

Methode particulaire multiniveaux pour la dynamique des gaz, application au calcul d'ecoulements multifluides

Lisl Weynans

▶ **To cite this version:**

Lisl Weynans. Methode particulaire multiniveaux pour la dynamique des gaz, application au calcul d'ecoulements multifluides. Mathématiques [math]. Université Joseph-Fourier - Grenoble I, 2006. Français. NNT : . tel-00121346

HAL Id: tel-00121346
https://theses.hal.science/tel-00121346

Submitted on 20 Dec 2006

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.

UNIVERSITE DE GRENOBLE I - JOSEPH FOURIER

THÈSE

pour obtenir le grade de :

DOCTEUR DE L'UNIVERSITÉ JOSEPH FOURIER

Spécialité MATHEMATIQUES APPLIQUEES

présentée par

Lisl Weynans

Méthode particulaire multi-niveaux pour la dynamique des gaz, application au calcul d'écoulements multifluides

---

Directeur de thèse :

Georges-Henri Cottet

Composition du jury :

Rémi	ABGRALL	Rapporteur
Georges-Henri	COTTET	Directeur de thèse
Serge	HUBERSON	Rapporteur
Iraj	MORTAZAVI	Examinateur
Bernard	REBOURCET	Co-directeur de thèse
Rémi	SENTIS	Examinateur
Patrick	WITOMSKI	Examinateur

Jamais je n'ai cherché la gloire
Ni voulu dans la mémoire
des hommes
Laisser mes chansons
Mais j'aime les mondes subtils
Aériens et délicats
Comme des bulles de savon.

J'aime les voir s'envoler,
Se colorer de soleil et de pourpre,
Voler sous le ciel bleu, subitement trembler,
Puis éclater.

A demander ce que tu sais
Tu ne dois pas perdre ton temps
Et à des questions sans réponse
Qui donc pourrait te répondre ?

Chantez en coeur avec moi :
Savoir ? Nous ne savons rien
Venus d'une mer de mystère
Vers une mer inconnue nous allons
Et entre les deux mystères
Règne la grave énigme
Une clef inconnue ferme les trois coffres
Le savant n'enseigne rien, lumière n'éclaire pas
Que disent les mots ?
Et que dit l'eau du rocher ?

Voyageur, le chemin
C'est les traces de tes pas
C'est tout ; voyageur,
il n'y a pas de chemin,
Le chemin se fait en marchant
Le chemin se fait en marchant
Et quand tu regardes en arrière
Tu vois le sentier que jamais
Tu ne dois à nouveau fouler
Voyageur ! Il n'y a pas de chemins
Rien que des sillages sur la mer.

Tout passe et tout demeure
Mais notre affaire est de passer

0

De passer en traçant
Des chemins
Des chemins sur la mer

Antonio Machado

Remerciements

J'aimerais tout d'abord adresser mes remerciements les plus chaleureux à **Georges-Henri Cottet**. Malgré des conditions géographiques un peu particulières, j'ai bénéficié durant mes trois années de thèse de ses très grandes connaissances scientifiques et de sa non moins grande disponibilité. Merci en particulier de m'avoir accueillie aussi souvent à Grenoble.

Je voudrais également exprimer ma plus vive reconnaissance à **Bernard Rebourcet**, mon encadrant CEA, pour sa bienveillance, la confiance qu'il m'a accordée, et surtout pour les discussions si instructives que nous avons eues durant ces trois ans.

Je suis très reconnaissante à **Rémi Abgrall** et **Serge Huberson** d'avoir accepté d'être les rapporteurs de cette thèse, et je les en remercie sincèrement.

J'adresse également de vifs remerciements à **Iraj Mortazavi**, **Rémi Sentis** et **Patrick Witomski** qui me font l'honneur de bien vouloir examiner mon travail.

Je tiens à remercier chaleureusement **Anne Burbeau** pour sa patience à répondre à mes interrogations sur les méthodes numériques utilisées au CEA, ainsi que **Stefan Hickel**, avec qui j'ai eu le plaisir de travailler sur un projet de Simulation des Grandes Echelles au cours du CEMRACS 2005. Merci à **Frédéric Lagoutière** d'avoir répondu à mes nombreuses questions avec tant de gentillesse. Merci également à **Yvon Maday** de m'avoir permis d'occuper de temps en temps une place dans les bureaux du Laboratoire Jacques-Louis Lions.

Durant ces trois annés, j'ai pu apprécier à d'innombrables reprises la gentillesse et la disponibilité de **Michèle Cohadier**, et je voudrais donc l'en remercier très sincèrement.

J'en profite également pour remercier les thésards, stagiaires et post-docs que j'ai cotoyés durant mon séjour sur les moquettes de la zone ouverte du CEA. Merci tout particulièrement à la "Dream Tream", pour nos rêves un peu fous, nos rigolades et nos discussions si agréables : David, Hugues, Constant, Julien, Marie-Christine. Merci de tout coeur à Marjolaine, Solène, Virginie, Marc-Antoine, ainsi qu'aux "petits nouveaux" du couloir : Christophe, Franck et Mathieu, aux gentils stagiaires que nous avons pris plaisir à maltraiter chaque été, et à tous les autres...

De la même manière, je tiens à remercier pour leur accueil les thésards de la salle 3 du LMC, à Grenoble, en particulier Marc, Céline, Morgan, Williams, Jocelyn et Carine. Un grand merci aussi au thésards du bureau 2D01 à Chevaleret : Nicole, Filipa, Stéphane, Jean-François, Jean-Baptiste, Vincent, ainsi qu'à Laurent Boudin, pour tous ses bons conseils, et à Ulrich.

M'éloignant du monde des maths, je profite de ces lignes pour remercier mes amis qui ne sont pas directement liés à l'univers de la thèse, mais dont la présence et l'affection ont embelli ma vie durant ces trois années, et bien avant.

Mes remerciements vont aussi à ma famille : à mes parents, pour tout leur amour et tout ce qu'ils m'ont appris, à mes deux soeurs, Adeline et Lucile, à ma belle-mère Odile. J'adresse une pensée particulière à ma grand-mère Marguerite, avec qui je peux partager tant de choses.

Enfin, et plus que tout, je veux remercier Didier, dont j'ai le grand bonheur de partager la vie. Pour tout ce que nous avons vécu et vivrons ensemble, pour ta confiance, ta patience, ton amour, ton humour, et tout ce qui fait de toi une personne que j'aime terriblement, merci un million de fois.

Table des matières

Remerciements **1**

Table des matières **3**

Introduction **7**

1 Méthode particulaire avec remaillage pour la dynamique des gaz **15**
1.1 Discrétisation particulaire 15
1.1.1 Equations d'Euler et dérivées lagrangiennes 15
1.1.2 La méthode particulaire 17
1.2 Résolution des équations d'Euler 20
1.2.1 Décomposition particules-grilles : "Lagrange + advection" 20
1.2.2 Résolution de la phase Lagrange 22
1.2.3 Résolution de la phase d'advection 24
1.3 Le remaillage des particules 24
1.3.1 Pourquoi remailler ? 24
1.3.2 Formules de remaillage 25
1.3.3 Grilles non cartésiennes uniformes 27
1.4 Conclusion du chapitre 1 30

2 Analyse de la méthode particules-grille avec remaillage **31**
2.1 Conditions de stabilité de la méthode particules-grille pour la dynamique des gaz 31
2.1.1 Stabilité de la résolution de la phase Lagrange 31
2.1.2 Stabilité de la résolution de la phase d'advection 34
2.2 Etude de la méthode pour un problème d'advection modèle 35
2.2.1 Discrétisation particulaire 35
2.2.2 Interprétation en termes de schémas aux différences finies 36
2.2.3 Analyse de la précision suivant le noyau d'interpolation 38
2.2.3.1 Cas d'un noyau à support impair 39
2.2.3.2 Cas d'un noyau à support pair 44
2.2.4 Schémas obtenus avec des noyaux d'interpolation connus 51
2.2.5 Correction de l'évaluation de la vitesse 53
2.2.6 Utiliser des formules décentrées sur les bords ? 56

2.2.7 Correction non-linéaire du schéma avec Λ_2 pour l'équation de Bürgers . . . 58
2.3 Conclusion du chapitre 2 . . . 61

3 Validation numérique **63**
3.1 Description de l'algorithme . . . 63
3.1.1 Définition des grilles . . . 63
3.1.2 Initialisation/ Début d'un nouveau pas de temps . . . 65
3.1.3 Résolution des équations d'Euler . . . 66
3.1.3.1 Détermination du pas de temps . . . 66
3.1.3.2 Résolution de la phase Lagrange . . . 67
3.1.3.3 Résolution de la phase d'advection . . . 67
3.1.3.4 Remaillage . . . 67
3.1.4 Ajout d'un terme de viscosité artificielle . . . 68
3.1.5 Traitement des conditions aux limites . . . 69
3.2 Validation numérique de la méthode . . . 70
3.2.1 Evaluation de l'ordre de convergence . . . 70
3.2.1.1 Advection linéaire uniforme d'un sinus . . . 70
3.2.1.2 Advection linéaire uniforme d'une gaussienne . . . 72
3.2.1.3 Advection linéaire uniforme d'une discontinuité . . . 75
3.2.2 Convergence vers la solution entropique ? . . . 77
3.2.3 Tube à chocs de Sod (1D) . . . 81
3.2.4 Cas-test de Sjogreen (1D) . . . 84
3.2.5 Problème de Noh (2D) . . . 97
3.2.5.1 Résolution avec grille polaire . . . 97
3.2.5.2 Résolution avec grille cartésienne . . . 99
3.3 Conclusion du chapitre 3 . . . 99

4 Méthode particulaire multi-niveaux et remaillage adaptatif **107**
4.1 Raffinement des grilles de remaillage . . . 107
4.1.1 Pourquoi utiliser plusieurs grilles ? . . . 107
4.1.2 Conditions de recouvrement des grilles . . . 108
4.1.3 Adaptation du pas de temps . . . 111
4.1.4 Critères d'adaptation des grilles . . . 112
4.1.5 Frontière entre les représentations associées à chaque grille . . . 114
4.1.6 Conservativité de la technique multi-niveaux . . . 116
4.2 Validation numérique . . . 123
4.2.1 Tube à chocs de Sod . . . 123
4.2.2 Etude numérique de la conservativité . . . 123
4.2.3 Problème de Noh . . . 123
4.2.4 Raffinement en temps : application à l'équation de Bürgers . . . 126
4.3 Conclusion du chapitre 4 . . . 129

5 Simulation de l'interface entre deux fluides 131
5.1 Choix d'une méthode de type level-set . . . 131
5.2 Implantation de la méthode . . . 132
5.2.1 Traitement de la fonction ϕ . . . 133
5.2.2 Evaluation et conservation des masses partielles . . . 134
5.2.3 Utilisation de la technique multi-niveaux . . . 135
5.3 Validation numérique . . . 135
5.3.1 Tube à chocs de Sod bifluide . . . 136
5.3.2 Instabilité de Kelvin - Helmholtz . . . 139
5.3.2.1 Résolution sur une seule grille . . . 139
5.3.2.2 Utilisation d'une grille plus fine pour la résolution de l'interface . . . 140
5.3.3 Interaction choc-bulle . . . 147
5.3.3.1 Résolution sur une seule grille . . . 147
5.3.3.2 Utilisation d'une grille plus fine autour de la bulle, portant toutes les variables . . . 150
5.3.3.3 Utilisation d'une grille plus fine autour de la bulle pour la résolution de l'interface . . . 150
5.4 Conclusion du chapitre 5 . . . 159

Conclusion 163

A Détails des calculs de la section 2.2.6 167
A.1 termes en λ . . . 169
A.2 termes en λ^2 . . . 169
A.3 termes en λ^3 . . . 171
A.4 termes en λ^4 . . . 172
A.5 terme en λ^5 . . . 175
A.6 terme en λ^6 . . . 176
A.7 On somme tout... . . . 177

B Toward implicit subgrid scale modeling by particle methods 179
B.1 Résumé . . . 179
B.2 Abstract . . . 180
B.3 Introduction . . . 181
B.4 The vortex-in-cell method . . . 183
B.5 Interpolation formulas . . . 184
B.6 Analysis in spectral space . . . 185
B.7 Choice and optimization of parameters . . . 187
B.8 Numerical results . . . 191
B.8.1 Two-dimensional Taylor-Green vortex . . . 191
B.8.2 Three-dimensional Taylor-Green vortex . . . 191
B.8.3 Comte-Bellot - Corrsin experiment . . . 192
B.9 Conclusions . . . 193

C Particle methods revisited : a class of high order finite-difference methods **197**
C.1 Introduction 198
C.2 Remeshed particle methods 199
C.3 The linear case : $g(u) = a$ 200
C.4 The non-linear case 201
C.5 Numerical illustrations 202
C.6 Conclusion 202

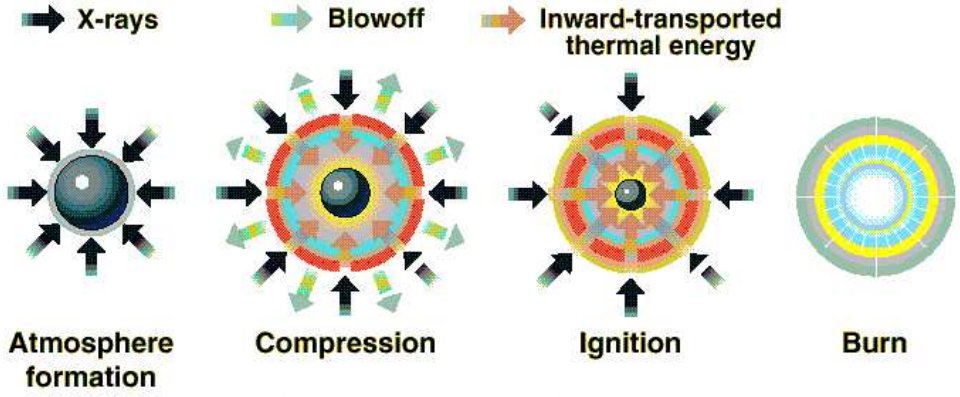
Bibliographie **210**

Table des figures **211**

Introduction

Contexte

La Fusion par Confinement Inertiel (FCI) est une voie prometteuse pour atteindre la fusion thermonucléaire d'un matériau fusible. Elle consiste à "imploser" des cibles (micro-ballons) remplies d'un mélange de deutérium et de tritium. Le succès de l'implosion passe par la maîtrise des instabilités hydrodynamiques qui se développent en régime linéaire puis non-linéaire. En effet, un petit défaut initial de rugosité à l'extérieur de la surface du microballon peut engendrer, à cause de ces instabilités hydrodynamiques, un défaut de sphéricité très important, nuisant gravement à la qualité de l'implosion.



The X-rays rapidly (1) heat the capsule, (2) causing its surface to fly outward. This outward force causes an opposing inward force that compresses the fuel inside the capsule. When the compression reaches the center, temperatures increase to 100,000,000 ℃, (3) igniting the fusion fuel and (4) producing a thermonuclear burn that yields many times the energy input (energy gain).

FIG. 1 –

Simulation numérique des écoulements de la FCI

Le Laser Méga-Joule (LMJ), un des grands projets actuels du CEA, doit permettre de réaliser la Fusion par Confinement Inertiel en laboratoire. Pour le dimensionner, les chercheurs ont recours à la simulation numérique en complément des expériences. Or l'hydrodynamique présente dans le LMJ est un phénomène complexe, régi par un système hyperbolique de lois de conservation fortement non-linéaires : les équations d'Euler compressibles. Les solutions de ce système, même pour des données initiales très régulières, peuvent présenter des discontinuités qui interagissent entre elles au cours du temps. Pour simuler correctement ce type d'écoulements, une méthode numérique doit à la fois être robuste, et résoudre précisément les discontinuités. Il est également préférable, même si ce n'est pas toujours le cas, que cette méthode conserve les quantités conservatives : la masse, la quantité de mouvement et l'énergie totale. Dans le cas plus particulier de la simulation des écoulements multifluides, qui permet d'étudier les instabilités hydrodynamiques, les méthodes numériques doivent, plus encore, ne pas avoir une dissipation numérique trop forte. En effet les phénomènes qu'on envisage de simuler sont très sensibles à la variation des taux de présence de chaque fluide.

Il est couramment admis que les schémas eulériens classiques, c'est-à-dire qui utilisent des éléments de discrétisation dont la position ne varie pas, ne sont pas assez précis pour simuler correctement le type de problèmes rencontrés en FCI. Pour obtenir des solutions numériques suffisamment précises, il faut utiliser des schémas basés sur une formulation lagrangienne, c'est-à-dire, dont les éléments de discrétisation, le plus souvent des volumes finis, se déplacent à la vitesse de l'écoulement. L'intérêt de cette formulation lagrangienne est d'éviter le calcul des termes convectifs, source de dissipation numérique. Par contre, les schémas lagrangiens souffrent souvent de problèmes de stabilité, dus en partie à la déformation des maillages causée par le déplacement lagrangien des éléments de discrétisation. Pour lutter contre ce problème, il est possible de projeter la solution numérique sur un nouveau maillage moins déformé. Cette procédure, appelée remaillage, apporte cependant une erreur numérique qui entâche la précision des calculs. La méthode que nous allons présenter comporte elle aussi une étape appelée remaillage, mais ce remaillage est complètement différent de celui utilisé pour les schémas lagrangiens classiques, puisqu'il s'agit d'interpoler des valeurs ponctuelles sur une grille uniforme.

Méthodes numériques classiques pour la FCI

Deux grandes familles de méthodes sont classiquement utilisées pour simuler les écoulements de l'hydrodynamique en présence de chocs : les schémas de type VNR (Von Neumann-Richmyer), et les schémas de type Godounov. Toutes deux sont des méthodes de volumes finis.

Le schéma VNR, et son extension 2D le schéma de Wilkins, sont formulés en énergie interne. Ils ne conservent ni l'énergie totale ni la quantité de mouvement. Ils utilisent une représentation étagée (aux noeuds et aux mailles) des inconnues. Ils doivent être utilisés avec un terme de viscosité artificielle pour tenir compte du dépôt d'entropie au niveau des chocs, faute de quoi ils y engendrent des oscillations. Ce type de schéma est difficile à étudier théoriquement mais facile à mettre en oeuvre et facile à adapter pour le couplage à la dynamique des gaz d'autres phénomènes physiques. Les chercheurs disposent d'une longue expérience de ces schémas et considèrent qu'ils fournissent des résultats numériques fiables.

Les schémas de type Godounov sont formulés en énergie totale, et utilisent une représentation centrée des inconnues. Ils sont basés sur la résolution exacte ou approchée de problèmes de Riemann pour calculer les flux à travers les faces des volumes de discrétisation, et ne nécessitent pas l'emploi d'un terme de viscosité artificielle. Ce type de méthode n'est pas très apte à simuler des écoulements fortement bidimensionnels, ni à traiter le couplage des équations de l'hydrodynamique à une physique plus complexe, et n'est donc pas considéré comme aussi fiable que les schémas de type VNR.

Pour en savoir plus sur ces différents schémas on renvoie à [4], [54] et [77].

En géométrie 2D, les méthodes de type Wilkins parviennent bien à restituer les phénomènes de la FCI, à condition d'avoir une gestion appropriée du maillage et des remaillages. Cependant, l'extension aux géométries 3D de ces méthodes fait apparaître des difficultés, liées notamment au problème du pavage de la sphère avec des quadrangles réguliers.

Méthodes particulaires

Les méthodes particulaires constituent une autre famille de méthodes lagrangiennes, et utilisent une discrétisation du fluide différente de celles des volumes finis. Elles considèrent non pas des volumes, constitués de noeuds reliés par des arêtes, mais des points sur lesquels sont concentrées des masses. Elles ne souffrent donc pas des problèmes de gestion de maillage des méthodes de volumes finis, ce qui leur permet d'être stables bien que lagrangiennes.

Les méthodes particulaires développées dans le passé pour simuler la dynamique des gaz (Particle-In-Cell, ou PIC, [32], [38], et Smoothed Particle Hydrodynamics, ou SPH, [63], [35] et [5]) ne sont pas assez précises pour pouvoir être utilisées dans les simulations de la FCI. Par contre, dans le cadre de la mécanique des fluides incompressibles, d'autres méthodes particulaires ont été développées (méthodes Vortex-In-Cell, ou VIC, [17], [45]), qui sont réputées être à la fois stables (comme toutes les méthodes particulaires) et précises. Ces méthodes utilisent un remaillage régulier à l'aide de fonctions d'interpolation de degré élevé. Notons que c'est le fait de discrétiser les particules

sur des points et non plus des volumes qui permet d'utiliser aussi facilement ce type de fonction d'interpolation. Le remaillage avec ces fonctions d'interpolation est conservatif et permet de maintenir une distribution régulière des particules, ce qui est important pour la précision des calculs. De nombreuses simulations numériques directes d'écoulements incompressibles ont montré que les interpolations successives détérioreraient peu la qualité de la solution numérique. Ces méthodes particulaires ont permis d'obtenir un certain nombre de résultats de référence, notamment dans le calcul de sillages bi- ou tridimensionnels ([72], [73], [71]). Cependant, à notre connaissaince, elles n'ont jusqu'à présent pas été utilisées pour simuler les écoulements de la dynamique des gaz.

Objectif de la thèse

L'objectif de la thèse est d'évaluer la capacité d'une méthode particulaire inspirée des méthodes Vortex-In-Cell, à simuler les écoulements de la dynamique des gaz, et plus particulièrement les écoulements multifluides. Il s'agit d'étudier systématiquement cette méthode, de manière numérique mais aussi théorique. Cette étude doit permettre de positionner la méthode particulaire par rapport aux méthodes classiquement utilisées pour la dynamique des gaz, afin de déterminer s'il est envisageable, à plus long terme, de l'utiliser à simuler les phénomènes de la Fusion par Confinement Inertiel.
Dans un premier temps nous développons donc une méthode particulaire - ou plutôt particules-grille, car une partie des calculs se fait sur une grille fixe - avec remaillage, pour la simulation d'écoulements compressibles non-visqueux. Nous analysons théoriquement et testons numériquement cette méthode. Nous mettons notamment en évidence des liens forts entre la méthode particules-grilles et des schémas aux différences finies d'ordre élevé, de type Lax-Wendroff, et nous proposons un nouveau schéma d'advection des particules, simple et précis. Puis nous implantons une technique multi-niveaux qui permet d'ajuster la précision des calculs aux caractéristiques de l'écoulement. Enfin, nous discrétisons une technique de type level-set sur les particules afin de simuler l'interface entre fluides compressibles, et utilisons la technique multi-niveaux pour résoudre plus précisément cette interface.

Résumé des chapitres

Dans le **chapitre 1** nous présentons le principe de la discrétisation particulaire, appliqué aux équations d'Euler compressibles. Le fluide est découpé en particules, possédant une masse, une vitesse, une énergie totale, qui se déplacent de manière lagrangienne, à la vitesse de l'écoulement. Nous résolvons les équations d'Euler en pratiquant une décomposition "Lagrange + advection", c'est-à-dire un splitting entre la prise en compte des efforts de pression et celle du déplacement de la matière. Cette décomposition permet de traiter l'advection de manière lagrangienne, et donne des conditions de stabilité moins strictes que si les deux étapes étaient résolues en même temps. On

maintient une distribution régulière de particules en interpolant de manière conservative les particules sur les centres des cellules d'une grille sous-jacente, afin d'y créer de nouvelles particules.

Cette méthode permet aux variables conservatives des équations d'Euler de bénéficier de la précision du transport lagrangien et d'un remaillage conservatif effectué avec des formules d'interpolation d'ordre élevé. Elle évite d'avoir à se préoccuper, comme c'est le cas avec les méthodes de volumes finis, de l'évolution des volumes de discrétisation, en particulier du problème de croisement des mailles.

Le **chapitre 2** concerne l'analyse de la méthode particulaire. Nous commençons par analyser la stabilité de la méthode présentée dans le chapitre 1. Ensuite, pour mieux comprendre les effets du déplacement des particules combiné au remaillage, nous analysons la méthode particulaire sur une équation d'advection modèle, monodimensionnelle et scalaire :

$$u_t + (g(u)u)_x = 0$$

qui décrit l'évolution de la quantité u advectée à la vitesse $g(u)$. Cette étude nous permet d'écrire les schémas particulaires sous forme de schémas aux différences finies et d'analyser leur précision en fonction du noyau d'interpolation utilisé. Nous proposons ensuite une correction de l'évaluation de la vitesse des particules, afin d'obtenir des schémas d'ordre 2. Nous étudions aussi, dans le cas d'un noyau d'interpolation particulier, et pour l'équation de Bürgers, l'ajout d'un terme de viscosité artificielle.

Dans le **chapitre 3**, nous présentons l'algorithme avec lequel sont résolues les équations d'Euler. Nous vérifions sur une équation d'advection uniforme que l'ordre de convergence numérique est le même que celui prédit théoriquement dans le chapitre 2. Nous mettons également en évidence une propriété intéressante des schémas particulaires : ceux-ci semblent converger vers la solution entropique. Nous testons ensuite la méthode sur trois problèmes typiques de la dynamique des gaz : le tube à chocs de Sod, la détente de Sjögreen et le problème de Noh.

Pour augmenter l'adaptivité de la méthode, nous implantons dans le **chapitre 4** une technique multi-niveaux adaptée aux méthodes particulaires avec remaillage. Nous rappelons les conditions de superposition des grilles nécessaires à un traitement consistant des conditions aux limites de chaque grille, et nous proposons des critères d'adaptation des grilles basés sur les valeurs des dérivées partielles spatiales. Nous intéressant à la conservativité de cette technique multi-niveaux, nous proposons une définition de la masse totale sur le domaine de calcul. Nous montrons que la masse totale ainsi définie est totalement conservée, sauf autour de la frontière entre les grilles, où elle est conservée à l'ordre 2. Puis nous testons numériquement le critère d'évolution des grilles précédemment défini et la conservation de la masse totale.

Dans le **chapitre 5** nous nous intéressons à la simulation d'écoulements bi-fluides.

Nous avons choisi d'adapter une méthode de type level-set à notre méthode particulaire. La fonction level-set ϕ, qui contient l'information sur l'interface entre les fluides, est discrétisée sur les particules, advectée et interpolée comme les autres variables. Nous testons cette méthode de simulation d'interfaces sur différents problèmes : tube à chocs de Sod bi-fluide, instabilité de Kelvin-Helmholtz, et l'interaction choc-bulle, qui nous montrent que l'information sur l'interface, portée par ϕ, est correctement transportée par les particules et préservée par l'interpolation.

Nous appliquons la technique multi-niveaux du chapitre 4 afin de résoudre l'interface avec une meilleure précision, et d'améliorer la conservation des masses partielles. Les particules associées à la grille plus fine qui suit l'interface servent soit uniquement à résoudre l'interface et à calculer les masses partielles, soit également à résoudre les équations d'Euler. La figure 2 montre que l'utilisation de particules plus fines pour la fonction level-set ϕ fait converger l'interface.

Enfin, nous présentons en **annexe** le résultat d'un travail réalisé dans le cadre du CEMRACS 2005 avec Stefan Hickel et Nikolaus Adams de l'Université Technique de Munich. Le projet sur lequel nous avons travaillé consistait à développer une méthode Vortex-In-Cell pour la simulation implicite des grandes échelles. Nous avons dans ce but cherché à ajuster la dissipation spectrale de la méthode VIC à des valeurs théoriques de diffusivité turbulente, en jouant sur des combinaisons linéaires de noyaux d'interpolation.

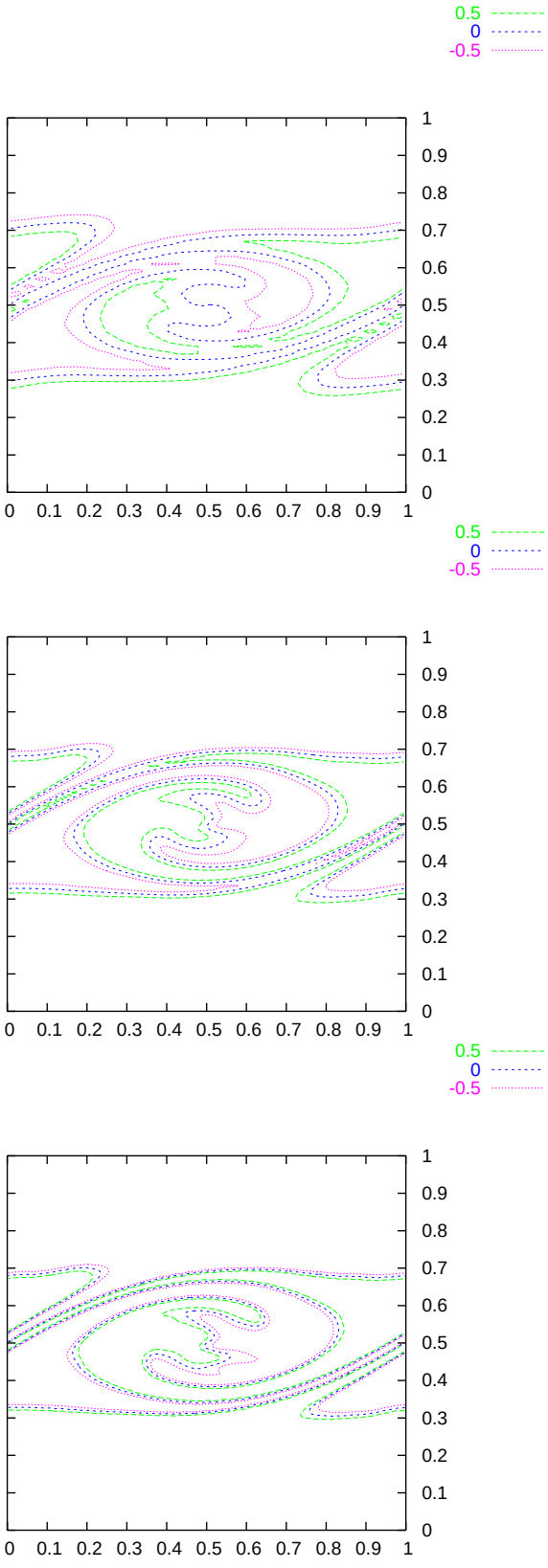


FIG. 2 – Instabilité de Kelvin-Helmholtz, isovaleurs de ϕ, à $t = 8$, équations d'Euler résolues avec 50^2 particules, de haut en bas ϕ résolu sur une grille 50^2, 100^2 et 200^2.

Chapitre 1

Méthode particulaire avec remaillage pour la dynamique des gaz

Dans ce premier chapitre, on présente la méthode particulaire qui a été utilisée dans ce travail. On introduit la discrétisation particulaire dans la section 1.1, puis on décrit dans la section 1.2 la manière dont sont résolues les équations d'Euler discrétisées sur les particules. En particulier, on rappelle le principe de la décomposition "Lagrange + advection". On présente enfin dans la section 1.3 la technique de remaillage utilisée pour interpoler les particules à la fin de l'étape d'advection.

1.1 Discrétisation particulaire

Dans cette section est décrite la manière dont on discrétise avec des particules les variables du système des équations d'Euler compressibles.

1.1.1 Equations d'Euler et dérivées lagrangiennes

Dans tout ce qui suit, on adopte les notations suivantes :

- ρ désigne la densité
- $\vec{U} = {}^t(u, v, w)$ désigne la vitesse
- P désigne la pression
- E désigne l'énergie totale
- e désigne l'énergie interne
- γ désigne la constante adiabatique du gaz

Le système des équations d'Euler pour la dynamique des gaz s'écrit :

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \nabla.(\rho\vec{U}) &= 0 \\ \frac{\partial \rho\vec{U}}{\partial t} + \nabla.(\rho(\vec{U}\otimes\vec{U} + pId) &= 0 \\ \frac{\partial \rho E}{\partial t} + \nabla.((\rho E + p.)\vec{U}) &= 0 \end{aligned}$$

Le système doit être complété par une équation d'état reliant la pression, la densité et l'énergie interne. Dans ce mémoire, cette équation d'état sera toujours celle des gaz parfaits :

$$p = (\gamma - 1)\rho e \tag{1.1}$$

Il existe deux manières de considérer l'évolution d'un écoulement fluide :

- La première consiste à se placer en un point fixe pour y observer comment le fluide évolue. C'est la description eulérienne du fluide. Les équations d'Euler présentées ci-dessus sont écrites dans ce formalisme eulérien. Les coordonnées spatiales utilisées dans cette description de l'écoulement sont les coordonnées spatiales habituelles.
- La deuxième manière consiste à suivre un point matériel de fluide au cours de son déplacement. Puisque sa position varie au cours du temps, on identifie ce point matériel par sa position initiale, dont les coordonnées sont appelées coordonnées lagrangiennes.

Il existe une relation entre coordonnées eulériennes et lagrangiennes : si on note $\vec{x}$ le vecteur des coordonnées eulériennes et $\vec{\xi}$ celui des coordonnées lagrangiennes, alors $\vec{x}$ est solution de l'équation différentielle suivante

$$\frac{d\vec{x}}{dt} = \vec{U}$$

avec pour condition initiale :

$$\vec{x}(0) = \vec{\xi}$$

Une variable A peut être exprimée de deux manières : en fonction des variables $(\vec{x}, t)$, ou des variables $(\vec{\xi}, t)$.

$$A = A(\vec{x}, t) = \overline{A}(\vec{\xi}, t)$$

La dérivée partielle de $\overline{A}$ par rapport au temps s'écrit :

$$\frac{\partial \overline{A}}{\partial t} = \frac{\partial A}{\partial \vec{x}}\frac{\partial \vec{x}}{\partial t} + \frac{\partial A}{\partial t}$$

Comme on a $\frac{\partial \vec{x}}{\partial t} = \vec{U}$, on peut écrire :

$$\frac{\partial \overline{A}}{\partial t} = \frac{\partial A}{\partial t} + \vec{U}.\frac{\partial A}{\partial \vec{x}}$$

$\frac{\partial \overline{A}}{\partial t}$ s'appelle la dérivée lagrangienne de la quantité A.
La plupart du temps, pour simplifier les notations, on note cette dérivée $\frac{d.}{dt}$:

$$\frac{dA}{dt} = \frac{\partial \overline{A}}{\partial t} = \frac{\partial A}{\partial t} + \vec{U}\frac{\partial A}{\partial \vec{x}}$$

Si on exprime les équations d'Euler en remplaçant la dérivée temporelle habituelle par la dérivée lagrangienne, on obtient le système suivant, qui exprime les dérivées temporelles en formalisme lagrangien et les dérivées spatiales en formalisme eulérien :

$$\frac{d\,\rho}{dt} = -\rho\nabla.\vec{U} \tag{1.2}$$

$$\rho\frac{d\vec{U}}{dt} = -\vec{\nabla}p \tag{1.3}$$

$$\rho\frac{dE}{dt} = -\nabla.(p\vec{U}) \tag{1.4}$$

Cette formulation a l'avantage de faire disparaître les termes de convection, désormais contenus dans la dérivée lagrangienne. L'équation pour l'énergie totale (1.4) peut être remplacée par une équation exprimant l'évolution de l'énergie interne ($e = E - \frac{||\vec{U}||^2}{2}$) :

$$\rho\frac{de}{dt} = -p\nabla.\vec{U} \tag{1.5}$$

Pour résoudre le déplacement de la matière, on ajoute l'équation :

$$\rho\frac{dx}{dt} = \vec{U} \tag{1.6}$$

1.1.2 La méthode particulaire

Les méthodes particulaires classiques pour simuler les fluides compressibles non-visqueux, telles que la méthode PIC (Particle In Cell) ou la méthode SPH (Smoothed Particle Hydrodynamics), ont pour point commun de représenter le fluide par des particules de masse constante, se déplaçant de manière lagrangienne.

La méthode PIC ([32], [38]) utilise simultanément une grille et des particules pour résoudre les équations d'Euler. Elle opère un splitting entre la résolution des efforts

de pression et celle du déplacement de la matière. Les centres des cellules de la grille portent des valeurs de densité, vitesse et énergie interne, qui permettent de résoudre (1.3) et (1.5) en utilisant des schémas aux différences finies. Chaque cellule de la grille contient un nombre variable de particules. Chaque particule q a une position x_q, une masse m_q (qui reste constante au cours du temps) et porte des valeurs de vitesse $\vec{U}_q$ et d'énergie interne e_q qui sont identiques à celles de la cellule à laquelle elles appartiennent. Une fois que de nouvelles valeurs de vitesse et d'énergie interne ont été calculées dans chaque cellule, les particules sont déplacées de manière lagrangienne en résolvant (1.6). Les valeurs de densité, vitesse et énergie interne portées par les centres des cellules sont actualisées pour tenir compte de ce déplacement : soit par interpolation des valeurs des particules, soit en considérant pour chaque cellule toutes les particules entrées ou sorties et en modifiant en conséquence la valeur de la cellule.

Dans la méthode SPH ([63], [35] et [5]), on attribue également à chaque particule q une position x_q, une masse m_q, une vitesse $\vec{U}_q$ et une énergie interne e_q. La représentation globale de l'écoulement s'obtient par une interpolation des quantités portées par les particules. Par exemple, si on note W le noyau d'interpolation utilisé, la densité d'une particule peut être calculée de la manière suivante :

$$\rho_q = \sum_i m_i W(|x_q - x_i|, h)$$

h est une longueur caractéristique de la distance entre les particules et dépend de la largeur du support de W.
On obtient la formulation SPH des équations d'Euler en multipliant les équations (1.3) et (1.5) par le noyau d'interpolation et en les intégrant sur un volume mobile, qui représente la particule. Comme la masse de chaque particule est constante, il n'est pas nécessaire de résoudre (1.2). En notant p_i la pression de la particule i on obtient :

$$\frac{d\vec{U}_q}{dt} = -\sum_i m_i (\frac{p_i}{\rho_i^2} + \frac{p_q}{\rho_q^2}) \vec{\nabla}_i W(|x_q - x_i|, h)$$

$$\frac{de_q}{dt} = \frac{p_q}{\rho_q^2} \sum_i m_i (\vec{U}_i - \vec{U}_q) \vec{\nabla}_i W(|x_q - x_i|, h)$$

La position de la particule varie suivant :

$$\frac{dx_q}{dt} = \vec{U}_q$$

Avec la méthode SPH, l'évolution du fluide décrit par les équations d'Euler se calcule en résolvant numériquement ces trois équations, de manière simultanée.

De manière générale, ces deux méthodes particulaires partagent une même aptitude à simuler des écoulements présentant de très larges déformations, et ceci grâce au caractère lagrangien des particules. Pour la même raison, elles sont également capables

de calculer précisément l'évolution d'interfaces. La méthode SPH offre par ailleurs la possibilité de simuler facilement des problèmes de frontière libre, puisque les particules suivent naturellement les mouvements de la matière. La méthode PIC ne bénéficie pas de cet avantage car elle utilise une grille en addition des particules. Mais en revanche elle est généralement considérée comme très stable, sauf sur certains écoulements lents. Les deux méthodes sont cependant réputées manquer de précision, pour des raisons différentes : la méthode SPH, parce que le calcul des dérivées avec les formules d'interpolation introduit de la dispersion, et la méthode PIC, parce que les noyaux d'interpolation (d'ordre peu élevé) qui transmettent l'information entre les particules et la grille peuvent être très dissipatifs.

La méthode particulaire que nous allons présenter combine certaines caractéristiques des deux méthodes que nous venons d'évoquer, dans le but de pallier leurs principaux défauts respectifs. Comme on le détaillera dans la suite, elle allie la stabilité de la résolution des équations d'Euler sur une grille (et non pas avec des dérivations de formules d'interpolation) à la précision de l'interpolation entre grille et particules grâce à l'emploi de formules d'ordre élevé.

Cette méthode résoud les équations d'Euler en utilisant l'énergie totale (1.4), et non pas, comme les méthodes SPH et PIC, celle de l'énergie interne (1.5). On attribue à chaque particule q une masse m_q, une vitesse $\vec{U_q}$, une énergie totale E_q, un volume V_q et une position x_q. Comme la particule se déplace de manière lagrangienne, sa masse est conservée au cours du temps. Il s'agit donc de résoudre pour chaque particule q les équations :

$$\begin{aligned}
\frac{dm_q}{dt} &= 0 \\
\frac{d\vec{U_q}}{dt} &= -\frac{1}{\rho_q}(\nabla p)_{x=x_q} \\
\frac{dE_q}{dt} &= -\frac{1}{\rho_q}(\nabla.(p\vec{U}))_{x=x_q} \\
\frac{dx_q}{dt} &= \vec{U_q}
\end{aligned}$$

Comme on le verra dans la section 1.1, nous interpolons à chaque pas de temps les quantités conservatives m_q, $m_q\vec{U_q}$ et m_qE_q. En pratique, on résoud donc plutôt le système équivalent :

$$\frac{dm_q}{dt} = 0 \tag{1.7}$$

$$\frac{dm_q\vec{U_q}}{dt} = -V_q(\nabla q)_{x=x_q} \tag{1.8}$$

$$\frac{dm_qE_q}{dt} = -V_q(\nabla.(p\vec{U}))_{x=x_q} \tag{1.9}$$

$$\frac{dx_q}{dt} = \vec{U_q} \tag{1.10}$$

Les variables véritablement utilisées sont finalement : V_q le volume, m_q la masse, $(m\vec{U})_q = m_q\vec{U}_q$ la quantité de mouvement, $(mE)_q = m_qE_q$ l'énergie totale et x_q la position de la particule q.

Remarque: *On peut également introduire la discrétisation particulaire de notre méthode en intégrant les équations d'Euler sur des volumes. On se rend alors compte que la manière de discrétiser les particules est proche de celle utilisée dans les méthodes de volumes finis pour les équations d'Euler. Mais, dans notre cas nous ne cherchons pas à suivre le contour du volume d'une particule, mais seulement sa position, à la différence des volumes finis.*

A chaque pas de temps, on résoud les équations (1.8) et (1.9) avec une méthode de différences finies et on déplace les particules selon (1.10). Pour le transport des particules, une étape essentielle de notre méthode est le remaillage : à chaque pas de temps les valeurs portées par les particules sont interpolées sur une grille régulière où sont réinitialisées de nouvelles particules. Notre méthode particulaire est donc en fait une méthode particules-grille.

Pour plus de détails sur le formalisme lagrangien, voir par exemple le livre de Godlewsky et Raviart [36], ou le livre de Bowers et Wilson [9]. Pour la discrétisation particulaire, on renvoie au livre de Cottet et Koumoutsakos [17].

1.2 Résolution des équations d'Euler

La méthode particulaire que nous présentons est une méthode dite eulérienne puisqu'à la fin de chaque itération temporelle nous obtenons la solution numérique sur la même grille fixe. Elle réalise ce qu'on appelle une décomposition "Lagrange + advection". Dans cette section, on présente d'abord l'idée générale de cette décomposition, puis les détails des calculs dans le cas de la méthode particules-grille.

1.2.1 Décomposition particules-grilles : "Lagrange + advection"

La décomposition "Lagrange + advection", couramment pratiquée dans le cadre de la résolution des équations d'Euler par des méthodes eulériennes ([75], [25]), consiste à résoudre les équations en deux étapes :

- une étape dite lagrangienne (appelée couramment "phase Lagrange"), où sont résolus les efforts de pression,
- une étape d'advection, où est résolu le déplacement de la matière.

Etape Lagrangienne Cette étape correspond à la résolution des équations d'Euler avec une méthode purement lagrangienne, c'est pourquoi elle porte ce nom. On résoud ici uniquement les efforts de pression exercés sur le fluide, à partir des valeurs portées par la grille (c'est-à-dire, portées par les particules initalisées sur la grille). Dans notre

cas, cela revient à résoudre pour chaque particule i, placée sur une cellule de la grille, les équations :

$$\frac{\partial m_i}{dt} = 0 \tag{1.11}$$

$$\frac{\partial m_i \vec{U_i}}{\partial t} = -V_i(\nabla p)_{x=x_i} \tag{1.12}$$

$$\frac{\partial m_i E_i}{\partial t} = -V_i(\nabla.(p\vec{U}))_{x=x_i} \tag{1.13}$$

ce que nous faisons par une méthode aux différences finies.

Etape d'advection On prend ici uniquement en compte le déplacement de la matière. Dans notre cas, cette étape consiste à déplacer les particules, donc à résoudre l'équation :

$$\frac{dx_i}{dt} = \vec{U_i} \tag{1.14}$$

puis à les remailler sur la grille. Les équations :

$$\frac{dm_i}{dt} = 0 \tag{1.15}$$

$$\frac{dm_i \vec{U_i}}{dt} = 0 \tag{1.16}$$

$$\frac{dm_i E_i}{dt} = 0 \tag{1.17}$$

dans lesquelles les termes de convection sont implicitement contenus se trouvent résolues par le déplacement des particules et leur interpolation. A l'issue du remaillage, on dispose à nouveau de quantités réparties sur une grille, ce qui nous permet de passer à la résolution de l'étape Lagrange suivante.

Conditions de stabilité Les schémas de volumes finis eulériens qui résolvent les équations d'Euler sans utiliser cette décomposition (schémas eulériens "directs") sont soumis à une unique contrainte de stabilité du type :

$$(|u| + c)dt \leq Cdx$$

avec c la vitesse du son, u la vitesse de l'écoulement et C une constante dépendant du schéma utilisé. La résolution en deux étapes des équations nécessite quant à elle de vérifier une condition de stabilité pour chaque étape :

- La première, pour l'étape lagrangienne, est du type : $c\ dt \leq C_1\ dx$ car l'étape lagrangienne ne prend pas en compte le déplacement du fluide.
- La seconde, pour l'étape d'advection, porte uniquement sur la vitesse matérielle de l'écoulement ; elle est donc en général du type : $|u|\ dt \leq C_2\ dx$.

Chacune de ces deux conditions de stabilité est moins stricte que celle d'un schéma eulérien direct. Le traitement en deux étapes diminue donc les contraintes de stabilité. Nous détaillerons quelles sont dans notre cas ces contraintes de stabilité dans le chapitre suivant, consacré à l'analyse de la méthode particules-grille. La décomposition "Lagrange + advection" permet aussi de résoudre chaque étape indépendamment, d'une manière qui lui est adaptée. C'est ce que nous faisons quand nous résolvons la phase Lagrange par des différences finies et la phase d'advection avec des noyaux d'interpolation.

Cette décomposition présente donc deux avantages, issus de la formulation Lagrangienne :

- Elle diminue les contraintes de stabilité.
- Elle permet d'utiliser des traitement numériques complètement différents pour les deux étapes.

1.2.2 Résolution de la phase Lagrange

Il s'agit de résoudre numériquement les équations (1.7) à (1.9). Au début de chaque itération temporelle les particules sont placées sur une grille. Comme au cours de l'étape lagrangienne on ne tient pas compte du déplacement des particules, la résolution des équations s'effectue entièrement sur cette grille. Nous avons utilisé deux techniques différentes pour les résoudre : des différences finies centrées d'ordre 2, et le solveur approché de Riemann construit sur l'approximation "double choc" ([28]). On note $\tilde{m}_i$, $(\tilde{mU})_i$, $(\tilde{mE})_i$ les nouvelles quantités résultant de la résolution de cette étape lagrangienne, c'est-à-dire de la résolution des efforts de pression.

Si on utilise des différences finies centrées d'ordre 2, le schéma correspondant à la résolution de cette étape est, en dimension un, pour chaque particule :

$$\begin{aligned}
\tilde{m}_i &= m_i \\
(\tilde{mU})_i &= (mU)_i - dt\ V_i\ \frac{p_{i+1} - p_{i-1}}{2dx} \\
(\tilde{mE})_i &= (mE)_i - dt\ V_i\ \frac{p_{i+1}U_{i+1} - p_{i-1}U_{i-1}}{2dx} \\
p_i &= (\gamma - 1)\frac{m_i}{V_i}\left(\frac{(mE)_i}{m_i} - \frac{(mU)_i^2}{2m_i^2}\right)
\end{aligned}$$

où p_i représente la pression de la particule i et U_i sa vitesse. Comme au cours de cette étape on ne tient pas compte du déplacement de la matière, le volume V_i de chaque particule reste constant.

En dimension deux, on note $\vec{U}_{i,j} = (u_{i,j}, v_{i,j})$ le vecteur vitesse et $p_{i,j}$ la pression de la

particule i, j. Le schéma devient :

$$\begin{aligned}
\tilde{m}_{i,j} &= m_{i,j} \\
(\tilde{mu})_{i,j} &= (mu)_{i,j} - dt\ V_{i,j}\ \frac{p_{i+1,j} - p_{i-1,j}}{2dx} \\
(\tilde{mv})_{i,j} &= (mv)_{i,j} - dt\ V_{i,j}\ \frac{p_{i,j+1} - p_{i,j-1}}{2dy} \\
(\tilde{mE})_{i,j} &= (mE)_{i,j} - dt\ V_{i,j}\ [\ \frac{p_{i+1,j}u_{i+1,j} - p_{i-1,j}u_{i-1,j}}{2dx} + \frac{p_{i,j+1}v_{i,j+1} - p_{i,j-1}v_{i,j-1}}{2dy}\ ] \\
p_{i,j} &= (\gamma - 1)\frac{m_{i,j}}{V_{i,j}}(\frac{(mE)_{i,j}}{m_{i,j}} - \frac{(mu)^2_{i,j} + mv)^2_{i,j}}{2m^2_{i,j}})
\end{aligned}$$

Avec ces différences finies centrées, pour avoir un schéma stable, il faut utiliser pour l'intégration en temps un schéma de Runge-Kutta d'ordre supérieur ou égal à 3. Nous avons testé un schéma de Runge-Kutta d'ordre 3 et un autre d'ordre 4. Sur les tests numériques présentés dans le chapitre 3, c'est-à-dire dans le cas monofluide, nous n'avons constaté pratiquement aucune différence entre les deux. Par contre, dans le cas bifluide, présenté dans le chapitre 5, en particulier pour l'instabilité de Kelvin-Helmholtz, les résultats étaient notablement meilleurs avec le schéma Runge-Kutta d'ordre 4. Par conséquent, dans le but d'homogénéiser l'implantation de la méthode, nous ne présentons dans ce mémoire que des résultats obtenus avec le schéma d'ordre 4.

La résolution avec le solveur de Riemann approché est par contre d'ordre 1 en temps. Le schéma en dimension un est alors :

$$\begin{aligned}
\tilde{m}_i &= m_i \\
(\tilde{mU})_i &= (mU)_i - dt\ V_i\ \frac{p^*_{i+\frac{1}{2}} - p^*_{i-\frac{1}{2}}}{2dx} \\
(\tilde{mE})_i &= (mE)_i - dt\ V_i\ \frac{p^*_{i+\frac{1}{2}}U^*_{i+\frac{1}{2}} - p^*_{i-\frac{1}{2}}U^*_{i-\frac{1}{2}}}{2dx}
\end{aligned}$$

où $p^*_{i+\frac{1}{2}}$ et $U^*_{i+\frac{1}{2}}$ sont les pression et vitesse calculées avec le solveur à l'interface entre les cellules i et $i-1$.

En dimension deux, le schéma est :

$$\begin{aligned}
\tilde{m}_{i,j} &= m_{i,j} \\
(\tilde{mu})_{i,j} &= (mu)_{i,j} - dt\ V_{i,j}\ \frac{p^*_{i+1,j} - p^*_{i-1,j}}{2dx} \\
(\tilde{mv})_{i,j} &= (mv)_{i,j} - dt\ V_{i,j}\ \frac{p^*_{i,j+1} - p^*_{i,j-1}}{2dy} \\
(\tilde{mE})_{i,j} &= (mE)_{i,j} - dt\ V_{i,j}\ [\ \frac{p*_{i+1,j}\ u*_{i+1,j} - p*_{i-1,j}\ u*_{i-1,j}}{2dx} \\
&\quad + \frac{p*_{i,j+1}\ v*_{i,j+1} - p*_{i,j-1}\ v*_{i,j-1}}{2dy}\ ]
\end{aligned}$$

1.2.3 Résolution de la phase d'advection

Cette étape consiste à déplacer les particules à la vitesse de l'écoulement, puis à les remailler sur la grille sous-jacente. Déplacer la particule i revient à modifier la valeur de x_i. On choisit de déplacer la particule à la vitesse évaluée après le traitement de l'étape lagrangienne.

$$\tilde{x_i} = x_i + dt\frac{(\tilde{mU})_i}{\tilde{m}_i}$$

On verra dans la section 2.2.4 du chapitre 2 que, dans le cas des équations d'Euler, le calcul du déplacement des particules avec un schéma d'Euler explicite permet de résoudre l'étape d'advection à l'ordre 2 (à condition que le remaillage s'effectue avec un noyau d'interpolation qui préserve les trois premiers moments, ce qui sera toujours le cas pour nous). La phase d'advection est donc résolue à l'ordre 2.

1.3 Le remaillage des particules

1.3.1 Pourquoi remailler ?

Les particules sont issues d'une formulation lagrangienne, elles sont portées par l'écoulement. Elles s'accumulent donc naturellement près des zones de fort gradient et se raréfient dans le reste du domaine. Si on les laisse évoluer sans intervenir, leur distribution devient de plus en plus irrégulière au fil du temps. Ce manque de régularité peut nuire à la précision de la résolution numérique du système d'équations vérifié par les particules. Par conséquent, il est important de préserver une distribution régulière des particules. Une solution consiste à "remailler" de temps en temps les particules. Par remailler, on entend ici créer, par une technique d'interpolation, de nouvelles particules régulièrement distribuées, à partir des particules existantes. Ces nouvelles particules sont placées au centre des cellules d'une grille sous-jacente. Cette technique d'interpolation a été en premier envisagée par Harlow ([32], [38]) lorsqu'il a introduit les méthodes PIC. Dans son cas, il s'agissait non pas de créer de nouvelles particules, mais de "convertir" les informations portées par les particules en informations réparties sur une grille. Les noyaux d'interpolation que nous utilisons ici sont cependant davantage inspirés de ceux utilisés dans les méthodes SPH et plus précis que ceux utilisés classiquement dans les méthodes PIC. Il ont déjà été utilisés pour des écoulements compressibles dans [47] et [16].

Le remaillage crée les nouvelles particules de manière conservative, en répartissant les quantités portées par chaque ancienne particule sur les points de grille environnants (voir la figure (1.1)). La manière dont sont réparties ces quantités (telles que la masse, la quantité de mouvement et l'énergie totale dans notre cas) dépend du noyau d'interpolation employé.

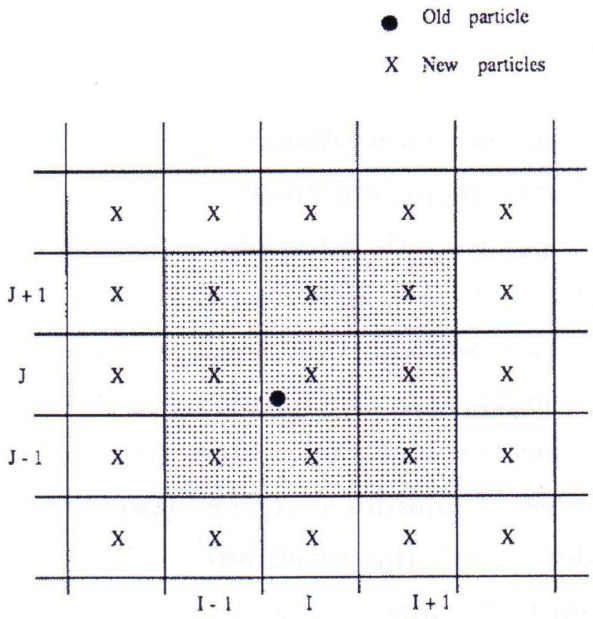


FIG. 1.1 – Remaillage 2D : la zone hachurée indique les cellules sur lesquelles la particule est redistribuée.

La fréquence à laquelle le remaillage doit être effectué dépend de l'écoulement ; elle est très empirique. Il est devenu d'usage ([6]) de remailler les particules à la fin de chaque itération temporelle. Ce choix se justifie si l'on considère que le temps au bout duquel on remaille doit être lié au taux de déformation, et que la condition de stabilité concernant le déplacement des particules, qui détermine le pas de temps, est elle aussi liée à ce taux de déformation. Nous avons donc choisi de remailler les particules à chaque pas de temps. C'est une solution simple, et qui permet de résoudre les équations d'Euler avec des variables disposées sur une grille uniforme, ce qui est plus facile que si elles étaient réparties irrégulièrement. Ce choix nous permettra également, comme nous le verrons dans le chapitre suivant, d'analyser la résolution de l'étape d'advection-remaillage par la méthode particulaire en utilisant le formalisme des différences finies.

1.3.2 Formules de remaillage

Les formules de remaillage que nous présentons sont conçues pour interpoler des quantités sur des grilles cartésiennes uniformes. Ce type d'interpolation a été introduit par Schoenberg ([78]). Pour remailler en dimension deux ou trois, il est courant, et c'est ce que nous ferons, d'utiliser des produits tensoriels de fonctions d'interpolation monodimensionnelles. Par exemple, en dimension 2 :

$$\Lambda(x, y) = \Lambda_1(x)\Lambda_2(y)$$

Dans cette section nous nous concentrons donc sur la présentation des formules de remaillage en dimension un.

Soit une distribution de particules q, situées en x_q, portant les quantités α_q (par exemple la masse, la quantité de mouvement ou l'énergie). Soit un noyau d'interpolation W. Le remaillage crée de nouvelles particules aux centres des cellules d'une grille uniforme, de pas dx. Les nouvelles quantités $\tilde{\alpha_i}$ aux points de grille $\tilde{x}_i$ se calculent en fonction des anciennes avec la formule :

$$\tilde{\alpha}_i = \sum_q \alpha_q W(\frac{\tilde{x}_i - x_q}{dx}) \tag{1.18}$$

Les noyaux d'interpolation usuels sont symétriques, pour ne pas privilégier une direction par rapport aux autres. Il existe des noyaux non-symétriques, créés pour le remaillage sur les bords du domaine [17], mais nous ne les avons pas utilisés. Nous en expliquerons la raison dans la deuxième partie du chapitre, dans la sous-section 2.2.5. Une analyse de Fourier ([45]) permet de montrer que l'ordre de l'interpolation est égal au nombre de moments préservés par la nouvelle distribution de particules par rapport à l'ancienne :

$$\begin{aligned}
\sum_i \tilde{\alpha}_i &= \sum_q \alpha_q \\
\sum_i \tilde{\alpha}_i(x - \tilde{x}_i) &= \sum_q \alpha_q(x - x_q) \\
\sum_i \tilde{\alpha}_i(x - \tilde{x}_i)^2 &= \sum_q \alpha_q(x - x_q)^2 \\
&\dots
\end{aligned}$$

Une famille de noyaux d'interpolation peut être construite en imposant la conservation d'un certain nombre de moments, par la répartition des valeurs portées par les particules sur un nombre de points minimal. Il s'agit de résoudre le système :

$$\sum_{i=1}^{M} W(x - x_i)x_i^a = x^a \quad \text{pour } 0 \leq a \leq M - 1$$

Voici le premier noyau construit suivant ce principe. Il conserve les deux premiers moments de la distribution de particules.

$$\Lambda_1(x) = \begin{cases} 1 - |x| & \text{si } |x| \leq 1 \\ 0 & \text{si } |x| > 1 \end{cases}$$

Ce noyau a un effet très diffusif sur la résolution numérique. Nous ne l'utiliserons pas en pratique.

Le noyau d'interpolation suivant, conservant les trois premiers moment de la distribution de particules, est :

$$\Lambda_2(x) = \begin{cases} 1 - x^2 & \text{si } |x| \leq 0.5 \\ (1 - |x|)(2 - |x|)/2 & \text{si } 0.5 < |x| \leq 1.5 \\ 0 & \text{si } |x| \geq 1.5 \end{cases}$$

Bien qu'utilisé avec succès dans le passé ([**?**], [80]), ce noyau présente l'inconvénient d'avoir un effet très dispersif. Cela s'explique en partie par le fait qu'il n'est pas continu (et que ses discontinuités sont importantes) : une petite erreur sur la position d'une particule peut donc se changer en une erreur importante sur les valeurs des particules créées par le remaillage. Une simulation effectuée avec ce noyau a de sérieux risques de créer des oscillations parasites, notamment au voisinage des discontinuités.

Le noyau d'interpolation suivant nécessite quatre points et préserve un moment supplémentaire. Il est continu et distribue l'information des particules sur un plus grand nombre de points. Il est donc moins dispersif que Λ_2.

$$\Lambda_3(x) = \begin{cases} (1 - x^2)(2 - |x|)/2 & \text{si } |x| \leq 1 \\ (1 - |x|)(2 - |x|)(3 - |x|)/6 & \text{si } 1 < |x| \leq 2 \\ 0 & \text{si } |x| \geq 2 \end{cases}$$

Le noyau M_4, défini ci-après, est plus régulier encore : il est de classe C^2. En revanche il ne préserve que les deux premiers moments : il ne réalise donc que de l'interpolation d'ordre 2 (linéaire). Il est également très diffusif et nous ne l'utiliserons donc pas non plus.

$$M_4(x) = \begin{cases} (2 - |x|)^3/6 - 4(1 - |x|)^3/6 & \text{si } |x| \leq 1 \\ (2 - |x|)^3/6 & \text{si } 1 < |x| \leq 2 \\ 0 & \text{si } |x| \geq 2 \end{cases}$$

Monaghan [62] a construit le très connu noyau M_4', à partir d'une combinaison linéaire de M_4 et de sa dérivée. Ce noyau nécessite quatre points, est de classe C^1 et conserve les trois premiers moments. Depuis quelques années ce noyau est très fréquemment employé dans le cadre des méthodes particulaires pour la mécanique des fluides incompressibles ([46], [17] et [69]).

$$M_4'(x) = \begin{cases} 1 - 5x^2/2 + 3|x|^3/2 & \text{si } |x| \leq 1 \\ (2 - |x|)^2(1 - |x|)/2 & \text{si } 1 < |x| \leq 2 \\ 0 & \text{si } |x| \geq 2 \end{cases}$$

1.3.3 Grilles non cartésiennes uniformes

Pour certaines géométries, il est préférable d'utiliser une grille non-cartésienne ou non-uniforme. C'est le cas notamment des problèmes possédant une symétrie cylindrique, comme le problème de Noh, que nous étudierons dans le chapitre suivant.

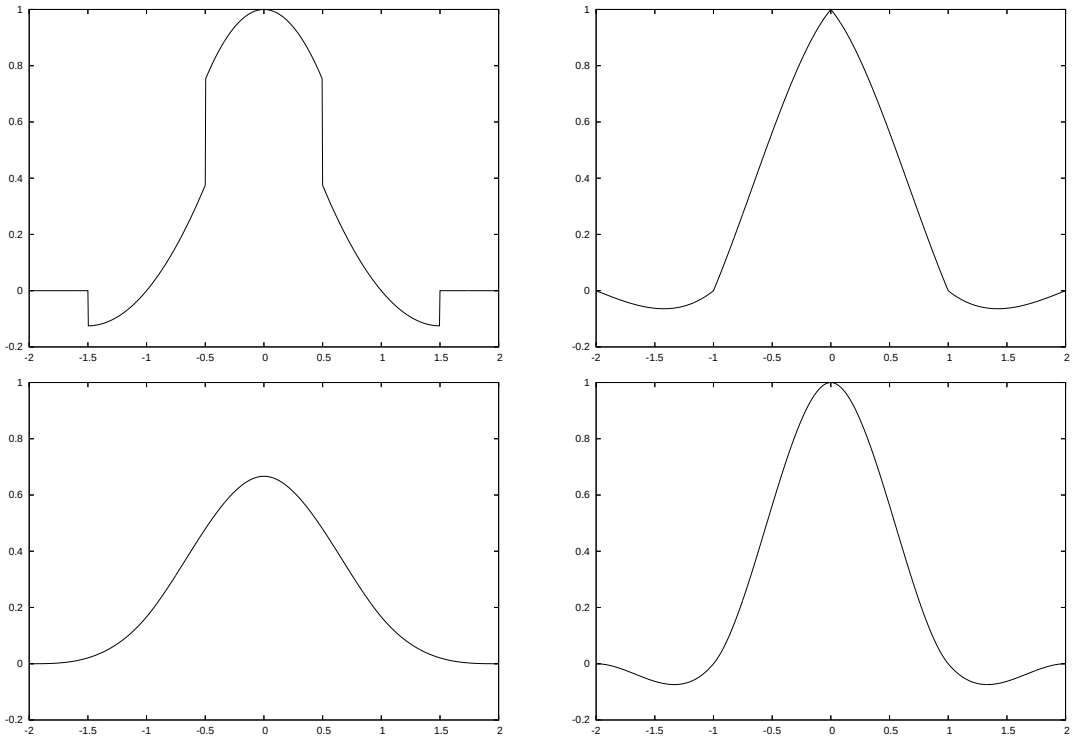


FIG. 1.2 – Noyaux d'interpolation, de haut en bas et de gauche à droite : Λ_2, Λ_3, M_4 et M_4'.

L'interpolation avec la formule 1.18 ne peut être réalisée directement sur une grille qui n'est pas cartésienne uniforme. Par contre, il est possible de réaliser un mapping de la grille dite "physique", utilisée pour la résolution du problème, vers une grille de référence, cartésienne uniforme, de pas dx. On note Ω le domaine physique, x les coordonnées d'un point de ce domaine, $\hat{\Omega}$ le domaine de référence, $\hat{x}$ les coordonnées d'un point de ce domaine, et G l'application inversible qui tranforme les coordonnées de Ω en coordonnées de $\hat{\Omega}$. On reprend les notations de la sous-section 1.3.1. Les positions des particules et les quantités qu'elles portent sont transportées par ce mapping sur la grille cartésienne uniforme (voir la figure (1.3) :

$$\begin{array}{rcl} x_q & \longmapsto & \hat{x_q} = G(x_q) \\ \alpha_q & \longmapsto & \hat{\alpha_q} = G'(x_q)\ \alpha_q \end{array}$$

Le remaillage est réalisé sur $\hat{\Omega}$:

$$\hat{\tilde{\alpha}}_i = \sum_q \hat{\alpha}_q W(\frac{\hat{\tilde{x}}_i - \hat{x}_q}{dx})$$

Puis la transformation inverse de G est appliquée aux positions et quantités portées par les particules ainsi créées, pour obtenir les nouvelles particules de la grille "physique" :

$$\begin{array}{rcl} \hat{\tilde{x}}_i & \longmapsto & \tilde{x_i} = G^{-1}(\hat{\tilde{x}}_i) \\ \hat{\tilde{\alpha}}_i & \longmapsto & \tilde{\alpha_i} = (G)'^{-1}(\hat{\tilde{x}}_i)\ \ \hat{\tilde{\alpha}}_i \end{array}$$

Le remaillage associé à ce mapping reste-t-il conservatif ? La "masse" totale portée par les particules avant remaillage vaut $M = \sum_q \alpha_q$ Après remaillage, la masse totale vaut :

$$\begin{array}{rcl} \tilde{M} & = & \displaystyle\sum_i \tilde{\alpha}_i = \sum_i \hat{\tilde{\alpha}}_i\ (G)'^{-1}(\hat{\tilde{x}}_i) = \sum_i \frac{\hat{\tilde{\alpha}}_i}{G'(\tilde{x}_i)} \\ & = & \displaystyle\sum_i \frac{1}{G'(\tilde{x}_i)} \sum_q \hat{\alpha}_q W(\frac{\hat{\tilde{x}}_i - \hat{x}_q}{dx}) = \sum_i \sum_q \frac{G'(x_q)}{G'(\tilde{x}_i)} \alpha_q W(\frac{\hat{\tilde{x}}_i - \hat{x}_q}{dx}) \end{array}$$

Si $\forall i, \forall q\ \ \dfrac{G'(x_q)}{G'(\tilde{x}_i)} = 1$ (par exemple, si G est une homothétie ou une translation) alors la masse est conservée :

$$\tilde{M} \quad = \quad \sum_q \alpha_q \sum_i W(\frac{\hat{\tilde{x}}_i - \hat{x}_q}{dx}) = \sum_q \alpha_q$$

Mais ce n'est pas le cas en général. Le remaillage associé au mapping n'est donc pas conservatif. Le chapitre 3, consacré à la validation numérique de la méthode, nous permettra de quantifier la perte de masse et de savoir si ce fait est gênant ou non.

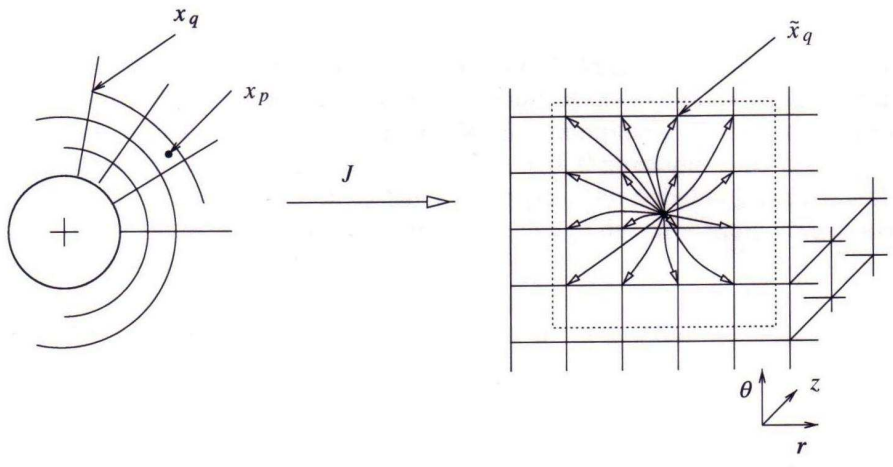


FIG. 1.3 – Mapping d'une grille polaire sur une grille cartésienne uniforme, où les particules sont interpolées.

1.4 Conclusion du chapitre 1

Dans ce chapitre nous avons présenté le principe de la discrétisation particulaire, appliqué aux équations d'Euler compressibles. Le fluide est découpé en particules, possédant une masse, une vitesse, une énergie totale, qui se déplacent de manière lagrangienne, à la vitesse de l'écoulement. Nous résolvons les équations d'Euler en pratiquant une décomposition "Lagrange + advection", c'est-à-dire un splitting entre la prise en compte des efforts de pression et celle du déplacement de la matière. La résolution des efforts de pression est effectuée avec des différences finies centrées ou un solveur de Riemann approché. Le déplacement de la matière est pris en compte par l'advection lagrangienne des particules. Nous avons également rappelé l'utilité de maintenir une distribution régulière de particules en interpolant les "anciennes" particules sur les centres des cellules d'une grille sous-jacente, afin d'y créer de nouvelles particules. Ce procédé, qu'on appelle remaillage, est totalement conservatif quand on utilise une grille cartésienne.

Maintenant que nous avons défini notre méthode, nous allons en étudier la stabilité dans le chapitre suivant. Nous y analyserons ensuite ce qui constitue l'élément essentiel de cette méthode : l'advection des particules, combinée à leur remaillage. Nous mettrons ainsi en évidence des liens forts entre ce type de méthode particulaire et des schémas aux différences finies d'ordre élevé.

Chapitre 2

Analyse de la méthode particules-grille avec remaillage

Dans ce chapitre on analyse la méthode particules-grille présentée dans le chapitre 1. On établit dans la section 2.1 les conditions de stabilité des deux étapes de la décomposition "Lagrange + advection" appliquée à notre méthode. Grâce à cette décomposition, il est possible d'analyser de manière séparée la manière dont l'étape d'advection est résolue par la méthode particulaire. Dans la section 2.2 on analyse donc sur une équation d'advection modèle (1D, scalaire) le schéma numérique correspondant au déplacement des particules combiné à leur interpolation. Du fait de l'étape de remaillage, qui crée à la fin de chaque pas de temps des particules disposées sur une grille, on sait que ce schéma est nécessairement une méthode de grille. On décide donc d'en faire l'analyse en adoptant un formalisme de type différences finies. Cette analyse fait apparaître des liens entre notre méthode particulaire et des schémas aux différences finies d'ordre élevé, de type Lax-Wendroff. En particulier, l'analyse de l'ordre de notre méthode nous amène à proposer un nouveau schéma d'advection des particules, plus précis.

2.1 Conditions de stabilité de la méthode particules-grille pour la dynamique des gaz

Dans cette partie on étudie quelles sont les conditions de stabilité à respecter lors de chacune des deux étapes de la décomposition "Lagrange + advection". Par souci de simplicité, nous effectuons cette analyse dans le cas 1D, l'extension aux 2D et 3D étant immédiate.

2.1.1 Stabilité de la résolution de la phase Lagrange

La résolution de la phase Lagrange consiste à résoudre le système suivant :

$$\begin{aligned}\frac{\partial \rho}{\partial t} &= 0 \\ \frac{\partial \rho U}{\partial t} &= -\frac{\partial p}{\partial x} \\ \frac{\partial \rho E}{\partial t} &= -\frac{\partial (pU)}{\partial x}\end{aligned}$$

On pose :

$$\begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix} = \begin{pmatrix} \rho \\ \rho U \\ \rho E \end{pmatrix}$$

Ces nouvelles variables vérifient :

$$\begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix}_t = (\gamma - 1) \begin{pmatrix} 0 & 0 & 0 \\ \frac{U_2^2}{2U_1^2} & -\frac{U_2}{U_1} & 1 \\ -\frac{U_2 U_3}{U_1^2} + \frac{U_2^3}{U_1^3} & \frac{U_3}{U_1} - \frac{3}{2}\frac{U_2^2}{U_1^2} & \frac{U_2}{U_1} \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix}_x$$

$$\begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix}_t = (\gamma - 1) \begin{pmatrix} 0 & 0 & 0 \\ \frac{U^2}{2} & -U & 1 \\ -EU + U^3 & E - \frac{3}{2}U^2 & U \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix}_x \tag{2.1}$$

Les valeurs propres de la matrice ci-dessus sont les racines du déterminant :

$$(\gamma-1)\begin{vmatrix} 0-X & 0 & 0 \\ \frac{U^2}{2} & -U-X & 1 \\ -EU+U^3 & E-\frac{3}{2}U^2 & U-X \end{vmatrix} = -(\gamma-1)X[(-U-X)(U-X)-(E-\frac{3}{2})] = -(\gamma-1)X(X^2+\frac{U^2}{2}-E)$$

Les valeurs propres sont donc $\lambda_1 = 0$, $\lambda_2 = -(\gamma - 1)(\sqrt{E - \frac{U^2}{2}})$ et $\lambda_3 = (\gamma - 1)(\sqrt{E - \frac{U^2}{2}})$.

