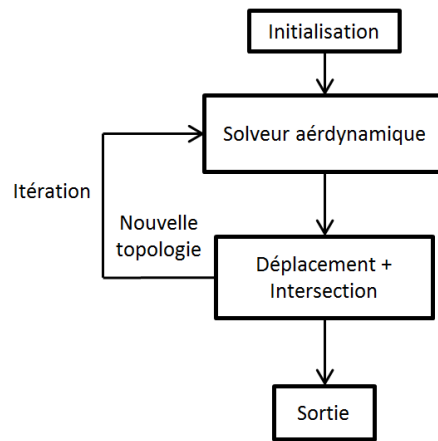


FIGURE 2.21 – Architecture de FLUSEPA V6

### 2.4.3 Version 7, parallélisation pour des machines à mémoire partagée et distribuée

La seconde version du code qui utilise un parallélisme sur deux niveaux, mémoire partagée, mémoire distribuée possède des caractéristiques qu'il est intéressant de décrire. La parallélisation en mémoire distribuée repose sur une décomposition de domaine. Cette décomposition est effectuée grâce au partitionneur de graphe SCOTCH [PR96]<sup>18</sup>. Un processus maître construit le graphe qui est envoyé à SCOTCH et, grâce au partitionnement, décide des actions à effectuer au cours du calcul en distribuant la charge de travail sur des processus esclaves. Un ensemble de processus esclaves assure l'intégration en temps. Une forme simplifiée de l'architecture de FLUSEPA V7 est présentée sur la figure 2.22. Les communications entre esclaves et maître sont faites avec la bibliothèque de fonctions *Message Passing Interface* (MPI). Lors de la construction du graphe pour le partitionnement, l'utilisation d'un schéma à pas de temps local nécessite que les nœuds (les cellules du maillages) et les arêtes (faces) soient pondérés selon leur coût calculatoire. Pour l'instant, seules les mailles sont pondérées via la formule 2.46 et les arêtes du graphe ne possèdent pas de poids particulier. On illustre sur la figure 2.23 un maillage coloré par numéro d'esclave d'un calcul avec adaptation de maillage qui utilise la pondération liée au coût calculatoire du schéma à pas de temps local.

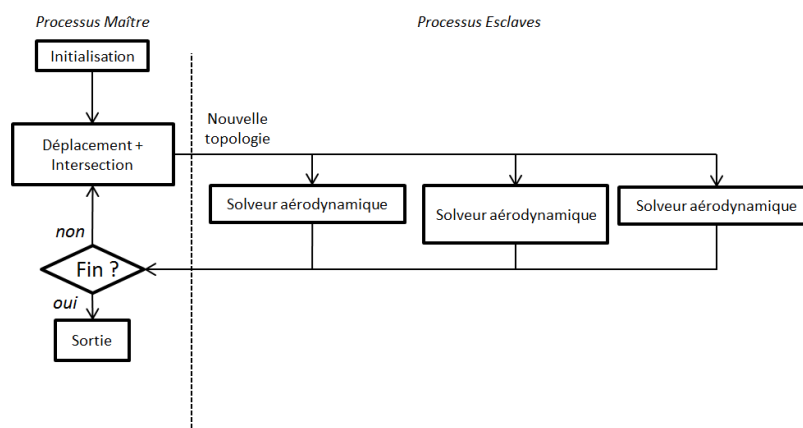


FIGURE 2.22 – Architecture de FLUSEPA V7

Le processus d'intersection étant exécuté de façon séparée des solveurs aérodynamiques, il existe une spécificité pour le calcul des propriétés des mailles assemblées que l'on a introduites au paragraphe 2.3. Les algorithmes 2

18. On pourra aussi citer les partitionneurs parMETIS [KSK97], ZOLTAN [DBH<sup>+</sup>02] et p4est [BWG11].

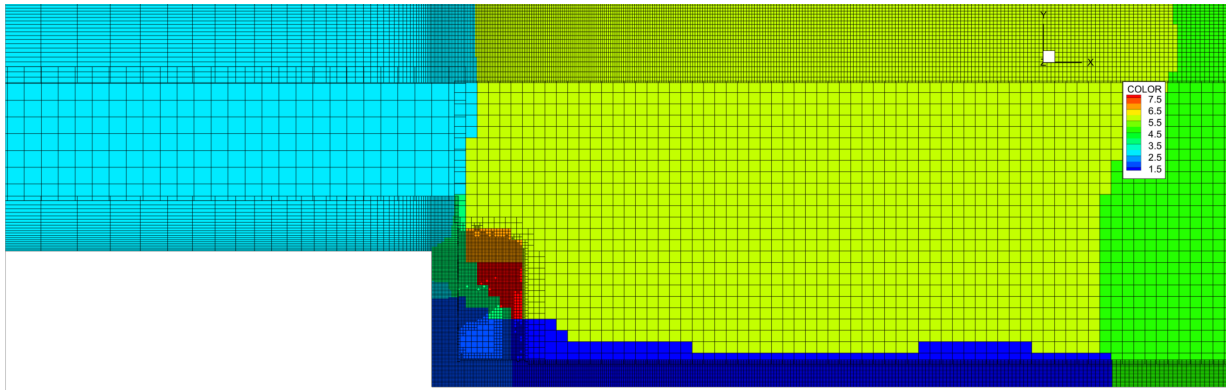


FIGURE 2.23 – Maillage coloré par numéro d’esclave au cours d’un calcul

et 3 permettent cette logique d’assemblage/désassemblage.

---

**Algorithme 2** Algorithme mise à jour de la topologie du maître
 

---

- 1: Estimation de la cinématique des corps à partir des efforts
  - 2: **si** déplacement suffisant **alors**
  - 3: Recalcul de la géométrie
  - 4: Envoi des nouvelles propriétés géométriques des anciennes mailles assemblées
  - 5: Envoi des nouveaux index de calcul par esclave
  - 6: Attribution d’un esclave pour les mailles qui ont été couvertes ou découvertes
  - 7: Création des piles d’échange pour les esclaves liées au changement de couleur des mailles
  - 8: Envoi des propriétés géométriques des mailles (avec assemblage)
  - 9: **fin si**
- 

---

**Algorithme 3** Algorithme d’assemblage / désassemblage par esclave aérodynamique
 

---

- 1: Réception des calculs d’intersection
  - 2: **si** nouvelle géométrie **alors**
  - 3: Réception des nouvelles propriétés géométriques des mailles anciennement assemblées
  - 4: Désassemblage des variables conservatives des mailles précédemment assemblées
  - 5: Création des pointeurs d’échange entre esclaves pour toutes les variables conservatives
  - 6: Envoi et réception des champs de variables conservatives entre esclaves
  - 7: Réception des nouveaux index avec leurs bornes pour le calcul en numérotation globale
  - 8: Création d’une numérotation locale par esclave
  - 9: Réception des nouvelles propriétés géométriques
  - 10: Assemblage des variables conservatives des mailles assemblées
  - 11: **sinon**
  - 12: Création des pointeurs d’échange entre esclaves pour toutes les variables conservatives et les propriétés géométriques
  - 13: Envoi et réception des champs de variables conservatives et des propriétés géométriques des mailles à échanger entre esclaves
  - 14: Création d’une numérotation locale par esclave
  - 15: **fin si**
- 

On utilise le terme index pour qualifier la table répertoriant uniquement la liste des cellules ou faces sur lesquels le solveur aérodynamique doit effectuer les boucles de calcul. Cette liste est triée en segments avec des bornes associées, par exemple les cellules sont classées en plusieurs catégories :

- les mailles de bord non assemblées ;
- les mailles de bord assemblées ;
- les mailles internes non assemblées ;
- les mailles internes assemblées.

Une fois triées dans un tableau, les listes se présentent sous la forme d’index comme présenté sur la figure 2.24.

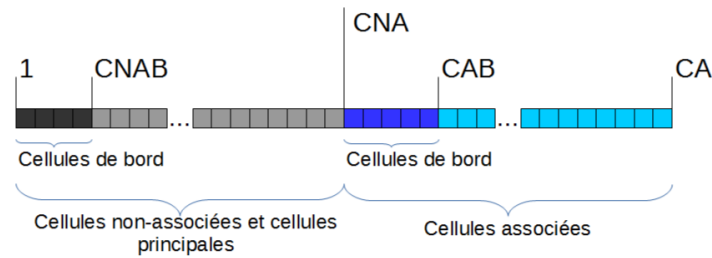


FIGURE 2.24 – Index des cellules triées en segments avec les bornes associées

## 2.5 Conclusion

Avant de détailler la méthodologie AMR construite au cours de la thèse, nous avons balayé globalement le contexte du code FLUSEPA, dans lequel elle s'insère. En particulier, nous avons exposé les différents aspects du code qui vont être touchés par l'implémentation de cette fonctionnalité AMR. Dans ce chapitre nous avons rappelé les équations physiques de Navier Stokes ainsi que les modèles de turbulence associés. Le principe d'une modélisation hybride RANS/LES qui consiste en l'hybridation continue d'un modèle RANS avec un modèle de sous-maille LES dans les zones suffisamment résolues a été présenté. Cette modélisation assure un compromis intéressant d'un point de vue industriel entre coût de calcul et précision. Pour résoudre ces équations, une formulation Volumes Finis centrée aux éléments est utilisée dans le code FLUSEPA. La notion de moments géométriques volumiques et surfaciques introduite dans ce chapitre autorise la montée en ordre de précision de la technique de résolution : ces moments permettent la description précise des caractéristiques géométriques sur lesquelles s'appuient les schémas numériques de FLUSEPA. L'algorithme d'intégration temporelle explicite adaptative présenté possède une grande efficacité pour la résolution des phénomènes instationnaires. Le coût algorithmique par maille selon les classes temporelles auxquelles elles appartiennent a été détaillé. Nous avons présenté en profondeur la méthode de chevauchement par intersection de maillages de FLUSEPA. Cette méthode, comme on a pu l'observer, se différencie des méthodes classiques de chevauchement de maillages par le fait qu'elle utilise la surface de l'intersection géométrique entre maillages pour calculer les flux plutôt que l'interpolation dans les cellules de la zone recouvrement pour transmettre l'information entre grilles. Cette technique d'intersection de maillage est particulièrement intéressante pour les écoulements fortement instationnaires et les écoulements confinés parce qu'elle est conservative par construction. En outre, cette méthode ne nécessite pas de traitement particuliers vis à vis du solveur ; les faces créées par intersection sont traitées comme les autres. Notons pour finir que le solveur aérodynamique est basé sur une technique qui lui permet de monter arbitrairement en ordre à condition de travailler sur des moments géométriques d'ordres suffisants (voir article de Pont *et al.* [PBCR17]). Enfin nous avons décrit brièvement l'architecture logicielle de FLUSEPA Version 7 permettant l'exécution de son solveur aérodynamique sur des machines à mémoire distribuée.

## Chapitre 3

# Construction d'une méthode d'adaptation de maillages par création d'arbres *octree* par cellule

### Introduction

Dans le chapitre précédent nous avons décrit l'architecture logicielle de FLUSEPA, ainsi que les méthodes numériques associées et un des modèles physiques utilisés pour la simulation d'écoulements instationnaires turbulents. Le présent chapitre explicite la construction d'un module de raffinement de maillage adaptatif (AMR) et son implémentation dans le code. Ce module vient compléter les capacités de FLUSEPA en posant les bases d'une stratégie globale d'adaptation de maillage pour des écoulements instationnaires, dans le contexte de maillages intersectants. Dans ce chapitre nous présentons, d'un point de vue algorithmique, une méthode de partition de mailles en 8 mailles « filles ». La détermination des cellules devant être raffinées ou déraffinées met en œuvre des critères qui sont décrits dans l'annexe A. Cette subdivision s'accompagne de la création d'arbres de relation entre les éléments ainsi que d'une redistribution conservative des variables (*cf.* chapitre 4). La méthode ainsi développée élargit le cadre des simulations pouvant être traitées efficacement par FLUSEPA. Dans un premier temps, nous exposons l'architecture logicielle globale de la méthode implémentée et la gestion partielle des maillages en mémoire distribuée. Dans un second temps, nous présentons le cœur des algorithmes d'adaptation de maillage et les problématiques qui ont été traitées pour des maillages non-structurés. En particulier nous nous intéressons :

- aux traitements spécifiques dans les zones non-conformes des maillages, ainsi qu'à la méthode d'équilibrage 2–1 par nœud, qui non seulement permet de réduire le nombre de configurations de « non-conformités » locales (et donc de configurations de polyèdres d'intersection identifiables) mais aussi de limiter la variation de tailles de mailles voisines,
- à la gestion des accès mémoire liés aux choix des structures de données représentant les mailles,
- à l'équilibrage de la forêt d'arbres de relation.

Enfin, nous décrivons les spécificités des algorithmes utilisés pour les solveurs aérodynamiques qui disposent d'une version distribuée des maillages, notamment l'utilisation d'arbres de relation de taille variable et leurs modifications pour la prise en compte des zones non-conformes.

### 3.1 Architecture logicielle du module d'adaptation de maillages pour une exécution en Mémoire Distribuée (MD)

Les modules permettant de faire l'adaptation de maillages par enrichissement sont répartis sur plusieurs processeurs qui s'exécutent en parallèle et sont intégrés dans les architectures détaillées au paragraphe 2.4. Deux types



de processus sont concernés et seront explicités dans ce chapitre : les processus de type « maître » qui effectuent l'ensemble des tâches autre que l'intégration en temps et les processus aérodynamiques. Dans tout ce chapitre, on suppose que la simulation s'effectue avec un seul processus maître et plusieurs processus aérodynamiques dits « esclaves ».

On notera en premier lieu que la logique mise en place vise à ce que le processus maître possède l'intégralité de la topologie des maillages et que les processus aérodynamiques n'en possèdent seulement qu'une partie très réduite. Plus précisément, les processus esclaves utilisent une topologie partielle des éléments qui leur sont attribués. Par exemple, il n'existe pas de structure élément  $\rightarrow$  face<sup>1</sup> et de structure élément  $\rightarrow$  nœud.

### 3.1.1 Schéma d'adaptation de maillages

Le pseudo-code d'une adaptation de maillage est décrit dans l'algorithme 4. On note  $t_{\text{final}}$  atteint en fin de simulation et  $t$  le temps au cours du calcul<sup>2</sup>.

---

#### Algorithme 4 Adaptation de maillage Maître/Esclaves

---

- 1: Envoi par le maître des paramètres et des critères aux esclaves
  - 2: **pour**  $t < t_{\text{final}}$  **faire**
  - 3:   Avancement en temps des esclaves et détection des mailles à raffiner et à déraffiner en fonction des critères
  - 4:   Envoi par les esclaves au maître de l'ensemble des listes des mailles à raffiner et/ou à déraffiner
  - 5:   Réception par le maître des listes des éléments à raffiner et/ou déraffiner envoyée par les esclaves
  - 6:   Contrôle par le maître de l'espace libre dont les esclaves disposent déterminant la prise de décision quant à la modification de maillage
  - 7:   Envoi par le maître de la décision prise aux esclaves qui sont prêts désormais à recevoir la nouvelle topologie
  - 8:   Adaptation de maillage par le maître nécessitant un nouveau calcul des intersections
  - 9:   Envoi par le maître de la nouvelle topologie aux esclaves
  - 10:   Réception par les esclaves de la nouvelle topologie
  - 11:   Mise à jour par les esclaves des propriétés géométriques des éléments raffinés/déraffinés et répartition des variables conservatives représentant les différents champs dans les nouvelles cellules
  - 12:   Relecture par le maître des paramètres d'adaptation et envoi aux esclaves si modification
  - 13: **fin pour**
- 

La liste des cellules que le maître reçoit de chaque esclave, devant être raffinées et/ou déraffinées, dépend des critères définis par l'utilisateur. Cette architecture permet à l'utilisateur d'appliquer différentes contraintes sur le maillage au cours du calcul. L'appel du module AMR, qui crée et supprime des éléments, est uniquement effectué par le processus maître dans l'étape 8. Le module, dont ce manuscrit fait l'objet reçoit comme entrée<sup>3</sup> :

- le type de senseur physique qui « pilote » l'adaptation avec les seuils associés,
- le type de critères géométriques (compatibilité des tailles de maille au niveau des intersections et les critères de qualité concernant les mailles raffinables/déraffinables).

En l'état, on remarque que l'utilisation d'un module de remaillage ne modifie que très peu l'exécution de la version actuelle de FLUSEPA. Les deux seules communications nécessaires et spécifiques à l'AMR sont situées dans les étapes 4 et 9<sup>4</sup>.

---

1. On utilise ici la notation «  $A \rightarrow B$  » signifiant « A renvoie B ».

2. On fait ici l'hypothèse que l'utilisateur a choisi ou défini lui-même les critères de raffinement ainsi qu'une taille minimale de maille admissible. On appelle taille minimale la taille caractéristique par maille en dessous de laquelle l'utilisateur ne veut plus raffiner.

3. Le lecteur pourra se référer à l'annexe A de ce manuscrit pour le détail des critères d'adaptation utilisés.

4. La logique des communications MPI entre processus est décrite dans la thèse de Couteyen [CC16].

### 3.1.2 Attribution de numéro d'esclave et équilibrage de la charge

Lorsque le calcul s'exécute en **MD** avec adaptation de maillage, les nouvelles cellules créées par raffinement et les cellules réactivées par déraffinement se voient attribuer un numéro d'esclave noté **KOLOR**, par héritage. Le principe en est le suivant : lorsqu'un élément est raffiné, ses mailles filles, de volume non nul après intersection, ont la même couleur que leur mère. Si une ou plusieurs des filles obtenues après raffinement sont totalement couvertes, alors elles sont discriminées par **KOLOR** = 0 et ne sont envoyées à aucun esclave. Concernant le déraffinement, la couleur de la maille mère reconstruite par déraffinement repose sur la relation :

$$\mathbf{KOLOR}(\text{maille\_mere}) = \max(\mathbf{KOLOR}(\text{mailles\_filles})) ,$$

étant donné que, par construction toutes les mailles sœurs de volume non-nul ont la même couleur. On notera que cette formule est aussi valide, dans le cas où toutes les mailles déraffinées sont couvertes<sup>5</sup>.

Si l'attribution du numéro de couleur par héritage entraîne un déséquilibre trop important de la charge entre les esclaves, il est possible d'effectuer un rééquilibrage global. Un module déduit, à partir de la liste des faces, une topologie reconstituée du maillage initial. Ce maillage « compact » est obtenu en remplaçant les éléments de part et d'autre des faces par leur cellule originelle. Les faces internes ainsi obtenues (qui possèdent le même élément des deux côtés) sont éliminées de la liste comptable. La fonction de coût  $C(\Omega)$  algorithmique de chaque maille  $\Omega$  du maillage initial est alors évaluée par sommation du coût de toutes les mailles actives  $\Omega_i$  issues de  $\Omega$  (ayant un volume non nul après calcul des intersections) :

$$C(\Omega) = \sum_{\Omega_i \subset \Omega / |\Omega_i| \neq 0} C(\Omega_i) . \quad (3.1)$$

Deux fonctions de coût  $C(\Omega_i)$  par cellule peuvent être utilisées :

- une pondération déterminée par la classe temporelle de la cellule pour l'intégration temporelle adaptative explicite (voir équation 2.46),
- une pondération unitaire dans le cas d'une intégration implicite.

Les faces obtenues ne sont pas pondérées. On représente sur la figure 3.1 l'utilisation de ces deux options pour un même maillage. Les figures 3.1(a) et 3.1(b) présentent la pondération par maille, liée au coût de calcul, obtenue sur le maillage initial reconstruit. Les figures 3.1(c) et 3.1(d) montrent le partitionnement fait par SCOTCH selon le type de pondération utilisée.

L'héritage de la couleur et le repartitionnement global assurent la propriété suivante :

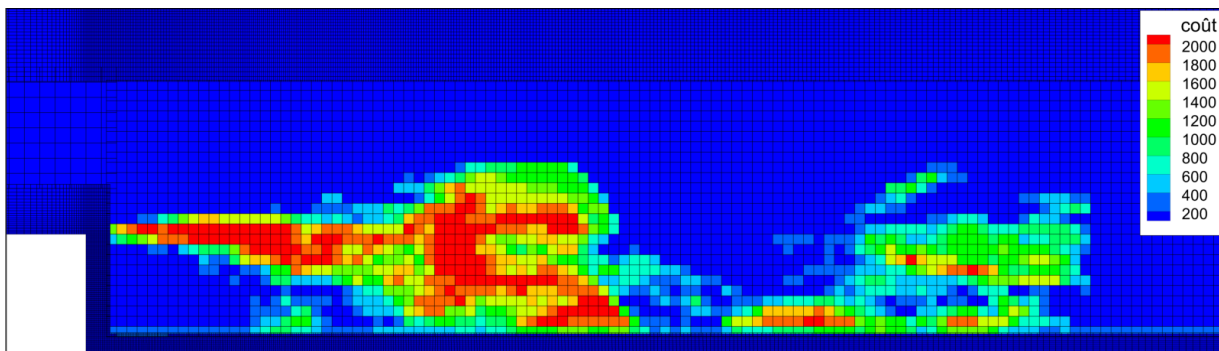
**Propriété 1.** *Si pour une cellule  $\Omega$  ayant été raffinée, il existe une cellule  $\Omega_i$  qui est une descendante de  $\Omega$  et tel que  $\mathbf{KOLOR}(\Omega_i) = k$  avec  $k \neq 0$  alors l'ensemble des  $N$  mailles issues de  $\Omega$  créées par raffinement  $\Omega_j, j \in \llbracket 1, N \rrbracket$  est tel que :*

$$\mathbf{KOLOR}(\Omega_j) = \begin{cases} k , & \text{si la maille est active et non couverte} \\ 0 , & \text{sinon} \end{cases} .$$

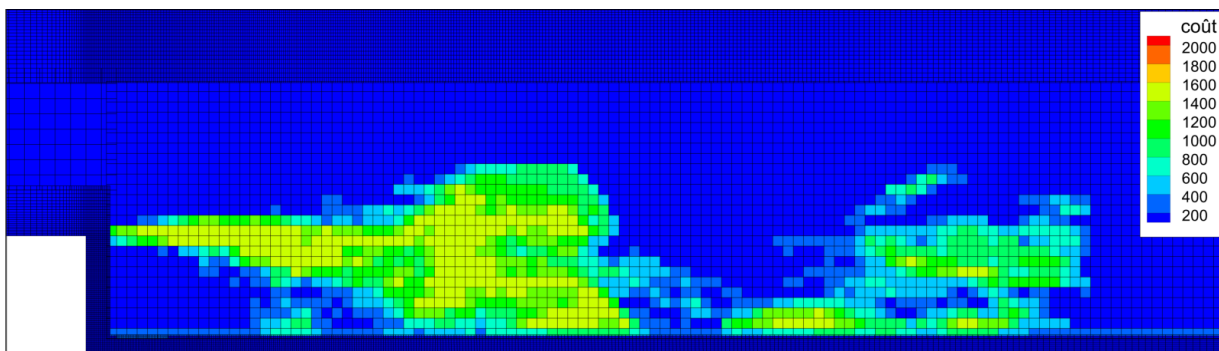
Ainsi, s'il existe d'autres mailles descendantes de  $\Omega$  et utilisées par le calcul alors elles ont toutes le même numéro d'esclave.

---

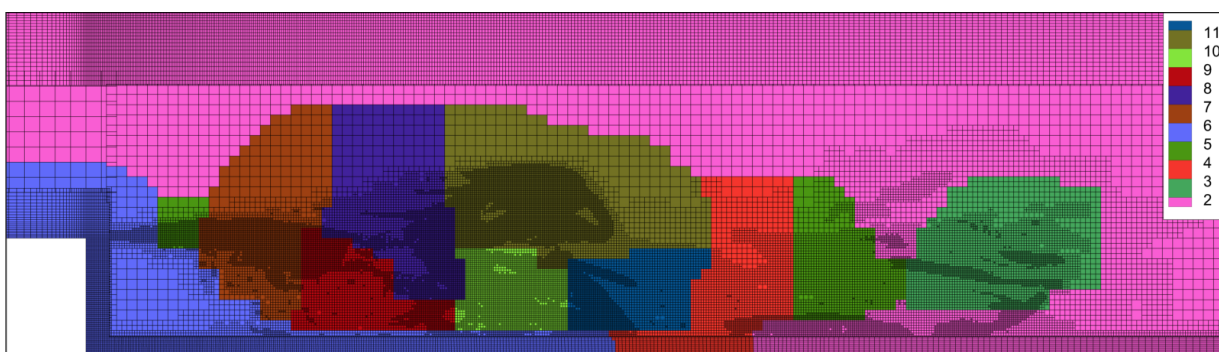
5. A noter qu'il existe quelques cas particuliers non détaillés, comme par exemple lors du mouvement de maillages qui peut entraîner des situations de déraffinement où  $\max(\mathbf{KOLOR}(\text{mailles\_filles})) = 0$ , alors que la cellule obtenue par déraffinement n'est pas couverte.



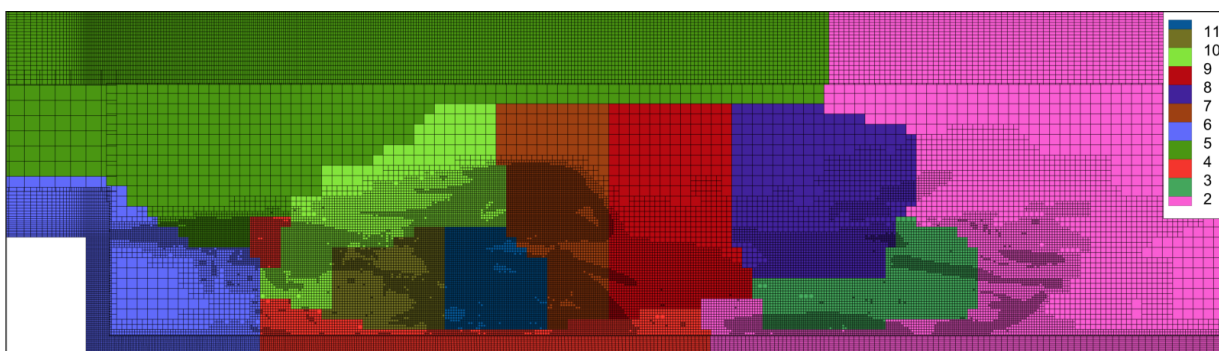
(a) Champ de coût par cellule sur le maillage initial reconstruit avec une pondération  $C(\omega) = 1$  pour toutes les cellules



(b) Champ de coût par cellule sur le maillage initial, pondération liée à celle de l'intégration temporelle adaptative par maille



(c) Répartition des esclaves déduite sur le maillage réel avec une pondération  $C(\omega) = 1$  pour toutes les cellules, coloration par numéro d'esclave



(d) Répartition des esclaves déduite sur le maillage réel avec un coût pondération basée sur le coût de l'intégration temporelle adaptative par maille, coloration par numéro d'esclave

FIGURE 3.1 – Reconstruction du maillage initial après adaptation et répartition en processus esclaves du volume à calculer

## 3.2 Processus maître

**Cadre restrictif** On suppose dans toute la suite, que les maillages sont non structurés mais constitués uniquement d’hexaèdres, non dégénérés, représentés par 8 nœuds et 6 faces, voir figure 2.5(a). Ainsi, toutes les faces liées à des mailles sont des quadrilatères. Cette hypothèse limite dans un premier temps les cas particuliers et la complexité des routines afin de poser simplement la logique globale.

### 3.2.1 Algorithme de modification de maillage

On détaille dans l’algorithme 5 le pseudo-code pour la modification de maillage du maître. On note  $nb_{\text{maille raff}}$  le nombre total de mailles à raffiner et  $nb_{\text{maille deraff}}$  le nombre de mailles à déraffiner. On rappelle que les index de calcul sont les tableaux qui référencent uniquement les mailles et les faces utilisées par les solveurs aérodynamiques. Ce sont les éléments de volume non-nul et les faces de surface non-nulle après calcul des intersections géométriques. L’index de calcul des faces comprend aussi les faces créées par intersection, comme présenté au paragraphe 2.4.3.

---

#### Algorithme 5 Modification de maillage (MOUVT)

---

```

1: si ( $nb_{\text{maille raff}} > 0$ ) alors
2:   Raffinement des cellules, mise à jour de la topologie
3:   Mise à jour des éléments constitutifs des polyèdres d’intersections liées aux mailles raffinées
4: fin si
5: si ( $\max_{\text{maillages}}(\text{priorité}) > 0$ ) alors
6:   Réinitialisation des propriétés géométriques sans prise en compte des chevauchements
7:   Calcul des intersections, mise à jour de la topologie et des propriétés géométriques
8:   si ( $nb_{\text{maille deraff}} > 0$ ) alors
9:     Déraffinement des cellules, mise à jour de la topologie
10:    Mise à jour de la topologie des faces créées par les intersections
11:    Mise à jour des propriétés géométriques des mailles déraffinées par assemblages des moments géométriques
12:   fin si
13: sinon
14:   si ( $nb_{\text{maille deraff}} > 0$ ) alors
15:     Déraffinement des cellules, mise à jour de la topologie
16:     Mise à jour des propriétés géométriques des mailles déraffinées par assemblages des moments géométriques
17:   fin si
18:   Détection des mailles à assembler et assemblage des moments géométriques
19:   Détermination des faces et des éléments servant au calcul
20: fin si
21: Mise à jour des coques liées aux mailles déraffinées

```

---

Il faut bien noter les deux impératifs concernant les algorithmes de raffinement et de déraffinement par rapport au calcul des intersections :

1. le calcul des intersections doit être effectué après le raffinement car effet, il n’est pas possible d’anticiper les propriétés géométriques des mailles créées par raffinement après intersection. En effet, on ne pas prévoir quel seront les moments géométriques des éléments intersectés **avant** d’avoir effectué le calcul des intersections,
2. le déraffinement est effectué après le calcul des intersections, la topologie des mailles que l’on obtient par déraffinement peut être déduite de la topologie des mailles précédemment existantes. On utilise cette propriété du déraffinement pour les arbres créés par fusion de cellules sur les esclaves (*cf.* paragraphe 3.3).

Ces deux propriétés expliquent l’ordonnancement atypique de l’algorithme 5 d’adaptation de maillages, dans lequel le raffinement est effectué avant le déraffinement, alors que pour la plupart des algorithmes AMR c’est

l'inverse qui est opéré<sup>6</sup>.

### 3.2.2 Subdivision d'une maille

La partition « isotrope » effectuée par le module AMR est présentée sur la figure 3.2. Cette division en plusieurs éléments fils nécessite de créer :

- les nœuds au milieu des arêtes, (en rouge sur la figure 3.2(a)), s'ils n'existaient pas déjà. La vérification de l'existence est faite via une procédure similaire à celle classique d'une table de *hashage*,
- les nœuds « centre des faces » au milieu des points 1 et 3 locaux des faces<sup>7</sup> (en vert), si la face n'était pas déjà divisée<sup>8</sup>,
- le nœud au centre de gravité de la maille (en bleu),
- 12 faces internes à la maille à raffiner en bleu sur la figure 3.2(b),
- 8 mailles filles.

Le principe est que les nouveaux nœuds sont construits uniquement à partir des nœuds d'origine de la maille et ne prennent pas en compte les moments géométriques d'ordre élevé de cellule et des faces.

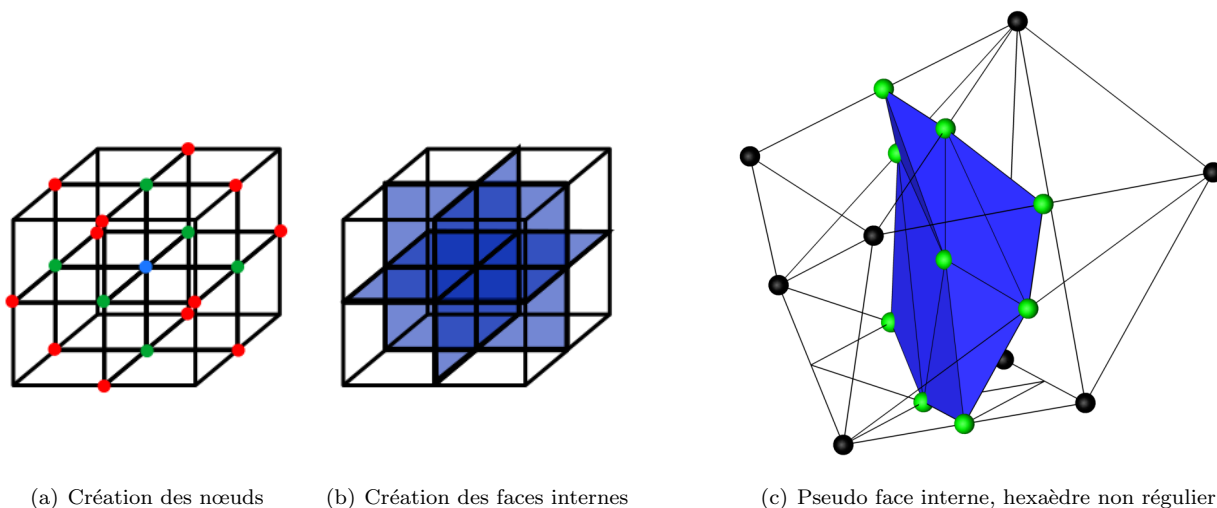


FIGURE 3.2 – Raffinement d'une cellule

Il faut bien noter que les faces internes créées lors de la découpe, présentées sur la figure 3.2(b) en bleu ne sont pas, en général, contenues dans trois plans. Le cas d'un hexaèdre non régulier découpé grâce à la méthode *octree* définie dans ce paragraphe est présenté sur la figure 3.2(c), avec :

- en bleu : 3 surfaces internes non-planes, chacune divisée en 4 faces à leur tour divisées en 2 triangles,
- en vert : les nœuds qui ont servi à construire ces faces,
- en noir : les arêtes de la maille originelle et les arêtes supports de la construction des nœuds sur les faces.

### 3.2.3 Création des nœuds sur les faces lors d'un raffinement

En suivant le principe précédent, lors d'une subdivision, la technique *octree* présentée ici ne construit et n'utilise pas le centre de gravité des faces à raffiner. La figure 3.3 illustre deux cas de raffinement d'une face que l'on affiche en transparence. On représente en vert le cas où le découpage se fait en s'appuyant sur le centre de gravité et les milieux des 4 arêtes du quadrilatère et en rouge la méthode implémentée qui utilise le milieu de l'arête des nœuds locaux 1 et 3 et les milieux des arêtes. Il existe un volume entre la face initiale découpée

6. On pourra se référer aux algorithmes de AMRClaw [BL97, LB<sup>+</sup>11], Chombo [Ada14], SAMRAI [WH00].

7. Voir paragraphe 2.2.1, figure 2.5(b).

8. Une face divisée est une face entre 2 cellules de niveau de raffinement différents.

en 2 triangles<sup>9</sup> et les nouvelles faces créées par ces deux découpes<sup>10</sup>. La géométrie initiale de la maille va donc conditionner la géométrie de toutes les mailles qui en seront issues. On assure ainsi la cohérence entre la définition des faces et la technique de raffinement de maillage. Si tel n'était pas le cas, il y aurait un flux de volume lors du raffinement autrement dit la somme des volumes des mailles filles ne serait pas égale au volume de la maille mère.

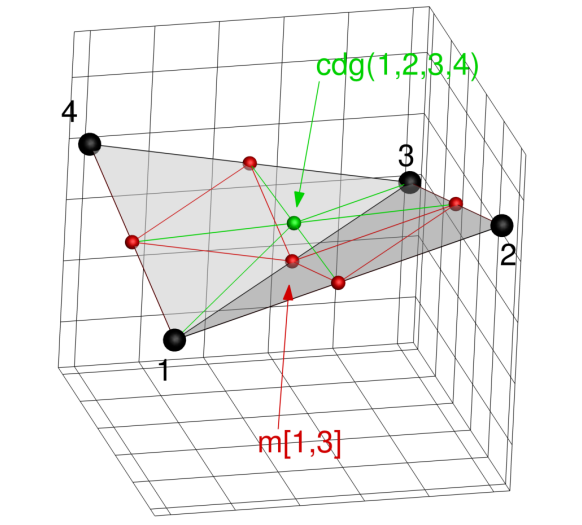


FIGURE 3.3 – Méthodes possibles pour le raffinement d'une face. En rouge : une découpe correcte, en vert : une découpe utilisant le centre de gravité de la face

### 3.2.4 Structure de données

Un arbre de relation est construit lors du raffinement de la maille mère et la lie à ses 8 mailles filles. L'arbre inverse liant les filles aux mères, dont elles sont issues, est aussi construit. La figure 3.4 présente une forêt à deux arbres contenant 23 éléments actifs représentés en vert et 3 éléments morts représentés en rouge. Ainsi, le maître manipule une forêt d'arbres de relation d'*octrees* comme représenté figure 3.4. Chaque cellule possède une table de correspondance qui renvoie le numéro de sa maille parent et des 8 mailles filles. Un niveau de raffinement **IRAFF** est aussi associé à chaque cellule. Un marqueur permet de signaler lorsqu'une maille a été raffinée. Quand elle n'est pas raffinée, elle fait partie des feuilles de l'arbre et est utilisée pour le calcul des intersections. L'ensemble des feuilles de la forêt forme les éléments actifs du calcul. Après le calcul des intersections, si une maille n'est pas totalement recouverte par les polyèdres des priorités supérieures alors elle est utilisée par les esclaves pour l'intégration temporelle.

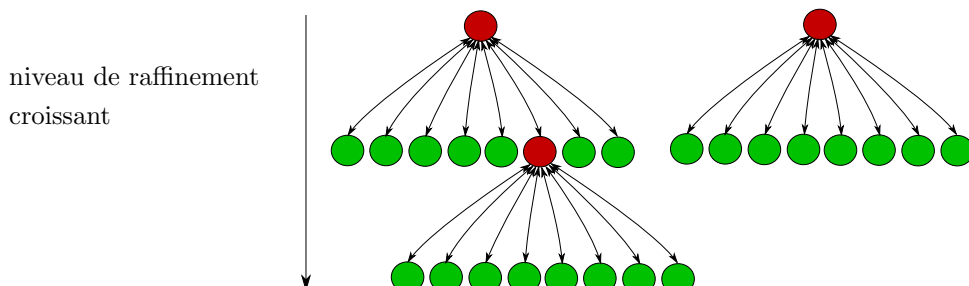


FIGURE 3.4 – Forêt d'*octrees* à deux arbres, en rouge : les éléments désactivés, en vert : les éléments actifs

Certaines limitations ont été implémentées : au cours d'une itération d'adaptation de maillage, une maille ne peut subir qu'une seule modification soit par raffinement soit par déraffinement. Par exemple il est impossible,

9. Telle qu'elle a été définie dans la section 2.2.1, figure 2.5(b).

10. On pourra remarquer que ce volume est un dodécaèdre.

quand une maille est raffinée, que l'ensemble des filles créées par raffinement soit déraffiné lors de la même itération. De façon similaire il est impossible, lorsqu'une maille est raffinée, qu'une des filles créées par raffinement soit elle-même raffinée à son tour <sup>11</sup>.

### 3.2.5 Équilibrage

La mise en place d'une règle de gestion de l'équilibrage de la forêt d'*octrees* est nécessaire pour restreindre le nombre de polyèdres pouvant être formés par maille et s'assurer de ne pas avoir de maille voisines ayant des tailles caractéristiques variant brutalement.

**Définition 7** (Valence). *La valence d'un nœud est le nombre de cellules qui utilisent ce nœud dans leur définition par la structure élément  $\rightarrow$  nœuds.*

**Définition 8** (Hanging nodes ou nœuds partiellement non-référencés). *Un hanging node ou nœud partiellement non-référencé est un nœud sur lequel s'appuient des éléments d'un maillage mais qui ne rentre pas dans la définition canonique de ceux-ci.*

Ainsi, si  $N$  est un *hanging node*, il existe toujours au moins un élément  $\Omega_j$  tel que  $N$  est situé sur la surface  $S_j$  de  $\Omega_j$  mais n'est pas référencé par la structure canonique élément  $\rightarrow$  nœuds. On représente sur la figure 3.5 l'exemple de deux triangles  $T_1 = \{N_3, N_6, N_5\}$ ,  $T_2 = \{N_5, N_6, N_2\}$  et d'un quadrilatère  $Q_1 = \{N_1, N_2, N_3, N_4\}$ . On suppose que  $N_3, N_5$  et  $N_2$  sont alignés. Le nœud  $N_5$  est un *hanging node* étant donné qu'il est, *a priori* suffisant de définir  $Q_1$  par 4 nœuds mais qu'il existe un cinquième nœud à sa surface.

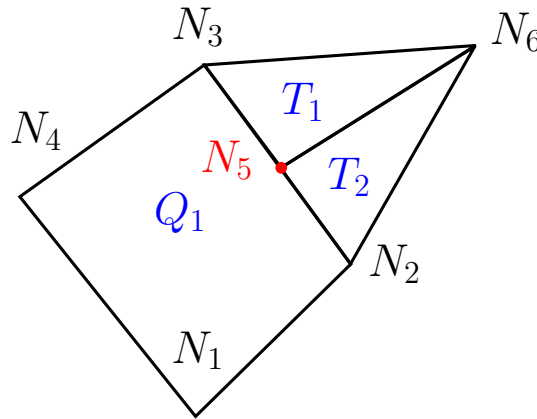


FIGURE 3.5 – Exemple de *hanging node*

**Définition 9** (Maillage conforme). *Un maillage est dit conforme s'il ne contient pas de hanging nodes.*

Les structures de données de FLUSEPA ont été modifiées de sorte que les maillages manipulés soient toujours conformes, les nœuds présents au milieu des arêtes ou des faces sont donc pris en compte par modification de la topologie des cellules concernées (*cf.* paragraphes 3.2.6 et 3.2.7).

**Définition 10** (Maillage équilibré). *Un maillage est dit équilibré 2 – 1 par nœud si pour tout nœud  $N_i$  de valence  $IVAL(N_i)$ , de boule de valence  $\mathcal{B}_{N_i}$  composée des éléments  $\Omega_j$  qui utilisent ce nœud, on a :*

$$\forall \{j, k\} / j \neq k, \quad \max_{\Omega_j \subset \mathcal{B}_{N_i}} (IRAFF(\Omega_j)) - \min_{\Omega_k \subset \mathcal{B}_{N_i}} (IRAFF(\Omega_k)) \leq 1 \quad (3.2)$$

Une règle d'équilibrage 2 – 1 par nœud a été imposée pour toutes les grilles des maillages manipulés. Ainsi, **toutes les feuilles des arbres (donc les mailles actives) qui ont un nœud en commun ne peuvent**

11. Ainsi, il est possible qu'après adaptation, les mailles créées soient encore sous-résolues par rapport au critère utilisé.

**pas avoir plus d'un niveau de raffinement d'écart.** Cette règle pour l'équilibrage permet de limiter la variation de taille d'une maille à une autre mais elle permet aussi de limiter le nombre de cas de non-conformité à traiter lors du raffinement et du déraffinement. Si des mailles doivent être raffinées par le maître, il est alors nécessaire de repérer et de trier préalablement les mailles qui peuvent effectivement être raffinées. Ce tri limite les déséquilibres dans la forêt et évite aussi la présence de plusieurs nœuds non référencés par une cellule sur une même arête, comme présenté sur la figure 3.6. Cet équilibrage limite aussi la complexité des bibliothèques permettant de construire les polyèdres conformes présentés au paragraphe 3.2.7.

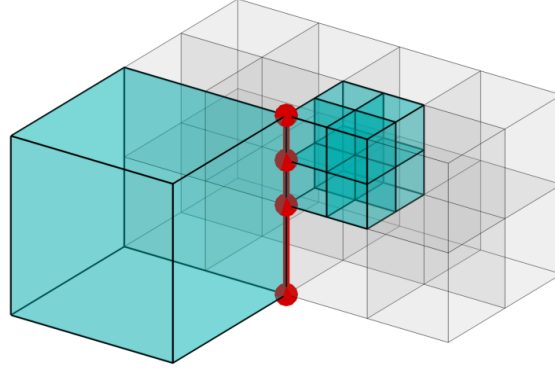


FIGURE 3.6 – En rouge *hanging nodes* double, mauvais équilibrage 2 – 1

Le détail de l'équilibrage est donné dans l'algorithme 6. On note  $\mathbf{IRFMAX}(\mathcal{N}_i) = \max_{\Omega_j \in \mathcal{B}_{\mathcal{N}_i}} (\mathbf{IRAFF}(\Omega_j))$  le tableau retournant pour un nœud  $\mathcal{N}_i$  le niveau de raffinement maximum des cellules qui l'utilisent et  $\mathbf{IRFMIN}$  le tableau retournant le niveau de raffinement minimum.  $nb_{grilles}$  est le nombre total de grilles dans le maillage manipulé et  $n_{\text{raff max}}$  le niveau de raffinement maximal des cellules dans la grille courante. Cet algorithme est itératif et « propage » le raffinement dans les cellules qui ne respectent pas la règle d'équilibrage, jusqu'à ce que le maillage final soit équilibré.

---

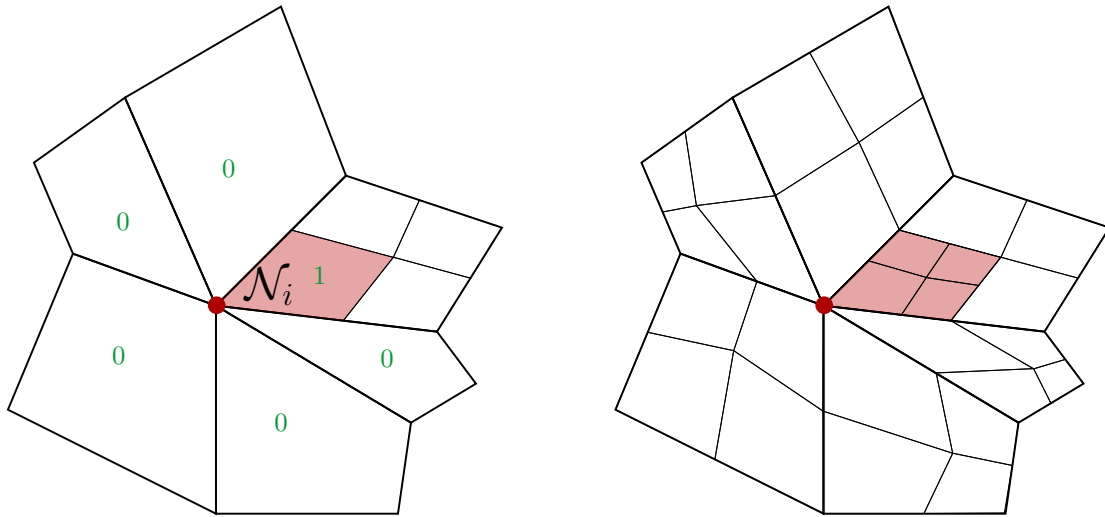
**Algorithme 6** Équilibrage 2 – 1 par nœud de l'arbre d'*octree*

---

- 1: Mise à jour de **IRFMAX**
  - 2: Prédiction de **IRFMAX** après raffinement
  - 3: Calcul  $n_{\text{raff max}}$  et  $n_{\text{ival max}}$  sur le maillage après raffinement
  - 4:  $\text{compteur}_{\text{nœud}} \leftarrow 0$
  - 5: **tant que**  $\text{compteur}_{\text{nœud}} < n_{\text{raff max}}$  **faire**
  - 6:   Evaluation de la présence après raffinement d'un déséquilibre sur les cellules voisines directes
  - 7:   **si** *déséquilibre == vrai* **alors**
  - 8:     Ajout des raffinables dans la boule de valence des nœuds non équilibrés
  - 9:     Prédiction de **IRFMAX** après raffinement
  - 10:    Calcul  $n_{\text{raff max}}$  et  $n_{\text{ival max}}$  sur le maillage après raffinement
  - 11:     $\text{compteur}_{\text{nœud}} \leftarrow 0$
  - 12:    **sinon**
  - 13:      $\text{compteur}_{\text{nœud}} \leftarrow \text{compteur}_{\text{nœud}} + 1$
  - 14:    **fin si**
  - 15: **fin tant que**
  - 16: Raffinement des cellules, mise à jour de la topologie
- 

On présente sur les figures 3.7(a) et 3.7(b) un cas d'équilibrage dans lequel :  $\mathbf{IRFMAX}(\mathcal{N}_i) - \mathbf{IRFMIN}(\mathcal{N}_i) = 1$  initialement. La maille représentée en rouge de niveau de raffinement 1 sur la figure 3.7(a) doit être raffinée, l'algorithme détecte alors l'ensemble des mailles devant être raffinées pour assurer l'équilibrage 2 – 1 de la boule de valence de  $\mathcal{N}_i$ . Le maillage équilibré résultant est représenté sur la figure 3.7(b).





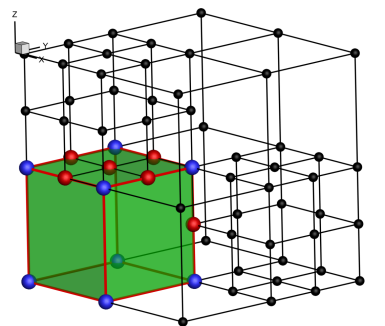
(a) Maillage non structuré équilibré avant raffinement. En rouge : la cellule devant être raffinée. En vert : le niveau de raffinement des cellules de la boule de valence de  $\mathcal{N}_i$

(b) Maillage résultant équilibré après raffinement

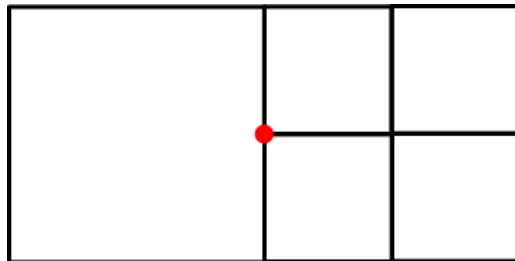
FIGURE 3.7 – Exemple d’une adaptation de maillage avec règle d’équilibrage 2 – 1 pour le nœud  $\mathcal{N}_i$

### 3.2.6 Traitement des *hanging nodes*

Le raffinement de maillage par *octree* entraîne la création de nœuds au milieu des faces, des arêtes et des éléments. Les nœuds au milieu des faces et des arêtes induisent une non-conformité du maillage, ces sont des *hanging nodes*. On présente un cas en 3 dimensions et un cas en 2 dimensions de *hanging nodes* rapportés en rouge sur les figures 3.8(a) et 3.8(b) respectivement. Pour que l’élément en vert sur la figure 3.8(a) soit correctement représenté, il faudrait en théorie qu’il possède 14 nœuds et 9 faces<sup>12</sup>. Or, par hypothèse, les éléments sont définis uniquement par 8 nœuds et 6 faces, donc la structure élément  $\rightarrow$  nœud ne référence que les nœuds en bleu pour la cellule verte. On constate donc que, dans le cas d’un maillage non-conforme, cette définition des éléments n’est plus suffisante et doit être enrichie pour pouvoir prendre en compte des cas particuliers.



(a) Cas courant d’un maillage volumique adapté, en rouge : les *hanging nodes* de cellule verte



(b) Maillage non conforme, en rouge : un *hanging node*

FIGURE 3.8 – Cas de maillages 2D et 3D non-conformes

Un contrôle de la présence de *hanging nodes* dans le cas de maillages adaptatifs a été introduit en même temps que le module d’adaptation de maillage. Cette gestion des *hanging nodes* est alors basée sur les variables suivantes :

12. Dans le cas d’un maillage non-conforme avec une règle d’équilibrage 2 – 1 par nœud, les mailles peuvent avoir jusqu’à 30 nœuds sur leur surface et 24 faces.

- le niveau de raffinement par élément **IRAFF**,
- une structure élément  $\rightarrow$  arête,
- une structure arête  $\rightarrow$  nœuds,
- le lien face  $\rightarrow$  nœud au milieu de l'arête des nœuds locaux 1 et 3,
- le niveau de raffinement maximal des éléments utilisant ce nœud, noté **IRFMAX**,
- la valence **IVAL** de chaque nœud.

Dans ce contexte, il peut être relativement onéreux de manipuler toutes ces structures, ainsi plusieurs règles ont été définies afin de limiter l'espace mémoire requis pour leur stockage :

- tout indice  $i$  non référencé par une structure élément  $\rightarrow$  nœud, élément  $\rightarrow$  arête ou élément  $\rightarrow$  face est considéré comme un espace libre et peut être utilisé lors du raffinement d'une cellule,
- les arêtes créées en cours de calcul servent uniquement au raffinement d'éléments. Ainsi, pour toute arête stockée  $\{1, 2\}$ , il existe un nœud 3 au milieu  $3 = m[1; 2]$ , voir figure 3.9.

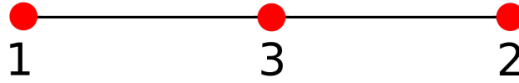


FIGURE 3.9 – Arête stockée lors du raffinement d'une cellule. Numérotation locale

Par conséquent, tout nœud référencé  $\mathcal{N}_i, i \in \llbracket 1, nb_{\text{nœud total}} \rrbracket$  avec  $nb_{\text{nœud total}}$  le nombre total de nœuds du maillage, sert de support à au moins une maille ou à un polyèdre d'intersection.

### 3.2.7 Polyèdre conforme par maille

Étant donné que les *hanging nodes* ne sont pas pris en compte par la structure d'origine élément  $\rightarrow$  nœud décrivant les cellules, la définition de dodécaèdre par élément, tel que présenté sur la figure 2.17, n'est plus suffisante dans le cas des maillages modifiés par l'AMR. Une librairie des différents cas rencontrés par face a donc été implémentée et permet de construire des polyèdres conformes<sup>13</sup> par maille pour les intersections. Cette nouvelle librairie est basée sur la présence ou non de nœuds au milieu des arêtes, comme présenté de façon non exhaustive sur la figure 3.10. Elle consiste en une combinaison des quatre cas possibles pour chaque demi-face (figures 3.10(a) à 3.10(g)). Les figures 3.10(b), 3.10(c) et 3.10(d) illustrent les découpes particulières possibles de la facette 1. Les figures 3.10(e), 3.10(f) et 3.10(g) illustrent les découpes particulières possibles de la facette 2. Un cas supplémentaire est défini quand une des deux cellules voisine de la face a été raffinée, tel que représenté sur la figure 3.10(h). Il existe donc 15 cas qui nécessitent un traitement particulier pour des quadrilatères avec présence de *hanging nodes* et 17 cas en tout par face.

**Lemme 1.** *Si une cellule  $\mathcal{T}_\alpha$  possédant 8 nœuds  $\{\mathcal{N}_{\alpha,i}, i \in \llbracket 1, 8 \rrbracket\}$  et un niveau de raffinement **IRAFF**( $\mathcal{T}_\alpha$ ) est telle que :*

$$\max_{i \in \llbracket 1, 8 \rrbracket} \mathbf{IRFMAX}(\mathcal{N}_{\alpha,i}) - \mathbf{IRAFF}(\mathcal{T}_\alpha) = 0$$

*alors elle ne possède pas de hanging nodes.*

La valeur de  $\max_{i \in \llbracket 1, 8 \rrbracket} \mathbf{IRFMAX}(\mathcal{N}_{\alpha,i})$  permet donc d'effectuer un test simple et d'appliquer des traitements spécifiques aux mailles ayant possiblement un polyèdre non conforme. Ce test permet de limiter l'appel de la librairie de découpe des faces et donc le coût algorithmique des intersections de maillage. Il faut noter que la réciproque du Lemme 1 n'est pas vraie ; une maille  $\mathcal{T}_\alpha$  telle que  $\max_{i \in \llbracket 1, 8 \rrbracket} \mathbf{IRFMAX}(\mathcal{N}_{\alpha,i}) - \mathbf{IRAFF}(\mathcal{T}_\alpha) \neq 0$  ne possède pas forcément de *hanging nodes*. Un exemple est donné sur la figure 3.11. La maille verte possède une cellule voisine (en bleu) ayant un nœud en commun de niveau de raffinement supérieur, donc  $\mathbf{IRFMAX} - \mathbf{IRAFF} \neq 0$  pour les nœuds de cellule verte mais elle n'a pas pour autant *hanging nodes*.

13. Ici, chaque polyèdre, lié à une cellule, est vu comme un maillage surfacique d'où l'expression « polyèdre conforme ».

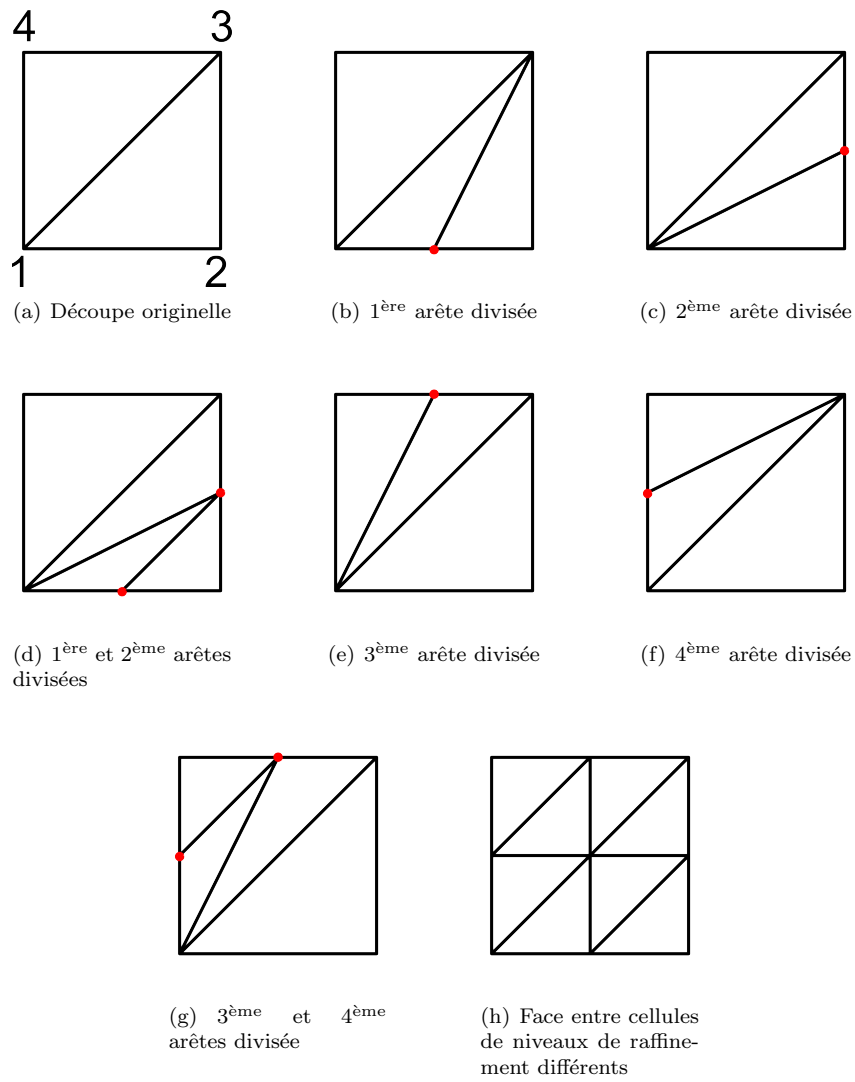


FIGURE 3.10 – Echantillon de la librairie de découpe des faces en fonction de la présence de nœuds sur les arêtes ou d’une maille raffinée

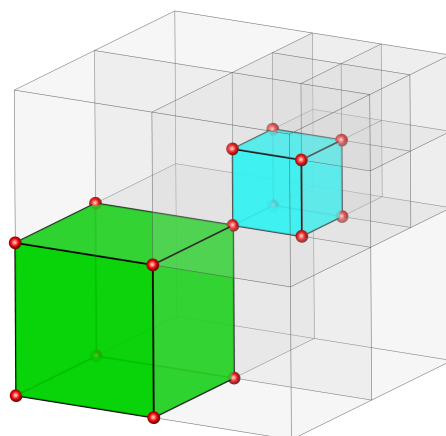


FIGURE 3.11 – Polyèdre conforme avec une maille voisine indirecte raffinée

L’utilisation d’une librairie par face permet de limiter le nombre de cas de polyèdre conforme par maille à traiter. Le dénombrement des polyèdres possibles  $nb_{\text{polydres}}$  connaissant le nombre de faces divisées  $f_{\text{divisées}}$  est effectué dans le tableau 3.1. Ainsi, pour un hexaèdre, il existe au total 6629 cas possibles de polyèdres conformes

avec un équilibrage 2 – 1 par nœud.

$f_{\text{divisées}}$	0	1	2	3	4	5	6
$nb_{\text{polyèdres}}$	4096	1536	480	480	30	6	1

TABLE 3.1 – Dénombrement des polyèdres conformes possibles par maille connaissant le nombre de faces entre mailles de niveaux de raffinements différents

On rappelle que cette définition de polyèdres conformes par maille permet d’obtenir une description naturellement conservative du volume. Elle assure notamment la compatibilité de notre méthode d’adaptation de maillage par enrichissement et autorise l’emploi de méthodes ALE. La figure 3.12(a) représente un polyèdre non conforme et non fermé obtenu après déformation d’une maille. La figure 3.12(b) illustre la même maille déformée mais obtenu à partir d’un polyèdre conforme. La portabilité de la définition des polyèdres par maille pour des méthodes d’adaptation par déformation est ainsi assurée.

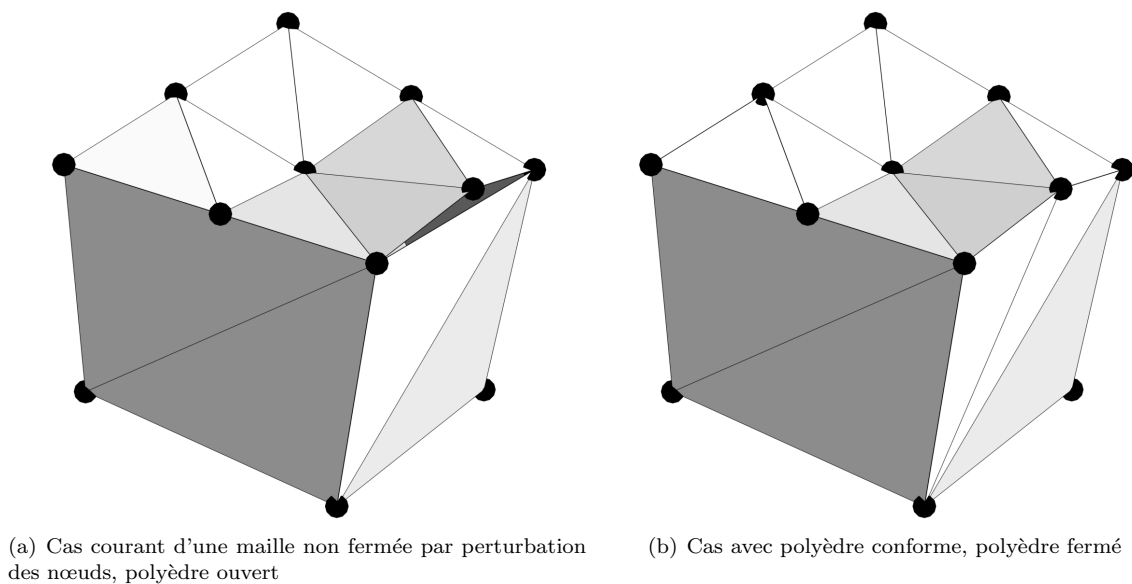


FIGURE 3.12 – Polyèdre par maille après perturbation des nœuds

### 3.2.8 Création des coques

Certaines faces servent à la construction de polyèdres d’intersection : ce sont les faces de conditions aux limites fluides intersectantes ou CLFi (présentées au paragraphe 2.3.4.2). Les calculs d’intersection utilisent les polyèdres formés par l’union de faces CLFi<sup>14</sup>. Si les polyèdres d’intersection sont constitués partiellement ou totalement de quadrilatères, ces quadrilatères sont décomposés en 2 triangles de sorte que la nappe surfacique ne soit composée que de triangles. Les triangles constitutifs de ces surfaces sont appelés des coques. Les coques associées aux nouvelles faces sont créées lors du raffinement de faces servant de support à la définition des nappes surfaciques formant les polyèdres d’intersection. De la même façon que les polyèdres par maille doivent être conformes, les polyèdres servant de support de calcul aux intersections doivent aussi être fermés. Un traitement similaire à celui appliqué pour la création de polyèdres conformes par maille a donc été implémenté. La librairie utilisée est similaire à celle présentée sur la figure 3.10. On présente un cas courant de raffinement d’une face liée à un polyèdre d’intersection et les conséquences de ce raffinement sur la nappes surfaciques des coques sur les figures 3.13(a) et 3.13(b). Le fait de raffiner les deux coques liées à la face rouge, oblige à « propager » le raffinement aux coques voisines pour que le polyèdre reste conforme.

14. Il existe un autre type de face qui génèrent des polyèdres d’intersection, ce sont les faces reliant deux cellules d’une même grille mais ayant des priorités différentes. On parle alors de faces de priorité intra-maillage. Nous n’en utiliserons pas dans le cadre de cette thèse.

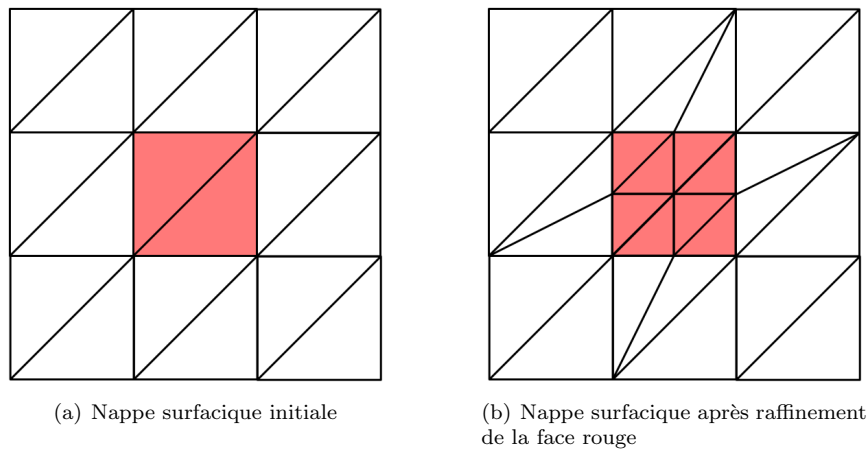


FIGURE 3.13 – Cas d’un raffinement d’une face qui est un élément constitutif d’un polyèdre d’intersection

### 3.2.9 Structure de données

L’ensemble des différentes structures (éléments, faces, nœuds, arêtes) du code est stocké dans des tables de référencement. Cette technique introduit une indirection dans les boucles de parcours des structures géométriques. Pour chaque sous-ensemble, on associe un nombre de cases occupées.

Pour les mailles, les sous-ensembles sont les suivants :

- les mailles vivantes : ce sont les feuilles de la forêt d’*octrees*,
- les mailles mortes : ce sont tous les éléments désactivés par raffinement,
- les mailles trous : ce sont les numéros de cellule libérés après déaffinement.

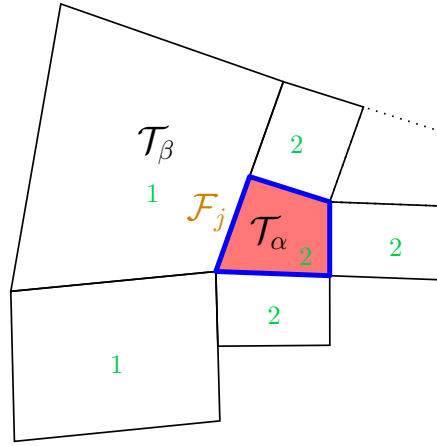
Pour les faces, les sous-ensembles sont les suivants :

- les faces fluides vivantes,
- les faces fluides mortes,
- les faces conditions aux limites fluides,
- les faces conditions aux limites pariétales,
- les faces conditions aux limites fluides mortes,
- les faces conditions aux limites pariétales mortes,
- les faces d’intersection liées à une intersection cellule/paroi,
- les faces d’intersection liées à une intersection cellule/cellule,
- les faces trous.

Les algorithmes de modification de maillage créent et modifient les éléments constitutifs de ceux-ci et donc l’ensemble de ces listes est constamment mis à jour en fonction du déroulé des calculs. Les algorithmes de modification de maillage ne comprennent pas de renumérotation dynamique.

### 3.2.10 Accès mémoire aux sous-faces

Le fait de manipuler des maillages non conformes entraîne des accès mémoire particuliers. Le tableau élément  $\rightarrow$  face est défini de la façon suivante : si l’on cherche à obtenir la  $j^{\text{ème}}$  face locale d’un élément  $\mathcal{T}_\alpha$  alors la face  $\mathcal{F}_j$  reliant  $\mathcal{T}_\alpha$  à  $\mathcal{T}_\beta$  est telle  $\mathbf{IRAFF}(\mathcal{T}_\alpha) \geq \mathbf{IRAFF}(\mathcal{T}_\beta)$  avec  $\mathbf{IRAFF}$  la variable indiquant le niveau de raffinement d’une cellule. Un cas courant est présenté en 2D sur la figure 3.14. On représente en rouge la cellule  $\mathcal{T}_\alpha$  et en bleu les faces de  $\mathcal{T}_\alpha$  renvoyées par la structure élément  $\rightarrow$  face. On précise pour plus de clarté en vert le niveau de raffinement des voisins de  $\mathcal{T}_\alpha$ . En particulier, on souligne que la face  $\mathcal{F}_j$  entre les éléments  $\mathcal{T}_\alpha$  et  $\mathcal{T}_\beta$  vérifie la propriété  $\mathbf{IRAFF}(\mathcal{T}_\alpha) \geq \mathbf{IRAFF}(\mathcal{T}_\beta)$ .