Analyser la stabilité linéaire du schéma numérique que nous employons pour résoudre le système (2.1) revient à analyser la stabilité de ce même schéma appliqué aux trois équations d'advection linéaire à coefficients constants :

$$\frac{\partial u}{\partial t} + \lambda_i \frac{\partial u}{\partial x} = 0 \ , \ i = 1, 2, 3$$

Différences finies centrées Nous avons décidé d'utiliser un schéma centré d'ordre 2 en espace, combiné à un schéma de Runge-Kutta TVD d'ordre 4 en temps. Le schéma

global est défini par :

$$\begin{aligned} u^{n+1} &= u^n - \frac{\lambda_i dt}{6}(f_1 + 2f_2 + 2f_3 + f_4) \\ f_1 &= f(u^n) \\ f_2 &= f(u^n - \frac{\lambda_i dt}{2} f_1) \\ f_3 &= f(u^n - \frac{\lambda_i dt}{2} f_2) \\ f_4 &= f(u^n - \lambda_i dt f_3) \\ f(u)_j &= \frac{u_{j+1} - u_{j-1}}{2dx} \end{aligned}$$

On pose $\alpha = \frac{sin(kdx)}{dx}$. Les coefficients de Fourier de u^{n+1} et de u^n vérifient :

$$\hat{u^{n+1}}_k = \hat{u^n}_k - \frac{\lambda_i dt}{6}(\hat{f^1}_k + 2\hat{f^2}_k + 2\hat{f^3}_k + \hat{f^4}_k)$$

Or :

$$\begin{aligned} \hat{f}_{1k} &= \hat{u^n}_k \; i \; \frac{sin(kdx)}{dx} \\ &= i\alpha \; \hat{u^n}_k \\ \hat{f}_{2k} &= i \; \alpha(\hat{u}^n_k - \frac{\lambda_i \; dt}{2} \; \hat{f}_{1k}) \\ &= \hat{u}^n_k[i \; \alpha + \alpha^2 \; \frac{\lambda_i \; dt}{2}] \\ \hat{f}_{3k} &= i \; \alpha(\hat{u}^n_k - \frac{\lambda_i dt}{2} \hat{f}_{2k}) \\ &= \hat{u}^n_k \Big[i \; \alpha + \alpha^2 \; \frac{\lambda_i \; dt}{2} - i \; \alpha^3 \; \frac{\lambda_i^2 \; dt^2}{4} \Big] \\ \hat{f}_{4k} &= i \; \alpha(\hat{u}^n_k - \lambda_i dt \hat{f}_{3k}) \\ &= \hat{u}^n_k \Big[i \; \alpha + \alpha^2 \; \lambda_i \; dt - i \; \alpha^3 \; \frac{\lambda_i^2 \; dt^2}{2} - \alpha^4 \; \frac{\lambda_i^3 \; dt^3}{4} \Big] \end{aligned}$$

Donc :

$$\hat{u^{n+1}}_k = \hat{u}^n_k \Big[1 - i\alpha\lambda_i \; dt - \frac{\alpha^2 \lambda_i^2 dt^2}{2} + i \; \alpha^3 \; \frac{\lambda_i^3 \; dt^3}{6} + \alpha^4 \frac{\lambda_i^4 \; dt^4}{24} \Big]$$

On note $\gamma_i = \alpha\lambda_i dt$. La condition de stabilité du schéma est :

$$|1 - i\gamma_i - \frac{\gamma_i^2}{2} + i\frac{\gamma_i^3}{6} + \frac{\gamma_i^4}{24}| \leq 1$$

Ce qui est équivalent à :

$$\frac{\gamma_i^2}{24} \leq \frac{1}{3}$$

Par conséquent le schéma est stable si

$$|\lambda_i|\frac{dt}{dx} \leq \sqrt{8} \quad \text{pour } i = 1, 2, 3$$

C'est-à-dire si :

$$\sqrt{E - \frac{U^2}{2}}\frac{dt}{dx} \leq \sqrt{8}$$

Ou encore :

$$\sqrt{\frac{(\gamma - 1)p}{\rho}}\frac{dt}{dx} \leq 2\sqrt{2} \tag{2.2}$$

Solveur de Riemann approché La condition CFL est dans ce cas([53]) :

$$c\frac{dt}{dx} \quad \leq \quad \frac{1}{2} \tag{2.3}$$

avec $c = \sqrt{\frac{\gamma p}{\rho}}$ la vitesse du son.

2.1.2 Stabilité de la résolution de la phase d'advection

La deuxième condition de stabilité concerne l'étape d'advection. Dans notre cas, elle est différente de celle mentionnée dans le chapitre précédent. En effet, la stabilité du transport des particules est assurée tant que ces dernières ne se croisent pas. Il faut donc vérifier pour chaque particule i que :

$$\text{si } U_i > U_{i+1} \text{ , alors } dt\, U_i < dx + dt\, U_{i+1}$$

ce qui revient à :

$$dt \quad < \quad \frac{dx}{\max_i \ |U_i - U_{i+1}|_+} \tag{2.4}$$

Dans le cas d'une advection linéaire à coefficients constants, la phase d'advection est donc inconditionnellement stable.

Les méthodes de volumes finis utilisées pour simuler les écoulements de la dynamique des gaz sont soumises au même type de condition de stabilité, puisque les noeuds des éléments de volume ne doivent pas non plus se croiser. Formellement, ce type de condition assure la positivité du jacobien J de l'application qui associe aux coordonnées lagrangiennes les coordonnées eulériennes. Cette propriété de positivité garantit l'existence d'une bijection entre coordonnées lagrangiennes et eulériennes.

2.2 Etude de la méthode pour un problème d'advection modèle

La particularité de notre méthode, par rapport aux méthodes de volumes finis et aux méthodes particulaires classiquement utilisées pour la dynamique des gaz, réside dans le traitement de l'étape d'advection : déplacement des particules puis remaillage avec des fonctions d'interpolation d'ordre élevé. Il est possible, grâce à la décomposition "Lagrange + advection", d'étudier de manière séparée cette étape, et c'est ce que nous faisons ici.

Pour comprendre les propriétés du transport et du remaillage tels qu'utilisés dans la méthode particulaire, on décide de les étudier dans un cas plus simple que les équations d'Euler : sur l'équation de transport scalaire, monodimensionnelle :

$$u_t + (g(u)u)_x = 0 \tag{2.5}$$

Cette équation décrit l'évolution de la quantité u transportée par le flot à la vitesse $g(u)$. Comme nous nous sommes placés en dimension un, si $g(u)_x$ est non identiquement nul alors l'écoulement est compressible. Le remaillage crée à la fin de chaque pas de temps des particules disposées sur une grille et par conséquent la méthode particulaire est une méthode de grille. Nous analysons donc cette dernière en utilisant le formalisme des différences finies. Nous en étudions la précision en fonction du nombre de moments préservés par le noyau d'interpolation. Cette étude fait apparaître des liens forts entre notre méthode particulaire et des schémas aux différences finies d'ordre élevé, de type Lax-Wendroff. Cela nous amène à proposer un nouveau schéma d'ordre 2 pour l'advection des particules, plus simple et plus facile à analyser que les schémas d'advection classiques d'ordre 2 pour les méthodes particulaires. Nous étudions également la stabilité, dans le cas de l'équation de Bürgers, du schéma particulaire utilisant le noyau d'interpolation Λ_2 auquel on a rajouté un terme non-linéaire de viscosité artificielle.

2.2.1 Discrétisation particulaire

On présente la discrétisation particulaire pour cette équation ; elle suit les mêmes principes que celle effectuée pour les équations d'Euler.

Si on exprime l'équation de transport modèle en utilisant la dérivée lagrangienne $\frac{du}{dt} = u_t + g(u)u_x$, on obtient :

$$\frac{du}{dt} + (g(u)_x)u = 0$$

Soit $V(t)$ un volume matériel se déplaçant avec l'écoulement. En appliquant le théorème

de Reynolds, on obtient :

$$\begin{aligned}\frac{d}{dt}\int_{V(t)} u\,dV &= \int_{V(t)} \frac{du}{dt}\,dV + \int_{V(t)} u\,(g(u))_x\,dV \\ \frac{d}{dt}\int_{V(t)} u\,dV &= 0\end{aligned}$$

On découpe le fluide en petits éléments, les particules. Chaque particule j a un volume V_j, porte la quantité constante $\alpha_j = V_j\ u_j$, et se déplace à la vitesse $g(u_j)$. Elle vérifie le système d'équations :

$$\begin{aligned}\frac{d\alpha_j}{dt} &= 0 \\ \frac{dx_j}{dt} &= g(u_j)\end{aligned}$$

2.2.2 Interprétation en termes de schémas aux différences finies

Ce système se résoud très facilement : il suffit, au cours d'un pas de temps, d'advecter les particules, et de les interpoler sur les centres des cellules de la grille uniforme sous-jacente. Les particules sont initialement placées au centre des cellules d'une grille uniforme, de pas dx. On note x_j la position au début du pas de temps de la particule j. Les volumes des particules sont égaux aux volumes des cellules. On note u_j^n la valeur de u portée par la particule j à l'instant ndt, et $\tilde{g}(u)_i^n$ la vitesse à laquelle on déplace cette particule. Cela peut être $g(u_i^n)$, mais pas forcément. $\tilde{g}(u)_i^n$ peut également être fonction de plusieurs valeurs u_j :

$$\tilde{g}(u)_i^n = F(u_{j-m}, ..., u_{j+m})$$

Nous reviendrons sur ce point par la suite. Pour des raisons de consistance on impose juste pour le moment que $F(u, ..., u) = g(u)$. Il est possible d'exprimer u_{j+1}^n en fonction des u_i^n :

Proposition 1 *Si on appelle $\tilde{g}(u)_i^n$ la vitesse à laquelle on déplace la particule i, et Λ le noyau d'interpolation utilisé pour remailler les particules, alors on a :*

$$u_j^{n+1} = \sum_i u_i^n\ \Lambda\big(j - i - \frac{dt\ \tilde{g}(u)_i^n}{dx}\big) \tag{2.6}$$

Remarque: *Dans la formule précédente, il est noté $\sum_i$ sans préciser à quel intervalle appartiennent les i. Les noyaux d'interpolation ayant tous des supports compacts, mais de largeurs différentes, cette notation nous évite de devoir préciser quel est le support correspondant au Λ considéré.*

Preuve: La position de la particule i après déplacement s'exprime :

$$\tilde{x}_i = x_i + dt\ \tilde{g}(u)_i^n$$

Les particules sont remaillées sur la même grille uniforme que celle sur laquelle elles étaient initialement définies, au moyen du noyau d'interpolation Λ. Les positions des nouvelles particules sont donc à nouveau les centres des cellules de la grille, et leurs volumes valent tous dx. La nouvelle distribution de particules $\sum_j \alpha_j^{n+1}\delta(x-x_j)$ se calcule en fonction de celle avant remaillage $\sum_i \alpha_i^n \delta(x-\tilde{x_i})$ par la formule :

$$\alpha_j^{n+1} = \sum_i \alpha_i^n \ \Lambda\big(\frac{x_i - \tilde{x}_i}{dx}\big)$$

Cette équation peut se réécrire :

$$u_j^{n+1}V_j^{n+1} = \sum_i u_i^n V_i^n \ \Lambda\big(\frac{x_j - x_i - dt\ \tilde{g}(u)_i^n}{dx}\big)$$

Comme les volumes des anciennes particules au début du pas de temps, et ceux des nouvelles à la fin de ce même pas de temps sont tous égaux à dx, cette relation se simplifie en :

$$u_j^{n+1} = \sum_i u_i^n \ \Lambda\big(\frac{x_j - x_i - dt\ \tilde{g}(u)_i^n}{dx}\big)$$

De plus, x_i et x_j représentant les centres de cellules d'une grille uniforme, de pas dx, on peut écrire :

$$\frac{x_j - x_i}{dx} = j - i$$

On obtient donc finalement :

$$u_j^{n+1} \quad = \quad \sum_i u_i^n \ \Lambda\big(j - i - \frac{dt\ \tilde{g}(u)_i^n}{dx}\big)$$

□

On reconnaît, sous une forme un peu particulière, un schéma aux différences finies. Par exemple, si on développe cette formule avec le noyau Λ_2 dans le cas linéaire $g(u) = a > 0$, on obtient le schéma de Lax-Wendroff. Nous développerons cette remarque par la suite. En tout cas, cela signifie que nous pouvons analyser la méthode particules-grille avec les outils habituels des différences finies.

Remarque: *A notre connaissance, l'analyse systématique des méthodes particulaires avec remaillage pour des équations hyperboliques sous l'angle des schémas de différences finies n'avait jamais été entreprise. Cependant Monaghan ([61]) avait déjà noté une similarité entre ces deux types de méthodes. Par ailleurs, dans [23], Wee et Ghoniem utilisent une analyse semblable pour construire des noyaux d'interpolation modifiés qui prennent en compte des termes de diffusion.*

2.2.3 Analyse de la précision suivant le noyau d'interpolation

Nous savons à présent que la méthode particulaire peut s'exprimer comme un schéma aux différences finies. Toutefois la formule (2.6), considérée telle quelle, est un peu obscure, car les poids affectés aux valeurs u_i^n s'expriment en fonction du noyau Λ. Est-il nécessaire de connaître l'expression de Λ et de développer le second terme de la formule 2.6 pour étudier les propriétés du schéma ? La réponse est non. La seule connaissance du nombre de moments conservés par le noyau Λ permet d'obtenir plusieurs propriétés du schéma particulaire :

Proposition 2 *Soit Λ un noyau d'interpolation qui conserve au moins les deux premiers moments. Le schéma (2.6) peut s'écrire sous forme conservative et est consistant avec l'équation (2.5).*

Quand on veut résoudre un écoulement où apparaissent des chocs, il est très important que le schéma utilisé puisse s'écrire sous forme conservative et soit consistant : en effet, ces propriétés permettent au schéma de vérifier les relations de Rankine-Hugoniot au travers des discontinuités. Elle assurent, d'après le théorème de Lax-Wendroff (voir [36] et [53]), que si le schéma converge vers une certaine fonction, alors cette fonction est une solution faible de l'équation que l'on cherche à résoudre numériquement. A l'inverse, un schéma non-conservatif utilisé pour résoudre des problème de lois de conservation aura une convergence très problématique ([8] et [42]).

Proposition 3 *Soit Λ un noyau d'interpolation, polynomial par morceaux de degré N, qui conserve les M premiers moments et u une solution de l'équation $u_t + (g(u)u)_x = 0$. Pour $n \geq 0$ donné, on pose $u_j^n = u(jdx, ndt)\ \forall j$.*
Si on suppose que les fonctions u et g sont au moins de classe C^{M-1}, alors u_j^{n+1} défini par (2.6) vérifie :

$$u_j^{n+1} = \sum_{i=0}^{M-1} \frac{\lambda^i}{i!} dx^i (-1)^i \frac{\partial^i (u\tilde{g}^i)}{\partial x^i}(jdx, ndt) + O(\sum_{i=0}^{N} \lambda^i dx^M) \tag{2.7}$$

$$\tag{2.8}$$

Comme les démonstrations des propositions 2 et 3 utilisent le même type de raisonnement et débutent de manière identique, nous les présentons l'une à la suite de l'autre. Nous devrons, lors de ces démonstrations, distinguer deux cas : noyaux d'interpolation dont la largeur du support est un entier impair, et noyaux dont la largeur du support est un entier pair. Par abus de language, nous écrirons par la suite par "noyaux à support pair" ou "noyaux à support impair". La raison en est que ces deux types de noyaux ne sont pas réguliers sur les mêmes intervalles. Nous commençons par présenter les calculs pour ceux à support impair, car les calculs sont plus simples.

2.2.3.1 Cas d'un noyau à support impair

Pour mener à bien les calculs, nous devons d'abord dégager certaines propriétés des noyaux d'interpolations qui nous seront utiles par la suite.

Proposition 4 *Soit Λ une fonction d'interpolation polynômiale par morceaux, à support impair et préservant les M premiers moments. On a alors, pour tout $m \geq 0$ et $0 \leq i \leq M-1$:*

$$\sum_k k^i \Lambda^{(m)}(k) = (-1)^i \, i! \; si \; m = i \tag{2.9}$$

$$= 0 \; sinon \tag{2.10}$$

Preuve: Λ préserve les M premiers moments, donc on peut écrire :

$$\sum_k k^i \Lambda(k-x) = x^i \quad \forall \, 0 \leq i \leq M-1$$

On déduit de cette formule que :

- si $i = 0$ et $m > 0$
$$\sum_k k^i \Lambda^{(m)}(k-x) = 0$$
- si $i = m = 0$
$$\sum_k k^i \Lambda^{(m)}(k-x) = \sum_k \Lambda(k-x) = 1$$
- si $i \neq 0$ et $0 \leq m \leq i$
$$\sum_k k^i (-1)^m \Lambda^{(m)}(k-x) = i(i-1)...(i-m+1)x^{i-m}$$
- si $i \neq 0$ et $m > i$
$$\sum_k k^i (-1)^m \Lambda^{(m)}(k-x) = 0$$

Donc, pour $x = 0$ on a :

- si $i = 0$ et $m > 0$
$$\sum_k k^i \Lambda(k) = 1 \text{ et } \sum k^i \Lambda^{(m)}(k) = 0$$
- si $i \neq 0$ et $m \neq i$
$$\sum_k k^i \Lambda^{(m)}(k) = 0$$
- si $i \neq 0$ et $m = i$
$$\sum_k k^i \Lambda^{(m)}(k) = (-1)^i i \, !$$

□

Proposition 5 *Soit Λ une fonction d'interpolation à support impair compris dans $[-d, d]$ avec d entier et préservant au moins les deux premiers moments.*

$$\sum_{k=1}^{d} k\,\Lambda^{(m)}(k) = -1/2 \quad si\ m = 1 \tag{2.11}$$

$$= 0 \quad sinon \tag{2.12}$$

Preuve: On utilise les résultats (2.9) et (2.10) de la proposition précédente, avec $M \geq 2$ et $i = 1$:

$$\sum_{k} k\,\Lambda^{(m)}(k) = -1 \text{ si } m = 1$$

$$= 0 \text{ sinon}$$

Cela se ré-écrit :

$$\sum_{k=-d}^{-1} k\,\Lambda^{(m)}(k) + \sum_{k=1}^{d} k\,\Lambda^{(m)}(k) = -1 \text{ si } m = 1$$

$$= 0 \text{ sinon}$$

Or, comme Λ est pair, on a :

$$\Lambda^{(m)}(-k) = (-1)^m \Lambda^{(m)}(k)$$

Donc :

$$\sum_{k=1}^{d} -(-1)^m\, k\,\Lambda^{(m)}(k) + \sum_{k=1}^{d} k\,\Lambda^{(m)}(k) = -1 \text{ si } m = 1$$

$$= 0 \text{ sinon}$$

On en conclut que :

$$2\sum_{k=1}^{d} k\,\Lambda^{(m)}(k) = -1 \text{ si } m = 1$$

$$= 0 \text{ sinon}$$

□

Ces propriétés vont nous permettre de simplifier certaines expressions lors des démonstrations qui suivent :

Preuve des propositions 2 et 3 dans le cas d'un noyau à support impair :

Soit u une solution de l'équation $u_t + (g(u)u)_x = 0$ pour une certaine condition initiale. On pose

$$u_j^n = u(jdx, ndt)\ \forall n \geq 0, \forall j \text{ tel que } x_j \text{ appartient au domaine de calcul}$$

On considère la formule (2.6) écrite sous une forme légèrement différente :

$$u_j{}^{n+1} = \sum_k u_{j+k}^n\ \Lambda(k + \frac{dt}{dx}\tilde{g}(u)_{j+k}^n)$$

Pour simplifier la présentation des calculs, on note $\tilde{g}_{j+k} = \tilde{g}(u)_{j+k}^n$, $u_j = u_j^n$ et $\lambda = \frac{dt}{dx}$:

$$u_j{}^{n+1} = \sum_k u_{j+k}\ \Lambda(k + \lambda\tilde{g}_{j+k})$$

Le noyau d'interpolation Λ est polynômial sur des intervalles du type $[k-0.5, k+0.5]$ avec k entier. Il est donc de classe C^∞ en k, pour tout k entier. On suppose que chaque particule, au cours de son déplacement, ne sort pas de la cellule où elle se trouve initialement. Cela signifie que $k + \lambda\tilde{g}_{j+k}$ appartient à $[k-0.5, k+0.5]$. On peut donc développer le terme $\Lambda(k + \lambda\tilde{g}_{j+k})$ en série de Taylor par rapport à k. Si N est le degré de Λ, on a :

$$\Lambda(k + \lambda\ \tilde{g}_{j+k}) = \sum_{i=0}^{N} \frac{\Lambda^{(i)}(k)}{i!}(\lambda\tilde{g}_{j+k})^i$$

On en déduit que :

$$u_j^{n+1} = \sum_k u_{j+k}\ \sum_{i=0}^{N} \frac{\Lambda^{(i)}(k)}{i!}(\lambda\tilde{g}_{j+k})^i \tag{2.13}$$

A partir de maintenant, les preuves des propositions 2 et 3 se scindent :

Fin de la preuve de la proposition 2 (consistance du schéma) :

Comme on a supposé que le support de Λ est compris dans $[-d, d]$, on peut écrire :

$$u_j^{n+1} = \sum_{k=-d}^{-1} u_{j+k}\ \sum_{i=0}^{N} \frac{\Lambda^{(i)}(k)}{i!}(\lambda\tilde{g}_{j+k})^i + \sum_{k=1}^{d} u_{j+k}\ \sum_{i=0}^{N} \frac{\Lambda^{(i)}(k)}{i!}(\lambda\tilde{g}_{j+k})^i + u_j\ \sum_{i=0}^{N} \frac{\Lambda^{(i)}(0)}{i!}(\lambda\tilde{g}_j)^i$$

$$u_j^{n+1} = \sum_{k=1}^{d}\ \sum_{i=0}^{N} \frac{\Lambda^{(i)}(k)}{i!}\lambda^i\left[u_{j+k}(\tilde{g}_{j+k})^i + (-1)^i u_{j-k}(\tilde{g}_{j-k})^i\right] + u_j\ \sum_{i=0}^{N} \frac{\Lambda^{(i)}(0)}{i!}\lambda^i\tilde{g}_j^i$$

Or, comme $\sum_k \Lambda(k) = 1$, on a :

$$\begin{aligned}\Lambda^{(i)}(0) &= \delta_0^i - \sum_{k\neq 0} \Lambda^{(i)}(k) \\ &= \delta_0^i - \sum_{k=-d}^{-1} \Lambda^{(i)}(k) - \sum_{k=1}^{d} \Lambda^{(i)}(k) \\ &= \delta_0^i - \sum_{k=1}^{d} \Lambda^{(i)}(k)\big[1+(-1)^i\big]\end{aligned}$$

Par conséquent :

$$u_j^{n+1} = u_j + \sum_{k=1}^{d}\sum_{i=0}^{N} \frac{\Lambda^{(i)}(k)}{i!}\lambda^i\big[u_{j+k}(\tilde{g}_{j+k})^i + (-1)^i u_{j-k}(\tilde{g}_{j-k})^i - u_j(\tilde{g}_j)^i - (-1)^i u_j(\tilde{g}_j)^i\big]$$

$$u_j^{n+1} = u_j - \lambda\sum_{k=1}^{d}\sum_{i=0}^{N} \frac{\Lambda^{(i)}(k)}{i!}\lambda^{i-1}\big[-u_{j+k}(\tilde{g}_{j+k})^i - (-1)^i u_{j-k}(\tilde{g}_{j-k})^i + u_j(\tilde{g}_j)^i + (-1)^i u_j(\tilde{g}_j)^i\big]$$

On remarque que :

$$-u_{j+k}(\tilde{g}_{j+k})^i + u_j(\tilde{g}_j)^i = \sum_{a=1}^{k} -u_{j+a}\tilde{g}_{j+a}^i + \sum_{a=0}^{k-1} u_{j+a}\tilde{g}_{j+a}^i$$

$$-u_{j-k}(\tilde{g}_{j-k})^i + u_j(\tilde{g}_j)^i = \sum_{a=-k+1}^{0} u_{j+a}\tilde{g}_{j+a}^i - \sum_{a=-k}^{-1} u_{j+a}\tilde{g}_{j+a}^i$$

Donc :

$$\begin{aligned}u_j^{n+1} &= u_j - \lambda \sum_{i=0}^{N}\sum_{k=1}^{d} \frac{\Lambda^{(i)}(k)}{i!}\lambda^{i-1}\big[\sum_{a=1}^{k} -u_{j+a}\tilde{g}_{j+a}^i + \sum_{a=0}^{k-1} u_{j+a}\tilde{g}_{j+a}^i + (-1)^i \sum_{a=-k+1}^{0} u_{j+a}\tilde{g}_{j+a}^i \\ &\quad -(-1)^i \sum_{a=-k}^{-1} u_{j+a}\tilde{g}_{j+a}^i\big]\end{aligned}$$

On peut donc écrire :

$$u_j^{n+1} = u_j - \lambda\,\big[G(u_{j+d},...,u_{j-d+1}) - G(u_{j+d-1},...,u_{j-d})\big]$$

avec :

$$G(u_{j+d},...,u_{j-d+1}) = \sum_{i=0}^{N}\sum_{k=1}^{d} \frac{\Lambda^{(i)}(k)}{i!}\lambda^{i-1}\big[\sum_{a=1}^{k} -u_{j+a}\tilde{g}_{j+a}^i + (-1)^i \sum_{a=-k+1}^{0} u_{j+a}\tilde{g}_{j+a}^i\big]$$

Donc le schéma (2.6) peut s'écrire sous forme conservative.
Par ailleurs :

$$\begin{aligned} G(u,...,u) &= \sum_{i=0}^{N}\sum_{k=1}^{d}\frac{\Lambda^{(i)}(k)}{i!}\lambda^{i-1}\big[\sum_{a=1}^{k}-ug(u)^i+(-1)^i\sum_{a=-k+1}^{0}ug(u)^i\big] \\ &= \sum_{i=0}^{N}\frac{\lambda^{i-1}}{i!}ug(u)^i(-1+(-1)^i)\sum_{k=1}^{d}\Lambda^{(i)}(k)k \end{aligned}$$

D'après la proposition 5, on a $G(u,...,u)=ug(u)$, et donc le schéma (2.6) est consistant avec l'équation (2.5).

Fin de la preuve de la proposition 3 (précision du schéma) :
Le but des calculs qui suivent est d'évaluer l'erreur de troncature du schéma particulaire. On revient à l'expression (2.13) :

$$u_j^{n+1}=\sum_k u_{j+k}\ \sum_{i=0}^{N}\frac{\Lambda^{(i)}(k)}{i!}(\lambda\tilde{g}_{j+k})^i$$

On pose $(f_i)_{j+k}=u_{j+k}\ (\tilde{g}_{j+k})^i$

$$u_j^{n+1}=\sum_{i=0}^{N}\ \sum_k\ \Lambda^{(i)}(k)\frac{\lambda^i}{i!}(f_i)_{j+k}$$

En supposant que les fonctions ont une régularité suffisante, on développe le terme $(f_i)_{j+k}$ en série de Taylor par rapport à $x_j=jdx$:

$$(f_i)_{j+k}^n=\sum_{a=0}^{M-1}(f_i)_j^{(a)}\ \frac{k^a dx^a}{a!}+O(dx^M)$$

La relation devient :

$$u_j^{n+1}=\sum_{i=0}^{N}\frac{\lambda^i}{i!}\ \sum_k\ \left[\sum_{a=0}^{M-1}(f_i)_j^{(a)}\ \frac{k^a dx^a}{a!}+O(dx^M)\right]\ \Lambda^{(i)}(k)$$

$$\Longrightarrow u_j^{n+1}=\sum_{i=0}^{N}\frac{\lambda^i}{i!}\ \sum_{a=0}^{M-1}\left[(f_i)_j^{(a)}\frac{dx^a}{a!}\ \sum_k k^a\Lambda^{(i)}(k)\right]+\sum_{i=0}^{N}O(\lambda^i dx^M)$$

Les propriétés (2.9) et (2.10) nous permettent de simplifier l'équation en :

$$u_j^{n+1}=\sum_{i=0}^{M-1}\frac{\lambda^i}{i!}\ (f_i)_j^{(i)}\ \frac{(-dx)^i}{i!}i!+\sum_{i=0}^{N}O(\lambda^i dx^M)$$

On en conclut que :

$$u_j^{n+1} = \sum_{i=0}^{M-1} \frac{(-dt)^i}{i!} (u_j \tilde{g}_j^i)^{(i)} + \sum_{i=1}^{N} O(\lambda^i dx^M)$$

ce qui finit la démonstration.

La formule (2.7) nous permet d'évaluer la précision de la méthode particulaire, en temps et en espace. Par exemple, pour la fonction Λ_2 on a $M-1 = N = 2$. Si on décide de prendre $\tilde{g}(u) = g(u)$, ce qui est le choix le plus simple, l'erreur de troncature E_j^{n+1} du schéma vaut :

$$\begin{aligned} E_j^{n+1} &= \frac{u(jdx,(n+1)dt) - u_j^{n+1}}{dt} \\ &= \frac{u(jdx,(n+1)dt - u(jdx,ndt))}{dt} + (g(u)u)_x + \frac{dt}{2}(g(u)^2 u)_{xx} + O(dx^2) + O(dxdt) \\ &= u_t + \frac{dt}{2}u_{tt} + (g(u)u)_x + \frac{dt}{2}(g(u)^2 u)_{xx} + O(dx^2) + O(dxdt) + O(dt^2) \\ &= O(dt) \end{aligned}$$

car $u_{tt} + (g(u)^2 u)_{xx} \neq 0$.

Le schéma est donc globalement d'ordre 1. Puisque $u_{tt} = [(g(u)u)'(g(u)u)_x]_x$, son équation équivalente est :

$$u_t + (g(u)u)_x + \frac{dt}{2}([(g(u)u)'(g(u)u)_x]_x + (g^2(u)u)_{xx}) = 0$$

En choisissant $\tilde{g}(u) = g(u)$, on obtient de l'ordre 1 quel que soit le nombre de moment préservés par le noyau, à cause de ce terme non-nul $u_{tt} + (g(u)^2 u)_{xx}$. On propose dans la suite une nouvelle évaluation de la vitesse qui permet de passer outre cette limitation.

2.2.3.2 Cas d'un noyau à support pair

Dans les calculs précédents, nous avons utilisé la régularité du noyau impair sur les intervalles $[k-0.5, k+0.5]$. C'est une propriété qui n'est pas vérifiée par les noyaux à support pair. Ceux-ci sont effet polynomiaux sur les intervalles du type $[k, k+1]$, mais n'ont pas de raison d'être dérivables en k. Il n'est donc pas possible d'obtenir exactement les mêmes propriétés de moments que (2.9), (2.10). Celles-ci restent cependant vraies si on remplace k par k^+ ou k^- et utilise les dérivées à gauche ou à droite de k, puisque Λ est C^∞ de chaque côté de k.

Proposition 6 *Soit Λ une fonction d'interpolation à support pair et préservant les M premiers moments. $\forall\ m, \forall\ 0 \leq i \leq M-1$:*

$$\begin{aligned} \sum k^i \Lambda^{(m)}(k^+) &= 0 \text{ si } m \neq i & (2.14) \\ &= (-1)^i\, i! \text{ si } m = i & (2.15) \end{aligned}$$

Proposition 7 *Soit Λ une fonction d'interpolation à support pair compris dans $[-d, d]$ avec d entier et préservant au moins les deux premiers moments.*

$$\sum_{k=1}^{d} k\,\Lambda^{(m)}(k^+) + \sum_{k=-d}^{-1} -(-1)^m k\,\Lambda^{(m)}(k^-) = -1 \quad si\ m = 1 \tag{2.16}$$

$$= 0 \quad sinon \tag{2.17}$$

Preuve: On utilise les résultats (2.14) et (2.15) de la proposition précédente, avec $M \geq 2$ et $i = 1$:

$$\sum_{k} k\,\Lambda^{(m)}(k^+) = -1 \text{ si } m = 1$$

$$= 0 \text{ sinon}$$

Cela se ré-écrit :

$$\sum_{k=-d}^{-1} k\,\Lambda^{(m)}(k^+) + \sum_{k=1}^{d} k\,\Lambda^{(m)}(k^+) = -1 \text{ si } m = 1$$

$$= 0 \text{ sinon}$$

Or, comme Λ est pair, on a :

$$\Lambda^{(m)}(-k^+) = (-1)^m \Lambda^{(m)}(k^-)$$

On en conclut que :

$$\sum_{k=1}^{d} -(-1)^m\ k\,\Lambda^{(m)}(k^-) + \sum_{k=1}^{d} k\,\Lambda^{(m)}(k^+) = -1 \text{ si } m = 1$$

$$= 0 \text{ sinon}$$

□

Les différences entre noyaux pairs et impairs se manifestent pendant le remaillage de la manière suivante : avec un noyau impair, une particule, tant qu'elle ne sort pas de sa cellule, est toujours distribuée sur les mêmes points de la grille, que son déplacement ait lieu dans un sens ou dans l'autre. Par contre, avec un noyau pair, la particule est redistribuée sur des points différents selon le sens de son déplacement.

Preuve des propositions 2 et 3 dans le cas pair :

On utilise les mêmes notations que dans le cas d'un noyau à support impair.

Le sens du déplacement de la particule j dépend du signe de la vitesse $g(u_j)$, or on ne connaît pas a priori ce signe (à moins de disposer d'hypothèses particulières sur g). Pour mener à bien les calculs pour un noyau à support pair dans le cas général,

il faut donc prendre en compte toutes les possibilités de mouvement des particules. C'est pourquoi on décide de décomposer chaque déplacement de particule en sa partie positive et sa partie négative :

$$\lambda\tilde{g}_{j+k} = \lambda|\tilde{g}_{j+k}|_+ + \lambda|\tilde{g}_{j+k}|_-$$

Une seule de ces valeurs est non nulle, et on a donc formellement :

$$\Lambda\big(k + \lambda\tilde{g}_{j+k}\big) = \Lambda\big(k + \lambda|\tilde{g}_{j+k}|_+\big) + \Lambda\big(k + \lambda|\tilde{g}_{j+k}|_-\big) - \Lambda(k)$$

Par conséquent on peut écrire :

$$u_j{}^{n+1} \;=\; \sum_k u_{j+k}\left[\Lambda\big(k + \lambda|\tilde{g}_{j+k}|_+\big) + \Lambda\big(k + \lambda|\tilde{g}_{j+k}|_-\big)\right] - \sum_k u_{j+k}\Lambda(k) \quad (2.18)$$

Soit N le degré de Λ. Λ étant dérivable sur chaque intervalle $[k, k+1[$, on a :

$$\Lambda\big(k + \lambda|\tilde{g}_{j+k}|_+\big) \;= \Lambda\big(k^+ + \lambda|\tilde{g}_{j+k}|_+\big) \;= \sum_{i=0}^{N} \frac{\Lambda^{(i)}(k^+)}{i!}\big(\lambda|\tilde{g}_{j+k}|_+\big)^i$$

$$\Lambda\big(k + \lambda|\tilde{g}_{j+k}|_-\big) \;= \Lambda\big(k^- + \lambda|\tilde{g}_{j+k}|_-\big) \;= \sum_{i=0}^{N} \frac{\Lambda^{(i)}(k^-)}{i!}\big(\lambda|\tilde{g}_{j+k}|_-\big)^i$$

On en déduit que :

$$u_j^{n+1} = \sum_k \sum_{i=0}^{N} u_{j+k}\frac{\lambda^i}{i!}\left[\Lambda^{(i)}(k^+)|\tilde{g}_{j+k}|_+^i + \Lambda^{(i)}(k^-)|\tilde{g}_{j+k}|_-^i\right] - \sum_k u_{j+k}\Lambda(k)$$

Or $\Lambda^{(i)}(k^-) = (-1)^i\Lambda^{(i)}(-k^+)$

$$u_j^{n+1} = \sum_k \sum_{i=0}^{N} u_{j+k}\frac{\lambda^i}{i!}\Lambda^{(i)}(k^+)|\tilde{g}_{j+k}|_+^i + \sum_k \sum_{i=0}^{N} u_{j+k}(-1)^i\frac{\lambda^i}{i!}\Lambda^{(i)}(-k^+)|\tilde{g}_{j+k}|_-^i - \sum_k u_{j+k}\Lambda(k)$$

On effectue un changement de variable sur la deuxième double somme ($k \leftarrow -k$) pour obtenir :

$$u_j^{n+1} = \sum_k \sum_{i=0}^{N} \frac{\lambda^i}{i!}\Lambda^{(i)}(k^+)\big[u_{j+k}|\tilde{g}_{j+k}|_+^i + (-1)^i u_{j-k}|\tilde{g}_{j-k}|_-^i\big] - \sum_k u_{j+k}\Lambda(k)$$

$$\Longrightarrow u_j^{n+1} = \sum_k \sum_{i=1}^{N} \frac{\lambda^i}{i!}\Lambda^{(i)}(k^+)\big[u_{j+k}|\tilde{g}_{j+k}|_+^i + (-1)^i u_{j-k}|\tilde{g}_{j-k}|_-^i\big]$$

A partir de maintenant, les preuves des propositions 2 et 3 se scindent.

Fin de la preuve de la proposition 2 (consistance du schéma) :

$$u_j^{n+1} = \sum_k \sum_{i=1}^{N} \frac{\lambda^i}{i!} \Lambda^{(i)}(k^+) \big[u_{j+k} |\tilde{g}_{j+k}|_+^i + (-1)^i u_{j-k} |\tilde{g}_{j-k}|_-^i \big]$$

Cette expression peut se ré-écrire :

$$u_j^{n+1} = \sum_{k=1}^{d} \sum_{i=1}^{N} \frac{\lambda^i}{i!} \Lambda^{(i)}(k^+) \big[u_{j+k} |\tilde{g}_{j+k}|_+^i + (-1)^i u_{j-k} |\tilde{g}_{j-k}|_-^i \big] + \sum_{i=1}^{N} \frac{\lambda^i}{i!} \Lambda^{(i)}(0^+) \big[u_j |\tilde{g}_j|_+^i + (-1)^i u_j |\tilde{g}_j|_-^i \big]$$

$$+ \sum_{k=-d}^{-1} \sum_{i=1}^{N} \frac{\lambda^i}{i!} \Lambda^{(i)}(k^+) \big[u_{j+k} |\tilde{g}_{j+k}|_+^i + (-1)^i u_{j-k} |\tilde{g}_{j-k}|_-^i \big]$$

Or, comme $\sum_k \Lambda(k^+) = 1$, on a :

$$\begin{aligned} \Lambda^{(i)}(0^+) &= \delta_0^i - \sum_{k \neq 0} \Lambda^{(i)}(k^+) \\ &= \delta_0^i - \sum_{k=-d}^{-1} \Lambda^{(i)}(k^+) - \sum_{k=1}^{d} \Lambda^{(i)}(k^+) \\ &= \delta_0^i - \sum_{k=1}^{d} (-1)^i \Lambda^{(i)}(k^-) - \sum_{k=1}^{d} \Lambda^{(i)}(k^+) \end{aligned}$$

Donc :

$$u_j^{n+1} = u_j + \sum_{k=1}^{d} \sum_{i=1}^{N} \frac{\lambda^i}{i!} \Lambda^{(i)}(k^+) \big[u_{j+k} |\tilde{g}_{j+k}|_+^i + (-1)^i u_{j-k} |\tilde{g}_{j-k}|_-^i - u_j |\tilde{g}_j|_+^i - (-1)^i u_j |\tilde{g}_j|_-^i \big]$$

$$+ \sum_{k=1}^{d} \sum_{i=1}^{N} \frac{\lambda^i}{i!} \Lambda^{(i)}(k^-) \big[u_{j+k} |\tilde{g}_{j+k}|_-^i + (-1)^i u_{j-k} |\tilde{g}_{j-k}|_+^i - u_j |\tilde{g}_j|_-^i - (-1)^i u_j |\tilde{g}_j|_+^i \big]$$

On remarque que :

$$-u_{j+k} |\tilde{g}_{j+k}|_+^i + u_j |\tilde{g}_j|_+^i = \sum_{a=1}^{k} -u_{j+a} |\tilde{g}_{j+a}|_+^i + \sum_{a=0}^{k-1} u_{j+a} |\tilde{g}_{j+a}|_+^i$$

$$-u_{j-k} |\tilde{g}_{j-k}|_-^i + u_j |\tilde{g}_j|_-^i = \sum_{a=-k+1}^{0} u_{j+a} |\tilde{g}_{j+a}|_-^i - \sum_{a=-k}^{-1} u_{j+a} |\tilde{g}_{j+a}|_-^i$$

De même :

$$-u_{j+k} |\tilde{g}_{j+k}|_-^i + u_j |\tilde{g}_j|_-^i = \sum_{a=1}^{k} -u_{j+a} |\tilde{g}_{j+a}|_-^i + \sum_{a=0}^{k-1} u_{j+a} |\tilde{g}_{j+a}|_-^i$$

$$-u_{j-k}|\tilde{g}_{j-k}|_+^i + u_j|\tilde{g}_j|_+^i = \sum_{a=-k+1}^{0} u_{j+a}|\tilde{g}_{j+a}|_+^i - \sum_{a=-k}^{-1} u_{j+a}|\tilde{g}_{j+a}|_+^i$$

Donc :

$$u_j^{n+1} = u_j - \lambda \sum_{k=1}^{d}\sum_{i=1}^{N} \frac{\lambda^{i-1}}{i!}\Lambda^{(i)}(k^+)\big[\sum_{a=1}^{k} -u_{j+a}|\tilde{g}_{j+a}|_-^i + \sum_{a=0}^{k-1} u_{j+a}|\tilde{g}_{j+a}|_-^i\big]$$

$$+\sum_{k=1}^{d}\sum_{i=1}^{N} \frac{\lambda^{i-1}}{i!}\Lambda^{(i)}(k^-)\big[\sum_{a=-k+1}^{0} u_{j+a}|\tilde{g}_{j+a}|_+^i - \sum_{a=-k}^{-1} u_{j+a}|\tilde{g}_{j+a}|_+^i\big]$$

On peut donc écrire :

$$u_j^{n+1} = u_j - \lambda\,\big[G(u_{j+d},...,u_{j-d+1}) - G(u_{j+d-1},...,u_{j-d})\big]$$

avec :

$$G(u_{j+d},...,u_{j-d+1}) = \sum_{i=0}^{N}\sum_{k=1}^{d} \frac{\Lambda^{(i)}(k^+)}{i!}\lambda^{i-1}\big[\sum_{a=1}^{k} -u_{j+a}|\tilde{g}_{j+a}|_+^i + (-1)^i \sum_{a=-k+1}^{0} u_{j+a}|\tilde{g}_{j+a}|_-^i\big]$$

$$+\sum_{i=0}^{N}\sum_{k=1}^{d} \frac{\Lambda^{(i)}(k^-)}{i!}\lambda^{i-1}\big[\sum_{a=1}^{k} -u_{j+a}|\tilde{g}_{j+a}|_-^i + (-1)^i \sum_{a=-k+1}^{0} u_{j+a}|\tilde{g}_{j+a}|_+^i\big]$$

Donc le schéma (2.6) peut s'écrire sous forme conservative.
Par ailleurs :

$$\begin{aligned}
G(u,...,u) &= \sum_{i=0}^{N}\sum_{k=1}^{d} \frac{\Lambda^{(i)}(k^+)}{i!}\lambda^{i-1}\big[-ku|g(u)|_+^i + (-1)^i ku|g(u)|_-^i\big] \\
&\quad +\sum_{i=0}^{N}\sum_{k=1}^{d} \frac{\Lambda^{(i)}(k^-)}{i!}\lambda^{i-1}\big[-ku|g(u)|_-^i + (-1)^i ku|g(u)|_+^i\big] \\
&= \sum_{i=0}^{N} \frac{\lambda^{i-1}}{i!}\sum_{k=1}^{d} u|g(u)|_+^i\big[-k\Lambda^{(i)}(k^+) + (-1)^i k\Lambda^{(i)}(k^-)\big] \\
&\quad +\sum_{i=0}^{N} \frac{\lambda^{i-1}}{i!}\sum_{k=1}^{d} u|g(u)|_-^i\big[(-1)^i k\Lambda^{(i)}(k^+) - k\Lambda^{(i)}(k^-)\big]
\end{aligned}$$

D'après la proposition 5, on a donc :

$$G(u,...,u) = u|g(u)|_+ + u|g(u)|_- = ug(u)$$

Le schéma (2.6) est donc consistant avec l'équation (2.5).

Fin de la preuve de la proposition 3 (précision du schéma) :

On repart de la formule :

$$u_j^{n+1} = \sum_k \sum_{i=1}^{N} \frac{\lambda^i}{i!} \Lambda^{(i)}(k^+) \left[u_{j+k} |\tilde{g}_{j+k}|_+^i + (-1)^i u_{j-k} |\tilde{g}_{j-k}|_-^i \right]$$

On développe les termes $|\tilde{g}_{j+k}|_+^i$ et $|\tilde{g}_{j-k}|_-^i$:

$$|\tilde{g}_{j+k}|_+^i = \frac{(\tilde{g}_{j+k} + |\tilde{g}_{j+k}|)^i}{2^i} = \sum_{n=0}^{i} \frac{C_i^n}{2^i} \tilde{g}_{j+k}^n |\tilde{g}_{j+k}|^{i-n}$$

$$|\tilde{g}_{j-k}|_-^i = \frac{(\tilde{g}_{j-k} - |\tilde{g}_{j-k}|)^i}{2^i} = \sum_{n=0}^{i} \frac{C_i^n}{2^i} (-1)^{i-n} \tilde{g}_{j-k}^n |\tilde{g}_{j-k}|^{i-n}$$

En insérant ces sommes dans l'expression de u_j^{n+1}, on obtient :

$$u_j^{n+1} = \sum_k \sum_{i=0}^{N} \frac{\lambda^i}{i!} \Lambda^{(i)}(k^+) \sum_{n=0}^{i} \frac{C_i^n}{2^i} \left[u_{j+k} \tilde{g}_{j+k}^n |\tilde{g}_{j+k}|^{i-n} + (-1)^{i-n} u_{j-k} \tilde{g}_{j-k}^n |\tilde{g}_{j-k}|^{i-n} \right]$$

On pose $(f_{i,n})_{j+k} = u_{j+k} \tilde{g}_{j+k}^n |\tilde{g}_{j+k}|^{i-n}$

$$(f_{i,n})_{j+k} = \sum_{a=0}^{M-1} \frac{(f_{i,n})_j^{(a)}}{a!} k^a dx^a + O(dx^M)$$

$$\Longrightarrow (f_{i,n})_{j-k} = \sum_{a=0}^{M-1} \frac{(f_{i,n})_j^{(a)}}{a!} (-1)^a k^a dx^a + O(dx^M)$$

On a :

$$u_{j+k} \tilde{g}_{j+k}^n |\tilde{g}_{j+k}|^{i-n} + (-1)^{i-n} u_{j-k} \tilde{g}_{j-k}^n |\tilde{g}_{j-k}|^{i-n} = \sum_{a=0}^{M-1} \frac{(f_{i,n})_j^{(a)}}{a!} (1 + (-1)^{a+n}) k^a dx^a + O(dx^M)$$

Soit :

$$u_j^{n+1} = \sum_k \sum_{i=0}^{N} \sum_{n=0}^{i} \sum_{a=0}^{M-1} \frac{\lambda^i}{i!} \frac{C_i^n}{2^i} \Lambda^{(i)}(k^+) \frac{(f_{i,n})_j^{(a)}}{a!} (1 + (-1)^{a+n}) k^a dx^a$$

$$+ \sum_k \sum_{i=0}^{N} \sum_{n=0}^{i} \frac{\lambda^i}{i!} \frac{C_i^n}{2^i} \Lambda^{(i)}(k^+) O(dx^M)$$

On permute les signes $\sum$ et on écrit le second terme plus simplement :

$$u_j^{n+1} = \sum_{i=0}^{N} \sum_{n=0}^{i} \sum_{a=0}^{M-1} \frac{\lambda^i}{i!} \frac{C_i^n}{2^i} \frac{(f_{i,n})_j^{(a)}}{a!} (1 + (-1)^{a+n}) dx^a \sum_k k^a \Lambda^{(i)}(k^+) \; + O(\sum_{i=0}^{N} \lambda^i dx^M)$$

Les formules (2.14) et (2.15) nous permettent de simplifier l'expression précédente :

$$u_j^{n+1} = \sum_{i=0}^{M-1} \sum_{n=0}^{i} \frac{\lambda^i}{i!} \frac{C_i^n}{2^i} (f_{i,n})_j^{(i)} (1 + (-1)^{i+n}) dx^i (-1)^i + O(\sum_{i=0}^{N} \lambda^i dx^M)$$

$$u_j^{n+1} = \sum_{i=0}^{M-1} \sum_{n=0}^{i} \frac{\lambda^i C_i^n}{i! 2^i} (u_j \tilde{g}_j^n |\tilde{g}_j|^{i-n})^{(i)} (1 + (-1)^{i+n}) dx^i (-1)^i + O(\sum_{i=0}^{N} \lambda^i dx^M)$$

$$u_j^{n+1} = \sum_{i=0}^{M-1} \sum_{n=0, i+npair}^{i} 2 \frac{\lambda^i C_i^n}{2^i i!} dx^i (-1)^i (u_j \tilde{g}_j^i)^{(i)} + O(\sum_{i=0}^{N} \lambda^i dx^M)$$

Il nous reste à remarquer que $\sum_{n=0, i+npair}^{i} C_i^n = 2^{i-1}$. (La preuve se trouve à la fin de la démonstration.)
On obtient finalement la même formule que dans le cas d'un support impair :

$$u_j^{n+1} = \sum_{i=0}^{M-1} \frac{\lambda^i}{i!} dx^i (-1)^i (u_j \tilde{g}_j^i)^{(i)} + O(\sum_{i=0}^{N} \lambda^i dx^M)$$

Remarque: *Les calculs précédents sont aussi valables dans le cas d'une fonction d'interpolation à support impair, bien qu'ils soient alors inutilement compliqués, puisqu'il n'est pas la peine de distinguer les signes des vitesses.*

Lemme 2.1

$$\sum_{n=0, i+npair}^{i} C_i^n = 2^{i-1}$$

Preuve: Si i et n sont pairs :

$$\begin{aligned}
C_i^0 &= C_{i+1}^0 \\
C_i^1 + C_i^2 &= C_{i+1}^2 \\
C_i^3 + C_i^4 &= C_{i+1}^4 \\
&\dots \\
C_i^{i-1} + C_i^i &= C_{i+1}^i \\
\sum_{n=0}^{i} C_i^n &= \sum_{n\, pair, n \le i} C_{i+1}^n = \sum_{n\, pair, n \le i+1} C_{i+1}^n
\end{aligned}$$

Si i et n sont impairs :

$$\begin{aligned} C_i^0 + C_i^1 &= C_{i+1}^1 \\ C_i^2 + C_i^3 &= C_{i+1}^3 \\ C_i^4 + C_i^5 &= C_{i+1}^5 \\ &\dots \\ C_i^{i-1} + C_i^i &= C_{i+1}^i \\ \sum_{n=0}^{i} C_i^n &= \sum_{n\, impair, n\leq i} C_{i+1}^n = \sum_{n\, impair, n\leq i+1} C_{i+1}^n \end{aligned}$$

On a donc montré que :

$$\sum_{n, n\leq i+1} C_{i+1}^n = \sum_{n=0}^{i} C_i^n$$

Or $\sum_{n=0}^{i} C_i^n = 2^i$, donc :

$$\sum_{n, n\leq i+1} C_{i+1}^n = 2^i$$

Et finalement :

$$\sum_{n, n\leq i} C_i^n = 2^{i-1}$$

□

2.2.4 Schémas obtenus avec des noyaux d'interpolation connus

Les résultats de cette section et de la section suivante ont fait l'objet d'une note aux Comptes-Rendus de l'Académie des Sciences de Paris ([22]).

L'étude que nous venons d'effectuer nous renseigne sur la précision des schémas particulaires, sans avoir besoin de développer l'expression 2.6. Cependant elle ne nous permet pas d'observer la différence entre des schémas utilisant des noyaux d'interpolation différents, mais qui préservent le même nombre de moments : par exemple, les schémas avec Λ_2 et M_4'. Il peut donc être intéressant d'étudier les expressions développées de ces schémas.

Commençons par étudier le schéma aux différences finies obtenu avec le noyau d'interpolation Λ_1. On prend $\tilde{g} = g$. Dans le cas de Λ_1, la formule (2.18) s'écrit :

$$u_j^{n+1} = u_{j-1}\, \Lambda_1\big(1 - \lambda|g_{j-1}|_+\big) + u_j\, \Lambda_1\big(\lambda g_j\big) + u_{j+1}\, \Lambda_1\big(1 + \lambda|g_{j+1}|_-\big)$$

Après développement on obtient cette expression du schéma avec Λ_1 :

$$u_j^{n+1} = u_j + \lambda\big(|g_{j-1}|_+ + g_j - |g_{j+1}|_+\big)$$

Le schéma avec Λ_1 est un schéma décentré d'ordre 1.

Remarque: *Le schéma avec Λ_1 est équivalent, pour cette équation, au schéma de Steger et Warming. Mais ce n'est pas vrai dans le cas général. Avec la méthode particulaire, on obtient des schémas décentrés suivant le signe de la vitesse matérielle. Dans le cas du schéma de Steger et Warming, et plus généralement des schémas de flux-splitting, le décentrage a lieu suivant le signe de la vitesse de propagation des différentes ondes qui composent la solution.*

Intéressons-nous maintenant au schéma utilisant la fonction d'interpolation Λ_2. En développant Λ_2, on obtient pour expression de u_j^{n+1} :

$$u_j^{n+1} = u_j - \frac{\lambda}{2}\big(\tilde{g}_{j+1} - \tilde{g}_{j-1}\big) + \frac{\lambda^2}{2}\big(\tilde{g}^2_{j+1}u_{j+1} - 2\tilde{g}^2_j u_j + \tilde{g}^2_{j-1}u_{j-1}\big) \qquad (2.19)$$

Proposition 8 *Dans le cas linéaire $g(u) = a = constante$, le schéma (2.19) est équivalent à la solution exacte pendant $\frac{1}{a}[\frac{adt}{dx}]$, combinée au schéma de Lax-Wendroff pendant $dt - \frac{1}{a}[\frac{adt}{dx}]$.*
Si on note $N = [\frac{adt}{dx}]$, le schéma avec Λ_2 s'écrit :

$$u_j^{n+1} = u_{j-N} - \frac{\lambda a}{2}(u_{j+1-N} - u_{j-1-N}) + \frac{a^2\lambda^2}{2}(u_{j+1-N} - 2u_j + u_{j-1-N}) \qquad (2.20)$$

Preuve: En posant $g(u) = a$ on retrouve le schéma de Lax-Wendroff :

$$u_j^{n+1} = u_j - \frac{\lambda a}{2}(u_{j+1} - u_{j-1}) + \frac{a^2\lambda^2}{2}(u_{j+1} - 2u_j + u_{j-1})$$

Pour obtenir cette expression, on a supposé que les particules ne sortent pas de la cellule où elles se trouvent au début du pas de temps. Or, dans le cas d'une advection à vitesse uniforme, les particules sont transportées de manière exacte, et il n'y a pas besoin de limiter ainsi leur mouvement. Le pas de temps peut donc être aussi large qu'on le souhaite. Le schéma de Lax-Wendroff, quant à lui, est soumis à la condition CFL $\frac{\lambda a}{dt} \leq 1$. On peut donc considérer que le schéma particulaire avec Λ_2, utilisant le pas de temps dt, est équivalent à la solution exacte pendant $\frac{1}{a}[\frac{adt}{dx}]$ (c'est le temps que met la particule pour traverser les $[\frac{adt}{dx}]$ premières cellules), combinée au schéma de Lax-Wendroff pendant $dt - \frac{1}{a}[\frac{adt}{dx}]$ (c'est le temps pendant lequel la particule se déplace dans la dernière cellule, où elle se trouve au moment du remaillage).
□

Dans le cas général, le schéma avec Λ_2 est différent du schéma de Lax-Wendroff. Il s'agit de toute manière d'un schéma d'ordre 1 si on prend $\tilde{g}(u) = g(u)$.

Les schémas particulaires avec M_4' et Λ_3 sont des schémas décentrés, qui ne nous ont pas rappelé de schémas aux différences finies connus. Leurs expressions développées sont assez longues, puisqu'on doit décomposer le déplacement de chaque particule impliquée

dans le remaillage en somme de sa partie positive et de sa partie négative. Voici celle obtenue avec M_4' :

$$\begin{aligned} u_j^{n+1} &= u_j - \frac{\lambda}{2}\left(\tilde{g}_{j+1}u_{j+1} - \tilde{g}_{j-1}u_{j-1}\right) \\ &\quad + \frac{\lambda^2}{2}\big(- |\tilde{g}_{j+2}|_-^2 u_{j+2} + 2|\tilde{g}_{j+1}|_+^2 u_{j+1} + 4|\tilde{g}_{j+1}|_-^2 u_{j+1} - 5\tilde{g}_j^2 u_j \\ &\quad + 4|\tilde{g}_{j-1}|_+^2 u_{j-1} + 2|\tilde{g}_{j-1}|_-^2 u_{j-1} - |\tilde{g}_{j-2}|_+^2 u_{j-2}\big) \\ &\quad - \frac{\lambda^3}{2}\big(|\tilde{g}_{j+2}|_-^3 u_{j+2} + |\tilde{g}_{j+1}|_+^3 u_{j+1} - 3|\tilde{g}_{j+1}|_+^3 u_{j+1} - 3\tilde{g}_j^3 u_j \\ &\quad + 3|\tilde{g}_{j-1}|_+^3 u_{j-1} - |\tilde{g}_{j-1}|_-^3 u_{j-1} - |\tilde{g}_{j-2}|_+^3 u_{j-2}\big) \end{aligned}$$

Voici celle obtenue avec Λ_3 :

$$\begin{aligned} u_j^{n+1} &= u_j - \lambda\big(- \frac{1}{6}|\tilde{g}_{j+2}|_- u_{j+2} + \frac{1}{3}|\tilde{g}_{j+1}|_+ u_{j+1} + |\tilde{g}_{j+1}|_- u_{j+1} + \frac{1}{2}|\tilde{g}_j| u_j \\ &\quad - |\tilde{g}_{j-1}|_+ u_{j-1} - \frac{1}{3}|\tilde{g}_{j-1}|_- u_{j-1} + \frac{1}{6}|\tilde{g}_{j-2}|_+ u_{j-2}\big) \\ &\quad + \frac{\lambda^2}{2}\big(|\tilde{g}_{j+1}|^2 u_{j+1} - 2|\tilde{g}_j| u_j + |\tilde{g}_{j-1}|^2 u_{j-1}\big) \\ &\quad - \lambda^3\big(\frac{1}{6}|\tilde{g}_{j+2}|_-^3 u_{j+2} + \frac{1}{6}|\tilde{g}_{j+1}|_+^3 u_{j+1} - \frac{1}{2}|\tilde{g}_{j+1}|_-^3 u_{j+1} - \frac{1}{2}|\tilde{g}_j|^3 u_j \\ &\quad + \frac{1}{2}|\tilde{g}_{j-1}|_+^3 u_{j-1} - \frac{1}{6}|\tilde{g}_{j-1}|_-^3 u_{j-1} - \frac{1}{6}|\tilde{g}_{j-2}|_+^3 u_{j-2}\big) \end{aligned}$$

2.2.5 Correction de l'évaluation de la vitesse

On a vu précédemment qu'en choisissant pour vitesse de déplacement des particules $\tilde{g}(u) = g(u)$, on obtient un schéma d'ordre un. C'est en effet un terme d'ordre 1 en temps, $\frac{dt}{2}(u_{tt} + (g^2(u)u)_{xx})$, qui limite la précision du schéma. Ce résultat est prévisible dans la mesure où le choix $\tilde{g}(u) = g(u)$ consiste à déplacer la particule pendant le pas de temps dt avec sa vitesse évaluée en début de pas de temps :

$$\tilde{x}_j = x_j + dt\ g(u_j)$$

Il s'agit d'un schéma d'Euler explicite et le déplacement de la particule est d'ordre un. Pour augmenter la précision du schéma, il faut calculer le déplacement, et donc la vitesse des particules au cours du pas de temps dt, avec davantage de précision. Nous proposons d'utiliser la vitesse qu'aurait la particule au milieu du pas de temps, à $t + \frac{dt}{2}$. Comme nous ne connaissons pas la valeur de la particule à cet instant, nous la remplaçons par une approximation à l'ordre 2 : $u + \frac{dt}{2}\frac{du}{dt}$. (Nous utilisons une dérivée lagrangienne car les particule se déplacent de manière lagrangienne.) Nous déplaçons donc les particules à la vitesse :

$$\tilde{g}(u) = g\left(u + \frac{dt}{2}\frac{du}{dt}\right) \tag{2.21}$$

Ce qui donne :

$$\tilde{x}_j = x_j + dt\ g\left(u + \frac{dt}{2}\frac{du}{dt}\right)_{x=x_j}$$

L'erreur de troncature correspondant à ce déplacement est :

$$\begin{aligned} E &= \frac{\tilde{x}_j - x_j}{dt} \\ &= \frac{dx}{dt} + \frac{1}{2}\frac{d^2x}{dt^2} - g(u) - g'(u)\frac{dt}{2}\frac{du}{dt} + O(dt^2) \\ &= O(dt^2) \end{aligned}$$

Le déplacement des particules est donc calculé à l'ordre deux.

Proposition 9 *Pour les solutions régulières, le schéma (2.6) utilisé avec (2.21) est d'ordre 2 si* Λ *conserve au moins les trois premiers moments.*

Preuve: L'erreur de troncature du schéma devient :

$$\begin{aligned} E_j^{n+1} &= \frac{u(jdx,(n+1)dt) - u_j^{n+1}}{dt} \\ E_j^{n+1} &= \frac{u(jdx,(n+1)dt) - \sum_{i=0}^{M-1}\frac{\lambda^i}{i!}dx^i(-1)^i\frac{\partial^i(u\tilde{g}^i)}{\partial x^i}(jdx,ndt) + O(\sum_{i=0}^{N}\lambda^i dx^M)}{dt} \\ E_j^{n+1} &= u_t + \frac{dt}{2}u_{tt} + O(dt^2) + (g(u+\frac{dt}{2}\frac{du}{dt})u)_x + \frac{dt}{2}(g(u+\frac{dt}{2}\frac{du}{dt})^2u)_{xx} + O(dx^2) + O(dxdt) \\ E_j^{n+1} &= u_t + \frac{dt}{2}u_{tt} + O(dt^2) + (g(u)u)_x + \frac{dt}{2}(\frac{du}{dt}g'(u))_x + O(dt^2) + \frac{dt}{2}(g(u)^2u)_{xx} \\ &\quad + O(dt^2) + O(dx^2) + O(dxdt) \end{aligned}$$

On s'intéresse à la somme $u_{tt} + (\frac{du}{dt}g'(u))_x + (g(u)^2u)_{xx}$.

$$u_{tt} = -(g(u)u)_{tx} = -\Big((g'(u)u + g(u))u_t\Big)_x = \Big((g'(u)u + g(u))(g(u)u)_x\Big)_x$$

$$\frac{du}{dt} = -(g(u))_x$$

On vérifie que :

$$\Big((g'(u)u + g(u))(g(u)u)_x\Big)_x - ((g(u))_x g'(u))_x + (g(u)^2u)_{xx} = 0$$

On obtient donc :

$$E_j^{n+1} = O(dt^2) + O(dxdt)$$

□

Habituellement, pour obtenir un déplacement des particules à l'ordre 2, ces dernières étaient advectées pendant un demi-pas de temps puis interpolées spatialement pour obtenir de nouvelles particules uniformément distribuées. Ces nouvelles particules étaient à leur tour déplacées pendant un demi-pas de temps. Le nouveau schéma que nous proposons pour déplacer les particules avec une précision d'ordre 2 est donc plus simple, et plus facile à analyser, car il ne fait pas intervenir d'interpolation supplémentaire.

Proposition 10 *Le schéma (2.6), (2.21) avec Λ_2 a pour équation équivalente :*

$$u_t + (g(u)u)_x + dt^2(-\frac{1}{6}[(g(u)u)'^2(2(g(u)u)''u_x^2 + (g(u)u)_{xx})]_x + \frac{1}{6\lambda^2}(g(u)u)_{xxx} + \frac{1}{8}(g(u)_x^2 g''(u)u^3)_x + \frac{1}{2}(g(u)g'(u)^2u^2u_x)_{xx}) = 0$$

En particulier, dans le cas de l'équation de Bürgers, l'équation équivalente de ce schéma est :

$$u_t + (\frac{u^2}{2})_x + dt^2(-\frac{1}{6}[2u^2u_x^2 + u^3u_{xx}]_x + \frac{1}{6\lambda^2}(\frac{u^2}{2})_{xxx} + \frac{1}{16}(u^3u_x)_{xx} = 0$$

Preuve: Pour obtenir ce résultat on calcule l'erreur de troncature jusqu'à l'ordre 3 et on exprime u_{ttt} en fonction de u, $g(u)$ et de leurs dérivées partielles
□

A titre de comparaison, le schéma de Lax-Wendroff a pour équation équivalente sur l'équation de Bürgers :

$$u_t + (\frac{u^2}{2})_x + dt^2(-\frac{1}{6}[2u^2u_x^2 + u^3u_{xx}]_x + \frac{1}{6\lambda^2}(\frac{u^2}{2})_{xxx}) = 0$$

Les équations équivalentes de ces deux schémas sont donc très proches. Elles ne nous permettent pas d'expliquer leur différence de comportement, qu'on observera dans le prochain chapitre : le schéma (2.6) utilisé avec Λ_2 converge vers la solution entropique dans un cas où le schéma de Lax-Wendroff converge vers une solution non-entropique.

En pratique, pour calculer numériquement la nouvelle évaluation de la vitesse (2.21), on remarque que :

$$\frac{du}{dt} = -g(u)_x u$$

Donc :

$$g(u + \frac{dt}{2}\frac{du}{dt}) = g(u(1 - \frac{dt}{2}g(u)_x))$$

On évalue avec des différences finies centrées d'ordre 2 le terme $g(u)_x$:

$$\tilde{g}_j^n = g(u_j^n - \frac{dt}{4dx}(u_{j+1} - u_{j+1})) \tag{2.22}$$

Cette approximation supplémentaire sur le champ d'advection des particules ne modifie pas l'ordre de la méthode :

Proposition 11 *Le schéma (2.6), (2.22) est d'ordre 2 pour des solutions régulières (si Λ conserve au moins les trois premiers moments).*

Preuve: Le calcul de l'erreur de troncature est presque le même que celui de la proposition 6 ; il faut juste montrer que (2.22) est une approximation de $g(u + \frac{dt}{2}\frac{du}{dt})$ à l'ordre 2.
□

Revenons au système de la dynamique des gaz : quelle serait cette correction de la vitesse lors de la résolution de l'étape d'advection des équations d'Euler ?

Proposition 12 *Pour le système des équations d'Euler, résoudre le déplacement des particules avec un schéma d'Euler explicite donne de l'ordre 2 (à condition que le noyau d'interpolation préserve au moins les trois premiers moments).*

Preuve: Au lieu d'une équation scalaire on considère le système (pour simplifier, en 1D) :

$$\begin{aligned} \rho_t + (\rho u)_x &= 0 \\ (\rho u)_t + (\rho u u)_x &= 0 \\ (\rho E)_t + (\rho u E)_x &= 0 \end{aligned}$$

qui s'écrit également :

$$\begin{aligned} \frac{d\rho}{dt} + \rho u_x &= 0 \\ \frac{d\rho u}{dt} + \rho u u_x &= 0 \\ \frac{d\rho E}{dt} + \rho E u_x &= 0 \end{aligned}$$

Appliquer la correction revient à advecter les particules à la vitesse $\tilde{u} = \dfrac{\rho u + \frac{dt}{2}\frac{d\rho u}{dt}}{\rho + \frac{dt}{2}\frac{d\rho}{dt}}$ au lieu de $\dfrac{\rho u}{\rho}$. Or :

$$\tilde{u} = \frac{\rho u - \frac{dt}{2}\rho u u_x}{\rho - \frac{dt}{2}\rho u_x} = \frac{\rho u}{\rho}$$

Par conséquent, pour les équations d'Euler, advecter les particules avec un schéma d'Euler explicite donne en fait de l'ordre 2. □

2.2.6 Utiliser des formules décentrées sur les bords ?

Des formules décentrées ont parfois été proposées ([45]) pour redistribuer les particules situées sur les bord du domaine de calcul : par exemple, la fonction nommée Λ_J, qui répartit les valeurs de la particule située sur la dernière cellule de la grille sur les trois dernières cellules (voir la figure (2.1)), en préservant les trois premiers moments. On pose $v = (x - x_q)/dx$ avec dx le pas de la grille et x_q la position de la particule

située dans la dernière cellule de la grille.

$$\begin{aligned}\Lambda_J(x) &= 1 - 3/2v + 1/2v^2 \text{ si x appartient à la dernière cellule} \\ &= v(2-v) \text{ si x appartient à l'avant-dernière cellule} \\ &= v(v-1)/2 \text{ si x appartient à l'avant-avant dernière cellule} \\ &= 0 \text{ sinon}\end{aligned}$$

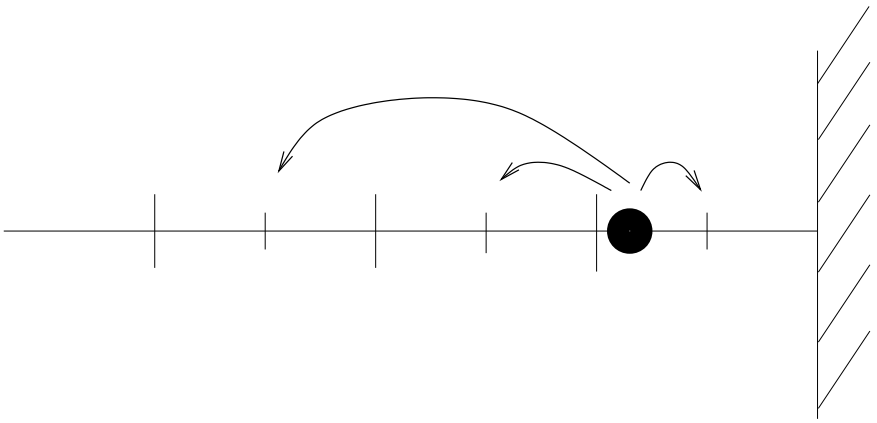


FIG. 2.1 – Utilisation de la fonction d'interpolation décentrée Λ_J : les valeurs portées par la particule située sur la dernière cellule sont distribuées de manière décentrée, sur les trois dernières cellules de la grille.

Cette formule permet d'éviter de créer des particules fictives pour le remaillage. Mais quel est son effet ? Supposons par exemple que le remaillage se fait partout ailleurs avec la fonction Λ_2. Comparons l'utilisation de Λ_J à l'utilisation de particules fictives. Avec Λ_2, il suffit de créer une seule particule fictive pour le remaillage. Quelle que soit la valeur attribuée à cette particule fictive (qui dépend des conditions aux limites), on constate que :

- Seule la dernière cellule de la grille reçoit une contribution de la particule fictive (car on utilise Λ_2, dont le support a pour largeur 3).
- Par contre, avec Λ_J, les trois dernières cellules recoivent des contributions a priori non-nulles.

On en déduit que Λ_J n'a pas un rôle équivalent à celui des particules fictives.

On considère maintenant plus particulièrement le cas uniforme : $\forall\ i \quad u_i = a$ avec comme des conditions aux limites périodiques. La particule fictive porte donc aussi la valeur a. On note V la vitesse à laquelle se déplacent les particules. Chaque cellule j recoit (voir la figure (2.2)) :

$$a\ \Lambda_2(1-\lambda V) + a\ \Lambda_2(\lambda V) + a\ \Lambda_2(1+\lambda V)$$

Cette somme vaut a car le support de Λ_2 a pour largeur 3. Donc les particules créées par le remaillage ont toutes pour valeur a. Si on utilise la formule Λ_J plutôt qu'une particule fictive, l'avant-avant-dernière cellule va recevoir (voir la figure (2.3)) :

$$a\ \Lambda_2(1-\lambda V) + a\ \Lambda_2(\lambda V) + a\ \Lambda_2(1+\lambda V) + a\ (\lambda V+2)(\lambda V+1)/2$$

$$= a + a\ (\lambda V+2)(\lambda V+1)/2 \neq a$$

Le remaillage avec Λ_J ne préserve donc pas les états constants, c'est-à-dire que si les particules, avant le remaillage, portent toutes la même valeur, ce ne sera plus forcément le cas après le remaillage. Ces raisonnements nous amènent à penser que ce genre de fonction ne semble pas interpoler de manière appropriée les particules du bord.

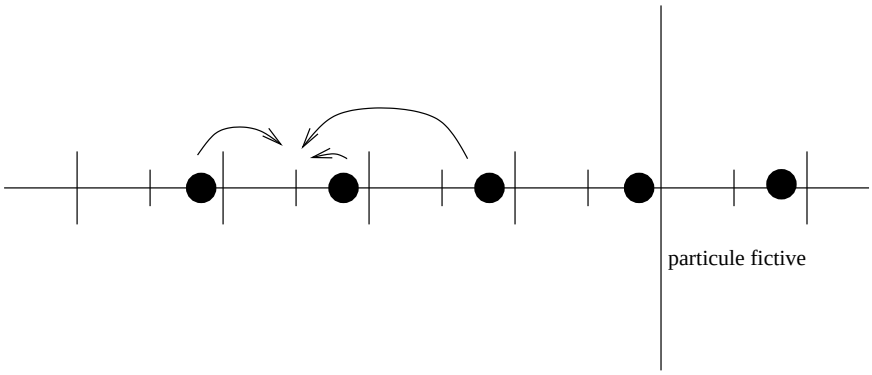


FIG. 2.2 – Remaillage sur le bord du domaine avec une particule fictive. La particule fictive n'intervient pas dans la création de la nouvelle particule située sur la troisième cellule à partir du bord. Cette dernière reçoit les contributions de trois particules.

2.2.7 Correction non-linéaire du schéma avec Λ_2 pour l'équation de Bürgers

Nous avons noté précédemment que le schéma particulaire avec Λ_2 était équivalent dans le cas linéaire au schéma de Lax-Wendroff. Ce dernier est connu pour être un schéma très dispersif. Majda et Osher ont montré dans [57] que l'ajout d'un terme correctif non-linéaire pouvait stabiliser le schéma de Lax-Wendroff appliqué à l'équation de Bürgers. Plus précisément, ils ont montré le résultat suivant :

Théorème (Majda-Osher, [57]) :
Pour une suite périodique u_j il existe un C tel que le terme correctif non-linéaire $C(|u_{j+1}-u_j|(u_{j+1}-u_j) - |u_j-u_{j-1}|(u_j-u_{j-1}))$ assure, sous la condition de type CFL, la décroissance de l'énergie pour le schéma de Lax-Wendroff :
Si les u_j^{n+1} résultent du schéma de Lax-Wendroff corrigé appliqué aux u_j^n, alors, sous la condition $\epsilon_0 = max_j|u_j|\dfrac{dt}{dx} \leq 0.24$:

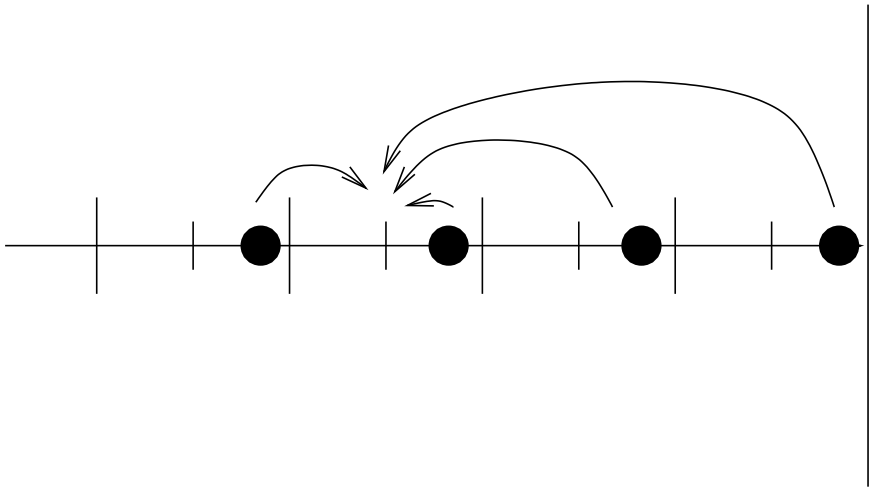


FIG. 2.3 – Remaillage sur le bord du domaine avec une fonction d'interpolation décentrée pour la dernière particule. La troisième cellule à partir du bord reçoit une contribution supplémentaire de la particule située sur la dernière cellule.

$$\sum_j u_j^{n+1} \leq \sum_j u_j^n$$

Inspirés par sa similarité avec le schéma de Lax-Wendroff, nous avons établi un résultat similaire pour le schéma particulaire (2.6), (2.21) avec le noyau d'interpolation Λ_2 :

Proposition 13 *Pour une suite périodique u_j il existe un C tel que le terme de correction non-linéaire $C(|u_{j+1}-u_j|(u_{j+1}-u_j)-|u_j-u_{j-1}|(u_j-u_{j-1}))$ assure la décroissance de l'énergie pour le schéma (2.6), (2.21) avec Λ_2, sous la condition $\epsilon_0 = max_j|u_j|\frac{dt}{dx} \leq 0.15$.*

On verra dans la section 3.2.1.3 du prochain chapitre que l'ajout de ce terme au schéma (2.6), (2.21) utilisé avec le noyau d'interpolation Λ_2, mais également avec M_4' ou Λ_3 atténue les oscillations parasites et améliore la résolution du pied d'une détente. Nous l'avons dans cette section utilisé avec une condition CFL moins stricte et un coefficient C plus petit. Cela suggère que le résultat de la proposition 13 n'est pas optimal.

Comme le calcul est assez long, nous en présentons seulement les grandes lignes. Les détails sont donnés en annexe.

Preuve: Dans la suite, on utilise les notations suivantes :

$$\begin{aligned} \Delta_+ u_j &= u_{j+1} - u_j \\ \Delta_- u_j = u_j - u_{j-1} &= \Delta_+ u_{j-1} \\ \Delta_0 u_j &= u_{j+1} - u_{j-1} \\ \epsilon_0 &= \lambda \max_j |u_j| \end{aligned}$$

Le schéma avec Λ_2 et la correction en vitesse s'écrit :

$$u_j^{n+1} = u_j^n - \frac{\lambda}{2}(u_{j+1}^n \tilde{g}_{j+1}^n - u_{j-1}^n \tilde{g}_{j-1}^n) + \frac{\lambda^2}{2}(u_{j+1}^n (\tilde{g}_{j+1}^n)^2 - 2u_j^n (\tilde{g}_j^n)^2 + u_{j-1}^n (\tilde{g}_{j-1}^n)^2)$$

avec $\tilde{g}_j^n = g(u_j^n(1 - \frac{dt}{4dx}(g(u_{j+1}) - g(u_{j-1}))))$

Avec les notations définies ci-dessus, le schéma se ré-écrit :

$$\begin{aligned} u_j^{n+1} &= u_j^n \underbrace{- \frac{\lambda}{4} \Delta_0 u_j^2}_{F_j} + \underbrace{\frac{\lambda^2}{32} \Delta_0 (\, u_j^2 \Delta_0 \,) + \frac{\lambda^2}{8} \Delta_- \Delta_+ u_j^3}_{G_j} \underbrace{- \frac{\lambda^3}{64} \Delta_- \Delta_+ (\, u_j^3 \Delta_0 u_j \,)}_{K_j} \\ &\quad + \underbrace{C\lambda \, \Delta_- (\, |\Delta_+ u_j| \, \Delta_+ u_j \,)}_{J_j} \end{aligned}$$

Le bilan d'énergie s'écrit :

$$\begin{aligned} \sum_j u_j^{n+1\,2} - \sum_j u_j^{n2} &= 2\sum u_j J_j + 2\sum u_j F_j + 2\sum J_j F_j + \sum {F_j}^2 \\ &\quad + 2\sum u_j G_j + \sum {J_j}^2 + 2\sum J_j G_j + 2\sum u_j K_j + 2\sum F_j G_j \\ &\quad + \sum {G_j}^2 + 2\sum F_j K_j + 2\sum J_j K_j + 2\sum G_j K_j + \sum {K_j}^2 \end{aligned}$$

On essaie de majorer chaque terme non-négatif par une expression où apparaît $\sum |\Delta_+ u_j|^3$. La somme, positive, de ces expressions en $\sum |\Delta_+ u_j|^3$ est censée être compensée par le terme strictement négatif $2\sum u_j J_j = -2\lambda C \sum |\Delta_+ u_j|^3$. On verra que c'est le cas pour un coefficient C approprié, si une certaine condition CFL est vérifiée.

Les termes pour lesquels cette majoration ne semble pas possible sont regroupés en une somme, que l'on prouve être négative sous une autre condition CFL.

Après une série de calculs (présentés en annexe), on obtient donc que :

$$\begin{aligned} \sum u_j^{n+1\,2} - \sum u_j^{n2} &\leq \lambda \Big[-2C + 1/6 + 8C^2\epsilon_0 + 4C\epsilon_0 + \frac{14}{16}\epsilon_0 + \frac{7}{2}C\epsilon_0^2 + \frac{3}{32}\epsilon_0^2 + \frac{62}{64}\epsilon_0^2 \\ &\quad + \frac{1}{2}C\epsilon_0^3 + \frac{56}{128}\epsilon_0^3 + \frac{190}{64}\epsilon_0^3 + \frac{59}{64}\epsilon_0^4 + \frac{280}{64^2}\epsilon_0^5 \Big] \\ &\quad + \lambda^2 \sum (\Delta_- \Delta_+ u_j)^2 \Big[-\frac{3}{32}(u_j^2 + u_{j+1}^2) + \frac{4}{128}(u_j^2 + u_{j+1}^2) \\ &\quad + \epsilon_0^2 (\frac{42}{256} u_j^2 + \frac{22}{256} u_{j+1}^2) + \frac{1}{64^2}\epsilon_0^4 (3u_j^2 + u_{j+1}^2) \Big] \end{aligned} \tag{2.23}$$

Le terme en λ^2 est négatif si $\epsilon_0 \leq 0.79$.

Le terme en λ est négatif si C est compris entre les racines, si elles existent, du polynôme :

$$P(x) = 8\epsilon_0 x^2 + (-2 + 4\epsilon_0 + \frac{7}{2}\epsilon_0^2 + \frac{1}{3}\epsilon_0^3)x + \frac{1}{6} + \frac{14}{16}\epsilon_0 + \frac{68}{64}\epsilon_0^2 + \frac{220}{64}\epsilon_0^3 + \frac{196}{256}\epsilon_0^4 + \frac{280}{64^2}\epsilon_0^5$$

On vérifie que P a des racines réelles si $\epsilon_0 \leq 0.15$. Dans ce cas, $C = 0.6$ convient.

Donc, pour un tel C, et sous la condition CFL $\epsilon_0 \leq 0.15$, on a :

$$\sum_j u_j^{n+1\,2} - \sum_j u_j^{n\,2} = 0$$

ce qui conclut la démonstration. □

2.3 Conclusion du chapitre 2

Dans ce chapitre nous avons tout d'abord analysé la stabilité de la méthode particulaire. Celle-ci doit vérifier deux conditions de stabilité, une pour chaque étape du splitting entre la prise en compte des efforts de pression et celle du déplacement de la matière. La première condition de stabilité est une condition CFL classique (2.2), alors que la deuxième (2.4) limite le pas de temps indépendamment du pas d'espace, en fonction du gradient de la vitesse.

Nous avons ensuite voulu étudier le point essentiel de notre méthode : la résolution de la phase d'advection, combinée au remaillage. Nous avons analysé le traitement de cette étape sur une équation d'advection modèle, monodimensionnelle et scalaire. Notre étude nous a permis de ré-écrire les schémas particulaires sous forme de schémas aux différences finies (2.6) et d'analyser leur précision en fonction de la fonction d'interpolation utilisée pour le remaillage (2.7). Cela a fait apparaître des liens forts entre ce type de méthode particulaire et des schémas aux différences finies d'ordre élevé, de type Lax-Wendroff. En particulier, dans le cas linéaire $g(u) = a$, la précision du schéma particulaire dépend uniquement du nombre de moments conservés par le noyau d'interpolation, et il n'y a aucune condition CFL à respecter, contrairement aux schémas aux différences finies classiques. Dans le cas non-linéaire, nous avons proposé une nouvelle manière d'évaluer la vitesse des particules (2.21, 2.22), afin d'obtenir des schémas d'ordre 2. Les schémas obtenus avec la méthode particulaire présentent aussi l'avantage d'être très facilement extensibles au 2D et au 3D, puisque le remaillage se fait alors simplement par produit tensoriel de noyaux d'interpolation 1D. Nous avons également montré, dans le cas d'un noyau d'interpolation particulier, et pour l'équation de Bürgers, que l'ajout d'un terme de viscosité artificielle proposé dans [57] stabilisait la méthode.

Il nous faut maintenant étudier numériquement la méthode. Il s'agit non seulement de valider les propriétés que nous avons mises en évidence dans le cas de notre équation modèle, mais aussi de tester le comportement de la méthode particulaire sur des problèmes typiques de la dynamique des gaz.

Chapitre 3

Validation numérique

L'objectif de ce chapitre est d'évaluer la capacité de la méthode particulaire à simuler les écoulements de la dynamique des gaz. On commence par détailler dans la section 3.1 l'algorithme utilisé, ainsi que certains détails d'implantation de la méthode. On présente ensuite dans la section 3.2 des résultats obtenus sur un certain nombre de tests numériques. Les premiers d'entre eux visent à valider numériquement les résultats théoriques de précision et de stabilité obtenus dans le chapitre précédent. On mettra d'ailleurs en évidence des propriétés des schémas particulaires qui n'avaient pas été prédites par notre étude théorique. Les tests suivants ont pour but d'évaluer le comportement de la méthode lors de la résolution de problèmes typiques de la dynamique des gaz. En particulier, on étudie l'influence de l'utilisation de différents noyaux d'interpolation, ainsi que des différences finies centrées et du solveur de Riemann approché.

3.1 Description de l'algorithme

Dans cette partie on décrit l'algorithme de résolution des équations d'Euler par la méthode particulaire : initialisation, résolution à chaque pas de temps des deux étapes de la décomposition "Lagrange + advection", remaillage, ajout de viscosité artificielle, traitement des conditions aux limites.

3.1.1 Définition des grilles

La méthode particulaire nécessite d'utiliser à chaque pas de temps une grille fixe ; sur celle-ci sont résolues les équations de l'étape lagrangienne et interpolées les particules. Suivant les domaines de calcul, il s'agit d'une grille cartésienne (en une ou deux dimensions), ou parfois d'une grille polaire (en deux dimensions).

Une grille cartésienne en une dimension est composée de cellules de taille dx (voir la figure (3.1)). Chaque cellule se repère par son indice j. On appelle x_j le centre de la cellule j. Si la grille s'étend sur le segment $[a, b]$, on identifie x_j par :

$$x_j = a + (j - 0.5)\, dx$$

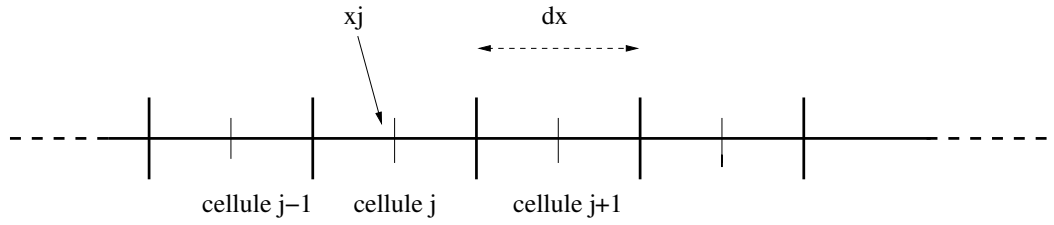


FIG. 3.1 – Grille cartésienne 1D.

Une grille cartésienne en deux dimensions est composée de cellules rectangulaires de taille $dx \times dy$ (voir la figure 3.2). Chaque cellule est repérée par un doublet i, j, correspondant à sa position dans la grille. Si la grille s'étend sur $[a, b] \times [c, d]$, le centre de la cellule i, j a pour coordonnées $x(i, j), y(i, j)$:

$$x(i, j) = a + (i - 0.5)\ dx$$

$$y(i, j) = c + (j - 0.5)\ dy$$

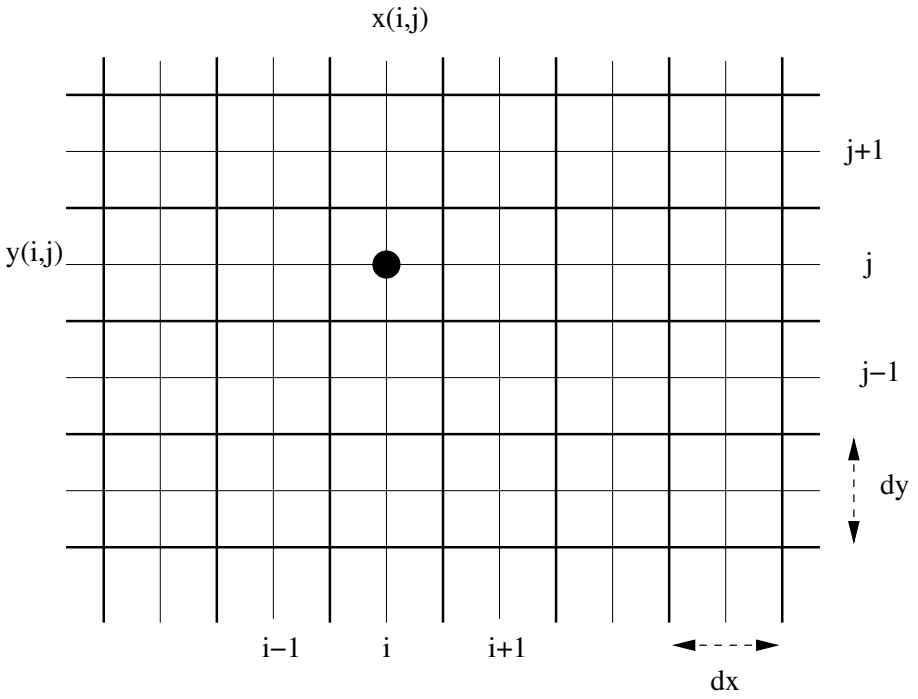


FIG. 3.2 – Grille cartésienne 2D.

Les cellules d'une grille polaire sont, quant à elles, des sections de couronnes (voir la figure (3.3)). On note dr le pas d'espace radial et $d\theta$ le pas angulaire. Une cellule

indexée par i, j a pour centre le point de coordonnées polaires

$$r(i,j) = (i - 0.5)\ dr$$

$$\theta(i,j) = (j - 0.5)\ d\theta$$

et pour volume

$$V(i,j) = r\ \times\ dr\ \times\ d\theta$$

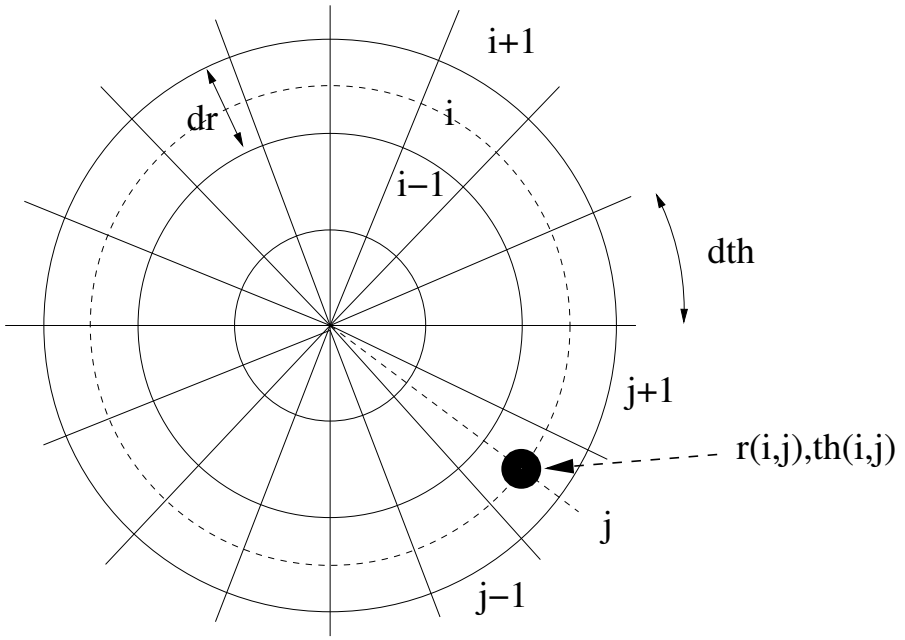


FIG. 3.3 – Grille polaire.

3.1.2 Initialisation/ Début d'un nouveau pas de temps

Au début d'un nouveau pas de temps, chaque particule est placée au centre d'une cellule de la grille, et son volume est égal à celui de la cellule. Il y a donc autant de particules que de cellules sur la grille. On rappelle que c'est la même grille fixe qui est utilisée à chaque itération temporelle. Chaque particule porte les mêmes valeurs de densité, vitesse et énergie que la cellule à laquelle elle appartient. Ces valeurs sont soit données par la condition initiale, soit issues du remaillage au pas de temps précédent.

Les variables portées par une particule i sont :

$$
\begin{array}{rcl}
V_i & = & \text{volume de la particule } i = \text{volume de la cellule où se trouve la particule} \\
m_i & = & \text{masse de la particule} \\
(\,m\vec{U}\,)_i & = & \text{quantité de mouvement de la particule} \\
(\,m\,E\,)_i & = & \text{énergie totale de la particule} \\
x_i & = & \text{position de la particule} = \text{centre de la cellule}
\end{array}
$$

3.1.3 Résolution des équations d'Euler

Comme on l'a vu dans le chapitre 1, il s'agit de résoudre pour chaque particule i le système :

$$\frac{dm_i}{dt} = 0 \tag{3.1}$$

$$\frac{d(m\vec{U})_i}{dt} = -V_i(\nabla p)_{x=x_i} \tag{3.2}$$

$$\frac{d(mE)_i}{dt} = -V_i(\nabla.(p\vec{U}))_{x=x_i} \tag{3.3}$$

$$\frac{dx_i}{dt} = \vec{U}_i \tag{3.4}$$

et on pratique pour cela une décomposition "Lagrange + advection". Le système est donc résolu en deux étapes : d'abord les équations (3.1), (3.2) et (3.3), puis l'équation (3.4).

3.1.3.1 Détermination du pas de temps

En résolvant les équations d'Euler par une décomposition "Lagrange + advection", le pas de temps est soumis à deux conditions de stabilité différentes. La première correspond à la résolution de la phase Lagrange. Avec les différences finies centrées nous avons vu dans le chapitre 2 qu'elle est (en 1D) :

$$\sqrt{\frac{(\gamma-1)p}{\rho}}\frac{dt}{dx} \leq 2\sqrt{2}$$

avec dt le pas de temps et dx le pas de la grille. Pour le solveur approché de Riemann, elle est :

$$c\frac{dt}{dx} \leq \frac{1}{2}$$

avec c la vitesse du son.
La deuxième condition de stabilité concerne l'étape d'advection. La stabilité du transport des particules est assurée tant que les particules ne se croisent pas. En 1D, comme nous l'avons vu dans le chapitre 2, il faut vérifier pour chaque particule i que :

$$dt < \frac{dx}{\max_i\ |U_i - U_{i+1}|_+}$$

3.1.3.2 Résolution de la phase Lagrange

Il s'agit de résoudre numériquement les équations (3.1) à (3.3). Les détails de cette résolution ont été donnés dans la section 1.2.1 du premier chapitre.

Les valeurs portées en début de pas de temps par la particule i : m_i, $(mU)_i$ et $(mE)_i$, deviennent après la résolution de cette étape $\tilde{m}_i$, $(\tilde{mU})_i$ et $(\tilde{mE})_i$

3.1.3.3 Résolution de la phase d'advection

Cette étape consiste à déplacer les particules à la vitesse de l'écoulement, puis à les remailler sur la grille sous-jacente, elle a déjà été évoquée dans la section 1.2.2 du premier chapitre.

La position de la particule i devient $\tilde{x}_i$:

$$\tilde{x_i} = x_i + dt\frac{(\rho\tilde{V}U)_i}{(\rho\tilde{V})_i}$$

3.1.3.4 Remaillage

Cette étape a également été décrite dans le chapitre 1. Elle consiste à remplacer les anciennes particules par de nouvelles, distribuées uniformément, tout en préservant un certain nombre de moments. Il s'agit de distribuer les quantités $\tilde{m}_j$, $(\tilde{mU})_j$ et $(\tilde{mE})_j$ portées par les anciennes particules, placées en $\tilde{x}_j$, sur les centres x_i des cellules de la grille, à l'aide d'une fonction d'interpolation Λ. On note m_i^{n+1}, $(m\vec{U})_i^{n+1}$ et $(mE)_i^{n+1}$les valeurs attribuées aux nouvelles particules. En dimension un, cela s'écrit :

$$\begin{aligned}
m_i^{n+1} &= \sum_j \tilde{m}_j \, \Lambda(\frac{x_i - \tilde{x_j}}{dx}) \\
(mU)_i^{n+1} &= \sum_j (\tilde{mU})_j \, \Lambda(\frac{x_i - \tilde{x_j}}{dx}) \\
(mE)_i^{n+1} &= \sum_j (\tilde{mE})_j \, \Lambda(\frac{x_i - \tilde{x_j}}{dx})
\end{aligned}$$

Les nouvelles particules sont placées au centre des cellules de la grille. Leur volume est égal à celui de la cellule à laquelle elles sont rattachées, soit dx. On a donc :

$$V_i^{n+1} = dx$$

En pratique, on utilisera le plus souvent les fonctions d'interpolations M_4' et Λ_3. Ces noyaux d'interpolation conservant tous les trois premiers moments, l'étape d'advection est bien résolue à l'ordre 2 (voir 2.2.4).

3.1.4 Ajout d'un terme de viscosité artificielle

De manière générale, la viscosité artificielle sert à atténuer les effets dispersifs des schémas numériques [44]. Plus particulièrement, elle peut être utilisée pour :

- rendre effectif le dépôt d'entropie au niveau des chocs
- atténuer le caractère dispersif des opérateurs discrets de dérivation spatiale ou d'intégration temporelle
- compenser les erreurs d'interpolation spatiale (phénomène de Gibbs).

En hydrodynamique lagrangienne, pour tenir compte du dépôt d'entropie, on utilise classiquement, dans les schémas formulés en énergie interne (schémas de type Von Neumann-Richtmeyer, dits "VNR"), une viscosité artificielle sous la forme d'un terme supplémentaire q rajouté à la pression dans les équations :

$$p' = p + q$$

p' est utilisé à la place de la pression p pour résoudre les équations d'évolution de la quantité de mouvement et de l'énergie. La forme originelle de cette viscosité artificielle, dite "quadratique", a été proposée par Von Neumann ([66], [85]) :

$$\begin{aligned} q_1 &= c_0^2\ \rho\ \Delta x^2\ \frac{\partial u}{\partial x}\ |\frac{\partial u}{\partial x}| \\ &= c_0^2\ \rho\ \Delta u\ |\Delta u| \end{aligned}$$

avec Δx la taille d'une cellule, Δu la différence de vitesse à travers la cellule, et c_0 une constante. En 1955, Landshoff [50] a proposé une autre forme (dite "linéaire") :

$$\begin{aligned} q_2 &= c_L\ \rho\ a_0\ \Delta x\ |\frac{\partial u}{\partial x}| \\ &= c_L\ \rho\ a_0\ |\Delta u| \end{aligned}$$

avec a_0 la vitesse locale du son et c_L une constante. La forme recommandée pour q est une combinaison de ces deux dernières :

$$q = q_1 + q_2 = c_0^2\ \rho\ |\Delta u|\ \Delta u\ +\ c_L\ \rho\ a_0\ |\Delta u|$$

Le premier terme est d'autant plus grand que le gradient est fort. Le second terme sert à amortir les longueurs d'onde d'échelle 2 dx, ce pour quoi le premier terme peut être insuffisant. Cette viscosité est utilisée pour faire apparaître un dépôt d'entropie aux niveaux des chocs. Il suffit donc de l'utiliser là où on suppose qu'il y a un choc. On ajoute donc ce terme de viscosité seulement si on détecte une compression.

Dans notre cas, nous n'utilisons un terme de viscosité artificielle que si les équations sont résolues avec des différences finies. En effet, si on utilise un solveur de Riemann, celui-ci est censé être entropique, et il n'est donc pas nécessaire de rajouter un terme pour tenir compte du dépôt d'entropie. De plus, le solveur que nous utilisons est d'ordre un, donc assez diffusif, et il n'y a donc pas non plus besoin de viscosité artificielle pour atténuer d'éventuels effets dispersifs du schéma. Avec les différences finies, on utilise

une formule issue d'une analyse de stabilité faite par Cottet [15] pour l'équation de Bürgers. En dimension un, ajouter ce terme de viscosité artificielle consiste à modifier les quantités m_i, $(mU)_i$ et $(mE)_i$ de la manière suivante :

$$\begin{aligned}
m_i &\leftarrow m_i + \frac{dt}{dx} \sum_{j=i-1}^{i+1} C(i,j)\ [m_j - m_i]\ [U_j - U_i] \\
(mU)_i &\leftarrow (mU)_i + \frac{dt}{dx} \sum_{j=i-1}^{i+1} C(i,j)\ [(mU)_j - (mU)_i]\ [U_j - U_i] \\
(mE)_i &\leftarrow (mE)_i + \frac{dt}{dx} \sum_{j=i-1}^{i+1} C(i,j)\ [(mE)_j - (mE)_i]\ [U_j - U_i]
\end{aligned} \tag{3.5}$$

$C(i,j)$ est un coefficient dont la valeur dépend du signe de $(U_j - U_i)(j-i)$. Il est plus grand s'il y a compression entre les cellules i et j, c'est-à-dire si $(U_j - U_i)(j-i) < 0$.

3.1.5 Traitement des conditions aux limites

La question du traitement des conditions aux limites se rencontre lors de la résolution de la phase Lagrange et surtout lors du remaillage.

Pendant la résolution de la partie lagrangienne des équations, il est nécessaire de connaître les conditions aux bords en vitesse et en pression pour calculer ∇p et $\nabla.(p\vec{U})$ près des bords du domaine de calcul. Ces conditions de bord sont toujours facilement disponibles : soit elles sont explicitement données, soit on peut les déduire du type de conditions aux limites considéré : par exemple, conditions de type "mur" ou conditions aux limites périodiques. Par conséquent, il n'y a pas vraiment de problème lié au traitement des conditions aux limites pour la résolution de la phase Lagrange.

Par contre, la technique que nous utilisons pour le remaillage nécessite de disposer de particules de chaque côté du point où nous voulons créer une nouvelle particule. Sur les bords du domaine, nous manquons donc d'informations pour remailler correctement. La solution que nous avons utilisée consiste à utiliser des particules "fictives". Celles-ci prolongent tous les bords du domaine sur une largeur suffisante pour qu'il soit possible, en utilisant ces particules "fictives" en addition des particules "normales", de créer par le remaillage de nouvelles particules dans toutes les cellules du domaine. Si les conditions au bord sont de type mur, ou sont périodiques, il est facile de créer des particules fictives par symétrie ou prolongement périodique. Le remaillage ne pose alors pas de problème. Si les conditions sont de type flux entrant, il faut aussi calculer le déplacement pendant le pas de temps précédent des particules fictives. Il n'est pas nécessaire de connaître les valeurs exactes de densité ou d'énergie de ces particules fictives. On peut simplement prolonger les valeurs connues à l'intérieur du domaine afin de disposer pour le remaillage d'un profil continu.

3.2 Validation numérique de la méthode

Nous commençons par étudier sur des problèmes d'advection l'ordre de convergence numérique de la méthode utilisée avec les fonctions d'interpolation Λ_2, M_4' et Λ_3. Cette analyse valide les résultats théoriques sur la précision de la méthode particulaire avec remaillage pour une équation de transport présentés dans le chapitre 1. Puis nous réalisons des cas-tests académiques : Tube à chocs de Sod, double détente de Sjögreen, problème de Noh, qui nous permettent d'analyser les capacités de la méthode à simuler correctement les phénomènes de la dynamique des gaz.

3.2.1 Evaluation de l'ordre de convergence

Si on dispose, pour un problème donné, d'une solution exacte et de solutions numériques calculées sur des grilles de tailles différentes, il est facile d'évaluer de manière empirique l'ordre de convergence de la méthode étudiée. On note u la solution exacte, u_1 et u_2 les solutions sur les grilles 1 et 2. Soit E_1 l'erreur de la solution obtenue sur la grille 1 de taille dx_1 et E_2 l'erreur de la solution obtenue sur la grille 2, de taille dx_2. L'ordre de convergence empirique p vaut :

$$p = \frac{ln(E_2/E_1)}{ln(dx_2/dx_1)}$$

Pour plus de détails sur les manières d'évaluer empiriquement l'ordre de convergence d'une méthode numérique, voir [76]. On choisit de mesurer l'écart entre les solutions numérique et la solution exacte dans les normes discrètes L^1, L^2 et L^∞ définies de la maniere suivante :

$$\begin{aligned} ||u||_{L^1} &= dx \sum_j |u_j| \\ ||u||_{L^2} &= dx \sum_j {u_j}^2 \\ ||u||_{L^\infty} &= \max_j |u_j| \end{aligned}$$

3.2.1.1 Advection linéaire uniforme d'un sinus

L'analyse théorique de l'erreur de troncature des méthodes particulaires a été réalisée pour la partie "transport + remaillage". Pour valider ces résultats théoriques, il faut donc étudier la convergence numérique de la méthode appliquée à un problème d'advection seule. Dans ce premier cas-test, on considère donc non pas les équations d'Euler, mais l'équation de transport linéaire scalaire en 1D. Comme l'analyse a été effectuée en supposant que la solution était régulière, il est préférable de commencer par considérer le cas d'une solution exacte régulière.

On choisit donc de résoudre numériquement l'équation

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 \ , \quad x \, \epsilon \, [0,1], \ t \geq 0.$$

assortie de la condition initiale

$$u(x,0) = sin(2\pi x), \quad x \,\epsilon\, [0,1]$$

avec des conditions aux limites périodiques. La solution exacte de ce problème est périodique en temps, de période $T = 1$.

Les tableaux (3.1), (3.2) et (3.3) présentent les évaluations d'erreurs au bout d'une période pour les différentes fonctions d'interpolation, et les ordres de convergence empiriques correspondants. Les solutions numériques ont été calculées avec les fonctions d'interpolation M'_4, Λ_3 et Λ_2, pour différents pas d'espace $dx = \frac{1}{N}$, avec le rapport $\lambda = \frac{dt}{dx}$ constant, égal à 0,4. L'ordre de convergence numérique est de 2 pour le schéma

	$\|\|u - u_{dx}\|\|_{L^1}$		$\|\|u - u_{dx}\|\|_{L^2}$		$\|\|u - u_{dx}\|\|_{L^\infty}$	
	erreur	ordre	erreur	ordre	erreur	ordre
N = 10	0.414	-	0.452	-	0.639	-
N = 20	0.110	1.91	0.122	1.89	0.172	1.89
N = 40	0.028	1.97	0.031	1.98	0.043	2.00
N = 80	0.0069	2.02	0.0077	2.01	0.0108	1.99

TAB. 3.1 – Advection d'un sinus, remaillage avec Λ_2 : erreur et ordre de convergence numérique.

	$\|\|u - u_{dx}\|\|_{L^1}$		$\|\|u - u_{dx}\|\|_{L^2}$		$\|\|u - u_{dx}\|\|_{L^\infty}$	
	erreur	ordre	erreur	ordre	erreur	ordre
N = 10	0.116	-	0.128	-	0.179	-
N = 20	0.021	2.47	0.023	2.48	0.032	2.48
N = 40	0.0043	2.29	0.0048	2.26	0.0068	2.23
N = 80	0.0010	2.10	0.0011	2.13	0.0016	2.09

TAB. 3.2 – Advection d'un sinus, remaillage avec M'_4 : erreur et ordre de convergence numérique.

avec Λ_2 et un peu supérieur à 2 pour M'_4, il est de 3 pour Λ_3. Ces résultats confirment notre analyse théorique (proposition 3 dans le chapitre 2), puisque Λ_2 et M'_4 conservent les 3 premiers moments, et Λ_3 les quatre premiers. On remarque par contre que le schéma avec Λ_2 produit des erreurs bien plus importantes que celui avec M'_4, quelle que soit la discrétisation N, et ce, bien que la solution exacte soit très régulière. Cela confirme le fait que ce noyau d'interpolation est très dispersif.

La figure (3.4) montre les résultats des calculs au bout d'une période, pour les trois noyaux d'interpolation, avec une discrétisation spatiale $N = 40$. On constate à nouveau que Λ_2 donne de moins bons résultats que les deux autres. En particulier, la solution

	$\|\|u-u_{dx}\|\|_{L^1}$		$\|\|u-u_{dx}\|\|_{L^2}$		$\|\|u-u_{dx}\|\|_{L^\infty}$	
	erreur	ordre	erreur	ordre	erreur	ordre
N = 10	0.098	-	0.112	-	0.151	-
N = 20	0.014	2.81	0.015	2.90	0.022	2.78
N = 40	0.0017	3.04	0.0019	2.98	0.0027	3.03
N = 80	0.00022	2.95	0.00024	2.98	0.00034	2.99

TAB. 3.3 – Advection d'un sinus, remaillage avec Λ_3 : erreur et ordre de convergence numérique.

numérique obtenue avec ce noyau est légèrement décalée par rapport à la solution exacte, ce qui constitue une preuve supplémentaire du caractère dispersif du noyau. Les résultats graphiques avec les noyaux M'_4 et Λ_3 sont, quant à eux, graphiquement presque identiques.

3.2.1.2 Advection linéaire uniforme d'une gaussienne

Pour confirmer ces résultats on s'intéresse maintenant à un cas plus raide : l'advection d'une gaussienne. Le problème est constitué de la même équation que dans la sous-section précédente, avec la condition initiale :

$$u(x,0) = \frac{1}{2} + \frac{3}{2}e^{-16(x-2)^2} \quad x \in [0, 2.[$$

Le rapport $\lambda = \frac{dx}{dt}$ est égal à 0.4 pour tous les calculs.

	$\|\|u-u_{dx}\|\|_{L^1}$		$\|\|u-u_{dx}\|\|_{L^2}$		$\|\|u-u_{dx}\|\|_{L^\infty}$	
	erreur	ordre	erreur	ordre	erreur	ordre
N = 10	0.356	-	0.336	-	0.506	-
N = 20	0.120	1.57	0.130	1.37	0.238	1.09
N = 40	0.031	1.95	0.036	1.85	0.066	1.85
N = 80	0.0079	1.97	0.0091	1.98	0.016	2.04

TAB. 3.4 – Advection d'une gaussienne avec Λ_2 : erreur et ordre de convergence numérique.

Les tableaux (3.4) à (3.6) présentent les évaluations d'erreur des solutions numériques obtenues pour les différentes fonctions d'interpolation, ainsi que les ordres de convergence empiriques correspondants. Ces derniers sont cohérents avec ceux obtenus précédemment pour l'advection d'un sinus. Les ordres de convergence calculés avec l'erreur pour la discrétisation N = 10 sont plus faibles que les autres, sans doute parce que cette discrétisation ne permet pas de représenter correctement la gaussienne. Les ordres de convergence de Λ_2 et Λ_3 valent à nouveau respectivement 2 et 3, alors que celui de M'_4 reste supérieur à 2. A nouveau, les résultats graphiques (voir figure 3.5) confirment

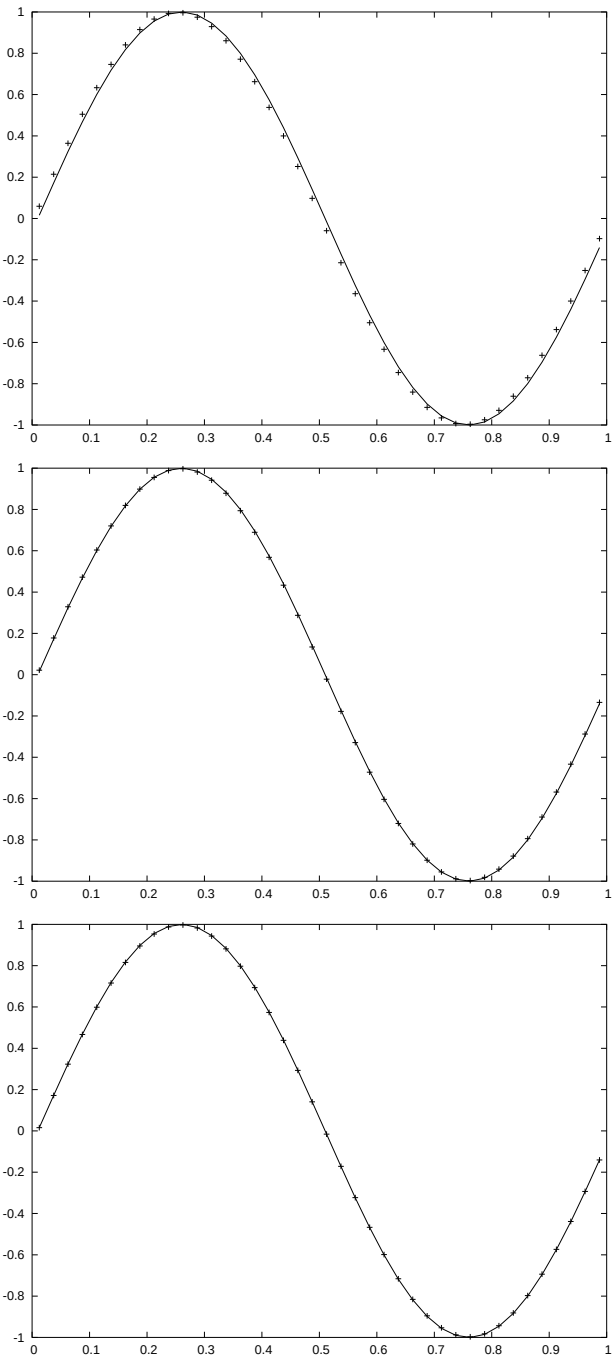


FIG. 3.4 – Transport à vitesse uniforme d'un sinus avec de haut en bas Λ_2, M_4' et Λ_3, avec 40 points.

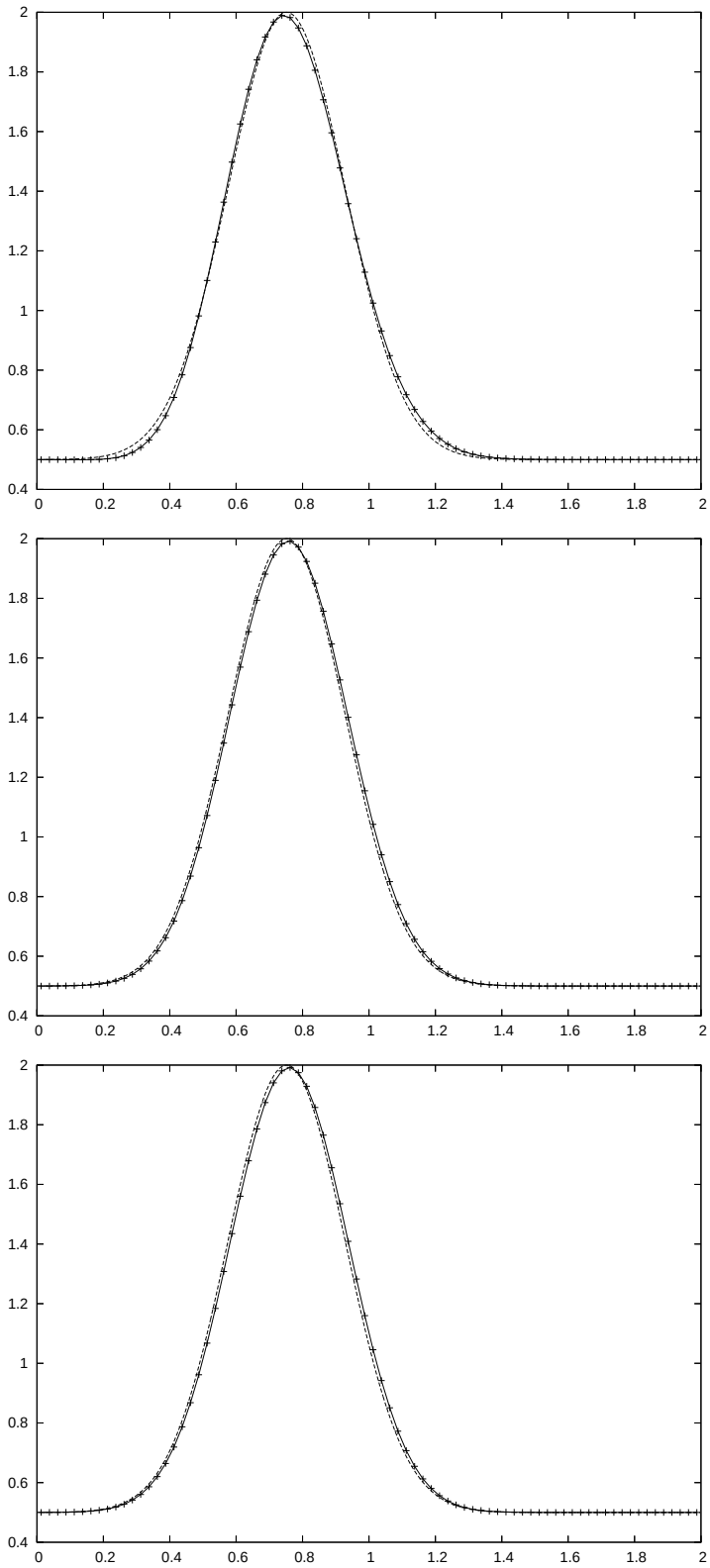


FIG. 3.5 – Advection d'une gaussienne, avec de haut en bas Λ_2, M'_4 et Λ_3, 40 points.

	$\|\|u-u_{dx}\|\|_{L^1}$		$\|\|u-u_{dx}\|\|_{L^2}$		$\|\|u-u_{dx}\|\|_{L^\infty}$	
	erreur	ordre	erreur	ordre	erreur	ordre
N = 10	0.129	-	0.132	-	0.219	-
N = 20	0.028	2.20	0.032	2.04	0.059	1.89
N = 40	0.0054	2.37	0.0063	2.34	0.0118	2.32
N = 80	0.0012	2.17	0.00138	2.19	0.0025	2.24

TAB. 3.5 – Advection d'une gaussienne avec M_4' : erreur et ordre de convergence numérique.

	$\|\|u-u_{dx}\|\|_{L^1}$		$\|\|u-u_{dx}\|\|_{L^2}$		$\|\|u-u_{dx}\|\|_{L^\infty}$	
	erreur	ordre	erreur	ordre	erreur	ordre
N = 10	0.120	-	0.123	-	0.205	-
N = 20	0.024	2.33	0.026	2.24	0.052	1.97
N = 40	0.0033	2.87	0.0038	2.77	0.0078	2.73
N = 80	0.00042	2.97	0.00048	2.92	0.0010	2.96

TAB. 3.6 – Advection d'une gaussienne avec Λ_3 : erreur et ordre de convergence numérique.

que Λ_2 donne de moins bons résultats que les deux autres noyaux. On remarque en particulier l'erreur de phase de part et d'autre de la base de la gaussienne.

3.2.1.3 Advection linéaire uniforme d'une discontinuité

Les tests de convergence précédents nous ont montré que le transport de profils réguliers avec la méthode particulaire fonctionne bien (surtout avec M_4' ou Λ_3). Qu'en est-il lorsque le profil est discontinu ? Les calculs qui nous ont permis de prédire l'ordre des différents schémas ont été menés sous l'hypothèse que la solution était continue. Par conséquent, il n'est pas du tout évident que nous obtenions les mêmes résultats quand la solution présente des discontinuités. Pour le savoir, on teste à nouveau la résolution numérique par la méthode de l'équation de transport linéaire, mais avec cette fois un créneau pour condition initiale :

$$\begin{aligned} u(x,0) &= 1 \text{ si } x \,\epsilon\, [0.25, 0.75] \\ &= -1 \text{ sinon} \end{aligned}$$

et des conditions aux limites périodiques. Ce test est effectué avec M_4' seulement, par souci de brièveté, et parce que d'après les tests précédents ce noyau semble donner de bons résultats.

Le tableau (3.7) présente les erreurs en norme L^1 et L^2 de la solution numérique pour différentes discrétisations, et les ordres de convergence empiriques correspondants. La

solution semble converger dans ces normes, mais bien plus lentement que dans le cas d'une solution exacte régulière. Nous n'avons pas calculé l'ordre de convergence pour la norme L^∞, parce qu'il est très visible, d'après les valeurs des erreurs, que le schéma ne converge pas dans cette norme.

On constate sur la figure (3.6) que la solution numérique présente des oscillations assez marquées autour des discontinuités. Celles-ci ont nécessairement été créées par le remaillage, car c'est la seule source d'erreur du schéma. Pour limiter ces oscillations, nous essayons une technique que nous appellons "seuillage", et qui consiste à empêcher l'apparition de nouveaux maxima ou minima locaux pendant le remaillage :

A chaque cellule i sont affectées deux valeurs : le maximum max_i et le minimum min_i des valeurs des anciennes particules qui servent à créer une nouvelle particule dans cette cellule i. Si la valeur de la particule créée sur la cellule i est supérieure à max_i, alors la particule prend comme nouvelle valeur max_i. De même, si la valeur de la particule créée sur la cellule i est inférieure à min_i, alors la particule prend comme nouvelle valeur min_i.

Cette technique n'est pas conservative, il peut donc être gênant de l'appliquer aux variables des équations d'Euler. Par contre, si elle fonctionne, on pourra l'utiliser sans état d'âme sur une variable dont on n'a pas besoin de s'assurer la conservativité. C'est notamment le cas de la fonction ϕ que nous introduirons dans le chapitre sur la simulation des écoulements bifluides. Pour illustrer le fait qu'il n'est pas souhaitable d'utiliser le seuillage sur des variables conservatives, la figure (3.7) présente, pour le test du tube à choc de Sod, le profil de densité obtenu avec la méthode particulaire à laquelle a été appliqué le seuillage, comparé à la solution exacte. On constate que la solution numérique ne respecte pas les paliers de la solution exacte, alors que, comme on le verra dans la suite, la solution numérique obtenue avec la méthode particulaire sans seuillage respecte ces paliers. Le tableau (3.8) représente les ordres de convergence du schéma particulaire avec seuillage. On constate que les ordres de convergence sont pratiquement les mêmes que sans seuillage. En revanche, on observe sur la figure (3.6) que la solution numérique ne présente plus d'oscillations près des discontinuités. On en conclut que le seuillage n'a pas d'effet sur l'ordre de convergence, mais permet d'éviter l'apparition d'oscillations non-physiques. Il pourra donc être intéressant de le pratiquer sur cette variable ϕ.

	$\|\|u-u_{dx}\|\|_{L^1}$		$\|\|u-u_{dx}\|\|_{L^2}$		$\|\|u-u_{dx}\|\|_{L^\infty}$	
	erreur	ordre	erreur	ordre	erreur	ordre
N = 20	0.241	-	0.355	-	0.855	-
N = 40	0.148	0.70	0.275	0.37	0.858	-
N = 80	0.091	0.70	0.213	0.37	0.854	-
N = 160	0.054	0.75	0.166	0.36	0.845	-

TAB. 3.7 – Advection d'une discontinuité avec M_4' : Ordre de convergence numérique.

	$\|\|u-u_{dx}\|\|_{L^1}$		$\|\|u-u_{dx}\|\|_{L^2}$		$\|\|u-u_{dx}\|\|_{L^\infty}$	
	erreur	ordre	erreur	ordre	erreur	ordre
N = 20	0.235	-	0.373	-	0.839	-
N = 40	0.141	0.74	0.289	0.37	0.837	-
N = 80	0.085	0.73	0.225	0.36	0.831	-
N = 160	0.051	0.74	0.175	0.36	0.889	-

TAB. 3.8 – Advection d'une discontinuité avec M_4', seuillage : Ordre de convergence numérique.

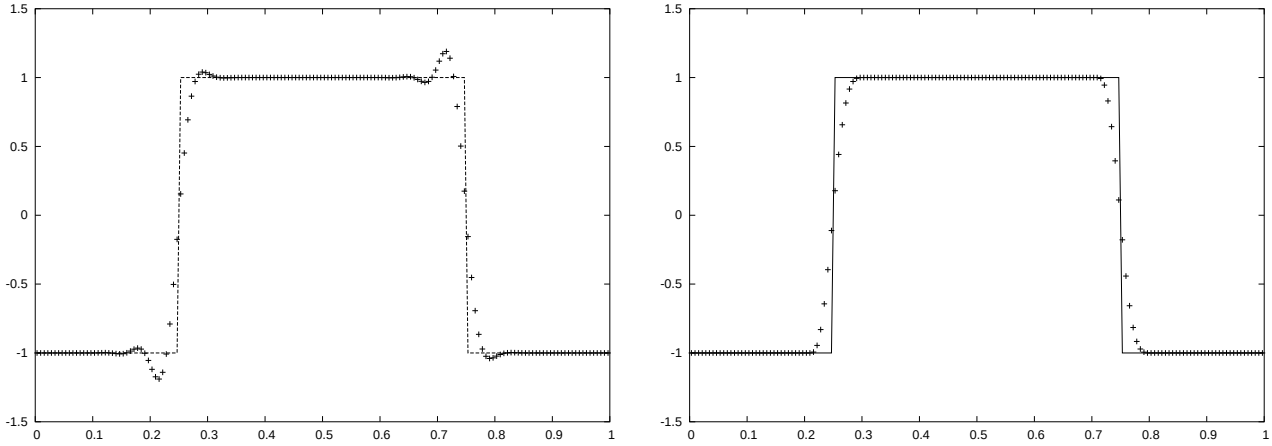


FIG. 3.6 – Transport à vitesse uniforme d'un créneau avec la fonction M_4', N= 160, sans seuillage (gauche), et avec seuillage (droite).

3.2.2 Convergence vers la solution entropique ?

Nous observons ici le comportement des schémas particulaires quand nous résolvons l'équation de Bürgers avec pour donnée initiale

$$\begin{aligned} u_0(x) &= 0 \text{ si } x \in [0, 0.5] \\ &= 1 \text{ si } x \in]0.5, 1] \end{aligned}$$

et des conditions aux limites périodiques.

Le schéma de Lax-Wendroff converge dans ce cas vers une solution faible de l'équation de Bürgers qui n'est pas la solution entropique. Sur la figure (3.8), on constate en effet que le schéma fait apparaître un choc non-entropique au lieu de la détente attendue. Sur la figure (3.9), la position et l'amplitude du choc obtenu avec Lax-Wendroff ne sont pas correctes. En revanche, les schémas avec Λ_2, M_4' et Λ_3 semblent, malgré un défaut au pied de la détente, converger vers la solution entropique. Sans surprise, c'est

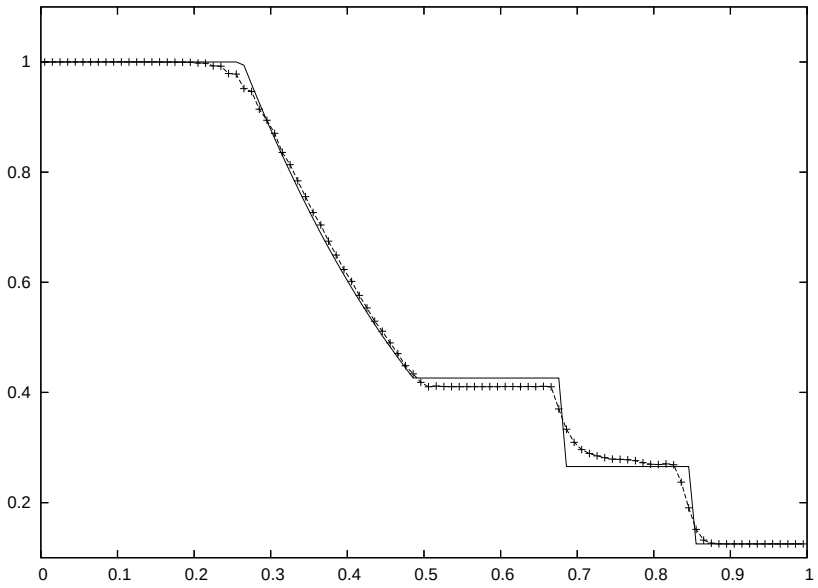


FIG. 3.7 – Tube à choc de Sod, solution numérique obtenue avec la méthode particulaire utilisée avec le seuillage, comparée à la solution exacte.

le schéma avec Λ_2 qui présente le défaut le plus marqué, puis, dans l'ordre, M_4' et Λ_3. Au temps $t = 1.5$, on voit clairement que les trois chémas particulaires convergent vers la solution entropique, alors que la solution obtenue avec le schéma de Lax-Wendroff a un profil complètement différent.

Nous ne savons pas pour le moment expliquer les différences de comportement entre le schéma de Lax-Wendroff et le schéma particulaire avec Λ_2, qui ont pourtant, comme on l'a vu dans la section 2.2.3.2 du chapitre 1, des équations équivalentes très proches. Plus généralement, nous ne savons pas s'il est possible de prouver que les schémas particulaires convergent vers la solution entropique. La tâche est rendue difficile par le fait que les schémas du type (2.6), construits avec les fonctions de remaillage que nous utilisons habituellement, ne sont pas monotones ni même TVD.

Nous avons également testé le terme de viscosité artificielle étudié dans la section 2.2.6 du chapitre 2 :

$$C(|u_{j+1} - u_j|(u_{j+1} - u_j) - |u_j - u_{j-1}|(u_j - u_{j-1}))$$

Nous avons pris pour le coefficient C une valeur plus petite que celle exigée par les calculs théoriques : $C = 0.2$, et nous avons ajouté ce terme aux différents schémas (Lax-Wendroff, particulaires avec Λ_2, Λ_3 et M_4').

L'ajout de ce terme change complètement le comportement du schéma de Lax-Wendroff, qui converge désormais vers la solution entropique du problème. Le compor-

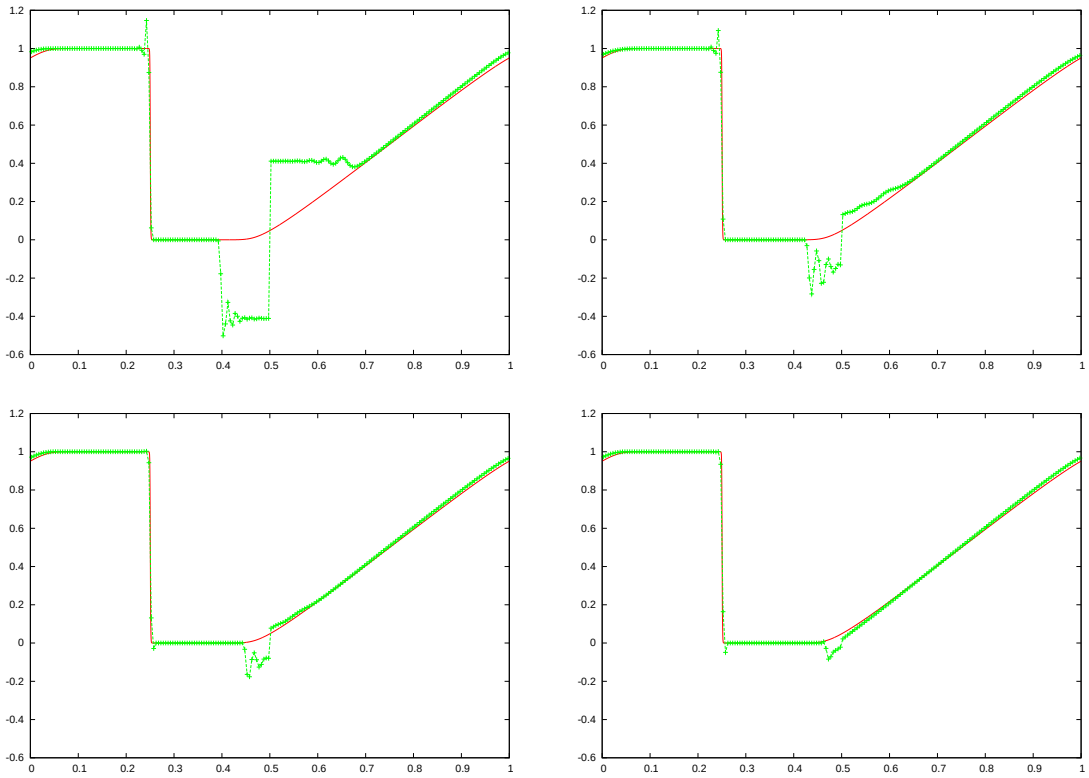


FIG. 3.8 – De haut en bas et de droite à gauche : schéma de Lax-Wendroff et schémas avec les noyaux Λ_2, M_4' et Λ_3, en vert, comparés à la solution entropique, en rouge, à t = 0.5, 200 particules ou points de grille.

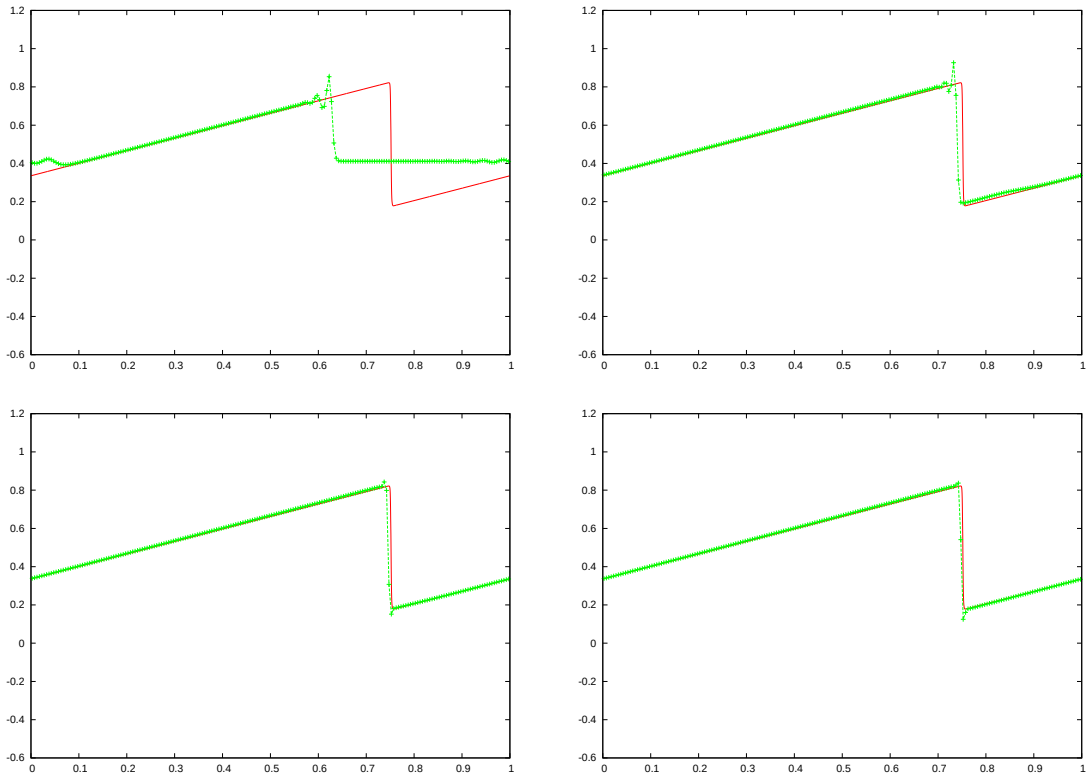


FIG. 3.9 – Même légende que la figure précédente, à $t = 1.5$.

tement des méthodes particulaires est lui aussi modifié par l'ajout de ce terme, bien que moins spectaculairement que Lax-Wendroff, puisqu'elles convergeaient déjà vers la solution entropique. Le terme de viscosité artificielle supprime le défaut qui avait été observé au pied de la détente, et, dans le cas du noyau Λ_2, diminue l'ampleur de l'overshoot qui précède le choc. Par ailleurs, il ne diminue pas de manière sensible la raideur du choc. Ces différentes observations nous confirment l'intérêt qu'il y aurait à trouver et utiliser un terme de viscosité artificielle semblable pour les équations d'Euler.

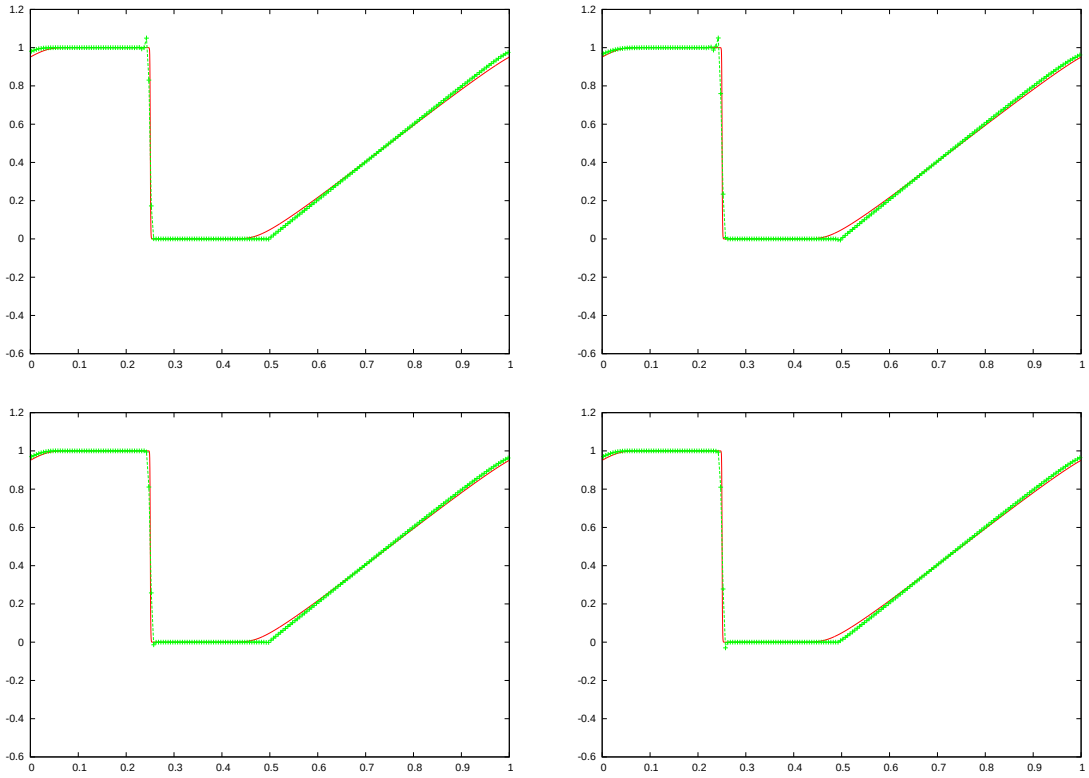


FIG. 3.10 – De haut en bas et de droite à gauche : schéma de Lax-Wendroff et schémas avec les noyaux Λ_2, M_4' et Λ_3, corrigés par le terme de viscosité artificielle étudié en 2.2.6 avec un coefficient $C = 0.2$, en vert, comparés à la solution entropique, en rouge, à t = 0.5, 200 particules ou points de grille.

3.2.3 Tube à chocs de Sod (1D)

Nous avons jusqu'à présent vérifié la convergence de la méthode particulaire sur des problèmes d'advection. Nous souhaitons maintenant étudier son comportement sur un problème représentatif de la dynamique des gaz.

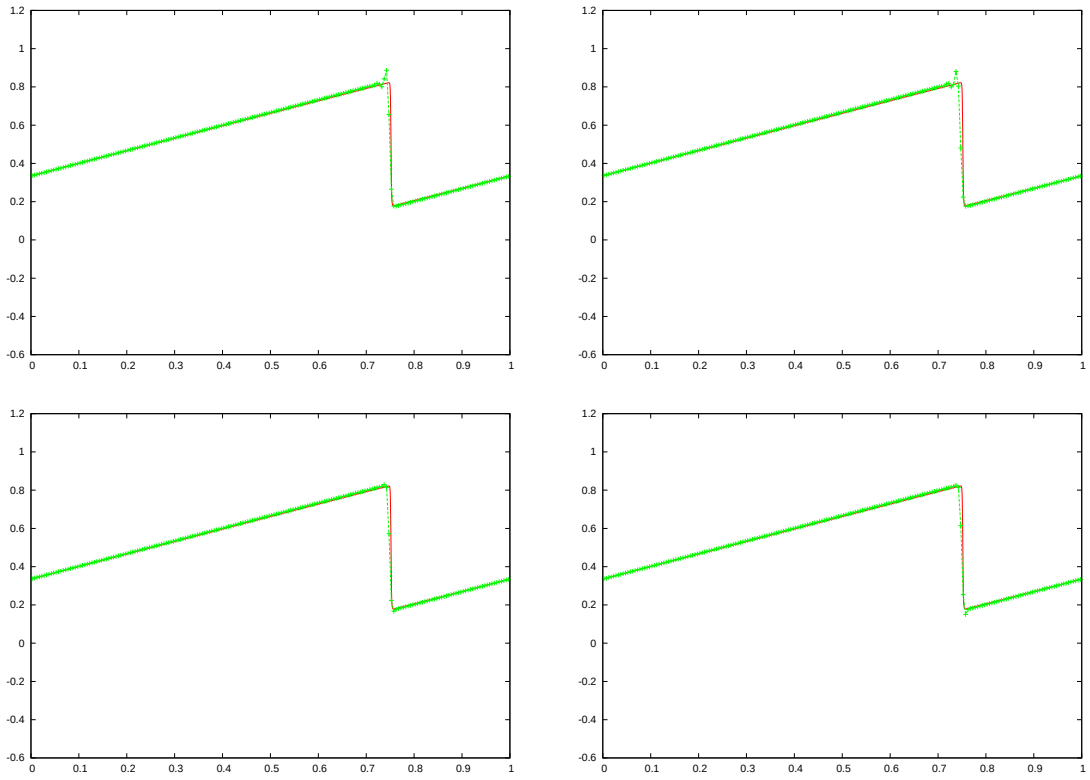


FIG. 3.11 – Même légende que la figure précédente, à $t = 1.5$.

Il s'agit du cas classique ([81]) d'un tube à choc constitué en phase initiale de deux compartiments contenant chacun un matériau assimilé à un gaz parfait ($\gamma = 1.4$). Ces deux matériaux possèdent des densités différentes. Ce problème possède une solution analytique composée des trois ondes élémentaires : détente, discontinuité de contact et choc, séparées par des états constants. Puisque sa solution fait intervenir les trois types d'ondes, ce problème permet de se faire une idée générale du comportement de la méthode pour la dynamique des gaz. Les conditions initiales sont les suivantes :

$$\begin{aligned}
\vec{U}(x,0) &= 0 \\
\rho(x,0) &= 1 \text{ si } 0 \leq x \leq 0.5 \\
&\quad\ 0.125 \text{ si } 0.5 < x \leq 1 \\
P(x,0) &= 2.5\,(\gamma-1) \text{ si } 0 \leq x \leq 0.5 \\
&\quad\ 0.25\,(\gamma-1) \text{ si } 0.5 < x \leq 1
\end{aligned}$$

Les conditions aux limites sont des conditions de type mur. Nous avons effectué les calculs avec des différences finies centrées (pour une fois, d'ordre 3), et avec le solveur de Riemann approché, et avec les noyaux d'interpolation Λ_2, M_4' et Λ_3. Dans le cas des différences finies, nous avons ajouté un terme de viscosité artificielle, dont le coefficient vaut 0.8 dans les zones de compression et 0.2 ailleurs. Dans le cas du solveur de Riemann, il n'est pas nécessaire d'utiliser ce terme de viscosité artificielle car le solveur est entropique et naturellement assez dissipatif. Les figures (3.12) à (3.17) présentent les différents résultats au temps $t = 0.2$ comparés à la solution exacte.

On remarque tout de suite que le noyau Λ_2, une fois encore, donne de bien plus mauvais résultats que les deux autres, tout particulièrement avec le solveur de Riemann approché. Ce dernier test nous convainct de ne plus utiliser ce noyau par la suite.

Le choc est un peu mieux résolu avec l'utilisation du solveur de Riemann approché qu'avec les différences finies (une à deux particules dans le choc dans le premier cas, deux à trois particules dans le second) mais il est un peu décalé par rapport à sa position exacte. Par contre, les résultats obtenus avec le solveur présentent des overshoots et des undershoots autour de la discontinuité de contact et du choc, particulièrement visibles pour l'énergie interne. Les résultats obtenus avec les différences finies centrées en présentent moins et de plus faible ampleur, sans doute grâce à l'utilisation de viscosité articifielle. La discontinuité de contact est mieux résolue avec les différences finies qu'avec le solveur de Riemann approché. Les différences finies présentent par contre des oscillations de faible ampleur à gauche de la détente dues à l'utilisation de formules centrées. Dans les deux cas, et pour tous les noyaux, la discontinuité de contact est sans doute l'élément de la solution le moins bien résolu. On conclut de cette analyse comparative que les différences finies centrées donnent pour ce test de meilleurs résultats que le solveur de Riemann approché. Les noyaux M_4' et Λ_3 donnent des résultats similaires. Il est difficile de déterminer lequel résoud le mieux ce problème.

Nous avons comparé les résultats obtenus avec les différences finies et les noyaux M_4' et Λ_3 avec ceux de

- Desprès et Mazeran (méthode de volumes finis, lagrangienne, équivalente en 1D au schéma de Godounov avec solveur acoustique, [24], N= 100),
- Loubère (méthode lagrangienne de type Galerkine discontinu, [55], N= 100),
- Cha et Whitworth (méthode SPH, [11], N=250).

Le choc et la discontinuité de contact semblent mieux résolus chez Loubère (à peine un point dans la discontinuité de contact, deux points dans le choc) ainsi que chez Desprès et Mazeran (sans qu'on puisse chiffrer le nombre de points car seule une ligne continue est tracée). Pour ces derniers, cela se fait toutefois au prix d'un overshoot assez important pour l'énergie interne. Notons que les résultats de ces auteurs sont obtenus avec des méthodes purement lagrangiennes, donc particulièrement aptes à résoudre avec précision de telles discontinuités. Le choc et la discontinuité de contact sont par contre plus étalés chez Cha et Whitworth que pour notre schéma. Les résultats de ces derniers présentent également une oscillation parasite sur le profil de pression à la position de la discontinuité de contact.

La sous-évaluation de l'extrémité gauche de la détente, commune à tous les auteurs, est moins prononcée pour notre schéma que pour tous les autres. Notre résolution de la partie droite de la détente nous a paru légèrement moins précise que celle de Loubère, et meilleure que celles de Desprès et Mazeran et de Cha et Whitworth. La capacité de notre méthode à résoudre la détente montre l'efficacité des noyaux d'interpolation que nous utilisons.

Pour résumer cette étude comparative, il semble que le schéma de Loubère donne des résultats supérieurs aux nôtres, car il fait apparaître moins d'oscillations ou d'overshoots (même s'il produit tout de même de légères oscillations juste avant le choc), et car il résoud mieux la discontinuité de contact. Le schéma de Desprès et Mazeran résoud mieux les discontinuités que le nôtre, mais semble moins bon au niveau de la détente et des overshoots. Le schéma de Cha et Whitworth nous a paru globalement moins performant.

Au moment d'implanter la méthode, notre choix a été de privilégier une résolution simple de la phase Lagrange. A ce stade de la validation numérique, il nous semble que la méthode gagnerait à utiliser des techniques un peu plus performantes pour résoudre cette étape lagrangienne, en particulier dans le cas de la résolution avec un solveur de Riemann appproché. Nous pourrions par exemple associer à ce solveur une technique de type MUSCL.