FIGURE 3.14 – Voisinage de la cellule  $\mathcal{T}_\alpha$ , en vert niveau de raffinement des cellules

Avec une telle structure élément  $\rightarrow$  face, quand  $\mathbf{IRAFF}(\mathcal{T}_\alpha) = \mathbf{IRAFF}(\mathcal{T}_\beta)$  mais que  $\mathcal{T}_\beta$  est une cellule morte, la face reliant  $\mathcal{T}_\alpha$  à  $\mathcal{T}_\beta$  est une face morte. L'accès aux sous-faces en lesquelles  $\mathcal{F}$  a été divisée n'est pas direct. On présente schématiquement sur la figure 3.15 le principe en plusieurs étapes d'un accès mémoire pour une cellule dont une des faces  $\mathcal{F}$  est morte. On se place du point de vue de la cellule  $\mathcal{T}_\alpha$  et on effectue une boucle sur ses faces. Un premier accès mémoire permet d'obtenir le numéro de la cellule  $\mathcal{T}_\beta$  de l'autre côté de la face comme présenté sur la figure 3.15(a). Si le marqueur créé pour l'AMR indique que cette cellule est raffinée, alors la face  $\mathcal{F}$  qui relie  $\mathcal{T}_\alpha$  à  $\mathcal{T}_\beta$  est une face morte qui a été remplacée par des sous faces  $\mathcal{F}_i$  comme présenté sur la figure 3.15(b). On cherche alors à avoir accès à ces faces  $\mathcal{F}_i$ . L'arbre *octree* permet d'accéder aux mailles filles  $\mathcal{T}_{\beta,i}$  de  $\mathcal{T}_\beta$ , voir figure 3.15(c). Ces mailles  $\mathcal{T}_{\beta,i}$  permettent alors d'obtenir les faces correspondantes  $\mathcal{F}_i$  qui ont remplacé la face morte  $\mathcal{F}$ .

### 3.2.11 Espace mémoire occupé par la méthode AMR

Les tableaux modifiés par le module AMR sont remplis de façon assez simple. La suppression d'éléments crée des espaces libres non utilisés, ces cases libres sont répertoriées et remplies jusqu'à ce qu'il n'y ait plus de trous (cf. figure 3.16(a)). S'il n'y a plus de trous, alors l'espace non utilisé après l'élément ayant le plus grand indice est rempli (cf. figure 3.16(b)). Pour l'instant, il n'existe pas de mécanisme de suppression des trous en cours de calcul, ce type de routine n'est pas nécessaire pour le moment. Toutefois, d'autres optimisations pour l'occupation de l'espace mémoire sont envisagées, voir Chapitre 6.

L'espace mémoire des structures de données permettant la gestion d'un arbre de relation entre éléments et des listes de faces, éléments, nœuds ramené au nombre de cellules stockées est de l'ordre de 50% pour le schéma spatial d'ordre 2 et de 15% pour le schéma spatial d'ordre 3.

## 3.3 Processus aérodynamiques (esclaves)

### 3.3.1 Structure de données des processus aérodynamiques

Les esclaves utilisent une version topologiquement simplifiée des maillages. Les éléments et les faces sont représentés uniquement par le point qui minimise la norme des moments d'ordre 1, et leurs moments en ce point<sup>15</sup>. Le stockage ainsi défini ne nécessite pas de structure élément  $\rightarrow$  nœud. Chaque processus aérodynamique possède des routines permettant d'effectuer une renumérotation persistante à partir des numéros des cellules du maître qui lui sont envoyées afin d'optimiser l'adressage dans les différents tableaux. On appelle numérotations globales ou de référence, les numérotations du maître et numérotations locales, les numérotations des tableaux sur les processus aérodynamiques. La renumérotation persistante prend en compte le schéma à pas de temps

15. Pour les volumes et les faces planes, il s'agit tout simplement de leurs centres de gravité; pour les faces non planes, ce point caractéristique minimise l'erreur d'ordre 1 d'intégration des flux cf. [PBCR17].

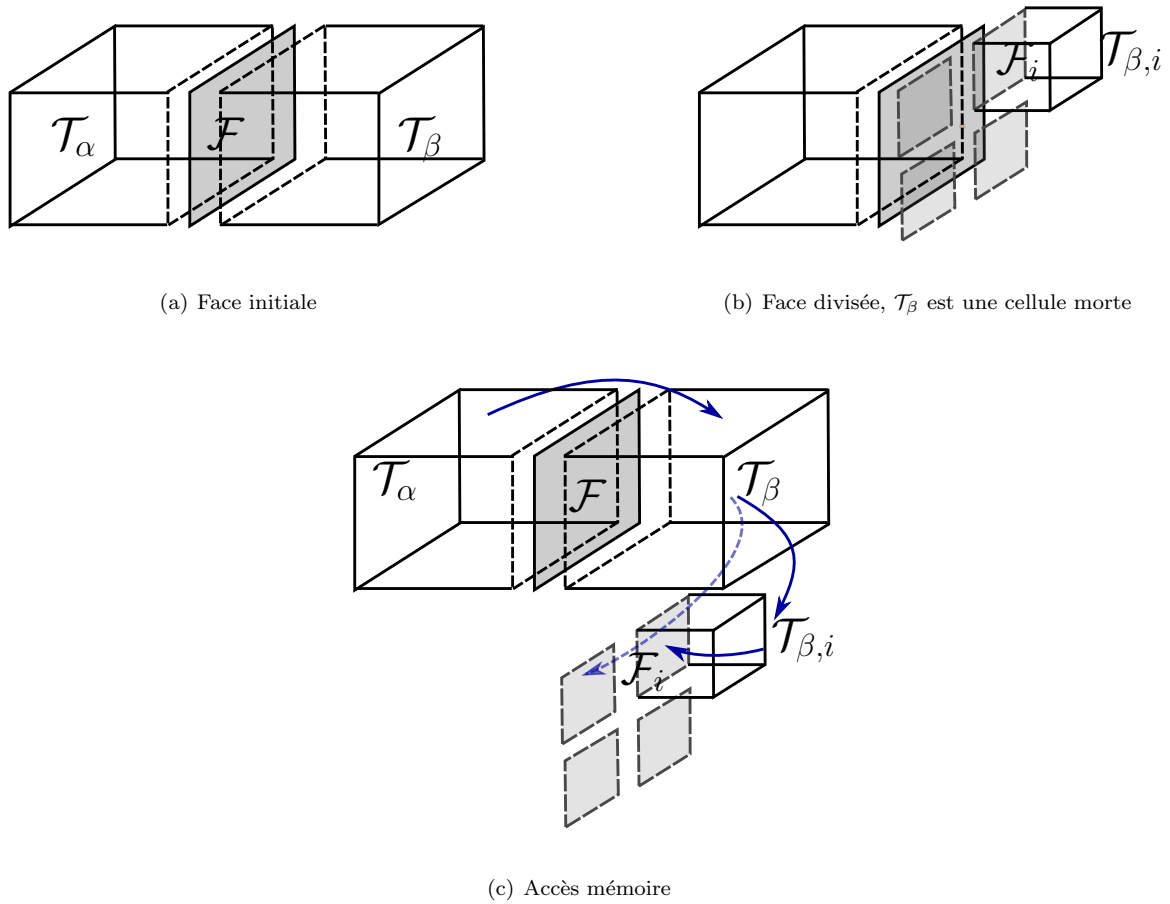


FIGURE 3.15 – Schéma du cheminement logique d'un accès mémoire aux sous-faces d'une face divisée

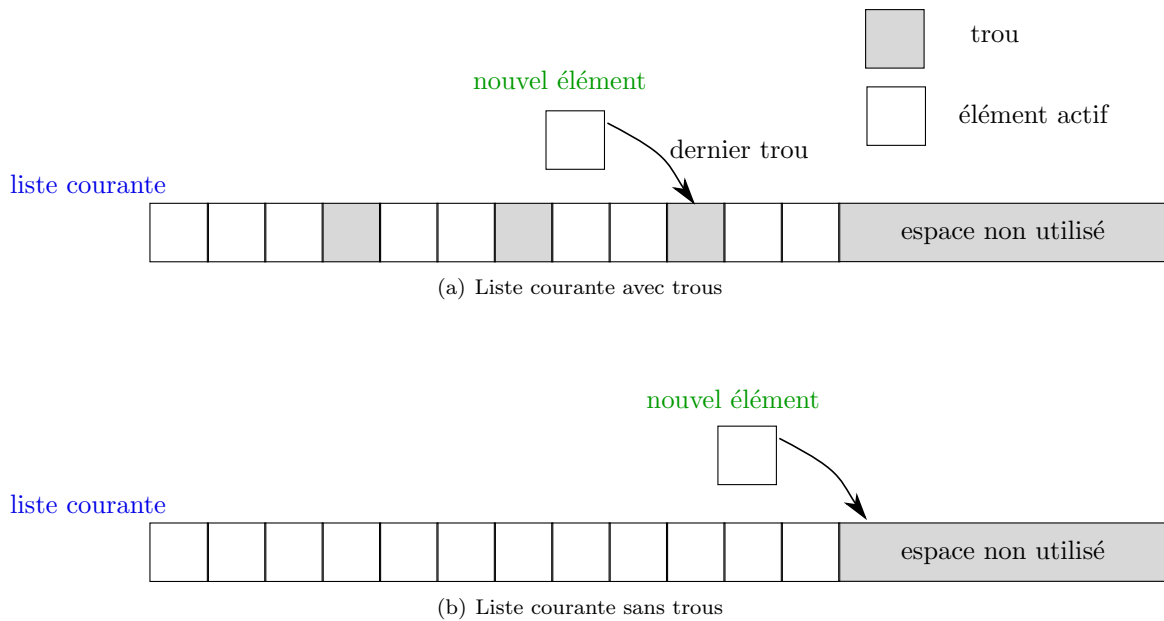


FIGURE 3.16 – Schéma de principe de l'ajout d'un objet dans un tableau contenant des trous

local en regroupant les cellules par niveau temporel et en regroupant les cellules de bord de niveau temporel<sup>16</sup>.

16. Les cellules de bord de niveau temporel possèdent une interface avec une cellule d'un autre niveau temporel (cf. paragraphe 2.2.2).

### 3.3.2 Description algorithmique

Le pseudo-code permettant d'effectuer une adaptation de maillage sur un processus aérodynamique est donné dans l'algorithme 7.

---

**Algorithme 7** Adaptation de maillage sur les esclaves
 

---

```

1: si ( $nb_{\text{maille raff}} > 0$  ou  $nb_{\text{maille deraff}} > 0$ ) alors
2:   Envoi de la liste des mailles à raffiner/déraffiner
3: fin si
4: Réception des actions à effectuer
5: si Remaillage effectué par le maître alors
6:   Réception de la liste des mailles raffinées et/ou déraffinées
7:   Réception des propriétés géométriques des mailles créées par raffinement et des mailles réactivées par déraffinement
8:   Reconstruction des mailles mères raffinées par assemblage des mailles filles créées
9:   Redistribution conservative à l'ordre choisi par l'utilisateur des variables conservatives des mailles mères raffinées sur leurs mailles filles créées par raffinement
10:  Assemblage des variables conservatives des mailles venant d'être déraffinées
11: fin si

```

---

Les processus aérodynamiques ne disposent que d'une topologie partielle du maillage et n'effectuent pas à proprement parler d'adaptation de maillage. Les esclaves reçoivent les cellules qui ont été modifiées par le maître. L'introduction d'arbres de relation entre les éléments doit donc être aussi légère en terme d'espace mémoire que possible. Les esclaves n'ont pas pour fonction d'effectuer des opérations topologiques sur les maillages, ils sont optimisés uniquement pour le calcul du champ des variables primitives  $q$  (*cf.* paragraphe 2.1.1). Ainsi, deux propriétés ont été définies pour limiter l'espace mémoire occupé par l'adaptation de maillage sur les esclaves :

1. les structures d'arbre parents/enfants ne sont pas gardées en mémoire d'une itération d'adaptation de maillage à une autre,
2. les arbres de relation sont de tailles variables pour chaque élément selon le nombre de cellules non-couvertes créées par raffinement, ou le nombre de sœurs non couvertes avant déraffinement.

La raison de la deuxième propriété énoncée est illustrée en 2D sur les figures 3.17(a) et 3.17(b). Prenons le cas d'une cellule  $\mathcal{T}_\alpha$  et d'un polyèdre  $\mathcal{P}$  représentés sur la figure 3.17(a). La partie grisée représente le volume occupé par le polyèdre  $\mathcal{P}$ . Supposons que  $\text{PRIORITE}(\mathcal{T}_\alpha) < \text{PRIORITE}(\mathcal{P})$ . Par conséquent la cellule  $\mathcal{T}_\alpha$  est partiellement couverte par le polyèdre  $\mathcal{P}$ . Supposons maintenant que le maître ait raffiné la cellule  $\mathcal{T}_\alpha$  en 4 cellules  $\{\mathcal{T}_{\alpha,i}, i \in \llbracket 1, 4 \rrbracket\}$ , figure 3.17(b). Ici, la fille notée  $\mathcal{T}_{\alpha,3}$ , représentée en rouge, est complètement couverte et possède un volume nul après calcul des intersections. Dans le cas ainsi présenté, l'arbre de relation entre la cellule  $\mathcal{T}_\alpha$  et ses filles ne nécessite que 3 cases mémoires et non 4 comme cela est fait sur le maître.

On représente schématiquement sur la figure 3.18 une forêt d'arbres créée au cours d'une adaptation de maillage. Les esclaves construisent, après réception de la liste des mailles modifiées, des arbres différents de ceux manipulés par le maître. Les arbres de relation mère  $\rightarrow$  fille sont de taille variable  $nb_{\text{mf}}$ , comprise entre 0 et 8, selon le nombre de cellules non-couvertes après raffinement. Pour le déraffinement le problème est similaire, la taille des arbres est fonction du nombre de sœurs non couvertes que l'on souhaite déraffiner. Comme présenté sur la figure 3.18, les esclaves ne possèdent pas de notion de niveau de raffinement, toutes les cellules sont initialement supposées de même niveau de raffinement. Sans notion de niveau de raffinement sur les esclaves, il n'est pas possible d'effectuer d'équilibrage d'arbres par conséquent l'équilibrage est supposé avoir été assuré par le maître. L'espace mémoire  $E$  occupé par les variables temporaires créées lors de ces modifications de maillage est proportionnel à :

$$T \propto nb_{\text{cell deraff}} + \sum_{i=1}^{nb_{\text{cell raff}}} nb_{\text{mf}}(i) \quad (3.3)$$



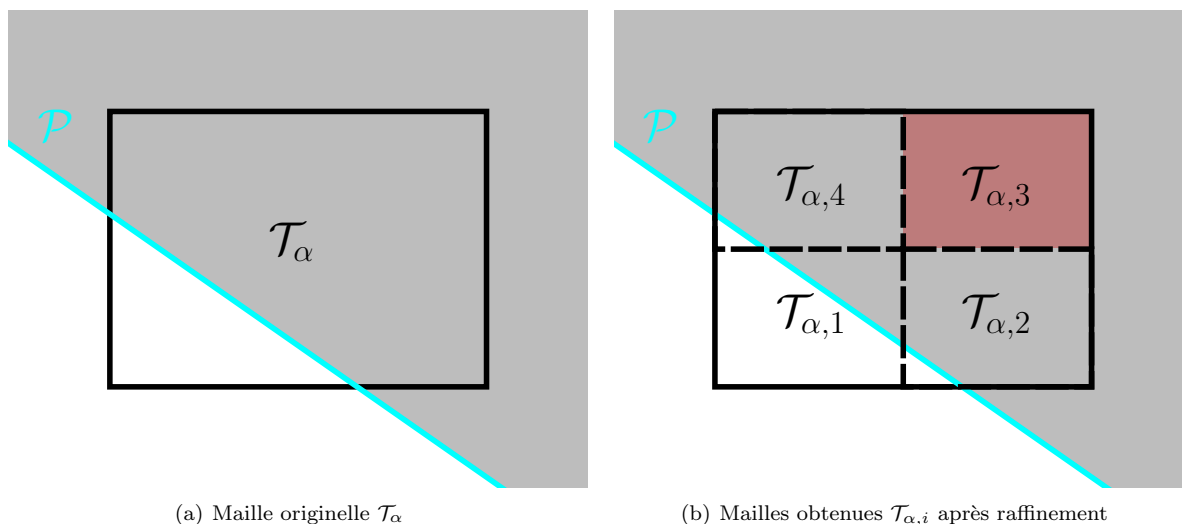


FIGURE 3.17 – Raffinement d’une maille  $\mathcal{T}_\alpha$  partiellement couverte par un polyèdre  $\mathcal{P}$ , partie grisée

avec  $nb_{mf}(i)$  le nombre de mailles filles de la cellule  $i$ .

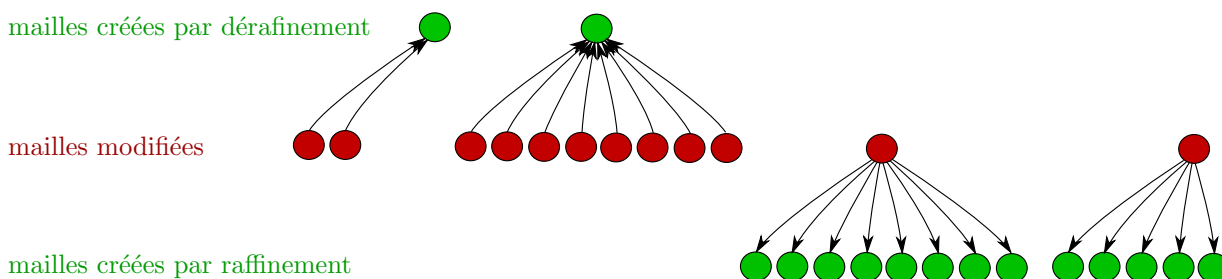


FIGURE 3.18 – Arbre de relation local d’un processus aérodynamique au cours d’une adaptation de maillage

### 3.3.3 Boucles de parcours des faces pour les maillages non-conformes

L’implémentation d’un module d’adaptation de maillage ne modifie pas le noyau des schémas numériques. En effet, les schémas numériques de FLUSEPA utilisent une formulation *cell-centered* et n’utilisent pas de liens cellule  $\rightarrow$  face, mais uniquement des liens face  $\rightarrow$  cellule. Il suffit donc de traiter les bornes des boucles de parcours des faces et de s’assurer que les différentes tables qui les répertorient soient toujours construites selon des règles qui garantissent le bon fonctionnement du solveur aérodynamique.

Historiquement, afin de pouvoir effectuer des calculs sur des processeurs vectoriels puis sur processeurs *multithread*, il a été choisi de trier les listes de faces du maillage dans deux tableaux  $LISTF_G$  et  $LISTF_D$  subdivisés respectivement en  $n_{born_G}$  et  $n_{born_D}$  segments dont les bornes sont regroupées dans les tableaux  $born_G$  et  $born_D$ . En choisissant une orientation des faces, on désigne « à gauche » la cellule située à l’origine du vecteur surface et « à droite » la cellule vers laquelle le vecteur surface pointe. Les tableaux associés sont alors respectivement  $LISTF_G$  et  $LISTF_D$ .

Les tableaux ainsi construits doivent garantir la règle suivante : si une cellule est référencée à gauche (ou à droite) pour une certaine face alors cette cellule n’apparaîtra qu’une seule fois entre les deux bornes d’un segment. Ce Tri noté TSID est donc Segmenté, Injectif et uni-Directionnel. Il permet de coder des boucles compactes de grande taille pouvant être parcourues de manière efficace des machines vectorielles et parallèles à mémoire

partagée. En particulier, des directives de parallélisme *multithread*, basées sur le standard OPEN-MP, peuvent être utilisées. Les algorithmes de parcours des faces se présentent alors sous la forme générique de l’algorithme 8.

---

**Algorithme 8** Boucle de modification de cellules à gauche des faces
 

---

```

1: pour  $n_{seg} = 1$  à  $n_{born_G}$  faire
2:   début fourche parallèle multithread
3:   pour  $ind = born_G(n_{seg} - 1) + 1$  à  $born_G(n_{seg})$  faire
4:      $n_{cell} = cell_G(LISTF_G(ind))$ 
5:      $q(n_{cell}) = q(n_{cell}) + \Delta q$ 
6:   fin pour
7:   fin fourche parallèle multithread
8: fin pour
  
```

---

Dans le cas courant sans chevauchement comprenant des maillages constitués uniquement d’hexaèdres, le tri s’effectue sur  $n_{born_{G/D}} = 6$  segments. Dans un cas avec chevauchement et intersections de maillages, le nombre de segments est augmenté de façon à prendre en compte les cellule possédant plus de 6 faces. Le nombre de segments supplémentaires est lié au nombre maximum de faces d’intersection avec les cellules d’autres maillages. Pour une cellule donnée, on note ce nombre  $n_{niv\ inter}$ . On a alors  $n_{born_{G/D}} = n_{niv\ inter} + 6$ . On isole sur la figure 3.19 une cellule  $\Omega_1$  en rouge intersectée par un maillage  $\mathcal{M}_2$  ayant une priorité supérieure. On représente chacune des 7 faces d’intersection de  $\Omega_1$  avec  $\mathcal{M}_2$ , on a donc  $n_{niv\ inter} \geq 7$ .

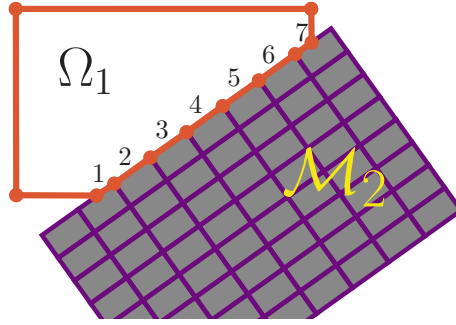


FIGURE 3.19 – Cellule  $\Omega_1$  intersectée par un maillage  $\mathcal{M}_2$  de priorité supérieure, numérotation locale des faces d’intersection obtenues

Dans le cas où le maillage traité par les esclaves a été raffiné du fait de l’adaptation de maillage, il est nécessaire de :

- modifier  $n_{born_{G/D}}$  pour avoir  $n_{born_{G/D}} = n_{niv\ inter} + 30$ . Ici  $30 = 24 + 6$  où 24 représente le nombre de sous-faces possibles et 6 les faces « de base » d’un hexaèdre,
- retirer des  $LISTF_X$  les faces mortes,
- attribuer un numéro compris entre 7 et 30 pour les faces supplémentaires vivantes sur les cellules possédant des faces mortes. La figure 3.20 présente l’exemple d’une numérotation locale déduite pour la face locale 1 de la cellule  $\mathcal{T}_\alpha$ .

Cette section décrit l’unique modification qui a été faite à l’algorithmique du schéma d’intégration des flux numériques de FLUSEPA. Ainsi, l’utilisation de maillages raffinés par le module AMR n’introduit pas d’inconsistance dans le calcul des flux. Le flux sur la face 1 de la figure 3.20(a) n’est plus calculé. Après raffinement, les flux des faces 7, 8, 9 et 10 de la figure 3.20(b) remplacent ce dernier. En effet, la méthode utilisée par le schéma numérique est liée au principe même des Volumes Finis qui consistent en l’intégration d’équations de bilan sur des volumes de contrôle et ce, qu’importe le nombre de faces que ceux-ci possèdent. La figure 3.19 montre bien que FLUSEPA supporte intrinsèquement l’usage de polyèdres généraux. C’est une nécessité liée l’utilisation d’un module d’intersection de maillage qui crée des faces d’intersections pour les éléments initiaux (hexaèdres, prismes, pyramides, tétraèdres).

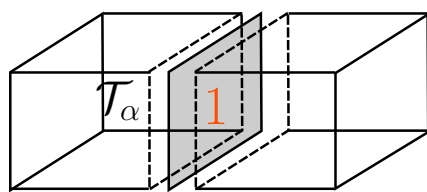
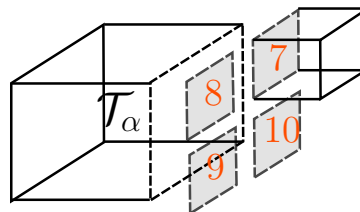
(a) Face initiale, numérotation locale pour  $\mathcal{T}_\alpha$ (b) Face découpée, numérotation locale déduite pour  $\mathcal{T}_\alpha$ 

FIGURE 3.20 – Exemple de la division de face et numérotation locale

### 3.4 Conclusion

Ce chapitre a présenté la stratégie d'intégration du module d'adaptation de maillage par raffinement *octree* dans le code industriel FLUSEPA. La cohérence de ce module avec les caractéristiques propres du code a permis de définir un ensemble de routines évolutives (possibilité d'ajout d'arbres de relation quad- et di-*tree*). Le cœur du schéma numérique d'intégration n'a pas été modifié par la présence de zones de maillages raffinées. En effet, les routines de calcul des intersections géométriques et la gestion des faces d'intersection créées permettaient déjà la manipulation de polyèdres généraux. Le schéma temporel adaptatif explicite de FLUSEPA étant similaire aux schémas rencontrés dans les méthodes AMR, l'intégration en temps explicite pour des écoulements instationnaires était donc déjà performante et n'a pas nécessité de développements spécifiques. La création d'arbres de relation par cellule a augmenté le coût de stockage par élément, mais ce surcoût est compensé par la réduction du nombre d'éléments nécessaires à précision équivalente et une diminution du temps ingénieur requis pour la mise en place d'un calcul (génération du maillage et données initiales plus légères).

En outre, il est important de souligner que FLUSEPA a été conçu dès l'origine dans une optique de gestion dynamique de maillages à topologie variable. Cette caractéristique propre à FLUSEPA a permis à la logique du module AMR présenté dans cette thèse de s'intégrer naturellement au code préexistant. En particulier, le principe de séparation du calcul des propriétés géométriques (volume, surface, moments géométriques, ...) de celui des variables physiques (pression, température, ...) a été conservé. Ainsi, l'ensemble du code effectuant le remaillage est réparti sur plusieurs modules tous intégrés directement dans FLUSEPA. Les routines d'adaptation construisent des arbres de relation selon les besoins des processus ; par exemple les solveurs aérodynamiques créent des arbres temporaires liés simplement aux modifications du maillage. Le découpage du maillage effectué par le processus maître à partir du maillage initial repose sur la pondération de chaque maille du maillage initial selon le nombre de feuilles de l'arbre *octree* courant qui lui est associé.

La version actuelle de l'AMR impose des surcoûts non négligeables, notamment le calcul intégral de la topologie (moments géométriques) et surtout le calcul des intersections après chaque modification du maillage ; les propriétés géométriques des éléments nouvellement créés ne pouvant pas être déduites des propriétés des éléments auxquels ils sont apparentés. L'ensemble ainsi formé est néanmoins robuste et facilement modifiable. En définissant l'ensemble des fonctions élémentaires pour le raffinement et la fusion de cellule, le module AMR développé permettra de faciliter la mise au point de nouvelles techniques d'adaptation de maillage par raffinement. De la sorte, les premiers résultats de calcul avec raffinement de maillage adaptatif sont exposés dans le chapitre 5.

# Chapitre 4

## Processus de redistribution conservative d'ordre élevé des variables sur une restriction d'un volume

Les chapitres précédents ont insisté sur l'importance de la cohérence globale de la stratégie de simulation numérique. Différentes études [Pon15, PBCR17] ont analysé la précision du code FLUSEPA et plus particulièrement de son schéma numérique qui utilise une reconstruction  $k$ -exacte par maille. Cette reconstruction est exacte si la solution du problème étudié est un polynôme d'ordre  $k$ . Ce chapitre répond à la problématique posée par la subdivision d'une maille en un ensemble de sous-maillages et à la redistribution associée des variables dans ces dernières<sup>1</sup>. Nous construisons dans ce chapitre une méthode de redistribution conservative, précise et bornée des variables conservatives, consécutive à cette subdivision et en validons l'implémentation sur des cas tests analytiques simples.

### 4.1 Positionnement du problème

Afin d'exposer simplement la problématique posée, nous définissons en 1D une fonction de référence d'une variable, par exemple la masse volumique et un maillage sur lequel on souhaite représenter cette fonction de façon discrète (figure 4.1). A partir de ce maillage et de cette fonction de référence, on calcule la valeur moyenne de la masse volumique dans chaque élément (figure 4.1 en bleu).<sup>2</sup> La valeur conservative de la variable considérée dans chaque élément, ici la masse, permet de reconstruire localement à l'ordre souhaité. Nous représentons une reconstruction affine et une reconstruction parabolique, en magenta et en vert respectivement sur la figure 4.1. Ces méthodes de reconstruction s'apparentent à la méthode MUSCL qui permet la montée en ordre globale de la solution.

Le maillage proposé est très largement sous-résolu et ne permet pas une représentation correcte de la fonction de référence. On se propose de raffiner la cellule centrale pour augmenter la résolution locale du maillage au niveau du pic local. Nous allons étudier l'influence de l'ordre de la méthode de redistribution des variables lors du raffinement d'une maille. La conservativité est garantie si la somme des valeurs des variables conservatives redistribuées sur les  $p$  filles  $|\Omega_i| \bar{w}_i$  est égale<sup>3</sup> à la valeur de la variable conservative sur la mère  $|\Omega_m| \bar{w}_m$ . Cette première condition permet de simplement redistribuer le champ des variables conservatives des cellules filles en posant :

---

1. Notons que dans la plupart des méthodes AMR, les champs des nouvelles cellules sont obtenus par interpolation (souvent bi/tri-linéaire, ce qui n'est pas forcément consistant avec la méthode de reconstruction utilisée et non conservatif (cf. section 1.3).

2. En s'appuyant sur cette valeur moyenne pour le calcul des flux, on se place dans le cadre des méthodes des Volumes Finis de Godounov et chaque interface du maillage définit alors un problème de Riemann, pour lequel il est possible de calculer exactement la solution.

3. Ici  $w = (\rho, \rho \mathbf{u}, \rho E)$  est le vecteur des variables conservatives (cf. paragraphe 2.1.1).

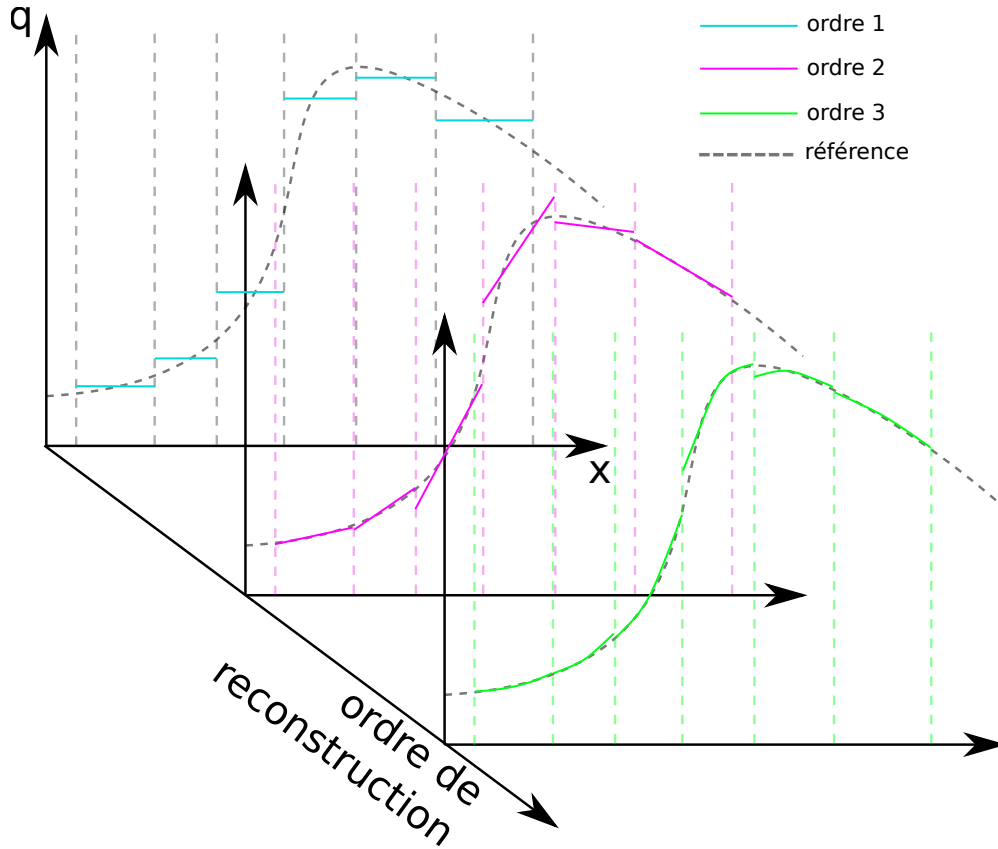


FIGURE 4.1 – Maillage 1D avec fonction de référence et différentes méthodes de reconstruction selon leur ordre

$$|\Omega_m| \bar{w}_m = \sum_{i=1}^p |\Omega_i| \bar{w}_i. \quad (4.1)$$

Une telle redistribution des variables est alors d'ordre 1 et conservative. Cependant, cette redistribution n'est pas suffisante pour des reconstructions d'ordre supérieur dans la mesure où elle n'est pas compatible avec la reconstruction utilisée par le schéma d'intégration numérique. Les figures 4.2(a), 4.2(b) et 4.2(c) représentent les reconstructions sur les variables primitives obtenues après redistribution d'ordre 1, 2 et 3 respectivement dans les nouvelles cellules créées par raffinement de la cellule grisée. Pour les redistributions d'ordre 2 et 3, on représente aussi en cyan la redistribution d'ordre 1 et en noir les valeurs moyennes volumiques obtenues par cellule pour chacune des redistributions. On constate qu'effectuer une redistribution d'un ordre inférieur à celui de la reconstruction dégrade la précision du schéma global et provoque potentiellement des discontinuités parasites importantes lors du calcul de l'évolution de l'écoulement. Dans le cas d'une adaptation extrêmement fréquente, par exemple toutes les itérations lié à des critères très sélectifs<sup>4</sup>, cette inconsistance entre ordre de précision de la reconstruction du champ des variables (lié au schéma d'intégration numérique) et ordre de précision de la projection lors de la subdivision devient extrêmement problématique.