3.2.4 Cas-test de Sjogreen (1D)

Dans ce test 1D, il s'agit de simuler l'évolution de deux ondes de raréfaction qui s'éloignent l'une de l'autre ([29], [84], [87], [64]). Ce test est appelé suivant les références

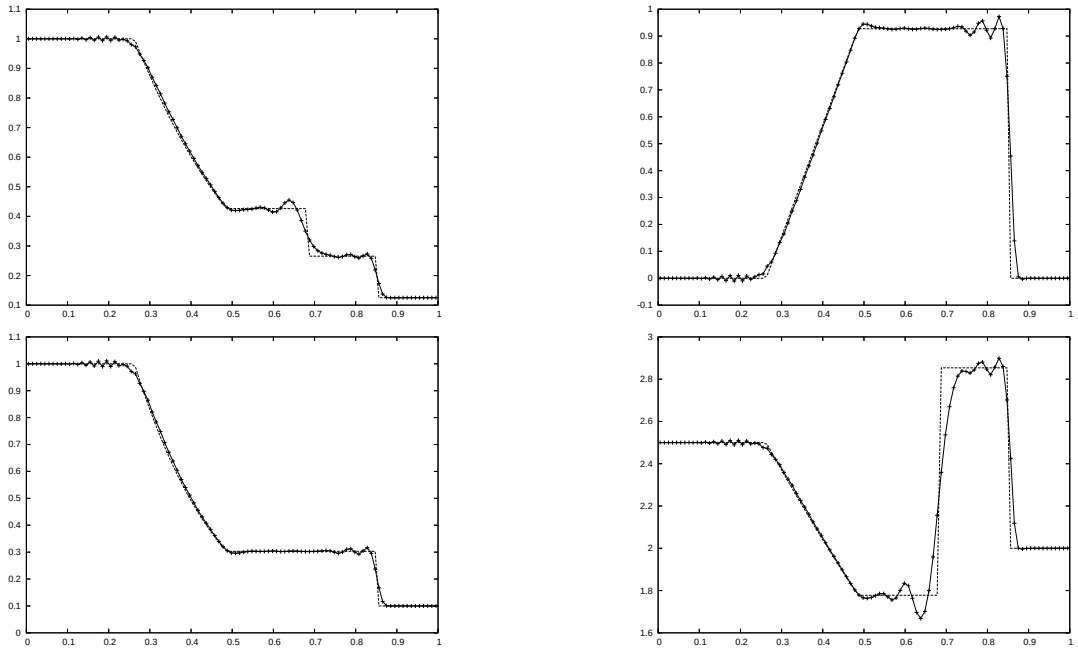


FIG. 3.12 – Tube à chocs de Sod, N=100, Λ_2, différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

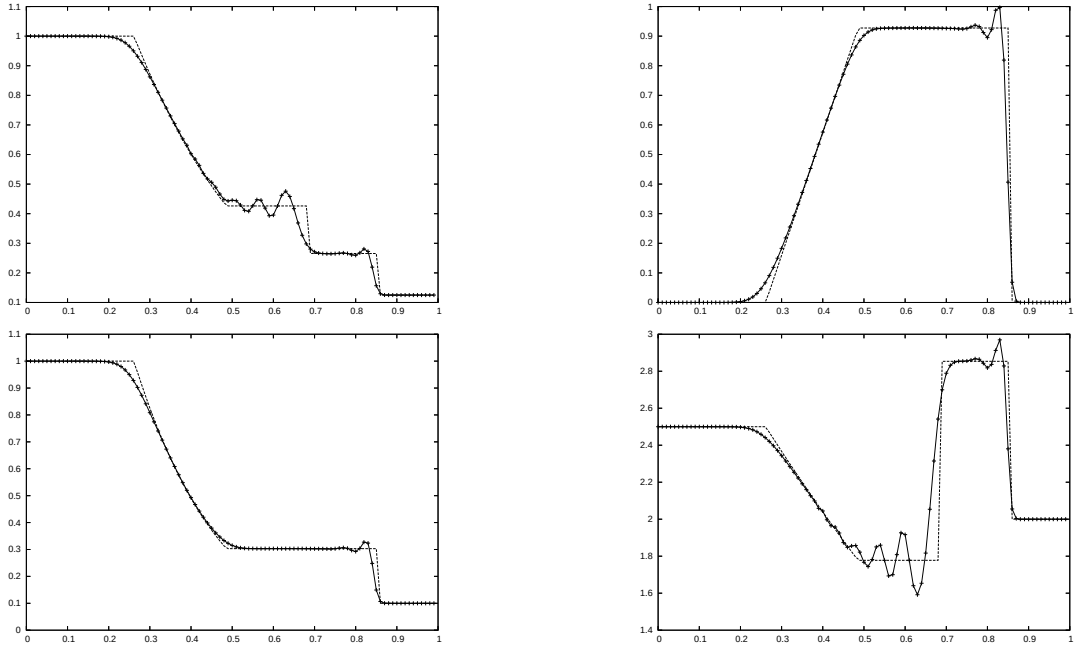


FIG. 3.13 – Tube à chocs de Sod, N=100 , Λ_2, solveur de Riemann approché, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

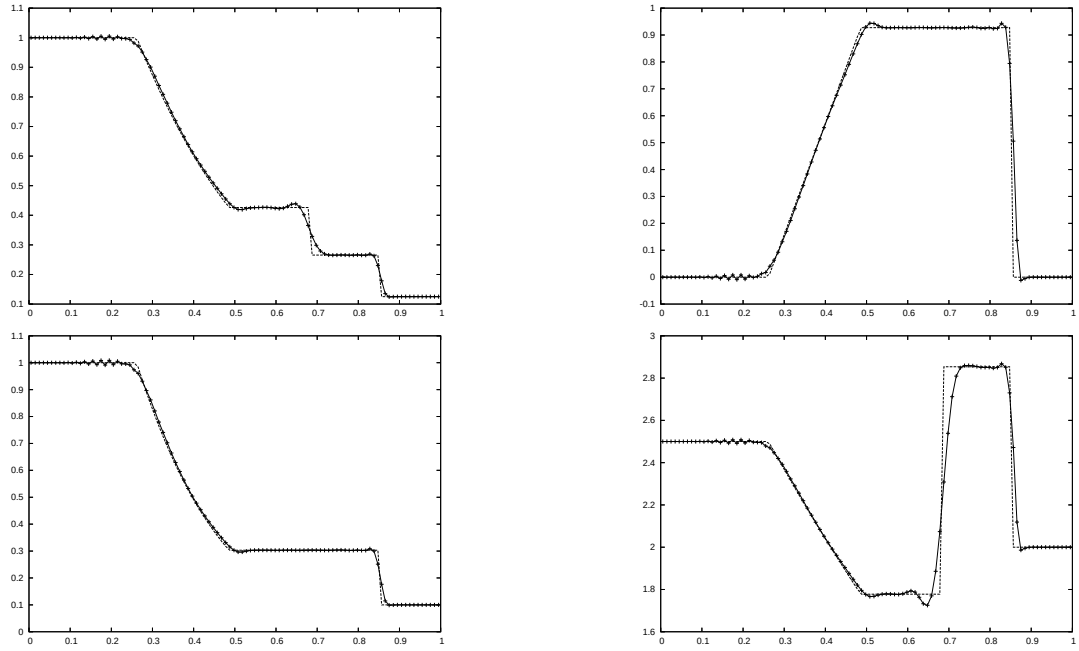


FIG. 3.14 – Tube à chocs de Sod, N=100, M_4', différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

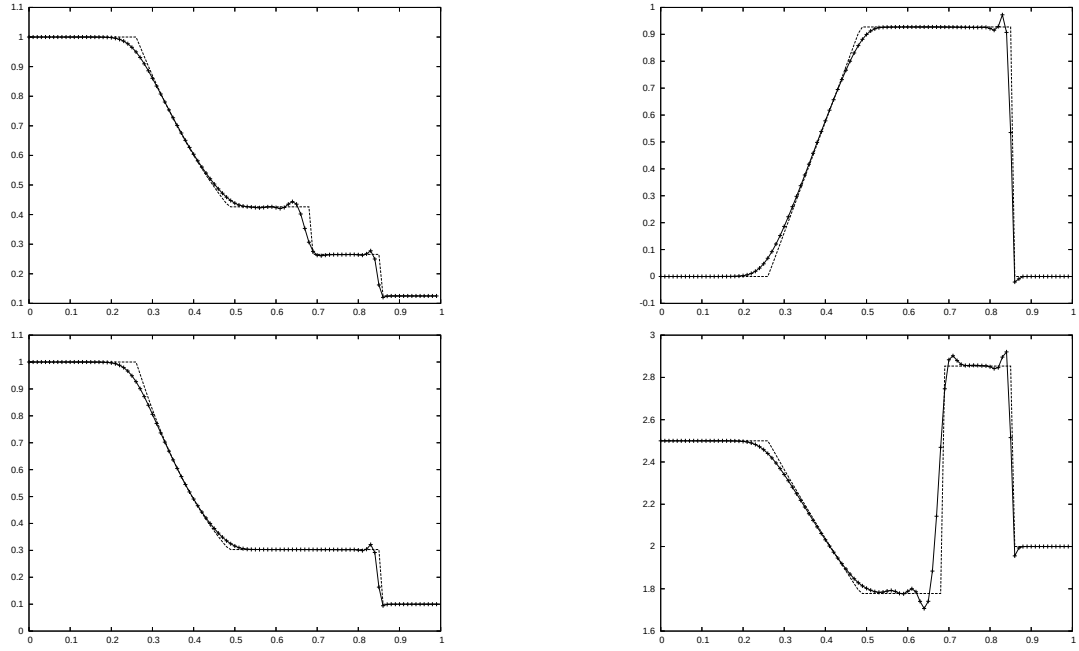


FIG. 3.15 – Tube à chocs de Sod, N=100, M_4', solveur de Riemann approché, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

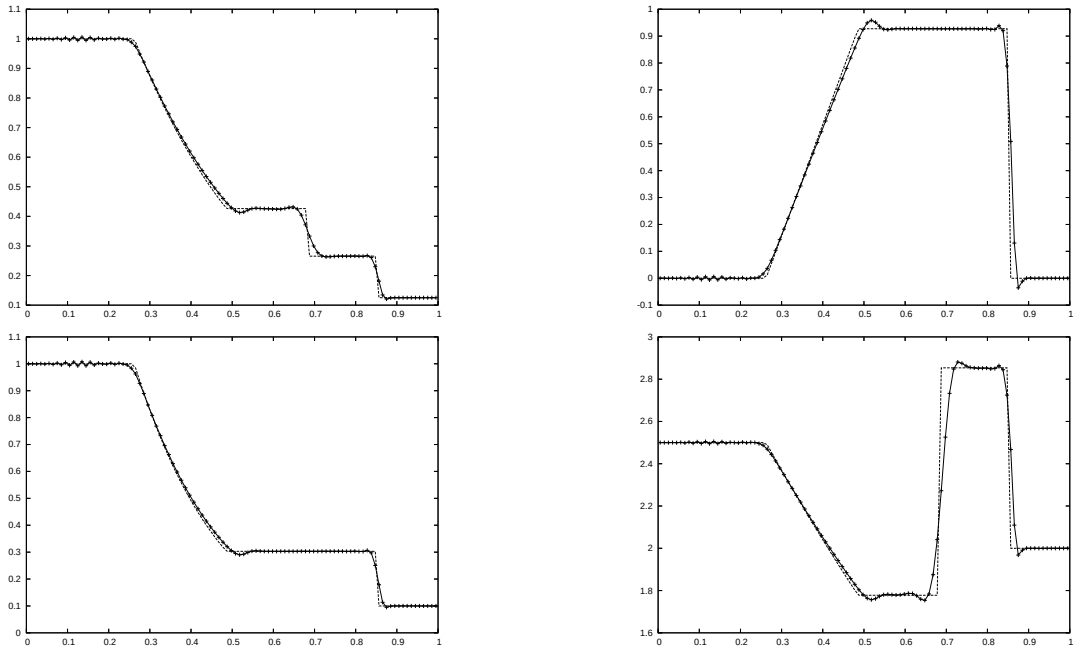


FIG. 3.16 – Tube à chocs de Sod, 100 particules, Λ_3, différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

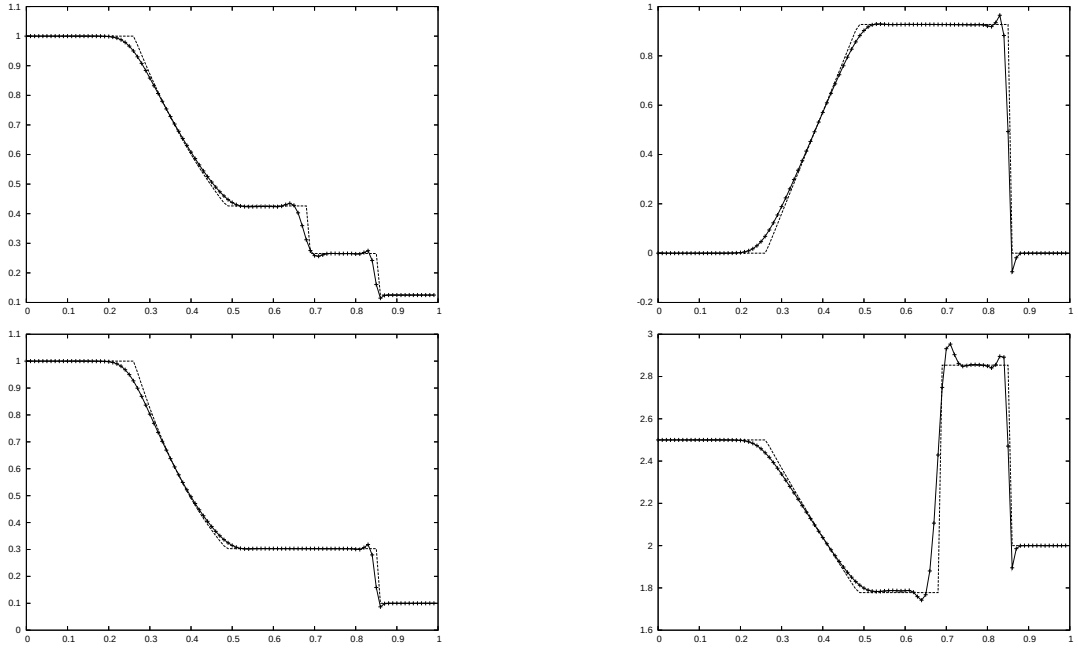


FIG. 3.17 – Tube à chocs de Sod, 100 particules, Λ_3, solveur de Riemann approché, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

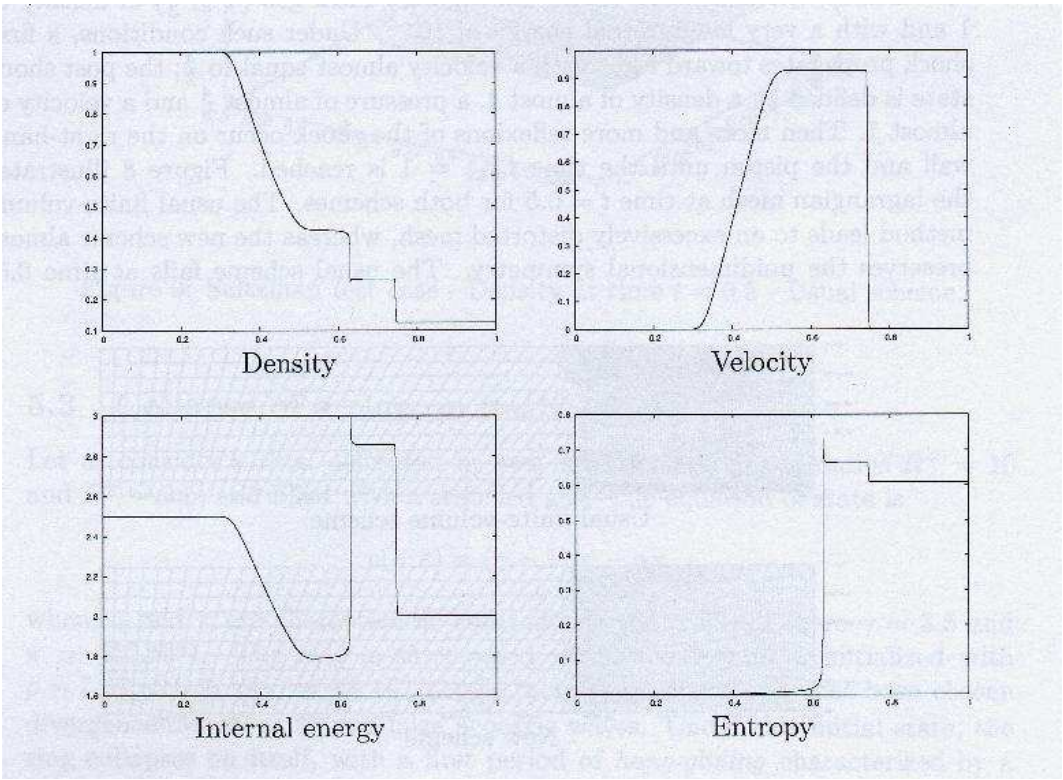


FIG. 3.18 – Résultats de Després et Mazeran [24] pour le tube à chocs de Sod, N =100.

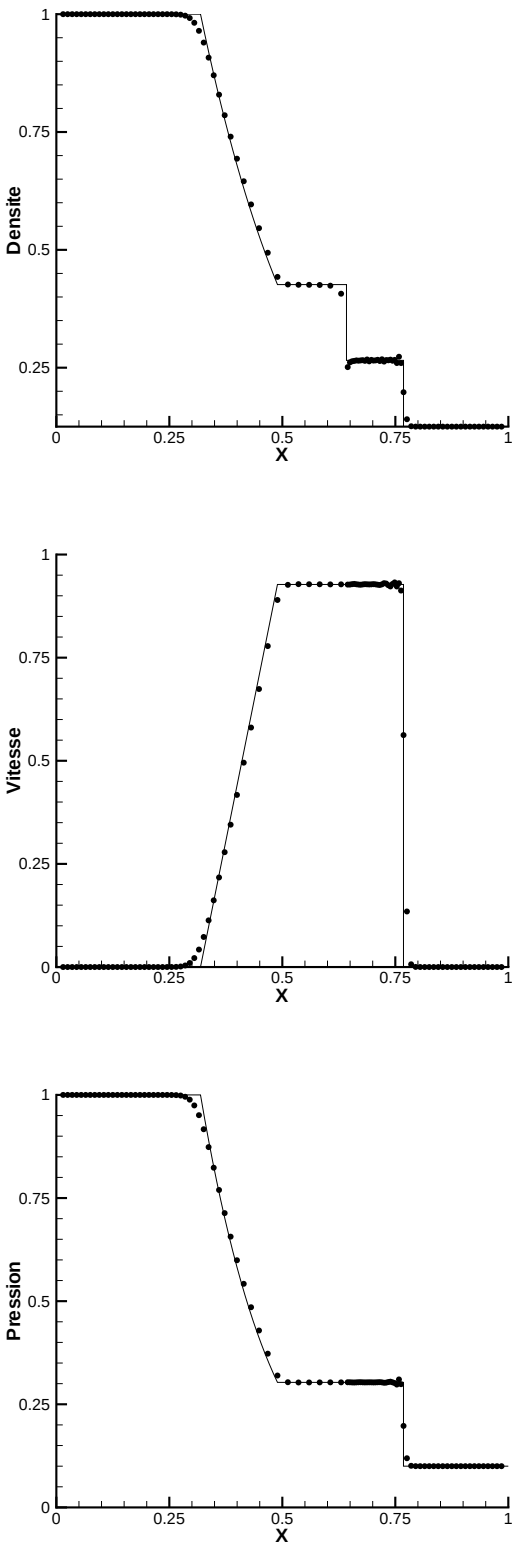


FIG. 3.19 – Résultats de Loubère ([55]) pour le tube à chocs de Sod, N =100.

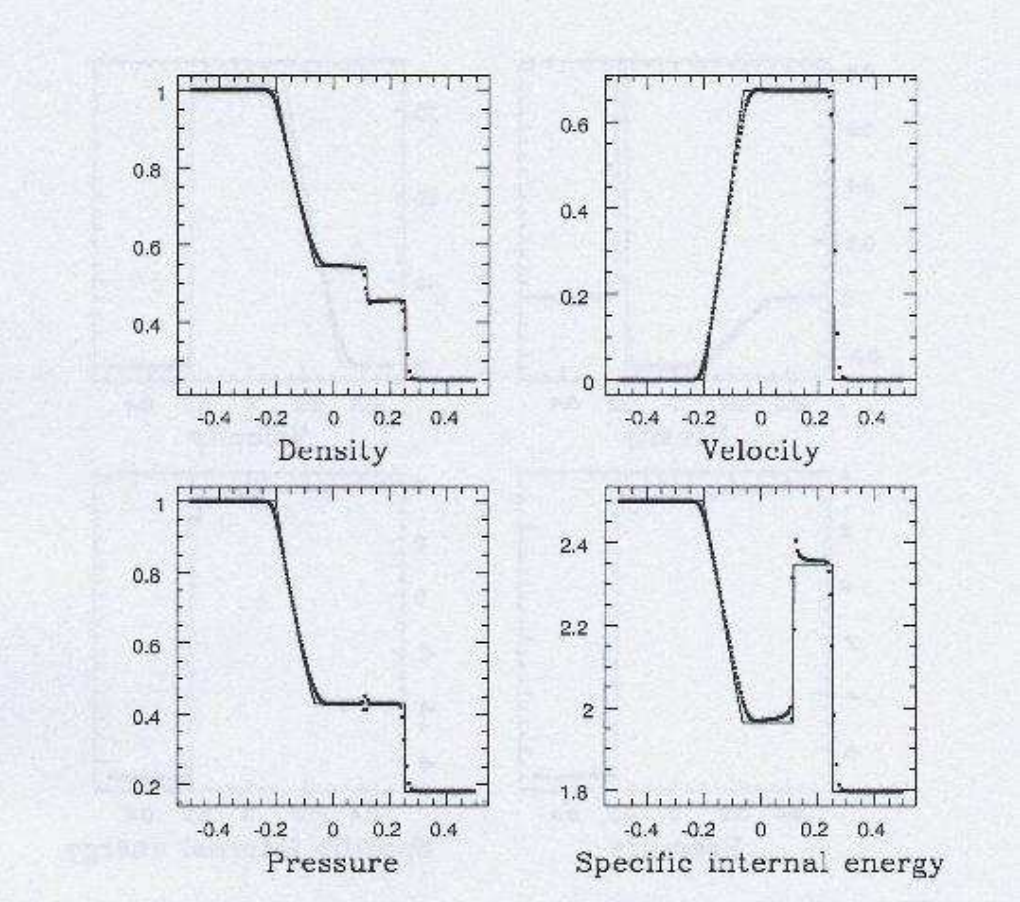


FIG. 3.20 – Résultats de Cha et Whitworth [11] pour le tube à chocs de Sod, N =250.

"test 123" ([84]) ou "test de Sjögreen" ([64]). Les conditions initiales sont les suivantes :

$$\begin{aligned} u(x,0) &= -2. \text{ si } x \leq 0.5 \\ &= 2. \text{ si } x > 0.5 \\ \rho(x,0) &= 1. \\ P(x,0) &= 0.4 \end{aligned}$$

On prend $\gamma = 1.4$ et on observe le résultat au temps $t = 0.18$. Les figures (3.21) et (3.22) présentent les résultats obtenus avec des différences finies, avec un terme de viscosité artificielle, dont le coefficient vaut 1.6 dans les zones de compression et 0.4 ailleurs. Nous ne présentons pas les résultats des calculs obtenus avec le solveur de Riemann, car ceux-ci n'ont pu être menés jusqu'à la fin du temps de simulation. Nous avons effectué les calculs avec M_4' seulement, puisque ce noyau et le noyau Λ_3 ont jusqu'à présent donné des résultats très semblables.

Les profils de la densité et de la pression sont correctement rendus par la méthode. La vitesse n'est pas très bien évaluée au centre : on devrait y observer un palier (correspondant à une zone de vide, de vitesse nulle). L'énergie interne devrait atteindre son minimum au centre du domaine de calcul, mais ce n'est pas le cas. On y observe au contraire un maximum local. Il y a sur-évaluation de l'énergie interne.

Nous avons comparé nos résultats avec ceux de :

- Monaghan (méthode SPH, [64], N = 200),
- Loubère (méthode lagrangienne de type Galerkin discontinu, [56], N= 100),
- Cha et Whitworth (méthode SPH, [11], N=250),
- Sun et Takayama (méthode de "flux vector splitting", [82], N=100).

Notons tout d'abord que les trois premières méthodes auxquelles nous avons comparé nos résultats, sont lagrangiennes et ne comportent donc que peu d'éléments de discrétisation dans la zone de raréfaction. Cela peut ici être considéré comme un avantage, car la zone située autour du centre du domaine de calcul, qui ne contient pratiquement que du vide, est très difficile à représenter et traiter numériquement. Les schémas de Monaghan et Loubère présentent aussi une sur-évaluation de l'énergie interne, mais moins forte que la nôtre. Leurs éléments de discrétisation, même s'ils sont peu nombreux au centre, y sont en adéquation avec la solution exacte (excepté pour l'énergie interne). Il y a vraiment très peu d'élements de discrétisation au centre chez Cha et Whitworth. Il nous est donc difficile de comparer nos résultats aux leurs. Notons tout de même que leur schéma, ainsi que le schéma de Sun, sur-évalue l'énergie interne plus que le nôtre. Le schéma de Sun semble également mal évaluer la vitesse au centre, au point de ne même pas y faire apparaître une ébauche de palier, comme c'est le cas avec notre méthode.

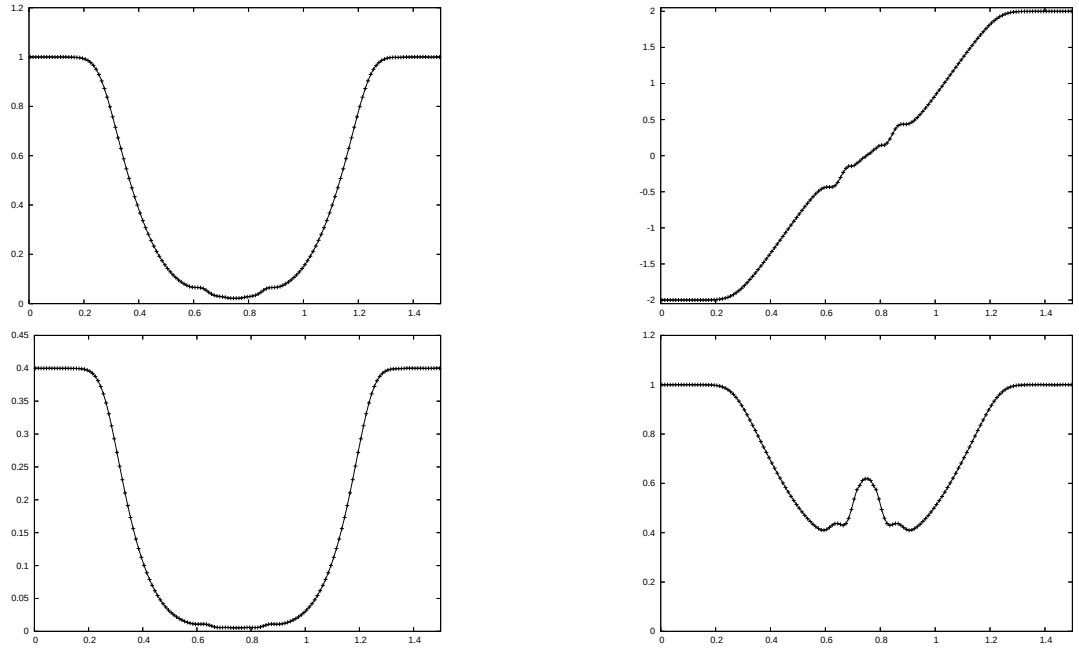


FIG. 3.21 – Test de Sjögreen, N = 100, M_4', différences finies centrées, de haut en bas et de gauche à droite : densité, vitesse, pression et énergie interne.

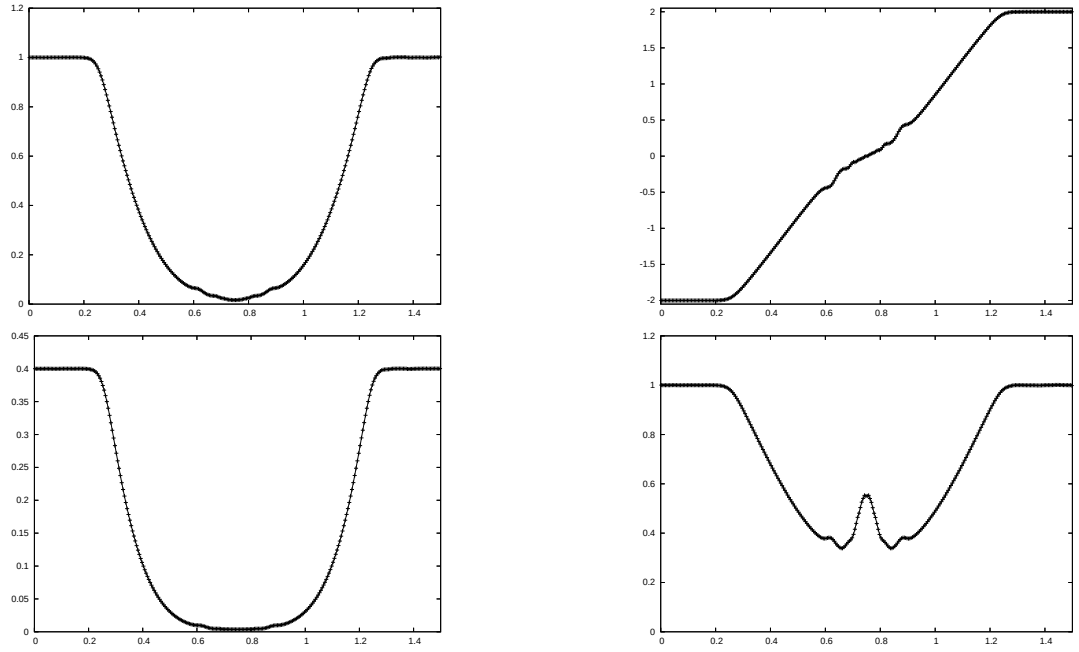


FIG. 3.22 – Même légende que la figure précédente, avec N = 200.

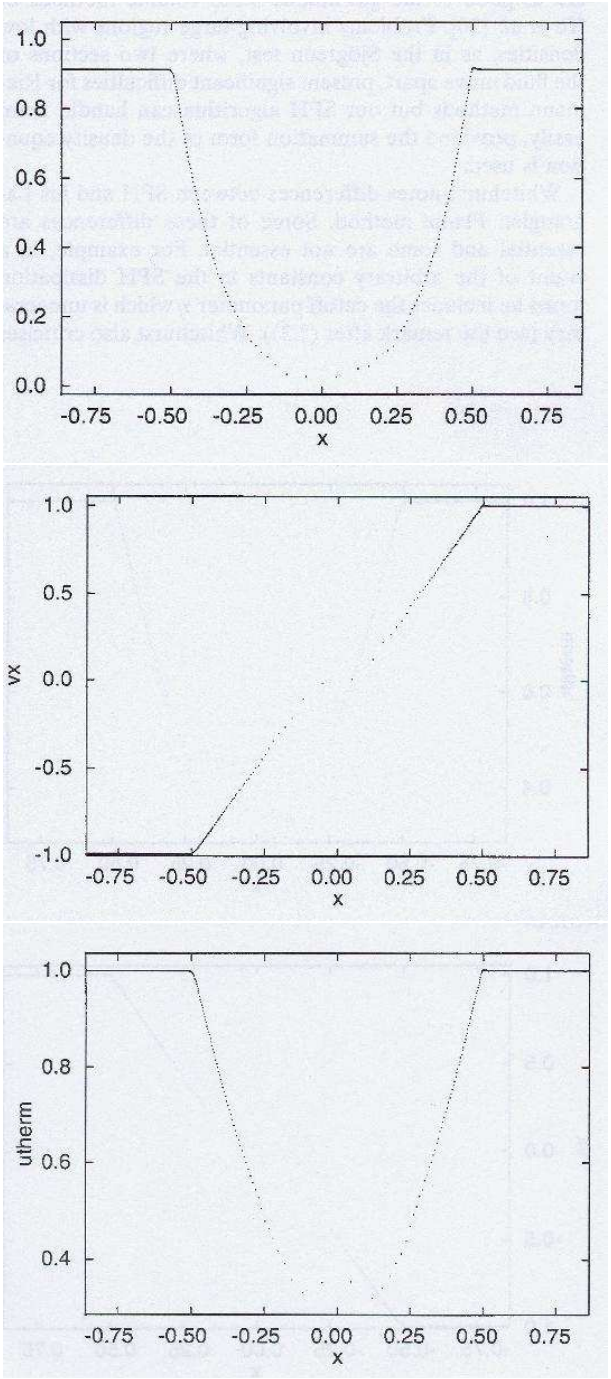


FIG. 3.23 – Résultats de Monaghan ([64]) pour le test de Sjögreen, N =200, de haut en bas : densité, vitesse, énergie interne.

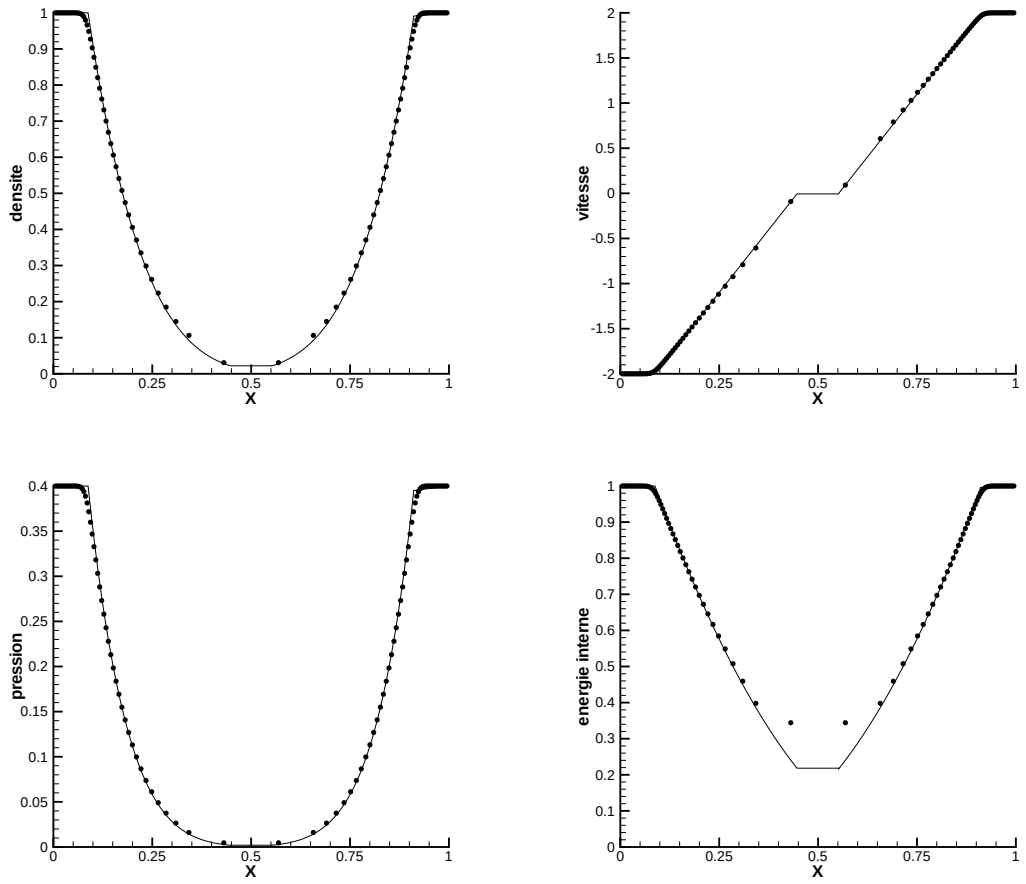


FIG. 3.24 – Résultats de Loubère ([55]) pour le test de Sjögreen, N =100.

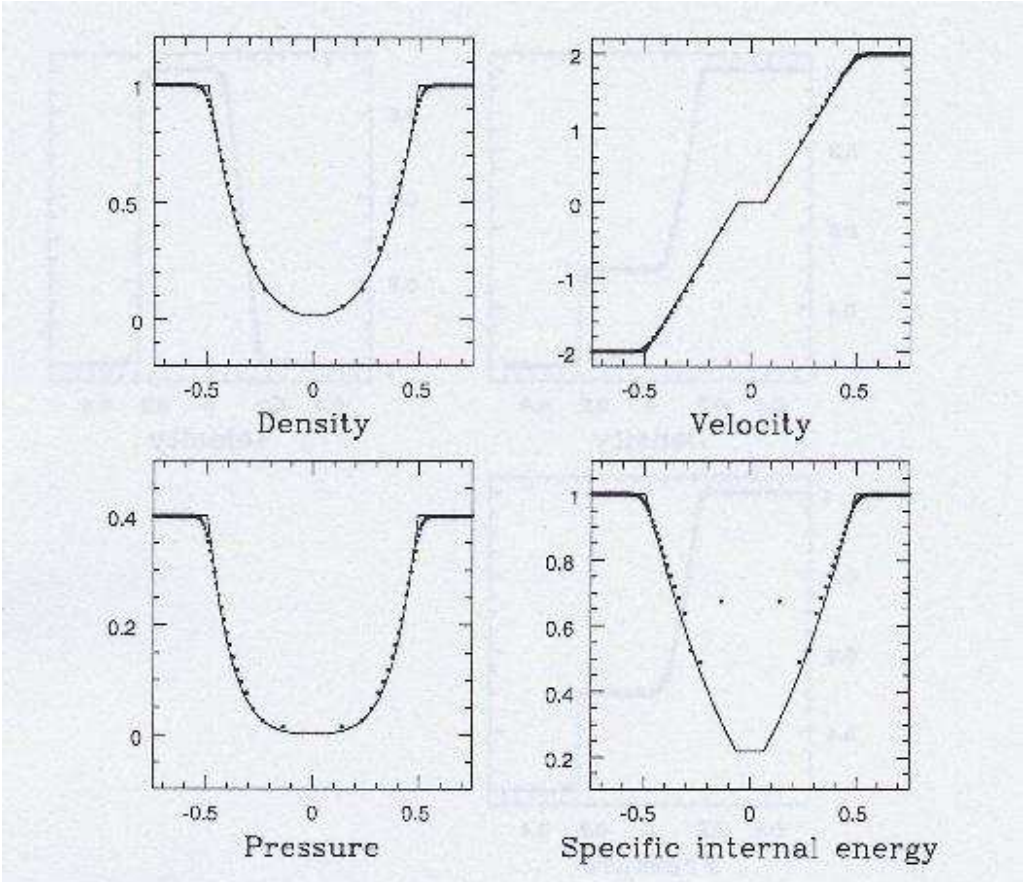


FIG. 3.25 – Résultats de Cha et Whitworth [11] pour le test de Sjögreen , N =250.

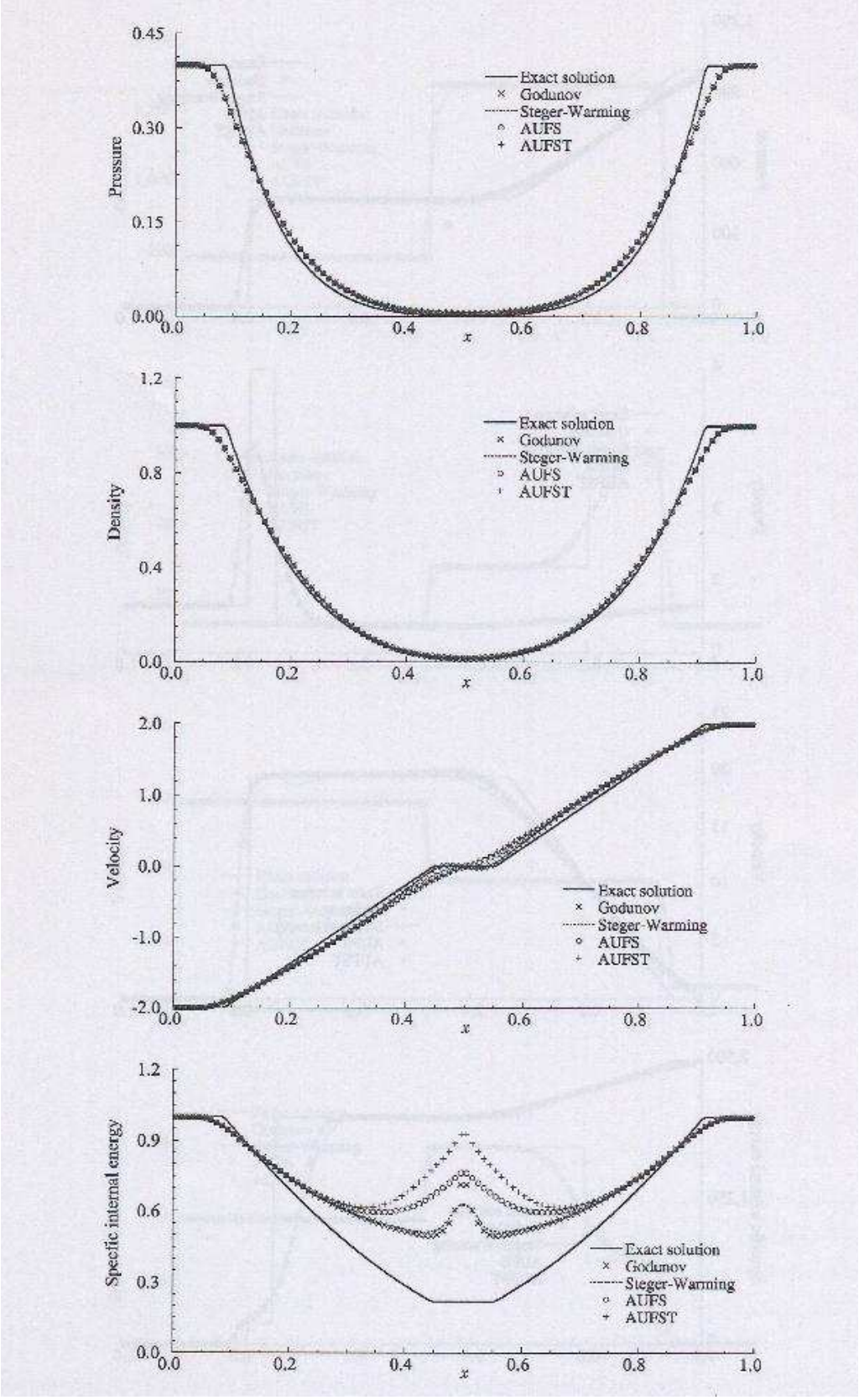


FIG. 3.26 – Résultats de Sun [82] pour le test de Sjögreen, N =100.

3.2.5 Problème de Noh (2D)

On s'intéresse maintenant à un problème 2D difficile à résoudre ([67]). On considère l'implosion d'un cylindre de rayon unité. Ce cylindre est constitué d'un gaz parfait monoatomique ($\gamma = \frac{5}{3}$). L'état initial est défini par :

$$\begin{aligned} \vec{U}(x,0) &= -\vec{e_R} \text{ avec } \vec{e_R} \text{ le vecteur unitaire radial} \\ \rho(x,0) &= 1 \\ p(x,0) &= 0 \end{aligned}$$

Ce problème admet une solution autosemblable qui consiste en l'expansion d'un choc cylindrique de force infinie à vitesse constante $V = \frac{1}{3}$.

L'initialisation effective du calcul est réalisée avec une pression $P = 10^{-3}$. On impose la vitesse normale et une pression $P^* = P$ sur le bord extérieur, initialement situé en $R = 1$.

3.2.5.1 Résolution avec grille polaire

Les figures (3.27) à (3.30) présentent les coupes des résultats obtenus à $t = 0.6$, avec M_4' et Λ_3, pour des différences finies centrées ou le solveur de Riemann approché, avec une discrétisation dr $= 1/200$, d$\theta = 2\pi/200$, comparés à la solution exacte. Dans le cas des différences finies, on ajoute un terme de viscosité artificielle, dont le coefficient vaut 1.6 dans les zones de compression et 0.4 ailleurs. De gauche à droite et de haut en bas sont représentées la densité, la vitesse, l'énergie interne et la pression.

A nouveau M_4' et Λ_3 donnent des résultats semblables. Pour les différences finies centrés, on remarque que Λ_3 crée des oscillations d'ampleur un peu supérieures à celles de M_4'. Pour le solveur, on note juste que la particule la plus proche du centre prend dans le cas de Λ_3 des valeurs différentes de celles avec M_4'.

Le choc est assez raide. Sa position n'est pas très bien résolue, puiqu'elle n'est pas centrée sur celle de la solution exacte. Cependant, quand on augmente le nombre de particules, on constate qu'elle converge vers la bonne position. On remarque aussi que la densité est sous-évaluée près du centre. Ce phénomène est plus visible pour le solveur de Riemann approché, mais est présent aussi pour les différences finies centrées, bien que caché sous les oscillations. A l'inverse, l'énergie interne est sur-évaluée. Cet effet, connu sous le nom de "wall-heating", est habituellement caractéristique des méthodes Lagrangiennes pour ce problème. A première vue, la sous-évaluation de la densité près du centre pourrait laisser penser à une perte globale de masse, mais ce n'est pas le cas. Pour le vérifier, nous avons évalué le rapport entre la masse initiale et la masse contenue dans le disque de rayon 0.4 (pour tenir compte de la compression du fluide initial) au temps final t =0.6. Ce rapport est de 0.9999. On en conclut qu'il n'y a pratiquement pas de perte de masse. La sous-évaluation de la densité au centre est en fait compensée

par la sur-évaluation de la densité au niveau du choc, due à un étalement de ce dernier. Dans le premier chapitre, nous avions remarqué dans la section 1.3.2 que le remaillage associé à un mapping n'était pas nécessairement conservatif. Ce test numérique nous montre qu'il l'est en pratique, au moins dans le cas d'une grille polaire. Le solveur de Riemann approché donne sur ce test de meilleurs résultats que les différences finies qui génèrent beaucoup trop d'oscillations.

Nous avons comparé les résultats du solveur de Riemann approché à ceux d'Abgrall et al. ([1]). Ces derniers sont clairement de meilleure qualité : sous-évaluation de la densité moindre, choc un peu plus raide. En améliorant la résolution de la phase Lagrange, c'est-à-dire des efforts de pression, par le solveur de Riemann approché, nous pourrions sans doute compenser, au moins partiellement, cet écart. Cette comparaison nous permet aussi de conclure que le fait que le remaillage associé à un mapping ne soit pas conservatif ne nous empêche pas d'obtenir des résultats satisfaisants.

Avec une méthode complètement lagrangienne (sans l'étape d'advection-projection), les éléments de discrétisation se comprimeraient et se rapprocheraient du centre au cours du temps. La résolution serait donc meilleure au centre du domaine que ce n'est le cas ici, où le nombre de particules qui s'y trouvent ne varie pas. Pour compenser cet inconvénient, il serait utile d'améliorer la résolution près du centre en y utilisant une grille et des particules plus fines. Cette idée sera développée dans le prochain chapitre.

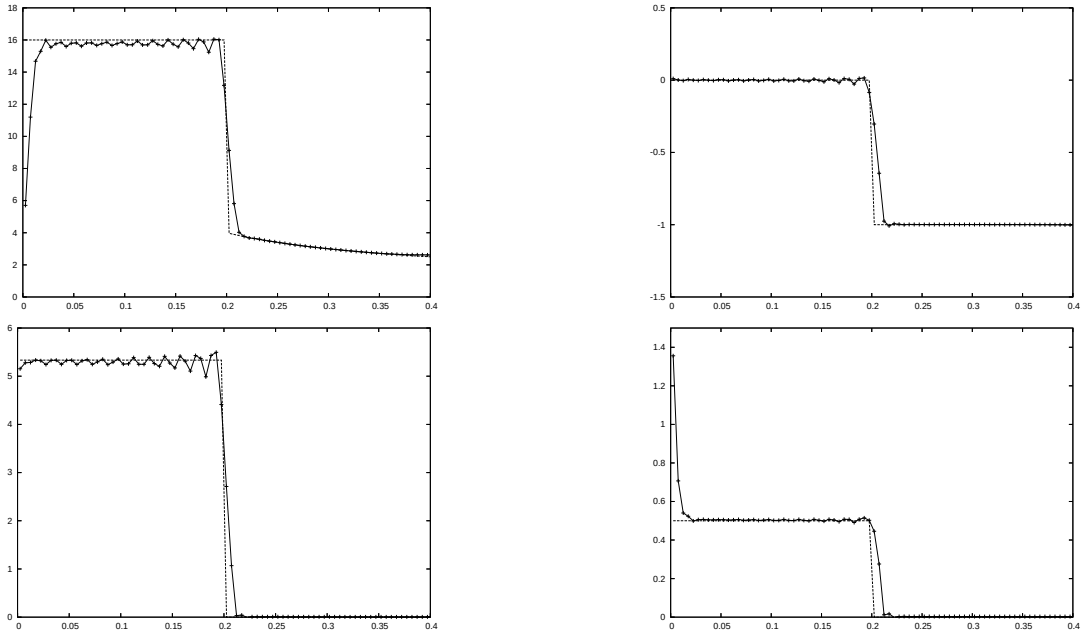


FIG. 3.27 – Problème de Noh, grille polaire, dr = 1/200, d$\theta = 2\pi/200$, M_4', différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

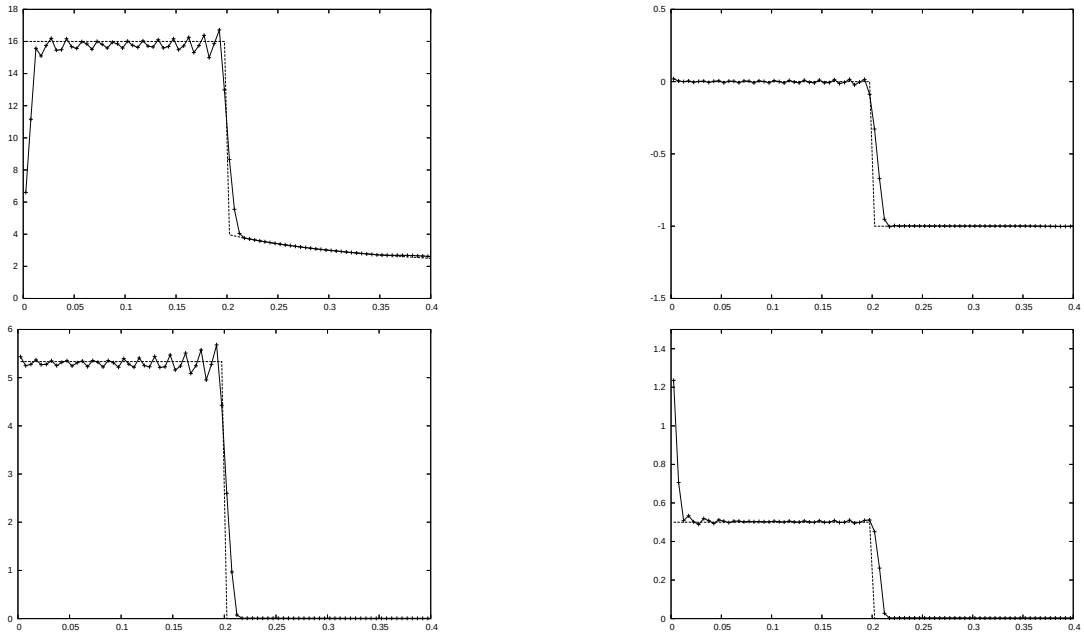


FIG. 3.28 – Problème de Noh, grille polaire, dr = 1/200, d$\theta = 2\pi/200$, Λ_3, différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

3.2.5.2 Résolution avec grille cartésienne

Une grille cartésienne est moins adaptée à la géométrie de l'écoulement. Résoudre le problème de Noh avec cette grille rend donc le test encore plus difficile. Cela permet d'observer comment se comporte la méthode dans un cas qui ne lui est pas favorable. A priori, on peut s'attendre à ce que, comme dans le cas des volumes finies, les résultats soient moins bons que sur la grille polaire et que la symétrie cylindrique ne soit pas préservée par les calculs. Ce test a été réalisé avec des différences finies centrées et le noyau M_4'. Un terme de viscosité artificielle a été ajouté, dont le coefficient vaut 1.6 dans les zones de compression et 0.4 ailleurs. La figure (3.32) présente les résultats selon la coupe $x = y \geq 0$ à t=0.6. Les résultats semblent être de qualité comparable à ceux obtenus pour la grille polaire. La figure (3.33) présente les isovaleurs de la densité autour du centre du domaine de calcul. On remarque que la solution est mal calculée le long des axes x et y, et que la symétrie cylindrique n'est pas préservée, surtout autour de ces axes.

3.3 Conclusion du chapitre 3

Dans ce chapitre, nous avons présenté l'algorithme avec lequel nous résolvons les équations d'Euler avec la méthode particulaire. Puis nous avons testé la méthode sur un certain nombre de problèmes académiques. Nous avons tout d'abord vérifié sur une

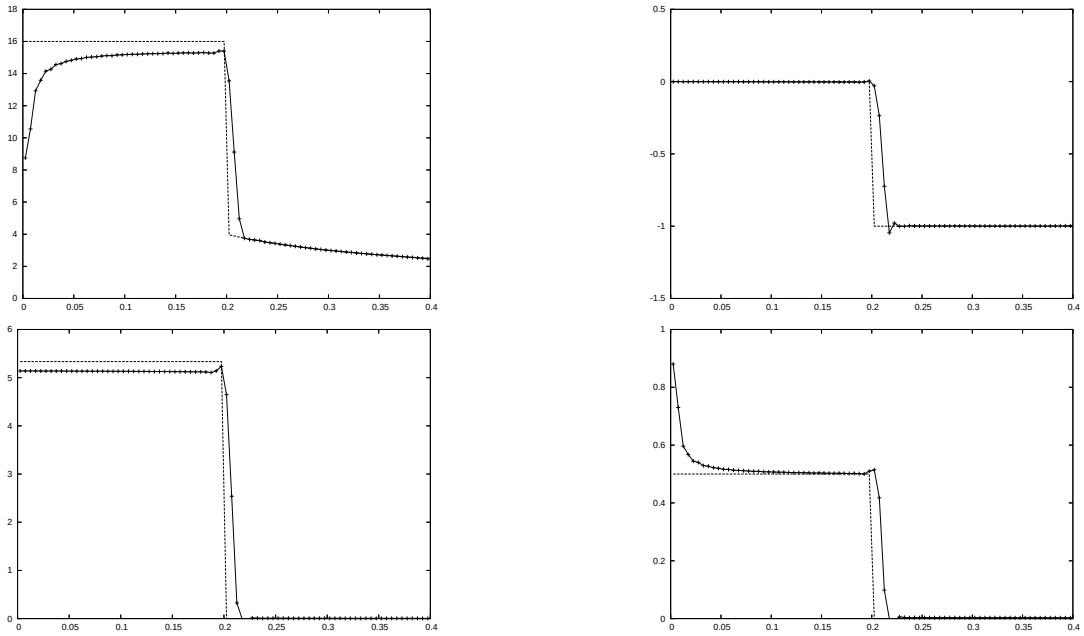


FIG. 3.29 – Problème de Noh, grille polaire, dr = 1/200, dθ = 2π/200, M_4', solveur de Riemann, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

équation d'advection uniforme que l'ordre de convergence numérique de la méthode particulaire était le même que l'ordre théorique prédit dans le chapitre 2. Nous avons également mis en évidence une propriété intéressante des schémas particulaires : ceux-ci semblent converger vers la solution entropique, ce que notre analyse théorique ne nous avait pas permis de prédire. Sur ces premiers tests, c'est incontestablement le noyau Λ_3 qui donne les meilleurs résultats. Nous avons ensuite testé la méthode sur trois problèmes : tube à chocs de Sod, détente de Sjögreen et problème de Noh. Sur ces derniers tests, nous n'avons pas trouvé de différence significative entre les comportements des noyaux d'interpolation M_4' et Λ_3. Le noyau Λ_2 a quant à lui été écarté en raison de son caractère très dispersif. Il nous semble finalement préférable d'utiliser le noyau Λ_3 plutôt que les deux autres, en raison de sa meilleure précision, en particulier quand on souhaite résoudre un problème d'advection, comme c'est le cas de la variable ϕ que nous utiliserons pour la simulation d'écoulements bifluides.

L'utilisation de différences finies centrées donne de meilleurs résultats que celle du solveur de Riemann approché pour les problèmes de Sod et de Sjögreen. Pour le problème de Noh, très sévère, c'est le solveur qui donne de meilleurs résultats, parce qu'il évite l'apparition d'oscillations non-physiques. Il nous semble que les différences finies centrées sont plus robustes que le solveur de Riemann approché. A moins de vouloir résoudre un problème faisant intervenir des chocs extrêmements forts, nous utiliserons

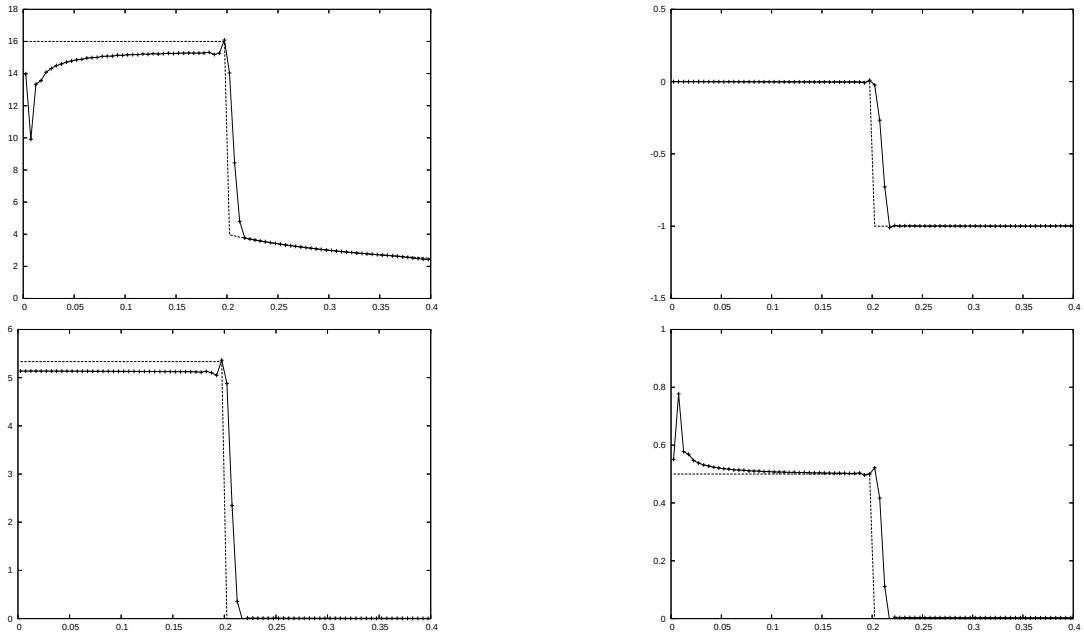


FIG. 3.30 – Problème de Noh, grille polaire, dr = 1/200, dθ = 2π/200, Λ_3, solveur de Riemann, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

donc dans la suite du travail les différences finies centrées. La résolution de la méthode serait sans doute améliorée si nous utilisions une technique de type MUSCL associée à des limiteurs pour résoudre la phase Lagrange : cela nous permettrait de bénéficier d'une meilleure précision que celle que nous obtenons avec notre implantation du solveur de Riemann approché, tout en évitant l'apparition d'oscillations, grâce au décentrage et à l'utilisation de limiteurs.

L'utilisation du terme de viscosité articielle (3.5) reste empirique, en particulier pour le choix de son coefficient. Nous utilisons d'ailleurs les mêmes valeurs pour des calculs avec M_4' et Λ_3, alors que ces noyaux ont des propriétés différentes. Bien que nous ayons utilisé des valeurs différentes pour le tube à chocs de Sod, nous avons remarqué, au cours de nos essais numériques, que les valeurs 1.6 dans les zones de compression, et 0.4 ailleurs, donnaient dans tous les cas des résultats au moins satisfaisants, parfois optimaux. Nous recommandons donc d'utiliser ces valeurs.

Nous avons vu sur le problème de Noh que notre méthode ne bénéficiait pas de l'adaptivité naturelle des méthodes purement lagrangiennes à cause de l'étape de remaillage qui empêche les particules de se regrouper dans les zones de fort gradient, et leur volume de varier beaucoup. Pour pallier cette limitation et améliorer la résolution de la méthode dans les zones intéressantes du domaine de calcul, il nous faut utiliser

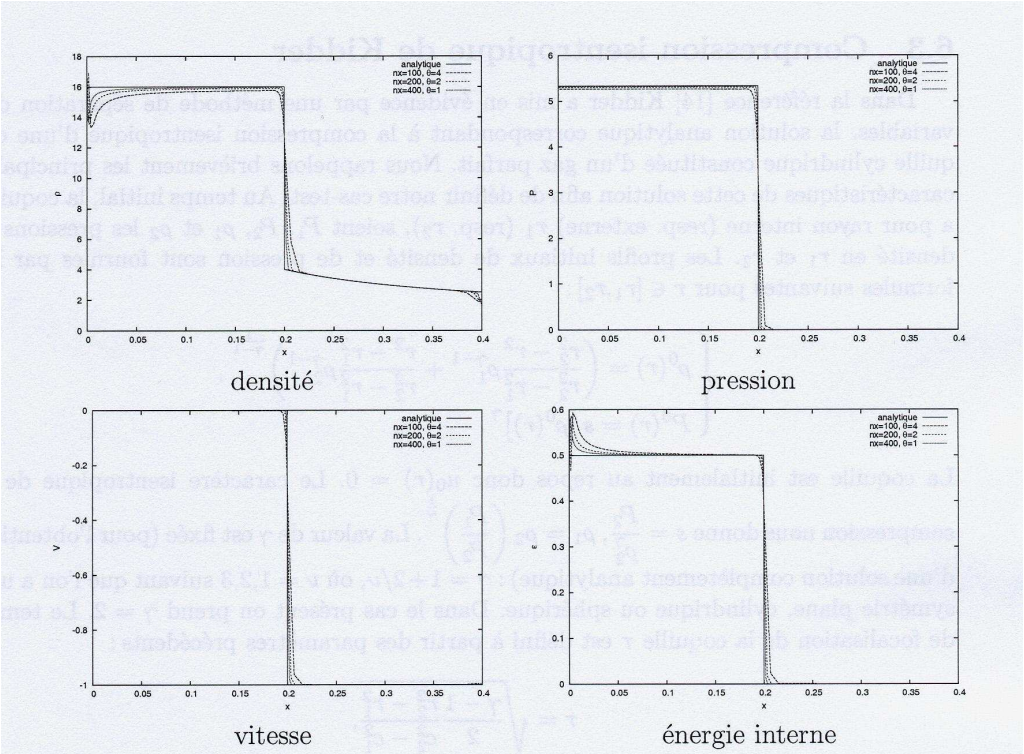


FIG. 13 – *Profils test de Noh à* $t = 0.6$*, comparaison avec la solution analytique*

FIG. 3.31 – Résultats de Abgrall et al. [1] pour le problème de Noh, $dr = 100$, 200 et 400.

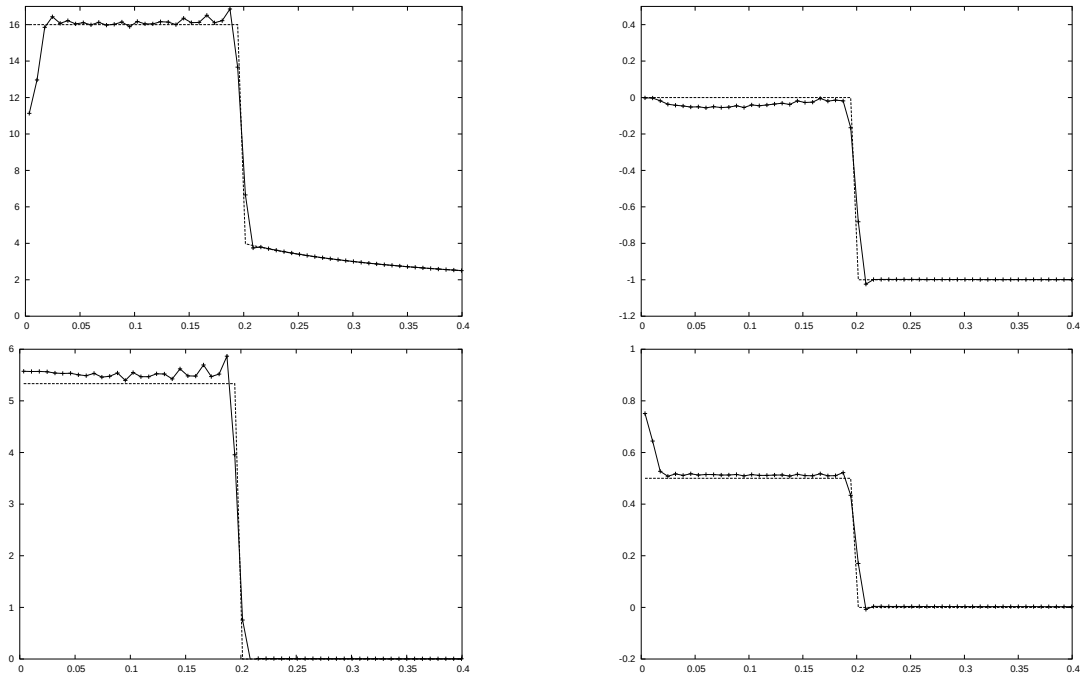


FIG. 3.32 – Problème de Noh, grille cartésienne, dx = dy = 1/200, M_4', dt = 0.0015, différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

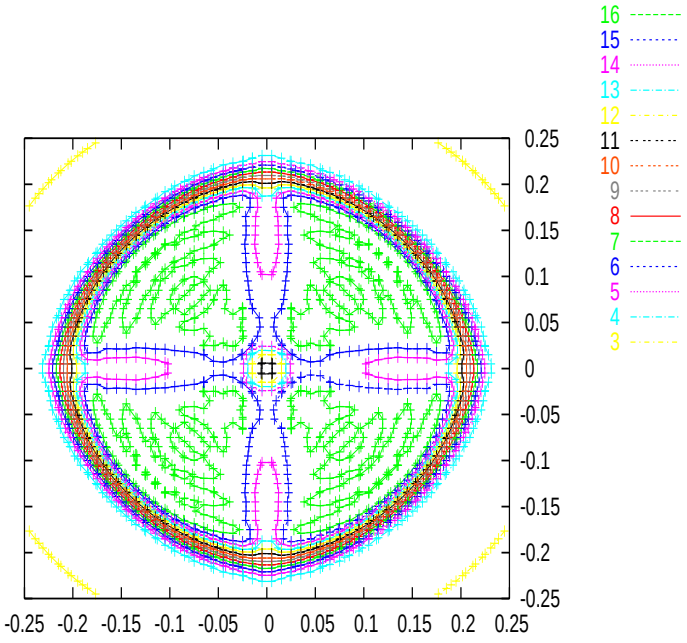


FIG. 3.33 – Problème de Noh, grille cartésienne, dx = dy = 1/100, dt = 0.003, isovaleurs de densité.

localement des grilles et des particules plus fines. C'est ce que nous allons étudier dans le chapitre suivant.

Chapitre 4

Méthode particulaire multi-niveaux et remaillage adaptatif

Nous avons vu dans le chapitre précédent, à propos du problème de Noh, qu'il serait utile de pouvoir adapter la taille des particules aux propriétés locales des écoulements simulés. Dans ce chapitre on implante donc une technique de raffinement de grilles. On décrit d'abord les conditions de superposition des grilles qui permettent de transmettre de l'information entre les différentes grilles et particules pendant le remaillage. On présente ensuite un critère d'adaptation des grilles, basé sur le gradient des variables dont on calcule l'évolution et on s'intéresse à la conservativité de la méthode particulaire utilisée avec la technique de raffinement de grilles. Puis on teste le fonctionnement de cette technique, sa conservativité, ainsi que le critère d'adaptation, sur des exemples numériques : tube à chocs de Sod, problème de Noh, résolus en utilisant deux grilles, et formation d'un choc avec l'équation de Bürgers.

4.1 Raffinement des grilles de remaillage

Dans cette partie on rappelle les principes de la technique de raffinement de grille choisie. On explique comment, c'est-à-dire sur quels critères, les grilles évoluent. On s'intéresse à la conservativité de la méthode particulaire quand plusieurs grilles coexistent : est-elle maintenue ?

4.1.1 Pourquoi utiliser plusieurs grilles ?

Une particule qu'on laisse évoluer librement se déplace à la vitesse de l'écoulement. Elle se déforme, se comprime ou se dilate suivant les caractéristiques de cet écoulement. Ainsi, par exemple, dans les zones de compression telles que les chocs, les particules s'accumulent et leur volume diminue ; leur représentation de l'écoulement y est par conséquent plus précise. On considère donc que les particules sont naturellement adap-

tatives. Cependant, le remaillage, qui a été introduit pour préserver la régularité de la distribution des particules, vient contrarier cette adaptivité naturelle. Il empêche les particules de se regrouper dans les zones de compression et uniformise leur volume. La précision de la méthode dépend alors non plus des caractéristiques locales de l'écoulement, mais de la taille des cellules de la grille (surtout si le remaillage est effectué tous les pas de temps, comme c'est notre cas). Pour augmenter la précision, il faut donc augmenter la finesse de la grille. Résoudre avec beaucoup de précision une zone précise impose, si on utilise la même grille sur tout le domaine de calcul, une résolution inutilement coûteuse dans d'autres zones.

Il est donc intéressant d'adapter la taille des particules aux propriétés locales de l'écoulement. Des particules de taille variable ont ainsi été introduites, dans le cadre des méthodes vortex ([43], [70], [18]). Cependant, la taille de ces particules n'évolue pas au cours du temps : l'adaptivité est uniquement spatiale. Cette technique n'est donc pas satisfaisante si on veut par exemple accompagner dans son mouvement une zone de fort gradient, tel un choc. Deux nouvelles techniques ont récemment été développées dans [6] :

- La première consiste à utiliser des particules de taille variable, associées aux cellules d'une grille non-uniforme. Le remaillage s'effectue par mapping sur une grille uniforme, puis, les particules sont projetées par un autre mapping sur une grille non-uniforme différente de la première. Cette deuxième grille est choisie en fonction de la manière dont évolue l'écoulement.
- La deuxième technique est inspirée de l'Adaptive Mesh Refinement (AMR), développé par Berger et Oliger [7], pour les méthodes de différences finies. Le domaine de calcul est recouvert de plusieurs grilles uniformes, plus ou moins fines, auxquelles correspondent des particules plus ou moins grosses, afin que la discrétisation soit adaptée aux exigences de précision. La position et la taille des grilles varient en fonction de l'évolution de l'écoulement. Les intersections de ces grilles doivent respecter certaines conditions afin de garantir au moment du remaillage la consistence des calculs.

Cette deuxième technique semble adaptée à la simulation de phénomènes décrits par les équations d'Euler compressibles (au CEA, interaction plasma-laser dans le cadre de la fusion par confinement inertiel) : par exemple le suivi de zones de très fort gradient (chocs) au cours de leur déplacement, ou le suivi de l'interface perturbée entre deux fluides. En effet, dans ces deux cas, les zones qui nécessitent d'être résolues avec plus de précision sont très localisées, et facilement repérables. On peut donc aisément placer une grille plus fine sur ces zones. C'est donc cette technique que nous allons utiliser.

4.1.2 Conditions de recouvrement des grilles

On explique dans cette section pourquoi les grilles doivent se recouvrir partiellement et quelle est la largeur de superposition nécessaire à la consistance des calculs. Nous supposons que le domaine de calcul est recouvert de deux grilles, une grossière située

dans la partie gauche du domaine, et une fine, située dans la partie droite. Soit dH le pas de la grille grossière, et dh celui de la grille fine. On s'intéresse ici uniquement à la zone du domaine où les grilles sont voisines, c'est pourquoi on note $]-\infty, b]$ le support de la grille grossière et $[a, \infty[$ celui de la grille fine, avec $a < b$ (voir la figure (4.1)).

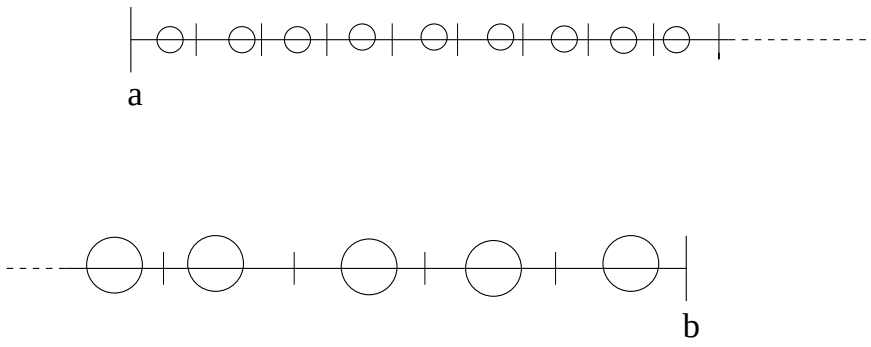


FIG. 4.1 – Grille et particules fines, grille et particules grossières.

Les calculs sont menés sur chaque grille indépendamment de l'autre : résolution de la phase Lagrange et advection des particules. Le déplacement des particules modifie la zone sur laquelle les particules de la grille décrivent de manière consistante l'écoulement. C'est ce qu'illustre pour la grille fine la figure (4.2). Soit u la vitesse du fluide. On peut considérer que l'écoulement reste correctement décrit, après déplacement des particules, sur :

$$\begin{aligned}]-\infty, b'] &=]-\infty, b - ||u||_\infty \, dt] \\ \left[a', +\infty\right[&= [a + ||u||_\infty \, dt, +\infty[\end{aligned}$$

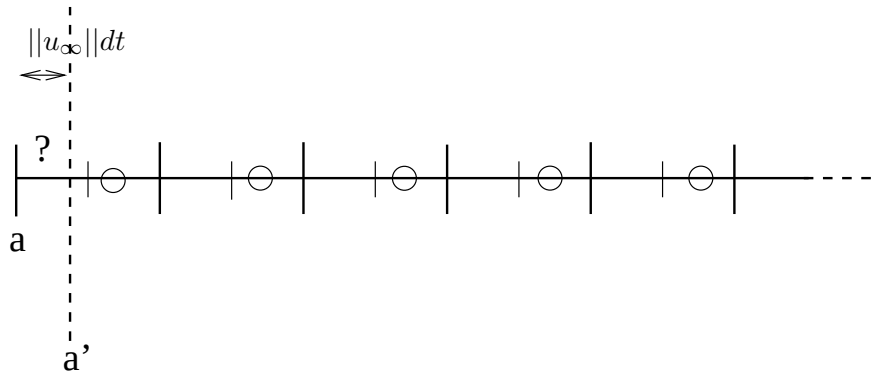


FIG. 4.2 – La description de l'écoulement par les particules n'est plus consistante sur la première cellule de la grille fine.

Ensuite vient l'étape du remaillage. Comme nous utilisons des noyaux d'interpolation symétriques, nous avons besoin, pour créer une nouvelle particule en un point donné, de disposer d'anciennes particules sur une certaine distance autour du point considéré. Cela n'est pas possible pour les points situés près des extrémités de la grille, qui ne disposent pas de suffisamment de particules sur un de leurs côtés. C'est ce qu'illustre pour la grille fine la figure (4.3), où on suppose que le remaillage se fait avec un noyau d'interpolation de support 4, comme par exemple M'_4.

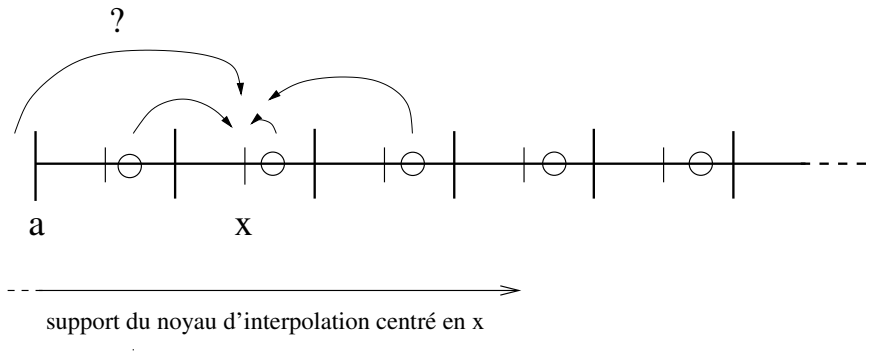


FIG. 4.3 – Il manque de l'information à gauche de la grille fine pour créer une nouvelle particule au centre de la deuxième cellule de la grille.

On note $]-\infty, b'']$ et $[a'', \infty[$ les parties de la grille où de nouvelles particules peuvent être créées sans ce problème par interpolation des anciennes. Pour simplifier, dans la suite on qualifiera de "zones fiables" ces parties des grilles. Evidemment, $]-\infty, b''] \subset]-\infty, b']$ et $[a'', \infty[\subset [a', \infty[$. On note dx le pas de la grille, et k la moitié de la largeur du support du noyau d'interpolation qu'on utilise. Ce nombre k est lié au nombre de points que nécessite dans chaque direction le noyau d'interpolation pour "remailler" les particules. Les paramètres a'' et b'' doivent vérifier :

$$\begin{aligned} a' - a'' &\geq k\ dh \\ b'' - b' &\geq k\ dH \end{aligned}$$

La figure (4.4), qui reprend la même configuration de grille que les figures précédentes, montre quelle est la zone fiable de la grille fine. Si on décide de déplacer uniformément les grilles d'une distance d, les conditions sur a'' et b'' deviennent :

$$\begin{aligned} a' - a'' &\geq d + k\ dh \\ b'' - b' &\geq d + k\ dH \end{aligned}$$

Les cellules dont les centres appartiennent à $[a, a'']$ ou $[b'', b]$ ne peuvent être remaillées de la manière habituelle. Pour y créer de nouvelles particules, la solution est

d'utiliser les particules de l'autre grille, ce qu'illustre la figure (4.5). Pour qu'une interpolation consistante puisse avoir lieu sur toutes les cellules de $[a, a'']$ il faut que :

$$b' - a'' \geq k\ dH$$

De même, pour qu'une interpolation consistante puisse avoir lieu sur toutes les cellules de $[b'', b]$ il faut que :

$$b'' - a' \geq k\ dH$$

Soit $D = b'' - a''$. Les deux conditions précédentes signifient que :

$$D > k(dH - dh)$$

Pour résumer :
- Dans les cellules appartenant à $[a'', \infty[$ et $]-\infty, b'']$, on crée de nouvelles particules par la procédure d'interpolation habituelle.
- Dans les cellules appartenant à $[a, a'']$ et $[b'', b]$, le remaillage se fait par interpolation des particules de l'autre grille.
- On doit avoir :

$$\begin{aligned} b - b'' &\leq ||u||_\infty\ dt + k\ dH \\ a'' - a &\leq ||u||_\infty\ dt + k\ dh \\ b'' - a'' &\geq k(dH - dh) \end{aligned}$$

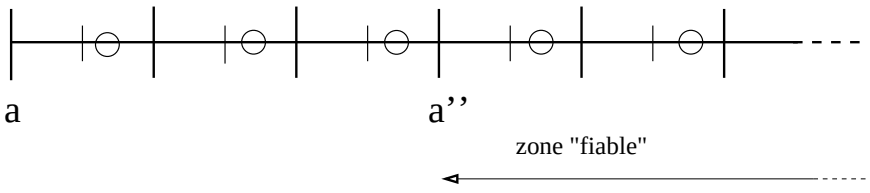


FIG. 4.4 – Zone fiable de la grille fine, composée des cellules où le remaillage habituel s'effectue en toute confiance.

4.1.3 Adaptation du pas de temps

Dans les sous-sections précédentes nous avons uniquement parlé de raffinement spatial. Les calculs peuvent être menés avec le même pas de temps sur toutes les grilles. Mais cela signifie que l'évolution des grosses particules sera calculée avec un pas de temps plus petit que nécessaire, dicté par les conditions de stabilité des particules fines, plus restrictives. Cela peut être gênant pour deux raisons : tout d'abord, parce qu'un pas de temps plus petit que nécessaire entraîne un surcroît de calculs, et ensuite, parce que ce pas de temps trop petit aura un effet dispersif sur la résolution numérique. Il parait plus approprié d'utiliser des pas de temps différents sur chaque grille, afin

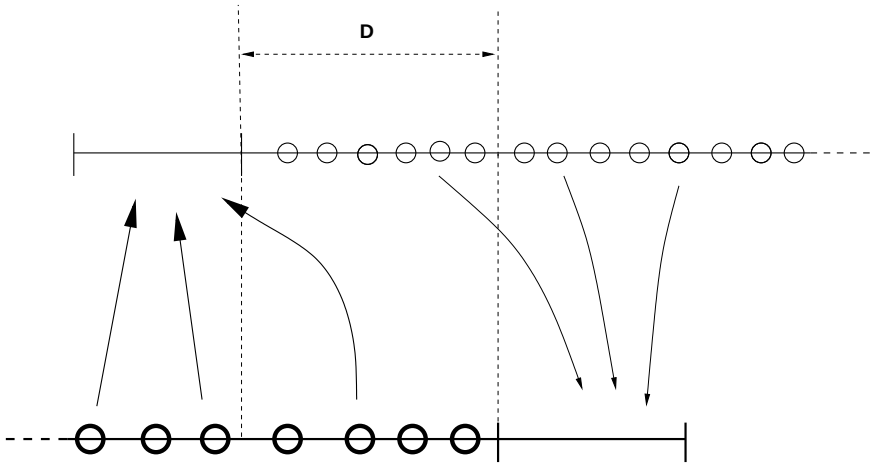


FIG. 4.5 – Des particules de l'autre grille permettent de remailler dans la zone non fiable de chaque grille.

que chaque niveau de discrétisation bénéficie d'un pas de temps qui lui est adapté. Toutefois, comme on vient de le voir dans la sous-section précédente, pour créer de nouvelles particules sur une grille, on a besoin de connaitre au même temps les valeurs de certaines particules de l'autre grille. Une solution consiste à calculer au début d'une itération sur la grille grossière quelle sera la variation des valeurs des particules de cette grille au cours d'un pas de temps grossier, puis d'interpoler linéairement cette variation pour obtenir les valeurs à chaque itération de la grille fine. L'interpolation linéaire ne demande pas beaucoup de calculs supplémentaires, et si on veut vraiment réduire le nombre d'opérations, on peut repérer les particules qui interviennent dans le remaillage de la grille fine, et ne pratiquer l'interpolation que sur celles-là. Il est aussi envisageable d'utiliser cette technique de raffinement en temps sans l'associer à un raffinement spatial. On effectue alors les calculs sur des grilles de même pas d'espace, mais avec des pas de temps différents. Nous étudierons dans la section suivante cette possibilité dans le cas de l'équation de Bürgers .

4.1.4 Critères d'adaptation des grilles

Pour que le raffinement de grilles soit réellement adaptatif, il faut que le nombre, la largeur et la position des grilles évoluent de façon automatique, en fonction des variations temporelles de l'écoulement. Pour cela, il faut établir des critères liés aux propriétés de l'écoulement, qui, lorsqu'ils sont vérifiés, entraînent la création, le déplacement, l'extension, ou le rétrécissement des grilles. Quels critères choisir ?

Généralement, on souhaite apporter une meilleure résolution, par l'emploi d'une grille plus fine, dans les zones du domaine de calcul où sont les phénomènes plus difficiles à simuler. Par exemple, les zones de fort gradient (de densité, vitesse,...) tels que

les chocs ou la discontinuité de la fonction permettant de repérer l'interface entre deux fluides (dont nous nous parlerons dans le prochain chapitre). Dans ce cas, les critères d'adaptation des grilles doivent tenir compte de l'amplitude des gradients des quantités qui nous intéressent.

Le premier critère concerne la création d'une grille plus fine sur une zone de fort gradient, de manière à résoudre plus précisément cette dernière. On note $(A_x)_i$ le gradient de la variable A évalué sur les particules i de la grille de départ, ou grille grossière. Si en un point i_0, $|(A_x)_{i_0}|$ est supérieure à une valeur seuil M, alors il y a création d'une grille fine autour de ce point i_0. Cela signifie que la grille grossière se trouve coupée en deux parties, de part et d'autre de la zone où la grille fine apparaît. Nous n'avons pas eu besoin, lors de nos tests numériques, d'utiliser ce critère, car les zones de forts gradients étaient connues dès l'instant initial. Il nous suffisait donc de placer une grille plus fine dessus. Cependant, dans le cas général, il faut l'utiliser pour être sûr de résoudre avec la même précision toutes les zones de fort gradient.

Pour que la grille fine continue au cours du temps de couvrir de manière satisfaisante ces zones de fort gradient, il faut s'assurer que ces dernières restent contenues dans la partie "fiable" de la grille fine. Sinon, on court le risque de perdre au cours du remaillage une partie de l'information à propos de ce fort gradient. En effet, l'information portée par des particules de la grille fine situées en dehors de la zone "fiable" sera remplacée pendant le remaillage par de l'information moins précise issue de l'interpolation des particules grossières. Pour s'assurer que les zones de fort gradient restent dans cette zone de la grille fine, on propose d'utiliser un second critère basé sur le maximum de la valeur absolue des gradients (dans le cas 1D, en 2D ou 3D on considère une à une les dérivées partielles par rapport aux variables d'espace) des quantités considérées sur la zone fiable de la grille fine : Soit $(A_x)_i$ le gradient de la variable A évalué sur les particules i de la grille fine.

$$\text{Si} \quad \max_{\text{particules} \notin \text{ zone fiable}} |(A_x)_i| \quad \geq \quad C \max_{\text{particules} \in \text{ zone fiable}} |(A_x)_i| \qquad (4.1)$$

alors on décide d'étendre la grille.

Pour être sûr de tenir compte de toutes les discontinuités on peut utiliser ce critère pour la densité, la vitesse et l'énergie totale. C est un coefficient qu'il faut choisir, la valeur $C = 0.2$ nous a paru efficace lors de nos essais numériques.

Si on décide d'utiliser des pas de temps différents sur les différentes grilles, il est commode de déterminer uniquement les pas de temps de la grille fine en fonction des conditions de stabilité, et d'additionner un nombre fixe N_0 de ces pas de temps pour obtenir ceux de la grille grossière. Il faut cependant savoir si un tel pas de temps serait compatible avec les propres conditions de stabilité de la grille grossière. La stabilité de la phase d'advection est assurée si les particules ne se croisent pas. Il s'agit d'une condition de type $dt < \frac{dx}{\max_i \ |U_i - U_{i+1}|_+}$, qui concerne donc le gradient de vitesse. Si on applique le critère d'évolution des grilles (et si aucune discontinuité n'apparaît après le

début des calculs), on sait que

$$\max_{\text{grille grossière}} |(U_x)_i| \leq C \max_{\text{grille fine}} |(U_x)_i|$$

En conséquence la condition de stabilité sur la grille grossière permet d'utiliser des pas de temps $1/C$ plus grands. Si on prend $N_0 \leq \frac{1}{C}$, la condition de stabilité de la phase d'advection est compatible avec le critère d'évolution des grilles. Par contre, la condition de stabilité de la phase "Lagrange", qui est du type $\sqrt{\frac{p(\gamma-1)}{\rho}}\frac{dt}{dx} \leq constante$ n'a pas de raison d'être vérifiée sur la grille grossière pour le pas de temps grossier si elle l'est sur la grille fine pour le pas de temps fin. Nous n'avons pas rencontré, parmi les cas-tests que nous avons effectué, de situation où la condition de stabilité de la phase Lagrange était plus restrictive sur la grille grossière que sur la grille fine. Il faut cependant, de manière générale, y prendre garde.

4.1.5 Frontière entre les représentations associées à chaque grille

Nous supposons que le domaine de calcul est recouvert de deux grilles, une grossière située dans la partie gauche du domaine, et une fine, située dans la partie droite, et que les cellules de la grille fine sont toutes comprises dans une seule cellule de la grille grossière. Avec la technique de raffinement de grilles que nous utilisons, il est nécessaire que les grilles se superposent partiellement. Les conditions de recouvrement des grilles nous disent plus précisément que l'intersection des zones des grilles peuplées de particules "fiables" est non vide. Dans cette intersection, deux types de particules, et donc deux représentations différentes de l'écoulement, toutes deux valables, co-existent. Cela pose un problème pour définir, par exemple, la masse totale du fluide. Pour résoudre ce problème, nous avons choisi de séparer cette zone d'intersection en deux parties, et de ne considérer dans chacune des deux parties qu'un seul type de particule. Nous avons décidé de prendre comme frontière entre ces deux parties la première extrémité droite d'une cellule de la grille grossière qui touche la zone "fiable" de la grille fine. Ainsi, on s'assure que la frontière ne passe pas au milieu d'une particule grossière, ce qui ne serait pas pratique pour compter la masse. Cela nous permet aussi d'utiliser sur la plus grande étendue possible la représentation associée à la grille fine. C'est ce que représente la figure (4.6).

Pour définir la masse totale, nous avons besoin de préciser quelques notations. Soit dH la taille d'une cellule de la grille grossière, et dh celle d'une cellule de la grille fine. Soit dt le pas de temps utilisé pour la résolution numérique. On pose aussi : $\lambda_1 = \frac{dt}{dH}$, $\lambda_2 = \frac{dt}{dh}$. On note N_1 l'indice de la dernière cellule grossière avant la frontière, et $N2+1$ l'indice de la première cellule fine après cette frontière (voir la figure (4.7)). Afin d'utiliser des notations plus simples, nous revenons au cas de l'équation scalaire 1D que nous avions étudié dans le premier chapitre :

$$u_t + (g(u)u)_x = 0$$

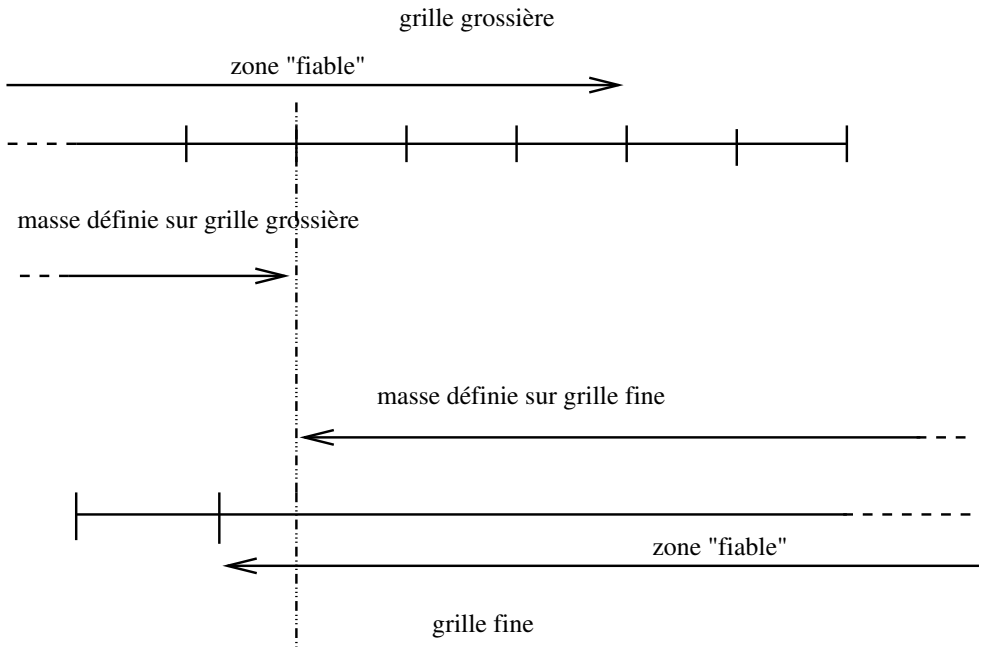


FIG. 4.6 – Définition de la masse totale quand il y a deux grilles.

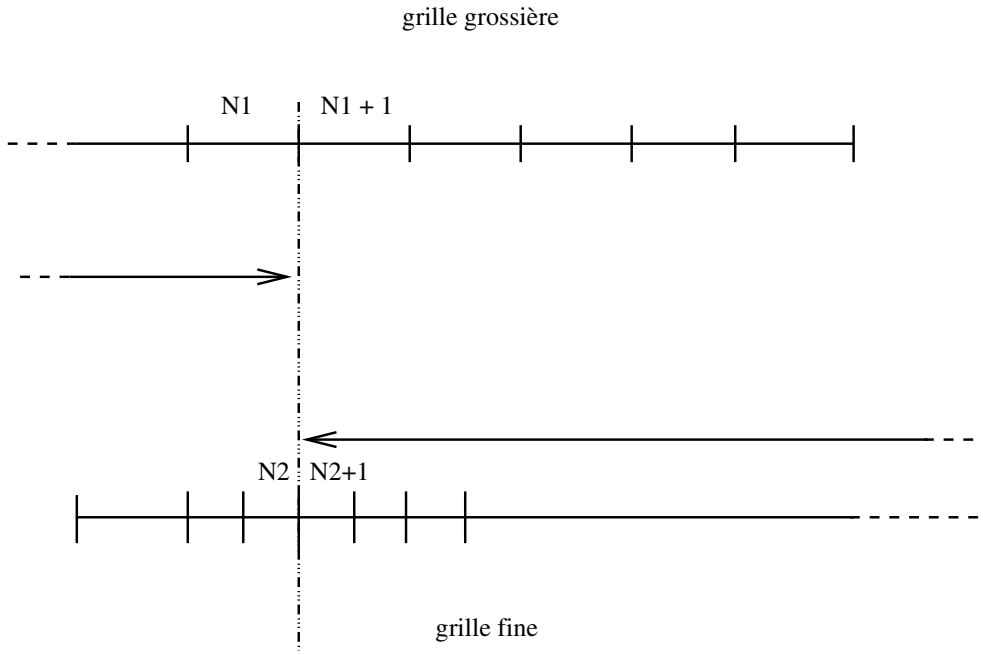


FIG. 4.7 – Indexation des cellules autour de la frontière entre les grilles.

On note U_i les valeurs de u portées par les particules grossières, et u_i celles portées par les particules fines. Pour simplifier l'écriture, on note :

$$G_i = \tilde{g}(U_i),\ g_i = \tilde{g}(u_i)$$

Remarque: *On ne précise pas comment est évaluée la vitesse des particules. En effet, dans cette section on ne cherche pas à analyser la précision de la solution numérique, mais uniquement si la masse totale est conservée. Il n'y a pas de raison pour que la précision du déplacement influe sur la conservation de la masse, donc nous ne nous en préoccupons pas.*

Définition 4.1 :
La masse totale du fluide est définie par :

$$M = \sum_{i=-\infty}^{N_1} U_i dH + \sum_{i=N_2+1}^{+\infty} u_i dh \tag{4.2}$$

$\diamond$

On utilise la notation un peu abusive $+/-\infty$ pour désigner les extrémités des grilles. En pratique, cela ne signifie pas nécessairement que les grilles ont une taille infinie, mais, en tout cas, qu'il n'y a pas à se préoccuper des problèmes liés au remaillage à ces extrémités. Cela peut être le cas par exemple si les particules près de ces bords vérifient $u_i = 0$ ou $\tilde{g}(u_i) = 0$.

4.1.6 Conservativité de la technique multi-niveaux

Nous savons que la méthode particulaire avec remaillage est conservative dans la cas d'une seule grille. Qu'en est-il lorsque deux grilles sont utilisées ? Nous avons établi le résultat suivant :

Proposition 14 *Avec les notations introduites précédemment, si on utilise le même pas de temps sur les deux grilles et si on note M la masse totale initiale, $\tilde{M}$ celle après un pas de temps, et S la largeur du support de Λ, on a :*

$$\tilde{M} - M = \sum_{a=1}^{+\infty} \sum_{n \leq a} \frac{\partial^a u g(u)^n}{\partial x^a} \frac{1}{n!\,(a-n)!} \sum_{-\frac{S}{2} < i \leq 0} (i-0.5)^{a-n} \left[dH^{a+1}(-\lambda_1)^n - dh^{a+1}(-\lambda_2)^n \right]$$

Le terme dominant est d'ordre 2, donc la masse totale est conservée à l'ordre deux au cours d'un pas de temps. La variation de masse dépend du gradient de la quantité considérée, et en particulier elle est nulle dans le cas de particules portant des valeurs uniformes. Ce résultat a pour conséquence que, si on se soucie de conserver exactement la masse, il faut s'arranger pour que la zone autour de la frontière entre les grilles se

trouve à un endroit où les particules portent des valeurs uniformes. Cependant, il ne doit pas être optimal dans la mesure où les résultats numériques montrent une meilleure conservativité que celle prédite.

Remarque: *On constate aussi que si $dH = dh$ alors la variation de masse est nulle, ce qui est cohérent.*

Avant de démontrer ce résultat, établissons une propriété des noyaux d'interpolation qui nous sera utile pour la preuve de la proposition 14 :

Proposition 15 *Soit Λ un noyau d'interpolation préservant les M premiers moments. Alors Λ vérifie :*

$$\begin{aligned}\sum_{i\leq 0}\sum_{k}\Lambda^{(n)}(i-k)\ (k+0.5)^a &= \sum_{i\leq 0}\frac{a!}{(a-n)!}(i-0.5)^{a-n}\ si\ n\leq a\\ &= 0\ sinon\end{aligned}$$

Preuve:

On note x_k les centres des cellules de la grille.
Λ vérifie :

$$\sum_k \Lambda(x_k-y)\ x_k^a = y^a \text{ pour } 0\leq a\leq M-1$$

Donc :

$$\begin{aligned}\sum_k \Lambda^{(n)}(y-x_k)\ x_k^a &= a(a-1)...(a-n+1)\ y^{a-n} \text{ pour } 0\leq a\leq M-1\\ &= 0 \text{ pour } a\geq M\end{aligned}$$

Si on prend $x_k = k+0.5$ et $y = i+0.5$ on obtient :

$$\begin{aligned}\sum_i \Lambda^{(n)}(i-k)\ (k+0.5)^a &= \frac{a!}{(a-n)!}(i+0.5)^{a-n} \text{ pour } 0\leq a\leq M-1\\ &= 0 \text{ pour } a\geq M\end{aligned}$$

□

Passons maintenant à la preuve de la proposition 14.

Preuve:

Pour plus de simplicité, nous avons effectué les calculs seulement dans le cas d'un noyau à support impair (et supposé que les résultats sont semblables dans le cas d'un

noyau à support pair). On suppose que les valeurs initiales U_i et u_i portées par les particules grossières et fines sont les valeurs exactes de la solution u de l'équation modèle à l'instant t. Les particules sont déplacées, à la vitesse G_i ou g_i, puis interpolées pour créer de nouvelles distributions de particules $\tilde{U}_i$ et $\tilde{u}_i$. Toutes les cellules considérées, qu'elles appartiennent à la grille fine ou à la grille grossière, sont situées dans les zones "fiables" de ces grilles. Les nouvelles particules sont donc toutes créées à partir de l'interpolation de particules de même type (fin ou grossier) qu'elles.

La nouvelle masse totale après déplacement et remaillage vaut :

$$\tilde{M} = \sum_{i=-\infty}^{N_1} \tilde{U}_i \, dH + \sum_{i=N_2+1}^{+\infty} \tilde{u}_i \, dh$$

La variation de masse vaut :

$$\tilde{M} - M = \sum_{i=-\infty}^{N_1} \tilde{U}_i \, dH + \sum_{i=N_2+1}^{+\infty} \tilde{u}_i \, dh - \sum_{i=-\infty}^{N_1} U_i \, dH + \sum_{i=N_2+1}^{+\infty} u_i \, dh$$

La variation de masse sur la grille grossière vaut :

$$\begin{aligned}
\sum_{i=-\infty}^{N_1} \tilde{U}_i \, dH - \sum_{i=-\infty}^{N_1} U_i \, dH &= \sum_{i=-\infty}^{N_1} \sum_{k} U_k \, \Lambda\left(i - k - \lambda_1 G_k\right) \, dH - \sum_{i=-\infty}^{N_1} U_i \, dH \\
&= \underbrace{\sum_{i=-\infty}^{N_1} \sum_{k \le N_1} U_k \, \Lambda\left(i - k - \lambda_1 G_k\right) \, dH - \sum_{i=-\infty}^{N_1} U_i \, dH}_{\text{perdu par la grille}} \\
&\quad + \underbrace{\sum_{i=-\infty}^{N_1} \sum_{k > N_1+1} U_k \, \Lambda\left(i - k - \lambda_1 G_k\right) \, dH}_{\text{gagné par la grille}}
\end{aligned}$$

Or :

$$\sum_{i=-\infty}^{N_1} \sum_{k \le N_1} U_k \, \Lambda\left(i - k - \lambda_1 G_k\right) \, dH = \sum_{k \le N_1} U_k \, dH \left[1 - \sum_{i > N_1} U_k \, \Lambda(i - k - \lambda_1 G_k)\right]$$

car, comme Λ conserve le premier moment de la distribution des particules :

$$\sum_{i=-\infty}^{N_1} \Lambda\left(i - k - \lambda_1 G_k\right) \, + \sum_{i > N_1} U_k \, \Lambda(i - k - \lambda_1 G_k) = 1$$

Le bilan sur la grille grossière est donc :

$$\sum_{i=-\infty}^{N_1} \tilde{U}_i \, dH - \sum_{i=-\infty}^{N_1} U_i \, dH = - \sum_{k \le N_1} \sum_{i > N_1} U_k \, \Lambda\left(i - k - \lambda_1 G_k\right) \, dH + \sum_{k > N_1} \sum_{i \le N_1} U_k \, \Lambda\left(i - k - \lambda_1 G_k\right) \, dH$$

On suit le même raisonnement pour la grille fine, et on obtient :

$$\begin{aligned}\tilde{M} - M &= -\sum_{k\leq N_1}\sum_{i>N_1} U_k\ \Lambda\left(i-k-\lambda_1 G_k\right)\ dH + \sum_{k>N_1}\sum_{i\leq N_1} U_k\ \Lambda\left(i-k-\lambda_1 G_k\right)\ dH \\ &\quad + \sum_{k\leq N_2}\sum_{i>N_2} u_k\ \Lambda\left(i-k-\lambda_2 g_k\right)\ dh - \sum_{k>N_2}\sum_{i\leq N_2} u_k\ \Lambda\left(i-k-\lambda_2 g_k\right)\ dh\end{aligned}$$

On suppose, pour simplifier les notations, que $N_1 = 0$ et $N_2 = 0$.

En intervertissant les indices du deuxième et du quatrième terme, on peut écrire :

$$\begin{aligned}\tilde{M} - M &= -\sum_{k\leq 0}\sum_{i>0} U_k\ \Lambda\left(i-k-\lambda_1 G_k\right)\ dH + U_i\ \Lambda\left(k-i-\lambda_1 G_i\right)\ dH \\ &\quad + u_k\ \Lambda\left(i-k-\lambda_2 g_k\right)\ dh - u_i\ \Lambda\left(k-i-\lambda_2 g_i\right)\ dh\end{aligned}$$

Les sommes sur les indices k et i ne sont pas infinies. Si on note S la largeur du support du noyau Λ, alors en fait les termes en Λ de l'expression précédente sont nuls pour $k \leq -\frac{S}{2}$ et $i > \frac{S}{2}$. On a donc :

$$\begin{aligned}\tilde{M} - M &= -\sum_{-\frac{S}{2}<k\leq 0}\ \sum_{\frac{S}{2}\geq i>0} U_k\ \Lambda\left(i-k-\lambda_1 G_k\right)\ dH + U_i\ \Lambda\left(k-i-\lambda_1 G_i\right)\ dH \\ &\quad + u_k\ \Lambda\left(i-k-\lambda_2 g_k\right)\ dh - u_i\ \Lambda\left(k-i-\lambda_2 g_i\right)\ dh\end{aligned}$$

Pour ne pas trop s'encombrer, on n'utilise pas dans la suite des calculs cette notation plus précise, on s'en rappellera juste à la fin pour simplifier l'expression obtenue.

Comme dans le premier chapitre, on note N le degré de Λ, et on développe le terme $\Lambda(i-k-\lambda_1 G_k)$ par rapport à $i-k$.

$$\Lambda\left(i-k-\lambda_1 G_k\right) = \sum_{n=0}^{N} \frac{\Lambda^{(n)}(i-k)}{n!}(-\lambda_1 G_k)^n$$

On développe de même les autres termes en Λ. On obtient :

$$\begin{aligned}\tilde{M} - M &= -\sum_{k\leq 0}\sum_{i>0}\sum_{n=0}^{N} -dH\ \frac{\Lambda^{(n)}(i-k)}{n!}\ U_k\ G_k^n(-\lambda_1)^n \ + \ dH\ \frac{\Lambda^{(n)}(k-i)}{n!}\ U_i\ G_i^n(-\lambda_1)^n \\ &\quad + \ dh\ \frac{\Lambda^{(n)}(i-k)}{n!}\ u_k\ g_k^n(-\lambda_2)^n \ - \ dh\ \frac{\Lambda^{(n)}(k-i)}{n!}\ u_i\ g_i^n(-\lambda_2)^n\end{aligned}$$

On note à présent x l'abcisse de la frontière entre les deux zones, et $f_n = ug(u)^n$.
On développe le terme $U_k G_k^n$ par rapport à x, ainsi que les autres termes semblables.

$$U_k\ G_k^n \ = \ f_n(x + kdH + 0.5\ dH) = \sum_{a=0}^{+\infty} \frac{f_n^{(a)}(x)}{a!}(k-0.5)^a\ dH^a$$

$$\begin{aligned}\tilde{M}-M &= \sum_{k\leq 0}\sum_{i>0}\sum_{n=0}^{N}\sum_{a=0}^{+\infty}\frac{f_n^{(a)}(x)}{a!n!}[\,-dH\;\Lambda^{(n)}(i-k)\;(-\lambda_1)^n\;dH^a\;(k+0.5)^a \\ &\quad +dH\;\Lambda^{(n)}(k-i)\;(-\lambda_1)^n\;dH^a\;(i+0.5)^a+dh\;u_k\;g_k^n\;\Lambda^{(n)}(i-k)\;(-\lambda_2)^n\;dh^a\;(k+0.5)^a \\ &\quad -dh\;u_i\;g_i^n\;\Lambda^{(n)}(k-i)\;(-\lambda_2)^n\;dh^a\;(i+0.5)^a]\end{aligned}$$

$$\begin{aligned}\tilde{M}-M &= \sum_{n=0}^{N}\sum_{a=0}^{+\infty}\frac{f_n^{(a)}(x)}{a!n!}\left[dH^{a+1}\;(-\lambda_1)^n-dh^{a+1}\;(-\lambda_2)^n\right]\times \\ &\quad \sum_{k\leq 0}\sum_{i>0}\left[-\Lambda^{(n)}(i-k)\;(k+0.5)^a+\Lambda^{(n)}(k-i)\;(i+0.5)^a\right]\end{aligned}$$

Pour simplifier cette expression, nous nous intéressons au terme
$\sum_{k\leq 0}\sum_{i>0}\left[-\Lambda^{(n)}(i-k)\;(k+0.5)^a+\Lambda^{(n)}(k-i)\;(i+0.5)^a\right]$.

$$\sum_{k\leq 0}\sum_{i>0}-\Lambda^{(n)}(i-k)\;(k+0.5)^a = \sum_{k\leq 0}(k+0.5)^a\sum_{i>0}-\Lambda^{(n)}(i-k)$$

$$\sum_{i>0}-\Lambda^{(n)}(i-k)=-\sum_{i}\Lambda^{(n)}(i-k)+\sum_{i\leq 0}\Lambda^{(n)}(i-k)$$

Or :

$$\begin{aligned}-\sum_i\Lambda^{(n)}(i-k) &= -1 \text{ si } n=0 \\ &= 0 \text{ sinon} \\ &= \delta_0^n\end{aligned}$$

$$\sum_{k\leq 0}\sum_{i>0}-\Lambda^{(n)}(i-k)(k+0.5)^a = -\sum_{k\leq 0}(k+0.5)^a\;\delta_0^n+\sum_{k\leq 0}\sum_{i\leq 0}\Lambda^{(n)}(i-k)\;(k+0.5)^a$$

$$\sum_{k\leq 0}\sum_{i>0}\left[-\Lambda^{(n)}(i-k)(k+0.5)^a+\Lambda^{(n)}(k-i)(i+0.5)^a\right]=$$

$$=-\sum_{k\leq 0}(k+0.5)^a\;\delta_0^n+\sum_{k\leq 0}\sum_{i\leq 0}\Lambda^{(n)}(i-k)(k+0.5)^a+\sum_{k\leq 0}\sum_{i>0}\Lambda^{(n)}(k-i)(i+0.5)^a$$

$$=-\sum_{k\leq 0}(k+0.5)^a\;\delta_0^n+\sum_{i\leq 0}\sum_{k\leq 0}\Lambda^{(n)}(i-k)(k+0.5)^a+\sum_{k>0}\Lambda^{(n)}(i-k)(k+0.5)^a$$

$$-\sum_{k\leq 0}(k+0.5)^a\;\delta_0^n+\sum_{i\leq 0}\sum_{k}\Lambda^{(n)}(i-k)(k+0.5)^a$$

On a montré (proposition 15) que :

$$\begin{aligned}\sum_{i\leq 0}\sum_{k}\Lambda^{(n)}(i-k)\ (k+0.5)^a &= \sum_{i\leq 0}\frac{a!}{(a-n)!}(i-0.5)^{a-n} \text{ si } n\leq a\\ &= 0 \text{ sinon}\end{aligned}$$

Donc :

$$\begin{aligned}\sum_{k\leq 0}\sum_{i>0}\left[-\Lambda^{(n)}(i-k)(k+0.5)^a+\Lambda^{(n)}(k-i)(i+0.5)^a\right] &= -\sum_{k\leq 0}(k+0.5)^a\ \delta_0^n\\ &\quad+\sum_{i\leq 0}\frac{a!}{(a-n)!}(i-0.5)^{a-n}\end{aligned}$$

Et finalement, on obtient le résultat de la proposition 14 :

$$\begin{aligned}\tilde{M}-M &= \sum_{n=0}^{N}\sum_{a=0}^{+\infty}\frac{f_n^{(a)}(x)}{a!\ n!}\left[-\sum_{k\leq 0}\delta_0^n\ (k+0.5)^a+\sum_{i\leq 0}\frac{a!}{(a-n)!}(i-0.5)^{a-n}\right]\\ &= \sum_{a=1}^{+\infty}\sum_{n\leq a}\frac{f_n^{(a)}(x)}{n!\ (a-n)!}\sum_{i\leq 0}(i-0.5)^{a-n}\ \left[\ dH^{a+1}(-\lambda_1)^n-dh^{a+1}(-\lambda_2)^n\ \right]\\ &= \sum_{a=1}^{+\infty}\sum_{n\leq a}\frac{f_n^{(a)}(x)}{n!\ (a-n)!}\sum_{\frac{S}{2}<i\leq 0}(i-0.5)^{a-n}\ \left[\ dH^{a+1}(-\lambda_1)^n-dh^{a+1}(-\lambda_2)^n\ \right]\end{aligned}$$

□

Le résultat de la proposition 14 peut sembler un peu décevant. On aurait pu s'attendre à ce que l'ordre de conservation de la masse dépende de la précision du noyau d'interpolation. Nous avons donc vérifié dans le cas du noyau Λ_3 que même si le noyau conservait les quatre premiers moments, la masse totale n'était conservée qu'à l'ordre 2. Pour cela, nous avons supposé que les particules se déplacent toutes dans le même sens ($g(u)\geq 0$). Cette hypothèse nous permet de nous ramener au cas d'un noyau à support impair, puisqu'on sait alors exactement sur quels points de grille chaque particule est distribuée. Dans ce cas, la grille grossière perd :

$$dH\ [U_{N_1}\ \Lambda_3(1-\lambda_1 G_{N_1})+U_{N_1}\ \Lambda_3(2-\lambda_1 G_{N_1})+U_{N_1-1}\ \Lambda_3(2-\lambda_1 G_{N_1-1})]$$

et gagne

$$dH\ [\ U_{N_1+1}\ \Lambda_3(1+\lambda_1 G_{N_1+1})\ ]$$

On développe chacun de ces termes :

$$\Lambda_3(1-\lambda_1 G_{N_1})=\lambda_1 G_{N_1}+\frac{1}{2}\lambda_1^2 G_{N_1}^2-\frac{1}{2}\lambda_1^3 G_{N_1}^3$$

$$\Lambda_3(2-\lambda_1 G_{N_1}) = -\frac{1}{6}\lambda_1 G_{N_1} + \frac{1}{6}\lambda_1^3 G_{N_1}^3$$

$$\Lambda_3(2-\lambda_1 G_{N_1-1}) = -\frac{1}{6}\lambda_1 G_{N_1-1} + \frac{1}{6}\lambda_1^3 G_{N_1-1}^3$$

$$\Lambda_3(1+\lambda_1 G_{N_1+1}) = -\frac{1}{3}\lambda_1 G_{N_1+1} + \frac{1}{2}\lambda_1^2 G_{N_1+1}^2 - \frac{1}{6}\lambda_1^3 G_{N_1+1}^3$$

Le bilan sur la grille grossière fait donc :

$$dH[\frac{1}{2}\lambda_1^2 G_{N_1}^2 U_{N_1} + \lambda_1 G_{N_1} U_{N_1} - \frac{1}{2}\lambda_1^3 G_{N_1}^3 U_{N_1} + \frac{1}{6}\lambda_1^3 G_{N_1}^3 U_{N_1} - \frac{1}{6}\lambda_1 G_{N_1} U_{N_1} + \frac{1}{6}\lambda_1^3 G_{N_1-1}^3 U_{N_1-1}$$

$$-\frac{1}{6}\lambda_1 G_{N_1-1} U_{N_1-1} + \frac{1}{3}\lambda_1 G_{N_1+1} U_{N_1+1} - \frac{1}{2}\lambda_1^2 G_{N_1+1}^2 U_{N_1+1} + \frac{1}{6}\lambda_1^3 G_{N_1+1}^3 U_{N_1+1}]$$

Le bilan sur la grille fine est similaire, mais avec des signes opposés.
On développe l'expression du bilan sur la grille grossière par rapport à x l'abcisse de la frontière entre les grilles.

On obtient, à l'ordre 1 (il s'agit des termes d'ordre 0 du développement limité) :

$$dH[\frac{1}{2}\lambda_1^2 (g(u)^2 u)(x) + \lambda_1 (g(u)u)(x) - \frac{1}{2}\lambda_1^3 (g(u)^3 u)(x) + \frac{1}{6}\lambda_1^3 (g(u)^3 u)(x) - \frac{1}{6}\lambda_1 (g(u)u)(x)$$

$$+\frac{1}{6}\lambda_1^3 (g(u)^3 u)(x) - \frac{1}{6}\lambda_1 (g(u)u)(x) + \frac{1}{3}\lambda_1 (g(u)u)(x) - \frac{1}{2}\lambda_1^2 (g(u)^2 u)(x) + \frac{1}{6}\lambda_1^3 (g(u)^3 u)(x)]$$

Les termes en λ^2 ou λ^3 s'annulent entre eux. Le terme en λ s'annule avec celui correspondant pour la grille fine.

A l'ordre 2 on obtient :

$$dH[-\frac{1}{2}\lambda_1^2 (g(u)^2 u)'(x)\frac{dH}{2} - \lambda_1 (g(u)u)'(x)\frac{dH}{2} + \frac{1}{2}\lambda_1^3 (g(u)^3 u)(x)\frac{dH}{2} - \frac{1}{6}\lambda_1^3 (g(u)^3 u)'(x)\frac{dH}{2}$$

$$+\frac{1}{6}\lambda_1 (g(u)u)'(x)\frac{dH}{6} - \frac{1}{2}\lambda_1^3 (g(u)^3 u)'(x)\frac{3dH}{2} + \frac{1}{6}\lambda_1 (g(u)u)'(x)\frac{3dH}{2} + \frac{1}{3}\lambda_1 (g(u)u)'(x)\frac{dH}{2}$$

$$-\frac{1}{2}\lambda_1^2 (g(u)^2 u)'(x)\frac{dH}{2} + \frac{1}{6}\lambda_1^3 (g(u)^3 u)'(x)\frac{dH}{2}]$$

Les termes en λ et λ^2 ne s'annulent pas entre eux. Ils ne sont pas non plus compensés par les termes correspondants du bilan sur la grille fine. La conservation de la masse est donc d'ordre 2.

4.2 Validation numérique

4.2.1 Tube à chocs de Sod

Ce premier test reprend le problème du tube à chocs de Sod. Une grille fine est placée autour de la discontinuité initiale, elle s'adapte selon le critère (4.1) appliqué à la densité, la vitesse et l'énergie totale, avec $C = 0.2$. Les deux grilles ont pour pas d'espace respectifs $dx_1 = 1/100$ et $dx_2 = 1/200$, soit un rapport 2 entre leurs finesses respectives. Nous utilisons le noyau d'interpolation M'_4.

Les résultats sont présentés à deux temps différents (t=0.1 sur la figure (4.8) et t= 0.2 sur la figure (4.9)), pour illustrer l'adaptivité du raffinement. Nous n'avons affiché qu'une particule sur deux afin de rendre plus visibles les deux populations de particules. Entre les instants $t = 0.1$ et $t = 0.2$ la grille fine s'est étendue vers la droite pour continuer de recouvrir le choc, et vers la gauche, pour couvrir l'onde de raréfaction. A ces deux instants, la grille fine couvre exactement les zones sur lesquelles il est intéressant de raffiner. La comparaison de ces résultats à ceux obtenus avec une seule grille (figure (3.14)) permet de noter que nulle oscillation parasite n'a été générée par la co-existence des grilles ou leur déplacement. Sur ce test, la méthode de raffinement de grille fonctionne donc exactement comme nous le souhaitions.

4.2.2 Etude numérique de la conservativité

Le tube à chocs de Sod ayant pour conditions aux limites des conditions de type "mur", on sait que la masse totale du fluide contenu dans le domaine de calcul doit être conservée. Elle l'est dans le cas d'une seule grille, puisque le remaillage sur une grille est conservatif et que les particules fictives correspondant aux conditions de mur n'enlèvent ni ne rajoutent de masse à l'intérieur du domaine de calcul. Pour vérifier que l'utilisation de deux grilles ne dégrade pas la conservativité de la méthode, on évalue le rapport $\frac{\text{masse}}{\text{masse initiale}}$ tout au long de la simulation. On évalue la masse totale de la manière indiquée dans la sous-section 4.1.5.

Les résultats sont présentés sur la figure (4.10). La masse totale oscille très légèrement au cours du temps. Cependant les variations sont très petites : inférieures à 0.03 pour cent. Elles indiquent que, si le critère utilisé permet globalement à la grille fine de bien suivre la propagation du choc et de l'onde de raréfaction, la frontière entre les deux grilles se trouve dans une zone non-uniforme. On en conclut tout de même que la technique de multigrilles préserve plutôt bien la propriété de conservativité de la méthode.

4.2.3 Problème de Noh

Ce test reprend le problème de Noh, en coordonnées polaires, avec deux grilles. Une grille fine ($dr1 = 1/400$) couvre initialement le centre, et une grille grossière

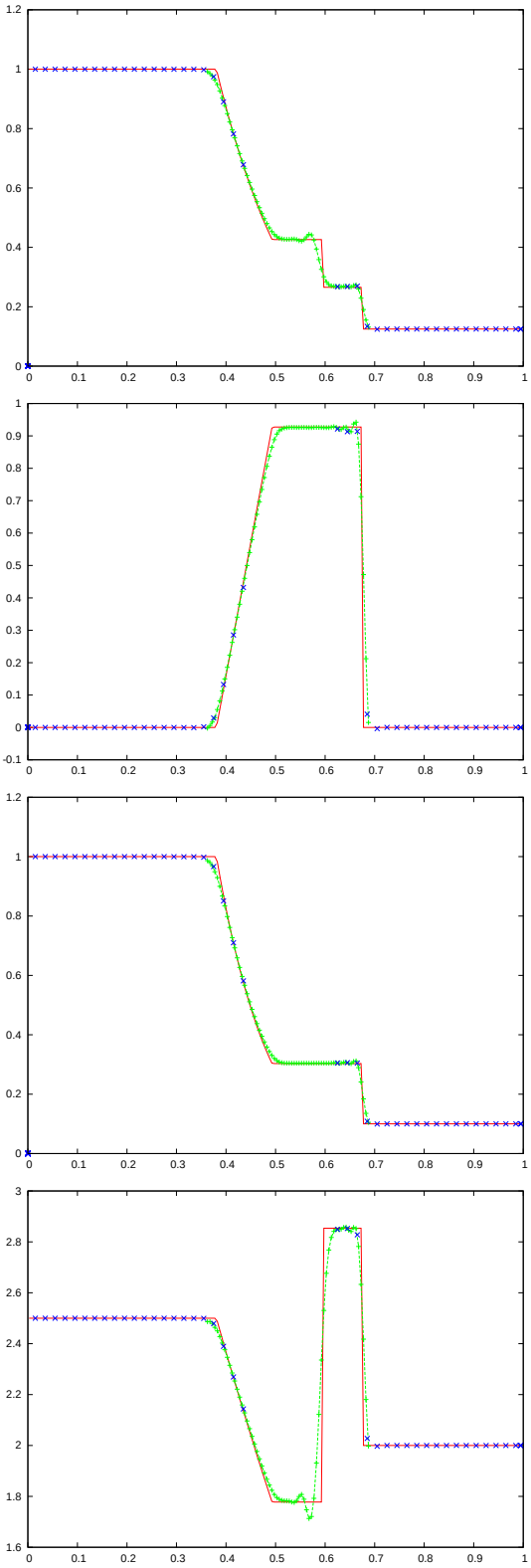


FIG. 4.8 – Tube à chocs de Sod, deux grilles, t = 0.1, adaptation des grilles suivant le critère (4.1), différences finies centrées. De haut en bas : densité, vitesse, pression et énergie interne. Solution exacte en rouge, solution sur la grille grossière en bleu, et solution sur la grille fine en vert. Une particule sur deux affichée.

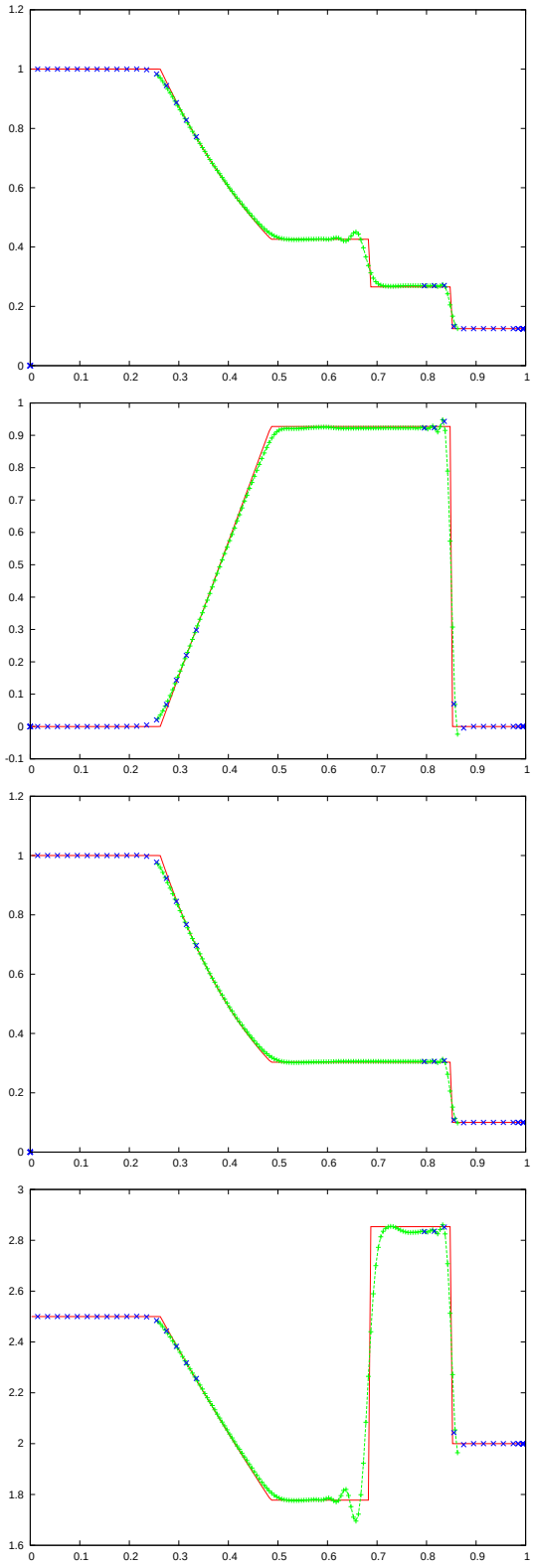


FIG. 4.9 – Même légende que la figure précédente, à t = 0.2.

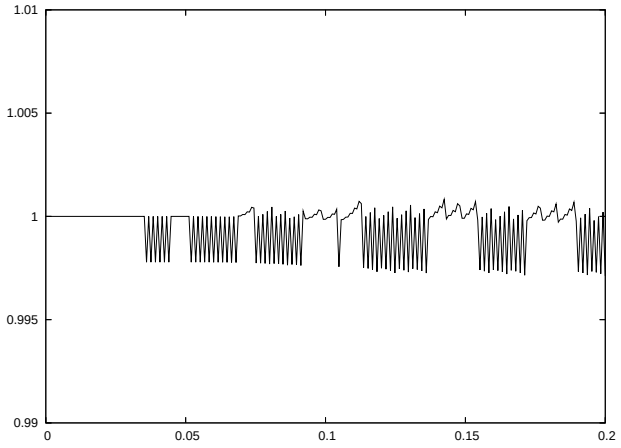


FIG. 4.10 – Tube à chocs de Sod, bigrille adaptative, évolution du rapport masse sur masse initiale en fonction du temps

($dr2 = 1/50$) le reste du domaine. On utilise le critère (4.1) pour la vitesse et la densité, avec le coefficient $C = 0.2$. Les résultats sont présentés sur la figure (4.11), avec de haut en bas et de gauche à droite la densité, la vitesse, la pression et l'énergie interne.

L'utilisation de deux grilles ne crée pas de phénomènes parasites. La grille s'étend de manière à suivre le choc. La limite de la grille fine peut sembler plus éloignée du choc que nécessaire, mais en fait les conditions de recouvrement des grilles imposent cette distance à cause du fort rapport $dr2/dr1$. Précisons que nous n'avons représenté sur ces figures que la zone [0,0.4], bien que le domaine de calcul soit plus grand, car au temps $t = 6$ tout la masse initiale y est contenue. La grille fine ne recouvre donc pas la majorité du domaine de calcul, comme cela pourrait sembler à première vue.

4.2.4 Raffinement en temps : application à l'équation de Bürgers

La technique multi-niveaux peut aussi être appliquée à des calculs avec des pas de temps différents sur des grilles qui possèdent les mêmes pas d'espace. Cela permet d'utiliser des pas de temps adaptés aux différentes zones du domaine de calcul. Nous avons appliqué cette approche au calcul de solutions de l'équation de Bürgers.

Nous avons étudié la formation d'un choc, en prenant des conditions initiales sinusoïdales. Une grille occupe la partie du domaine où le choc se forme et une autre grille le reste du domaine. Chaque grille possède son propre pas de temps, calculé en fonction de la condition de stabilité portant sur le non-croisement des particules. Les valeurs des particules de la grille qui possède les plus grands pas de temps sont interpolées linéairement en temps afin de permettre le remaillage des particules de l'autre grille. La figure (4.12) permet de comparer le résultat de ce calcul avec deux grilles à celui d'un calcul avec une seule grille. Les résultats des deux calculs sont identiques.

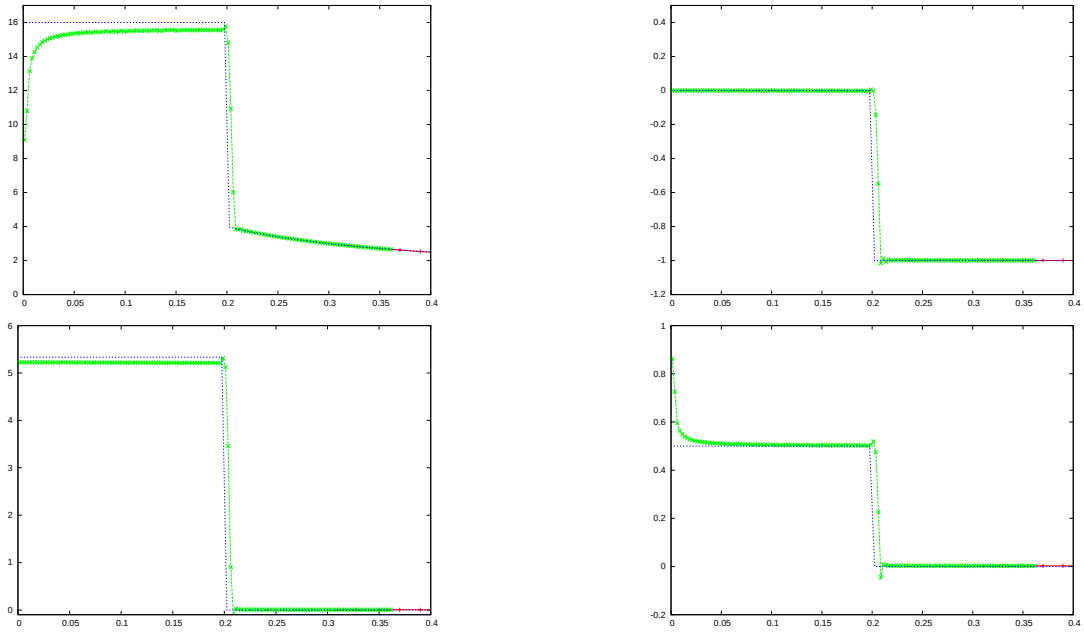


FIG. 4.11 – Problème de Noh, deux grilles, $dx1 = 1/50$, $dx2 = 1/400$, M_4', solveur de Riemann. De haut en bas et de gauche à droite : densité, vitesse, pression et énergie interne. Solution exacte en bleu, solution sur la grille grossière en rouge, et solution sur la grille fine en vert.

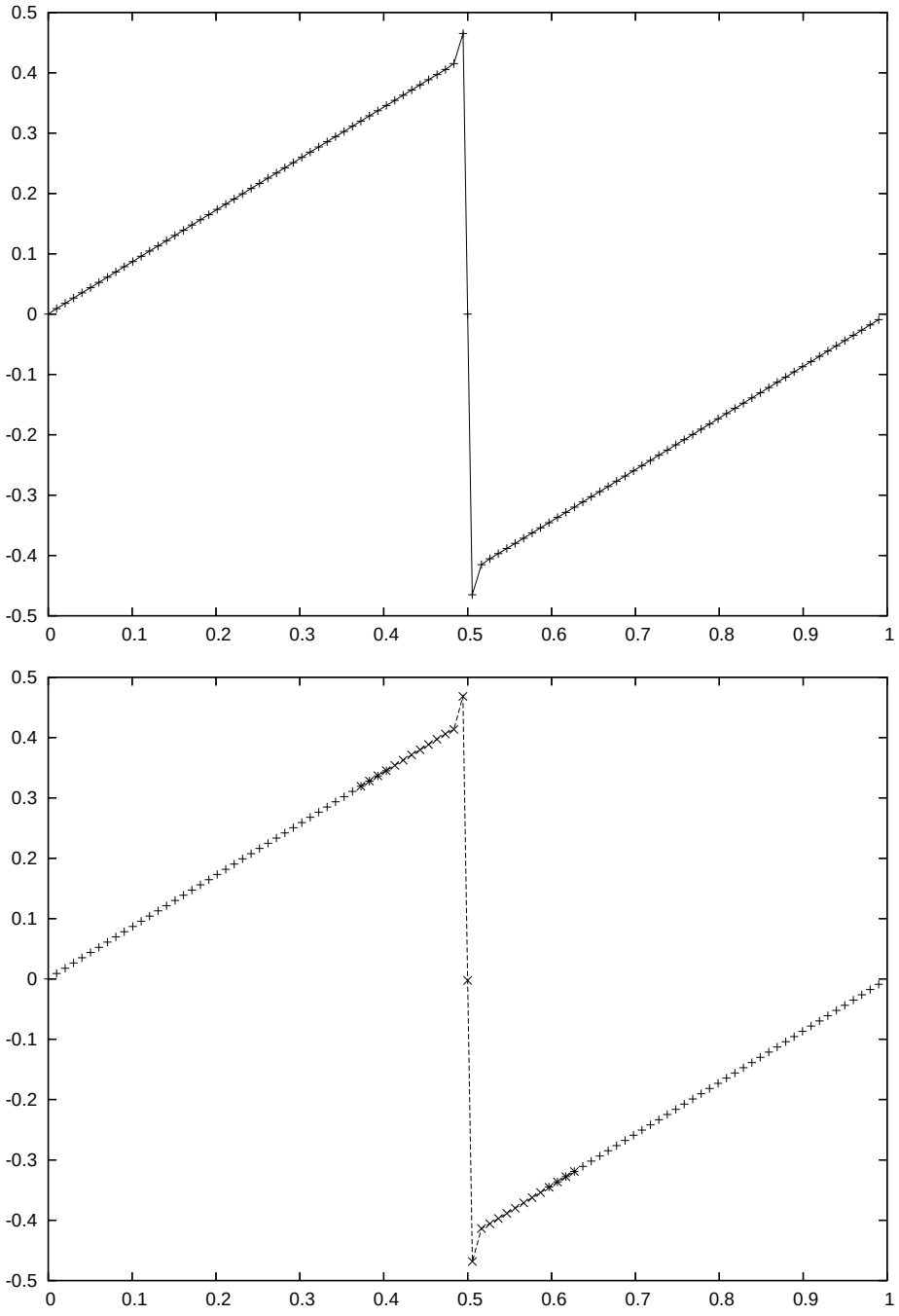


FIG. 4.12 – Equation de Bürgers, formation d'un choc, remaillage avec M_4', en haut le calcul est effectué avec une seule grille, en bas avec deux grilles qui possèdent des pas de temps différents.

4.3 Conclusion du chapitre 4

Dans ce chapitre nous avons implanté une technique multi-niveaux adaptée aux méthodes particulaires avec remaillage. Nous avons rappellé quelles étaient les conditions de recouvrement des grilles nécessaires à un traitement consistant des conditions aux limites de chaque grille. Nous avons proposé des critères d'adaptation des grilles (4.1), basés sur les valeurs des dérivées partielles de la densité, de la vitesse et de l'énergie. Ces critères garantissent à la grille fine de couvrir automatiquement les zones de fort gradient pendant la durée de la simulation.

Nous nous sommes aussi intéressés à la conservativité de la méthode particulaire utilisée avec cette technique multi-niveaux. Nous avons proposé une définition de la masse totale (ou de la quantité totale de toute autre variable extensive) évaluée sur tout le domaine de calcul (4.2), et montré que cette quantité était conservée à l'ordre 2 (proposition 14). Il existe en effet une zone autour de la frontière entre les grilles, dont la largeur est liée à la taille du support du noyau d'interpolation, où la masse n'est conservée qu'à l'ordre 2. Pour que la méthode soit totalement conservative, il faut que la frontière entre les grilles soit située dans une zone où toutes les variables portées par les particules sont constantes.

Nous avons ensuite testé la technique multi-niveaux sur le problème de Sod et le problème de Noh. Cela nous a permis de vérifier que la grille fine suivait effectivement les zones de fort gradient, et que la masse totale était finalement correctement préservée. Nous avons également testé sur l'équation de Bürgers l'utilisation de deux grilles de même pas d'espace, avec juste une adaptation locale du pas de temps, et constaté qu'avec cette technique nous obtenions exactement les mêmes résultats qu'avec une seule grille, tout en utilisant localement des pas de temps plus grands.

La technique multi-niveaux sera utilisée dans le chapitre suivant, dédié à la simulation bi-fluides, afin de résoudre avec plus de précision l'interface entre les différents fluides.

Chapitre 5

Simulation de l'interface entre deux fluides

Dans ce chapitre on cherche à simuler l'interface entre deux fluides non-miscibles régis par les équations d'Euler compressibles. Il s'agit donc d'adapter notre méthode particulaire de manière à simuler des écoulements bifluides. Dans la première partie on présente la stratégie que nous avons adoptée : une méthode level-set discrétisée sur les particules. Dans la deuxième partie on détaille le traitement de la fonction level-set ϕ. Pour améliorer la conservation des masses partielles, qui avec les méthodes level-set est détériorée par la dissipation numérique, nous utilisons la technique multi-niveaux présentée dans le chapitre 4 :
La fonction level-set ϕ est discrétisée sur une grille plus fine superposée à la grille usuelle où sont résolues les équations d'Euler. Cette méthode nous permet d'obtenir une meilleure résolution de l'interface, pour un faible coût supplémentaire, car les particules qui portent ϕ ne sont soumises qu'à la condition de non-croisement des particules, qui est généralement moins restrictive que la condition de stabilité de la phase Lagrange. On implante aussi la méthode multi-niveaux classique, où chaque grille porte toutes les variables, dans le but de résoudre plus précisément les équations d'Euler et l'équation d'évolution de ϕ autour de l'interface.
Dans la troisième partie on présente des tests numériques. tube à chocs de Sod bi-fluide, instabilité de Kelvin-Helmholtz et interaction choc-bulle. On étudie sur les deux derniers les améliorations apportée par les deux types de méthodes multi-niveaux : l'utilisation d'une grille plus fine juste pour la résolution de l'interface améliore notablement la conservation des masses partielles. L'utilisation d'une grille locale plus fine pour résoudre toutes les variables permet d'obtenir autour de l'interface les mêmes résultats que si cette grille avait été utilisée sur tout le domaine.

5.1 Choix d'une méthode de type level-set

Dans cette partie on explique quelle méthode nous avons choisie pour simuler des interfaces entre fluides compressibles.

Parmi les différentes approches envisageables pour simuler des écoulements multi-fluides, on distingue deux catégories. La première regroupe les méthodes qui considèrent de manière effective la présence de fluides différents. Ces méthodes nécessitent donc des relations de fermeture pour résoudre les équations formulées pour chaque fluide. Cette première approche nous a paru difficile à appliquer à la méthode particulaire que nous utilisons. Elle n'est pas incompatible avec la discrétisation particulaire, mais avec le procédé de remaillage que nous utilisons. Celui-ci crée de nouvelles particules qui sont chacune la somme des contributions des anciennes particules voisines. Il ne préserve donc pas la séparation entre les fluides prescrite par la première approche. La deuxième approche regroupe les méthodes qui ne considèrent, d'un point de vue numérique, qu'un seul fluide, dont les propriétés varient spatialement. Elle nous a paru beaucoup plus simple à appliquer à notre méthode. Elle consiste à considérer un seul type de particules qui portent des valeurs différentes suivant le fluide où elles se trouvent. Il n'y a alors aucun problème pour traiter l'interpolation.

Parmi les méthodes correspondant à cette seconde approche, notre choix s'est porté sur la méthode level-set. Ce type de méthode résoud implicitement l'interface entre deux fluides en calculant l'évolution d'une fonction ϕ, régie par l'équation :

$$\frac{\partial \phi}{\partial t} + \vec{U}.\vec{\nabla}\phi = 0$$

où $\vec{U}$ représente la vitesse de l'écoulement. L'interface est initialement représentée par l'ensemble des points où ϕ s'annule. Comme ϕ est transportée à la vitesse de l'écoulement, cet ensemble de points continue de représenter l'interface au cours du temps. Cette méthode est très facile à implanter, puisqu'il suffit de résoudre une équation supplémentaire. Elle présente également l'avantage de traiter automatiquement les changements topologiques, tels qu'une fragmentation ou une coalescence. Le fait de discrétiser la fonction ϕ sur des particules permet en outre de bénéficier pour la résolution de ϕ de la prise en compte lagrangienne des termes de transport offerte par les particules. C'est d'ailleurs pour ces raisons que des méthodes particulaires ont déjà été associées à des méthodes level-set [19, 48, 30].

5.2 Implantation de la méthode

Dans cette partie sont décrits les détails d'utilisation de la méthode de type level-set que nous avons choisie pour simuler les interfaces entre fluides : traitement de la fonction ϕ, évaluation des masses partielles, et utilisation de la technique multi-niveaux du chapitre 4 pour améliorer la résolution de l'interface et la conservation des masses partielles.

5.2.1 Traitement de la fonction ϕ

Il est fréquent, dans le cadre des méthodes level-set, d'utiliser pour ϕ la fonction distance signée. Nous avons préféré choisir une fonction qui ne varie que dans une zone proche de l'interface. Cela nous permet de ne discrétiser ϕ que dans cette zone, et simplifie la création de particules fictives au moment du remaillage. Nous donc avons pris pour valeur initiale de ϕ :

$$\begin{aligned}\phi(x) &= 1 \text{ si } d \geq \epsilon \\ &= -1 \text{ si } d \leq -\epsilon \\ &= \frac{1}{2}(1 + cos(\frac{\pi d}{\epsilon})) \text{ sinon}\end{aligned}$$

où d représente la distance entre x et l'interface, multipliée par -1 si x se trouve dans le deuxième fluide, et où ϵ est un paramètre de longueur, valant environ deux pas d'espace.

La fonction ϕ est discrétisée sur les particules, chaque particule p porte donc une valeur ϕ_q. Cette variable ne nécessite pas de traitement particulier pendant la résolution de la phase Lagrange. Par contre, il faut au moment du remaillage l'interpoler sur les cellules de la grille pour transmettre aux nouvelles particules l'information qu'elle porte sur l'interface. Or ϕ n'est pas une variable extensive comme la masse ou la quantité de mouvement. Il ne suffit donc pas de l'interpoler telle quelle sur la grille. Si nous le faisions, la valeur de ϕ augmenterait dans les zones de compression et diminuerait dans les zones de détente, indépendamment du fluide dans lequel elle se trouverait. Cela pourrait après un certain nombre d'itérations faire apparaître de nouveaux zéros de la fonction ϕ, ce qui n'est pas souhaitable. Par ailleurs, il est commode d'utiliser ϕ pour évaluer la valeur de variables comme γ en un point donné. On peut par exemple décider, et c'est ce que nous faisons, que :

$$\gamma_q = \frac{(1+\phi_q)}{2}\,\gamma_1 + \frac{(1-\phi_q)}{2}\,\gamma_2$$

où on note γ_1 et γ les valeurs de γ dans le premier et le deuxième fluide. Si ϕ prend des valeurs supérieures à 1 ou inférieures à -1, γ prendra des valeurs qui ne seront pas pertinentes. Il est donc préférable d'interpoler ϕ d'une manière qui préserve l'allure de cette fonction.

Nous avons donc choisi d'interpoler la valeur de ϕ portée par la particule, multipliée par son volume calculé à la fin du pas de temps, c'est-à-dire non pas le volume V_i de la cellule où se trouvait initialement la particule, mais le volume $\tilde{V}_i$ tenant compte du déplacement de la matière au cours du pas de temps. On calcule donc, sur chaque cellule i :

$$\phi_i^{n+1} = \frac{1}{V_i}\sum_j \phi_j^n \tilde{V}_j^n\,\Lambda(\frac{x_i - \tilde{x_j}}{dx})$$

L'interpolation a les mêmes effets dispersifs et dissipatifs sur ϕ que sur les autres variables. Au fil du temps des oscillations apparaissent sur le profil de ϕ, tandis qu'augmente la largeur de la zone de mélange numérique, c'est-à-dire la zone où la valeur absolue de ϕ est inférieure à 1. Pour remédier à l'apparition d'oscillations, on pratique le seuillage, introduit dans la section 3.8 du chapitre 3 : on force les valeurs de ϕ des nouvelles particules issues de l'interpolation à être comprises entre le maximum et le minimum des valeurs des particules qui ont contribué à leur création. Pour lutter contre l'étalement progressif de cette zone de mélange numérique, il est possible de renormaliser la fonction de phase [19], mais nous ne l'avons pas fait dans ce travail.

5.2.2 Evaluation et conservation des masses partielles

Le fait de considérer numériquement un seul fluide au lieu de plusieurs simplifie le traitement des systèmes multi-fluides mais pose un problème si on cherche à calculer les masses partielles des différents constituants. Dans notre cas, la méthode level-set crée une zone de mélange numérique, où ϕ est strictement comprise entre -1 et 1, et où les particules sont composées des deux fluides. La seule information sur les proportions respectives de ces deux fluides dans les particules provient de la variable ϕ. Celle-ci ne peut pas vraiment être considérée comme une fraction volumique. Comment peut-on donc définir des masses partielles ?

Une idée consiste à distinguer les fluides selon le signe de ϕ. Toutes les particules qui portent une valeur de ϕ positive seront considérées comme appartenant au premier fluide, et toutes les autres au second. Si on note $masse_1$ et $masse_2$ les masses partielles des deux fluides, cela donne :

$$\text{Si } \phi_q > 0 \text{ alors } masse_1 = masse_1 + m_q$$

$$\text{Si } \phi_q \leq 0 \text{ alors } masse_2 = masse_2 + m_q$$

Nous étudierons dans la prochaine section de ce chapitre la conservation des masses partielles calculée avec cette méthode.

Le principal défaut reproché aux méthodes level-set est leur manque de conservativité, puisque la conservation des masses partielles des différents fluides n'est pas assurée. Cela est dû à la diffusion numérique subie par la fonction de phase ϕ. Différentes méthodes ont été introduites pour combattre ce problème. Toutes cherchent soit à augmenter la précision de la représentation de l'interface par la fonction ϕ, soit à la maintenir au cours du calcul . Certains couplent la méthode level-set à une méthode de type Volume of Fluid [83], ou à une méthode de front-tracking [30, 31], avec des particules lagrangiennes qui permettent de reconstruire plus précisément l'interface. Ces techniques améliorent de fait la conservation des masses partielles, mais augmentent la complexité globale de la méthode, et peuvent introduire des oscillations dans la courbure de l'interface. Une autre solution consiste à utiliser une discrétisation plus fine

autour de l'interface, afin d'en obtenir une meilleure résolution. Dans le cadre de cette stratégie, la technique de remaillage adaptatif présentée dans le chapitre précédent peut être utilisée pour essayer d'obtenir une meilleure conservation des masses partielles.

5.2.3 Utilisation de la technique multi-niveaux

Avec la technique de remaillage adaptatif, si on veut résoudre plus précisément l'interface entre deux fluides, il faut placer une grille plus fine autour de cette interface. Comme la fonction ϕ varie rapidement autour de l'interface, il est possible d'utiliser le même critère que celui (4.1) présenté dans le chapitre précédent, appliqué à ϕ. Un autre choix, plus simple, est possible, qui consiste à déplacer la grille fine en fonction de la vitesse sur ses bords. Les limites des grilles sont sont déplacées à la vitesse de la particule/cellule de la grille sur laquelle elles se trouvent. Ce choix évite d'avoir à estimer les dérivéees partielles de ϕ. Nous l'avons utilisé lors de nos tests numériques.

Il est possible d'utiliser la grille fine de deux manières différentes : la première consiste à discrétiser toutes les variables des équations d'Euler, ainsi que la fonction ϕ, sur la grille fine. La zone située autour de l'interface est alors entièrement résolue avec la grille plus fine. La deuxième possibilité consiste à discrétiser uniquement ϕ sur la grille fine. Dans ce cas, il est nécessaire que la grille grossière couvre tout le domaine afin d'y résoudre les équations d'Euler. Le champ de vitesse est interpolé de la grille grossière vers la grille fine pour pouvoir déplacer les particules fines. Cette technique ne permet pas de résoudre les équations d'Euler aussi précisément que si toutes les variables étaient discrétisées sur la grille fine, mais on espère qu'elle peut, en améliorant la résolution de l'interface, améliorer la conservation des masses partielles. Elle n'est pas très coûteuse, car le même pas de temps peut être utilisé sur les deux grilles. En effet, les particules fines ne font que se déplacer et ne sont donc soumises qu'à la condition de stabilité portant sur le non-croisement des particules, qui est en général moins restrictive que celle correspondant à la résolution de la phase Lagrange. Pour évaluer avec précision les masses partielles des différents fluides, il convient de résoudre aussi la densité sur cette grille fine. Comme cette variable nécessite juste d'être interpolée, et que les coefficients de l'interpolation sont déjà calculés pour ϕ, le calcul de l'évolution de la densité n'ajoute qu'un faible coût supplémentaire.

Au cours de la validation numérique, nous comparerons ces deux manières d'utiliser la grille fine et leurs conséquences pour l'évaluation des masses partielles.

5.3 Validation numérique

Dans cette partie on présente des tests numériques : tube à chocs de Sod bifluide, instabilité de Kelvin-Helmholtz et interaction choc-bulle. On étudie sur ces tests l'utilisation de la technique multi-niveaux et la conservation des masses partielles.

5.3.1 Tube à chocs de Sod bifluide

On reprend ici le problème classique du tube à chocs, mais cette fois les fluides situés de part et d'autre de la discontinuité initiale ont des valeurs de leurs coefficients γ différentes :

$$\begin{aligned} \gamma &= 1.4 \text{ si } x \leq 0.5 \\ &= 2.4 \text{ sinon} \end{aligned}$$

Ce test est réalisé avec la fonction d'interpolation Λ_3. On utilise des différences finies centrées d'ordre 3 et le terme de viscosité artificielle (3.5) avec un coefficient qui vaut 0.8 dans les zones de compression et 0.2 ailleurs.

La figure (5.3.1) présentent les résultats obtenus avec un pas d'espace $dx = 1/200$, comparés à la solution exacte. On remarque que le choc est décalé par rapport à la solution exacte. Une petite oscillation est visible sur les profils de vitesse et de pression sur la position de la discontinuité de contact, mais elle est très légère.

Sur la figure (5.3.1) est représentée l'évolution du rapport masse/masse initiale, calculé avec la méthode définie dans la section 5.2.1, du fluide situé à droite de la discontinuité de contact, pour différentes discrétisations : $dx = 1/50, 1/100, 1/200, 1/400$. Comme la méthode conserve la masse totale des fluides (car l'interpolation est conservative et les conditions aux limites sont de type mur), la conservation de la deuxième masse partielle se déduit de la première. La masse partielle augmente fortement pendant les premiers pas de temps de la simulation, puis oscille autour d'une valeur moyenne à-peu-près constante. La conservation de la masse s'améliore avec l'augmentation du nombre de particules. L'amplitude des variations de la masse partielle semble d'ailleurs être inversement proportionnelle au nombre de particules utilisées. Sur la figure 5.3.1 sont représentés les profils de la densité et de ϕ obtenus avec la méthode particulaire, ainsi que la solution exacte pour la densité. Comparer la position de la discontinuité de contact à l'abcisse de la valeur zéro de ϕ nous permet de constater que notre interface numérique est bien placée.