## 4.2 Méthode

En partant des variables conservatives  $\bar{w} = (\bar{\rho}, \bar{\rho\mathbf{u}}, \bar{\rho E_t})$  moyennées en volume<sup>5</sup>, les relations 2.38 permettent de passer simplement de ces dernières aux variables primitives brutes  $\tilde{q} = f(\bar{w})$ <sup>6</sup>. Ainsi, nous travaillerons uniquement sur le vecteur  $\tilde{q}$  des variables primitives brutes  $\tilde{q} = (\tilde{P}, \tilde{T}, \tilde{\mathbf{u}})$ . On se place dans le cadre de la méthode

4. Pour être efficace, il est nécessaire de limiter le sur-raffinement de certaines zones

5. Ce sont les variables de travail.

6. Par définition.

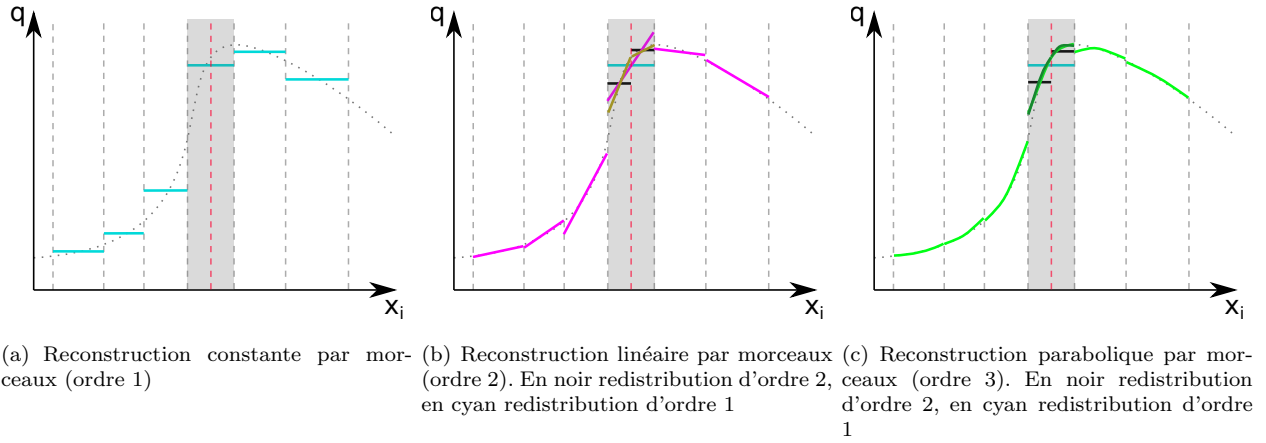


FIGURE 4.2 – Maillage 1D avec fonction de référence. Redistribution d'ordre consistante avec l'ordre de la méthode de reconstruction des variables

de reconstruction d'une variable par corrections successives décrite Brenner [Bre08], puis Haider [Hai09] et Pont [Pon15] et les notations utilisées sont celles introduites dans le paragraphe 2.2.1.2.

Soit  $\mathcal{T}_\alpha$  une cellule et  $\mathcal{T}_{\alpha_1}, \mathcal{T}_{\alpha_2}$  deux sous ensembles de  $\mathcal{T}_\alpha$  tels que  $\mathcal{T}_{\alpha_1} \cup \mathcal{T}_{\alpha_2} = \mathcal{T}_\alpha$  comme présenté sur la figure 4.3. On cherche alors à redistribuer la valeur de la variable extensive dans chacun des deux sous-volumes  $\mathcal{T}_{\alpha_i}, i = 1, 2$ . On note  $G_\alpha, G_{\alpha_1}$  et  $G_{\alpha_2}$ , le centre de gravité de  $\mathcal{T}_\alpha, \mathcal{T}_{\alpha_1}$  et  $\mathcal{T}_{\alpha_2}$  respectivement.

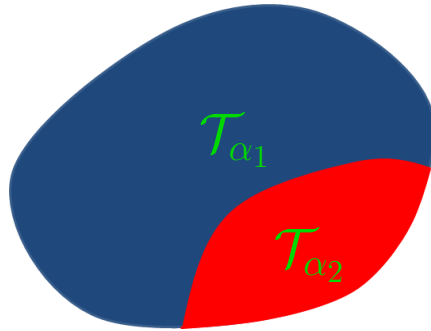


FIGURE 4.3 – Partition en deux sous-ensembles  $\mathcal{T}_{\alpha_1}$  et  $\mathcal{T}_{\alpha_2}$  de la cellule  $\mathcal{T}_\alpha$

On rappelle des tenseurs de dérivées :

$$\left( D^{(j)} q \right)_{i_1 \dots i_j} = \frac{\partial^j q}{\partial x_{i_1} \partial x_{i_j}} .$$

Supposons que l'on dispose de l'ensemble des variables géométriques permettant la description de chaque cellule pour une méthode d'ordre  $n$ , à savoir :  $\{v_\alpha^{(i)}, i \in \llbracket 1, n-1 \rrbracket\}$ . Supposons que l'on possède aussi l'ensemble des tenseurs des dérivées :  $\{D^{(i)} q, i \in \llbracket 1, n-1 \rrbracket\}$ . Afin d'affecter une valeur  $\tilde{q}_{\alpha_1}$  à la cellule  $\mathcal{T}_{\alpha_1}$  à l'ordre  $n$ , on cherche à obtenir une équation du type :

$$\tilde{q}_{\alpha_1} = \tilde{q}_\alpha + g \left( v_\alpha^{(i)}, v_{\alpha_1}^{(i)}, D^{(i)} w \right) + \mathcal{O}(h^n), \quad i \in \llbracket 1, n-1 \rrbracket \quad (4.2)$$

avec  $h$  une taille caractéristique de  $\mathcal{T}_\alpha$  et  $g$  une fonction à déterminer.

Avec les relations précédemment introduites, il est judicieux de mettre la relation sous la forme :

$$\begin{aligned} g(v_\alpha^{(i)}, v_{\alpha_1}^{(i)}, D^{(i)}w) + \mathcal{O}(h^n) &= \tilde{q}_{\alpha_1} - \tilde{q}_\alpha + \mathcal{O}(h^n) \\ &= \left( \underbrace{\tilde{q}_{\alpha_1} - \bar{q}_{\alpha_1}}_{A_1} + \underbrace{\bar{q}_{\alpha_1} - q_{G_{\alpha_1}} + q_{G_{\alpha_1}}}_{B_1} \right) \\ &\quad - \left( \underbrace{\tilde{q}_\alpha - \bar{q}_\alpha}_{A_2} + \underbrace{\bar{q}_\alpha - q_{G_\alpha} + q_{G_\alpha}}_{B_2} \right) + \mathcal{O}(h^n) \end{aligned} \quad (4.3)$$

$$= A_1 - A_2 + B_1 - B_2 + \underbrace{(q_{G_{\alpha_1}} - q_{G_\alpha})}_C + \mathcal{O}(h^n), \quad (4.4)$$

les termes  $A_1$  et  $A_2$  sont déduits de la relation 2.45, les développements sont donnés en annexe C et aboutissent aux relations C.4, C.10, C.17. Ces équations donnent les formules pour le passage de  $\tilde{q}$  à  $\bar{q}$  et utilisent uniquement les moments volumiques d'ordres 1 à  $n - 1$  et les tenseurs des dérivées première à  $(n - 1)$ -ième des variables primitives. En particulier, pour une reconstruction d'ordre 3, ces relations montrent qu'il est possible à partir des valeurs de gradients évalués à l'ordre 2 et des dérivées secondes évaluées à l'ordre 1, d'établir des relations d'ordre 3 sur les variables primitives. Les termes  $B_1$  et  $B_2$  sont déduits de la relation 2.43.

Le terme  $C$  restant est évalué en effectuant un développement de Taylor au sein de la cellule  $T_\alpha$  en  $G_\alpha$  :

$$q(\mathbf{x}) = \sum_{j=0}^{k-1} \frac{1}{j!} D^{(j)}q \Big|_{G_\alpha} : (\mathbf{x} - \mathbf{x}_{G_\alpha})^{\otimes j} + \mathcal{O}(h^k)$$

ou encore :

$$q(\mathbf{x}) = q(\mathbf{x}_{G_\alpha}) + \sum_{j=1}^{k-1} \frac{1}{j!} D^{(j)}q \Big|_{G_\alpha} : (\mathbf{x} - \mathbf{x}_{G_\alpha})^{\otimes j} + \mathcal{O}(h^k) \quad (4.5)$$

ce qui donne, en notant  $\mathbf{x}_{G_\alpha G_{\alpha_1}} = \mathbf{x}_{G_{\alpha_1}} - \mathbf{x}_{G_\alpha}$  :

$$C = q(\mathbf{x}_{G_{\alpha_1}}) - q(\mathbf{x}_{G_\alpha}) = \sum_{j=1}^{k-1} \frac{1}{j!} D^{(j)}q \Big|_{G_\alpha} : \mathbf{x}_{G_\alpha G_{\alpha_1}}^{\otimes j} + \mathcal{O}(h^k). \quad (4.6)$$

Nous avons ainsi détaillé l'ensemble des termes de la relation 4.4 et décrit une méthode de redistribution d'ordre  $p$  depuis un volume  $\mathcal{T}_\alpha$  vers un volume  $\mathcal{T}_{\alpha_1} \subset \mathcal{T}_\alpha$ . En appliquant cette formule, l'ordre global est préservé et les informations fournies directement par le schéma numérique sont exploitées.

L'extension à un ensemble de sous-volumes  $\{\mathcal{T}_{\alpha_i}, i \in \llbracket 1, p \rrbracket\}$  tels que  $\bigcup_{i=1}^p \mathcal{T}_{\alpha_i} = \mathcal{T}_\alpha$  se fait naturellement avec la méthode ainsi introduite. Lors d'une subdivision, liée à un raffinement du module AMR ou d'un désassemblage, lié à la présence d'une maille fortement couverte par un maillage de priorité supérieure, ou de la combinaison des deux, il est possible de redistribuer simplement les variables au sein de l'ensemble des sous-maillages.

### 4.2.1 Procédé de redistribution conservative de l'erreur

Supposons que l'on ait effectué une redistribution à un ordre  $n$  des différentes variables primitives brutes  $\tilde{q}$  depuis une cellule  $\mathcal{T}_\alpha$  à un ensemble de sous-cellules  $\{\mathcal{T}_{\alpha_i}, i \in \llbracket 1, p \rrbracket\}$ . Après passage aux variables conservatives  $w$ , on obtient une relation de la forme :

$$|\Omega_m| \bar{w}_\alpha = \sum_{i=1}^p |\Omega_{\alpha_i}| \bar{w}_{\alpha_i} + \epsilon_w, \quad (4.7)$$

avec  $w_\alpha$  et  $w_{\alpha_i}$  le vecteur des variables conservatives de  $\mathcal{T}_\alpha$  et de  $\mathcal{T}_{\alpha_i}$  respectivement et  $\epsilon_w$  le terme lié à l'erreur d'ordre  $n$  lors des différentes reconstructions. On notera plus particulièrement que par construction :

$$\frac{\epsilon_w}{\Omega_m} = \mathcal{O}(h^n) . \quad (4.8)$$

La deuxième étape consiste à redistribuer cette erreur liée à l'ordre de précision de la méthode que l'on vient d'utiliser. Cette seconde correction garantit la conservativité stricte et la positivité des différentes variables.

Une correction pour chacune des variables peut alors être définie en redistribuant proportionnellement au volume l'erreur commise. On prend ici l'exemple d'une redistribution sur la masse volumique  $\bar{\rho}$  :

$$\bar{\rho}_{\alpha_i}^* = \bar{\rho}_{\alpha_i} \left( 1 + \frac{\epsilon_\rho}{\sum_{i=1}^p |\Omega_{\alpha_i}| \bar{\rho}_{\alpha_i}} \right) , \quad (4.9)$$

qui vérifie alors l'équation :

$$|\Omega_\alpha| \bar{\rho}_\alpha = \sum_{i=1}^p |\Omega_{\alpha_i}| \bar{\rho}_{\alpha_i}^* . \quad (4.10)$$

ce qui traduit la conservativité stricte de la masse.

Un raisonnement similaire a été mené pour les autres variables incriminées (impulsion et énergie). La répartition ainsi corrigée permet de distribuer de l'erreur de conservativité en respectant l'ordre formel de précision d'ordre  $n$ , tout en garantissant la positivité de la redistribution sur les sous-éléments.

### 4.3 Limiteurs associés pour une projection à l'ordre 3

La méthode de redistribution a été implémentée jusqu'à l'ordre 3. Cependant, sans limitation lors de la répartition, des problèmes de stabilité peuvent apparaître. Deux limiteurs permettent de s'assurer de la positivité des variables après application de la méthode de redistribution à l'ordre 3. Ces limiteurs empêchent l'apparition d'oscillations parasites et la création d'extremum local. Le second processus est itératif et limite les dérivées premières et secondes des différentes variables primitives.

#### 4.3.1 Limitation passage bar-tilde sur la Pression

La première étape de limitation est effectuée lors du calcul de la valeur barre de la pression,  $\bar{P}_\alpha$ , à partir de la valeur tildée  $\tilde{P}$ . La condition suivante est imposée :

$$\min_{\gamma \in \mathbb{V}_\alpha \cup \{\alpha\}} (\tilde{P}_\gamma) \leq \bar{P}_\alpha , \quad (4.11)$$

en notant  $\mathbb{V}_\alpha$  l'ensemble des cellules ayant une face en commun avec la cellule  $\mathcal{T}_\alpha$ .

En reprenant l'équation d'état et l'équation des gaz parfaits et en considérant une variation des concentrations des différentes espèces chimiques négligeable, l'équation 2.45 évaluée à l'ordre 3 devient :

$$\tilde{P} - \bar{P} = \underbrace{(\tilde{\gamma} - 1) \frac{\bar{\rho}}{2!} v_\alpha^{(2)}}_{\geq 0} : \left( \frac{\partial \mathbf{u}}{\partial x_i} \otimes \frac{\partial \mathbf{u}}{\partial x_j} \right) + \mathcal{O}(h)^3 . \quad (4.12)$$

avec  $N_1$  un point de référence de  $\mathcal{T}_\alpha$ . Une fois la relation 4.12 réinjectée dans 4.11, on obtient :

$$(\tilde{\gamma} - 1) \frac{\bar{\rho}}{2!} v_\alpha^{(2)} : \left( \frac{\partial \mathbf{u}}{\partial x_i} \otimes \frac{\partial \mathbf{u}}{\partial x_j} \right) \leq \min_{\gamma \in \mathbb{V}_\alpha \cup \{\alpha\}} (\tilde{P}_\gamma) - \bar{P}_\alpha , \quad (4.13)$$



qui est une condition uniquement sur la dérivée première du vecteur vitesse. Cette limitation empêche l'apparition d'un minimum sur la pression moyenne de l'agrégat.

### 4.3.2 Limitation sur le voisinage

La deuxième partie de la limitation est plus habituelle dans le sens où l'on s'assure simplement que la valeur redistribuée  $q_{\alpha_i}$ ,  $i \in \llbracket 1, p \rrbracket$  est bornée dans chacune des  $p$  cellules  $\mathcal{T}_{\alpha_i}$  :

$$\min_{\mathcal{T}_{\beta} \in \mathbb{V}_{\alpha} \cup \{\mathcal{T}_{\alpha}\}} (\tilde{q}_{\beta}) \leq \tilde{q}_{\alpha_i} \leq \max_{\mathcal{T}_{\beta} \in \mathbb{V}_{\alpha} \cup \{\mathcal{T}_{\alpha}\}} (\tilde{q}_{\beta}) , i \in \llbracket 1, p \rrbracket \quad (4.14)$$

avec  $\mathbb{V}_{\alpha}$  l'ensemble des cellules ayant une face en commun avec  $\mathcal{T}_{\alpha}$ .

Si après redistribution des variables, l'inégalité 4.14 n'est pas vérifiée alors les dérivées premières et secondes sont limitées par un même coefficient  $\theta \in [0; 1]$  pour ne pas introduire de maxima ni de minima. Les nouvelles dérivées sont alors réinjectées dans les relations 2.45, 2.43 et 4.6 et les termes  $A_1$ ,  $A_2$ ,  $B_1$  et  $B_2$  et  $C$  sont recalculés. Si l'inégalité n'est toujours pas vérifiée, le processus est répété jusqu'à convergence<sup>7</sup>.

## 4.4 Cas tests de validation

Deux cas analytiques 2D simples, un tourbillon stationnaire et un tourbillon avec vitesse d'advection, permettent de valider l'implémentation de la méthode décrite précédemment. Les solutions sont toutes initialisées à l'ordre 6 par une méthode d'intégration utilisant les points de Gauss. Les équations de Navier Stokes sont discrétisées par une méthode Volume Finis d'ordre 3 qui a déjà été validée sur des maillages non structurés généraux avec et sans intersections géométriques par Pont *et al.* [PBCR17]. Dans le cas présent, pour ces cas tests, seul l'ordre de la redistribution après subdivision des mailles varie. L'adaptation de maillages est effectuée uniquement sur un niveau de raffinement. Le senseur d'adaptation mis en œuvre est basé sur le gradient de pression (*cf.* annexe A.1). L'erreur numérique entre la solution avec adaptation de maillage et la solution théorique est évaluée uniquement dans les mailles raffinées en fin de calcul et permet d'évaluer l'ordre de précision de la méthode implantée<sup>8</sup>.

Toutes les grilles utilisées sont cartésiennes, de pas en espace  $\Delta x = \Delta y$  fixe, ainsi l'ordre  $n$  de la redistribution effectuée sur les mailles créées dépend alors de la relation suivante :

$$q_{G_{ref}} = q_{G_{sim}} + \mathcal{O}(h^n) , \quad (4.15)$$

avec  $G$  le centre de gravité de la cellule considérée,  $q_{G_{ref}}$  la valeur de la fonction de référence calculée au centre de gravité,  $q_{G_{sim}}$  la valeur du champ  $q$  calculé après reconstruction au centre de gravité. On note alors  $\varepsilon = q_{G_{ref}} - q_{G_{sim}}$  l'erreur commise et dans le cas où le maillage considéré est suffisamment fin, on a une relation asymptotique du type :

$$\log \varepsilon = n \log \Delta x + C , \quad (4.16)$$

où  $C$  est une constante et  $\Delta x$  le pas de discrétisation de la grille. Ainsi, en faisant varier le pas de discrétisation de la grille et en évaluant l'erreur dans les mailles raffinées, il est possible d'évaluer l'ordre global de la simulation

7. Les simulations effectuées ont montré une convergence en, au maximum, 2 itérations de limitation dans le cas de raffinement de cellules situées dans des chocs forts.

8. On pourra remarquer que des mailles sont créées en profondeur dans la troisième direction étant donné que pour l'instant, il n'existe pas d'arbre *quad-tree*. Cependant, l'influence des dérivées relatives à la troisième direction reste négligeable pour ces cas tests.

### 4.4.1 Tourbillon stationnaire

Le premier cas test analytique est le vortex stationnaire sans vitesse d'advection. L'adaptation est effectuée juste après l'initialisation du domaine de calcul. Les grilles utilisées sont cartésiennes 2D. On représente sur la figure 4.4(a) la structure générale des maillages obtenus et sur la figure 4.4(b) le champ de Pression obtenu après adaptation.

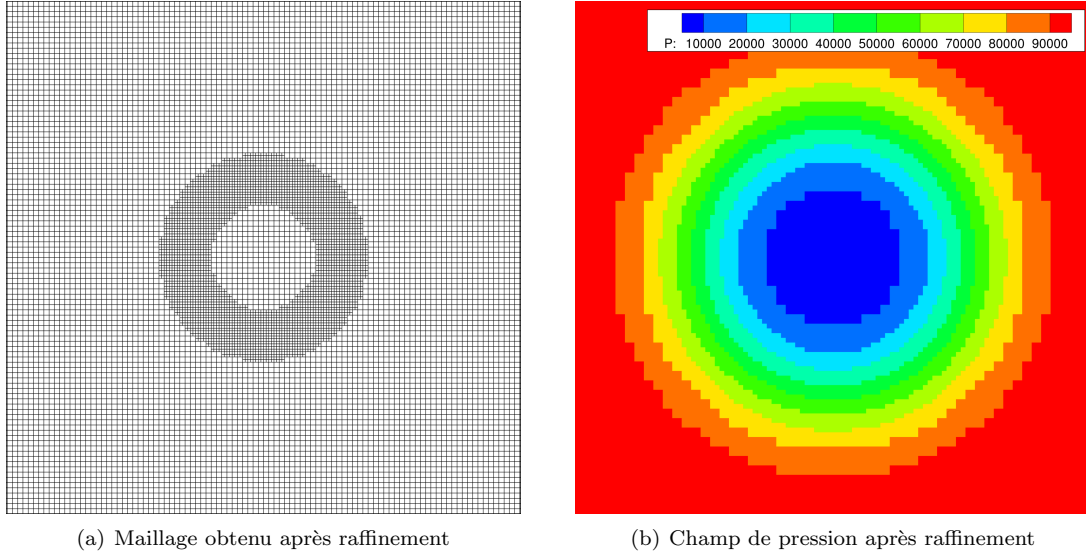


FIGURE 4.4 – Résultats tourbillon stationnaire

La figure 4.5 présente une comparaison des champs de dérivée première de la vitesse obtenus par redistribution sur la grille 1 et la grille 3, référencées dans la table 4.1, en utilisant une redistribution d'ordre 1 et d'ordre 3. On remarque clairement un déficit de dérivée au niveau des interfaces inter-niveaux de raffinement pour la méthode d'ordre 1 dans les deux maillages. Les champs affichés sont ceux des valeurs au centre de gravité, sans interpolation du logiciel de visualisation.

On présente sur la figure 4.6 et la table 4.1 les résultats obtenus en utilisant plusieurs grilles avec des pas de discrétisation différents. Les ordres obtenus sont ceux attendus et les résultats sont concluants pour ce premier cas test. Ainsi dans les zones raffinées, le facteur limitant pour l'ordre de la solution après une seule itération est l'ordre de la redistribution faite par l'AMR. On peut alors comprendre l'intérêt d'une méthode de redistribution d'ordre élevé. Pour un écoulement stationnaire, si la redistribution des variables n'est pas d'ordre élevée, la convergence est ralentie. Dans le cas d'un écoulement instationnaire, ces erreurs sont propagées dans le calcul et si l'adaptation est fréquente, l'ordre global de la solution chute à un ordre inférieur à celui attendu.

Grilles	$\Delta x$
1	$1.8 \times 10^{-1}$
2	$9 \times 10^{-2}$
3	$4.5 \times 10^{-2}$

TABLE 4.1 – Détail des calculs effectués

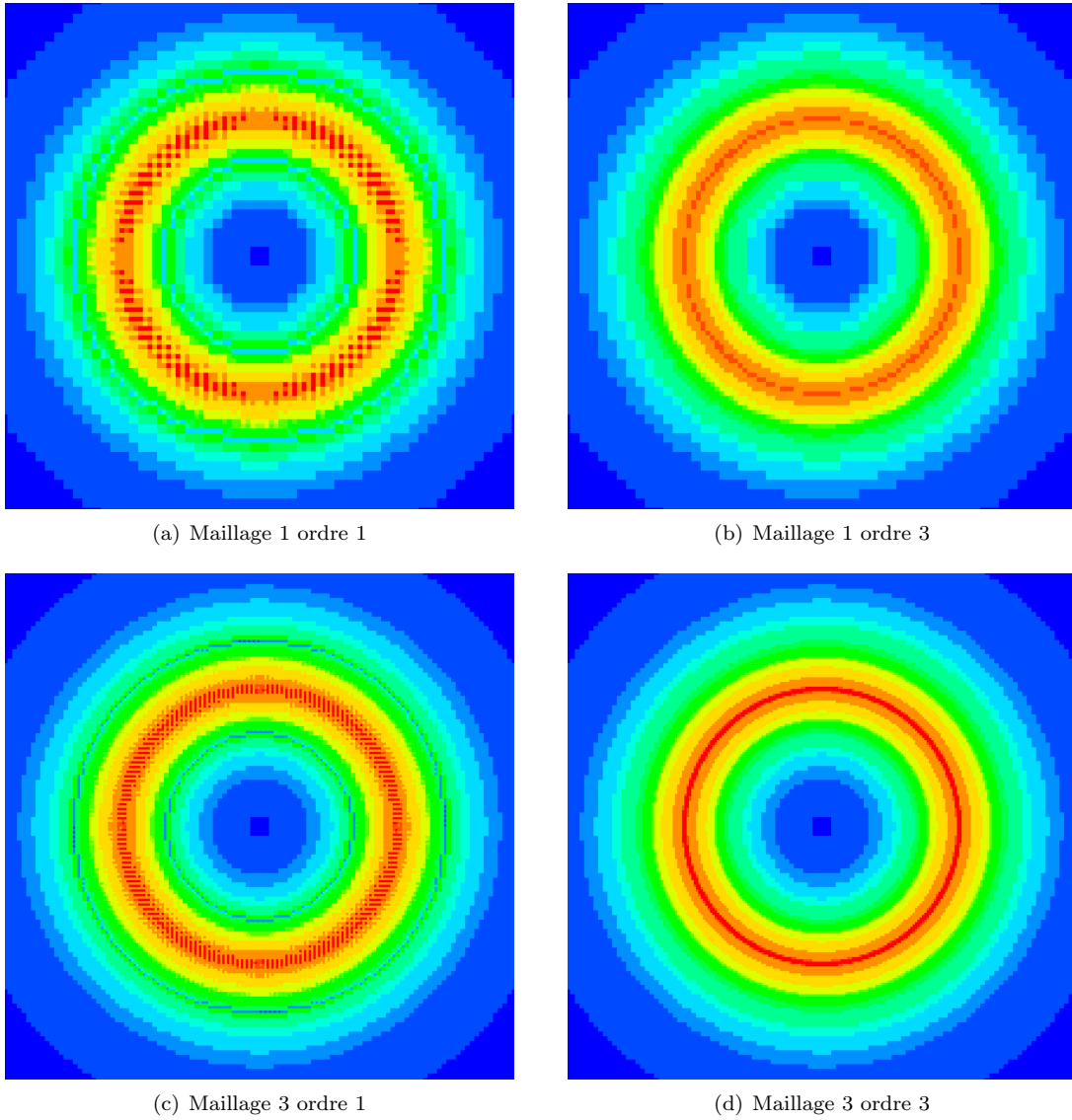


FIGURE 4.5 – Dérivée première de la vitesse, norme de froebenius

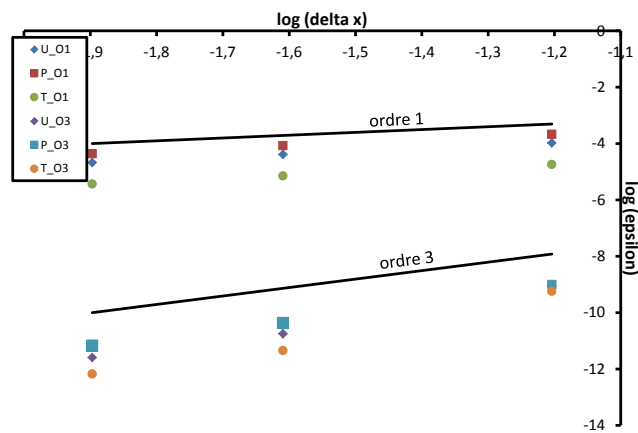


FIGURE 4.6 – Tracé de l'erreur en fonction du pas de discrétisation

### 4.4.2 Tourbillon en mouvement

Le second cas test est un tourbillon en mouvement. L'étude a été menée sur des maillages avec et sans chevauchement. Le tableau 4.2 détaille l'ensemble des maillages sans intersection utilisés pour l'étude. La figure 4.7 montre un maillage avec intersection utilisant la grille de fond 2 dans le tableau 4.2.

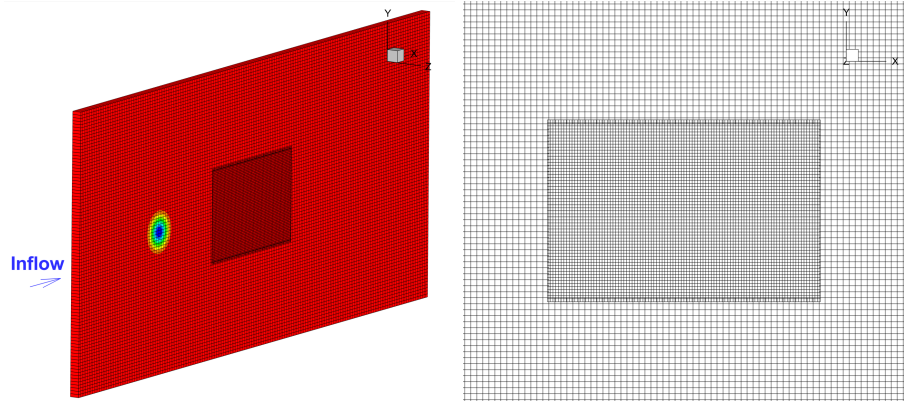


FIGURE 4.7 – Champ de température initial et zoom sur la zone de chevauchement de maillage

Maillage de fond	$\Delta x = \Delta y$	Points par tourbillon (maillage avant raffinement)
1	0.2142	11
2	0.1500	15
3	0.1153	20

TABLE 4.2 – Nombre de cellules par grille

L'erreur  $L_2$  est évaluée sur les cellules raffinées en fin de calcul. La figure 4.8 compile les résultats obtenus, on obtient ainsi une pente du 3<sup>ème</sup> ordre pour les variables considérées à la fin du calcul pour le cas test sans intersection.

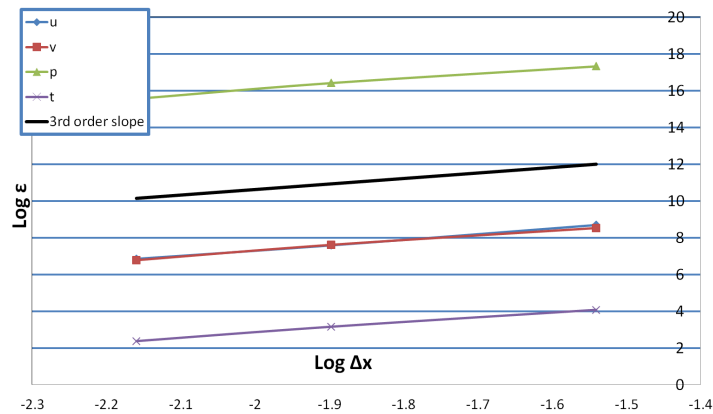
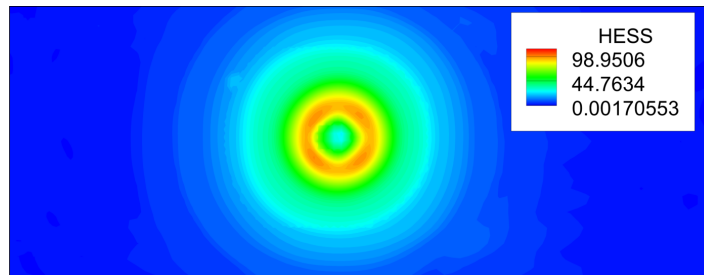


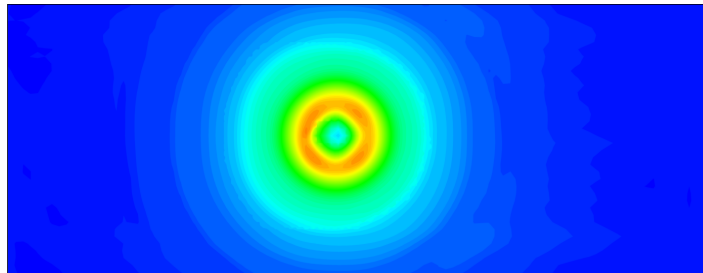
FIGURE 4.8 – Erreur en fonction du pas de discrétisation pour le tourbillon en mouvement sans intersection

Les calculs réalisés avec intersection utilisent les mêmes grilles de fond que celles montrées précédemment. Un grille plus fine recouvre ces grilles de fond et ne sera pas raffinée au cours du calcul. Les résultats obtenus avec intersection montrent aussi une pente du 3<sup>ème</sup> ordre, cependant l'erreur est divisée par deux. En effet la grille raffinée permet d'augmenter le nombre de points par tourbillon pendant que celui-ci traverse la grille de priorité supérieure. Par comparaison, le calcul a été fait avec une redistribution d'ordre 1 et l'on présente un zoom du vortex en fin de calcul sur les figures 4.9(a), 4.9(b) et 4.9(c) dans les cas avec et sans intersection avec projection d'ordre 3 et sans intersection avec projection d'ordre 1. Le tourbillon final obtenu est bien plus

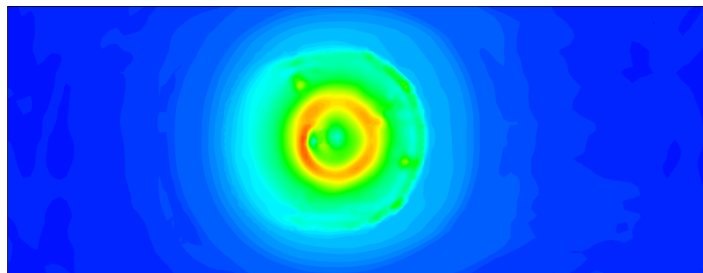
dégradé dans le cas d'une redistribution d'ordre 1 ce qui confirme l'intérêt d'une cohérence globale entre schémas de discrétisation et de redistribution quant à la qualité de la simulation.



(a) Intersection et redistribution d'ordre 3



(b) Sans intersection et redistribution d'ordre 3



(c) Sans d'intersection et redistribution d'ordre 1

FIGURE 4.9 – Champ de la norme de Frobenius de la matrice hessienne de la vitesse, comparaison des résultats obtenus en présence ou non d'intersection, en fonction de l'ordre de précision de la redistribution des variables

## Chapitre 5

# Validation de la méthodologie AMR implantée dans FLUSEPA

### Introduction

Ce chapitre présente l'application de la méthode AMR décrite et construite précédemment, à trois cas d'études : une marche montante, une marche descendante et une onde de souffle simplifiée. Ces trois cas permettent chacun d'isoler certaines des fonctionnalités recherchées du module AMR et mettent en exergue les besoins futurs pour une utilisation industrielle intensive.

La mise en place des deux premiers calculs a été détaillée dans le rapport de Dmytrus-Saurin [DS16] ; ce manuscrit rappelle et complète cette étude. Le cas de la marche montante permet de valider simplement l'utilisation du module AMR et du comportement de la redistribution conservative des variables en présence d'ondes de choc avec un régime transitoire instationnaire. Le second cas vise à valider le couplage d'une stratégie d'adaptation de maillage avec une modélisation hybride RANS/LES de la turbulence en présence d'un chevauchement de maillage. Enfin, le cas d'une onde de souffle est un cas particulier pour lequel le nombre de mailles théorique doit croître continuellement puisque les ondes de pression simulées sont sphériques. Ainsi, les zones raffinées pour l'onde de souffle sont topologiquement assimilables à des coquilles de rayon  $d$  et le nombre de mailles nécessaires pour correctement les résoudre est proportionnel à  $d^2$ . En l'absence de redimensionnement des tableaux, c'est donc un des pires cas en terme de performance pour le module AMR *octree*. Il pourra servir de cas de référence pour les améliorations futures de la gestion mémoire. En outre, c'est un cas pour lequel l'optimisation du coût global d'une simulation par rapport à un calcul sans adaptation de maillage peut être maximale.

Ces différents cas test montrent ainsi que les enjeux industriels de l'adaptation de maillage sont considérables. Ce n'est pas uniquement par le *brute-forcing*, *i.e.* en se reposant sur la croissance des moyens de calculs, que l'on parviendra à un gain significatif sur la précision et la qualité des simulations, mais bien en ayant des codes plus intelligents et mieux optimisés. En complément à l'AMR, les méthodes utilisant un ordonnancement en graphe de tâches [CC16] semblent, à ce titre, être très prometteuses dans le sens où elles apportent un niveau d'abstraction au code de calcul qui permet de simplifier son développement et de maximiser ses performances en distribuant les charges de calcul de façon dynamique sur les processeurs.

### 5.1 Marche montante

Le premier cas étudié est un cas Euler (non visqueux) de marche supersonique face à un écoulement à Mach 3. Ce cas a été mis en place pour valider des stratégies d'adaptation de maillage en situation de capture de chocs. Plusieurs études ont déjà été faites par Emery [Eme68], Woodward et Colella [WC84], Jouhaud [Jou97]

puis plus récemment par Harris et Williams [HW13] (méthode AMR) et Harris [Har13] (méthode d'extraction de caractéristiques). Ce cas test est bidimensionnel, ainsi les vitesses selon  $Oz$  ne seront pas prises en compte, même si le module *octree* raffine dans cette direction<sup>1</sup>. Le pas de discrétisation initial est  $\Delta x = \Delta y = \frac{L}{6}$ . Les conditions initiales amont sont regroupées dans la table 5.1.

$\rho$	$\rho u$	$\rho E$	$\rho \nu$
1.4	4.2	8.8	0

TABLE 5.1 – Données de calcul marche ascendante

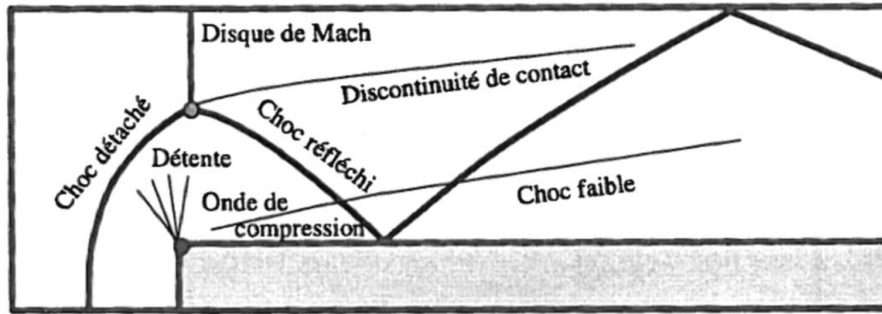


FIGURE 5.1 – Schéma des structures caractéristiques de l'écoulement [Jou97]

La physique attendue lors de ce cas test est présentée sur la figure 5.1. Le coin de la marche est une singularité et peut donner naissance à des oscillations en aval de la marche et une erreur sur l'entropie à la paroi du bas<sup>2</sup>. Dans l'article de Woodward et Colella [WJK<sup>+</sup>13] plusieurs traitements sont appliqués pour limiter et contenir ces erreurs numériques. Nous n'appliquons aucun traitement particulier dans ce calcul, tout comme l'avaient fait Harris et Williams [HW13]. Le raffinement du maillage  $\mathcal{M}_1$  est autorisé sur deux niveaux ce qui nous permet de comparer les résultats avec adaptation de maillage, et les résultats sur un maillage  $\mathcal{M}_2$  raffiné initialement deux fois dans les directions  $Ox$  et  $Oy$  sans AMR. Le critère choisi pour piloter le raffinement utilise le gradient de pression.

### 5.1.1 Résultats et analyse

On effectue une comparaison entre les maillages  $\mathcal{M}_1$  et  $\mathcal{M}_2$  à différents instants au cours du calcul du champ de pression entre le maillage AMR et le maillage raffiné intégralement sur la figure 5.2. Les résultats obtenus dans les deux simulations sont très similaires. Sur le dernier instant, image 5.2(e) et 5.2(f), on peut observer que la dernière réflexion du choc en haut à droite n'est pas captée par l'AMR, ce qui est probablement dû à un seuil de raffinement fixé à une valeur trop élevée. En conséquence, il existe un léger sur-raffinement au niveau du coin de la marche lié à la détente, cependant aucune oscillation n'a été observée.

Le maillage AMR initial contient 8400 mailles, le maillage après adaptation présenté figure 5.3(c) contient 40 000 mailles, le temps de calcul est de 20min sur 24 processeurs. Le calcul a été mené sur un seul processus aérodynamique étant donné le faible nombre de mailles et l'absence de chevauchement de maillage. Par comparaison, le maillage  $\mathcal{M}_2$  contient 500 000 mailles et le temps de calcul est de 2h. Le coût des algorithmes de remaillage est faible, inférieur à  $< 5\%$ . Le surcoût principal lié à l'AMR est le calcul, à chaque itération, des propriétés géométriques de l'ensemble des éléments du maillage, ainsi que des tableaux d'indirection (face, cellules) qui sont reconstitués afin de pouvoir effectuer la simulation. On donne sur la figure 5.3 des comparaisons

1. Un noyau AMR *quadtree* sera certainement développé par la suite pour des configurations 2D qui servent lors des études industrielles préliminaires.

2. Problème de gradient de pression infini :  $\lim_{R \rightarrow 0} \left( \frac{\rho u^2}{R} \right) = 0$ , avec  $R$  le rayon de courbure.

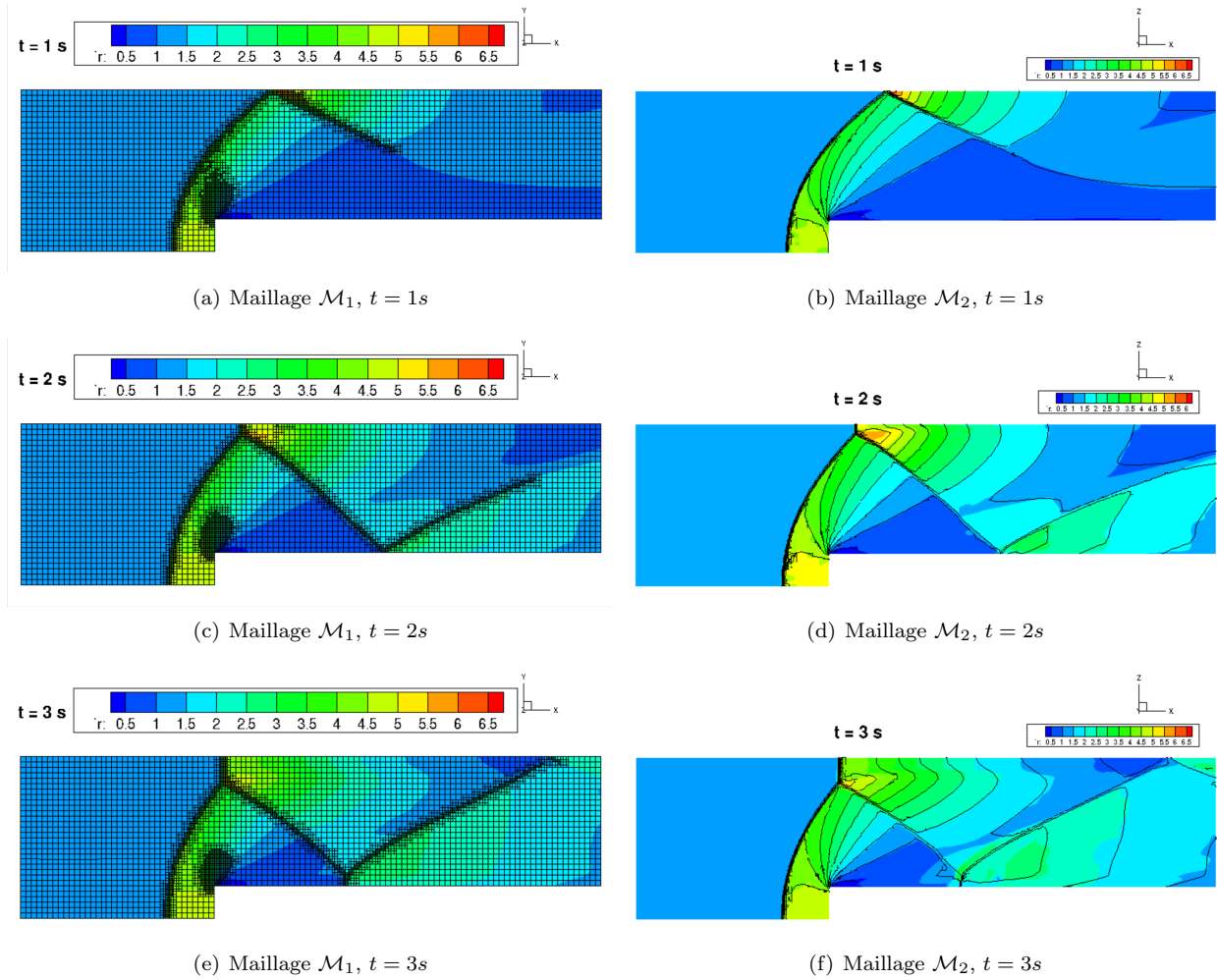


FIGURE 5.2 – Comparaison calcul AMR, calcul raffiné sur deux niveaux

des calculs FLUSEPA avec ceux de Harris et Williams [HW13] et Jouhaud [Jou97]. Par rapport aux maillage obtenus par Harris et Williams, les zones de maillage raffinées du calcul FLUSEPA sont moins étendues, ceci s’observe principalement dans la zone de la discontinuité de contact.

En abaissant les seuils d’adaptation, il est possible de capter la discontinuité de contact et le choc faible au niveau de la marche. Cependant, le critère de gradient de pression n’est pas suffisamment sélectif et des calculs non présentés ici ont montré que ce critère n’était pas adapté. Harris [HW13] note que pour limiter l’acoustique, il est possible d’utiliser la valeur de  $U \cdot \text{grad}P$  comme seuil d’adaptation. Une étude effectuée par [DS16] a permis de mieux capter les variations plus faibles de pression et de limiter l’acoustique. L’analyse physique n’a pas été poussée plus en avant, étant donné le contexte très spécifique de cas test. Nous avons principalement effectué la validation fonctionnelle des différents modules et vérifié que la redistribution conservative d’ordre élevé était opérante dans le cas du raffinement sur une discontinuité. La comparaison des résultats avec une simulation standard sans AMR, a aussi permis de réaliser un cas de non régression de la fonctionnalité AMR.

## 5.2 Marche descendante

Le deuxième cas test est une marche descendante qui fait partie de la base de validation de FLUSEPA. Cet écoulement est très similaire à un écoulement d’arrière corps qui est caractérisé par un décollement massif de la couche limite, voir figure 5.4. Différentes études expérimentales ont été menées par Moreau *et al.* [MLDB87]



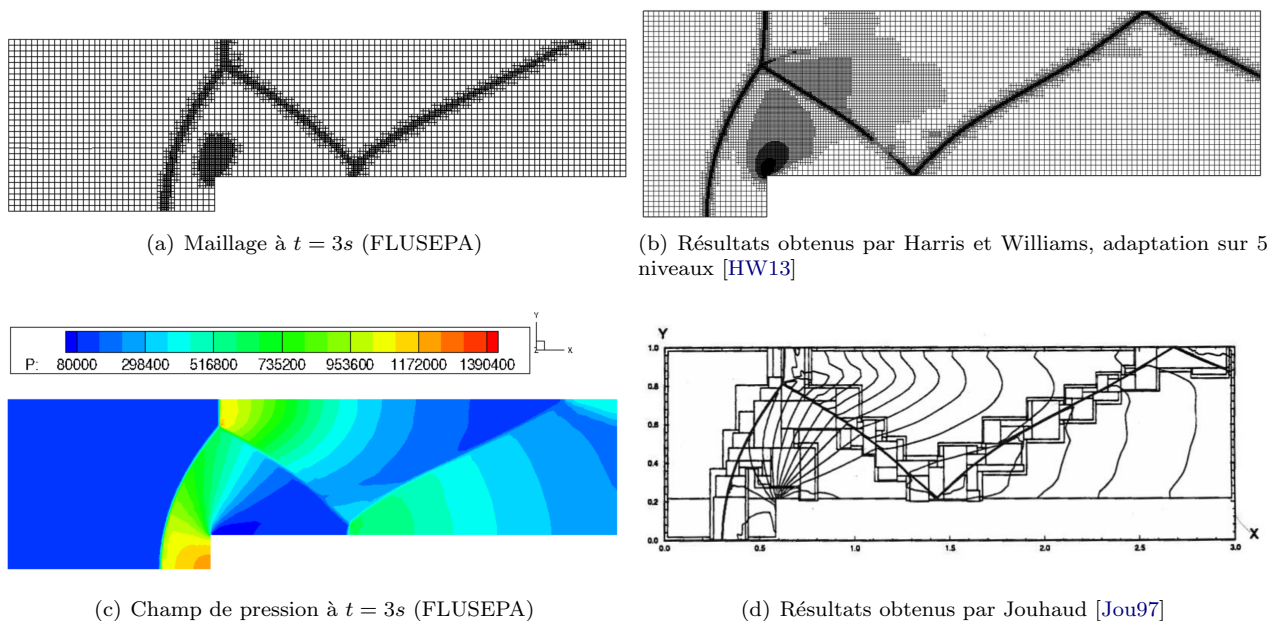


FIGURE 5.3 – Résultats obtenus après adaptation de maillage

qui a considéré ce cas comme une configuration de chambre de combustion puis par Driver *et al.* [DSM87] et enfin par Hall *et al.* [HBFM03]. La simulation numérique de référence pour notre étude est celle de Pont [Pon15] qui a développé et validé les modèles de turbulence HRL de FLUSEPA et s'était lui-même comparé aux résultats de Sainte-Rose *et al.* [SRBDD09] et Deck [Dec12]. On note ici que l'AMR est compatible avec l'approche numérique qui consiste en un recentrage local du schéma numérique. En effet, le schéma spatial d'ordre 3 est recentré lorsque le Reynolds de maille le permet, cette méthode a été décrite par Pont *et al.* [PBCR17]. Or, le fait de raffiner localement le maillage diminue le Reynolds de maille, et plus le maillage est raffiné, plus la résolution du maillage augmente, mais aussi plus le schéma numérique est recentré et donc plus l'erreur de résolution (dissipation et dispersion) diminue.

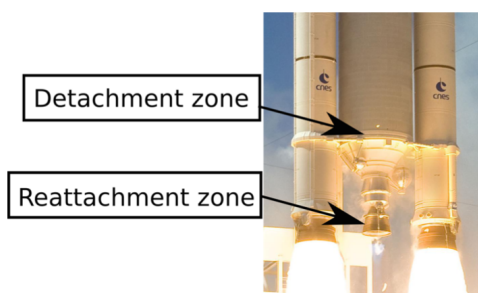


FIGURE 5.4 – Visualisation des zones de décollement et de rattachement de la couche limite sur Ariane 5

### 5.2.1 Construction du maillage initial

La géométrie utilisée est donnée figure 5.5. La hauteur de la marche est notée  $h$  et la physique moyenne de l'écoulement est présentée sur la figure 5.6. Une couche limite amont se forme dans la zone d'abscisse négative, cette couche limite décolle en  $x = 0$ . Ce décollement induit la formation d'une bulle de recirculation primaire entre la marche et l'abscisse  $X_r$  et d'une bulle secondaire, plus petite, dans le coin de la marche. Par la suite, nous étudierons principalement la bulle de recirculation primaire. Au niveau du décollement, le maillage doit être suffisamment fin pour capter la formation d'instabilités de Kelvin Helmholtz liée à la présence d'une couche

de cisaillement. La zone représentée en rouge, qui s'étend de la marche à  $1,5X_r$ , est fortement instationnaire et l'on souhaite appliquer une résolution de type LES pour correctement capter les structures tourbillonnaires, leur contenu fréquentiel ainsi que le point de recollement de l'écoulement moyen sur la paroi. Les autres zones de l'écoulement, représentées en vert, à savoir la couche limite amont, la marche et toutes les zones d'ordonnées supérieures à  $0,5h$ , avec  $h$  la taille de la marche, peuvent être simplement modélisées en RANS. Ainsi, il est possible d'utiliser une modélisation HRL de la turbulence et les grilles initiales seront construites et superposées de façon à apporter la résolution nécessaire.

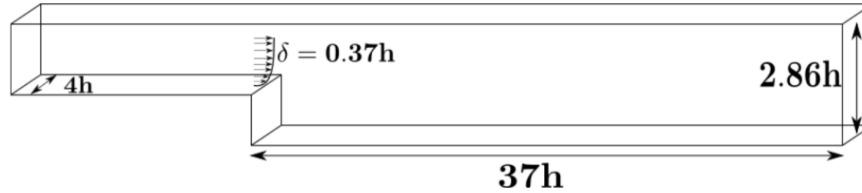


FIGURE 5.5 – Schéma de la géométrie pour la simulation de la marche descendante

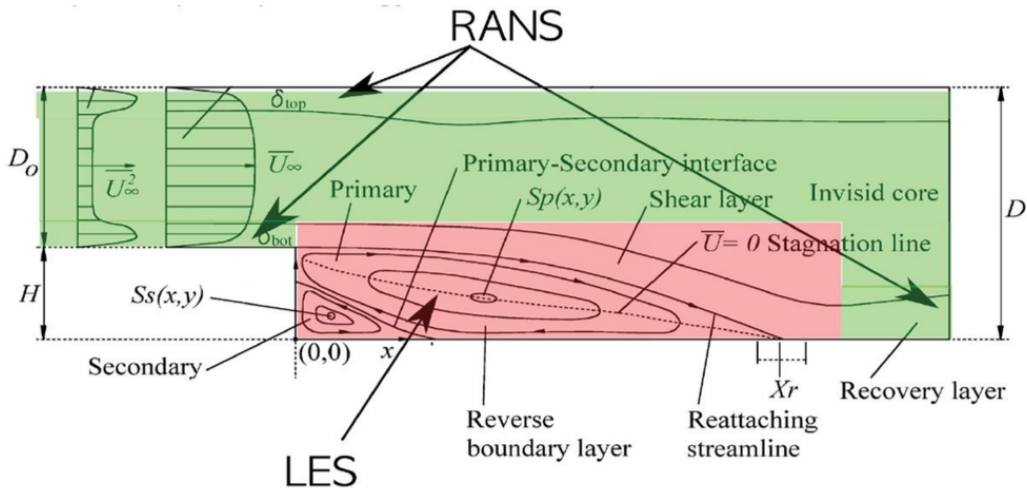


FIGURE 5.6 – Physique de l'écoulement de la marche descendante. En noir : Descriptif de la physique de l'écoulement moyen, en rouge : zone de modélisation LES, en vert : zones RANS

La configuration de l'écoulement variant très brutalement lors de l'initialisation du calcul, les échelles de turbulence varient elles aussi rapidement, par conséquent il est intéressant d'utiliser le chevauchement de maillage pour restituer un saut dans la résolution du maillage. Ainsi, le maillage initial pour la simulation AMR comprend  $4 \times 10^5$  mailles, il est construit par chevauchement de plusieurs grilles parallélépipédiques dont on représente schématiquement les bords sur la figure 5.7 :

- deux grilles  $\mathcal{G}_1$  et  $\mathcal{G}_2$  permettent de simuler les paroi en haut et en bas,
- deux grilles très grossières  $\mathcal{G}_3$  et  $\mathcal{G}_4$  avant la marche hors de la couche limite, et après la marche après la zone d'intérêt sont des grilles de fond,
- une grille  $\mathcal{G}_5$  sous-résolue pour l'écoulement considéré, est l'unique grille dans laquelle la stratégie AMR sera appliquée.

La priorité des grilles est telle que :

$$\text{PRIORITE}(\mathcal{G}_1) = \text{PRIORITE}(\mathcal{G}_2) > \text{PRIORITE}(\mathcal{G}_5) > \text{PRIORITE}(\mathcal{G}_3) = \text{PRIORITE}(\mathcal{G}_4)$$

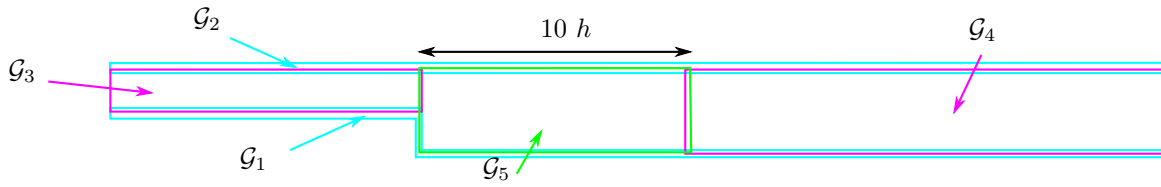


FIGURE 5.7 – Maillage utilisé pour le calcul. En cyan : les grilles de paroi, en magenta : les grilles des fond, en vert : la grille AMR

En pratique, la résolution en  $z$  des grilles  $\mathcal{G}_1$  à  $\mathcal{G}_4$  est initialement inférieure à la taille minimale imposée comme taille limite pour le module AMR, ce qui bloque le raffinement *octree* dans ces grilles. Ce choix de résolution en  $z$  conduit à un sur-raffinement d'un facteur 3 du maillage, l'écoulement étant essentiellement 2D dans ces grilles. Cependant, le senseur pilotant le raffinement est la taille de Von Kármán décrite dans l'annexe A. On considère que cette taille est adaptée uniquement dans des zones LES, mais pas dans les zones RANS, dans lesquelles le senseur d'adaptation et la stratégie d'adaptation doivent être modifiés. L'utilisation d'un critère d'adaptation de maillage licite à la fois, dans des zones RANS et des zones LES. Ce critère doit permettre de traiter les zones fortement anisotropes dans la partie RANS et aussi être couplé à une stratégie d'adaptation anisotrope, par exemple en autorisant des raffinements de type *di-* ou *quad-tree*. La résolution spatiale de la grille  $\mathcal{G}_5$  située dans la zone de recirculation a été choisie de telle sorte, qu'après trois raffinements par *octree*, la résolution obtenue soit au moins celle utilisée par Pont [Pon15], les détails de la résolution des grilles sont donnés dans le paragraphe suivant.

### 5.2.2 Cadre numérique et physique

Le schéma numérique est un schéma Volume Fini d'ordre 3 en espace, avec recentrage local lorsque le Reynolds de maille le permet. Ainsi, ce type de schéma tend vers l'ordre 4 lorsque la résolution est suffisante et tire profit de l'utilisation d'une méthode AMR. Le schéma d'intégration temporelle est le schéma temporel adaptatif décrit paragraphe 2.2.2. L'intégration est appliquée sur 4 niveaux temporels et l'adaptation de maillage est effectuée à chaque macro-itération. Le modèle de turbulence utilisé est le modèle DDES de Spalart décrit dans la section 2.1.5. La longueur avant la marche est choisie de telle sorte que la couche limite fasse  $0,37h$  au niveau de la marche. Le temps final adimensionné de la simulation est de  $t_{sim} = \frac{TU_\infty}{h} = 1200$  et les moyennes statistiques des différentes variables, permettant la validation des résultats, sont amorcées à partir de  $t_{start} = 120$ . Le Reynolds de l'écoulement basé sur la taille de la marche est de 40000. La validation des résultats est faite en comparant les champs de valeurs efficaces de la vitesse (champ *root mean square* ou RMS) à ceux obtenus par Pont, sur un maillage de  $3,6 \times 10^6$  mailles adapté à la turbulence (voir tableau 5.2).

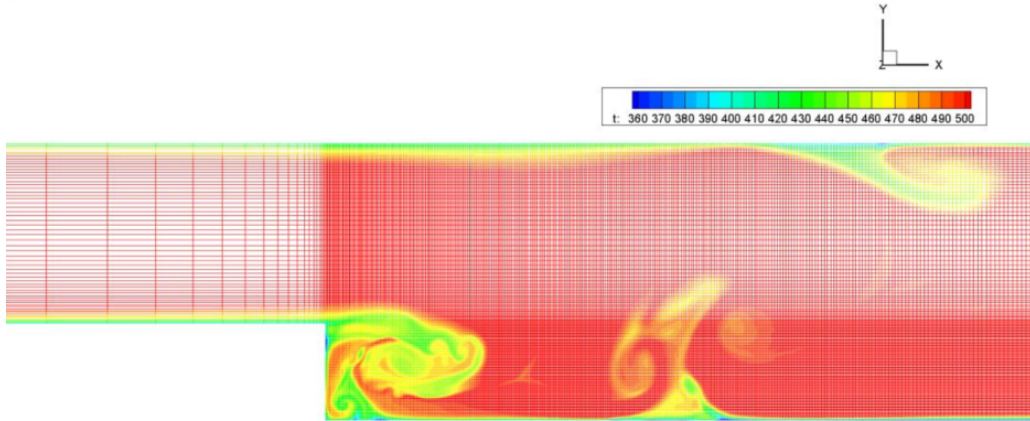
	$x^+$	$y^+$	$z^+$
PONT	15	15	75
LIMARE	130	130	550

TABLE 5.2 – Comparaison de la résolution initiale du maillage avec le calcul de Pont [Pon15]

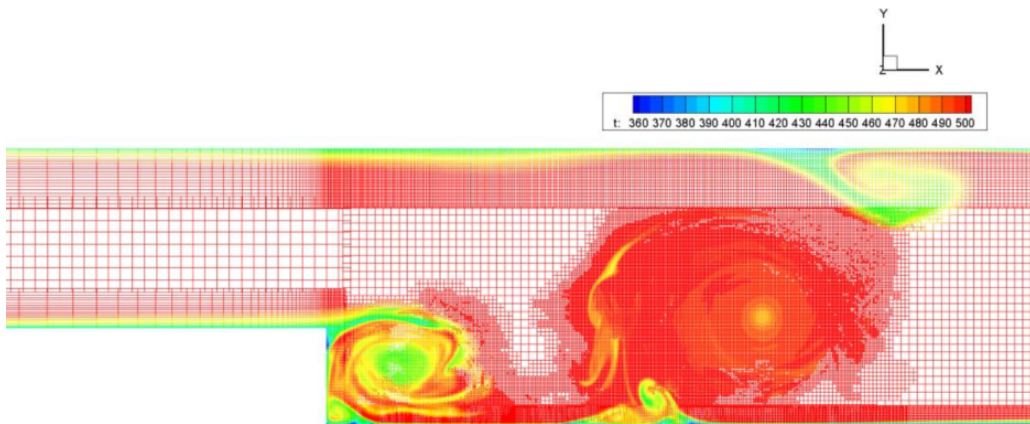
### 5.2.3 Résultats et analyse

Le maillage durant la convergence des statistiques comprend entre  $2.3$  et  $2.8 \times 10^6$  mailles. Le pas de temps du calcul est  $\Delta t = 8.5 \times 10^{-6}$ , le nombre de macro-itérations est  $1 \times 10^5$  et le calcul a duré 15 jours sur 200 cœurs. Le maillage de Pont était optimisé pour la simulation à effectuer, cependant la résolution en  $x$  diminuait à partir de  $\frac{x}{h} = 7$ , or dans notre cas les structures turbulentes sont captées jusqu'à  $\frac{x}{h} = 9.5$  et le maillage de Pont était aussi déraffiné à partir de  $y = 0$ . En effet, le maillage de Pont a une taille de maille selon l'ordonnée qui suit une progression géométrique de 1,2 (généralement utilisé pour la résolution des couches limites). Notre maillage

possède uniquement cette contrainte dans la couche limite amont mais pas dans la zone de recirculation, la grille  $\mathcal{G}_5$  étant parfaitement isotrope en  $x$  et en  $y$ . L'influence de cette résolution peut s'observer qualitativement sur la figure 5.8 qui montre les deux champs instantanés de température dans la partie transitoire au même instant, le calcul AMR résout des structures tourbillonnaires de plus petite taille et le raffinement n'est effectué que dans la zone tourbillonnaire.



(a) Champ de température, calcul de référence



(b) Champ de température, calcul AMR

FIGURE 5.8 – Comparaison entre le maillage fixe et le maillage AMR, coloration par la température

On représente sur les figures 5.9(a) et 5.9(b) une coupe du maillage coloré par la température au cours du calcul. D'un point de vue, le module AMR se comporte bien, les structures tourbillonnaires sont correctement détectées par le critère de Von Kármán. Le schéma d'ordre 3 avec recentrage est très peu dissipatif dans la grille  $\mathcal{G}_5$  et les structures tourbillonnaires ne sont dissipées que par effet de maillage à partir de  $\frac{x}{h} = 10$ . L'intersection est bien supportée et ne gêne pas le départ des instabilités de Kelvin Helmholtz. En effet, la résolution du maillage de paroi imposée initialement est suffisante pour le départ de ces instabilités. On représente un zoom de la zone d'intersection sur la figure 5.9(b) juste après la marche qui montre la continuité de la résolution après adaptation sur 3 niveaux de raffinement en  $x$  et en  $y$ , à travers l'intersection. Ce cas illustre la compatibilité de l'adaptation de maillage avec le chevauchement de maillages.

La figure 5.10(a) représente le champ moyen obtenu dans le plan médian en  $z$ , ainsi que la position des capteurs utilisés pour l'étude ci-après. On compare la taille de la bulle de recirculation obtenue par adaptation de maillage aux résultats de Pont [Pon15] pour le modèle DDES aux différents relevés expérimentaux reportés dans la table 5.3. La taille obtenue est identique dans tous les cas. Il n'y a donc pas d'effet notable du raffinement sur la taille la zone de recirculation.

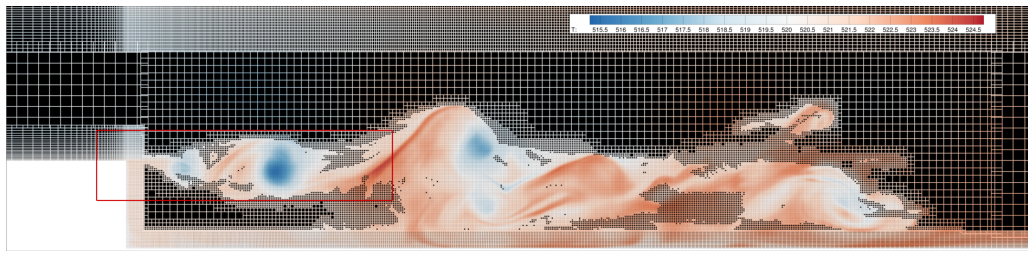
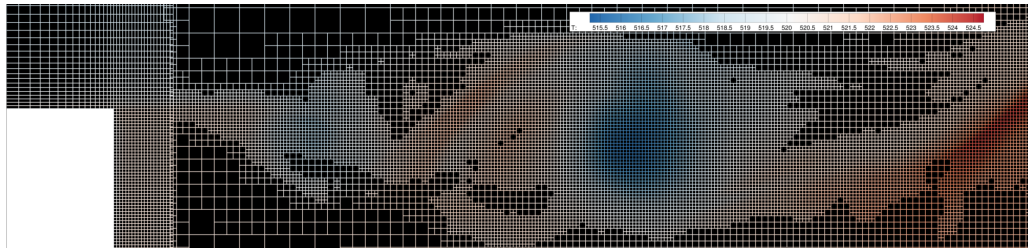
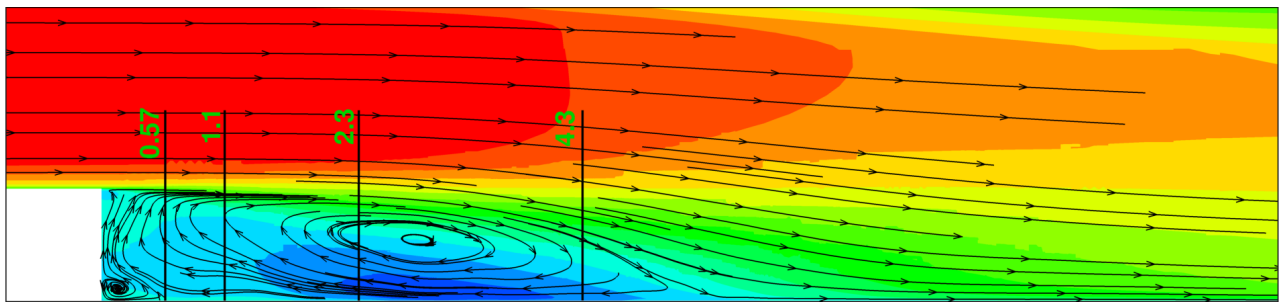
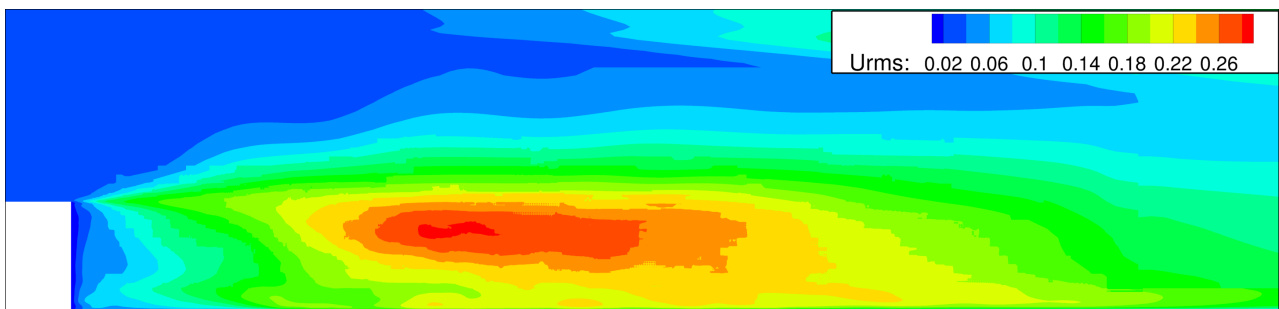
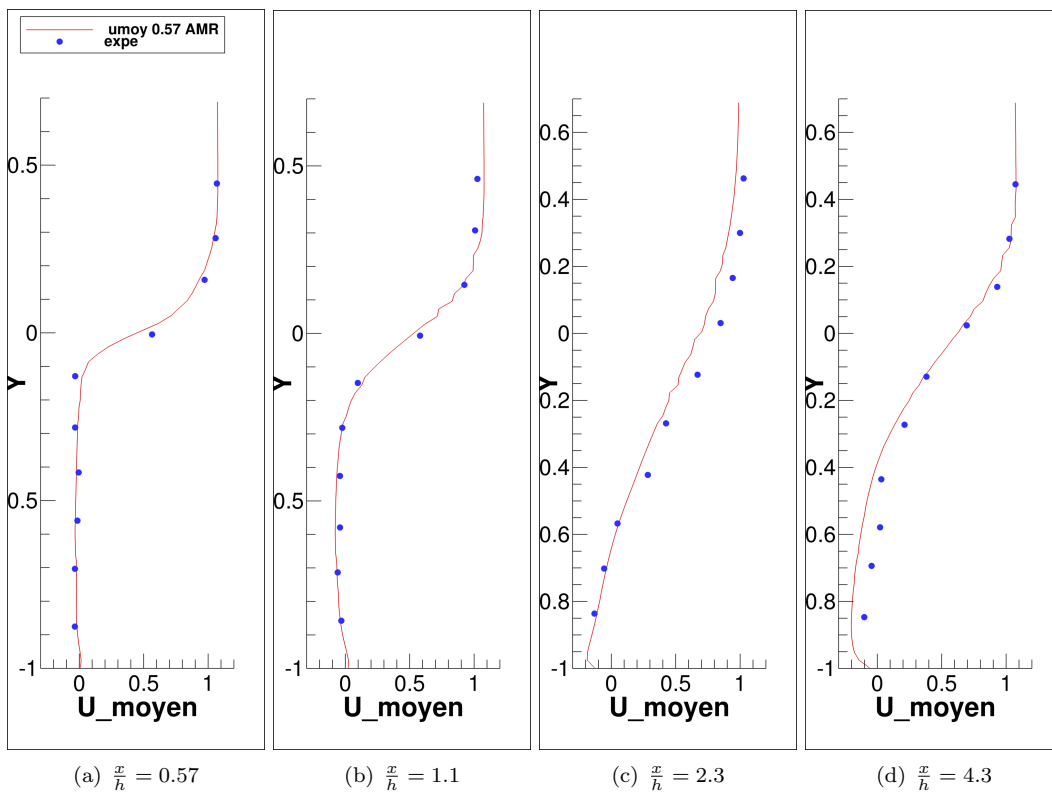
(a) Maillage  $\mathcal{G}_5$  après adaptation(b) Zone d'intersection entre les grilles  $\mathcal{G}_1$  et  $\mathcal{G}_5$  au niveau de la marcheFIGURE 5.9 – Maillage coloré par la température au cours du calcul dans le plan  $z = 0$ (a)  $\bar{U}$ . Positionnement des capteurs de l'expérience(b)  $U_{rms}$ 

FIGURE 5.10 – Champs statistiques

Les capteurs représentés sur la figure 5.10(a) sont comparés aux résultats expérimentaux de Moreau *et al.* [MLDB87], à la fois pour le profil de vitesse moyen selon  $y$  figure 5.11 et pour le champ de vitesse efficace  $u_{rms}$  figure 5.12. L'ensemble des courbes pour le profil moyen montre une bonne résolution du champ moyen de la vitesse dans la bulle de recirculation. Les champ  $u_{rms}$  sont aussi globalement en accord avec les résultats expérimentaux. La figure 5.13 rappelle les résultats numériques obtenus par Pont [Pon15], et ne montrent pas de différences notables entre la DDES de ces calculs et le calcul AMR présenté dans cette section. La stratégie d'adaptation n'a donc pas apparemment d'influence notable sur les champs de  $u_{rms}$ .