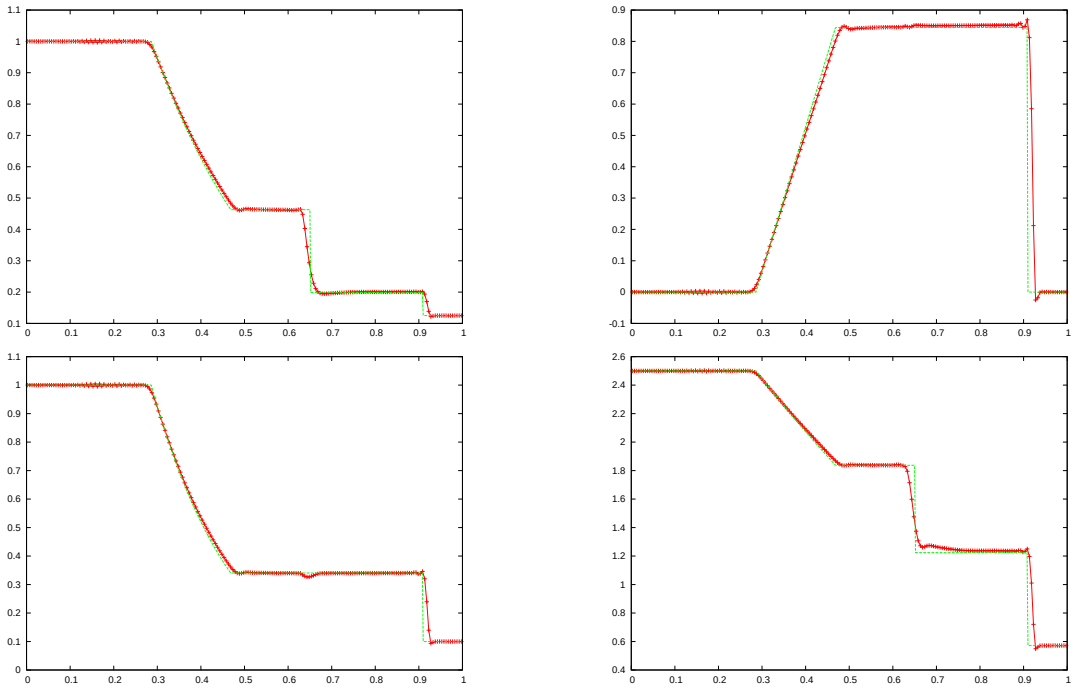


FIG. 5.1 – Tube à chocs de Sod avec $\gamma = 1.4$ à gauche de la discontinuité initiale, et $\gamma = 2.4$ à droite, N = 200, remaillage avec le noyau Λ_3, différences finies centrées d'ordre 3, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne.

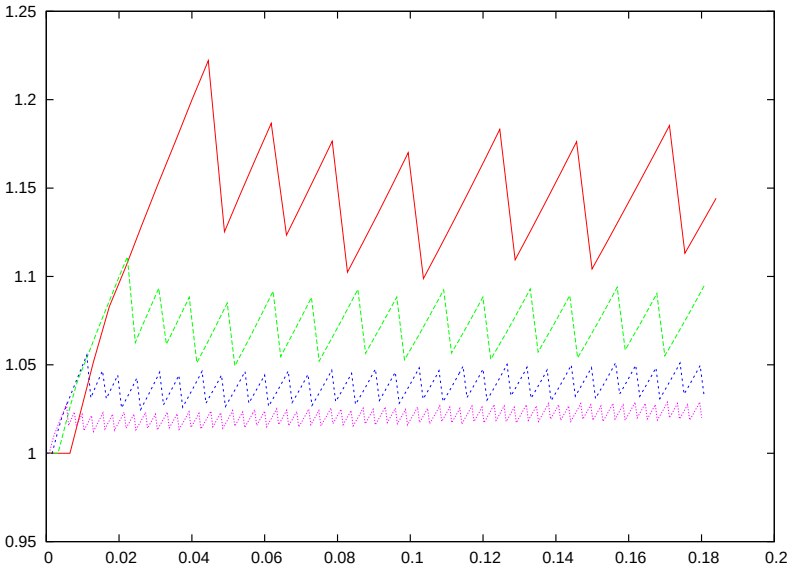


FIG. 5.2 – Même légende que la figure précédente. Evolution du rapport $\frac{\text{masse}}{\text{masse initiale}}$ du fluide situé à droite, pour dx = 1/50 (rouge), 1/100 (vert), 1/200 (bleu), 1/400 (rose).

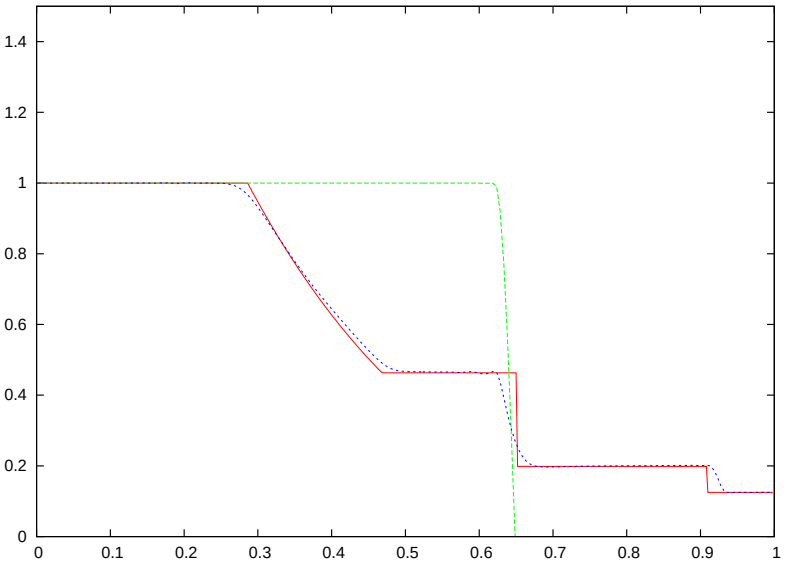


FIG. 5.3 – Même légende que la figure précédente. Densité (bleu) et ϕ (vert) obtenues avec la méthode, comparées à la solution exacte de la densité (rouge).

5.3.2 Instabilité de Kelvin - Helmholtz

L'instabilité de Kelvin-Helmholtz se produit lorsqu'il existe un cisaillement à l'interface de deux fluides, ou lorsqu'un jet de fluide est injecté dans un deuxième fluide. Les méthodes particulaires de type SPH classiques ne sont pas capables de résoudre correctement ce type d'instabilité (voir [3]). On prend ici comme domaine de calcul $[0,1] \times [-0.5, 0.5]$. Nous disposions de deux références bibliographiques sur cette instabilité. Ces références avaient des conditions initiales différentes et donnaient des résultats assez différents. Nous avons donc testé les deux jeux de conditions initiales. Le premier est celui proposé par Mulder et al. [65], qui utilisent aussi une méthode level-set :

$$\begin{aligned} \rho(x,y,0) &= 1. \\ U_x(x,y,0) &= 0.25 \text{ si } y \geq 0.1\ sin(2\pi x - \pi/2) \\ &= -0.25 \text{ sinon} \\ U_y(x,,y,0) &= 0. \\ p(x,y,0) &= 1. \end{aligned}$$

Le deuxième est proposé par Després et Lagoutière [26], qui utilisent un schéma eulérien avec des concentrations massiques de chaque fluide et des relations de fermeture. Leur jeu de conditions initiales diffère surtout du premier par l'amplitude de la perturbation initiale.

$$\begin{aligned} \rho(x,y,0) &= 1. \\ U_x(x,y,0) &= -0.25 \text{ si } y \geq 0.025\ sin(2\pi x) \\ &= 0.25 \text{ sinon} \\ U_y(x,,y,0) &= 0. \\ p(x,y,0) &= 1. \end{aligned}$$

Le coefficient γ vaut 5/3. On utilise des différences finies centrées d'ordre 2, le noyau d'interpolation Λ_3, et le terme de viscosité artificielle (3.5) avec un coefficient qui vaut 1.6 dans les zones de compression et 0.4 ailleurs.

5.3.2.1 Résolution sur une seule grille

Les figures 5.4 et 5.5 présentent les résultats obtenus avec les conditions initiales de Mulder et al., avec une grille 128^2 pour avoir la même résolution que ces derniers, aux temps $t = 1, 2, 3, 4, 5, 6, 7, 8$. On compare nos résultats à ceux de ces auteurs, représentés sur la figure 5.6. Il n'existe pas de solution analytique à laquelle se référer, mais les instabilités obtenues ont des structures très semblables à celles de Mulder et al.. Précisons que si nos instabilités ont l'air plus tassées que les leurs, c'est à cause de l'échelle que nous avons utilisée, qui n'est pas la même pour les deux axes. Les calculs de Mulder et al. sont effectués dans une boîte deux fois plus haute que large, ce qui

n'est pas notre cas, et cela a probablement un impact sur les instabilités secondaires derrière le tourbillon principal. Notre méthode particulaire est donc tout à fait capable de résoudre l'instabilité de Kelvin-Helmholtz, ce qui n'est pas le cas des méthodes SPH classiques. Sur la figure 5.4 nous avons voulu représenter, à titre de comparaison, les mêmes isolignes de ϕ que celles des résultats de Mulder et al. Comme ceux-ci utilisent pour ϕ la fonction distance signée, leurs isolignes sont initialement beaucoup plus écartées que les nôtres, et la comparaison n'est guère possible. A ce titre, la figure 5.5, qui présente plus d'isolignes, permet mieux de visualiser la manière dont la fonction ϕ évolue au cours de la simulation.

La figure (5.7) présente les résultats obtenus avec les conditions initiales de Després et Lagoutière au temps : $t = 8.$, pour différentes discrétisations : 50×50, 100×100 et 200×200. Les structures des instabilités que nous obtenons sont très semblables à celles des résultats obtenus par Després et Lagoutière, représentés sur la figure 5.8. Nous obtenons, avec la grille 100^2, soit 10 000 cellules, un résultat très proche de celui de Després et Lagoutière pour 250 000 cellules. Comme ces derniers, nous constatons que des échelles de plus en plus petites apparaissent quand on augmente le nombre d'éléments de discrétisation. Sur la figure 5.9 est représentée pour les différentes grilles l'évolution au cours du temps du rapport masse/masse initiale du fluide initialement situé à droite de l'interface. Excepté pendant le dernier tiers du calcul effectué sur la grille 50×50, ce rapport ne varie pas de plus de 0.5 pour cent au cours des calculs. L'utilisation d'une grille plus fine diminue l'amplitude des oscillations des valeurs de ce rapport, mais ne change pas beaucoup l'amplitude de ses variations moyennes. L'utilisation d'une grille plus fine pour résoudre toutes les variables n'améliore donc pas dans ce cas la conservation des masses partielles. Cela est compréhensible dans la mesure où l'utilisation d'une grille plus fine fait apparaître des échelles plus petites. L'apparition de ces petites échelles compense le fait que ϕ est résolu sur une grille plus fine. Pour savoir si l'utilisation d'une discrétisation plus fine améliore la conservation des masses partielles, on peut par contre comparer l'évolution de ces dernières en utilisant, pour une résolution de l'hydrodynamique donnée, une discrétisation de plus en plus fine de ϕ. C'est ce que nous ferons dans la section suivante.

5.3.2.2 Utilisation d'une grille plus fine pour la résolution de l'interface

Nous utilisons maintenant une grille plus fine pour résoudre la fonction ϕ, superposée à la grille courante où sont résolues les variables des équations d'Euler. Nous utilisons uniquement les conditions initiales de Després et Lagoutière, car celles-ci font apparaître plus de petites échelles que celles de Mulder et al., et une amélioration de la conservation des masses partielles sera donc plus significative.

La figure (5.10) présente les résultats obtenus pour le noyau Λ_3, avec des discrétisations $dx_1 = 1/50, dx_2 = 1/100$ et $dx_1 = 1/100, dx_2 = 1/200$. Pour un même nombre de particules utilisées pour discrétiser les variables des équations d'Euler, les interfaces

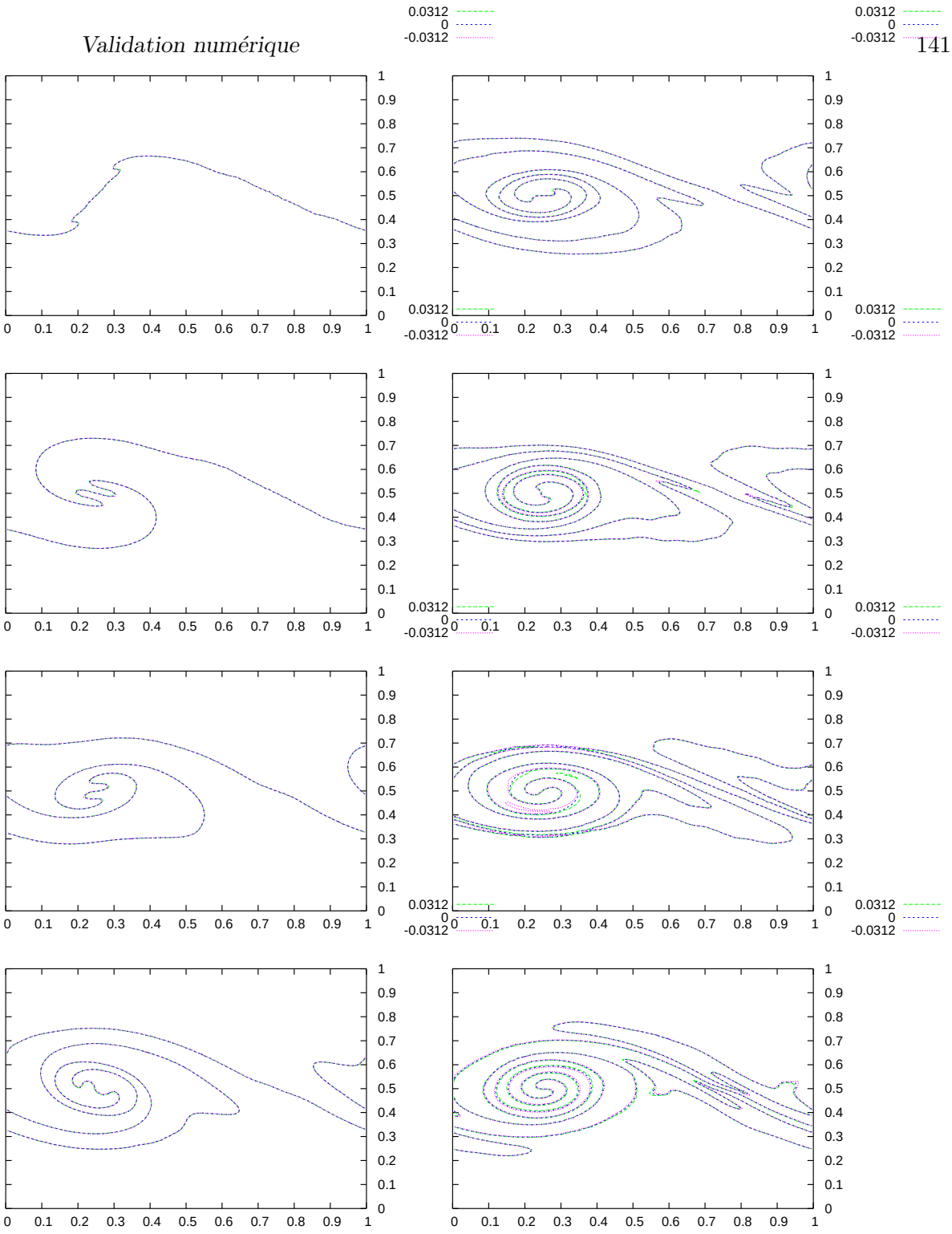


FIG. 5.4 – Instabilité de Kelvin Helmholtz, résultats avec la méthode particulaire pour les conditions initiales de Mulder et al., isolignes de ϕ : -1/32, 0., 1/32, de haut en bas et de gauche à droite aux temps $t = 1., 2., 3., 4., 5., 6., 7., 8.$.

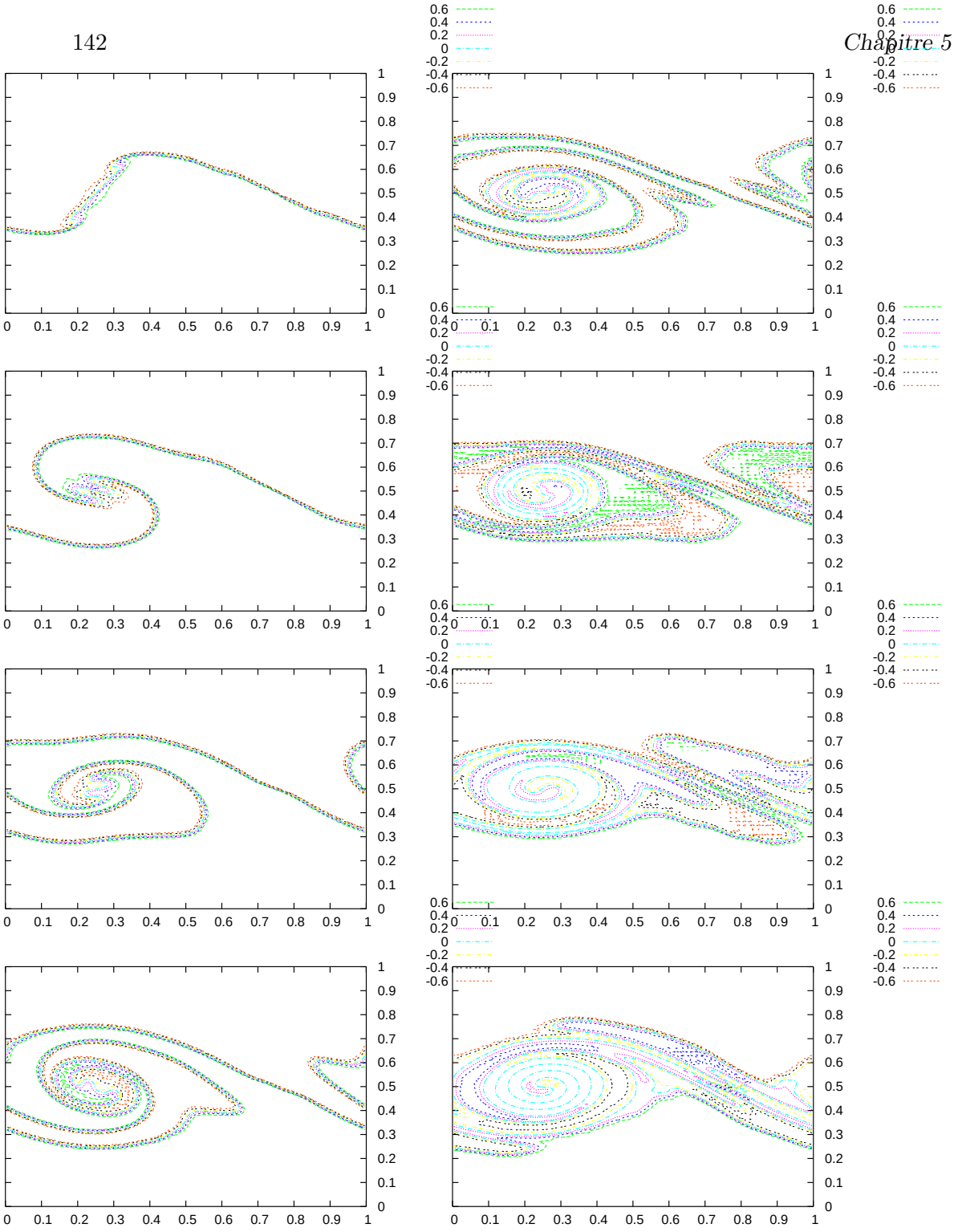


FIG. 5.5 – Instabilité de Kelvin Helmholtz, résultats avec la méthode particulaire pour les conditions initiales de Mulder et al., isolignes de ϕ : -0.6, -0.4, -0.2, 0., 0.2, 0.4, 0.6, de haut en bas et de gauche à droite aux temps $t = 1., 2., 3., 4., 5., 6., 7., 8.$.

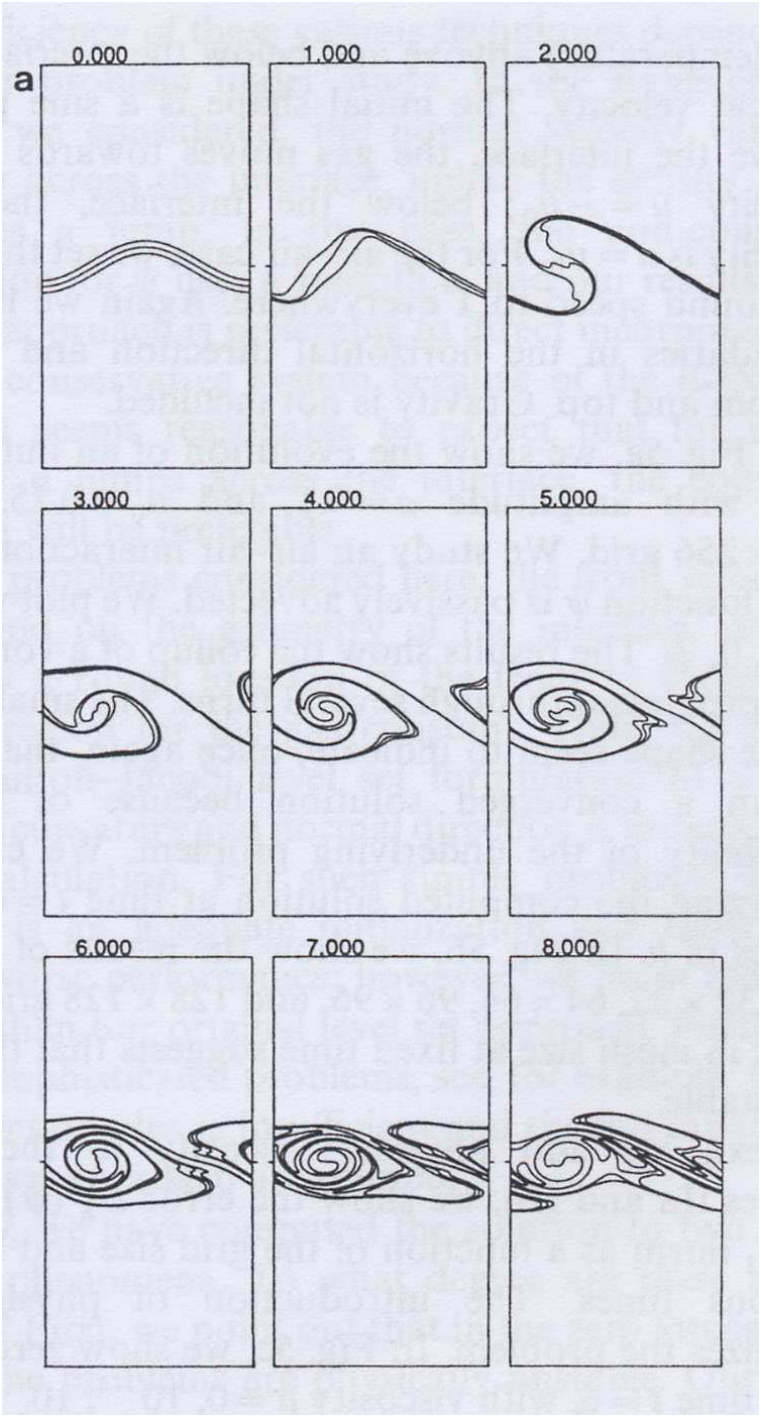


FIG. 5.6 – Résultats de Mulder et al. pour l'instabilité de Kelvin-Helmholz : 128*256 points, isolignes -1/32, 0, 1/32 pour la fonction level-set.

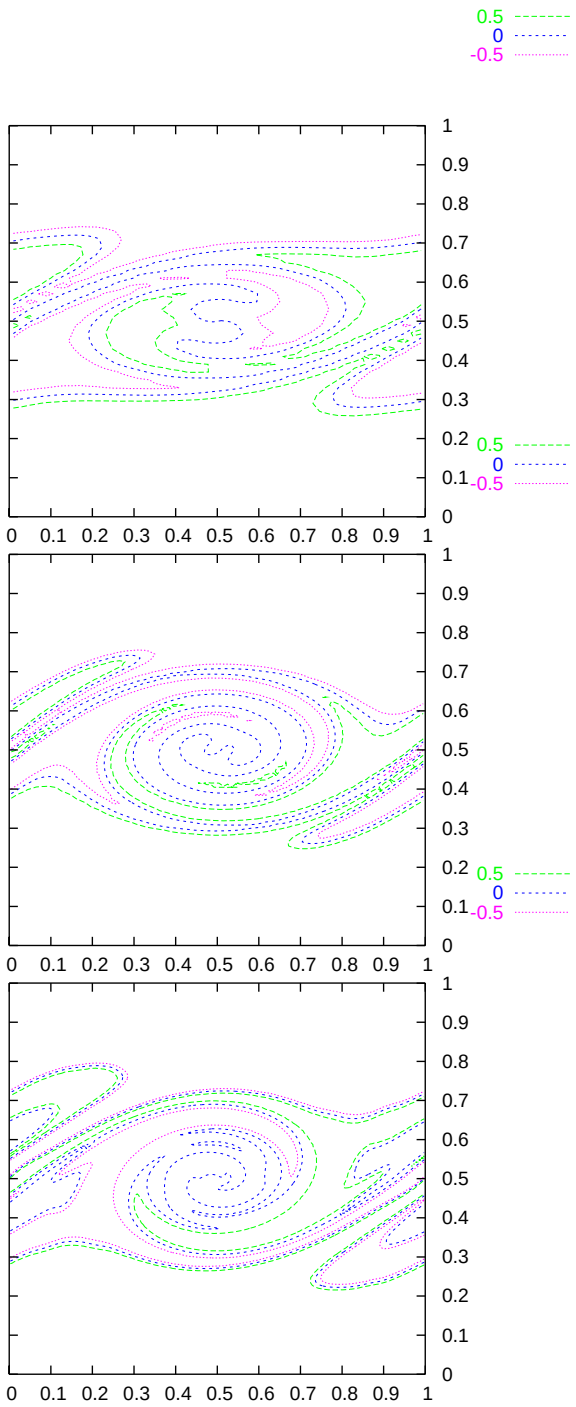


FIG. 5.7 – Instabilité de Kelvin-Helmholtz, résultats avec la méthode particulaire pour les conditions initiales de Després et Lagoutière, isovaleurs de ϕ, à $t = 8.$, remaillage avec Λ_3, de haut en bas : 50^2, 100^2 et 200^2 particules,.

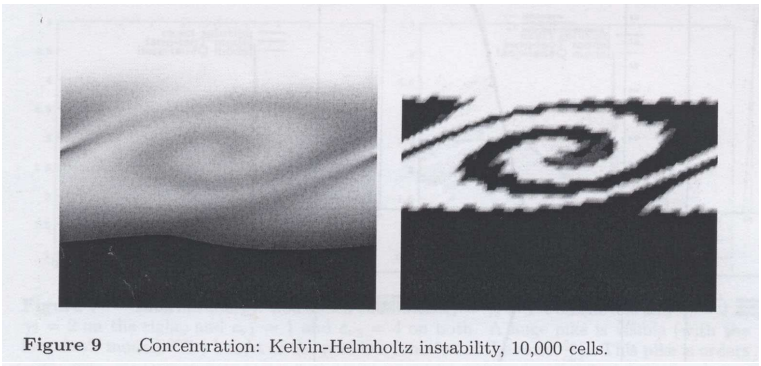


Figure 9 Concentration: Kelvin-Helmholtz instability, 10,000 cells.

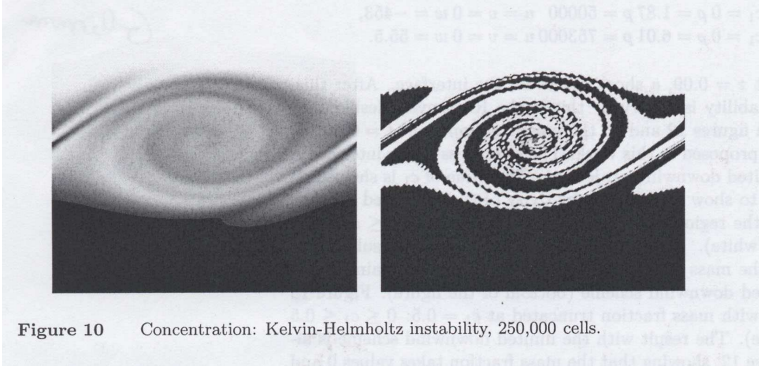


Figure 10 Concentration: Kelvin-Helmholtz instability, 250,000 cells.

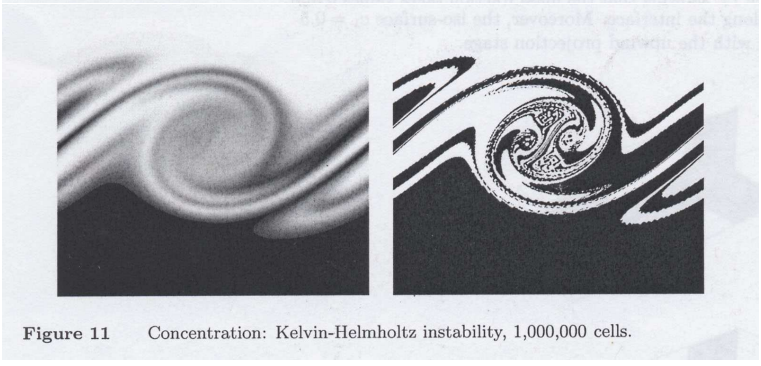


Figure 11 Concentration: Kelvin-Helmholtz instability, 1,000,000 cells.

FIG. 5.8 – Résultats de Després et Lagoutière pour l'instabilité de Kelvin Helmholtz au temps $t = 8$.

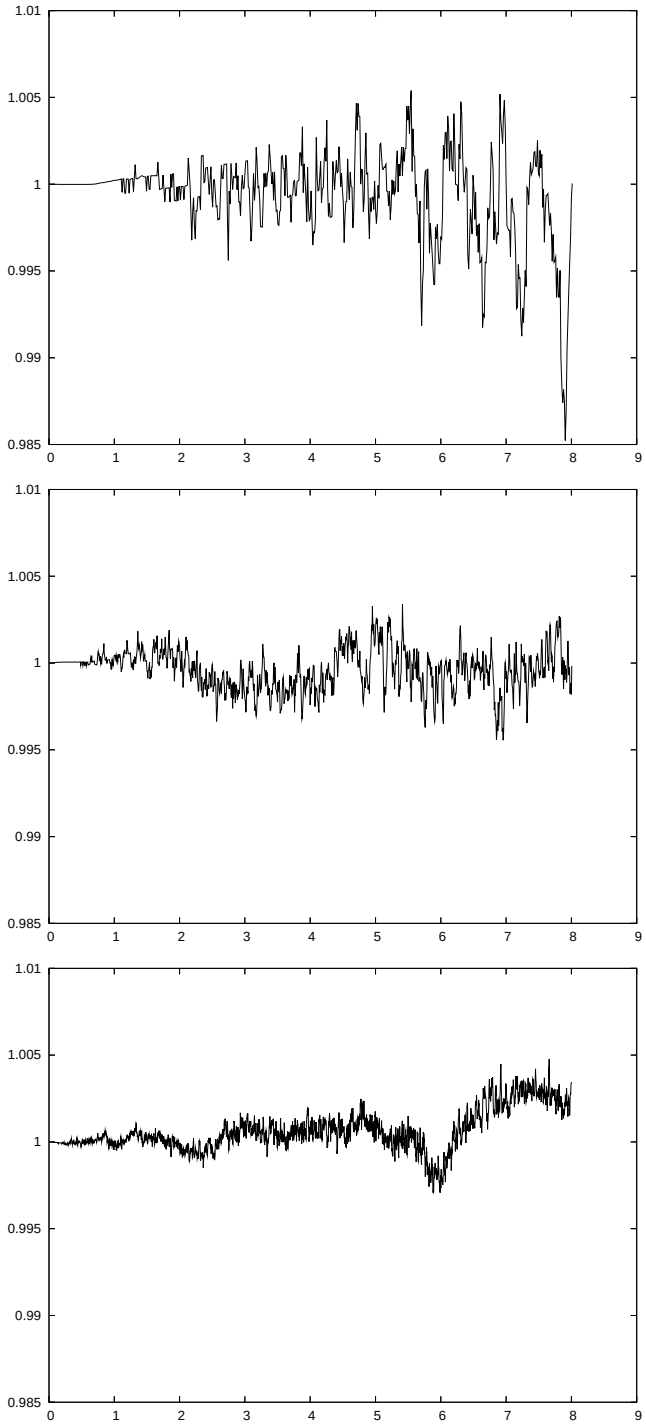


FIG. 5.9 – Instabilité de Kelvin-Helmholtz, résultats avec la méthode particulaire pour les conditions initiales de Després et Lagoutière, évolution de la masse du fluide situé initialement à droite, de haut en bas : grilles de taille 50×500, 100×1000 et 200×2000.

sont plus précises quand elles sont calculées avec une fonction level-set discrétisée sur un plus grand nombre de particules. La largeur entre les isolignes $\phi = -0.5$ et $\phi = 0.5$ est alors plus petite. La figure 5.11 présente l'évolution du rapport masse/masse initiale du fluide situé à droite de l'interface. L'amplitude des oscillations diminue quand on utilise une grille plus fine pour résoudre ϕ, ainsi que l'amplitude des variations moyennes du rapport masse/masse initiale. L'utilisation d'une grille plus fine uniquement pour la résolution de l'interface permet donc de mieux conserver les masses partielles. Nous avons mesuré le temps de calcul nécessaire pour simuler la formation de l'instabilité de Kelvin-Helmholtz jusqu'au temps $t = 8$. Pour un calcul sur une unique grille portant 50^2 particules, ce temps est de 5 s. Il est de 318 s pour une unique grille portant 200^2 particules, et de seulement 49 s si on utilise une grille grossière 50^2 pour résoudre les variables d'Euler et une grille fine 200^2 pour résoudre l'interface, ce qui représente un gain significatif.

5.3.3 Interaction choc-bulle

On étudie ici l'interaction entre un choc et une bulle d'hélium placée dans de l'air. Nous avons repris les conditions initiales de Marquina et Mulet [60]. Cette interaction entre un choc et une bulle a auparavant été étudiée par Haas and Sturtevant [37] et Quirck et Karni [74]. On utilise des différences finies centrées d'ordre 2, le noyau d'interpolation Λ_3 et le terme de viscosité artificielle (3.5) avec un coefficient qui vaut 1.6 ou 0.4 suivant qu'il y ait ou non compression. Les calculs sont effectués d'abord avec une seule grille et une seule taille de particules, puis avec une grille fine placée autour de la bulle. Les particules associées à cette grille fine portent soit uniquement la fonction ϕ et la densité ρ, soit également toutes les autres variables des équations d'Euler.

La figure 5.12 présente le résultat obtenu avec une grille 50×500, sur tout le domaine de calcul, afin d'avoir une vue d'ensemble de la manière dont la bulle évolue. Comme le domaine de calcul est large et la zone qui nous intéresse beaucoup plus petite, on affichera par la suite les résultats uniquement dans la zone autour de l'interface. On présente à chaque fois les résultats aux temps : $t = 0.5, 1.0, 1.5$.

5.3.3.1 Résolution sur une seule grille

On commence par résoudre ce problème avec une seule taille de particules, et une seule grille. Les figures (5.13) à (5.15) présentent les différents résultats obtenus. Avec les grilles 50×500 et 100×1000 apparaissent des instabilités numériques sur la tige de la structure issue de la bulle initiale. Ces instabilités sont probablement dues à la discrétisation initiale de la bulle (cylindrique) sur une grille cartésienne. Elles s'estompent quand le pas d'espace diminue, comme c'est le cas pour les résultats avec la grille 200×2000.

La figure 5.16 présente l'évolution du rapport masse/ masse initiale de la bulle, calculée avec les différentes grilles. Pour chacune des simulations, ce rapport reste constant

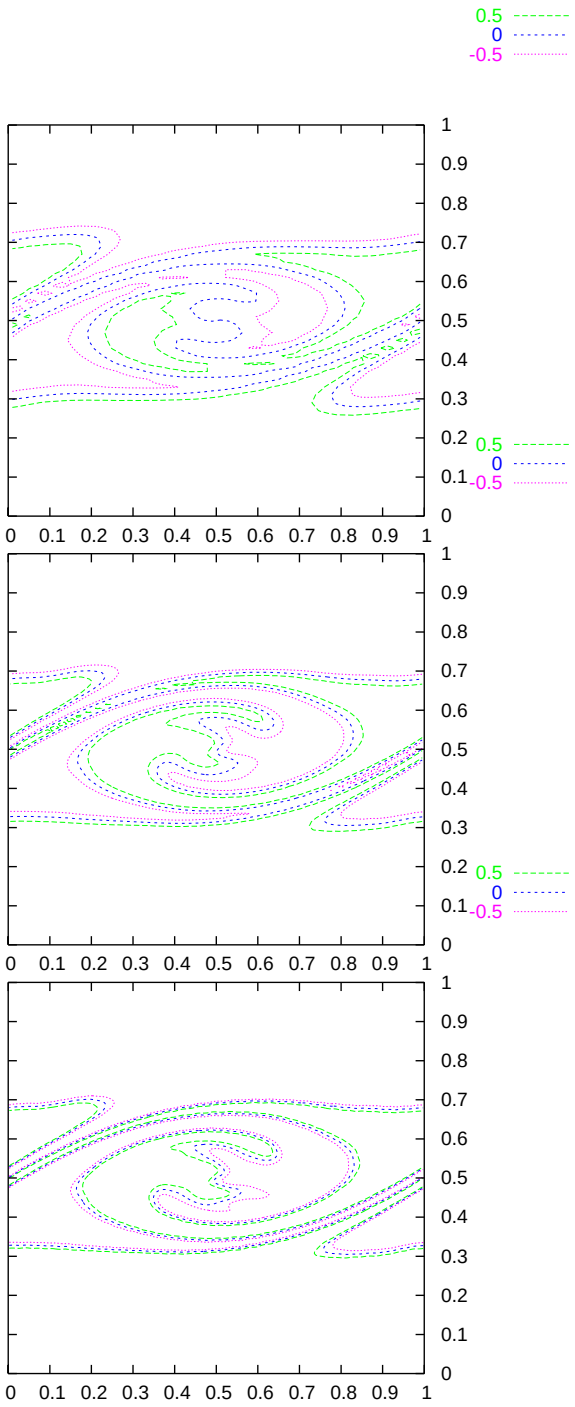


FIG. 5.10 – Instabilité de Kelvin-Helmholtz, isovaleurs de ϕ, à $t = 8.$, de hqut en bas, 50^2 particules, 50^2 particules sur la grille grossière et 100^2 particules sur la grille fine, qui porte seulement ϕ, et 50^2 particules sur la grille grossière et 200^2 sur la grille fine, remaillage avec Λ_3.

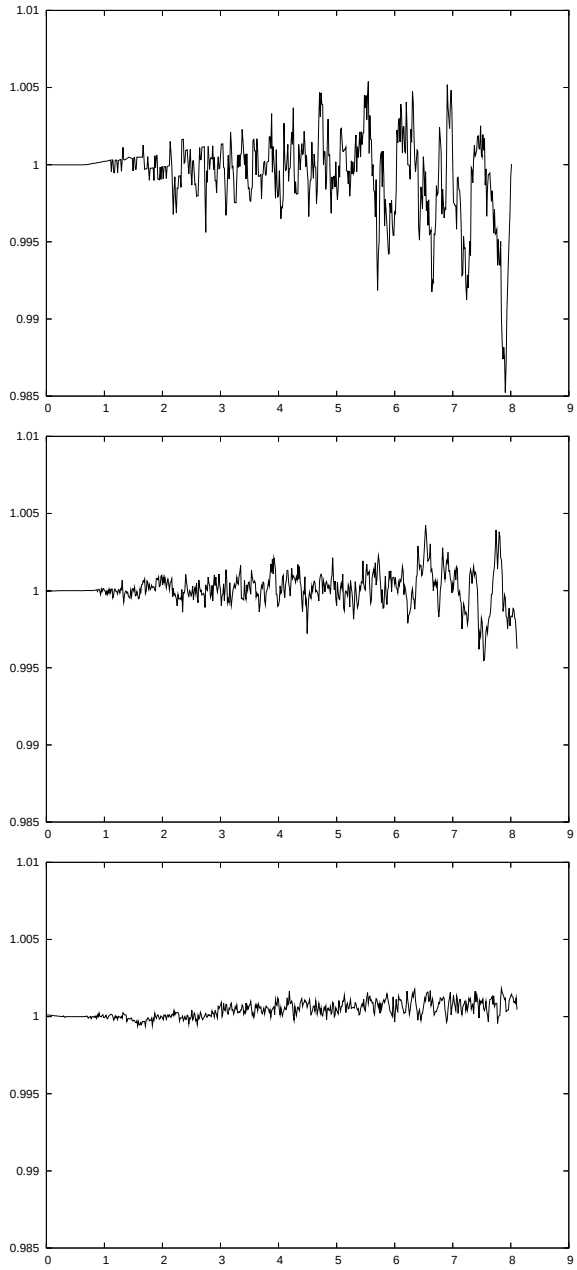


FIG. 5.11 – Evolution de la masse du fluide situé initialement à droite, de haut en bas, grille grossière de taille 50^2, puis avec ϕ résolu sur une grille 100^2 et 200^2.

jusqu'au temps $t = 0.2$, qui correspond à l'impact du choc sur la bulle. Ensuite ce rapport augmente d'autant plus vite que la grille utilisée pour les calculs est grossière. Dans le cas de l'interaction choc-bulle, l'utilisation d'une grille et de particules plus fines ne fait pas vraiment apparaître d'échelles plus petites. La conservation des masses partielles peut donc bénéficier de la meilleure résolution de l'interface offerte par la fonction level-set résolue plus finement.

5.3.3.2 Utilisation d'une grille plus fine autour de la bulle, portant toutes les variables

Nous souhaitons maintenant évaluer si l'utilisation d'une grille plus fine située autour de l'interface, et qui la suit au cours de son déplacement, permet d'obtenir des résultats aussi précis qu'avec une grille plus fine couvrant tout le domaine. Nous réalisons les calculs avec une grille grossière de taille 50×500 et une grille plus fine autour de la bulle, de discrétisation $dx = 1/100$, puis une grille grossière de taille 100×1000, et une grille plus fine de discrétisation $dx = 1/200$.

Les figures 5.17 et 5.18 présentent les isovaleurs de la fonction ϕ. Ces résultats sont très semblables à ceux obtenus dans la section précédente, avec les calculs réalisés sur une seule grille.

L'évolution de la conservation de la masse est présentée sur la figure 5.19. Elle aussi est très semblable aux résultats équivalents présentés dans la section précédente. La conservation de la masse est donc améliorée dans les mêmes proportions quand une grille plus fine est placée autour de l'interface ou quand tout le domaine de calcul est résolu avec cette grille fine.

5.3.3.3 Utilisation d'une grille plus fine autour de la bulle pour la résolution de l'interface

Ici, nous cherchons à évaluer si, comme dans le cas de l'instabilité de Kelvin-Helmholtz, l'utilisation d'une grille plus fine uniquement pour la résolution de l'interface améliore la conservation des masses partielles. Nous réalisons les calculs avec une grille grossière de taille 50×500 qui couvre tout le domaine, et une grille plus fine autour de la bulle, de discrétisation $dx = 1/100$, puis une grille grossière de taille 100×1000, et une grille plus fine de discrétisation $dx = 1/200$.

La résolution de l'interface se trouve améliorée par cette utilisation d'une grille plus fine : comme dans le cas de l'instabilité de Kelvin-Helmholtz, la distance entre les isolignes $\phi = 0.5$ et $\phi = -0.5$ est diminuée. Toutefois, par rapport aux résultats présentés dans les sections précédentes, l'utilisation d'une grille plus fine juste pour l'interface ne permet pas de faire disparaître les instabilités numériques sur la tige, issues de la résolution de l'hydrodynamique sur une grille plus grossière. Elle semble même les amplifier, car elle leur permet d'être résolues plus précisément.

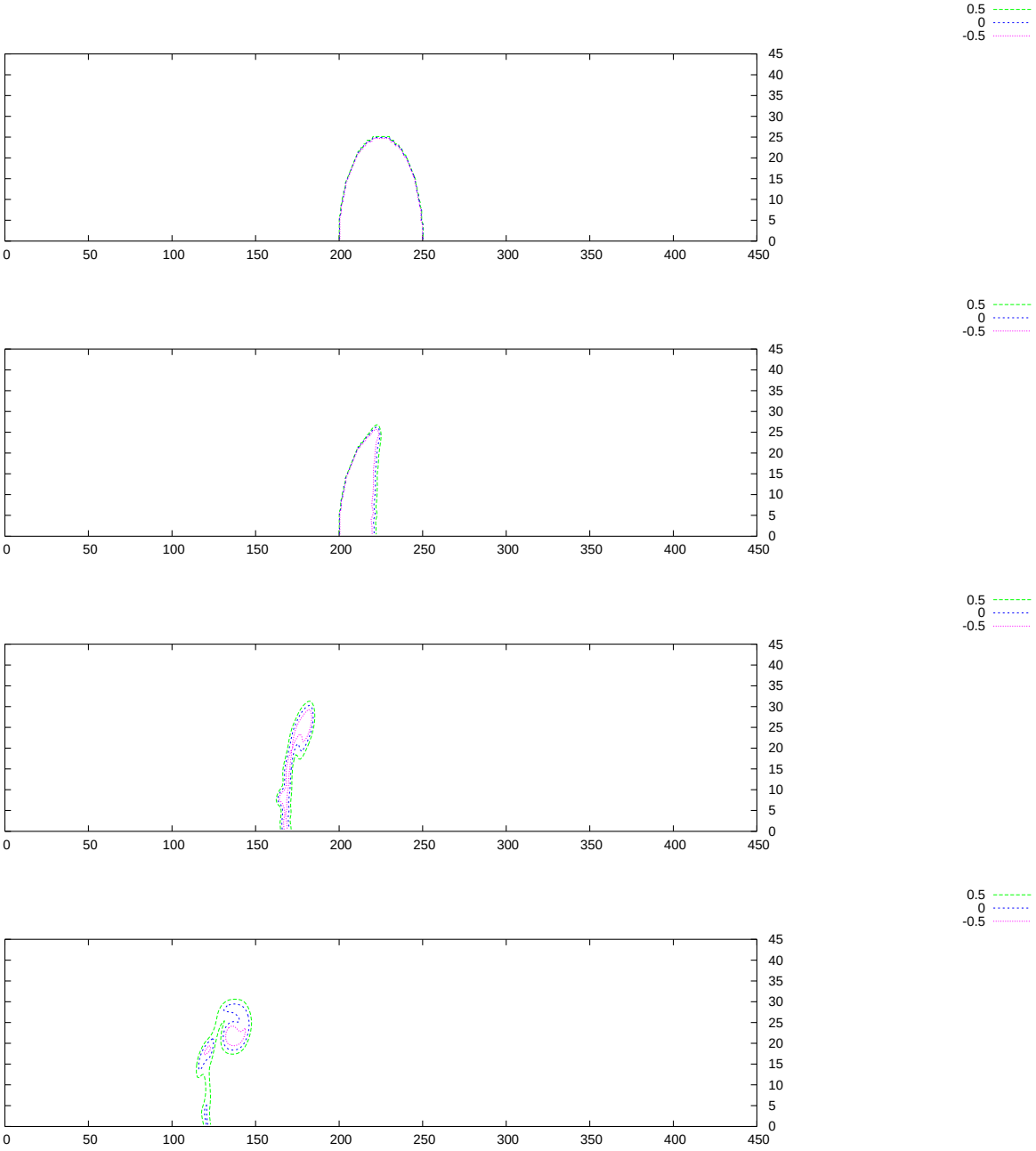


FIG. 5.12 – Interaction choc-bulle, isovaleurs de ϕ, à $t = 0., 0.5, 1., 1.5$, 50×500 particules, remaillage avec Λ_3.

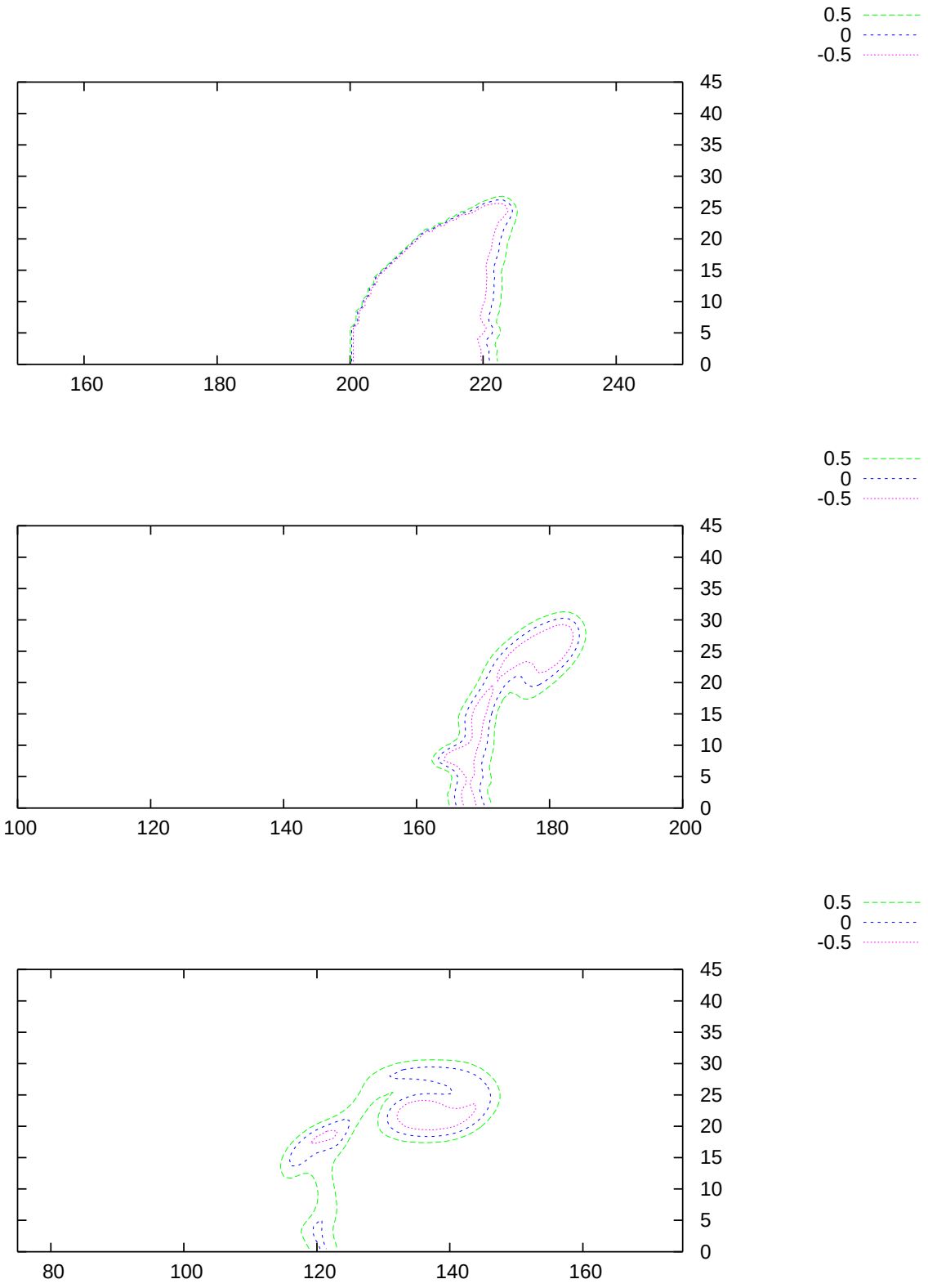


FIG. 5.13 – Interaction choc-bulle, isovaleurs de ϕ, à $t = 0.5, 1., 1.5$, 50×500 particules, remaillage avec Λ_3.

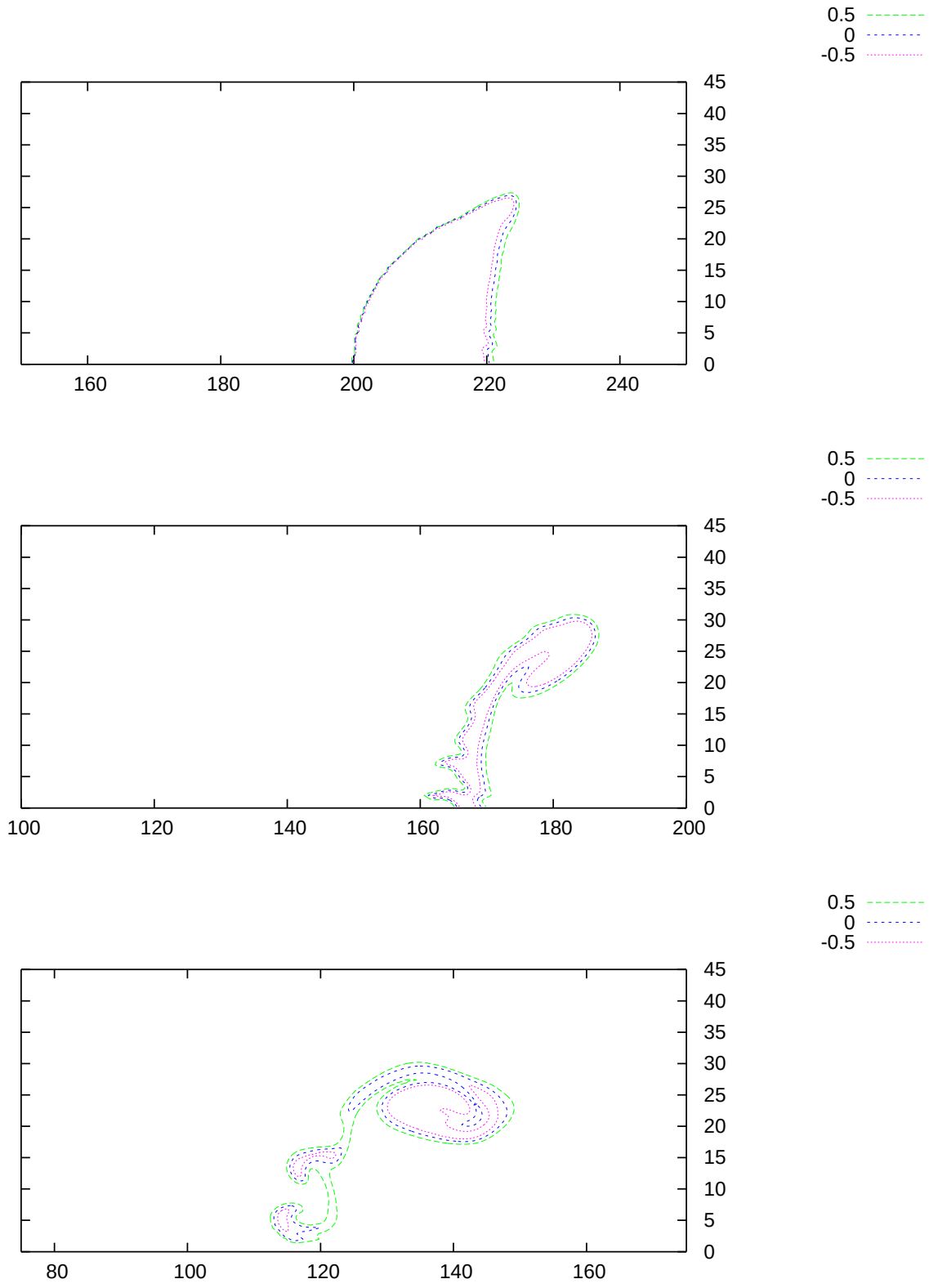


FIG. 5.14 – Même légende que la figure précédente, avec 100×1000 particules.

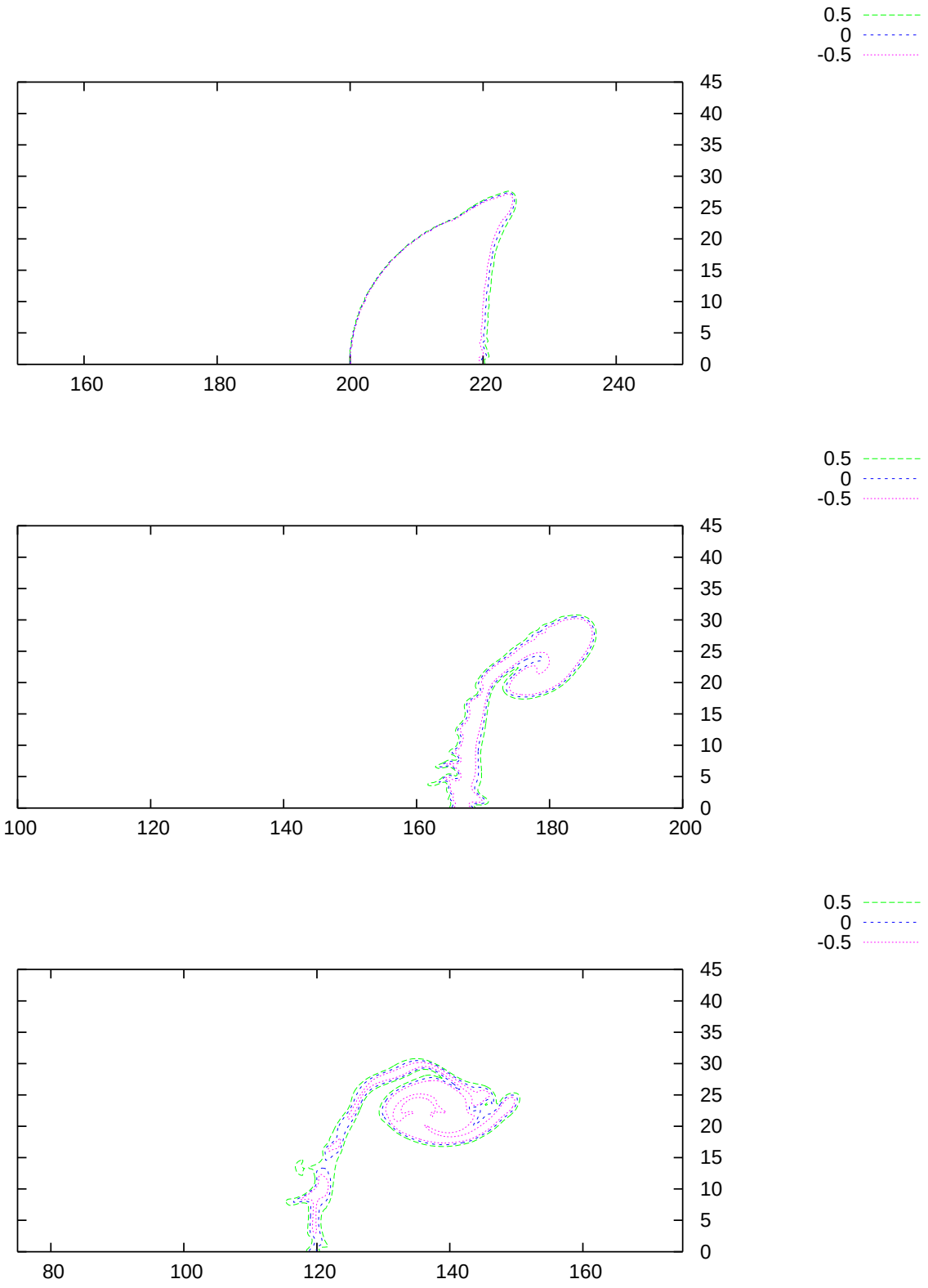


FIG. 5.15 – Même légende que la figure précédente, avec 200×2000 particules.

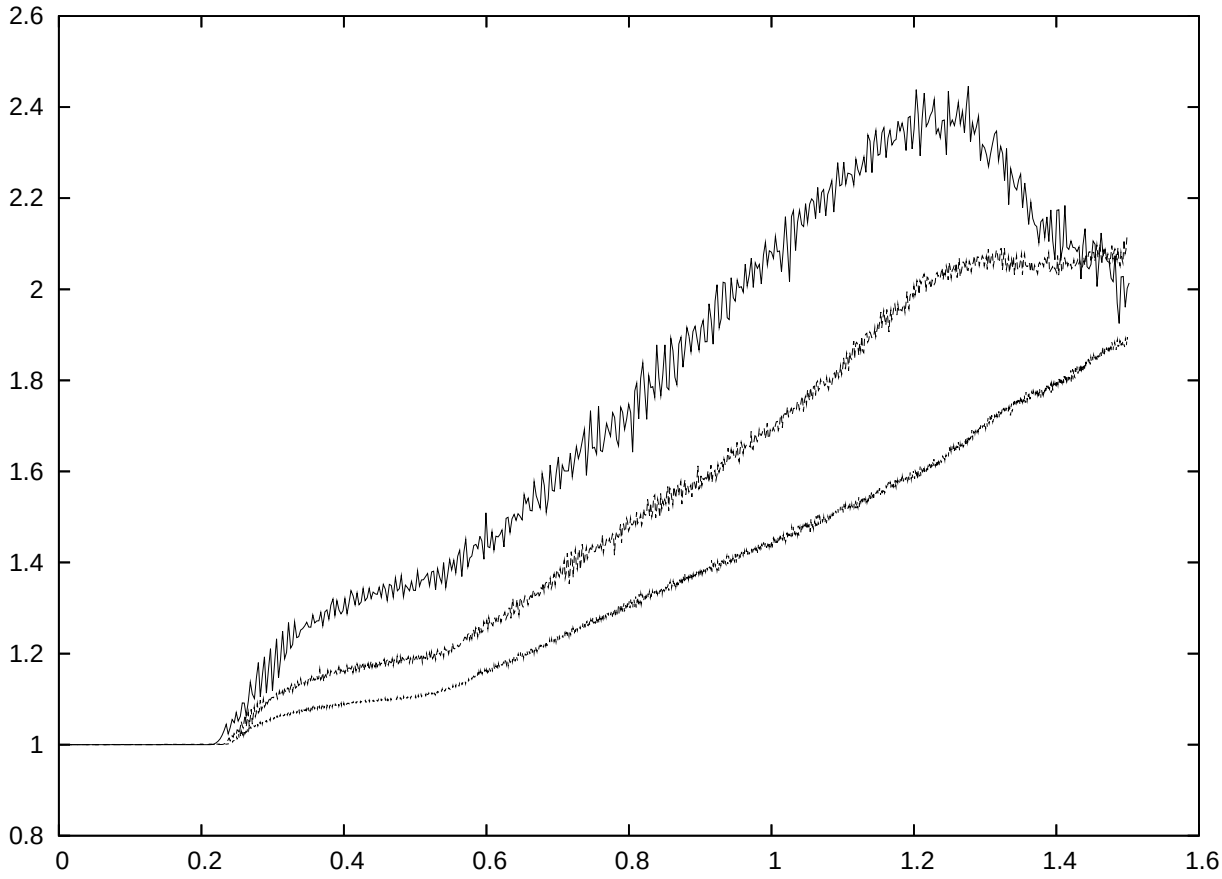


FIG. 5.16 – Interaction choc-bulle, évolution du rapport masse/masse initiale de la bulle, pour (de haut en bas) 50×500, 100×1000 et 200×2000 particules, remaillage avec Λ_3.

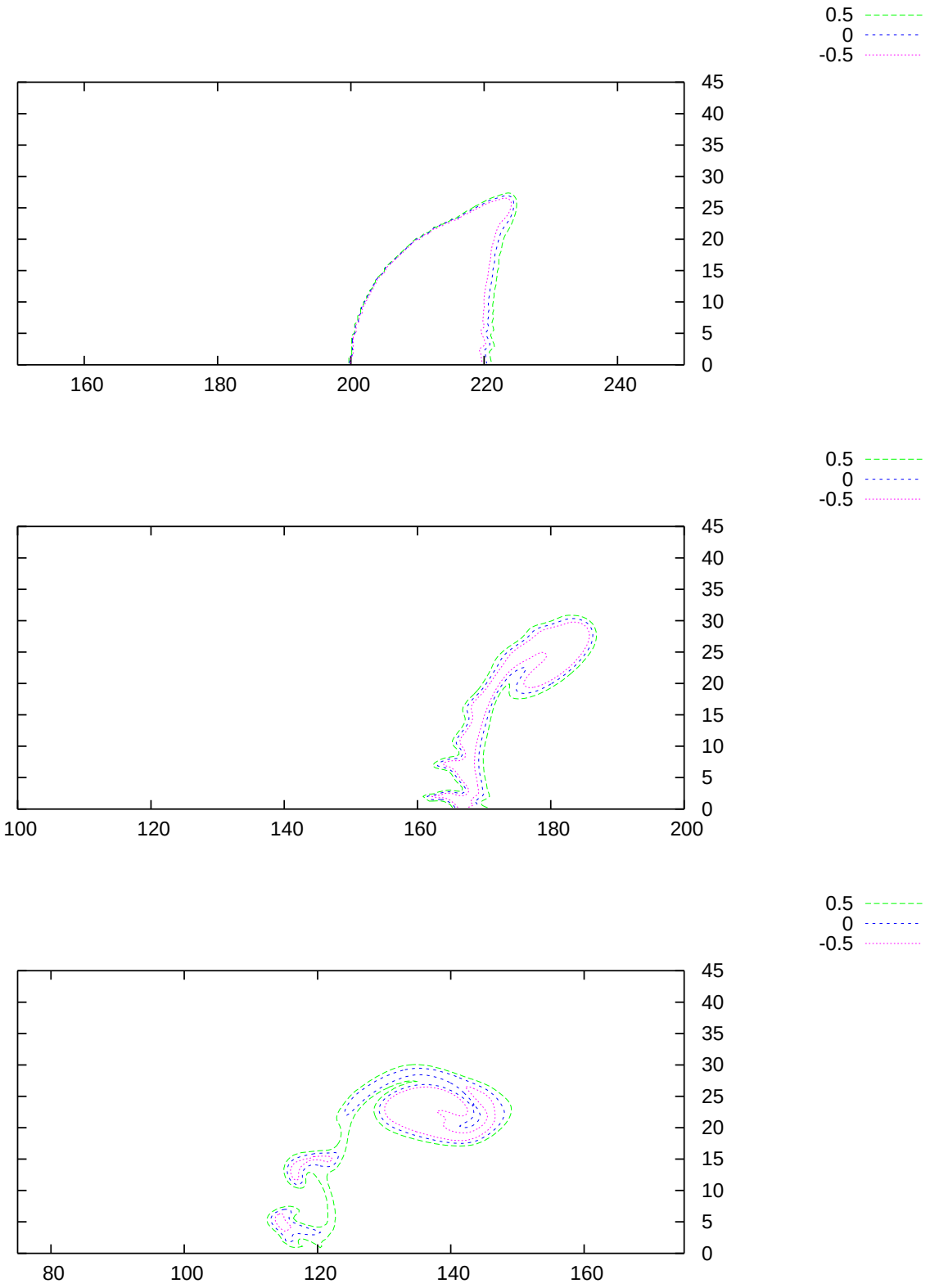


FIG. 5.17 – Interaction choc-bulle, isovaleurs de ϕ, à $t = 0.5, 1., 1.5$, utilisation de deux tailles de particules portant toutes deux les variables des équations d'Euler : $dx = 1/100$ autour de la bulle et $dx = 1/50$ ailleurs, remaillage avec Λ_3.

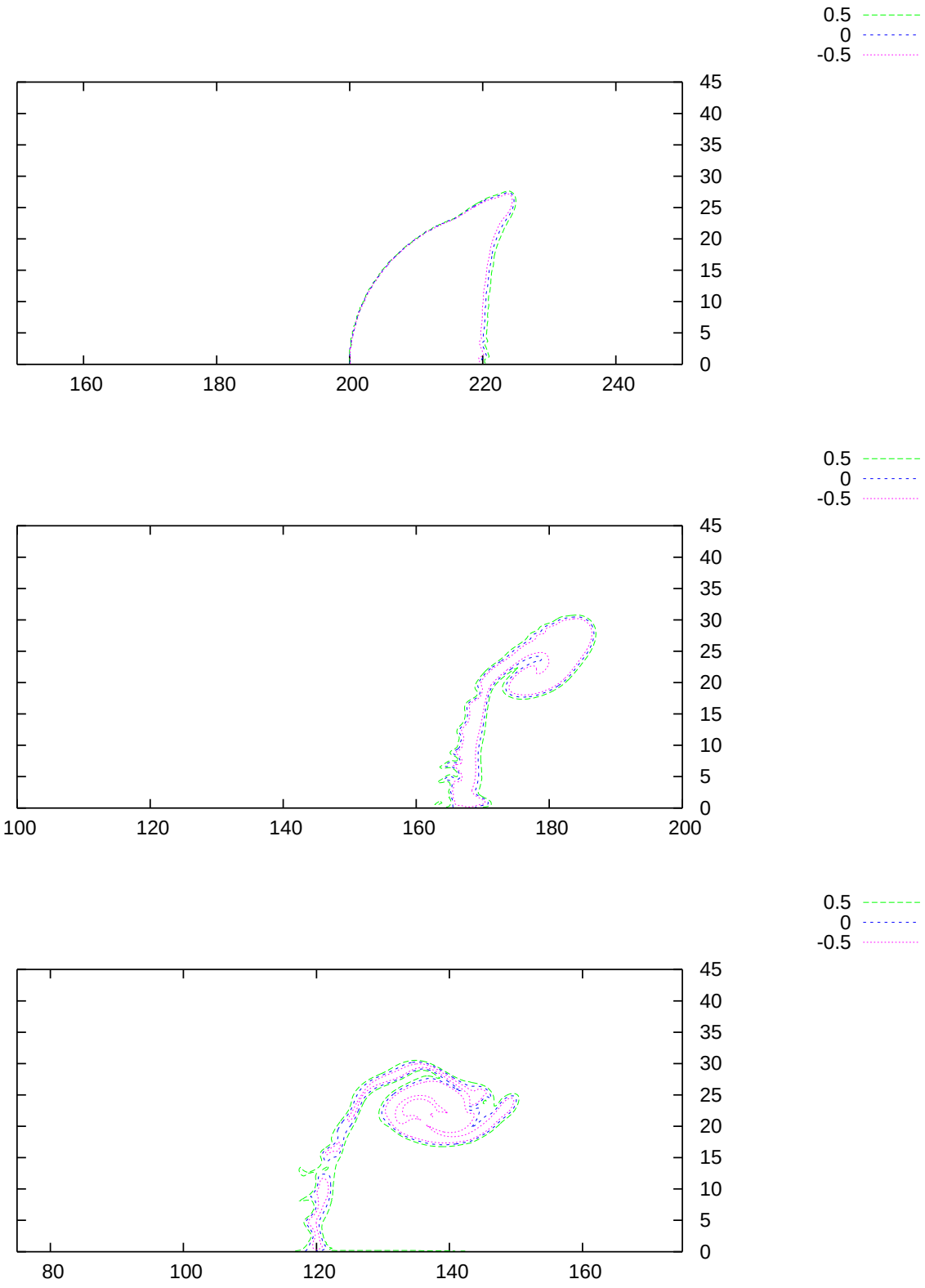


FIG. 5.18 – Même légende que la figure précédente, avec $dx = 1/200$ autour de la bulle et $dx = 1/100$ ailleurs

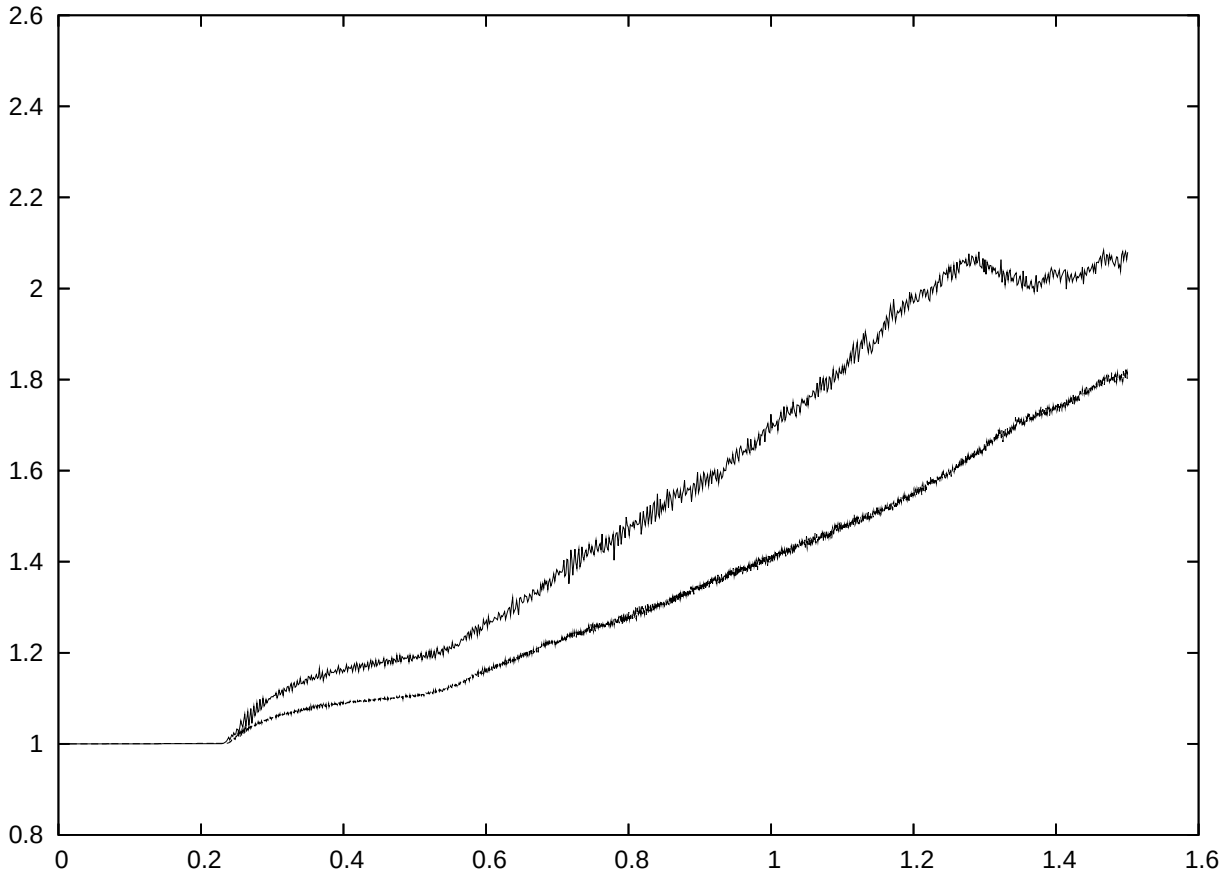


FIG. 5.19 – Interaction choc-bulle, évolution de la masse de la bulle, utilisation de deux tailles de particules portant toutes deux les variables des équations d'Euler : $dx = 1/50$ et $dx = 1/100$ autour de la bulle, remaillage avec Λ_3.

La figure 5.22 représente l'évolution du rapport masse/masse initiale pour les deux configurations étudiées dans cette section : $dx_1 = 1/50$, $dx_2 = 1/100$, et $dx_1 = 1/100$, $dx_2 = 1/200$. On constate que les deux rapports évoluent avec des valeurs très proches. Dans les deux cas, la conservation des masses partielles est meilleure que toutes celles évaluées dans les sections précédentes. Cela confirme à nouveau l'intérêt de l'utilisation d'une grille plus fine uniquement pour résoudre l'interface.

5.4 Conclusion du chapitre 5

Dans ce chapitre nous nous sommes intéressés à la simulation d'écoulements bi-fluides. Nous avons choisi d'adapter une méthode de type level-set à notre méthode particulaire. La fonction level-set ϕ, qui contient l'information sur l'interface entre les fluides, est discrétisée sur les particules, advectée et interpolée (multipliée par le volume de la particule) comme les autres variables. Les différents tests étudiés : tube à chocs de Sod bifluide, instabilité de Kelvin-Helmholtz, et l'interaction choc-bulle, nous montrent que l'information sur l'interface, portée par ϕ, est transportée avec précision par les particules et préservée par l'interpolation. Notre méthode particulaire résoud sans aucun problème l'instabilité de Kelvin-Helmholtz, alors que les méthodes de type SPH en sont incapables.

Nous avons adapté la technique multi-niveaux présentée dans le chapitre 4 afin que l'interface soit résolue avec une meilleure précision. Les particules associées à la grille plus fine qui suit l'interface servent soit uniquement à résoudre l'interface et à calculer les masses partielles, soit également à résoudre les équations d'Euler. L'utilisation d'une grille plus fine autour de l'interface pour résoudre toutes les variables permet d'obtenir les mêmes résultats que si cette grille fine couvrait tout le domaine, ce qui permet de diminuer le coût des calculs. L'utilisation d'une grille plus fine pour résoudre uniquement l'interface améliore nettement la conservation des masses partielles. Cette amélioration n'ajoute qu'un faible coût de calcul par rapport à la résolution uniquement sur la grille grossière, car les particules qui portent ϕ (ainsi que ρ, pour évaluer la masse) sont soumises à la seule condition de stabilité portant sur le non-croisement des particules.

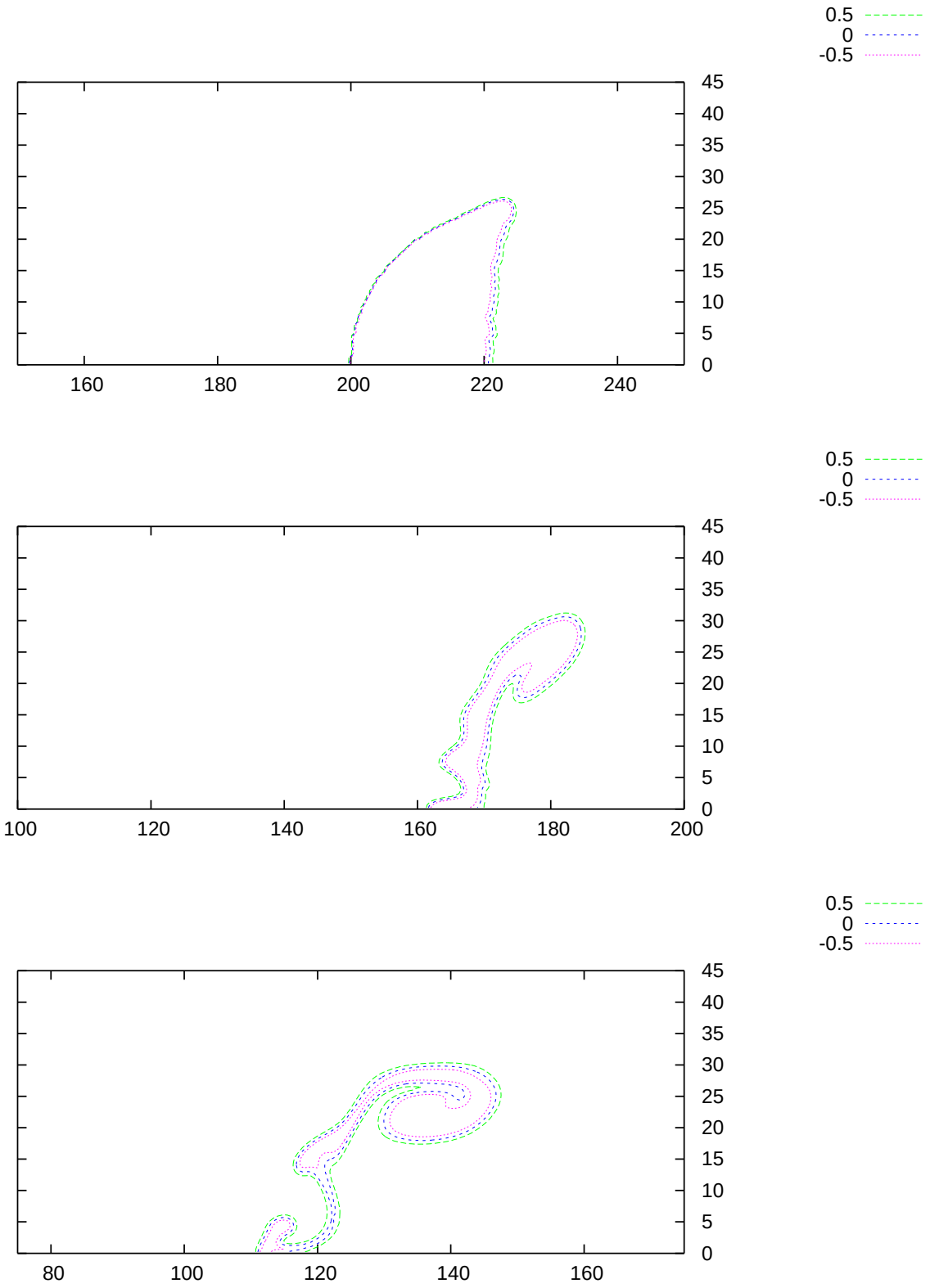


FIG. 5.20 – Interaction choc-bulle, isovaleurs de ϕ, à $t = 0.5, 1., 1.5$, 50×500 particules (discrétisation N= 50) pour résoudre l'hydrodynamique, et des particules plus fines (N= 100) pour résoudre l'interface, remaillage avec Λ_3.

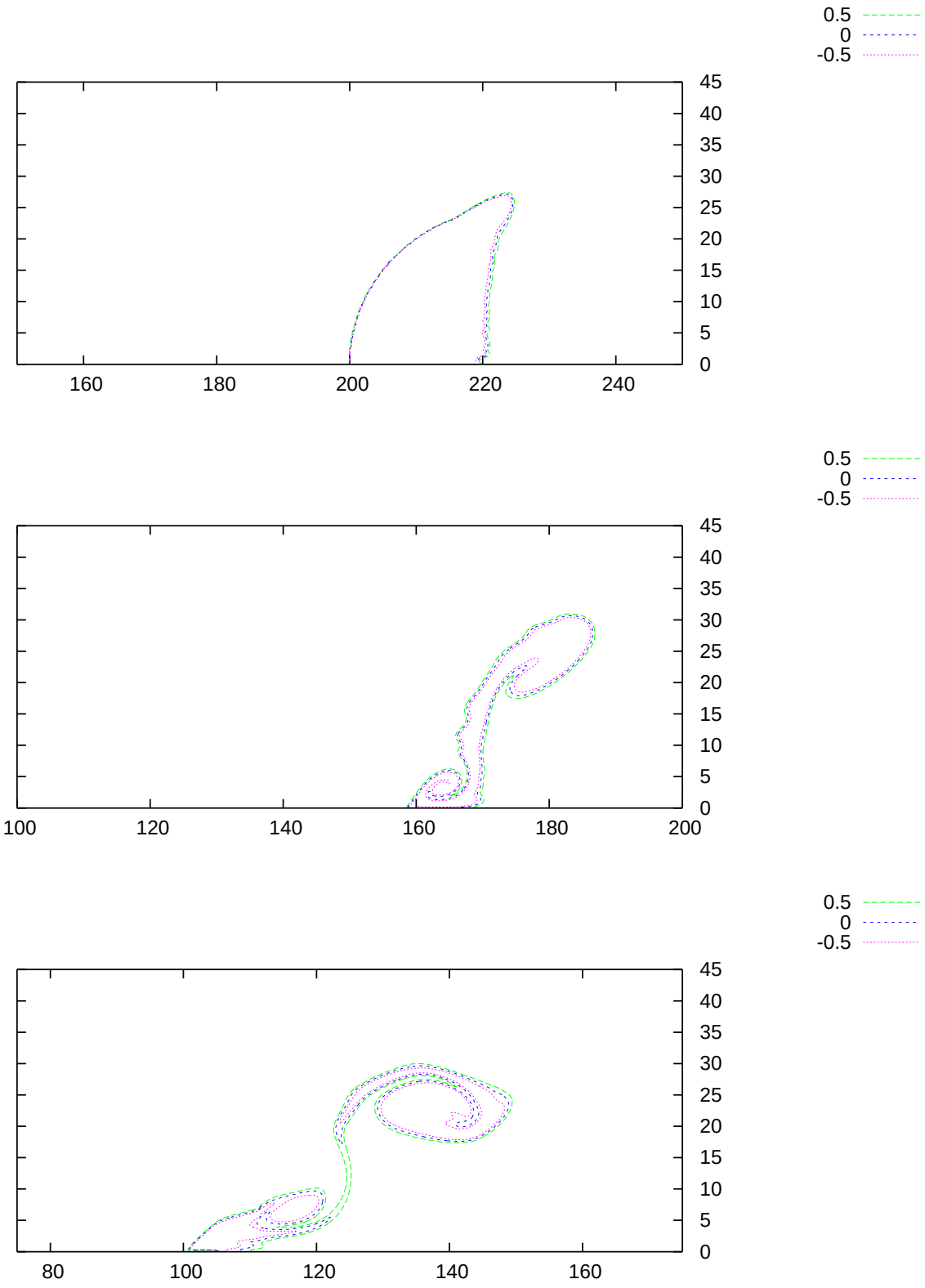


FIG. 5.21 – Même légende que la figure précédente, avec des grilles 100×1000 et 200×2000.

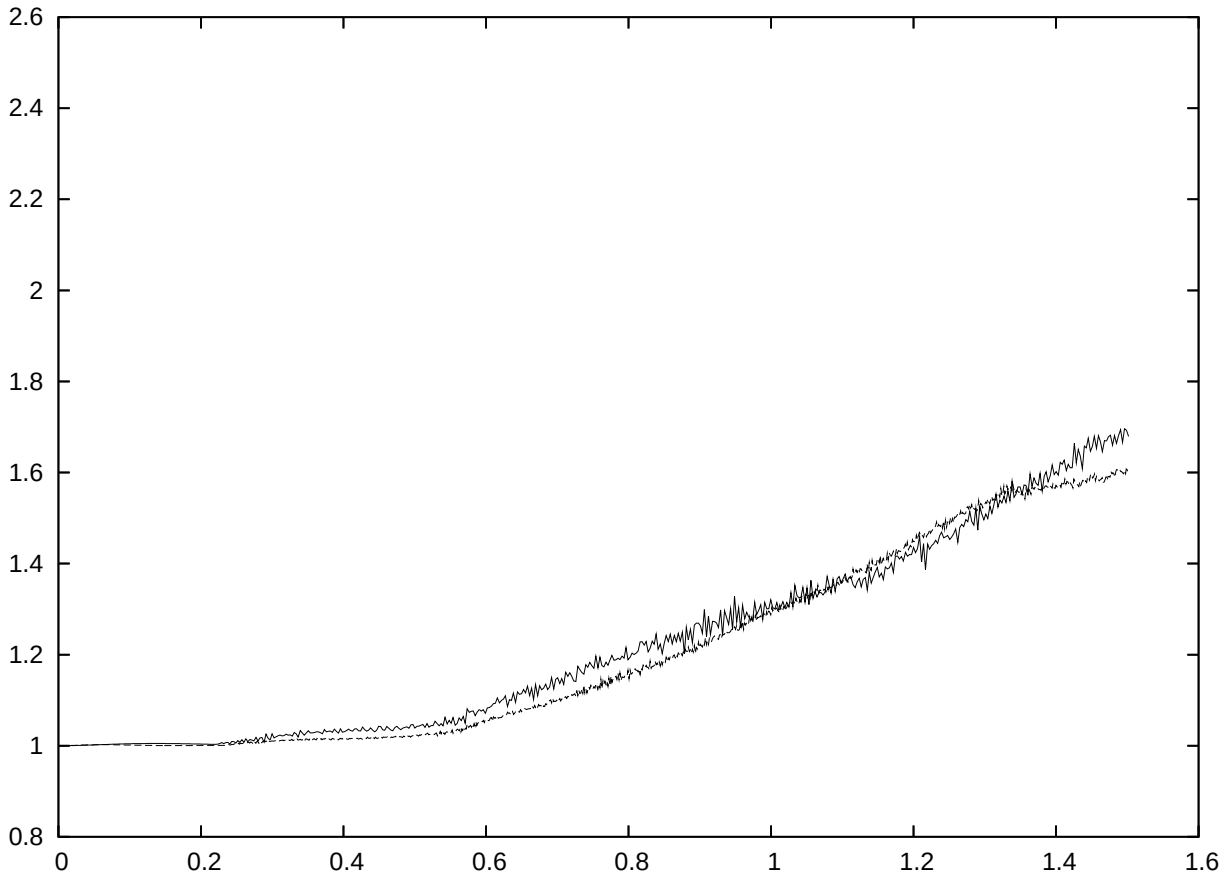


FIG. 5.22 – Interaction choc-bulle, évolution de la masse de la bulle, avec 50×500 particules pour l'hydrodynamique, et des particules plus fines ($dx = 1/100$) pour l'interface (ligne plus oscillante, située en haut au début), en bas avec 100×1000 particules pour l'hydrodynamique, et une discrétisation $dx = 1/100$ l'interface (ligne moins oscillante, située en bas au début), et remaillage avec Λ_3.

Conclusion

Dans ce travail nous avons développé et analysé une méthode particulaire avec remaillage, appliquée à la simulation de la dynamique des gaz, en particulier dans le cas d'écoulements bi-fluides.

Dans le chapitre 1 nous avons présenté le principe de la discrétisation particulaire appliqué aux équations d'Euler compressibles. Le fluide est découpé en particules, possédant une masse, une vitesse, une énergie, qui se déplacent de manière lagrangienne, à la vitesse de l'écoulement. Nous résolvons les équations d'Euler en pratiquant une décomposition "Lagrange + advection", c'est-à-dire un splitting entre la prise en compte des efforts de pression et celle du déplacement de la matière. Nous maintenons une distribution régulière de particules en interpolant de manière conservative les particules sur les centres des cellules d'une grille sous-jacente, afin d'y créer de nouvelles particules. Cette méthode permet de bénéficier de la précision du transport lagrangien des variables conservatives des équations d'Euler et d'un remaillage conservatif effectué avec des formules d'interpolation d'ordre élevé. Elle évite de se préoccuper, comme c'est le cas avec les méthodes de volumes finis, de l'évolution des volumes de discrétisation (problème de croisement des mailles), puisqu'il suffit d'éviter que les particules ne se croisent.

Dans le chapitre 2 nous avons tout d'abord analysé la stabilité de la méthode particulaire. La première condition de stabilité concerne le traitement des efforts de pression et est une condition CFL classique. La deuxième concerne le déplacement des particules et limite le pas de temps indépendamment du pas d'espace, en fonction du gradient de la vitesse. Ensuite, pour mieux comprendre les effets du déplacement des particules combiné au remaillage (qui constituent la particularité et le point essentiel de notre méthode), nous l'avons analysé sur une équation d'advection modèle, monodimensionnelle et scalaire. Notre étude nous a permis de ré-écrire les schémas particulaires sous forme de schémas aux différences finies et d'analyser leur précision en fonction du noyau d'interpolation utilisé, révélant des liens étroits avec des méthodes de différences finies d'ordre élevé, de type Lax-Wendroff. Nous avons proposé une nouvelle évaluation de la vitesse des particules, simple à implanter, qui permet d'obtenir dans le cas non-liméaire des schémas d'ordre 2. Nous avons aussi montré, dans le cas d'un noyau d'interpolation particulier et pour l'équation de Bürgers, que l'ajout d'un certain terme non-linéaire de viscosité artificielle stabilisait la méthode.

Il serait intéressant d'améliorer encore l'évaluation de la vitesse des particules, afin d'obtenir des schémas particulaires d'ordre supérieur à 2. Il faudrait pour cela trouver un moyen d'évaluer avec davantage de précision la valeur moyenne de la vitesse sur le volume de la particule pendant un pas de temps. Nous aimerions également étendre l'étude du terme de viscosité artificielle à d'autres noyaux d'interpolation, et l'adapter aux équations d'Euler, comme cela a été proposé par Majda et Osher pour le schéma de Lax-Wendroff dans [58].

Dans le chapitre 3, nous avons présenté l'algorithme avec lequel sont résolues les équations d'Euler. Nous avons vérifié sur une équation d'advection uniforme que l'ordre de convergence numérique était le même que celui prédit théoriquement dans le chapitre 2. Nous avons également mis en évidence une propriété intéressante des schémas particulaires : ils semblent converger vers la solution entropique, bien que nous ne sachions pas pour le moment expliquer pourquoi. Trouver une explication à cette observation est une piste de travail que nous aimerions explorer par la suite.

Nous avons testé la méthode sur trois problèmes typiques de la dynamique des gaz : le tube à chocs de Sod, la détente de Sjögreen et le problème de Noh. Les résultats sont comparables à ceux d'autres méthodes présentes dans la littérature. Dans l'état actuel de la méthode, nous recommandons d'utiliser le noyau Λ_3 pour le remaillage, ainsi que des différences finies centrées plutôt que le solveur de Riemann approché. Toutefois, la résolution de la méthode serait sans doute améliorée si on utilisait une technique plus performante que celles actuellement utilisées pour résoudre la phase Lagrange : par exemple, en appliquant une technique de type MUSCL associée à des limiteurs.

Pour augmenter l'adaptivité de notre méthode particulaire, nous avons implanté dans le chapitre 4 une technique multi-niveaux adaptée aux méthodes particulaires avec remaillage. Nous avons rappellé quelles étaient les conditions de recouvrement des grilles nécessaires à un traitement consistant des conditions aux limites de chaque grille. et proposé des critères d'adaptation des grilles basés sur les dérivées partielles spatiales. Nous intéressant à la conservativité de cette technique multi-niveaux, nous avons proposé une définition de la masse totale sur le domaine de calcul et montré que cette quantité était totalement conservée, sauf dans une zone autour de la frontière entre les grilles, où elle est conservée à l'ordre 2. Nos tests numériques nous ont montré que le critère d'adaptation des grilles permettait effectivement de suivre les zones de fort gradient, et que la masse totale était correctement préservée.

Dans le chapitre 5 nous nous sommes intéressés à la simulation d'écoulements bifluides. Nous avons choisi d'adapter une méthode de type level-set à notre méthode particulaire. La fonction level-set ϕ, qui contient l'information sur l'interface entre les fluides, est discrétisée sur les particules, advectée et interpolée comme les autres variables. Les différents tests étudiés : tube à chocs de Sod bifluide, instabilité de Kelvin-Helmholtz, et l'interaction choc-bulle, nous ont montré que l'information sur l'interface,

portée par ϕ, était correctement transportée par les particules et préservée par l'interpolation. Nous avons appliqué la technique multi-niveaux du chapitre 4 afin de résoudre l'interface avec une meilleure précision. Les particules associées à la grille plus fine qui suit l'interface servent, soit uniquement à résoudre l'interface et à calculer les masses partielles, soit également à résoudre les équations d'Euler. L'utilisation d'une grille plus fine autour de l'interface pour résoudre toutes les variables permet d'obtenir les mêmes résultats que si cette grille fine couvrait tout le domaine. L'utilisation d'une grille plus fine à seul usage de résoudre l'interface améliore la conservation des masses partielles, pour un faible coût supplémentaire.

Pour poursuivre ce travail, il serait intéressant d'appliquer cette méthode à des problèmes plus réalistes issus de l'étude de la Fusion par Confinement Inertiel, qui est un axe de recherche très important du CEA. Il faudrait pour cela rajouter aux équations d'Euler des termes pour tenir compte du transfert radiatif. Il s'agirait d'étudier par exemple la formation d'instabilités à l'interface entre deux coquilles presque sphériques qui subiraient une très forte compression. Un autre champ d'application concerne la simulation d'écoulements bifluides, non plus constitués de deux gaz, mais d'un gaz et d'un liquide, comme par exemple la simulation de liquides contenant des bulles.

Annexe A

Détails des calculs de la section 2.2.6

On présente ici le détail des calculs de la section 2.2.6 du chapitre 1, qui concernent un terme correctif non-linéaire pour le schéma particulaire avec Λ_2 pour l'équation de Bürgers.

On utilise les notations suivantes :

$$\begin{aligned}
\Delta_+ u_j &= u_{j+1} - u_j \\
\Delta_- u_j &= u_j - u_{j-1} = \Delta_+ u_{j-1} \\
\Delta_0 u_j &= u_{j+1} - u_{j-1} \\
\epsilon_0 &= \lambda \max_j |u_j|
\end{aligned}$$

Majda et Osher [57] ont montré que pour l'équation de Bürgers il existe un C tel que le terme de correction non-linéaire $C\Delta_-(|\Delta_+ u_j|\Delta_+ u_j)$ assure la décroissance de l'énergie $\sum u_j^{n+1} \leq \sum u_j^n$ pour le schéma de Lax-Wendroff, sous la condition $\epsilon_0 \leq 0.24$. On cherche à montrer que ce terme de viscosité artificielle stabilise aussi le schéma corrigé avec Λ_2.

Avant de détailler les calculs, quelques propriétés :

$$\begin{aligned}
\Delta_-\Delta_+ u_j &= \Delta_+ u_j - \Delta_- u_j \\
\Delta_0 u_j &= \Delta_+ u_j + \Delta_- u_j \\
\Delta_+ {u_j}^3 &= \Delta_+ u_j (u_{j+1}^2 + u_{j+1}u_j + {u_j}^2) = \Delta_+ u_j b_j \\
\Delta_0\Delta_+ {u_j}^2 &= \Delta_+ u_{j+1}(u_{j+2} + u_{j+1}) - \Delta_+ u_{j-1}(u_j + u_{j-1}) \\
&\quad -\Delta_+ u_{j-1}(u_{j+2} + u_{j+1}) + \Delta_+ u_{j-1}(u_{j+2} + u_{j+1}) \\
&= \Delta_0\Delta_+ u_j (u_{j+2} + u_{j+1}) + \Delta_+ u_{j-1}(\Delta_0 u_{j+1} + \Delta_0 u_j) \\
\Delta_-\Delta_+ {u_j}^3 &= \Delta_-(\Delta_+ u_j b_j) = \Delta_+ u_j b_j - \Delta_+ u_{j-1} b_{j-1} - \Delta_+ u_{j-1} b_j + \Delta_+ u_{j-1} b_j \\
&= b_j \Delta_-\Delta_+ u_j + \Delta_- b_j \Delta_+ u_{j-1}
\end{aligned}$$

$$\begin{aligned}\Delta_0({u_j}^2\Delta_0 u_j) &= {u_{j+1}}^2\Delta_0 u_{j+1} - {u_{j-1}}^2\Delta_0 u_{j-1} - {u_j}^2\Delta_0 u_{j+1} + {u_j}^2\Delta_0 u_{j+1} + {u_j}^2\Delta_0 u_{j-1} \\ &\quad -{u_j}^2\Delta_0 u_{j-1} \\ &= \Delta_+{u_j}^2\Delta_0 u_{j+1} + \Delta_+{u_{j-1}}^2\Delta_0 u_{j-1} + {u_j}^2\Delta_0\Delta_0 u_j\end{aligned}$$

$$\begin{aligned}\Delta_+({u_j}^3\Delta_0 u_j) &= {u_{j+1}}^3\Delta_0 u_{j+1} - {u_j}^3\Delta_0 u_j - {u_j}^3\Delta_0 u_{j+1} + {u_j}^3\Delta_0 u_{j+1} \\ &= \Delta_+{u_j}^3\Delta_0 u_{j+1} + {u_j}^3\Delta_+\Delta_0 u_j\end{aligned}$$

$$\sum |\Delta_+ u_j|^2|\Delta_0 u_j| \le \frac{2}{3}\sum |\Delta_+ u_j|^3 + \frac{1}{3}\sum |\Delta_- u_j|^3 = \sum |\Delta_+ u_j|^3$$

Sur le même principe, on a également :

$$\sum |\Delta_+ u_j|^2|\Delta_0 u_j| \le 2\sum |\Delta_+ u_j|^3$$

$$\sum |\Delta_+ u_j||\Delta_- u_j||\Delta_0 u_j| \le 2\sum |\Delta_+ u_j|^3$$

$$\sum \Delta_+(|\Delta_+ u_j||\Delta_- u_j|^2)| \le 2\sum |\Delta_+ u_j|^3$$

etc...

Le schéma réécrit avec les notations définies en début de section devient :

$$\begin{aligned}u_j^{n+1} &= u_j^n \underbrace{-\frac{\lambda}{4}\Delta_0 u_j^2}_{F_j} + \underbrace{\frac{\lambda^2}{32}\Delta_0(u_j^2\Delta_0) + \frac{\lambda^2}{8}\Delta_-\Delta_+ u_j^3}_{G_j} \\ &\quad \underbrace{-\frac{\lambda^3}{64}\Delta_-\Delta_+(u_j^3\Delta_0 u_j)}_{K_j} + \underbrace{C\lambda\Delta_-(|\Delta_+ u_j|\Delta_+ u_j)}_{J_j}\end{aligned}$$

$$\begin{aligned}\sum_j {u_j^{n+1}}^2 - \sum_j {u_j^n}^2 &= 2\sum u_j J_j + 2\sum u_j F_j + 2\sum J_j F_j + \sum {F_j}^2 \\ &\quad +2\sum u_j G_j + \sum {J_j}^2 + 2\sum J_j G_j + 2\sum u_j K_j + 2\sum F_j G_j \\ &\quad + \sum {G_j}^2 + 2\sum F_j K_j + 2\sum J_j K_j + 2\sum G_j K_j + \sum {K_j}^2\end{aligned}$$

Le but est de montrer que sous une certaine condition CFL, il existe un C tel que $\sum_j {u_j^{n+1}}^2 - \sum_j {u_j^n}^2 \le 0$. Pour arriver à ce résultat, on essaie de majorer chaque terme non-négatif par une expression où apparaît $\sum |\Delta_+ u_j|^3$, qui sera compensée par le terme $2\sum u_j J_j = -2\lambda C\sum |\Delta_+ u_j|^3$. Les termes pour lesquels cette majoration ne semble pas possible sont regroupés en une somme, que l'on prouve être négative sous une autre condition CFL.

A.1 termes en λ

$$\begin{aligned} 2\sum u_j J_j &= 2\lambda C \sum u_j \Delta_-(|\Delta_+ u_j|\Delta_+ u_j) \\ &= -2\lambda C \sum |\Delta_+ u_j|^3 \end{aligned} \tag{A.1}$$

$$2\sum u_j F_j = -\frac{\lambda}{2}\sum u_j \Delta_0 u_j^2$$

Or : $\sum u_j \Delta_o u_j^2 = \frac{1}{3}\sum(u_j^3 + 3u_j u_{j+1} - 3u_{j+1}u_j^2 - u_{j+1}^3) = -\frac{1}{3}\sum(\Delta_+ u_j)^3$

$$\Rightarrow 2\sum u_j F_j = \frac{\lambda}{6}(\Delta_+ u_j)^3 \tag{A.2}$$

A.2 termes en λ^2

$$\begin{aligned} \sum J_j^2 &= C^2\lambda^2 \sum(\Delta_-(\Delta_+ u_j|\Delta_+ u_j))^2 \\ &= 2C^2\lambda^2 \sum |\Delta_+ u_j|^4 - |\Delta_+ u_j|(\Delta_+ u_j)|\Delta_+ u_{j-1}|(\Delta_+ u_{j-1}) \\ &\leq 8C^2\lambda\epsilon_0 \sum |\Delta_+ u_j|^3 \end{aligned} \tag{A.3}$$

$$\begin{aligned} 2\sum F_j J_j &= -\frac{\lambda^2 C}{2}\sum \Delta_-(|\Delta_+ u_j|\Delta_+ u_j)\Delta_0 u_j^2 \\ &\leq 4\epsilon_0 C\lambda \sum |\Delta_+ u_j|^3 \end{aligned} \tag{A.4}$$

$$\sum F_j^2 + 2\sum u_j G_j = -\frac{\lambda^2}{16}\sum(\Delta_0 u_j)^2 u_j^2 - \frac{\lambda^2}{4}\sum \Delta_+ u_j \Delta_+ {u_j}^3 + \frac{\lambda^2}{16}\sum(\Delta_0 u_j^2)^2$$

$\sum(\Delta_0 u_j^2)^2 = \sum(\Delta_+ u_j(u_{j+1}+u_j) + \Delta_+ u_{j-1}(u_j + u_{j-1})^2$
$= 2\sum(\Delta_+ u_j)^2(u_{j+1}+u_j)^2 + \Delta_+ u_j \Delta_+ u_{j-1}(u_{j+1}+u_j)(u_j+u_{j-1})$

$\sum \Delta_+ u_j \Delta_+ u_j^3 = \sum(\Delta_+ u_j)^2(u_{j+1}^2 + u_{j+1}u_j + u_j^2)$

$\sum(\Delta_0 u_j)^2 u_j^2 = 2\sum(\Delta_+ u_j)^2(u_j^2 + u_{j+1}^2) + \Delta_+ u_j \Delta_+ u_{j-1} u_j^2$

Donc :

$$\begin{aligned}\sum F_j^2 + 2\sum u_j G_j &= -\frac{\lambda^2}{8}\sum(\Delta_+ u_j)^2(u_j^2+u_{j+1}^2) - \frac{\lambda^2}{8}\sum \Delta_+ u_j \Delta_+ u_{j-1} u_j^2 \\ &\quad -\frac{\lambda^2}{4}\sum(\Delta_+ u_j)^2(u_{j+1}^2 + u_{j+1}u_j + u_j^2) + \frac{\lambda^2}{8}\sum(\Delta_+ u_j)^2(u_{j+1}+u_j)^2 \\ &\quad +\frac{\lambda^2}{8}\sum \Delta_+ u_j \Delta_+ u_{j-1}(u_{j+1}+u_j)(u_j+u_{j-1})\end{aligned}$$

$$\begin{aligned}\sum F_j^2 + 2\sum u_j G_j &= -\frac{3\lambda^2}{16}\sum(\Delta_+ u_j)^2(u_j^2+u_{j+1}^2) \\ &\quad +\frac{2\lambda^2}{16}\sum \Delta_+ u_j \Delta_+ u_{j-1}(u_{j+1}u_j + u_{j+1}u_{j-1} + u_j u_{j-1}) \\ &\quad +\frac{3\lambda^2}{16}\sum \Delta_+ u_j \Delta_+ u_{j-1}(u_j^2+u_{j+1}^2) - \frac{3\lambda^2}{16}\sum \Delta_+ u_j \Delta_+ u_{j-1}(u_j^2+u_{j+1}^2)\end{aligned}$$

$$\begin{aligned}\sum F_j^2 + 2\sum u_j G_j &= -\frac{3\lambda^2}{16}\sum \Delta_+ u_j \Delta_- \Delta_+ u_j (u_j^2+u_{j+1}^2) \\ &\quad +\frac{\lambda^2}{16}\sum \Delta_+ u_j \Delta_+ u_{j-1}(-3u_j^2 - 3u_{j+1}^2 + 2u_{j+1}u_j + 2u_{j+1}u_{j-1} + 2u_j u_{j-1})\end{aligned}$$

Or :
$-3u_j^2 - 3u_{j+1}^2 + 2u_{j+1}u_j + 2u_{j+1}u_{j-1} + 2u_j u_{j-1} =$
$-2u_j\Delta_+ u_{j-1} - 2u_{j+1}\Delta_0 u_j + (u_j - u_{j+1})\Delta_+ u_j$
Et :
$(\Delta_+ u_j + \Delta_+ u_{j-1})\Delta_-\Delta_+ u_j = \Delta_-((\Delta_+ u_j)^2)$
$(\Delta_+ u_j + \Delta_+ u_{j-1} + \Delta_+ u_j - \Delta_+ u_{j-1})\Delta_-\Delta_+ u_j = \Delta_-((\Delta_+ u_j)^2) + (\Delta_-\Delta_+ u_j)^2$
$\Rightarrow \Delta_+ u_j \Delta_-\Delta_+ u_j = \frac{1}{2}\Delta_-((\Delta_+ u_j)^2) + \frac{1}{2}(\Delta_-\Delta_+ u_j)^2$
Donc :

$$\begin{aligned}\sum \Delta_+ u_j \Delta_-\Delta_+ u_j (u_j^2+u_{j+1}^2) &= \frac{1}{2}\sum \Delta_-(\Delta_+ u_j)^2(u_j^2+u_{j+1}^2) + (\Delta_-\Delta_+ u_j)^2(u_j^2+u_{j+1}^2) \\ &= -\frac{1}{2}\sum(\Delta_+ u_j)^2 \Delta_0 u_{j+1}(u_{j+2}+u_j) + (\Delta_-\Delta_+)^2(u_j^2+u_{j+1}^2)\end{aligned}$$

$$\begin{aligned}\sum F_j^2 + 2\sum u_j G_j &= \frac{3\lambda^2}{32}\sum(\Delta_+ u_j)^2 \Delta_0 u_{j+1}(u_{j+2}+u_j) \\ &\quad +\frac{\lambda^2}{16}\sum \Delta_+ u_j \Delta_+ u_{j-1}[-2u_j\Delta_+ u_{j-1} - 2u_{j+1}\Delta_0 u_j - \Delta_+ u_j^2] \\ &\quad -\frac{3\lambda^2}{32}\sum(\Delta_-\Delta_+ u_j)^2(u_j^2+u_{j+1}^2)\end{aligned}$$

$$\sum F_j^2 + 2\sum u_j G_j \;\leq\; \frac{14}{16}\epsilon_0\lambda\sum|\Delta_+u_j|^3 \underbrace{-\frac{3\lambda^2}{32}\sum(\Delta_-\Delta_+u_j)^2(u_j^2+u_{j+1}^2)}_{(a)} \tag{A.5}$$

A.3 termes en λ^3

$$\begin{aligned} 2\sum J_jG_j &= \frac{C\lambda^3}{16}\sum\Delta_-(|\Delta_+u_j|\Delta_+u_j)\Delta_0(u_j^2\Delta_0u_j) + \frac{C\lambda^3}{4}\sum\Delta_-(|\Delta_+u_j|\Delta_+u_j)\Delta_-\Delta_+u_j^3 \\ &\leq \frac{7}{2}\lambda\epsilon_0^2C\sum|\Delta_+u_j|^3 \end{aligned} \tag{A.6}$$

$$\begin{aligned} 2\sum u_jK_j &= -\frac{\lambda^3}{32}\sum u_j\Delta_-\Delta_+(u_j^3\Delta_0u_j) = -\frac{\lambda^3}{32}\sum\Delta_-\Delta_+u_ju_j^3\Delta_0u_j \\ &= -\frac{\lambda^3}{32}\sum(\Delta_+u_j-\Delta_+u_{j-1})(\Delta_+u_j+\Delta_+u_{j-1})u_j^3 \\ &= -\frac{\lambda^3}{32}\sum u_j^3\Delta_(\Delta_+u_j)^2) = \frac{\lambda^3}{32}\sum\Delta_-u_j^3(\Delta_+u_j)^2 \\ &\leq \frac{3}{32}\lambda\epsilon_0^2\sum|\Delta_+u_j|^3 \end{aligned} \tag{A.7}$$

$$2\sum F_jG_j \;=\; -\frac{\lambda^3}{64}\underbrace{\sum\Delta_0u_j^2\Delta_0(u_j^2\Delta_0u_j)}_{(b)} - \frac{\lambda^3}{64}\underbrace{\sum\Delta_0u_j^2)\Delta_-\Delta_+u_j^3}_{(c)}$$

$\Delta_0({u_j}^2\Delta_0uj) = {u_j}^2\Delta_0\Delta_0u_j + \Delta_+{u_j}^2\Delta_0u_{j+1} + \Delta_+{u_{j-1}}^2\Delta_0u_{j-1}$

$(b) = \underbrace{\sum\Delta_0{u_j}^2\Delta_0\Delta_0u_j}_{(*)} + \underbrace{\Delta_0{u_j}^2[\Delta_+{u_j}^2\Delta_0u_{j+1} + \Delta_+{u_{j-1}}^2\Delta_0u_{j-1}]}_{(**)}$

$(*) = \sum\Delta_0u_j\Delta_0\Delta_0u_j{u_j}^2(u_{j+1}+u_j) = \sum\Delta_-(\Delta_0u_{j+1}\Delta_0u_j){u_j}^2(u_{j+1}+u_{j-1})$
$= -\sum\Delta_0u_{j+1}\Delta_0u_j\Delta_+({u_j}^2(u_{j+1}+u_{j-1}))$
$= -\sum\Delta_0u_{j+1}\Delta_0u_j[{u_{j+1}}^2\Delta_0u_{j+1} + 2u_{j+1}u_j\Delta_+{u_j}^2\Delta_0u_j]$

$(**) = \sum\Delta_0u_j(u_{j+1}+u_{j-1})[\Delta_+u_j\Delta_0u_{j+1}(u_{j+1}+u_j) + \Delta_+u_{j-1}\Delta_0u_{j-1}(u_j+u_{j-1})]$

$(c) = \sum(\Delta_+{u_j}^2 + \Delta_+{u_{j-1}}^2)(\Delta_+{u_j}^3 - \Delta_+{u_{j-1}}^3)$
$= \sum -\Delta_+{u_j}^2\Delta_+{u_{j-1}}^3 + \Delta_+{u_{j-}}^2\Delta_+{u_j}^3$
$= \sum\Delta_+u_j\Delta_+u_{j-1}[-(u_{j+1}+u_j)b_{j-1} + (u_j+u_{j-1})b_j]$

$= \sum \Delta_+ u_j \Delta_+ u_{j-1} \Delta_0 u_j [u_{j+1}u_{j-1} + u_j u_{j+1} + u_j u_{j-1}]$

$$2\sum F_j G_j \quad \leq \quad \frac{62}{64}\lambda {\epsilon_0}^2 \sum |\Delta_+ u_j^3| \tag{A.8}$$

A.4 termes en λ^4

$$\begin{aligned} 2\sum J_j K_j &= -\frac{C\lambda^4}{32}\sum \Delta_-(|\Delta_+ u_j|\Delta_+ u_j)\Delta_-\Delta_+(u_j^3 \Delta_0 u_j) \\ &\leq \frac{1}{2}\lambda \epsilon_0^3 C \sum |\Delta_+ u_j|^3 \end{aligned} \tag{A.9}$$

$$\begin{aligned} 2\sum F_j K_j &= \frac{\lambda^4}{128}\sum \Delta_0 {u_j}^2 \Delta_-\Delta_+({u_j}^3 \Delta_0 u_j) \\ &= \frac{\lambda^4}{128}\sum ({\Delta_+ u_j}^2 + {\Delta_+ u_{j-1}}^2)(\Delta_+({u_j}^3\Delta_0 u_j) - \Delta_-({u_{j-1}}^3 \Delta_0 u_{j-1})) \\ &= \frac{\lambda^4}{128}\sum -{\Delta_+ u_j}^2 \Delta_+({u_{j-1}}^3 \Delta_0 u_{j-1}) + \Delta_+ u_{j-1} 2\Delta_+({u_j}^3 \Delta_0 u_j) \end{aligned}$$

$\Delta_+({u_j}^3 \Delta_0 u_j) = \Delta_+ {u_j}^3 \Delta_0 u_{j+1} + {u_j}^3 \Delta_+ \Delta_0 u_j$

$$\begin{aligned} 2\sum F_j K_j &= \frac{\lambda^4}{128}\sum -{\Delta_+ u_j}^2 {\Delta_+ u_{j-1}}^3 \Delta_0 u_j - {\Delta_+ u_j}^2 {u_{j-1}}^3 \Delta_+ \Delta_0 u_{j-1} + {\Delta_+ u_{j-1}}^2 \Delta_+ {u_j}^3 \Delta_0 u_{j+1} \\ &\quad + {\Delta_+ u_{j-1}}^2 {u_j}^3 \Delta_+ \Delta_0 u_j \\ &= \frac{\lambda^4}{128}\sum \Delta_+ u_j \Delta_+ u_{j-1}[\Delta_0 u_{j+1} b_j (u_j + u_{j-1}) - \Delta_0 u_j b_{j-1}(u_{j+1} + u_j)] \\ &\quad \underbrace{-\Delta_0 \Delta_+ {u_j}^2 \Delta_+ \Delta_0 u_j {u_j}^3}_{(h)} \end{aligned}$$

$\Delta_0 \Delta_+ {u_j}^2 = \Delta_0 \Delta_+ u_j (u_{j+2} + u_{j+1}) + \Delta_+ u_{j-1}(\Delta_0 u_{j+1} + \Delta_0 u_j)$

$(h) = -\Delta_+ u_{j-1}(\Delta_0 u_{j+1} + \Delta_0 u_j)\Delta_+ \Delta_0 u_j {u_j}^3 - \Delta_+ \Delta_0 u_j \Delta_0 \Delta_+ u_j {u_j}^3 (u_{j+2} + u_{j+1})$

$$2\sum F_j K_j \quad \leq \quad \frac{40}{128}\lambda {\epsilon_0}^3 \sum |\Delta_+ u_j^3| - \frac{\lambda^4}{128}\sum (\Delta_+ \Delta_0 u_j)^2 {u_j}^3 (u_{j+2} + u_{j+1})$$

$\lambda^4 \sum(\Delta_+\Delta_0 u_j)^2 {u_j}^3(u_{j+2}+u_{j+1}) = \sum(\Delta_+\Delta_0 u_j)^2({u_j}^3(u_{j+2}+u_{j+1}) - 2u_{j+1}^4$
$= \lambda^4 \sum(\Delta_+\Delta_0 u_j)^2({u_j}^3\Delta_+ u_{j+1} + u_{j+1} b_j \Delta_+ u_j) + 2[(\Delta_-\Delta_+ u_{j+1})^2 + (\Delta_-\Delta_+ u_j)^2$
$+2\Delta_-\Delta_+ u_{j+1}\Delta_-\Delta_+ u_j]u_{j+1}^4$
$\leq 16\lambda{\epsilon_0}^3 \sum|\Delta_+ u_j^3| + 4\sum(\Delta_-\Delta_+ u_j)^2(u_{j+1}^4 + u_j^4)$

$$\begin{aligned} 2\sum F_j K_j &\leq \frac{56}{128}\lambda{\epsilon_0}^3 \sum|\Delta_+ u_j^3| + \frac{4}{128}\lambda^4 \sum(\Delta_-\Delta_+ u_j)^2(u_{j+1}^4 + u_j^4) \\ &\leq \frac{56}{128}\lambda{\epsilon_0}^3 \sum|\Delta_+ u_j^3| + \frac{4}{128}\lambda^2{\epsilon_0}^2 \sum(\Delta_-\Delta_+ u_j)^2(u_{j+1}^2 + u_j^2) \end{aligned}$$

$$\begin{aligned} \sum {G_j}^2 &= \frac{\lambda^4}{32^2}\sum(\Delta_0({u_j}^2\Delta_0 u_j))^2 + \frac{\lambda^4}{64}\sum(\Delta_-\Delta_+ {u_j}^3)^2 + \frac{\lambda^4}{128}\sum\Delta_0({u_j}^2\Delta_0 u_j)\Delta_-\Delta_+ {u_j}^3 \\ &= \frac{\lambda^4}{32^2}\sum({\Delta_+ u_j}^2\Delta_0 u_{j+1} + {\Delta_+ u_{j-1}}^2\Delta_0 u_{j-1} + {u_j}^2\Delta_0\Delta_0 u_j)^2 \\ &\quad + \frac{\lambda^4}{64}\sum(b_j\Delta_-\Delta_+ u_j + \Delta_- b_j \Delta_+ u_{j-1})^2 \\ &\quad + \frac{\lambda^4}{128}\sum({\Delta_+ u_j}^2\Delta_0 u_{j+1} + {\Delta_+ u_{j-1}}^2\Delta_0 u_{j-1} + {u_j}^2\Delta_0\Delta_+\Delta_0 u_j) \\ &\quad (b_j\Delta_-\Delta_+ u_j + \Delta_- b_j\Delta_+ u_{j-1}) \\ &= \frac{\lambda^4}{32^2}\sum({\Delta_+ u_j}^2\Delta_0 u_{j+1})^2 + ({\Delta_+ u_{j-1}}^2\Delta_0 u_{j-1})^2 + 2{\Delta_+ u_j}^2{\Delta_+ u_{j-1}}^2\Delta_0 u_{j+1}\Delta_+ u_{j-1} \\ &\quad + 2{\Delta_+ u_j}^2\Delta_0 u_{j+1}\Delta_0\Delta_0 u_j {u_j}^2 + 2{\Delta_+ u_{j-1}}^2\Delta_0 u_{j+1}\Delta_0\Delta_0 u_j {u_j}^2 + \frac{\lambda^4}{64}\sum(\Delta_- b_j\Delta_+ u_{j-1})^2 \\ &\quad + 2\Delta_-\Delta_+ u_j\Delta_- b_j\Delta_+ u_{j-1} b_j + \frac{\lambda^4}{128}\sum({\Delta_+ u_j}^2\Delta_0 u_{j+1} + {\Delta_+ u_{j-1}}^2\Delta_0 u_{j-1}) \\ &\quad (\Delta_-\Delta_+ u_j b_j + \Delta_- b_j\Delta_+ u_{j-1}) + {u_j}^2\Delta_0\Delta_0 u_j\Delta_- b_j\Delta_+ u_{j-1} + \frac{\lambda^4}{32^2}\sum(\Delta_0\Delta_0 u_j)^2 \\ &\quad + \frac{\lambda^4}{64}\sum {b_j}^2(\Delta_-\Delta_+ u_j)^2 + \frac{\lambda^4}{128}\sum {u_j}^2 b_j\Delta_0\Delta_0 u_j\Delta_-\Delta_+ u_j \\ &\leq \frac{172}{64}\lambda{\epsilon_0}^3\sum|\Delta_+ u_j^3| + \frac{\lambda^4}{32^2}\sum(\Delta_0\Delta_0 u_j)^2 + \frac{\lambda^4}{64}\sum {b_j}^2(\Delta_-\Delta_+ u_j)^2 \\ &\quad + \frac{\lambda^4}{128}\sum {u_j}^2 b_j\Delta_0\Delta_0 u_j\Delta_-\Delta_+ u_j \end{aligned}$$

$(\Delta_0\Delta_0 u_j)^2 = (\Delta_-\Delta_+ u_{j+1} + 2\Delta_-\Delta_+ u_j + \Delta_-\Delta_+ u_{j-1})^2$

$$\begin{aligned}
\lambda^4 \sum u_j^4 (\Delta_0\Delta_0 u_j)^2 &= \lambda^4 \sum u_j^4 (\Delta_-\Delta_+ u_{j+1})^2 + 4u_j^4 (\Delta_-\Delta_+ u_j)^2 + (\Delta_-\Delta_+ u_{j-1})^2 \\
&\quad +4u_j^4 \Delta_-\Delta_+ u_{j+1} \Delta_-\Delta_+ u_j + 2u_j^4 \Delta_-\Delta_+ u_{j+1} \Delta_-\Delta_+ u_{j-1} \\
&\quad +4u_j^4 \Delta_-\Delta_+ u_j \Delta_-\Delta_+ u_{j-1} \\
&\leq \lambda^4 \sum (u_{j-1}^4 + 4u_j^4 + u_{j+1}^4)(\Delta_-\Delta_+ u_j)^2 + \lambda^4 \frac{4}{2} \sum u_j^4 (\Delta_-\Delta_+ u_{j+1})^2 \\
&\quad +\lambda^4 \frac{4}{2} \sum u_j^4 (\Delta_-\Delta_+ u_j)^2 + \lambda^4 \frac{2}{2} \sum u_j^4 (\Delta_-\Delta_+ u_{j+1})^2 + \lambda^4 \frac{2}{2} \sum u_j^4 (\Delta_-\Delta_+ u_{j-1})^2 \\
&\quad +\lambda^4 \frac{4}{2} \sum u_j^4 (\Delta_-\Delta_+ u_j)^2 + \lambda^4 \frac{4}{2} \sum u_j^4 (\Delta_-\Delta_+ u_{j-1})^2 \\
&\leq \lambda^4 \sum (4u_{j-1}^4 + 8u_j^4 + 4u_{j+1}^4)(\Delta_-\Delta_+ u_j)^2 - 4{u_{j-1}}^2 {u_j}^2 (\Delta_-\Delta_+ u_j)^2 \\
&\quad +4{u_{j-1}}^2 {u_j}^2 (\Delta_-\Delta_+ u_j)^2 \\
&\leq \lambda^4 \sum ((4u_{j-1}^2 {u_j}^2 + 8u_j^4 + 4u_{j+1}^4)(\Delta_-\Delta_+ u_j)^2 - 4{u_{j-1}}^2 \Delta_+ {u_{j-1}}^2 (\Delta_-\Delta_+ u_j)^2 \\
&\leq \lambda^4 \sum ((4u_{j-1}^2 {u_j}^2 + 8u_j^4 + 4u_{j+1}^4)(\Delta_-\Delta_+ u_j)^2 + 32{\epsilon_0}^3 \lambda \sum |\Delta_+ u_j|^3
\end{aligned}$$

$$\begin{aligned}
\lambda^4 \sum u_j^2 b_j \Delta_0\Delta_0 u_j \Delta_-\Delta_+ u_j &= \lambda^4 \sum ({u_j}^2 b_j - 3u_j^4) \Delta_0\Delta_0 u_j \Delta_-\Delta_+ u_j + 3\lambda^4 \sum u_j^4 \Delta_0\Delta_0 u_j \Delta_-\Delta_+ u_j \\
&= \lambda^4 \sum ({u_j}^2 \Delta_+ {u_j}^2 + {u_j}^3 \Delta_+ u_j) \Delta_0\Delta_0 u_j \Delta_-\Delta_+ u_j \\
&\quad +3\lambda^4 \sum u_j^4 \Delta_0\Delta_0 u_j \Delta_-\Delta_+ u_j \\
&\leq 24{\epsilon_0}^3 \lambda \sum |\Delta_+ u_j|^3 + \frac{3}{2}\lambda^4 \sum u_j^4 (\Delta_-\Delta_+ u_{j+1})^2 \\
&\quad +\frac{3}{2}\lambda^4 \sum u_j^4 (\Delta_-\Delta_+ u_j)^2 + 6\lambda^4 \sum u_j^4 (\Delta_-\Delta_+ u_j)^2 \\
&\quad +\frac{3}{2}\lambda^4 \sum u_j^4 (\Delta_-\Delta_+ u_{j-1})^2 + \frac{3}{2}\lambda^4 \sum u_j^4 (\Delta_-\Delta_+ u_j)^2 \\
&\leq 24{\epsilon_0}^3 \lambda \sum |\Delta_+ u_j|^3 + \lambda^4 \sum (\frac{3}{2} u_{j-1}^4 + 9u_j^4 + \frac{3}{2} u_{j+1}^4)(\Delta_-\Delta_+ u_j)^2 \\
&\leq 36{\epsilon_0}^3 \lambda \sum |\Delta_+ u_j|^3 + \lambda^4 \sum (\frac{3}{2} u_{j-1}^2 {u_j}^2 + 9u_j^4 + \frac{3}{2} u_{j+1}^4)(\Delta_-\Delta_+ u_j)^2
\end{aligned}$$

$$\begin{aligned}
\sum {G_j}^2 &= \frac{190}{64}\lambda {\epsilon_0}^3 \sum |\Delta_+ u_j^3| + \lambda^4 \sum (\Delta_-\Delta_+ u_j)^2 [\frac{1}{32^2}(4{u_{j-1}}^2 {u_j}^2 + 8u_j^4 + 4u_{j+1}^4) \\
&\quad +\frac{1}{64}{b_j}^2 + \frac{1}{128}(\frac{3}{2}{u_{j-1}}^2 + 9u_j^4 + \frac{3}{2}{u_{j+1}}^2)] \\
&\leq \frac{190}{64}\lambda {\epsilon_0}^3 \sum |\Delta_+ u_j^3| + \lambda^2 {\epsilon_0}^2 \sum (\Delta_-\Delta_+ u_j)^2 [\frac{42}{256}{u_j}^2 + \frac{22}{256}{u_{j+1}}^2] \quad \text{(A.10)}
\end{aligned}$$

A.5 terme en λ^5

$$2\sum G_jK_j = -\frac{\lambda^5}{16*64}\underbrace{\sum \Delta_0({u_j}^2\Delta_0u_j)\Delta_-\Delta_+({u_j}^3\Delta_0u_j)}_{d} - \frac{\lambda^5}{4*64}\underbrace{\sum \Delta_-\Delta_+{u_j}^3\Delta_-\Delta_+({u_j}^3\Delta_0u_j)}_{e}$$

$$\begin{aligned}
(d) &= \sum[\Delta_+({u_j}^2\Delta_0u_j)+\Delta_+({u_{j-1}}^2\Delta_0u_{j-1})][\Delta_+({u_j}^3\Delta_0u_j)-\Delta_+({u_{j-1}}^3\Delta_0u_{j-1})]\\
&= \sum \Delta_+({u_j}^2\Delta_0u_j)\Delta_+({u_{j-1}}^3\Delta_0u_{j-1})+\Delta_+({u_{j-1}}^2\Delta_0u_{j-1})\Delta_+({u_j}^3\Delta_0u_j)\\
&= \sum -({u_{j+1}}^2\Delta_0u_{j+1}-{u_j}^2\Delta_0u_j)({u_j}^3\Delta_0u_j-{u_{j-1}}^3\Delta_0u_{j-1})\\
&\quad +({u_j}^2\Delta_0u_j-{u_{j-1}}^2\Delta_0u_{j-1})({u_{j+1}}^3\Delta_0u_{j+1}-{u_j}^3\Delta_0u_j)\\
&= \sum \Delta_0u_{j+1}\Delta_0u_j\Delta_+u_j{u_j}^2{u_{j+1}}^2-\Delta_0u_{j-1}\Delta_0u_{j+1}\Delta_0u_j{u_{j-1}}^2{u_{j+1}}^2\\
&\quad +\Delta_0u_j\Delta_0u_{j-1}\Delta_+u_{j-1}{u_j}^2{u_{j-1}}^2
\end{aligned}$$

$$\begin{aligned}
(e) &= \sum(\Delta_+{u_j}^3-\Delta_+{u_{j-1}}^3)(\Delta_+({u_j}^3\Delta_0u_j)-\Delta_+({u_{j-1}}^3\Delta_0u_{j-1}))\\
&= \sum 2\Delta_+{u_j}^3\Delta_+({u_j}^3\Delta_0u_j)-\Delta_+{u_j}^3\Delta_+({u_{j-1}}^3\Delta_0u_{j-1})-\Delta_+{u_{j-1}}^3\Delta_+({u_j}^3\Delta_0u_j)
\end{aligned}$$

$$\Delta_+({u_j}^3\Delta_0u_j) = \Delta_+{u_j}^3\Delta_0u_{j+1}+{u_j}^3\Delta_+\Delta_0u_j$$

$$\begin{aligned}
(e) &= \sum(\Delta_+{u_j}^3)^2\Delta_0u_{j+1}+2\Delta_+{u_j}^3\Delta_+\Delta_0u_j{u_j}^3-\Delta_+{u_j}^3\Delta_+{u_{j-1}}^3\Delta_0u_j-\Delta_+{u_j}^3\Delta_+\Delta_0u_{j-1}{u_{j-1}}^3\\
&\quad -\Delta_+{u_{j-1}}^3\Delta_+{u_j}^3\Delta_0u_{j+1}-\Delta_+{u_{j-1}}^3\Delta_+\Delta_0u_j{u_j}^3\\
&= \sum 2(\Delta_+u_j)^2\Delta_0u_{j+1}{b_j}^2-\Delta_+u_j\Delta_+u_{j-1}b_jb_{j-1}(\Delta_0u_j+\Delta_0u_{j+1})\\
&\quad -\underbrace{\Delta_-\Delta_+(\Delta_+{u_j}^3)\Delta_+\Delta_0u_j{u_j}^3}_{(f)}
\end{aligned}$$

$$\begin{aligned}
(f) &= \sum(\Delta_+\Delta_+{u_j}^3-\Delta_+\Delta_+{u_{j-1}}^3)(\Delta_+\Delta_+u_j+\Delta_+\Delta_+u_{j-1}){u_j}^3\\
&= \sum \Delta_+\Delta_+{u_j}^3\Delta_+\Delta_+u_j{u_j}^3+\Delta_+\Delta_+{u_j}^3\Delta_+\Delta_+u_{j-1}{u_j}^3-\Delta_+\Delta_+{u_{j-1}}^3\Delta_+\Delta_+u_j{u_j}^3\\
&\quad -\Delta_+\Delta_+{u_{j-1}}^3\Delta_+\Delta_+u_{j-1}{u_j}^3\\
&= \sum -\Delta_+\Delta_+{u_j}^3\Delta_+\Delta_+u_j\Delta_+{u_j}^3+{u_j}^3(\Delta_+\Delta_+\Delta_+{u_j}^3\Delta_+\Delta_+u_{j-1}-\Delta_+\Delta_+{u_{j-1}}^3\Delta_+\Delta_+u_j)
\end{aligned}$$

$$\Delta_+\Delta_+{u_j}^3 = \Delta_+b_j\Delta_+u_{j+1}+b_j\Delta_+\Delta_+u_j$$

$$\begin{aligned}(f) &= \sum -\Delta_+ b_j \Delta_+ u_{j+1} \Delta_+ \Delta_+ u_j \Delta_+ {u_j}^3 - b_j (\Delta_+ \Delta_+ u_j)^2 \Delta_+ {u_j}^3 + {u_j}^3 (\Delta_+ b_j \Delta_+ u_{j+1} \Delta_+ \Delta_+ u_{j-1} \\ &\quad + b_j \Delta_+ \Delta_+ u_j \Delta_+ \Delta_+ u_{j-1} - \Delta_+ b_{j-1} \Delta_+ u_j \Delta_+ \Delta_+ u_j - b_{j-1} \Delta_+ \Delta_+ u_{j-1} \Delta_+ \Delta_+ u_j) \\ &= \sum -\Delta_+ u_{j+1} \Delta_+ \Delta_+ u_j \Delta_+ u_j \Delta_+ b_j b_j - (\Delta_+ \Delta_+ u_j)^2 \Delta_+ u_j {b_j}^2 + {u_j}^3 (\Delta_+ b_j \Delta_+ u_{j+1} \Delta_+ \Delta_+ u_{j-1} \\ &\quad - \Delta_+ b_{j-1} \Delta_+ u_j \Delta_+ \Delta_+ u_j + \Delta_+ \Delta_+ u_{j-1} \Delta_+ \Delta_+ u_j \Delta_+ b_{j-1})\end{aligned}$$

$\Delta_+ b_j = \Delta_0 u_{j+1}(u_{j+2} + u_{j+1} + u_j)$

$$2 \sum G_j K_j \leq \frac{59}{64} \lambda {\epsilon_0}^4 \sum |\Delta_+ u_j^3| \qquad \text{(A.11)}$$

A.6 terme en λ^6

$$\begin{aligned}\sum {K_j}^2 &= \frac{\lambda^6}{64^2} \sum (\Delta_- \Delta_+ (u_j^3 \Delta_0 u_j))^2 \\ &= \frac{\lambda^6}{64^2} \sum (\Delta_+ (u_j^3 \Delta_0 u_j) - \Delta_+ (u_{j-1}^3 \Delta_0 u_{j-1}))^2 \\ &= \frac{\lambda^6}{64^2} \sum 2(\Delta_+ (u_j^3 \Delta_0 u_j))^2 - 2\Delta_+ (u_j^3 \Delta_0 u_j) \Delta_+ (u_{j-1}^3 \Delta_0 u_{j-1}) \\ &= \frac{\lambda^6}{64^2} \sum 2(\Delta_+ {u_j}^3 \Delta_0 u_{j+1})^2 + 4\Delta_+ {u_j}^3 \Delta_0 u_{j+1} \Delta_+ \Delta_0 u_j {u_j}^3 + 2{u_j}^3 \Delta_+ uj_3 \Delta_0 u_{j+1} \Delta_+ \Delta_0 u_j \\ &\quad -2\Delta_+ {u_j}^3 \Delta_+ {u_{j-1}}^3 \Delta_0 u_{j+1} \Delta_0 u_j - 2\Delta_+ {u_j}^3 \Delta_0 u_{j+1} \Delta_+ \Delta_0 u_{j-1} {u_{j-1}}^3 \\ &\quad -2{u_j}^3 \Delta_+ \Delta_0 u_j \Delta_+ {u_{j-1}}^3 \Delta_0 u_j - \underbrace{\frac{\lambda^6}{64^2} \sum 2{u_j}^3 {u_{j-1}}^3 \Delta_+ \Delta_0 u_j \Delta_+ \Delta_0 u_{j-1}}_{(g)}\end{aligned}$$

$$\begin{aligned}(g) &= -\frac{\lambda^6}{64^2} \sum 2({u_j}^3 {u_{j-1}}^3 - u_j^6 + u_j^6) \Delta_+ \Delta_0 u_j \Delta_+ \Delta_0 u_{j-1} \\ &= \frac{\lambda^6}{64^2} \sum 2{u_j}^3 b_{j-1} \Delta_+ u_{j-1} \Delta_+ \Delta_0 u_j \Delta_+ \Delta_0 u_{j-1} \\ &\quad -\frac{\lambda^6}{64^2} \sum u_j^6 (\Delta_- \Delta_+ u_{j+1} + \Delta_- \Delta_+ u_j)(\Delta_- \Delta_+ u_j + \Delta_- \Delta_+ u_{j-1}) \\ |(g)| &\leq \frac{8}{64^2} \lambda {\epsilon_0}^5 \sum |\Delta_+ u_j|^3 + \frac{\lambda^6}{64^2} \sum (u_{j-1}^6 + 2u_j^6 + u_{j+1}^6)(\Delta_- \Delta_+ u_j)^2 \\ |(g)| &\leq \frac{8}{64^2} \lambda {\epsilon_0}^5 \sum |\Delta_+ u_j|^3 + \frac{\lambda^6}{64^2} \sum (u_{j-1}^4 {u_j}^2 + 2u_j^6 + u_{j+1}^6)(\Delta_- \Delta_+ u_j)^2 \\ &\quad -\frac{\lambda^6}{64^2} \sum u_{j-1}^4 \Delta_+ {u_{j-1}}^2 (\Delta_- \Delta_+ u_j)^2 \\ |(g)| &\leq \frac{16}{64^2} \lambda {\epsilon_0}^5 \sum |\Delta_+ u_j|^3 + \frac{\lambda^2 {\epsilon_0}^4}{64^2} \sum (3{u_j}^2 + u_{j+1}^2)(\Delta_- \Delta_+ u_j)^2\end{aligned}$$

$$\sum {K_j}^2 \leq \frac{280}{64^2}\lambda {\epsilon_0}^5 \sum |\Delta_+ u_j^3| + \frac{\lambda^2 {\epsilon_0}^4}{64^2}\sum (3{u_j}^2 + u_{j+1}^2)(\Delta_-\Delta_+ u_j)^2 \tag{A.12}$$

A.7 On somme tout...

$$\begin{aligned}\sum u_j^{n+1} + {}^2 - \sum u_j^{n2} \leq\ & \lambda(-2C + 1/6 + 8C^2\epsilon_0 + 4C\epsilon_0 + \frac{14}{16}\epsilon_0 + \frac{7}{2}C\epsilon_0^2 + \frac{3}{32}\epsilon_0^2 + \frac{62}{64}\epsilon_0^2 \\ & + \frac{1}{2}C\epsilon_0^3 + \frac{56}{128}\epsilon_0^3 + \frac{190}{64}{\epsilon_0}^3 + \frac{59}{64}\epsilon_0^4 + \frac{280}{64^2}\epsilon_0^5) \\ & + \lambda^2 \sum (\Delta_-\Delta_+ u_j)^2 (\frac{3}{32}({u_j}^2 + {u_{j+1}}^2) + \frac{4}{128}({u_j}^2 + {u_{j+1}}^2) \\ & + {\epsilon_0}^2(\frac{42}{256}{u_j}^2 + \frac{22}{256}{u_{j+1}}^2) + \frac{1}{64^2}\epsilon_0^4(3{u_j}^2 + {u_{j+1}}^2))\end{aligned} \tag{A.13}$$

On vérifie que le terme en λ^2 est négatif si $\epsilon_0 \leq 0.79$.

Le terme en λ est négatif si C est compris entre les racines, si elles existent, du polynôme :

$$P(x) = 8\epsilon_0 x^2 + (-2 + 4\epsilon_0 + \frac{7}{2}{\epsilon_0}^2 + \frac{1}{3}{\epsilon_0}^3)x + \frac{1}{6} + \frac{14}{16}\epsilon_0 + \frac{68}{64}{\epsilon_0}^2 + \frac{220}{64}\epsilon_0^3 + \frac{196}{256}\epsilon_0^4 + \frac{280}{64^2}\epsilon_0^5$$

On vérifie que P a des racines réelles si $\epsilon_0 \leq 0.15$. Dans ce cas, $C = 0.6$ convient.

Annexe B

Toward implicit subgrid scale modeling by particle methods

B.1 Résumé

La Simulation des Grandes Echelles vise à ne calculer que les échelles significatives des écoulements turbulents, c'est-à-dire celles qui peuvent être résolues avec la discrétisation (pas d'espace et pas de temps) utilisé. L'effet des échelles non résolues sur ces grandes échelles devant être pris en compte, il est paramétré par ce qu'on appelle un modèle sous-maille. Ce dernier permet notamment de dissiper l'énergie s'accumulant au bout du spectre des grandes échelles.

On considère qu'un modèle sous-maille est explicite si ses termes sont explicitement discrétisés et ajoutés au schéma numérique. De nombreux modèles sous-mailles explicites ont été développés sans tenir compte ni du schéma numérique, ni de la grille de calcul utilisés pour résoudre les équations. Or il a été montré [34, 49] que l'erreur de troncature du schéma numérique utilisé peut, si elle est suffisamment grande, contrebalancer l'action d'un modèle sous-maille explicite. Une alternative aux modèles explicites consiste à ajuster l'erreur de troncature du schéma numérique pour l'utiliser comme un modèle sous-maille. Cette approche, qui tire parti de la manière dont on discrétise les équations à résoudre, évite le calcul explicite d'un terme supplémentaire correspodant au modèle sous-maille, et porte le nom de Simulation Implicite des Grandes Echelles.

Récemment, une méthode de Simulation Implicite des Grandes Echelles, nommée Adaptive Local Deconvolution Method (ALDM) [2, 40], a été introduite. Il s'agit d'une méthode de volumes finis, qui utilise la similarité entre d'une part le filtrage et la déconvolution en modélisation sous-maille, et d'autre part les procédés de moyenne et de reconstruction propres aux volumes finis, pour établir un modèle sous-maille implicite. Une reconstruction locale, adaptative, de type WENO est utilisée dans cette méthode. Les poids attribués aux différents polynomes utilisés dans cette reconstruction sont pondérés par des coefficients qui constituent les paramètres de la modélisation sous-maille.

Ces paramètres ont été choisis de manière à ajuster la dissipation spectrale numérique du schéma à la diffusivité turbulente théorique [12], afin que l'erreur de troncature du schéma prenne une signification physique.

On se propose ici d'adapter une méthode Vortex-In-Cell à la Simulation Implicite des Grandes Echelles, à l'aide de cette méthodologie. Les méthodes Vortex-In-Cell utilisent des noyaux d'interpolation pour reconstruire le champ de vitesse à partir du champ de vorticité et pour remailler les particules. Ce procédé d'interpolation, de part son action régularisante, peut être considéré comme participant à un effet sous-maille. Il a d'ailleurs déjà été remarqué [20] que les méthodes Vortex-In-Cell se comportent comme des modèles sous-maille lors de calculs sous-résolus.

Dans un premier temps, on analyse donc la viscosité spectrale numérique de la méthode Vortex-In-Cell associée à différents noyaux d'interpolation couramment utilisés dans le cadre des méthodes particulaires pour la mécanique des fluides incompressibles [17]. Puis on cherche à optimiser la forme de la viscosité numérique spectrale produite par une combinaison linéaire de ces différents noyaux d'interpolation, en la comparant à la diffusivité turbulente prédite par la théorie.

Le résultat de cette optimisation est testé avec la méthode Vortex-In-Cell sur plusieurs problèmes : tout d'abord, sur le tourbillon de Taylor-Green, en deux et trois dimensions, puis sur un cas de turbulence isotrope décroissante reproduisant l'expérience de Comte-Bellot et Corrsin. Les résultats pour le tourbillon de Taylor-Green 2D montrent que la décroissance de l'énergie cinétique obtenue avec la méthode est en excellent accord avec sa valeur théorique. Le taux de dissipation de l'énergie cinétique obtenu avec la méthode pour le tourbillon de Taylor-Green 3D est en bon accord avec les résultats de Brachet [10], issus d'une simulation numérique directe. Pour le cas de turbulence isotrope décroissante, les spectres énergétiques obtenus avec la méthode Vortex-In-Cell et le noyau d'interpolation optimisé sont passablement en accord avec les résultats de Wray [86]. Cependant, ils sont moins bons que ceux obtenus avec la même méthode et le classique noyau d'interpolation M'_4, utilisé seul.

Cette constatation nous laisse supposer que le procédé de combinaison linéaire de noyaux d'interpolation ne permet pas d'agir suffisamment sur la viscosité spectrale numérique pour améliorer le comportement sous-maille de la méthode Vortex-In-Cell. Pour mieux ajuster la viscosité numérique spectrale à la diffusivité turbulente théorique, nous pensons qu'il faudrait augmenter l'adaptivité locale de la méthode, afin d'obtenir une modélisation sous-maille non-linéaire.

B.2 Abstract

The numerical truncation error of vortex-in-cell methods is analyzed *a-posteriori* through the effective spectral numerical viscosity for simulations of three-dimensional

isotropic turbulence. The interpolation kernels used for velocity-smoothing and remeshing are identified as the most relevant components affecting the shape of the spectral numerical viscosity as a function of wave number. A linear combination of well-known standard kernels leads to new kernels assigned to the specific use for implicit large-eddy simulation of turbulent flows, i.e. their truncation errors acts as subgrid-scale model. Numerical results are provided to show the potential and drawbacks of the approach.

B.3 Introduction

In Large-Eddy Simulation (LES) of turbulent flows the evolution of non-universal larger scales is computed, whereas their interaction with unresolved subgrid scales (SGS) is modeled. Modeling SGS effects amounts to the modification of the underlying conservation laws to remove the coupling with spatial and temporal scales which cannot be resolved by the numerical grid functions. A mathematical framework based on explicit filtering was proposed by Leonard [51]. Following his concept, explicit SGS models were commonly derived without reference to a computational grid and without taking into account a discretization scheme. It should be noted that Leonard's ansatz implies a *subsequent* discretization of the filtered equations. However, numerically computed SGS stresses are affected by the truncation error of the discretization scheme [34]. This interference between subgrid-scale model and numerical discretization can result in an unpredictable lack of accuracy of the LES. Along with the change of turbulence structure near flow boundaries, such as walls or fluid interfaces, it constitutes a major obstacle for the application of LES for standard use in industrial research and development.

SGS effects are modeled *explicitly* if the underlying conservation law is modified and subsequently discretized. Employing an explicit SGS model, Schumann [79] already argued that discretization effects should be taken into account within the SGS model formulation. Systematic investigation of the problem [34], [49] showed that within the standard formulation of eddy-viscosity SGS models a second-order accurate numerical discretization is insufficient in general.

As an alternative to standard formulations, Implicit Large-Eddy Simulation (ILES) denotes the situation when the unmodified conservation law is discretized and the numerical truncation error acts as SGS model. Since the SGS model is *implicitly* contained within the discretization scheme an explicit computation of model terms becomes unnecessary. Implicit SGS modeling requires systematic procedures for design and analysis of appropriate discretization schemes. A connection between grid-truncation and the filtering approach can be made through numerical methods which employ an inherent filter operation. There are two numerical methods with inherent filtering : the finite-volume method and the particle method.

One recent approach employing a finite-volume discretization is the Adaptive Local Deconvolution Method (ALDM) [2, 41]. ALDM was developed using a systematic framework for design, analysis, and optimization of nonlinear discretization schemes

for implicit LES. In this framework parameters inherent to the discretization scheme are determined in such a way that the numerical truncation error acts as a physically motivated SGS model. Optimal model parameters for ALDM were determined by systematically minimizing a cost function which measures the difference between spectral numerical viscosity of the discretization and the eddy viscosity from EDQNM theory for isotropic turbulence. With the optimized discretization parameters ALDM matches the theoretical requirements of EDQNM. The spectral eddy viscosity of the ALDM scheme exhibits a low-wavenumber plateau at the correct level and reproduces the typical cusp shape up to the cut-off wave number at the correct magnitude, see figure B.1b. Computational results for LES with ALDM show that an implicit model can perform at least as well as established explicit models [41, 40]. For a comprehensive review on alternative approaches to ILES, see the textbook of Grinstein [59].

The present paper is dedicated to the extension of the implicit modeling methodology to particle methods for the LES of incompressible turbulent flows. Particle methods are Lagrangian techniques that have been proposed as an alternative to more conventional grid-based methods. Their conceptional advantages over purely grid-based methods are self-adaptivity and numerical stability. Due to the Lagrangian formulation, the inertial non-linear term in the flow equations is implicitly accounted for by the transport of particles such that the resulting method is not subject to the convective stability constraint.

Being a Lagrangian method, the convection of particles requires the reconstruction of the velocity field. The choice of Particle-In-Cell (PIC) methods, rather than totally grid-free particle methods, is dictated by the fact that for three-dimensional incompressible flows particle methods are in general based on the vorticity formulations of the Navier-Stokes equations. The velocity calculation is much less expensive when it relies on grid-based Poisson solver than when it uses N-body type summation formulas. In the context of LES the PIC approach inherently contains a kernel-regularization for reconstructing the velocity field from the vorticity field. The discretization particles are convected with this smoothed velocity field, thus providing a regularized solution [17]. Cottet [15] has demonstrated that a regularized kernel amounts to a modified transport equation whose structure resembles the Navier-Stokes-α model of Domaradzki and Holm [27]. Additional contributions to modeling of non-resolved scales arise from a re-meshing procedure required to maintain consistency of the discretization. Analyzing the modified differential equation Cottet and Weynans [22] found that re-meshing once each time step results in an effective discretization which is equivalent to well-known finite-difference schemes for the Eulerian formulation.

An approach to the optimization of these two contributions with respect to implicit SGS modeling is detailed in the following.

B.4 The vortex-in-cell method

We consider the evolution of a vorticity field in a three dimensional incompressible viscous flow. The Navier-Stokes equations in velocity-vorticity formulation read

$$\frac{\partial \omega}{\partial t} + u \cdot \nabla \omega - \omega \cdot \nabla u - \nu \nabla \cdot \nabla \omega = 0 \ , \tag{B.1a}$$

$$u = \nabla \times \psi \ , \tag{B.1b}$$

$$\nabla \cdot \nabla \psi = -\omega \ , \tag{B.1c}$$

where ω is the vorticity, ψ is the vector potential, u is the velocity, and ν is the kinematic viscosity. This system has to be supplemented with appropriate initial and boundary conditions. Throughout the present paper the computational domain is a triply-periodic box with periodicity length $L = 2\pi$.