	Pont	Driver <i>et al.</i> [DSM87]	AMR
$\frac{x_r}{h}$	6.1	6.1	6.1

TABLE 5.3 – Comparaison des tailles de la bulle de recirculation

FIGURE 5.11 – Résultat AMR  $\frac{\bar{U}}{\bar{U}_\infty}$  en fonction de  $y$  à  $x = 0$ , comparaison avec les mesures expérimentales de [MLDB87]

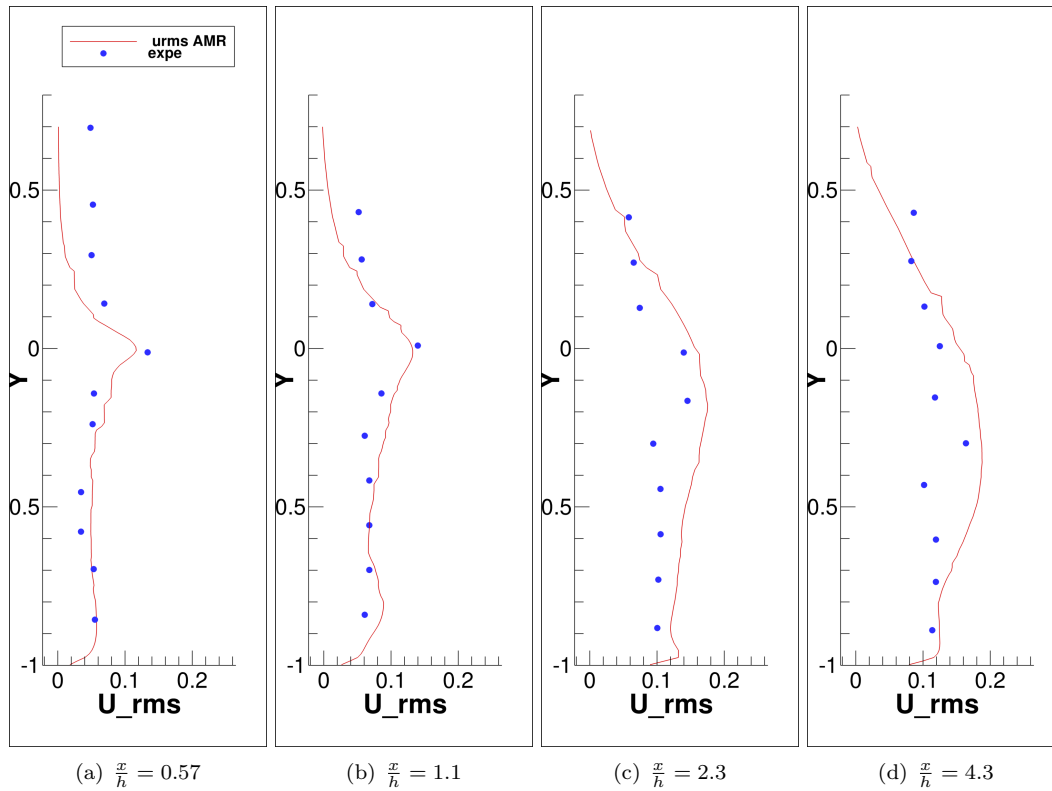


FIGURE 5.12 – Résultat AMR  $\frac{U_{rms}}{U_{\infty}}$  en fonction de  $y$  à  $x = 0$ , comparaison avec les mesures expérimentales de [MLDB87]

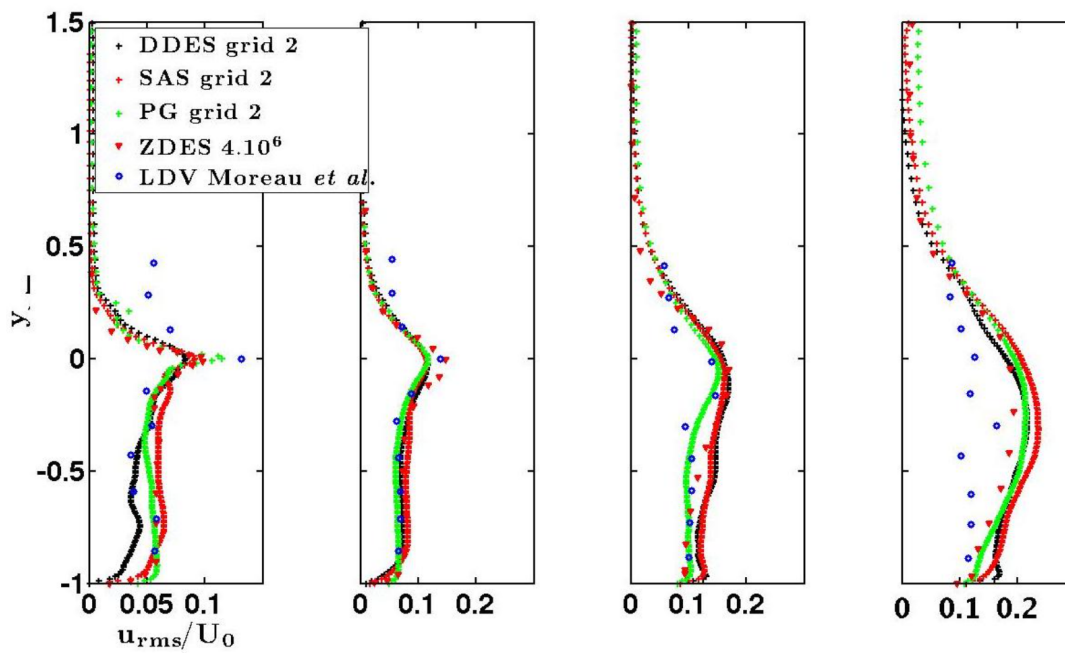


FIGURE 5.13 – Résultats  $\frac{U_{rms}}{U_{\infty}}$  en fonction de  $y$  à  $x = 0$  de Pont [Pon15] pour les modèles HRL de FLUSEPA, comparés aux résultats expérimentaux de Moreau [MLDB87] et à l'étude ZDES de Deck [Dec02]

### 5.3 Simulation d'une onde de souffle

Afin de dimensionner les boosters de lanceurs spatiaux, il est nécessaire de calculer les efforts liés au niveaux de pression qui s'exercent sur les parois de ceux-ci. Ainsi, des essais expérimentaux sont effectués et comparés aux résultats numériques. L'étude expérimentale de référence pour ArianeGroup lors de ses simulations d'onde de souffle liées à l'allumage de booster est le banc martel<sup>3</sup>. Le banc martel est une installation expérimentale toute particulière puisqu'elle permet l'étude d'une onde de souffle sans utilisation de propergol solide. Le système générateur d'onde de souffle, conçu par l'ONERA avec le soutien du Laboratoire de Combustion et de Détonique de l'Université de Poitiers, est formé d'une première chambre sphérique qui est remplie d'un mélange air et méthane à moins de 10 bars tel que présenté sur la figure 5.14. Une fois ce mélange allumé, la déflagration fait monter la pression jusqu'à 100 bars et la température jusqu'à 2500K. Un opercule est alors perforé ce qui libère ces gaz sous haute pression et haute température vers une seconde chambre équipée d'une tuyère d'éjection. La rapidité de la montée en pression peut être contrôlée en faisant varier la pression initiale des gaz et le rapport des volumes des deux chambres. Cette montée peut être très rapide, jusqu'à 10 bars en une milliseconde ce qui correspond au phénomène observé à l'échelle 1. De nombreux travaux de recherche visant à mieux comprendre la phénoménologie de l'onde de souffle, et à mettre en place des procédés d'atténuation ont été menés par ArianeGroup. On pourra se référer par exemple à la publication de Thémiot *et al.* [TLGC17].



FIGURE 5.14 – Dispositif expérimental du banc martel

La figure 5.15 montre une isosurface de pression obtenue lors du calcul par FLUSEPA du développement transitoire d'onde de souffle sur un pas de tir à échelle 1. Les simulations numériques de ce type peuvent prendre jusqu'à 7 jours sur 256 processeurs. Les simulations de l'expérience du banc martel à échelle réduite sont moins complexes et ne nécessitent que quelques jours de calcul. Les maillages utilisés doivent pouvoir capter des fréquences de l'ordre de 100Hz et le nombre de mailles est en général de l'ordre de  $1 \times 10^7$ . Des études de ce type sont donc relativement coûteuses, alors que les zones d'intérêts sont spatialement très réduites mais traversent l'intégralité du maillage. En effet, la tuyère se comporte comme un générateur d'ondes de pression sphériques très énergétiques. L'utilisation d'un module AMR pour raffiner de façon très locale le maillage dans ces ondes de pression permet d'optimiser les calculs effectués. Ainsi, ce cas test permet de vérifier :

---

3. Moyen Aéroacoustique de Recherche et Technologie pour l'Environnement Acoustique des Lanceurs qui est un banc d'essais financé par le Centre National d'Etudes Spatiales construit au Centre d'Etudes Aérodynamiques et Thermiques (CEAT) de l'Université de Poitiers en collaboration avec l'Office National d'Etudes et de Recherches Aérospatiales (ONERA) et ArianeGroup.



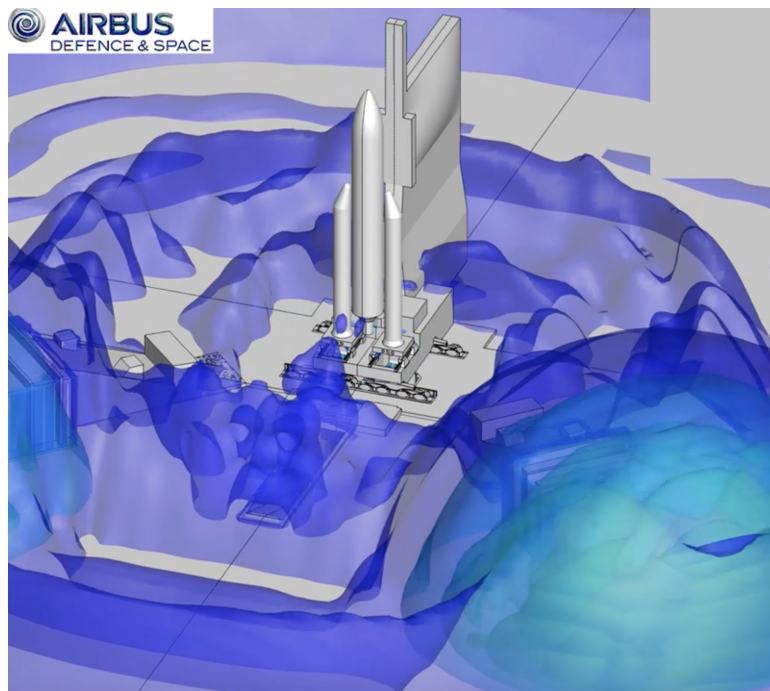


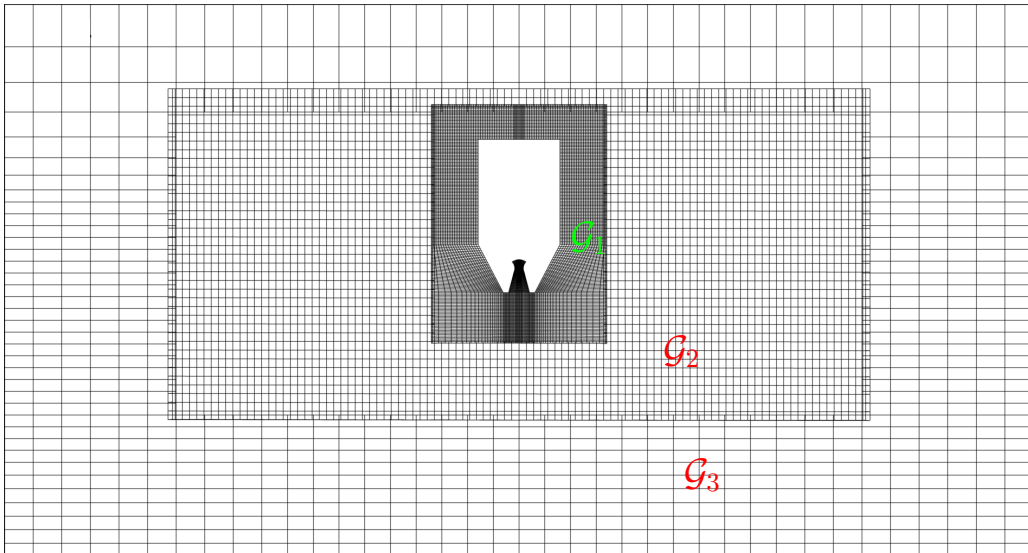
FIGURE 5.15 – Résultats d’une simulation d’onde de souffle par FLUSEPA

1. la stabilité de la stratégie mise en place. En effet la présence d’ondes de pression très énergétiques ne doit pas poser de problèmes pour la projection conservative d’ordre 3 des variables décrite dans le chapitre 4 et ce, malgré la présence d’intersections de maillages. La conservativité de la méthode est primordiale du fait de la présence d’ondes acoustique, une erreur de conservativité pouvant avoir des effets désastreux sur la propagations de celles-ci.
2. le fonctionnement des algorithmes sur une architecture en mémoire distribuée sur une configuration de calcul similaire à celles des calculs industriels d’ArianeGroup.

Le maillage initial du dispositif expérimental est représenté dans sa totalité sur la figure 5.16. Le « bas » du maillage est donc le sol (une paroi) sur laquelle les ondes se réfléchissent. Dans notre calcul FLUSEPA avec module AMR, la tuyère n’est pas à la même altitude que dans l’expérience du banc martel, ce qui fait que le retour de l’onde ne sera pas exploitable. Ce calcul est principalement une démonstration des capacités de calcul et un *stress-test* pour le module AMR. Le maillage est composé de plusieurs cylindres  $\mathcal{G}_i$  imbriqués, dont la résolution diminue à mesure que le diamètre augmente. Les grilles ont été étudiées pour limiter le ratio de taille caractéristique des mailles de part et d’autre des faces d’intersection à un facteur 10. Le développement de critères intégrant l’évolution temporelle des structures caractéristiques devrait améliorer la qualité de l’adaptation de maillage et ainsi augmenter le rapport de tailles caractéristiques des mailles dans les faces d’intersection.

Ce calcul utilise le modèle de turbulence de Spalart-Allmaras et le schéma explicite temporel adaptatif sur 5 niveaux temporels. La taille minimale autorisée par maille pour le module AMR lors du calcul est de  $6 \times 10^{-2}D$ , avec  $D$  le diamètre de sortie de la tuyère. Un calcul avec une résolution équivalente sur tout le maillage comprendrait environ 80 millions de mailles pour la grille  $\mathcal{G}_3$  et n’est aujourd’hui pas envisageable pour des applications récurrentes d’Airbus Safran Launchers. On donne sur les figures 5.17(a), 5.17(b), 5.18(a) et 5.18(b) le déroulement du calcul en représentant le maillage coloré par niveau gradient de pression ce qui permet d’observer à la fois la résolution du maillage étudié et la physique que ce maillage permet de capter.

On peut remarquer la continuité de la résolution du maillage à travers les intersections. En particulier, selon la grilles et la taille caractéristique des mailles qu’elle contient, le niveau de raffinement maximal varie, et ce, par grille. Après passage de l’onde au niveau de l’intersection entre les grilles  $\mathcal{G}_2$  et  $\mathcal{G}_3$ , les mailles ne sont pas déraffinées sur la figure 5.18(b). Ceci est lié à l’utilisation du module de compatibilité entre grilles, décrit dans

FIGURE 5.16 – Maillage initial.  $1.10^5$  cellules

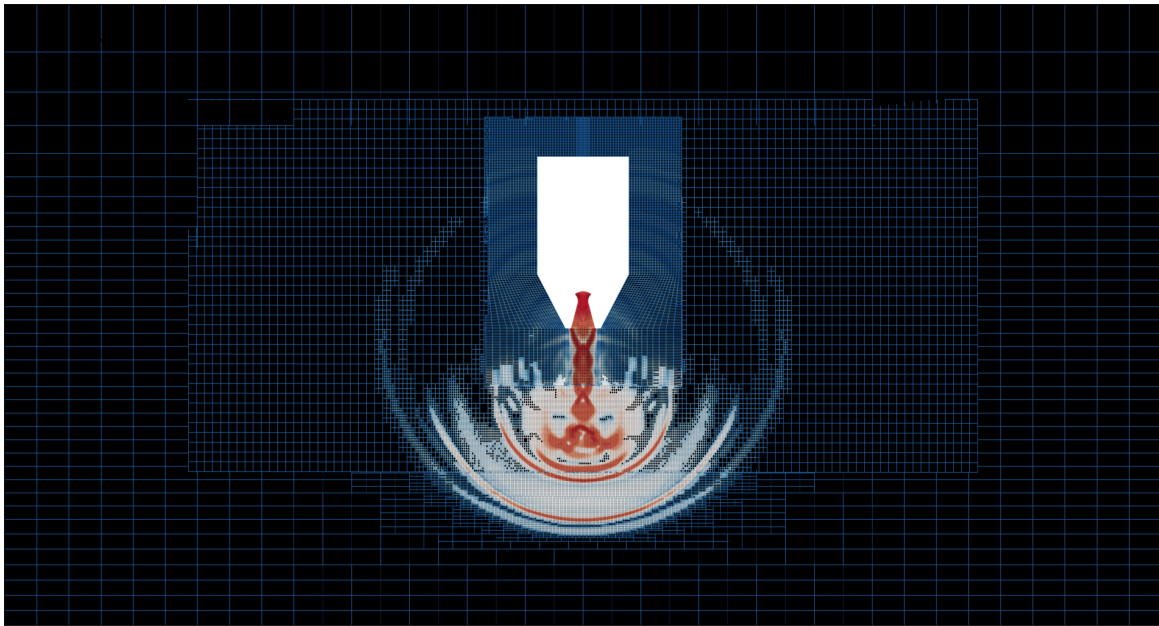
l'annexe B, qui accompagne le passage des structures caractéristiques. Le maillage reste contraint au niveau des intersections intermaillages une fois qu'il a été raffiné.

On représente sur la figure 5.19 l'écart de pression par rapport à la pression ambiante en fonction du temps pour le calcul FLUSEPA avec (trait plein) et sans (trait pointillé) AMR en divers points. Le calcul FLUSEPA sans utilisation du module AMR utilise un maillage de  $1.3 \times 10^7$  mailles qui est beaucoup plus raffiné à proximité de la tuyère que le maillage du calcul utilisant le module AMR. Dans les régions plus éloignées de la tuyère, la densité du maillage décroît très rapidement (typiquement au delà de  $5D$ ) donc des gains conséquents en contenu fréquentiel sont attendus pour ces zones. La comparaison est limitée simplement à la période qui précède la réflexion de l'onde sur la paroi.

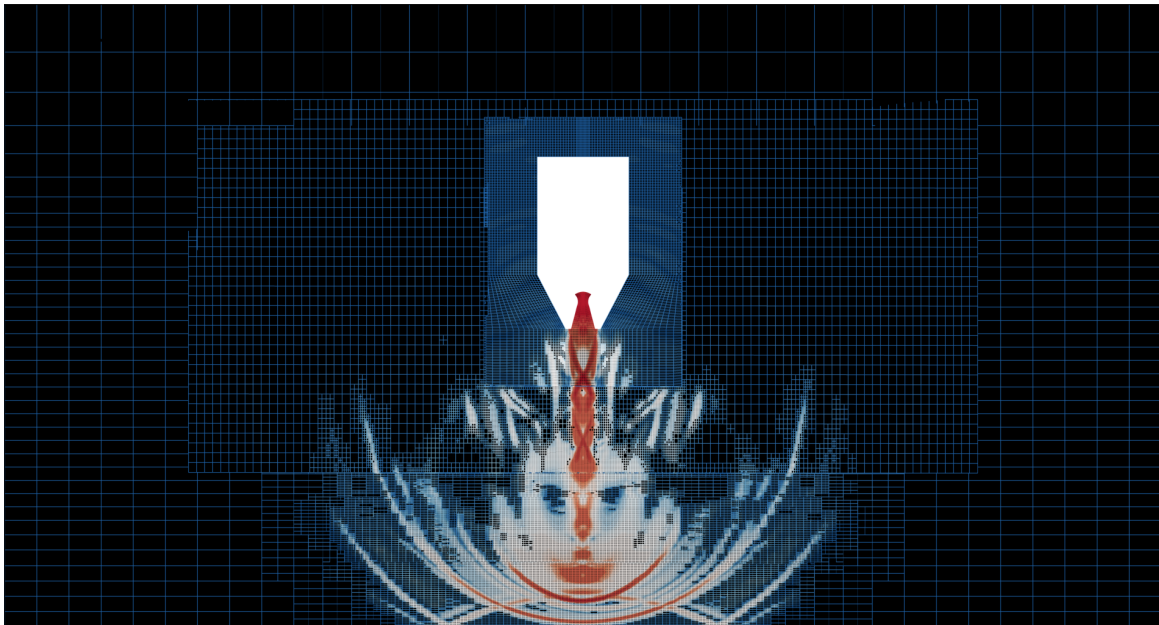
Les résultats ne font pas apparaître de gains significatifs liés à l'utilisation de l'AMR pour le capteur L3 (figure 5.19(a)) qui est positionné près de la paroi, dans la mesure où cette zone présente dans le calcul AMR une densité de maillage supérieure à celle autorisée pour le calcul AMR. Cependant l'amplitude du premier pic est correctement captée par la méthode AMR, or ce pic est dimensionnant pour la structure du booster. Pour les capteurs 5D et 5Dsym, le module AMR permet de capter une deuxième onde très rapprochée, que FLUSEPA sans AMR, ne capte pas, ce qui confirme un contenu fréquentiel supérieur avec AMR. De façon similaire, le capteur 10D présente une meilleure résolution fréquentielle au niveau de la première graduation en temps pour le calcul AMR, avec un pic double que FLUSEPA sans AMR ne permettrait pas de résoudre, ainsi qu'un pic de pression entre la deuxième et la troisième graduation en temps. L'onde de compression se redresse en chocs pour l'AMR de façon naturelle, ce qui n'est pas possible sur le maillage figé (capteur 10D). A grande distance, on constate que l'AMR introduit une correction sur la vitesse de phase du signal et permet de maintenir la fréquence principale dans le temps sans atténuation.

Cette comparaison de simulations, avec et sans AMR, permet de mettre d'ores et déjà en évidence toute l'attention qu'il faut porter au réglage préalable des seuils de raffinement spécifiquement lors l'on effectue une simulation instationnaire, visant à restituer des phénomènes physiques fortement évolutifs.

Un calcul plus abouti avec optimisations algorithmiques devra être effectué. La tuyère sera alors positionnée à l'altitude correspondant à celle du banc martel et la résolution locale sera augmentée, ce qui limitera les différences entre le calcul FLUSEPA sans AMR et le calcul FLUSEPA avec AMR. Ce calcul permettra ainsi de valider la pertinence de l'utilisation d'une méthode de raffinement de maillage adaptatif pour l'étude d'ondes



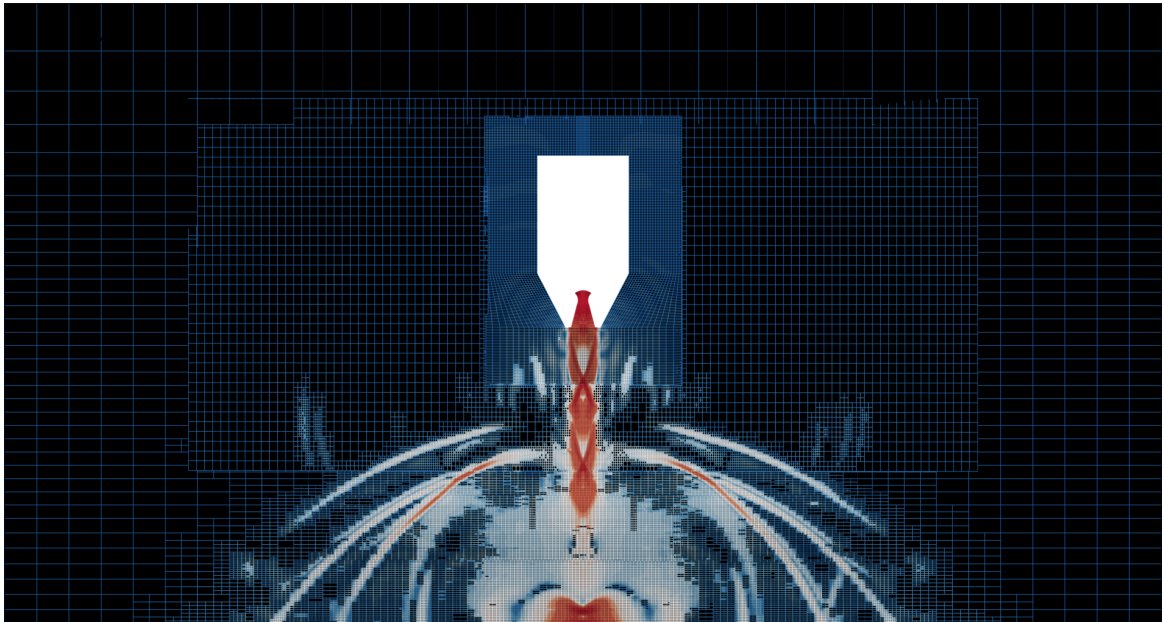
(a)



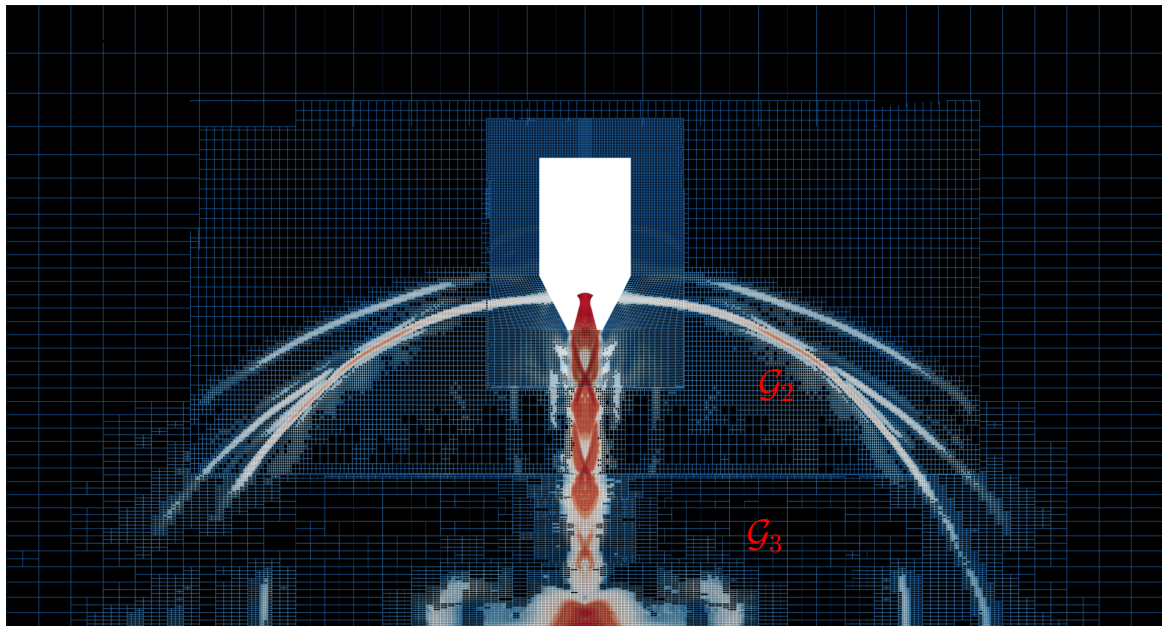
(b)

FIGURE 5.17 – Maillages au cours du calcul, coloriés par niveau de gradient de pression, avant réflexion de l'onde principale

de souffle.

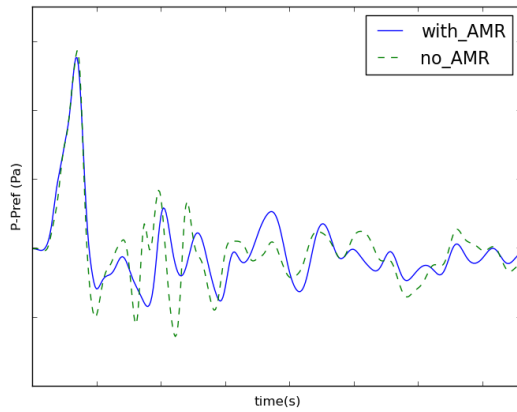


(a)

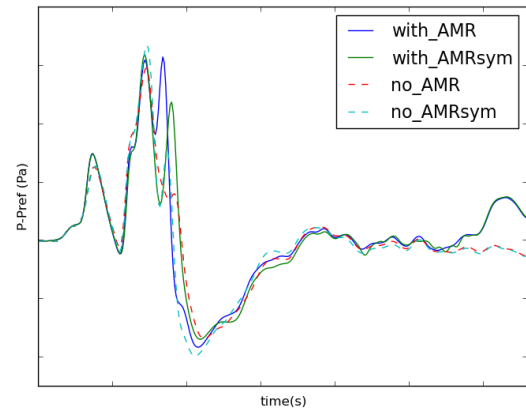


(b)

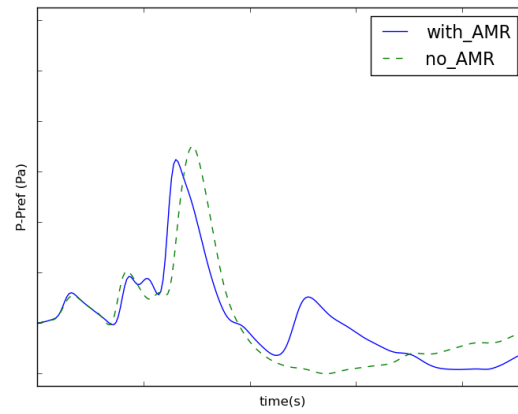
FIGURE 5.18 – Maillages au cours du calcul, coloriés par niveau de gradient de pression, après réflexion de l'onde principale et retour sur la paroi du booster



(a) Capteur L3



(b) Capteur 5D et 5Dsym



(c) Capteur 10D et 10Dsym

FIGURE 5.19 – Signaux temporels de pression. Comparaison des calculs FLUSEPA avec et sans utilisation de l'AMR

## Chapitre 6

# Conclusions et perspectives

Ce manuscrit présente le développement d'une méthode numérique permettant le raffinement adaptatif de maillages. Elle repose sur la création d'arbres *octree* par cellule, s'accompagnant d'un équilibrage 2 – 1 par nœud pour des maillages hexaédriques non structurés. Cette méthode est compatible avec la technique de chevauchement de maillages basée sur le calcul exact des intersections géométriques entre grilles implantée dans FLUSEPA. Le traitement des interfaces non-conformes est totalement transparent puisque chaque maille est définie, dans le solveur natif de FLUSEPA, comme étant un polyèdre au nombre arbitraire de faces. Une technique de redistribution des variables conservatives, après subdivision des mailles a été développée pour permettre une évolution sans accrocs des simulations instationnaires. Elle est précise, strictement conservative et bornée. Cette redistribution a été validée sur deux cas tests. Ces propriétés permettent la simulation efficace de phénomènes instationnaires « raides » (chocs, turbulence), sur des maillages chevauchants par intersection, tout en s'intégrant naturellement aux noyaux de calculs du code FLUSEPA.

Dans le chapitre 1, nous faisons un état de l'art des modules AMR actuels et rappelons les grandes catégories existantes. La plupart du temps, les méthodes d'intégration temporelle adaptative sont couplées aux méthodes AMR, cependant elles sont souvent mises en défaut dans les zones inter-niveaux de raffinement. D'autre part, les techniques actuelles de redistribution des variables associées à l'AMR ne semblent pas satisfaisantes pour nos applications. Enfin, nous citons quelques modules AMR qui ont été codés et couplés à des méthodes de chevauchement de maillages. Ces modules AMR adaptent uniquement des maillages structurés.

Le chapitre 2 expose la stratégie numérique du code FLUSEPA. Ce code est bâti sur une méthode Volume Finis d'ordre élevé, dite  $k$ -exacte, fondée sur une définition originale des propriétés géométriques des éléments. Cette dernière constitue aussi l'un des piliers de la méthode de chevauchement de maillages par intersection dont nous avons décrit les grands principes. Ce recouvrement de maillages conservatif simplifie grandement la génération de maillages et autorise la simulation d'écoulements comprenant de fortes discontinuités sans provoquer d'instabilités parasites liées aux interpolations des méthodes *chimère* classiques. Finalement, un schéma d'intégration temporelle adaptative explicite consistant et conservatif, des modèles de turbulence auto-adaptatifs Hybride RANS/LES, qui pilotent le recentrage du schéma numérique spatial, ainsi qu'une structure logicielle adaptée autorisent une utilisation efficace des grappes de serveurs.

Le chapitre 3 présente le développement du module AMR implanté dans FLUSEPA. Ce module comprend deux composants : le noyau des routines de raffinement *octree* qui est embarqué sur le processus maître et l'ensemble des routines permettant la projection et la redistribution par des arbres de relation de taille variable, qui est intégré aux processus contenant les solveurs aérodynamiques. L'implantation de la méthode AMR a été menée dans un souci constant de préserver les structures algorithmiques de FLUSEPA. Cette conservation des structures existantes assure une portabilité et facilitera les développements futurs. Les performances actuelles et la marge de progression envisageable sont extrêmement encourageantes pour les développements de la stratégie globale d'adaptation.

Le chapitre 4 décrit de façon détaillée un algorithme de redistribution conservative d'ordre  $k$  des variables (consécutif au raffinement) qui respecte le principe du maximum. Cette méthode de projection a pu être étendue

au module d'assemblage / désassemblage lié aux intersections de maillage lors de simulations de corps en mouvement relatif. La validation de l'ordre 3 pour la projection a été effectuée sur deux cas élémentaires d'advection d'un tourbillon à partir d'une grille simple et de deux grilles intersectantes.

Le chapitre 5 présente l'ensemble des résultats sur plusieurs cas d'applications qui discriminent chacune des propriétés souhaitées pour notre méthodologie AMR. La compatibilité du module de raffinement de maillage adaptatif avec le module chevauchement de maillages est ainsi démontrée dans les cas de simulations de turbulence instationnaire et de phénomènes de propagation de chocs.

De manière générale, il apparaît qu'une méthode de raffinement de maillages ouvre à elle seule un vaste champs de recherches quant à la manière de procéder de façon optimale et maîtrisée. De nombreuses pistes d'améliorations ont ainsi été mises en évidence.

## Pistes d'amélioration à court terme

Plusieurs améliorations peuvent être mises en place pour améliorer très rapidement la qualité des simulations, notamment en intégrant la variation de la taille caractéristique des cellules dans la formulation des modèles de turbulence LES, mais aussi en augmentant les types de subdivisions possibles pour mieux suivre l'anisotropie voire la discontinuité des phénomènes étudiés et enfin en tenant compte de la courbure locale de la surface des corps modélisés :

- **Modèles de turbulence LES** : l'équation 2.15 montre que si l'on utilise un modèle de turbulence reposant sur le filtrage spatial  $\Delta$  d'un maillage régulier cartésien,  $\Delta = (\Delta x, \Delta y, \Delta z)$ , alors la subdivision d'une maille va introduire localement une discontinuité, car le terme  $\frac{\partial \Delta}{\partial x_i}$  n'est alors absolument pas négligeable. Les hypothèses sur lesquelles reposent les modèles de turbulence classiques LES ne sont alors plus vérifiées et il devient nécessaire de prendre en compte la non-commutativité des opérateurs de dérivée et de filtrage comme a pu le faire Hamba [Ham03]<sup>1</sup>.
- **Méthodes  $k - d$ -tree** : la méthode *octree* est par essence isotrope et aura tendance à créer une trop grande quantité de mailles pour des phénomènes fortement anisotropes (chocs, couches limites). Les premiers cas tests d'utilisation (voir chapitre 5) ont rapidement montré les limites d'une adaptation par enrichissement isotrope. Pour étendre les capacités du module d'adaptation de maillage par enrichissement, celui-ci doit intégrer d'autres types de subdivisions que la création d'arbres *octree*. Le fait de lier le raffinement des cellules à des arbres de type  $k - d$ -trees semble être une solution prometteuse et permettrait de ne découper les cellules que selon une ou deux directions, plutôt que les 3 de la méthode *octree*.
- **Techniques de scission arbitraire de maille** : Dans les zones contenant des discontinuités (écoulements multi-phasiques, avec chocs, *etc.*), il est possible d'envisager une scission des mailles par le plan qui approxime localement « au mieux » cette discontinuité comme présenté sur la figure 6.1(a). Cette stratégie de scission de maille doit cependant être distinguée des méthodes de *di-tree*, étant donné qu'elle n'introduit pas de notion de niveau de raffinement à proprement parler et que les interfaces ainsi créées au sein des mailles doivent être mobiles pour suivre le déplacement de la discontinuité. Cette bi-section pourra à son tour être couplée à la méthode d'adaptation de maillage par enrichissement pour capter la courbure locale de la discontinuité représentée par l'ensemble de mailles bissectées<sup>2</sup>, figure 6.1(b).

1. Dans le cadre de nos simulations utilisant des modèles HRL et une méthode AMR, nous supposons que les interfaces inter-niveaux de raffinement sont suffisamment éloignées des zones d'intérêt dans lesquelles le modèle devient un modèle sous-maille.

2. Les propriétés topologiques des éléments de part et d'autre de l'interface peuvent ainsi être recalculées par une méthode itérative similaire à celle du calcul des propriétés géométriques après intersection, décrit dans le paragraphe 2.3.4.2. Cette interface interne à la maille permet de poser simplement un problème de Riemann et donc de déterminer la vitesse de déplacement de cette dernière.

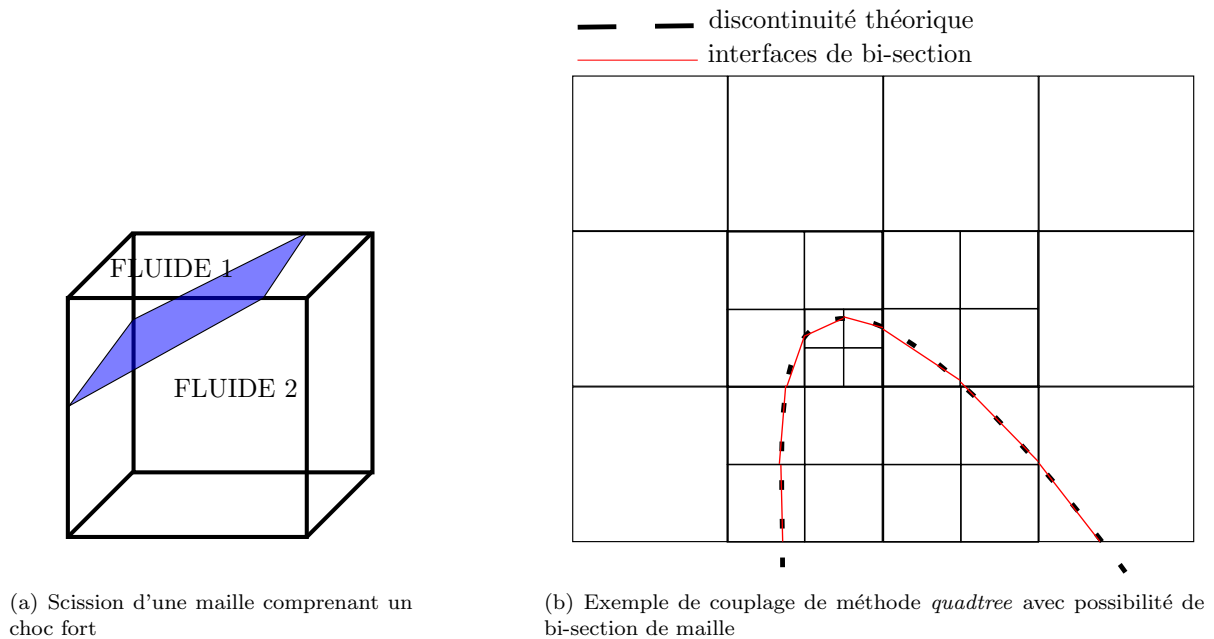


FIGURE 6.1 – Exemple d'applications d'une méthode de bi-section

- **Positionnement des nœuds lors du raffinement de parois courbes** : des travaux sont actuellement en cours et afin d'obtenir une reconstruction (bi)cubique par morceaux des surfaces pariétales pour augmenter l'ordre de précision des schémas numériques de FLUSEPA en paroi. Pour rester compatible avec ces développements et minimiser les pertes de précisions locales, il est impératif, lors de toute subdivision de face pariétale, de replacer les points nouvellement créés sur la vraie géométrie.

## Optimisations logicielles

Le module AMR présenté dans ce manuscrit modifie fortement l'exécution du code. Ainsi, la charge au cours d'un calcul devient variable, ce qui requiert un nouveau travail d'optimisation. A ce titre plusieurs développements sont envisagés.

- **Optimisations de l'utilisation mémoire** : La première version du module de remaillage n'utilise pas de ré-allocation dynamique sur les processus maître et esclaves, ce qui signifie que l'espace alloué initialement doit suffire pour pouvoir effectuer un calcul complet. Une ré-allocation dynamique optimisée devrait non seulement apporter plus de souplesse à la méthode AMR, mais aussi réduire le temps d'accès mémoire.
- **Formalisme en graphe de tâches** : L'architecture logicielle reposant sur un processus maître qui prend l'ensemble des décisions est fortement limitante. Une gestion efficace consisterait à distribuer l'ensemble de la charge aux nœuds de calcul qui pourraient alors effectuer l'intégralité des tâches (intersection, solveur aérodynamique). La granularité des tâches deviendrait alors plus fine, ce qui optimiserait la charge liée aux différents modules et maximiserait la possibilité de recouvrement des communications inter-nœuds.



## Réactivité du module de raffinement pour des cas fortement instationnaires

Nous décrivons ici plusieurs développements nécessaires pour la simulation de phénomènes instationnaires « raides » comprenant des solutions pour la prise en compte de l'instationnarité des phénomènes, *via* des méthodes de correction rapides et de traitement des cas de défaillance dans l'anticipation de l'évolution des structures de l'écoulement.

- **Anticipation du mouvement de structures et raffinement sur plusieurs niveaux** : actuellement, l'application des critères de raffinement/déraffinement n'est effectuée qu'*a posteriori*, la subdivision d'une maille ne s'effectue qu'après la détection d'une sous-résolution par le senseur d'adaptation. Par conséquent, la stratégie d'adaptation n'est au mieux que d'ordre 1 en temps et il peut exister un retard important entre la position réelle d'une structure que l'on cherche à capter et le moment où l'on adapte le maillage. L'utilisation d'une méthode d'extrapolation de Richardson (voir Kamkar [Kam11]) semble être une solution intéressante qui permettrait d'obtenir, pour un coût faible, une anticipation du mouvement de structures. L'idée est d'employer un second solveur sur un maillage déraffiné construit à partir du maillage parent. Cette méthode anticiperait l'évolution des structures fines de l'écoulement et opèrerait une adaptation prévisionnelle sur le maillage fin <sup>3</sup>.
- **Module AMR embarqué dans les solveurs aérodynamiques** : cette organisation logicielle permettrait d'intégrer une méthode d'adaptation très réactive dans les solveurs aérodynamiques. Müller et Stiriba [OS83, MS07] ont introduit un schéma temporel à pas de temps local où les structures fines peuvent se déplacer de l'équivalent d'une maille de niveau temporel le plus élevé au cours d'une macro itération. Le schéma à pas de temps local est alors couplé à une stratégie d'adaptation de maillages AMR. Cependant, chaque nœud de calcul doit disposer de la topologie des mailles à raffiner, ce qui pourrait introduire un surcoût mémoire important pour un nombre d'opérations à effectuer relativement faible.

---

3. On notera que cette diminution de la fréquence d'adaptation peut aussi nécessiter de raffiner certaines mailles sur plusieurs niveaux, au cours d'une seule itération.

# Bibliographie

- [Ada14] M. Adams. Chombo software package for amr applications design document. 2014.
- [AFGM07] F. Alauzet, P. J. Frey, P-L. George, and B. Mohammadi. 3d transient fixed point mesh adaptation for time-dependent problems : Application to cfd simulations. *Journal of Computational Physics*, 222(2) :592–623, 2007.
- [AGM<sup>+</sup>03] F. Alauzet, P-L. George, B. Mohammadi, P. Frey, and H. Borouchaki. Transient fixed point-based unstructured mesh adaptation. *International journal for numerical methods in fluids*, 43(6-7) :729–745, 2003.
- [Ala16] F. Alauzet. A parallel matrix-free conservative solution interpolation on unstructured tetrahedral meshes. *Computer Methods in Applied Mechanics and Engineering*, 299 :116–142, 2016.
- [BBHK15] C. Brehm, M. F. Barad, J. A. Housman, and C. C. Kiris. A comparison of higher-order finite-difference shock capturing schemes. *Computers & Fluids*, 122 :184–208, 2015.
- [BC89] M. J Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. *Journal of computational Physics*, 82(1) :64–84, 1989.
- [BCG87] J. Bell, P. Colella, and H. Glaz. A second-order projection method for viscous, incompressible flow. In *8th Computational Fluid Dynamics Conference*, page 1176, 1987.
- [BCHQ97] D. L. Brown, G. S. Chesshire, W. D. Henshaw, and D. J. Quinlan. Overture : An object-oriented software system for solving partial differential equations in serial and parallel environments. Technical report, Los Alamos National Lab., NM (United States), 1997.
- [Bel05] J. Bell. Amr for low mach number reacting flow. In *Adaptive Mesh Refinement-Theory and Applications*, pages 203–221. Springer, 2005.
- [Ber82] M. Berger. *Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations*. PhD thesis, Stanford University, CA, 1982.
- [BHK16] C. Brehm, J. A. Housman, and C. C. Kiris. Noise generation mechanisms for a supersonic jet impinging on an inclined plate. *Journal of Fluid Mechanics*, 797 :802–850, 2016.
- [Bir31] G. D. Birkhoff. Proof of the ergodic theorem. *Proceedings of the National Academy of Sciences*, 17(12) :656–660, 1931.
- [BL97] M. Berger and R. LeVeque. Amrclaw, adaptive mesh refinement+ clawpack, 1997.
- [BMR<sup>+</sup>16] A. J. Barlow, P-H. Maire, W. J. Rider, R. N. Rieben, and M. J. Shashkov. Arbitrary lagrangian–eulerian methods for modeling high-speed compressible multimaterial flows. *Journal of Computational Physics*, 322 :603–665, 2016.
- [BO84] M. J. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of computational Physics*, 53(3) :484–512, 1984.
- [Bre91] P. Brenner. Three dimensional aerodynamics with moving bodies applied to solid propellant. *27th Joint Propulsion Conference - AIAA*, 2304, 1991.
- [Bre96] P. Brenner. Unsteady flows about bodies in relative motion. In *First AFOSR Conference on Dynamic Motion CFD*, 1996.
- [Bre08] P. Brenner. Race to accuracy : finite volume formulation still in the game. Technical report, ASTRIUM ST, 2008.

- [BWG11] C. Burstedde, L. C Wilcox, and O. Ghattas. p4est : Scalable algorithms for parallel adaptive mesh refinement on forests of octrees. *SIAM Journal on Scientific Computing*, 33(3) :1103–1133, 2011.
- [CB95] W. Chan and P. Buning. Zipper grids for force and moment computation on overset grids. In *12th Computational Fluid Dynamics Conference*, page 1681, 1995.
- [CC16] J-M. Cousteyen Carpaye. *Contribution à la parallélisation et au passage à l'échelle du code FLU-SEPA*. PhD thesis, Université de Bordeaux, 2016.
- [CG12] J. Chicheportiche and X. Gloerfelt. Study of interpolation methods for high-accuracy computations on overlapping grids. *Computers & Fluids*, 68 :112–133, 2012.
- [CGJ<sup>+</sup>09] P. Colella, D. T. Graves, J. N. Johnson, H. S. Johansen, N. D. Keen, T. J. Ligocki, D. F. Martin, P. W. McCorquodale, D. Modiano, P. O. Schwartz, et al. Chombo software package for amr applications design document, 2009.
- [Cha09a] W. Chan. Enhancements to the hybrid mesh approach to surface loads integration on overset structured grids. In *19th AIAA Computational Fluid Dynamics*, page 3990. 2009.
- [Cha09b] W. M. Chan. Overset grid technology development at nasa ames research center. *Computers & Fluids*, 38(3) :496–503, 2009.
- [CKCM01] S. Cieslak, S. B. Khelil, I. Choquet, and A. Merlen. Cut cell strategy for 3-d blast waves numerical simulations. *Shock Waves*, 10(6) :421–429, 2001.
- [CL05] D. Calhoun and R. J. LeVeque. An accuracy study of mesh refinement on mapped grids. In *Adaptive Mesh Refinement-Theory and Applications*, pages 91–101. Springer, 2005.
- [CMG<sup>+</sup>13] S. L. Cornford, D. F. Martin, D. T. Graves, D. F. Ranken, A. M. Le Brocq, R. M. Gladstone, A. J. Payne, E. G. Ng, and W. H. Lipscomb. Adaptive mesh, finite volume modeling of marine ice sheets. *Journal of Computational Physics*, 232(1) :529–549, 2013.
- [Cona] Concept de moteur réutilisable, airbus safran launchers. <https://airbusdefenceandspace.com/reuse-launchers>. Accessed : 2017-05-25.
- [Conb] Concept de premier étage réutilisable par spacex. <https://medium.com/@rsn/>. Accessed : 2017-05-25.
- [Conc] Spacex merlin (& raptor ) engine r&d, gpu-powered. <https://www.youtube.com/watch?v=txk-V01hzBY>. Accessed : 2017-08-08.
- [CPWJ<sup>+</sup>16] D. T. Chan, J. T. Pinier, F. J. Wilcox Jr, D. J. Dalle, S. E. Rogers, and R. J. Gomez III. Space launch system booster separation aerodynamic database development and uncertainty quantification. In *54th AIAA Aerospace Sciences Meeting*, page 0798, 2016.
- [DAB<sup>+</sup>14] A. Dubey, A. Almgren, J. Bell, M. Berzins, S. Brandt, G. Bryan, P. Colella, D. Graves, M. Lijewski, F. Löffler, et al. A survey of high level frameworks in block-structured adaptive mesh refinement packages. *Journal of Parallel and Distributed Computing*, 74(12) :3217–3227, 2014.
- [DBH<sup>+</sup>02] K. Devine, E. Boman, R. Heaphy, B. Hendrickson, and C. Vaughan. Zoltan data management service for parallel dynamic applications. *Computing in Science & Engineering*, 4(2) :90–97, 2002.
- [Dec02] S. Deck. *Simulation numérique des charges latérales instationnaires sur des configurations de lanceur*. PhD thesis, 2002.
- [Dec12] S. Deck. Recent improvements in the zonal detached eddy simulation (zdes) formulation. *Theoretical and Computational Fluid Dynamics*, pages 1–28, 2012.
- [DM98] L. Dagum and R. Menon. Openmp : an industry standard api for shared-memory programming. *IEEE computational science and engineering*, 5(1) :46–55, 1998.
- [DR05] H. B. Dhia and G. Rateau. The arlequin method as a flexible engineering design tool. *International journal for numerical methods in engineering*, 62(11) :1442–1462, 2005.
- [DS16] F. Dmytrus-Saurin. Industrialisation d'une méthode d'adaptation de maillage. Master's thesis, Université Pierre et Marie Curie, PARIS-VI, 2016.

- [DSM87] D. M. Driver, H. L. Seegmiller, and J. G. Marvin. Time-dependent behavior of a reattaching shear layer. *AIAA journal*, 25(7) :914–919, 1987.
- [DZ05] L. J. Dursi and M. Zingale. Efficiency gains from time refinement on amr meshes and explicit timestepping. In *Adaptive Mesh Refinement-Theory and Applications*, pages 103–113. Springer, 2005.
- [els] elsa homepage. <http://elsa.onera.fr/index.html>. Accessed : 2017-08-08.
- [Eme68] Ashley F Emery. An evaluation of several differencing methods for inviscid fluid flow problems. *Journal of Computational Physics*, 2(3) :306–331, 1968.
- [Fid07] K. J. Fidkowski. *A simplex cut-cell adaptive method for high-order discretizations of the compressible Navier-Stokes equations*. PhD thesis, Massachusetts Institute of Technology, 2007.
- [GAL<sup>+</sup>03] T. Goodale, G. Allen, G. Lanfermann, J. Massó, T. Radke, E. Seidel, and J. Shalf. The cactus framework and toolkit : Design and applications. *High Performance Computing for Computational Science—VECPAR 2002*, pages 15–36, 2003.
- [GC03] J. Gullbrand and F. K. Chow. The effect of numerical errors and turbulence models in large-eddy simulations of channel flow, with and without explicit filtering. *Journal of Fluid Mechanics*, 495 :323–341, 2003.
- [God59] S. K. Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 89(3) :271–306, 1959.
- [God76] S. K. Godunov. Résolution des problèmes multidimensionnels de la dynamique des gaz. *Editions MIR, Moscou*, 1976.
- [GZI<sup>+</sup>76] S. K. Godunov, A. V. Zabrodin, M. I. Ivanov, A. N. Kraiko, and G. P. Prokopov. Numerical solution of multidimensional problems of gas dynamics. *Moscow Izyearl Nauka*, 1976.
- [HAC74] C. W. Hirt, A. A. Amsden, and J. L. Cook. An arbitrary lagrangian-eulerian computing method for all flow speeds. *Journal of Computational Physics*, 14(3) :227–253, 1974.
- [HAC97] C. W. Hirt, A. A. Amsden, and J. L. Cook. An arbitrary lagrangian-eulerian computing method for all flow speeds. *Journal of computational physics*, 135(2) :203–216, 1997.
- [Hai09] F. Haider. *Discretisation en maillage non structure general et applications LES*. PhD thesis, Universite Pierre et Marie Curie - PARIS VI, 2009.
- [Ham03] F. Hamba. A hybrid rans/les simulation of turbulent channel flow. *Theoretical and computational fluid dynamics*, 16(5) :387–403, 2003.
- [Har13] M. J. Harris. *Flow feature aligned mesh generation and adaptation*. PhD thesis, University of Sheffield, 2013.
- [HBFM03] S. D. Hall, M. Behnia, C. A. Fletcher, and G. L. Morrison. Investigation of the secondary corner vortex in a benchmark turbulent backward-facing step using cross-correlation particle imaging velocimetry. *Experiments in fluids*, 35(2) :139–151, 2003.
- [HHL11] R. Hartmann, J. Held, and T. Leicht. Adjoint-based error estimation and adaptive mesh refinement for the rans and k- $\omega$  turbulence model equations. *Journal of Computational Physics*, 230(11) :4268–4284, 2011.
- [HP09] T. Holst and T. Pulliam. Overset solution adaptive grid approach applied to hovering rotorcraft flows. In *27th AIAA Applied Aerodynamics Conference*, page 3519, 2009.
- [HPW<sup>+</sup>15] D. Hue, S. Péron, L. Wiant, O. Atinault, E. Gournay, P. Raud, C. Benoit, and J. Mayeur. Validation of a near-body and off-body grid partitioning methodology for aircraft aerodynamic performance prediction. *Computers & Fluids*, 117 :196–211, 2015.
- [HS06] W. D. Henshaw and D. W. Schwendeman. Moving overlapping grids with adaptive mesh refinement for high-speed reactive and non-reactive flow. *Journal of Computational Physics*, 216 :744–779, 2006.

- [HW13] R. E. Harris and B. Williams. Ventus : An overset adaptive cartesian simulation framework for moving boundary problems. In *21st AIAA Computational Fluid Dynamics Conference*, page 2866, 2013.
- [Jou97] J. C. Jouhaud. *Méthode d'Adaptation de Maillages Structurés par Enrichissement*. PhD thesis, University of Bordeaux I, 1997.
- [Jou05] H. Jourden. Hera : a hydrodynamic amr platform for multi-physics simulations. In *Adaptive Mesh Refinement-Theory and Applications*, pages 283–294. Springer, 2005.
- [Kam11] S. J. Kamkar. *Mesh Adaption Strategies for Vortex-Dominated Flows*. PhD thesis, Stanford University, 2011.
- [KBW92] W. L. Kleb, J. T. Batina, and M. H. Williams. Temporal adaptive euler/navier-stokes algorithm involving unstructured dynamic meshes. *AIAA journal*, 30(8) :1980–1985, 1992.
- [KHB<sup>+</sup>16] C. C. Kiris, J. A. Housman, M. F. Barad, C. Brehm, E. Sozer, and S. Moini-Yekta. Computational framework for launch, ascent, and vehicle aerodynamics (lava). *Aerospace Science and Technology*, 55 :189–219, 2016.
- [KJWS10] S. J. Kamkar, A. Jameson, A. M. Wissink, and V. Sankaran. Feature-driven cartesian adaptive mesh refinement in the helios code. In *48th AIAA Aerosciences Conference*, pages 2010–0171, 2010.
- [KK05] Y. Kallinderis and C. Kavouklis. A dynamic adaptation scheme for general 3-d hybrid meshes. *Computer methods in applied mechanics and engineering*, 194(48) :5019–5050, 2005.
- [KK10] C. Kavouklis and Y. Kallinderis. Parallel adaptation of general three-dimensional hybrid meshes. *Journal of Computational Physics*, 229(9) :3454–3473, 2010.
- [KSK97] G. Karypis, K. Schloegel, and V. Kumar. Parmetis : Parallel graph partitioning and sparse matrix ordering library. *Version 1.0, Dept. of Computer Science, University of Minnesota*, 1997.
- [Lan15] S. Landier. Boolean operations on arbitrary polyhedral meshes. *Procedia Engineering*, 124 :200–212, 2015.
- [Lan17] S. Landier. Boolean operations on arbitrary polygonal and polyhedral meshes. *Computer-Aided Design*, 85 :138–153, 2017.
- [LB<sup>+</sup>11] R. J. LeVeque, M. J. Berger, et al. clawpack software version 4.5, 2011.
- [LBB16] A. Limare, P. Brenner, and H. Borouchaki. An adaptive remeshing strategy for unsteady aerodynamics applications. In *46th AIAA Fluid Dynamics Conference*, page 3180, 2016.
- [LCBD16] F. Löffler, Z. Cao, S. R. Brandt, and Z. Du. A new parallelization scheme for adaptive mesh refinement. *Journal of Computational Science*, 16 :79–88, 2016.
- [Lee08] Y-L. Lee. *On overset grids connectivity and vortex tracking in rotorcraft cfd*. University of Maryland, College Park, 2008.
- [LG16] J-M. Le Gouez. High-order overset interpolation via weighted least-square polynomials reconstruction for finite volume cfd. In *54th AIAA Aerospace Sciences Meeting*, page 2051, 2016.
- [LH08] T. Leicht and R. Hartmann. Anisotropic mesh refinement for discontinuous galerkin methods in two-dimensional aerodynamic flow simulations. *International Journal for Numerical Methods in Fluids*, 56(11) :2111–2138, 2008.
- [LH10] T. Leicht and R. Hartmann. Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations. *Journal of Computational Physics*, 229(19) :7344–7360, 2010.
- [LS05] R. Loubère and M. J. Shashkov. A subcell remapping method on staggered polygonal grids for arbitrary-lagrangian–eulerian methods. *Journal of Computational Physics*, 209(1) :105–138, 2005.
- [LSRW16] V. Lakshminarayan, J. Sitaraman, B. Roget, and A. Wissink. Development and validation of a multi-strand solver for complex aerodynamic flows. In *54th AIAA Aerospace Sciences Meeting*, page 1581, 2016.

- [MB15] S. May and M. Berger. An explicit implicit scheme for cut cells in embedded boundary meshes. *Journal of Scientific Computing*, pages 1–25, 2015.
- [MC00] D. F. Martin and P. Colella. A cell-centered adaptive projection method for the incompressible euler equations. *Journal of computational Physics*, 163(2) :271–312, 2000.
- [MCG08] D. F. Martin, P. Colella, and D. Graves. A cell-centered adaptive projection method for the incompressible navier–stokes equations in three dimensions. *Journal of Computational Physics*, 227(3) :1863–1886, 2008.
- [Mea93] R. L. Meakin. Moving body overset grid methods for complete aircraft tiltrotor simulations. *AIAA paper*, 3350 :1993, 1993.
- [Mea98] R. L. Meakin. Composite overset structured grids. In *Handbook of Grid Generation*. CRC Press, 1998.
- [Mea00] R. L. Meakin. Adaptive spatial partitioning and refinement for overset structured grids. *Computer methods in applied mechanics and engineering*, 189(4) :1077–1117, 2000.
- [Mea01] R. Meakin. Automatic off-body grid generation for domains of arbitrary size. In *15th AIAA Computational Fluid Dynamics Conference*, page 2536, 2001.
- [ML89] Y. Moon and M-S. Liou. Conservative treatment of boundary interfaces for overlaid grids and multi-level grid adaptations. In *9th Computational Fluid Dynamics Conference*, page 1980, 1989.
- [MLDB87] P. Moreau, J. Labbe, F. Dupoirieux, and R. Borghi. Experimental and numerical study of a turbulent recirculation zone with combustion. In *Turbulent Shear Flows 5*, pages 337–346. Springer, 1987.
- [MS89] R. Meakin and N. Suhs. Unsteady aerodynamic simulation of multiple bodies in relative motion. In *9th Computational Fluid Dynamics Conference*, page 1996, 1989.
- [MS07] S. Müller and Y. Stiriba. Fully adaptive multiscale schemes for conservation laws employing locally varying time stepping. *Journal of scientific computing*, 30(3) :493–531, 2007.
- [MYBS<sup>+</sup>12] S. Moini-Yekta, M. Barad, E. Sozer, J. Housman, C. Brehm, and C. Kiris. Towards hybrid grid simulations of the launch environment. In *Seventh International Conference on CFD, ICCFD7*, volume 3102, 2012.
- [NBKC09] R. Noack, D. Boger, R. Kunz, and P. Carrica. Suggar++ : An improved general overset grid assembly capability. In *19th AIAA Computational Fluid Dynamics*, page 3992. 2009.
- [OS83] S. Osher and R. Sanders. Numerical approximations to nonlinear conservation laws with locally varying time and space grids. *Mathematics of computation*, 41(164) :321–336, 1983.
- [Pag] Boxlib homepage. <https://ccse.lbl.gov/BoxLib/>. Accessed : 2017-06-10.
- [PB79] J. W. Purvis and J. E. Burkhhalter. Prediction of critical mach number for store configurations. *AIAA Journal*, 17(11) :1170–1177, 1979.
- [PB89] M. Pollet and P. Brenner. Aerodynamics with moving bodies applied to solid propulsion. In *25th Joint propulsion conference*, 1989.
- [PB13] S. Peron and C. Benoit. Automatic off-body overset adaptive cartesian mesh method based on an octree approach. *Journal of Computational Physics*, 232(1) :153–173, 2013.
- [PBCR17] G. Pont, P. Brenner, P. Cinnella, and J-C. Robinet. Multiple-correction hybrid k-exact schemes for high-order compressible rans-les simulations on general unstructured grids. *Journal of Computational Physics*, 2017. Article under review.
- [PBG<sup>+</sup>16] S. Péron, C. Benoit, V. Gleize, I. Mary, and M. Terracol. A mixed overset grid/immersed boundary approach for cfd simulations of complex geometries. In *54th AIAA Aerospace Sciences Meeting*, page 2055, 2016.
- [Per14] S. Peron. *Méthode d’assemblage de maillages recouvrants autour de géométries complexes pour des simulations en aérodynamique compressible*. PhD thesis, Paris, ENSAM, 2014.

- [PLK<sup>+</sup>16] M. A. Park, A. Loseille, J. Krakos, T. R. Michal, and J. J. Alonso. Unstructured grid adaptation : Status, potential impacts, and recommended investments towards cfd 2030. In *46th AIAA Fluid Dynamics Conference*, page 3323, 2016.
- [Pon15] G. Pont. *Self-adaptive turbulence models for unsteady compressible flows*. PhD thesis, DynFluid, Arts et Metiers ParisTech, 2015.
- [Pop11] S. Popinet. Quadtree-adaptive tsunami modelling. *Ocean Dynamics*, 61(9) :1261–1285, 2011.
- [PR96] F. Pellegrini and J. Roman. Scotch : A software package for static mapping by dual recursive bipartitioning of process and architecture graphs. In *International Conference on High-Performance Computing and Networking*, pages 493–498. Springer, 1996.
- [Rai86] M. M. Rai. A conservative treatment of zonal boundaries for euler equation calculations. *Journal of Computational Physics*, 62(2) :472–503, 1986.
- [RDG17] B. Re, C. Dobrzynski, and A. Guardone. An interpolation-free ale scheme for unsteady inviscid flows computations with large boundary displacements over three-dimensional adaptive grids. *Journal of Computational Physics*, 340 :26–54, 2017.
- [RNO<sup>+</sup>17] L. Ramírez, X. Nogueira, P. Ouro, F. Navarrina, S. Khelladi, and I. Colominas. A higher-order chimera method for finite volume schemes. *Archives of Computational Methods in Engineering*, pages 1–16, 2017.
- [RSD03] S. E. Rogers, N. E. Suhs, and W. E. Dietz. Pegasus 5 : an automated preprocessor for overset-grid computational fluid dynamics. *AIAA journal*, 41(6) :1037–1045, 2003.
- [SBK14] E. Sozer, C. Brehm, and C. C. Kiris. Gradient calculation methods on arbitrary polyhedral unstructured meshes for cell-centered cfd solvers. In *52nd Aerospace Sciences Meeting, no. AIAA*, volume 1440, 2014.
- [SDB83] J. L. Steger, F. C. Dougherty, and J. A. Benek. A chimera grid scheme. 1983.
- [SDS<sup>+</sup>06] P. R. Spalart, S. Deck, M. L. Shur, K. D. Squires, M. K. Strelets, and A. Travin. A new version of detached-eddy simulation, resistant to ambiguous grid densities. *Theoretical and computational Fluid Dynamics*, 20 :181–195, 2006.
- [SJS<sup>+</sup>97] P. Spalart, W. Jou, M. Strelets, Allmaras S., et al. Comments on the feasibility of les for wings, and on a hybrid rans/les approach. *Advances in DNS/LES*, 1 :4–8, 1997.
- [SKA<sup>+</sup>14] J. Slotnick, A. Khodadoust, J. Alonso, D. Darmofal, W. Gropp, E. Lurie, and D. Mavriplis. Cfd vision 2030 study : a path to revolutionary computational aerosciences. 2014.
- [SRBDD09] B. Sainte-Rose, N. Bertier, S. Deck, and Dupoirieux. A des method applied to a backward facing step reactive flow. *Combustion for aerospace propulsion*, 337 :340–351, 2009.
- [SSP14] R. Shenoy, M. J. Smith, and M. A. Park. Unstructured overset mesh adaptation with turbulence modeling for unsteady aerodynamic interactions. *Journal of Aircraft*, 51(1) :161–174, 2014.
- [SSST08] M. L. Shur, P. R. Spalart, M. K. Strelets, and A. K. Travin. A hybrid rans-les approach with delayed-des and wall-modelled les capabilities. *International Journal of Heat and Fluid Flow*, 29(6) :1638–1649, 2008.
- [STA] Statistiques annuelles de mises en orbite. <http://www.spacelaunchreport.com/logyear.html>. Accessed : 2017-06-07.
- [Sta77] G. Starius. Composite mesh difference methods for elliptic boundary value problems. *Numerische Mathematik*, 28(2) :243–258, 1977.
- [Sta80] G. Starius. On composite mesh difference methods for hyperbolic differential equations. *Numerische Mathematik*, 35(3) :241–255, 1980.
- [TDZGP06] G. Tóth, D. L. De Zeeuw, T. I. Gombosi, and K. G. Powell. A parallel explicit/implicit time stepping scheme on block-adaptive grids. *Journal of Computational Physics*, 217(2) :722–758, 2006.

- [TDZM<sup>+</sup>08] G. Toth, D. L. De Zeeuw, Y-J. Ma, T. I. Gombosi, and K. G. Powell. Parallel explicit/implicit time stepping scheme on block-adaptive grids. In *Numerical Modeling of Space Plasma Flows*, volume 385, page 237, 2008.
- [Tey02] R. Teyssier. Cosmological hydrodynamics with adaptive mesh refinement—a new high resolution code called ramses. *Astronomy & Astrophysics*, 385(1) :337–364, 2002.
- [TLGC17] C. Themiot, V. Le Gallo, and J-M. Carrac. Design optimization and test campaign of a 1/50th ariane 5 representative dynamic model subjected to blast waves. In Milan, editor, *7th European Conference for aeronautics and space sciences (EUCASS)*, July 2017.
- [TSG<sup>+</sup>05] G. Tóth, I. V. Sokolov, T. I. Gombosi, D. R. Chesney, C. R. Clauer, D. L. De Zeeuw, K. C Hansen, K. J Kane, W. B. Manchester, R. C. Oehmke, et al. Space weather modeling framework : A new tool for the space science community. *Journal of Geophysical Research : Space Physics*, 110(A12), 2005.
- [VL79] B. Van Leer. Towards the ultimate conservative difference scheme. v. a second-order sequel to godunov’s method. *Journal of computational Physics*, 32(1) :101–136, 1979.
- [VLS13] J. Velechovský, R. Liska, and M. Shashkov. High-order remapping with piece-wise parabolic reconstruction. *Computers & Fluids*, 83 :164–169, 2013.
- [Vol66] E. A. Volkov. Net method for finite and infinite regions with piecewise smooth boundary. *Doklady Akademii Nauk SSSR*, 168(5) :978, 1966.
- [Vol68] E. A. Volkov. The method of composite meshes for finite and infinite regions with piecewise smooth boundary. *Trudy Matematicheskogo Instituta imeni VA Steklova*, 96 :117–148, 1968.
- [W<sup>+</sup>98] D. C. Wilcox et al. *Turbulence modeling for CFD*, volume 2. DCW industries La Canada, CA, 1998.
- [WC84] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of computational physics*, 54(1) :115–173, 1984.
- [WH00] A. M. Wissink and R. Hornung. Samrai : A framework for developing parallel amr applications. In *5th Symposium on Overset Grids and Solution Technology*, pages 18–20, 2000.
- [WJK<sup>+</sup>13] P. R. Woodward, J. Jayaraj, M. Knox, D. H. Porter, C. L. Fryer, G. Dimonte, C. C. Joggerst, G. M. Rockefeller, W. W. Dai, R. J. Kares, et al. Simulating turbulent mixing from richtmyer-meshkov and rayleigh-taylor instabilities in converging geometries using moving cartesian grids. Technical report, Los Alamos National Laboratory (LANL), 2013.
- [WKP<sup>+</sup>10] A. M. Wissink, S. Kamkar, T. H. Pulliam, J. Sitaraman, and V. Sankaran. Cartesian adaptive mesh refinement for rotorcraft wake resolution. In *28th Applied Aerodynamics Conference*, 2010.
- [YCI00] G. Yang, D. M. Causon, and D. M. Ingram. Calculation of compressible flows about complex moving geometries using a three-dimensional cartesian cut cell method. *International Journal for Numerical Methods in Fluids*, 33(8) :1121–1151, 2000.





# Annexe A

## Critères d'adaptation

Les critères simples qui ont été utilisés au cours de la thèse sont répertoriés dans cette Annexe.

### A.1 Critère seuil de gradient de pression

Le critère le plus simple utilisé pour valider l'implémentation des modules de raffinement est le suivant :

$$\text{si } |D^{(1)}P| \begin{cases} < \text{seuil}_1, \text{ alors déraffinement} \\ > \text{seuil}_2, \text{ alors raffinement} \end{cases} \quad (\text{A.1})$$

avec  $|\cdot| = \sqrt{(D^{(1)}P_1)^2 + (D^{(1)}P_2)^2 + (D^{(1)}P_3)^2}$  la norme 2 du vecteur gradient de dérivée de la pression

### A.2 Critère gradient de pression scalaire vitesse

Plus intéressant pour des ondes de chocs, il permet de moins capter les ondes acoustiques lors du raffinement que le critère précédent :

$$\text{si } \mathbf{u} \cdot D^{(1)}P \begin{cases} < \text{seuil}_1, \text{ alors déraffinement} \\ > \text{seuil}_2, \text{ alors raffinement} \end{cases} \quad (\text{A.2})$$

Utilisé par exemple pour des simulations d'onde de souffle lors de l'allumage de moteurs fusée.

### A.3 Critère basé sur le Hessien de la vitesse

Critère simple utilisé pour l'intégration des modules et les cas tests de non-régression.

$$\text{si } |D^{(2)}\mathbf{u}|_{\mathbb{F}} \begin{cases} < \text{seuil}_1, \text{ alors déraffinement} \\ > \text{seuil}_2, \text{ alors raffinement} \end{cases} \quad (\text{A.3})$$

Ici ,  $|\cdot|_{\mathbb{F}}$  est la norme de Froebenius,  $|D^{(2)}\mathbf{u}|_{\mathbb{F}} = \sqrt{\sum_{i=1}^3 \sum_{j=1}^3 \left( \frac{\partial^2 u_i}{\partial x_j \partial x_i} \right)^2}$

### A.4 Critère basé sur la taille de Von Kármán

Critère basé sur la taille de Von Kármán permettant d'adapter le maillage dans des zones turbulentes. On s'assure que, dans le maillage courant qui effectue la simulation de l'écoulement, on ait :

$$\alpha L_{vK} \leq h \leq \beta L_{vK} \quad (\text{A.4})$$

où  $L_{vK}$  est l'échelle de Von Kármán ,qui est définie par la relation :

$$L_{vK} = K \frac{|U'|}{|U''|} \quad (\text{A.5})$$

$$\text{with } U'' = \sqrt{\frac{\partial^2 U_i}{\partial x_k^2} \frac{\partial^2 U_i}{\partial x_j^2}}, U' = S = \sqrt{2.S_{ij}S_{ij}}$$

$$\text{and } S_{ij} = \frac{1}{2} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right)$$

avec  $K = 0.41$  qui est la constante de Von Kármán,  $\alpha$  et  $\beta$  qui sont des constantes définies par l'utilisateur et  $S = \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i})$  la partie symétrique du tenseur  $D^{(1)\mathbf{u}}$  du gradient des vitesses. Dans nos simulations on a principalement utilisé  $\alpha = 0.6$  et  $\beta = 0.9$ . Cette échelle est intéressante, dans le sens où elle utilise une taille caractéristique de la turbulence et permet de lier le maillage avec la physique à simuler. En pratique pour un modèle de turbulence hybride RANS/LES, ce genre d'échelle renvoie la taille nécessaire pour les zones RANS et une taille plus fine, pour les zones LES.

## Annexe B

# Module de compatibilité géométrique d'intersections

Les règles d'équilibrage 2 – 1 du module AMR permettent d'assurer une transition « douce » des tailles de mailles (cf. 3.2.5), mais cette transition est basée sur le niveau de raffinement des mailles et ne s'applique qu'au sein d'une même grille. Il peut donc exister, dans des zones raffinées, de fortes disparités au niveau des faces d'intersection entre grilles comme présenté sur la figure B.1. Un module de prétraitement de maillages a donc été ajouté à FLUSEPA et permet d'assurer la compatibilité au niveau des intersections, en terme de taille caractéristique<sup>1</sup> de maille. Pour les faces concernées, cette taille ne doit pas varier trop brutalement.

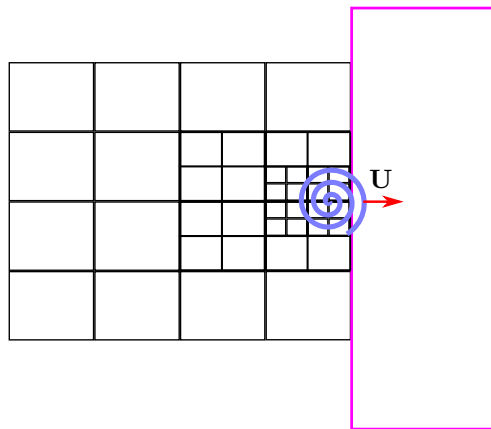


FIGURE B.1 – Advection d'une structure tourbillonnaire. En noir : grille  $\mathcal{G}_1$ , en magenta : grille  $\mathcal{G}_2$

### B.1 Critère de compatibilité simple

Pour toute interface d'intersection de maillage  $\mathcal{F}_{\alpha\beta}$  reliant deux cellules  $\mathcal{T}_\alpha$  et  $\mathcal{T}_\beta$ , on impose :

$$C_1 h_{\mathcal{T}_\beta} \leq h_{\mathcal{T}_\alpha} \leq C_2 h_{\mathcal{T}_\beta} \quad (\text{B.1})$$

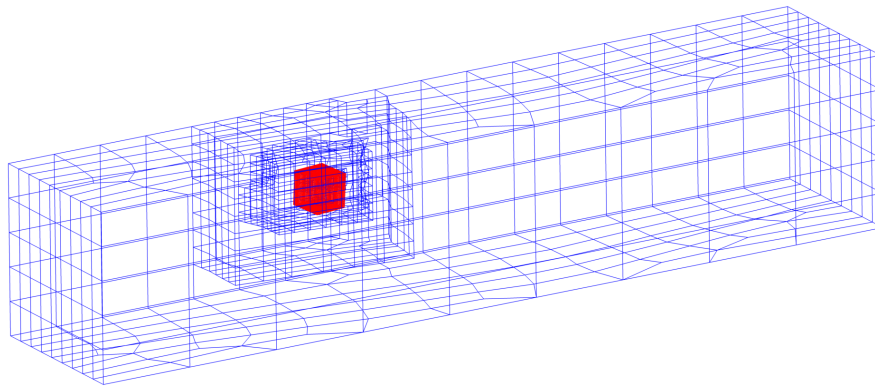
en notant  $h_{\mathcal{T}_\alpha}$  et  $h_{\mathcal{T}_\beta}$  les tailles caractéristiques des cellules  $\mathcal{T}_\alpha$  et  $\mathcal{T}_\beta$  respectivement et  $C_1 \simeq 1.3$  et  $C_2 \simeq 2.5$  deux constantes.

---

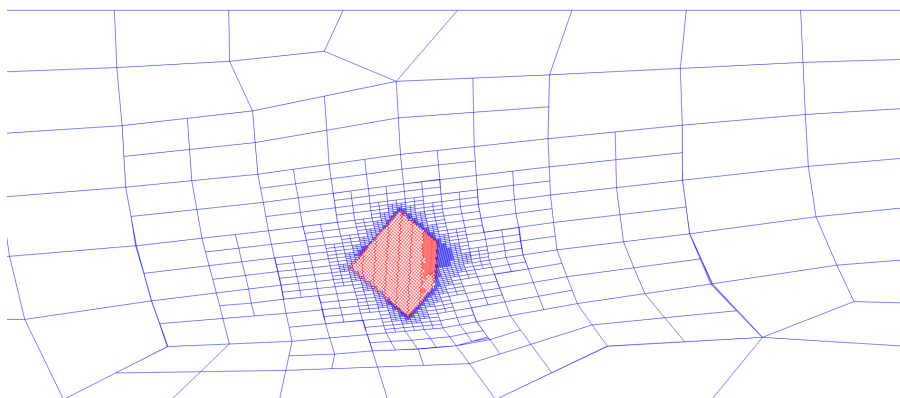
1. Cette taille caractéristique peut être définie à partir d'un phénomène physique ou d'une erreur d'extrapolation admissible

Ce critère permet simplement la validation de la propagation d'un senseur au niveau des intersections. Ici, la contrainte est purement géométrique et assure une progression de la taille des mailles au niveau des faces d'intersection indépendamment de l'écoulement. La validation fonctionnelle de ce critère a été réalisée sur un cas simple. Il s'agit d'un cube maillé contenant un maillage cartésien très fin en intersection avec une grille de fond parallélépipédique contenant un maillage hexaédrique très grossier. Afin d'assurer la compatibilité des tailles de maille au niveau des intersections, le module AMR doit raffiner la grille de fond sur 6 niveaux.

On visualise le maillage obtenu après compatibilité géométrique à la figure B.2. On peut remarquer qu'il y a aussi la présence de mailles raffinées dans le cube, qui sont liées à la présence de très petites mailles après raffinement des mailles de la grille de fond. Ce cas particulier est expliqué sur la figure B.3. On suppose deux mailles  $\mathcal{T}_\alpha$  et  $\mathcal{T}_\beta$  de tailles caractéristiques respectives  $h$  et  $h_2$ , appartenant à deux grilles distinctes B.3(a). L'utilisation du critère de compatibilité géométrique simple impose un raffinement de la maille  $\mathcal{T}_\beta$ , ce qui conduit au cas présenté sur la figure B.3(b). La maille  $\mathcal{T}_\gamma$  de taille caractéristique  $h_3$  ainsi obtenue impose alors à son tour le raffinement de la cellule  $\mathcal{T}_\delta$  de taille  $h_4$  comme présenté sur la figure B.3(c). Il y a donc sur-contrainte du maillage après application du critère de compatibilité géométrique simple.



(a) vue 3D



(b) vue 2D

FIGURE B.2 – Maillage obtenu après application du critère de compatibilité géométrique simple. Grilles colorées par niveau de priorité

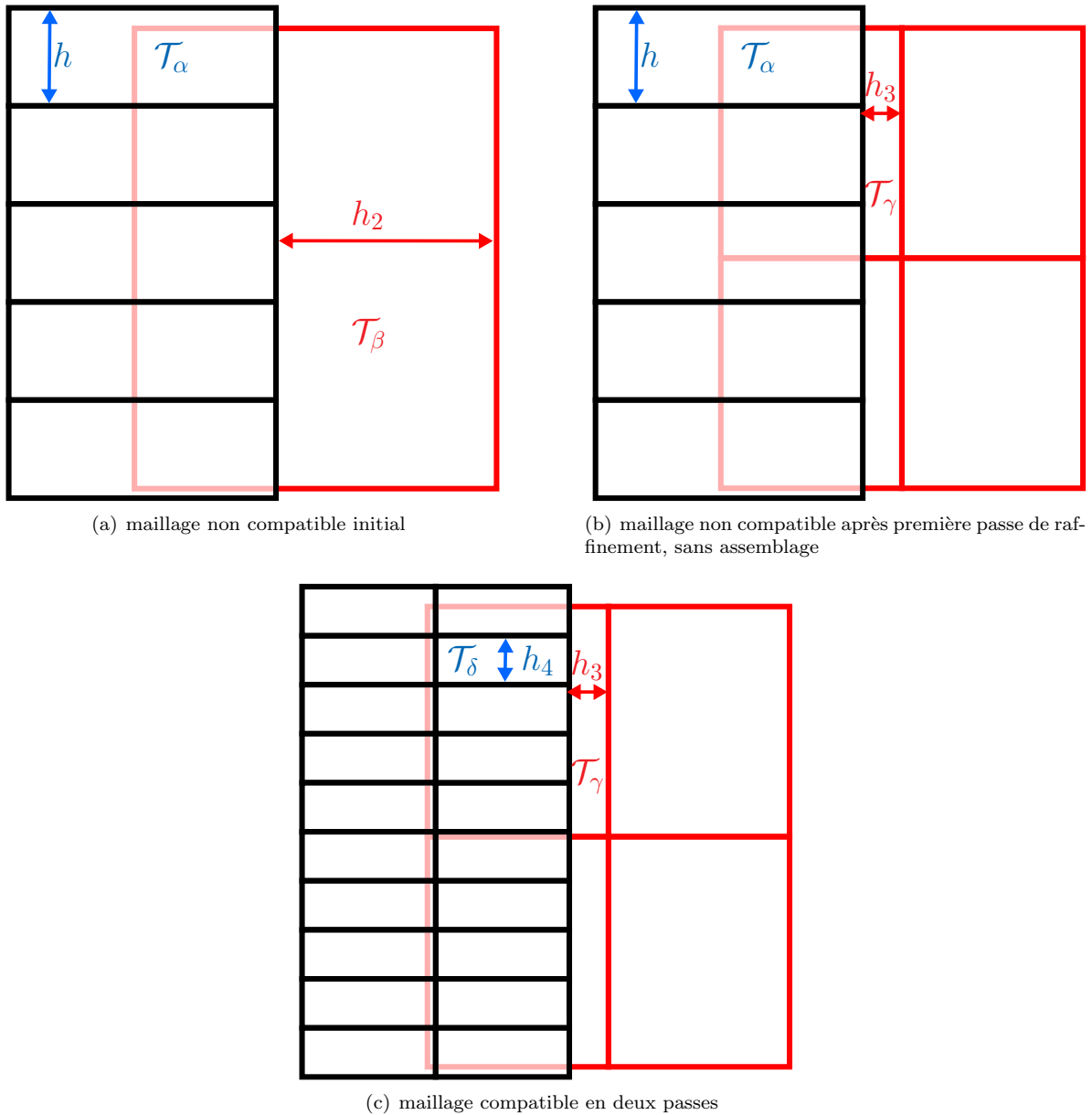


FIGURE B.3 – Propagation de raffinement au niveau d'une intersection

## B.2 Compatibilité géométrique pour le passage de structures caractéristiques

Pour toute interface d'intersection de maillage  $\mathcal{F}_{\alpha\beta}$  reliant deux cellules  $\mathcal{T}_\alpha$  et  $\mathcal{T}_\beta$ , si  $\mathbf{IRAFF}(\mathcal{T}_\alpha)$  ou  $\mathbf{IRAFF}(\mathcal{T}_\beta) \geq 1$  alors :

$$C_1 h_{\mathcal{T}_\beta} \leq h_{\mathcal{T}_\alpha} \leq C_2 h_{\mathcal{T}_\beta} \quad (\text{B.2})$$

en notant  $h_{\mathcal{T}_\alpha}$  et  $h_{\mathcal{T}_\beta}$  les tailles caractéristiques des cellules  $\mathcal{T}_\alpha$  et  $\mathcal{T}_\beta$  respectivement et  $C_1, C_2$  deux constantes fixées respectivement à 1.3 et 2.5.

Pour des maillages fixes, cette compatibilité géométrique conduit à un surcoût négligeable en terme de nombre de mailles ( $< 1\%$ ) et permet simplement d'adapter le maillage au niveau des intersections quand une structure détectée par les senseurs est proche. En effet, ce critère de raffinement ne s'applique que si une des mailles de part et d'autre de l'intersection a déjà été raffinée au moins une fois ( $\mathbf{IRAFF} \geq 1$ ). La figure 5.18(b) illustre le maillage obtenu après application de ce critère de compatibilité dans le cas du banc martel simplifié.

On peut remarquer que dans les zones où il n'y a pas encore eu le passage des chocs, la compatibilité inter-maillage n'est pas assurée, mais que les mailles restent raffinées dans les zones d'intersection une fois qu'un choc a traversé une face d'intersection.

## Annexe C

### Relation entre $\tilde{q}$ et $\bar{q}$

L'ensemble de cette Annexe utilise les expressions introduites dans le paragraphe 2.2.1.5 et le chapitre 4. Plus particulièrement nous développons les expressions

$$\tilde{q} = f(\bar{w}), \quad \begin{cases} \tilde{P}_\alpha &= (\gamma - 1) \left( (\overline{\rho E})_\alpha - \frac{1}{2} \frac{|\overline{\rho \mathbf{u}}|_\alpha^2}{\bar{\rho}_\alpha} \right) \\ \tilde{T}_\alpha &= \frac{\overline{P}_\alpha}{R \bar{\rho}_\alpha} \\ \tilde{\mathbf{u}}_\alpha &= \frac{(\overline{\rho \mathbf{u}})_\alpha}{\bar{\rho}_\alpha} \end{cases}$$

avec  $\bar{q}_\alpha = \frac{1}{|\mathcal{T}_\alpha|} \iiint_{\mathcal{T}_\alpha} q dV$  à un ordre  $n$ . Enfin nous explicitons les formules obtenues à l'ordre 3, pour les besoins propres aux travaux décrits dans ce manuscrit.

#### C.1 Relation pour la vitesse

L'équation 2.43 évaluée pour la vitesse est :

$$\bar{u}_\alpha = u(G_\alpha) + \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{V}_\alpha^{(m)} : D^{(m)} u \Big|_{x_G} + \mathcal{O}(h^n). \quad (\text{C.1})$$

En rappelant :

$$\tilde{u}_\alpha = \frac{(\overline{\rho u})_\alpha}{\bar{\rho}_\alpha} \quad (\text{C.2})$$

On effectue un développement de Taylor de  $\rho$ ,  $\bar{\rho u}$  et l'expression précédente devient :

$$\tilde{u}_\alpha = u_\alpha - \frac{u_\alpha}{\bar{\rho}_\alpha} \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{V}_\alpha^{(m)} : D^{(m)} \rho \Big|_{G_\alpha} + \frac{1}{\bar{\rho}_\alpha} \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{V}_\alpha^{(m)} : D^{(m)} \rho \mathbf{u} \Big|_{G_\alpha} + \mathcal{O}(h^n) \quad (\text{C.3})$$

Par soustraction de C.3 à C.1, on obtient :

$$\bar{u}_\alpha - \tilde{u}_\alpha = \Delta u^{m-2} - \frac{1}{\bar{\rho}_\alpha} \mathcal{V}_\alpha^{(m-1)} : \left( \sum_{l=1}^{n-2} \binom{n-1}{l} D^{(m-1-l)} \rho \Big|_{x_G} \otimes D^{(l)} u \Big|_{G_\alpha} \right) + \mathcal{O}(h^n), \quad (\text{C.4})$$

avec :

$$\Delta u^{m-2} = -\frac{1}{\bar{\rho}_\alpha} \left( \sum_{m=2}^{n-2} \frac{1}{m!} \mathcal{V}_\alpha^{(m)} : \left( \sum_{l=0}^m \binom{m}{l} D^{(m-l)} \rho \Big|_{G_\alpha} \otimes D^{(l)} u \Big|_{G_\alpha} - \bar{u}_\alpha D^{(m)} \rho \Big|_{G_\alpha} - \bar{\rho}_\alpha D^{(m)} u \Big|_{G_\alpha} \right) \right) \quad (\text{C.5})$$



L'expression tenseur de densité de densité  $D^{(m)}\rho|_{x_G}$  peut être déduite de l'équation d'état :

$$\rho = \frac{P}{RT}$$

dont on déduit :

$$D^{(m)}\rho|_{G_\alpha} = \frac{1}{R} \sum_{l=0}^m \binom{m}{l} D^{(m-l)}P|_{G_\alpha} \otimes \left( (-1)^l l! T_\alpha^{-(l+1)} D^{(m)}T|_{G_\alpha} \right) \quad (\text{C.6})$$

Pour une reconstruction 3-exacte,  $\Delta u^{m-2} = 0$  ainsi :

$$\bar{u}_\alpha - \tilde{u}_\alpha = -\frac{1}{\rho_\alpha} \mathcal{V}_\alpha^{(2)} : \left( D^{(1)}\rho|_{G_\alpha} \otimes D^{(1)}u|_{G_\alpha} \right) \quad (\text{C.7})$$

qui nécessite une évaluation au premier ordre des gradients au point  $G_\alpha$ , pour obtenir une précision du troisième ordre.

## C.2 Relation pour la pression

En utilisation la relation 2.43 et l'équation :

$$\tilde{P}_\alpha = (\gamma - 1) \left( (\rho E)_\alpha - \frac{1}{2} \frac{|\overline{\rho u}_\alpha|^2}{\rho_\alpha} \right) \quad (\text{C.8})$$

par développement de Taylor, on obtient :

$$\tilde{P}_\alpha = P_\alpha - (\gamma - 1) \frac{\rho_\alpha |u_\alpha|}{\rho_\alpha} \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{V}_\alpha^{(m)} : D^{(m)}\rho|_\alpha + (\gamma - 1) \left( \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{V}_\alpha^{(m)} : \left( D^{(m)}(\rho E)|_\alpha - \frac{1}{2\rho_\alpha} D^{(m)}|\rho u_\alpha|^2|_{G_\alpha} \right) \right) \quad (\text{C.9})$$

De façon similaire au calcul pour la vitesse, on obtient :

$$\bar{P}_\alpha - \tilde{P}_\alpha = \Delta P^{n-2} - \frac{\gamma - 1}{\bar{\rho}_\alpha} \frac{\mathcal{V}_\alpha^{(m-1)}}{(n-1)!} : \left( \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{V}_\alpha^{(m)} : \left( D^{(m)}\rho E|_{G_\alpha} - \frac{1}{2\rho_\alpha} D^{(m)}|\rho u|^2|_{G_\alpha} \right) \right) + \mathcal{O}(h^n) \quad (\text{C.10})$$

avec :

$$\Delta P^{n-2} = - \sum_{m=2}^{n-2} \frac{1}{m!} \mathcal{V}_\alpha^{(m)} : \left( \frac{\gamma - 1}{\rho_\alpha} \left( \frac{\rho_\alpha}{\rho_\alpha} D^{(m)}(\rho E)|_{G_\alpha} + \rho_\alpha + \frac{|u_\alpha|^2}{2} D^{(m)}\rho|_{G_\alpha} - \frac{1}{2} D^{(m)}|\rho u_\alpha|^2|_{G_\alpha} \right) - D^{(m)}P|_{G_\alpha} \right) \quad (\text{C.11})$$

Les dérivées de produits  $D^{(m)}(\rho E)$  ou  $D^{(m)}|\rho u_\alpha|^2$  peuvent être développé en n'utilisant que les variables primitives :

$$D^{(m)}|\rho u_\alpha|^2|_{G_\alpha} = \sum_{l=0}^m \binom{m}{l} D^{(m-l)}\rho|_\alpha \otimes D^{(l)}|u_\alpha|^2|_{G_\alpha} \quad (\text{C.12})$$

$$\begin{aligned} D^{(m)}(\rho E)|_{G_\alpha} &= \sum_{l=0}^m \binom{m}{l} D^{(m-l)}\rho|_{G_\alpha} \otimes D^{(l)}E|_{G_\alpha} \\ &= \sum_{l=0}^m \binom{m}{l} D^{(m-l)}\rho|_{G_\alpha} \otimes \left( C_v D^{(l)}T|_{G_\alpha} + \frac{1}{2} D^{(m)}|u_\alpha|^2|_{G_\alpha} \right) \end{aligned} \quad (\text{C.13})$$

et :

$$D^{(m)}|u_\alpha|^2|_{G_\alpha} = \sum_{l=0}^m \binom{m}{l} D^{(m-l)}u_\alpha|_{G_\alpha} \otimes D^{(l)}u_\alpha|_{G_\alpha} \quad (\text{C.14})$$

Pour une reconstruction 3-exacte, la relation devient :

$$\bar{P}_\alpha - \tilde{P}_\alpha = -\frac{\gamma-1}{\rho_\alpha} \mathcal{V}_\alpha^{(2)} : \left( D^{(1)}\rho \Big|_{G_\alpha} \otimes \left( D^{(1)}\rho \otimes |u_\alpha|^2 + 2\rho_\alpha D^{(1)}|u_\alpha| \Big|_{G_\alpha} \right) \right) + \mathcal{O}(h^3) \quad (\text{C.15})$$

### C.3 Relation pour la température

De façon similaire :

$$\tilde{T} = \frac{\tilde{P}}{R\bar{\rho}} \quad (\text{C.16})$$

ce qui donne après développement :

$$\bar{T}_\alpha - \tilde{T}_\alpha = \Delta T^{n-2} - \frac{1}{\rho_\alpha} \frac{1}{(n-1)!} \mathcal{V}_\alpha^{(n-1)} : \left( \sum_{l=1}^{n-2} \binom{n-1}{l} D^{(m-1-l)}T \Big|_{G_\alpha} \otimes \left( D^{(l)}\rho \Big|_{G_\alpha} - \frac{1}{2C_v\rho_\alpha} D^{(l)}\rho |u|^2 \Big|_{G_\alpha} \right) \right) + \mathcal{O}(h^n) \quad (\text{C.17})$$

soit :

$$\Delta T^{n-2} = - \sum_{m=2}^{n-2} \frac{1}{m!} \mathcal{V}_\alpha^{(m)} : \left( -\frac{T_\alpha}{\rho_\alpha} D^{(m)}\rho \Big|_{G_\alpha} + (\gamma-1) \frac{\rho_\alpha |u_\alpha|^2}{\rho_\alpha \rho_\alpha} D^{(m)}\rho \Big|_{G_\alpha} + \frac{\gamma-1}{\rho_\alpha R} D^{(m)}P \Big|_{G_\alpha} - D^{(m)}T \Big|_{G_\alpha} \right) \quad (\text{C.18})$$

Ici  $\Delta T^{n-2}$  est nul pour  $n = 3$ , ce qui donne la reconstruction 3-exacte suivante de la Température :

$$\bar{T}_\alpha - \tilde{T}_\alpha = -\frac{1}{\rho_\alpha} \mathcal{V}_\alpha^{(2)} : \left( D_\alpha^{(1)}\rho \Big|_{G_\alpha} \right) \otimes \left( D^{(1)}T \Big|_{G_\alpha} - \frac{1}{2C_v\rho_\alpha} D^{(1)}\rho |u|^2 \Big|_{G_\alpha} \right) + \mathcal{O}(h^n) \quad (\text{C.19})$$

# Alexandre LIMARE

## Doctorat : Matériaux, Mécanique, Optique et Nanotechnologie

Année 2017

### Raffinement adaptatif de maillages intersectants, en Volumes Finis d'ordre élevé, pour l'aéropulsion

Cette thèse concerne le développement d'outils industriels de simulation d'écoulements instationnaires compressibles autour de corps en mouvements relatifs pour des lanceurs spatiaux. Le code FLUSEPA, développé par ArianeGroup, s'appuie sur une formulation Volumes Finis d'ordre élevé et utilise un chevauchement de maillage conservatif par intersections géométriques. Les faces d'intersection autorisent le calcul des flux entre mailles chevauchantes et mailles coupées, permettant le traitement correct des chocs et de l'ensemble des structures instationnaires dans les zones de recouvrement. Au cours de cette thèse, une technique de raffinement de maillage adaptatif (AMR) par création d'arbres « octree » par maille a été implémentée pour des maillages non-structurés hexaédriques. Elle permet de simplifier la création des maillages et de garantir une résolution locale en adéquation avec la physique captée. Le module AMR s'intègre avec cohérence aux schémas de discrétisation spatio-temporels existants afin d'assurer la conservativité et la précision, de plus, il dégrade peu les performances algorithmiques. Ainsi, l'exécution du programme s'effectue sur plusieurs processus avec un équilibrage de charge spécifique au schéma explicite temporel adaptatif et comporte une procédure assurant une projection conservative d'ordre élevé des variables de calcul sur les mailles raffinées. Ces deux propriétés assurent la cohérence avec la stratégie numérique existante. La simulation de plusieurs cas tests montre le potentiel de ce module et permet de valider son implémentation.

Mots clés : méthode des Volumes Finis, calcul adaptatif, modélisation CFD, maillages intersectants.

### Adaptive Refinement of intersecting Meshes, in a High-order Finite Volume Framework, for Aeropropulsion Applications

This thesis is part of an effort to develop numerical industrial tools for the simulation of unsteady compressible flows about bodies in relation motion often encountered in the context of space launchers. FLUSEPA, a code developed by ArianeGroup, relies on a high-order Finite Volume formulation and a conservative overlapping of meshes using geometric intersections. In the overlapping regions, geometric faces allow the calculation of fluxes and the advection of shocks and unsteady structures. This manuscript describes the implementation of a cell-based Adaptive Mesh Refinement (AMR) technique for unstructured meshes composed of hexahedra. This new method eases the mesh construction process and ensures a local resolution adapted to the physical properties captured. In order to be functional, the AMR module must be consistent with the pre-existing spatio-temporal numerical schemes (*i.e.* be conservative and precise) and also keep the algorithmic performance. Thus, the obtained solution is divided between several processes with a load balancing specific to the explicit temporal adaptive numerical scheme was devised and includes a high-order conservative projection of the variables for the refined cells. These two properties compose a consistent global numerical strategy. Several test cases are run using this module and validate its implementation.

Keywords: Finite Volume method, adaptive mesh refinement, computational fluid dynamics, intersecting meshes.

Thèse réalisée en partenariat entre :