Vortex-in-Cell (VIC) methods are based on the Lagrangian formulation of these equations, which has the advantage of a robust and accurate treatment of the convection terms in eq. (B.1a). That is, only vortex stretching $\omega \cdot \nabla u$ and diffusion $\nu \nabla \cdot \nabla \omega$ are computed explicitly, whereas the vorticity flux $u \cdot \nabla \omega$ is inherently accounted for by the advection of particles

$$\frac{\partial \omega}{\partial t} + u \cdot \nabla \omega = \frac{D\omega}{Dt} \ . \tag{B.2}$$

The Poisson equations (B.1c) are solved on an Eulerian grid. Periodic boundary conditions allow to employ fast FFT-based Poisson solvers.

For an inviscid flow, Helmholtz's and Kelvin's theorems assert the conservation of the circulation of material elements. More specifically, for VIC methods the idea is to use particles to transport approximately conserved quantities (circulation) and grid based formulas to compute auxiliary fields (velocity and strain). Transfers between grid and particles are carried out with interpolation formulas, which will be described in the following section. The computational domain is sampled into cells with volume v_p for which the circulation $\int \omega(.,t)dx$ is represented by a single particle. Thus the particle approximation of the vorticity field is

$$\omega(x) = \sum_p \alpha_p \delta(x - x_p) = \sum_p v_p \ \omega_p \delta(x - x_p) \ . \tag{B.3}$$

where x_p ,α_p ,ω_p and v_p denote the location, circulation, vorticity and volume of the respective particle.

The Lagrangian formulation of particle methods avoids the explicit discretization of the convective term in the governing transport equations and the associated stability constraints. The particle positions are modified according to the local flow. Large velocity gradients lead to local accumulation or spreading of particles, which result in a loss of accuracy of the computation. Non-linear stability implies that particle paths are not allowed to cross, which results in the time-step constraint $dt \leq C||\nabla u||_\infty^{-1}$. This condition is often less demanding than classical CFL conditions. However, it implies

that maintaining a regular particle distribution is an important issue. Therefore rather irregular particle positions are projected onto a regular mesh by interpolating the particle data onto the grid. Thereafter new particles are created at regularly distributed positions. Solving the diffusion part of the equations uses explicit solvers that are stable under the condition $\nu dt \leq Ch^{-2}$, which is generally not a severe limitation for 3D computations with moderate to high Reynolds numbers.

To summarize, each time step of the method consists of the following sequence :

- Interpolate particle vorticity onto the grid.
- Compute vector potential, velocity, and strain on the grid using an FFT-based Poisson solver.
- Interpolate velocity and strain at the particle positions.
- Compute updated values of particle vorticity and location.
- Re-mesh particles on the grid.
- Solve viscous part of the equations with a Particle Strength Exchange (PSE) algorithm.

For more details the reader is referred to [13], [17], and [68].

B.5 Interpolation formulas

The computation of velocity and strain and the creation of new particles through the re-gridding process requires frequent transfers between grid and particles. All these operations are performed with the same interpolation kernels. Assume, for instance, that we are given a distribution of particles located at x_p carrying circulations α_p. With an interpolation kernel W the new circulations $\tilde{\alpha}_i$ at the grid points $\tilde{x}_i$ are computed from the following formula

$$\tilde{\alpha}_i = \sum_p \alpha_p W(\frac{\tilde{x}_i - x_p}{h}) \ , \tag{B.4}$$

where h is the grid size. The smoothing length of W is proportional to h in general. As in our case the volumes of the particles are equal to the volumes of the grid cells, eq. (B.4) is equivalent to

$$\tilde{\omega}_i = \sum_p \omega_p W(\frac{\tilde{x}_i - x_p}{h}) \ . \tag{B.5}$$

To obtain the particle velocities u_p from the grid values $\tilde{u}_i$, we compute

$$u_p = \sum_i \tilde{u}_i W(\frac{x_p - \tilde{x}_i}{h}) \ . \tag{B.6}$$

As the two- or three-dimensional interpolation kernels are obtained by tensorial products of one-dimensional kernels, we now focus, for the sake of simplicity, on explanations about the one-dimensional formulas.

A natural way to control the accuracy of re-gridding formulas is by enforcing the original and re-meshed particles to share the same total circulation, linear momentum,

angular momentum, and so on

$$\begin{aligned} \sum \tilde{\alpha}_i &= \sum \alpha_p \\ \sum \tilde{\alpha}_i (x - \tilde{x}_i) &= \sum \alpha_p (x - x_p) \\ \sum \tilde{\alpha}_i (x - \tilde{x}_i)^2 &= \sum \alpha_p (x - x_p)^2 \\ &\ldots \quad . \end{aligned}$$

A hierarchy of interpolating kernels with increasing number of preserved moments and increasing order of accuracy can be constructed this way [17]. The three-point interpolation kernel conserving the first three moments is

$$\Lambda_2(x) = \begin{cases} 1 - x^2 & \text{if } |x| \leq 0.5 \\ (1 - |x|)(2 - |x|)/2 & \text{if } 0.5 < |x| \leq 1.5 \\ 0 & \text{if } |x| \geq 1.5 \end{cases} \tag{B.7}$$

Although successfully used in the past this kernel has a major drawback. It is not continuous and can create or amplify spurious oscillations. There are two possibilities to avoid this problem : increase the number of preserved moments, or increase the regularity of the interpolation function. The following kernel, which uses four points and preserves one additional moment, is continuous

$$\Lambda_3(x) = \begin{cases} (1 - x^2)(2 - |x|)/2 & \text{if } |x| \leq 1 \\ (1 - |x|)(2 - |x|)(3 - |x|)/6 & \text{if } 1 < |x| \leq 2 \\ 0 & \text{if } |x| \geq 2 \end{cases} \tag{B.8}$$

A more regular kernel is M_4. It uses four points, is C^2, but preserves only two moments

$$M_4(x) = \begin{cases} (2 - |x|)^3/6 - 4(1 - |x|)^3/6 & \text{if } |x| \leq 1 \\ (2 - |x|)^3/6 & \text{if } 1 < |x| \leq 2 \\ 0 & \text{if } |x| \geq 2 \end{cases} \tag{B.9}$$

Monaghan proposed the well-known M_4' from a combination of M_4 and its first derivative. This kernel requires four grid points, is C^1 and preserves three moments

$$M_4'(x) = \begin{cases} 1 - 5x^2/2 + 3|x|^3/2 & \text{if } |x| \leq 1 \\ (2 - |x|)^2(1 - |x|)/2 & \text{if } 1 < |x| \leq 2 \\ 0 & \text{if } |x| \geq 2 \end{cases} \tag{B.10}$$

B.6 Analysis in spectral space

In the following we briefly summarize the methodology of a modified-differential equation (MDE) analysis in spectral space. For details on the employed algorithm we refer to [41].

We consider the discretization of the Navier-Stokes equation (B.1a-B.1c) on a $(2\pi)^3$-periodic domain. Once a numerical scheme is defined, the modified-differential-equation analysis leads to a differential equation for the velocity u in the form of

$$\frac{\partial u}{\partial t} + \frac{\partial F(u)}{\partial x} = \mathcal{G}_N \ , \tag{B.11}$$

where $F(u)$ is the exact flux function and $\mathcal{G}_N$ is the truncation error of the discretization. If $\mathcal{G}_N$ approximates the divergence of the subgrid stress tensor in some sense for a finite grid size we obtain an implicit subgrid-scale model contained within the discretization. Using Fourier transforms the MDE can be written in spectral space as

$$\frac{\partial \widehat{u}_C}{\partial t} + \mathrm{i}P(\xi) \cdot \cdot \ \widehat{u_C u_C} + \nu \xi^2 \widehat{u}_C = \widehat{\mathcal{G}}_C \tag{B.12}$$

The hat denotes the Fourier transform, i is the imaginary unit, and ξ is the wave-number vector. The tensor $P(\xi)$ is defined by $P_{lmn}(\xi) = \xi_m \delta_{ln} - \xi_l \xi_m \xi_n \ |\xi|^{-2}$. $\xi_C = N/2$ is the cut-off wave number and N is the corresponding number of grid points in one dimension. A physical-space discretization covers contributions to the numerical solution to wave numbers up to $|\xi| = \sqrt{3}\xi_C$. On the represented wave-number range the kinetic energy is

$$\widehat{E}(\xi) = \frac{1}{2} \widehat{u}_C(\xi) \cdot \widehat{u}_C^*(\xi) \tag{B.13}$$

Multiplying equation (B.12) by the complex-conjugate $\widehat{u}_C^*$ of $\widehat{u}_C$ we obtain

$$\frac{\partial \widehat{E}(\xi)}{\partial t} - \widehat{T}_C(\xi) + 2\nu\xi^2 \widehat{E}(\xi) = \widehat{u}_C^*(\xi) \cdot \widehat{\mathcal{G}}_C(\xi) \ , \tag{B.14}$$

where $\widehat{T}_C(\xi)$ is the nonlinear energy transfer and $\varepsilon_{vis}(\xi) = 2\nu\xi^2 \widehat{E}(\xi)$ is the molecular dissipation. The right-hand side of this equation is the numerical dissipation

$$\varepsilon_{num}(\xi) = \widehat{u}_C^*(\xi) \cdot \widehat{\mathcal{G}}_C(\xi) \tag{B.15}$$

implied by the discretization. Now we investigate how to model the physical subgrid dissipation ε_{SGS} by ε_{num}.

An exact match between ε_{num} and ε_{SGS} cannot be achieved since ε_{SGS} involves interactions with non-represented scales. For modeling it is therefore necessary to invoke theoretical energy-transfer expressions. Employing an eddy-viscosity hypothesis the subgrid-scale dissipation is

$$\varepsilon_{SGS}(\xi) = 2\nu_{SGS}\xi^2 \widehat{E}(\xi) \tag{B.16}$$

Similarly, the numerical dissipation can be expressed as

$$\nu_{num} = \frac{\varepsilon_{num}(\xi)}{2\xi^2 \widehat{E}(\xi)} \ . \tag{B.17}$$

In general ν_{num} is a function of the wavenumber vector ξ. For isotropic turbulence, however, statistical properties of eq. (B.14) follow from the scalar evolution equation for the 3D energy spectrum

$$\frac{\partial \widehat{E}(\xi)}{\partial t} - \widehat{T}_C(\xi) + 2\nu\xi^2 \widehat{E}(\xi) = \varepsilon_{num}(\xi) \ . \tag{B.18}$$

For a given numerical scheme $\nu_{num}(\xi)$ can be computed from

$$\nu_{num}(\xi) = -\frac{1}{2\xi^2 \widehat{E}(\xi)} \int\limits_{|\xi|=\xi} \widehat{u}_C^*(\xi) \cdot \widehat{\mathcal{G}}_N(\xi) d\xi \tag{B.19}$$

Convenient for our purposes is a normalization by

$$\nu^+_{num}(\xi^+) = \nu_{num}\left(\xi^+ \ \xi_C\right) \sqrt{\frac{\xi_C}{\widehat{E}(\xi_C)}} \quad , \ \text{with} \quad \xi^+ = \frac{\xi}{\xi_C} \ . \tag{B.20}$$

In the following the objective is to define model parameters and to adjust them in such a way that the implicit model's dissipative properties are consistent with analytical theories of turbulence. The concept of a wavenumber-dependent spectral eddy viscosity was first proposed by Heisenberg [39]. For high Reynolds numbers and under the assumption of a Kolmogorov range Chollet [12] proposes the expression

$$\nu^+_{Chollet}(\xi^+) = 0.441 C_K^{-3/2} \left(1 + 34.47 e^{3.03\xi^+}\right) \tag{B.21}$$

as best fit to the exact solution, where general shape and the pre-factor $0.441 C_K^{-3/2}$ is also supported by EDQNM turbulence theory [52].

The spectral numerical viscosity of a discretization scheme is analyzed in the specific real-world environment of a numerical simulations of three-dimensional freely decaying homogeneous isotropic turbulence. For this purpose a trial computation is advanced for a small number of time steps. An *a-posteriori* analysis of the data allows to identify the spectral eddy viscosity inherent in the discretization [41]. This method can be applied to most numerical schemes. Since the analysis is performed in spectral space, only a dealiased spectral discretization will yield $\nu^+_{num}(\xi^+) \equiv 0$. We have analyzed the numerical viscosity of several standard schemes, such as central finite difference methods (FD), see figure B.1. The results are consistent with the analysis performed by Ghosal [34].

B.7 Choice and optimization of parameters

Interpolation kernels are a key element of the vortex-in-cell method. In the first part of this section we analyze the spectral numerical viscosity associated with well-established interpolation kernels when used within the VIC method. In the second part, a new kernel for implicit LES is derived from the linear combination of standard kernels. Pursuing this natural approach, the weights are parameters of an implicit sub-grid

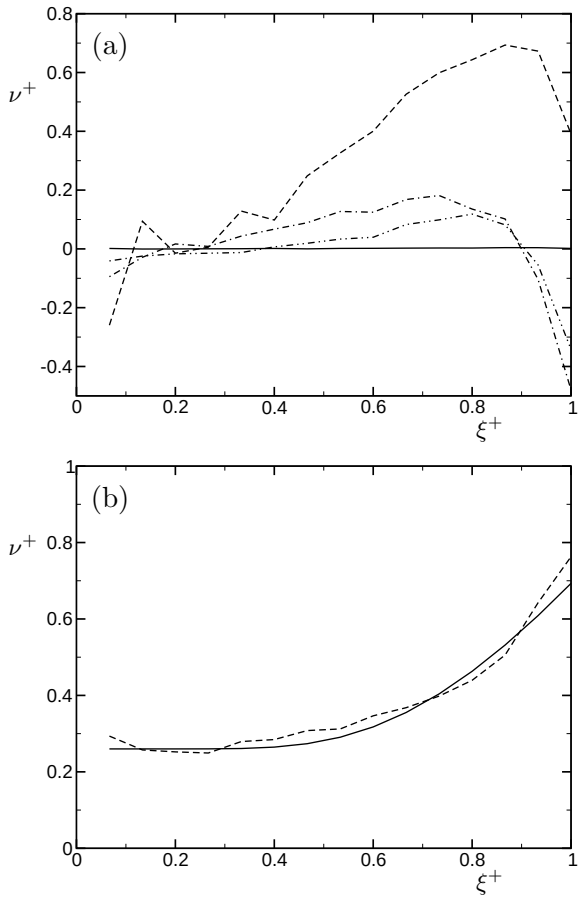


FIG. B.1 – Spectral numerical viscosity of (a) —— de-aliased spectral method, ·—·—·— 2nd-order central FD, —··—·· 4th-order central FD, – – – – 2nd-order central FD with staggered grid. (b) —— EDQNM theory [**?**], – – – – ALDM with optimized parameters [41].

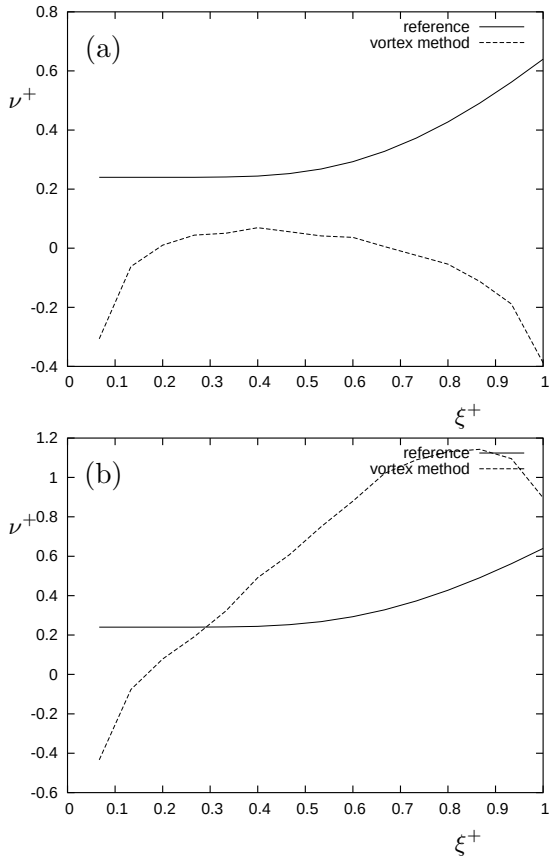


FIG. B.2 – Spectral numerical viscosity of the VIC method with (a) M_4' and (b) Λ_3.

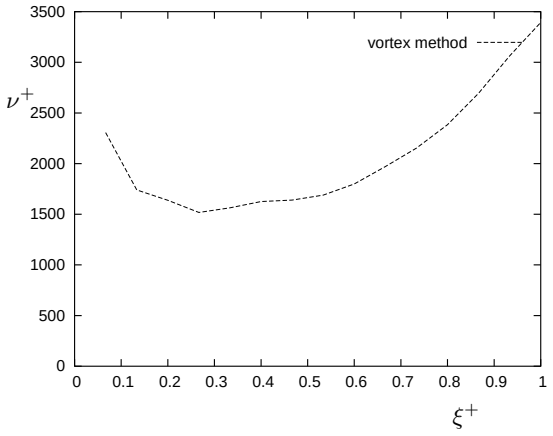


FIG. B.3 – Spectral numerical viscosity of the VIC method with M_4.

scale model.

Figure B.2 shows the results obtained with M_4' and Λ_3. The values of the numerical spectral viscosity are acceptable, but the shapes do not match with the theoretical curve. Particularly, the numerical viscosity decreases at the last wave numbers, whereas turbulence theory requires an increase of the eddy viscosity. We also notice that the values of the spectral viscosity are smaller than expected at low wave numbers. However, this ought to be less important for the scheme's ability to model SGS effects.

The magnitude of the numerical viscosity computed for the scheme with kernel M_4 is huge compared to the theoretical values, see Figure B.3. However its global shape is appealing, because the viscosity increases at the last wave numbers.

At this point, we decide to try to built an implicit SGS model by using a linear combination of the interpolation functions M_4' and Λ_3, potentially corrected at large wave-numbers by a small contribution of M_4

$$K = c_1 M_4' + c_2 \Lambda_3 + c_3 M_4 \ . \tag{B.22}$$

Consistency requires $\sum_i c_i = 1$, i.e. two parameters, c_2 and c_3, are available for SGS modeling and $c_1 = 1 - c_2 - c_3$.

Figure B.4 shows the numerical spectral viscosity evaluated for several combinations of M_4' and Λ_3, without M_4. The result obtained with the combination $c = (0.7, 0.3, 0.0)$ is not satisfactory, because the numerical viscosity becomes negative at the last wave number and $c = (0.5, 0.5, 0.0)$ is too dissipative. Hence, the optimum value for c_2 is bounded by $0.3 < c_2 < 0.5$. The combination $c = (0.6, 0.4, 0.0)$ gives good results in the range of normalized wave-numbers $\xi^+ = 0.3...0.8$. We thus decide to consider this as our best combination of M_4' and Λ_3.

Finally, we study the influence of M_4 on the spectral viscosity with the combination $(0.6-c_3, 0.4, c_3)$. As shown in Figure B.4, this additional parameter increases the average value of the spectral numerical viscosity substantially. However, it does not change the

shape in the last wave-numbers. This finding reflects the nonlinearity of the Navier-Stokes equations, but it is contrary to what was expected. Consequently, we use the parameters $c = (0.6, 0.4, 0.0)$ in the following, even if the compliance with EDQNM theory shows room for improvement.

We do not claim that $(0.6, 0.4, 0.0)$ is the optimum solution to the given problem. For the purpose of the present study, however, this set of parameters is sufficiently close to the optimum. In order to determine an optimum solution to a nonlinear problem, systematic (and time-consuming) optimization strategies, e.g. an evolutionary optimization algorithm as employed by Adams et al. [2], are required.

B.8 Numerical results

For an a posteriori validation of the implicit SGS model provided by the VIC method with the selected linear combination of interpolating kernels M_4' and Λ_3 we perform LES of a 2D Taylor-Green vortex, of a 3D Taylor-Green vortex, and of decaying isotropic turbulence.

All simulations presented in this section are carried out in a $(2\pi)^3$-periodic computational domain, discretized with a uniform cartesian grid of $32 \times 32 \times 32$ cells, and the same number of particles. For time advancement, we use an explicit fourth-order Runge-Kutta scheme.

B.8.1 Two-dimensional Taylor-Green vortex

The two-dimensional Taylor-Green vortex in a periodic domain has an analytical solution that reads

$$\begin{aligned} u_1 &= -\sqrt{a}\cos(kx_1)\sin(kx_2)\exp(-(2k)^2 t/Re) \\ u_2 &= \sqrt{a}\sin(kx_1)\cos(kx_2)\exp(-(2k)^2 t/Re) \\ u_3 &= 0 \end{aligned} \tag{B.23}$$

with u_1, u_2, u_3 being the components of the velocity vector. The parameters of our numerical simulation are $k = 1$, $Re = 1$, and $a = 0.0016$.

The analytical decay exponent of the kinetic energy is $-4t/Re$. Figure B.5 shows the line $y = -4t$ and the logarithm of the kinetic energy evaluated from the numerical simulation. It is observed that the numerical results are in excellent agreement with the theoretical prediction. The vortex is well-resolved by 32^3 particles. This test validates the correct implementation of the method.

B.8.2 Three-dimensional Taylor-Green vortex

The 3D viscous Taylor-Green Vortex (TGV) characterized by the initial data

$$\begin{aligned} u_1 &= 0 \\ u_2 &= \cos(kx_1)\sin(kx_2)\cos(kx_3) \\ u_3 &= -\sin(kx_1)\cos(kx_2)\sin(kx_3) \end{aligned} \tag{B.24}$$

has no analytical solution. It was subject to a large number of investigations since it was supposed that the initially laminar flow develops singularities at $Re \to \infty$. Numerically, no blow-up has been observed, but there is still no proof [33].

In order to assess the quality of LES using the VIC method, the characteristic growth and decay of the dissipation rate is compared with DNS data. The DNS of Brachet et al. [10] provides detailed reference data for the 3D Taylor-Green vortex. These spectral DNS exploit spatial symmetries of the TGV to reduce the effective computational cost by a factor of 8. It was therefore possible to resolve Reynolds numbers up to Re = 5000 on a grid of 864^3 modes. For the present LES, spatial symmetries are not imposed, i.e. the $(2\pi)^3$-periodic computational domain contains 8 counter-rotating vortices and $k = 1$ in eq. (B.24).

Using a grid of 32^3 cells, a fair comparison of LES with Brachet's DNS is possible for Reynolds numbers Re = 100, Re = 200, and Re = 400. Figure B.6 shows the dissipation rate for LES and DNS at Reynolds number Re = 400. LES and DNS compare reasonably well. Note that the resolution requirement of the DNS is two orders of magnitude higher than that of the LES. These results for the 3D Taylor-Green vortex confirm that the PIC method does not affect the growth of the instability modes of the initially laminar flow and does provide sufficient SGS dissipation to stabilize the simulation during the ensuing quasi-turbulent stage.

B.8.3 Comte-Bellot - Corrsin experiment

A different situation is encountered when the VIC method is applied to decaying grid-generated turbulence. The computations are initialized with energy spectrum and Reynolds numbers adapted to the wind-tunnel experiments of Comte-Bellot and Corrsin [14], denoted hereafter as CBC. Among other space-time correlations CBC provides streamwise energy spectra for grid-generated turbulence at three positions downstream of a mesh.

In the simulation this flow is modeled as decaying turbulence in a $(2\pi)^3$-periodic computational domain. Based on the Taylor hypothesis the temporal evolution in the simulation corresponds to a downstream evolution in the wind-tunnel experiment with the experimental mean-flow speed which is approximately constant. The initial velocity field is obtained by performing a sharp truncation in Fourier space of the 512^2 initial data from A. Wray [86]. The SGS model is verified by comparing our computational 3D energy spectra with those obtained by A. Wray with DNS.

In this test case a correct representation of the energy containing range of the spectrum is as important as the proper modeling of the subgrid stresses. The present VIC method rather meets these requirements, but the large-scale turbulence decays too slowly, whereas the method over-predicts the dissipation at the smallest resolved scales, see fig. B.7a. In order to know if our linear combination of M_4' and Λ_3 really improves the ability of the vortex method to act as a subgrid-scale model, we performed the same computations with the kernel M_4' alone, see fig. B.7b. We decided to compare the results with the linear combination of kernels precisely to the kernel M_4' because it is currently the more widely used interpolating kernel in vortex methods. It appears that

M_4' alone gives better results that the optimized linear combination of M_4' and Λ_3 : the large-scale turbulence decays faster, and the small scales are less dissipated than with the combination.

B.9 Conclusions

We have proposed a new approach to design interpolation kernels assigned to the specific use in the implicit large-eddy simulation (LES) of turbulent flows. In implicit LES the subgrid-scale model is implicitly contained within the discretization. An explicit computation of model terms therefore becomes unnecessary.

The numerical truncation error of the vortex-in-cell method is analyzed *a-posteriori* through the effective spectral numerical viscosity in simulations of three-dimensional isotropic turbulence. The interpolation kernels used for velocity-smoothing and remeshing were identified as the most relevant components affecting the shape of the spectral numerical viscosity.

Turbulence theories, such as EDQNM [52, 12], define strict requirements for the spectral eddy viscosity to be met by a SGS model. The inherent eddy viscosity of standard kernels is found to be insufficient. The dissipative properties of the investigated kernels differ a lot. Exploiting these differences, a linear combination of three kernels leads to a new kernel with improved numerical viscosity.

The improved kernel consumes no extra computational resources because the linearity of the discretization is preserved. However, this advantage can also be considered as a major drawback. Explicit SGS modeling has clearly demonstrated the benefits of dynamic, solution-adaptive strategies. With respect to implicit SGS modeling this has to be translated into a non-linear discretization.

Computational results confirm that the truncation error of the vortex-in-cell methods with optimized interpolation kernels can act as implicit SGS model. It is however not evident that the present approach really improves the SGS modeling properties of vortex-in-cell methods. More precisely, it appears that linear combinations of usual interpolating kernels are not able to modify enough the shape of the numerical viscosity to improve the SGS modeling effect of vortex methods.

Future interpolation kernels will incorporate more degrees of freedom allowing for an improved compliance of numerical viscosity and turbulence theory. Our results suggest that a solution-adaptive strategy, e.g. through local time-stepping or nonlinear weight functions, may be required.

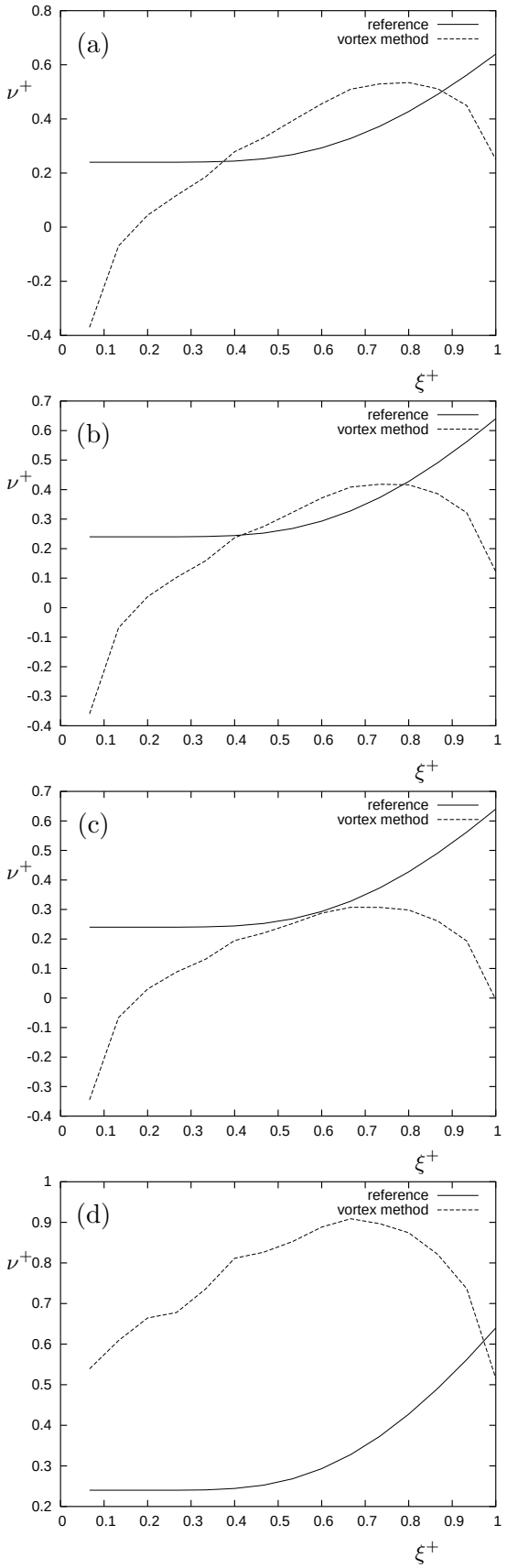


FIG. B.4 – Spectral numerical viscosity of the vortex method with a combination of M_4', Λ_3 and M_4 with respective coefficients (a) $c = (0.5, 0.5, 0.0)$, (b) $c = (0.6, 0.4, 0.0)$, (c) $c = (0.7, 0.3, 0.0)$, and (d) $c = (0.599, 0.4, 0.001)$.

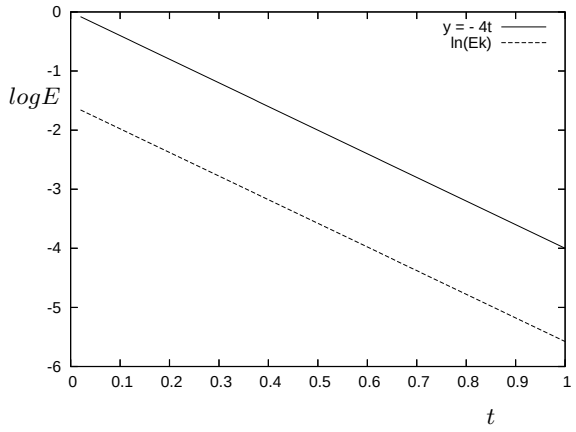


FIG. B.5 – Decay of the kinetic energy of the 2D Taylor-Green vortex.

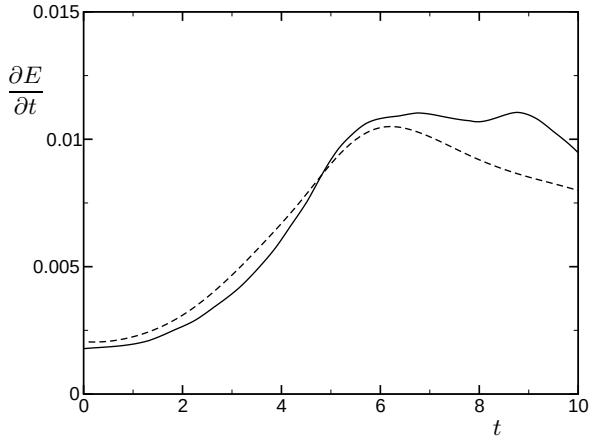


FIG. B.6 – Rate of kinetic-energy dissipation for LES of the 3D Taylor-Green vortex at Re = 400. —— DNS data from Ref. [10], −−−− implicit LES with VIC method.

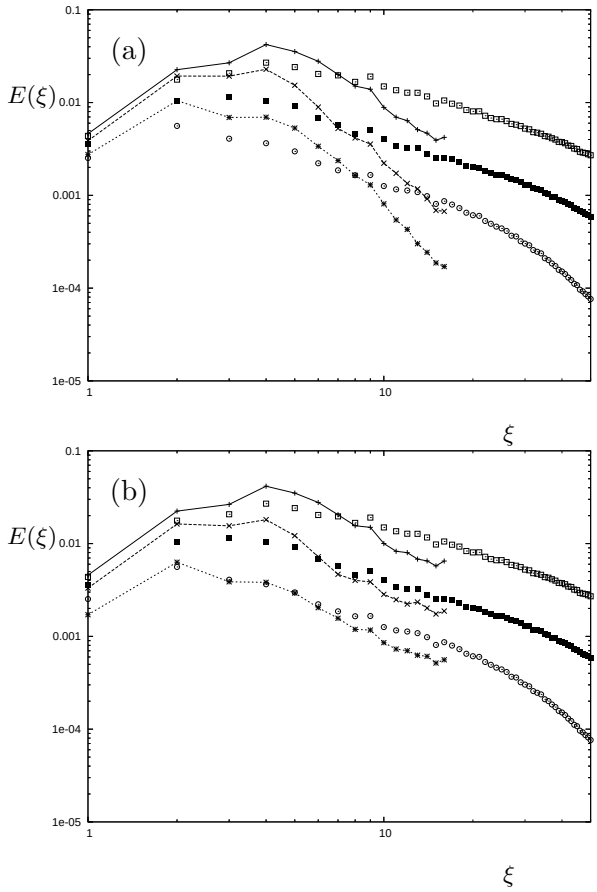


FIG. B.7 – Instantaneous 3D energy spectra for LES with the VIC method of the Comte-Bellot – Corrsin experiment, LES with 32^3 cells/particles, $dt = 0.03$. (a) Remeshing performed with a combination of M_4' and Λ_3 $(0.6, 0.4, 0.0)$. (b) Remeshing performed with a combination of M_4'. Crosses represent DNS data of A. Wray [86]. ($+$ $t' = 42$, $\times$ $t' = 98$ and $*$ $t' = 171$), symbols denote the present LES.

Annexe C

Particle methods revisited : a class of high order finite-difference methods

We propose a new analysis of particle method with remeshing. We derive a class of high-order finite difference methods. Our analysis is completed by numerical comparisons with Lax-Wendroff schemes for the Burger equation. *To cite this article : G.-H. Cottet, L. Weynans, C. R. Acad. Sci. Paris, Ser. I 340 (2006).*

Résumé

Les méthodes particulaires revisitées : une classe de schémas de différences finies d'ordre élevé. On propose dans cette note une nouvelle analyse des méthodes particulaires utilisant des remaillages. Cette analyse fait apparaître une classe de schémas de différences finies d'ordre élevé. Des illustrations numériques viennent compléter cette analyse et permettent de comparer les schémas particulaires avec le schéma de Lax-Wendroff. *Pour citer cet article : G.-H. Cottet, L. Weynans, C. R. Acad. Sci. Paris, Ser. I 340 (2006).*

Version française abrégée

Méthodes particulaires avec remaillage Les méthodes particulaires (C.2),(C.3) sont des méthodes lagrangiennes bien adaptées aux problèmes d'advection de la forme (C.1). Lorsque le flot subit de fortes distortions il est souvent conseillé de remailler les particules sur une grille régulière. On considère en général que les remaillages successifs introduisent une erreur de troncature supplémentaire dans l'analyse numérique de ces méthodes. Dans cette note, nous étudions directement, dans le cas scalaire, les méthodes particulaires avec remaillage à chaque pas de temps. Ceci nous conduit à la construction d'une classe de schémas de différences finies d'ordre élevé que nous analysons dans le cas scalaire.

Le cas linéaire - Nous considérons d'abord le cas d'une équation d'advection linéaire.

Nous nous intéressons aux formules de remaillage construites à l'aide de noyaux d'interpolation (C.5) conservant les 3 ou 4 premiers moments de la distribution de particules. Nous obtenons selon le cas un schéma de Lax-Wendroff, ou un schéma décentré d'ordre 3.

Le cas non-linéaire - Pour étendre ces résultats au cas non-linéaire, nous sommes conduits à introduire une méthode originale d'advection des particules, basée sur une évaluation de vitesses utilisant la formule (C.10), ou sa forme discrète (C.13). Nous pouvons alors donner une équation équivalente (C.12) pour le schéma de différences finies associé (C.11), qui montre en particulier que, pour la formule de remaillage utilisant le noyau Λ_2, la méthode particulaire est d'ordre 2. On montre aussi qu'avec un terme de viscosité artificielle du même type que pour le schéma de Lax-Wendroff l'énergie est une fonction décroissante du temps.

Illustrations numériques - Nous comparons les résultats obtenus par une méthode particulaire et par le schéma de Lax-Wendroff sur l'équation de Burgers (figure 1). Les deux modèles, complétés par les mêmes termes de viscosité artificielle, donnent des résultats semblables. Cependant,la méthode particulaire sans terme de viscosité, au contraire du schéma de Lax-Wendroff, semble converger vers la solution entropique, ce que notre analyse ne permet pas de prévoir. Par ailleurs, de meilleurs résultats pour la méthode particulaire sont obtenus lorsque le pas de temps est adapté au taux de déformation, ce qui illustre une autre différence essentielle avec les schémas de différences finies.

Conclusion - Une nouvelle analyse de méthodes particulaires avec remaillage met en évidence les liens existant entre ces méthodes et des méthodes de différences finies d'ordre élevé. Les méthodes particulaires offrent cependant l'avantage de traiter naturellement des systèmes et le décentrage. L'absence de condition CFL reste aussi une différence notable entre méthodes particulaires et méthodes de différences finies. Cette analyse doit être complétée, notamment pour évaluer les termes de viscosité artificielle optimaux pour ces méthodes.

C.1 Introduction

Particle methods are Lagrangian techniques that have been designed for advection-dominated physical problems. Among the features that are generally acknowledged for these methodes are the lack of numerical dissipation and the robustness due to the absence of Courant type stability condition. A well recognized drawback is the potential accuracy deterioration resulting form high distortion in the flow. A commonly used remedy to this problem is the periodic remeshing of particles in a way that conserves as much as possible the physical invariants of the flow. Truncation errors related to remeshing can be measured on the basis of the conservation of these invariants [17]. In practical implementation of the method times-scales which controls the particle advection schemes and on which particle remeshing is done are of the same order, which may lead to remeshing particles at every time-step. In that case, it may be

advisable to revisit the truncation error analysis. This is in particular true if one is interested in subgrid behavior of particle methods for turbulent flows or entropy balance in particle methods. The goal of this work is to start this analysis in the case of particle methods with remeshing at every time-step. We rewrite particle methods as finite-difference methods. We investigate two remeshing kernels which are commonly used in the literature. In the linear case we show remeshing with these kernels lead to Lax-Wendroff like schemes[1]. In the non-linear case, this interpretation leads us to propose new RK2 particle advancing schemes. With this time-stepping, we write an equivalent equation to leading order of the method, which resembles that of Lax-Wendroff scheme. For Burger's equation we are able to derive an energy balance which allows to propose artificial viscosity schemes that enforce energy control. We finally show some numerical experimental comparisons with the Lax-Wendroff scheme to complete our numerical analysis. These preliminary results essentially focus on the 1D case ; extensions to gas dynamics are given in [21] and in ongoing work.

C.2 Remeshed particle methods

Let us consider the model non-linear scalar equation, describing the evolution of the quantity u carried by the flow at the material velocity $g(u)$:

$$u_t + (g(u)u)_x = 0 \qquad (1) \tag{C.1}$$

Particle methods consist of sampling u on particles advected with velocity $g(u)$ and constant strength :

$$u(x) \simeq \sum_p \alpha_p \delta(x - x_p) \ , \ \dot{x}_p = g(u_p) \tag{C.2}$$

The strength of particles combines local volumes v_p and local u values u_p : $\alpha_p = v_p u_p$. Note that, while particle strengths are constant, volumes and local values evolve according to

$$\dot{v}_p = (\partial g(u)/\partial x)(x_p) v_p \ , \ \dot{u}_p = -(\partial g(u)/\partial x)(x_p) u_p \tag{C.3}$$

In purely lagrangian particle methods, velocities and their derivatives are computed by smoothing particle strength over a space scale containing a *few* particles. The smoothing range must adapt to the flow conditions to smooth out irregular motions [35]. In remeshed particle methods, every *few* time-steps particles are remeshed on a predefined regular grid. Here we consider the case of a uniform grid. Remeshing is done by interpolating particle strength with a kernel Λ with compact support. If $\sum_p \alpha_p \delta(x - x_p)$ and $\sum_q \tilde{\alpha}_q \delta(x - x_q)$ are respectively the original and remeshed particle distribution, with $\tilde{x}_q = qh$ on a regular grid of grid-size h, remeshing formulas read

$$\alpha_q = \sum_p \alpha_p \Lambda(\frac{x_p - \tilde{x}_q}{h}) \tag{C.4}$$

Conservation of successive moments $\sum \tilde{\alpha}_q = \sum \alpha_p$, $\sum \tilde{x}_q \tilde{\alpha}_q = \sum x_p \alpha_p$, $\sum \tilde{x}_q^2 \tilde{\alpha}_q = \sum x_p^2 \alpha_p \cdots$ can be enforced by using kernels extending to an increasing number of

[1] as we were completing this paper, the reference [23] was brought to our attention. In this reference the authors use the Lax-Wendroff analysis of particle methods to propose modified remeshing kernels

points. The following formulas respectively guarantee conservation of the 3 and 4 first moments and are widely used in CFD :

$$\Lambda_2(x) = \begin{cases} 1-x^2 & \text{if } |x| \le 0.5 \\ (1-|x|)(2-|x|)/2 & \text{if } 0.5 < |x| \le 1.5 \\ 0 & \text{if } |x| \ge 1.5 \end{cases}, \; \Lambda_3(x) = \begin{cases} (1-|x|^2)(2-|x|)/2 & \text{if } |x| \le 1 \\ (1-|x|)(2-|x|)(3-|x|)/6 & \text{if } 1 < |x| \le 2 \\ 0 & \text{if } |x| \ge 2. \end{cases} \tag{C.5}$$

C.3 The linear case : $g(u) = a$

We denote respectively by Δt and h the time-step and grid-size. Let us first consider the above 3-points remeshing formula Λ_2, and assume that $\lambda = a\Delta t/h \le 1/2$. In this case, a particle initially at grid point number p with $x_p = ph$ will after one time-step be remeshed onto 3 particles located at grid points p, $p-1$ and $p+1$. If we denote by $\alpha_p^n = h\,u_p^n$ the particle strength at grid-point p and time $t_n = n\Delta t$, we can write

$$u_p^{n+1} = c_0 u_p^n + c_{+1} u_{p+1}^n + c_{-1} u_{p-1}^n \tag{C.6}$$

The weights c_i determined from the conservation of the 3 first moments must satisfy the following relations :

$$c_{-1} + c_0 + c_{+1} = 1 \; , \; c_{-1} - c_{+1} = \lambda \; , \; c_{-1} + c_{+1} = \lambda^2 \tag{C.7}$$

from which we readily get $c_0 = 1-\lambda^2, c_{\pm 1} = \mp\lambda(1 \mp \lambda)/2$. This is the Lax-Wendroff scheme. If $1 \ge \lambda > 1/2$ one obtains an upwind version of the Lax-Wendroff scheme. Finally if $\lambda > 1$ and if we set $m = [\lambda]$ and $\tilde{t} = mh/a$, the particle method can be interpreted as solving exactly the advection equation from t_n to $t_n + \tilde{t}$ then using one of the above finite-difference schemes from $\tilde{t}$ to t_{n+1}. This reflects the well-known fact that particle methods are not constrained by CFL conditions.

If one now considers the case when we use the function Λ_3 to remesh particles, under the CFL condition $\lambda \le 1$ we obtain a 4-points upwind finite-difference scheme :

$$u_p^{n+1} = c_0 u_p^n + c_{+1} u_{p+1}^n + c_{-1} u_{p-1}^n + c_{-2} u_{p-2}^n \tag{C.8}$$

Conservation of the 4 first moments gives the relations

$$c_{-2}+c_{-1}+c_0+c_{+1} = 1 \; , \; 2\,c_{-2}+c_{-1}-c_{+1} = \lambda \; , \; 4\,c_{-2}+c_{-1}+c_{+1} = \lambda^2 \; , \; 8\,c_{-2}+c_{-1}+c_{+1} = \lambda^3 \tag{C.9}$$

The Fourier analysis of this scheme shows that it is third order accurate : if we denote by $\hat{u}_k^n$ the $k-$th fourier mode of the scheme at time t_n we have

$$\hat{u}_k^{n+1} = \hat{u}_k^n [c_0 + c_{-2}\exp(-2ikh) + c_{-1}\exp(-ikh) + c_{+1}\exp(ikh)]$$

and a Taylor expansion in $K = kh$ gives, to third order,

$$\begin{aligned} \hat{u}_k^{n+1} &= \hat{u}_k^n \big[c_{-2} + c_{-1} + c_0 + c_{+1} + (-2c_{-2} - c_{-1} + c_{+1})iK - (4c_{-2} + c_{-1} + c_{+1})K^2/2 \\ &+ (8c_{-2} + c_{-1} - c_{+1})iK^3/6 + O(K^4)\big] = \hat{u}_k^n \left[1 - \lambda iK - \lambda^2 K^2/2 + \lambda^3 K^3/6 + O(K^4)\right]. \end{aligned}$$

This is, to third order, the Taylor expansion of the exact solution of the advection equation. This scheme is therefore third order accurate. More generally it is not difficult to see that a particle method with a remeshing formula conserving the p first moments is equivalent to a finite-difference scheme of order p.

C.4 The non-linear case

Here we restrict the discussion to the Λ_2 remeshing formula and we assume that the CFL condition $\Delta t|\max_p g(u_p)| \leq 1/2$ is satisfied. To obtain a second order method in the non-linear case it is clearly necessary to use a second-order time-stepping method to advance particles. In classical implementations of particle methods, particles are moved with *e.g.* Runge-Kutta methods. In a mid-point second order RK time-stepping, particles would be moved for half a time-step, and velocities would be computed at these locations by interpolation and combined with velocities at the beginning of the time-step. Here we propose a different method which is both numerically more efficient and analytically more tractable. To evaluate particle velocity at time $t_n + \Delta t/2$, we use (C.3) and write

$$u_p^{n+1/2} = u_p^n \left(1 - \Delta t\, g(u)_x(x_p)/2\right). \tag{C.10}$$

The finite-difference formulas corresponding to this method can be derived along the same lines as in the linear case :

$$u_p^{n+1} = u_p^n - \frac{\Delta t}{2h}(\tilde{g}_{p+1}^n u_{p+1}^n - \tilde{g}_{p-1}^n u_{p-1}^n) + \frac{\Delta t^2}{2h^2}\left((\tilde{g}_{p+1}^n)^2 u_{p+1}^n - 2(\tilde{g}_p^n)^2 u_p^n + (\tilde{g}_{p-1}^n)^2 u_{p-1}^n\right) \tag{C.11}$$

where we have set $\tilde{g}_p^n = g(u_p^{n+1/2})$.

Proposition 16 *The scheme (C.10),(C.11) is second order accurate for smooth solutions. If we set $\mu = \Delta t/h$, its equivalent equation to second order is*

$$u_t + (g(u)u)_x + h^2\left[\frac{1}{6}(g(u)u)_{xxx} + \frac{\mu^2}{6}u_{ttt} + + \frac{\mu^2}{8}(g(u)_x^2 u^3 g''(u))_x + \frac{\mu^2}{2}\left(g(u)g'(u)u^2 g(u)_x\right)_{xx}\right] = 0 \tag{C.12}$$

The proof classically relies on Taylor expansions the details of which will be given elsewhere. In practice the calculation of $u_p^{n+1/2}$ is done by finite-difference approximations of the right hand side of (C.10). We obtain a fully discrete scheme if we replace formula (C.10) by

$$u_p^{n+1/2} = u_p^n \left[1 - \frac{\mu}{4}\left(g(u_{p+1}) - g(u_{p-1})\right)\right]. \tag{C.13}$$

Proposition 17 *The scheme (C.13),(C.11) is second order accurate for smooth solutions. Its equivalent equation to second order is (C.12). Moreover, for the Burger's equation ($g(u) = u/2$) this scheme is energy decreasing when supplemented with an artificial viscosity term of the form $C\mu\Delta_-\left(|\Delta_+ u_j|\Delta_+ u_j\right)$ for $C = 0.48$.*

In the above result, we have used the usual notations : $\Delta_\pm f_j = \pm(f_{j\pm1} - f_j)$. The first part of our assertion only results from the fact that the truncation error from the centered approximation of the derivatives of g only adds third order error terms that do not modify the equivalent equation. Our second assertion comes from lengthy calculations in the spirit of [57]. As a matter of fact the equivalent equations (C.12) and that of Lax-Wendroff are very similar. However, the numerical results that we show below suggest that their behavior are rather different and that the form of the

artificial viscosity proposed in Proposition 2 is not optimal. Note that similar results can be obtained when the remeshing formula Λ_3 is used. The resulting upwind 4-points finite-difference scheme is second order in time and third order in space.

To finish this section, let us comment on the computational complexity of the scheme (C.13),(C.11) for multidimensional problems. If u is a scalar or vector quantity and $\mathbf{G}(u)$ is the advection velocity field, to compute $u_p^{n+1/2}$ one must evaluate the divergence of $\mathbf{G}(u)$. This can be easily done by a centered formula using $2d$ points. Although strictly speaking based on a 5 points stencil, the scheme complexity is thus of order $2d + 3^d$, to be compared respectively to 3^d and 5^d for 3-points and 5-points schemes.

C.5 Numerical illustrations

In this section we complement our numerical analysis with experimental comparisons with the Lax-Wendroff scheme. We considered the Burger's equation with initial condition $u_0(x) = 0$ if $x < 0.5$ and $u_0(x) = 1$ if $x \geq 0.5$. This initial profile produces a shock and an expansion wave that eventually merge together. The top row of figure 1 shows results for the Lax-Wendroff scheme and the particle method with Λ_2 remeshing at two subsequent times. Both calculations used $h = 0.01$, $\mu = 0.8$ and an artificial viscosity coefficient 0.3 produced similar results. The bottom row illustrates the main differences between the two schemes. The left picture shows results without artificial viscosity. These calculations suggest that, unlike the Lax-Wendroff schemes, the particle method, at least in this case, does converge towards the entropy solution, something that the present analysis is not able to predict. The right picture illustrates the fact that particle method are not constrained by CFL conditions. In this calculation, the time step was locally (in space and time) adapted to the local strain in the flow to prevent particles to cross. The corresponding μ varied from 0.07 to 2.5. The artificial coefficient was 0.5. This calculation actually seems to give the best results for the particle method. In all these figures, the solid lines (resp. dotted lines) correspond to the exat (resp. computed) solutions.

C.6 Conclusion

We have presented a new analysis of particle method with remeshing. We have shown analogies of these method with a class high order finite-difference methods, with however definite advantages for particle methods : natural handling of up-winding through particle advection, simple extensions to system and added flexibility in the choice of time-steps. Further works are needed to continue this analysis and in particular to evaluate the optimal artificial viscosity terms to implement in the particle schemes.

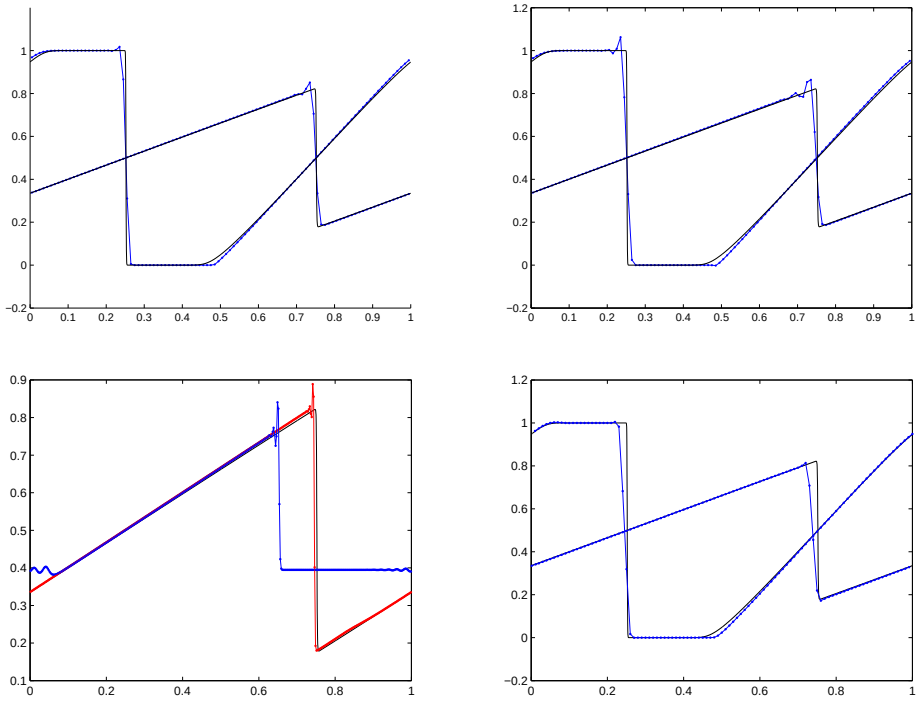


FIG. C.1 – Comparison of Lax-Wendroff (LW) and particle schemes. Top row : LW (left) and particle (right) schemes with $h = 0.01$, artificial viscosity coefficient 0.2 and $\mu = 0.84$. Bottom left : LW (blue curve) and particle (red curve) schemes without artificial viscosity at $t = 1.5$ and $h = 0.0025$. Bottom right, particle method with adaptive time-step and $h = 0.01$.

Bibliographie

[1] R. Abgrall, J. Breil, P.-H. Maire, and J. Ovadia. Un schéma centré pour l'hydrodynamique lagrange bidimensionnelle, 2004.

[2] N. A. Adams, S. Hickel, and S. Franz. Implicit subgrid-scale modeling by adaptive deconvolution. *J. Comput. Phys.*, 200 :412–431, 2004.

[3] O. Agertz, B. Moore, J. Stadel, D. Potter, J. Read, L. Mayer, P. Chatelain, G.-H. Cottet, A. Gawryszczak, A. Kravstov, F. Miniati, J. Monaghan, A. Nordlund, F. Pearce, V. Quilis, D. Rudd, V. Springel, J. Stone, E. Tasker, R. Teyssier, J. Wadsley, and R. Walder. Fondamental differences between sph and grid methods, mon. not. r. astron. soc., soumis.

[4] D. J. Benson. Computational methods in lagrangian and eulerian hydrocodes. *Computer Methods in Applied Mechanics and Engineering*, 99 :235–394, 1992.

[5] W. Benz. *The Numerical Modelling of Nonlinear Stellar Pulsations, Problems and Prospects*, chapter Smooth Particle Hydrodynamics : A Review, pages 269–287. NATO ASIS Series, 1989.

[6] M. Bergdorf, G.-H. Cottet, and P. Koumoutsakos. Multilevel adaptive particle methods for convection-diffusion equations. *SIAM Multiscale Modeling and Simulation*, 4 :328–357, 2005.

[7] M. J. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comput. Phys.*, 53(3) :484–512, 1984.

[8] C. Berthon. *Contribution à l'analyse numérique des équations de Navier-Stokes compressibles à deux entropies spécifiques. Application à la turbulence compressible.* PhD thesis, Université Paris VI, 1998.

[9] R. L. Bowers and J. R. Wilson. *Numerical modeling in applied physics and astrophysics.* Jones and Bartlett Publishers, Boston, 1991.

[10] M. E. Brachet, D. I. Meiron, S. A. Orszag, B. G. Nickel, R. H. Morf, and U. Frisch. Small-scale structure of the taylor-green vortex. *J. Fluid. Mech.*, 130 :411–452, 1983.

[11] S.-H. Cha and A. P. Withworth. Implementations and tests of godunov-type particle hydrodynamics. *Mon. Not. R. Astron. Soc.*, 340(1) :73–90, 2002.

[12] J.-P. Chollet. Two-point closures as a subgrid-scale modeling tool for large-eddy simulations. In F. Durst and B. E. Launder, editors, *Turbulent Shear Flows IV*, pages 62–72. Springer, 1984.

[13] J. P. Christiansen. Numerical simulation of hydrodynamics by the method of point vortices. *J. Comput. Phys.*, 13 :363–379, 1973.

[14] G. Comte-Bellot and S. Corrsin. Simple eulerian time correlation of full and narrow band velocity signals in grid generated 'isotropic' turbulence. *J. Fluid. Mech.*, 48 :273–337, 1971.

[15] G.-H. Cottet. Artificial viscosity models for vortex and particle methods. *J. Comput. Phys.*, 127 :299–308, 1996.

[16] G.-H. Cottet. Multi-physics and particle methods. In *Computational Fluid and Solid Mechanics*, pages 1296–8. Elsevier, 2003.

[17] G.-H. Cottet and P. D. Koumoutsakos. *Vortex methods*. Cambridge University Press, 2000.

[18] G.-H. Cottet, P. D. Koumoutsakos, and M. L. Ould Salihi. Vortex methods with spatially varying cores. *J. Comput. Phys.*, 162(1) :164–185, 2000.

[19] G.-H. Cottet and E. Maitre. A level set method for fluid-structure interactions with immersed surfaces. *Mathematical Models and Methods in the Applied Sciences*, 16 :415–438, 2006.

[20] G.-H. Cottet, B. Michaux, S. Ossia, and G. Vanderlinden. A comparison of spectral and vortex methods in three-dimensional incompressible flow. *J. Comput. Phys.*, 175 :702–712, 2002.

[21] G.-H. Cottet, B. Rebourcet, and L. Weynans. A multilevel adaptive particle-grid method for gas dynamics, 2005.

[22] G.-H. Cottet and L. Weynans. Particle methods revisited : a class of high-order finite-difference schemes. *C. R. Acad. Sci. Paris, Ser. I*, 343 :51–56, 2006.

[23] A. Ghoniem D. Wee. Modified interpolation kernels for treating diffusion and remeshing in vortex methods. *J. Comput. Phys.*, 213 :239–263, 2006.

[24] Bruno Desprès and Constant Mazeran. Lagrangian gas dynamics in two dimensions and lagrangian systems. *Arch. Ration. Mech. Anal.*, 178(3) :327–372, 2005.

[25] B. Desprès. Inégalités entropiques pour un solveur de type lagrange+convection des équations de l'hydrodynamique. Technical report, CEA, 1997.

[26] B. Després and F. Lagoutière. Numerical resolution of a two-component compressible fluid model with interfaces.

[27] J. A. Domaradzki and D. D. Holm. Navier-stokes-alpha model : Les equations with nonlinear dispersion. In B. Geurst, editor, *Modern Simulation Stategies for Turbulent Flow*, pages 107–122. R. T. Edwards, 2001.

[28] John K. Dukowicz. A general, non-iterative riemann solver for godunov's method.

[29] B. Einfeldt, C. D. Munz, P. L. Roe, and B. Sjögreen. On godunov type methods near low densities. *J. Comp. Phys.*, 92 :273–.., 1991.

[30] D. Enright, R. Fedkiw, J. Ferziger, and I. Mitchell. A hybrid particle level set method for improved interface capturing. *J. Comp. Phys.*, 183 :83–116, 2002.

[31] D. Enright, F. Losasso, and R. Fedkiw. A fast and accurate semi-lagrangian particle level set method, 2004.

[32] M. W. Evans and F. H. Harlow. The particle-in-cell method for hydrodynamics calculations. Technical report, Los Alamos Scientific Laboratory, 1956.

[33] U. Frisch. *Turbulence*. Cambridge University Press, 1995.

[34] S. Ghosal. An analysis of numerical errors in large-eddy simulations of turbulence. *J. Comput. Phys.*, 125 :187–206, 1996.

[35] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics : theory and application to non-spherical stars. *Mon. Not. R. astr. Soc.*, 181 :375–389, 1977.

[36] E. Godlewski and P.-A. Raviart. *Hyperbolic systems of conservation laws*. Collection Mathématiques et Applications de la SMAI, Ellipse, Paris, 1990.

[37] J.-F. Haas and B. Sturtevant. Interaction of weak shock waves with cylindrical and spherical gas inhomogeneities. *J. Fluid Mech.*, 181 :41–76, 1987.

[38] F. H. Harlow. Hydrodynamic problems involving large fluid distorsion. *J. Assoc. Comp. Mach.*, 4 :137–.., 1957.

[39] W. Heisenberg. Zur statistischem theorie der turbulenz. *Z. Phys.*, 124 :628–657, 1946.

[40] S. Hickel, N. A. Adams, and J. A. Domaradzki. An adaptive local deconvolution method for implicit les. *J. Comput. Phys.*, 213 :413–436, 2006.

[41] S. Hickel, T. Kempe, and N. A. Adams. On implicit subgrid-scale modeling in wall-bounded flows. In *Proceedings of the EUROMECH Colloquium 469*, pages 36–37, Dresden, Germany, 2005.

[42] T. Hou and P. G. Lefloch. Why non-conservative schemes converge to wrong solutions : error analysis. *Mathematics of Computation*, 62 :497–530, 1994.

[43] T. Y. Hou. Convergence of a variable blob vortex method for the euler and navier-stokes equations. *SIAM J. Numer. Anal.*, 27(6) :1387–1404, 1990.

[44] J. M. Hyman. Numerical methodologies for solving partial differential equations. In *The Numerical Modelling of Nonlinear Stellar Pulsations*, pages 215–237. Kluwer Academic Publishers, 1990.

[45] P. Koumoutsakos. *Direct Numerical Simulations of Unsteady Separated Flows Using Vortex Methods*. PhD thesis, California Institute of Technology, 1993.

[46] P. Koumoutsakos. Inviscid axisymmetrization of an elliptic vortex. *J. Comput. Phys.*, 138(2) :821–857, 1997.

[47] P. Koumoutsakos. Remeshed smoothed particle hydrodynamics for the simulation of viscous and heat conducting flows. *J. Comput. Phys.*, 182 :67–90, 2002.

[48] P. Koumoutsakos and S. Hieber. A lagrangian particle level set method. *J. Comp. Phys.*, 210 :342–367, 2005.

[49] A. Kravchenko and P. Moin. On the effect of numerical errors in large-eddy simulation of turbulent flows. *J. Comput. Phys.*, 130 :310–322, 1997.

[50] R. Landshoff. A numerical method for treating fluid flow in the presence of shocks. Technical report, Los Alamos Scientific Laboratory Report LA-1930, 1955.

[51] A. Leonard. Energy cascade in large eddy simulations of turbulent fluid flows. *Adv. Geophys.*, 18A :237–248, 1974.

[52] M. Lesieur. *Turbulence in Fluids*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 3 edition, 1997.

[53] R. J. Leveque. *Finite-volume methods for hyperbolic problems*. Cambridge University Press, 2002.

[54] R. Liska, M. Shashkov, and B. Wendroff. Lagrangian composite schemes on triangular unstructured grids.

[55] R. Loubère. *Une Méthode Particulaire Lagrangienne de type Galerkin Discontinu, Application à la Mécanique des Fluides et l'Interaction Laser/Plasma*. PhD thesis, Université Bordeaux 1, Ecole Doctorale de Mathématiques et Informatique, 2002.

[56] R. Loubère, J. Ovadia, and R. Abgrall. A lagrangian discontinuous galerkin type method on unstructured meshes to solve hydrodynamics problems, 2000.

[57] Majda and S. Osher. A systematic approach for correcting nonlinear instability : the lax-wendroff scheme for scalar conservation laws. *Num. Math.*, 30 :429–452, 1978.

[58] A. Majda and S. Osher. Numerical viscosity and the entropy condition. *SIAM J. Numer. Anal.*, 32 :797–838, 1979.

[59] L. Margolin and William Rider. *Implicit Large Eddy Simulation*. Cambridge University Press, 2006.

[60] A. Marquina and P. Mulet. A flux-split algorithm applied to conservative models for multicomponent compressible flows. *J. Comput. Phys.*, 185 :120–138, 2003.

[61] J. J. Monaghan. Why particle methods work. *SIAM J. Sci. Stat. Comput*, 3(4) :422–433, 1982.

[62] J. J. Monaghan. Extrapolating b-splines for interpolation. *J. Comput. Phys.*, 60(2) :253–262, 1985.

[63] J. J. Monaghan. Smoothed particle hydrodynamics. *Annu. Rev. Astron. Astrophys.*, 30 :543–.., 1992.

[64] J. J. Monaghan. Sph and riemann solvers. *J. Comput. Phys.*, 136(2) :298–307, 1997.

[65] W. Mulder, S. Osher, and J. Sethian. Computing interface motion in compressible gas dynamics. *J. Comput. Phys.*, 100 :209–228, 1992.

[66] J. Von Neumann and R. D. Richtmeyer. *J. Appl. Phys.*, 21 :232–.., 1950.

[67] W. F. Noh. Errors for calculations of strong shocks using artificial viscosity and an artificial heat flux. *J. Comput. Phys.*, 72 :78–120, 1987.

[68] M. Ould-Salihi, G.-H. Cottet, and M. El Hamraoui. Blending finite-difference and vortex methods for incompressible flow calculations. *SIAM J. Sci. Comp.*, 22 :1655–1674, 2000.

[69] M. Ould-Salihi, G.-H. Cottet, and M. El Hamraoui. Blending finite-difference and vortex methods for incompressible flow computations. *SIAM J. Sci. Comp.*, 22 :1655–1674, 2000.

[70] P. Ploumhans and G. S. Winckelmans. Erratum : Vortex methods for high-resolution simulations of viscous flow past bluff bodies of general geometry [j. comput. phys. 165(2000), no. 2, 354-406 ; mr 2001]. *J. Comput. Phys.*, 170(1) :449–456, 2001.

[71] P. Ploumhans, G. S. Winckelmans, J. K. Salmon, A. Leonard, and M. S. Warren. Vortex methods for direct numerical simulation of three-dimensional bluff body flows : application to the sphere at re = 300, 500, and 1000. *J. Comput. Phys.*, 178(2) :427–463, 2002.

[72] P. Poncet. *Méthodes particulaires pour la simulation des sillages tridimensionnels*. PhD thesis, Université Joseph Fourier, 2001.

[73] P. Poncet. Topological aspects of the three-dimensional wake behind rotary oscillating circular cylinder. *J. Fluid Mech.*, 517 :27–53, 2004.

[74] J. Quirk and S. Karni. On the dynamics of a shock-bubble interaction. *J. Fluid Mech.*, 318 :129–163, 1996.

[75] B. Rebourcet. Une introduction à l'intégration numérique de certains systèmes de l'hydrodynamique, cours de master.

[76] P. J. Roache. Quantification of uncertainty in computational fluid dynamics. *Ann. Rev. Fluid Mech.*, 29 :123–160, 1997.

[77] B. Scheurer. *Quelques schémas numériques pour l'hydrodynamique lagrangienne*. Rapport CEA, 2000.

[78] I. J. Schoenberg. Cardinal spline interpolation, 1973.

[79] U. Schumann. Subgrid scale model for finite-difference simulations of turbulence in place channels and annuli. *J. Comput. Phys.*, 18 :376–404, 1975.

[80] D. Shiels. *Simulation of controlled bluff body flow with a viscous vortex method*. PhD thesis, Caltech, 1998.

[81] G. A. Sod. A survey of several finite difference methods for systems of non-linear hyperbolic conservation laws. *J. Comput. Phys.*, 27 :1–131, 1978.

[82] M. Sun and K. Takayama. An artificially upstream flux vector splitting scheme for the euler equations. *J. Comput. Phys.*, 189 :305–329, 2003.

[83] M. Sussman and E. Puckett. A coupled level set and volume of fluid method for computing 3d and axisymmetric incompressible two-phase flows. *J. Comp. Phys.*, 162 :301–337, 2000.

[84] E. F. Toro. *Riemann Solvers and Numerical Methods for fluid dynamics*. Springer-Verlag, second edition, 1999.

[85] M. L. Wilkins. Use of artificial viscosity in multidimensional fluid dynamic calculations. *J. Comput. Phys.*, 36 :281–303, 1980.

[86] A. Wray. A selection of test cases for the validation of large-eddy simulations of turbulent flows. Technical report, AGARD Advisory Report 345, 1998.

[87] K. Xu, L. Martinelli, and A. Jameson. Gas-kinetic finite volume methods, flux-vector splitting, and artificial diffusion. *J. Comp. Phys.*, 120 :48–65, 1995.

Table des figures

1 7

2 Instabilité de Kelvin-Helmholtz, isovaleurs de ϕ, à $t = 8$, équations d'Euler résolues avec 50^2 particules, de haut en bas ϕ résolu sur une grille 50^2, 100^2 et 200^2. 13

1.1 Remaillage 2D : la zone hachurée indique les cellules sur lesquelles la particule est redistribuée. 25

1.2 Noyaux d'interpolation, de haut en bas et de gauche à droite : Λ_2, Λ_3, M_4 et M'_4. 28

1.3 Mapping d'une grille polaire sur une grille cartésienne uniforme, où les particules sont interpolées. 30

2.1 Utilisation de la fonction d'interpolation décentrée Λ_J : les valeurs portées par la particule située sur la dernière cellule sont distribuées de manière décentrée, sur les trois dernières cellules de la grille. 57

2.2 Remaillage sur le bord du domaine avec une particule fictive. La particule fictive n'intervient pas dans la création de la nouvelle particule située sur la troisième cellule à partir du bord. Cette dernière reçoit les contributions de trois particules. 58

2.3 Remaillage sur le bord du domaine avec une fonction d'interpolation décentrée pour la dernière particule. La troisième cellule à partir du bord reçoit une contribution supplémentaire de la particule située sur la dernière cellule. 59

3.1 Grille cartésienne 1D. 64

3.2 Grille cartésienne 2D. 64

3.3 Grille polaire. 65

3.4 Transport à vitesse uniforme d'un sinus avec de haut en bas Λ_2, M'_4 et Λ_3, avec 40 points. 73

3.5 Advection d'une gaussienne, avec de haut en bas Λ_2, M'_4 et Λ_3, 40 points. 74

3.6 Transport à vitesse uniforme d'un créneau avec la fonction M'_4, N= 160, sans seuillage (gauche), et avec seuillage (droite). 77

3.7 Tube à choc de Sod, solution numérique obtenue avec la méthode particulaire utilisée avec le seuillage, comparée à la solution exacte. 78

3.8 De haut en bas et de droite à gauche : schéma de Lax-Wendroff et schémas avec les noyaux Λ_2, M_4' et Λ_3, en vert, comparés à la solution entropique, en rouge, à t = 0.5, 200 particules ou points de grille. 79
3.9 Même légende que la figure précédente, à $t = 1.5$. 80
3.10 De haut en bas et de droite à gauche : schéma de Lax-Wendroff et schémas avec les noyaux Λ_2, M_4' et Λ_3, corrigés par le terme de viscosité artificielle étudié en 2.2.6 avec un coefficient $C = 0.2$, en vert, comparés à la solution entropique, en rouge, à t = 0.5, 200 particules ou points de grille. 81
3.11 Même légende que la figure précédente, à $t = 1.5$. 82
3.12 Tube à chocs de Sod, N=100, Λ_2, différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 85
3.13 Tube à chocs de Sod, N=100 , Λ_2, solveur de Riemann approché, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 85
3.14 Tube à chocs de Sod, N=100, M_4', différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 86
3.15 Tube à chocs de Sod, N=100, M_4', solveur de Riemann approché, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 86
3.16 Tube à chocs de Sod, 100 particules, Λ_3, différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. . . . 87
3.17 Tube à chocs de Sod, 100 particules, Λ_3, solveur de Riemann approché, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 87
3.18 Résultats de Desprès et Mazeran [24] pour le tube à chocs de Sod, N =100. 88
3.19 Résultats de Loubère ([55]) pour le tube à chocs de Sod, N =100. 89
3.20 Résultats de Cha et Whitworth [11] pour le tube à chocs de Sod, N =250. 90
3.21 Test de Sjögreen, N = 100, M_4', différences finies centrées, de haut en bas et de gauche à droite : densité, vitesse, pression et énergie interne. . 92
3.22 Même légende que la figure précédente, avec N = 200. 92
3.23 Résultats de Monaghan ([64]) pour le test de Sjögreen, N =200, de haut en bas : densité, vitesse, énergie interne. 93
3.24 Résultats de Loubère ([55]) pour le test de Sjögreen, N =100. 94
3.25 Résultats de Cha et Whitworth [11] pour le test de Sjögreen , N =250. . 95
3.26 Résultats de Sun [82] pour le test de Sjögreen, N =100. 96
3.27 Problème de Noh, grille polaire, dr = 1/200, d$\theta = 2\pi/200$, M_4', différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 98
3.28 Problème de Noh, grille polaire, dr = 1/200, d$\theta = 2\pi/200$, Λ_3, différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 99

3.29 Problème de Noh, grille polaire, dr = 1/200, d$\theta = 2\pi/200$, M_4', solveur de Riemann, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 100

3.30 Problème de Noh, grille polaire, dr = 1/200, d$\theta = 2\pi/200$, Λ_3, solveur de Riemann, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 101

3.31 Résultats de Abgrall et al. [1] pour le problème de Noh, dr = 100, 200 et 400. 102

3.32 Problème de Noh, grille cartésienne, dx = dy = 1/200, M_4', dt = 0.0015, différences finies, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 103

3.33 Problème de Noh, grille cartésienne, dx = dy = 1/100, dt = 0.003, iso-valeurs de densité. 104

4.1 Grille et particules fines, grille et particules grossières. 109

4.2 La description de l'écoulement par les particules n'est plus consistante sur la première cellule de la grille fine. 109

4.3 Il manque de l'information à gauche de la grille fine pour créer une nouvelle particule au centre de la deuxième cellule de la grille. 110

4.4 Zone fiable de la grille fine, composée des cellules où le remaillage habituel s'effectue en toute confiance. 111

4.5 Des particules de l'autre grille permettent de remailler dans la zone non fiable de chaque grille. 112

4.6 Définition de la masse totale quand il y a deux grilles. 115

4.7 Indexation des cellules autour de la frontière entre les grilles. 115

4.8 Tube à chocs de Sod, deux grilles, t = 0.1, adaptation des grilles suivant le critère (4.1), différences finies centrées. De haut en bas : densité, vitesse, pression et énergie interne. Solution exacte en rouge, solution sur la grille grossière en bleu, et solution sur la grille fine en vert. Une particule sur deux affichée. 124

4.9 Même légende que la figure précédente, à t = 0.2. 125

4.10 Tube à chocs de Sod, bigrille adaptative, évolution du rapport masse sur masse initiale en fonction du temps 126

4.11 Problème de Noh, deux grilles, $dx1 = 1/50$, $dx2 = 1/400$, M_4', solveur de Riemann. De haut en bas et de gauche à droite : densité, vitesse, pression et énergie interne. Solution exacte en bleu, solution sur la grille grossière en rouge, et solution sur la grille fine en vert. 127

4.12 Equation de Bürgers, formation d'un choc, remaillage avec M_4', en haut le calcul est effectué avec une seule grille, en bas avec deux grilles qui possèdent des pas de temps différents. 128

5.1 Tube à chocs de Sod avec $\gamma = 1.4$ à gauche de la discontinuité initiale, et $\gamma = 2.4$ à droite, N = 200, remaillage avec le noyau Λ_3, différences finies centrées d'ordre 3, de gauche à droite et de haut en bas : densité, vitesse, pression, énergie interne. 137
5.2 Même légende que la figure précédente. Evolution du rapport $\frac{\text{masse}}{\text{masse initiale}}$ du fluide situé à droite, pour dx = 1/50 (rouge), 1/100 (vert), 1/200 (bleu), 1/400 (rose). 138
5.3 Même légende que la figure précédente. Densité (bleu) et ϕ (vert) obtenues avec la méthode, comparées à la solution exacte de la densité (rouge). 138
5.4 Instabilité de Kelvin Helmholtz, résultats avec la méthode particulaire pour les conditions initiales de Mulder et al., isolignes de ϕ : -1/32, 0., 1/32, de haut en bas et de gauche à droite aux temps $t = 1., 2., 3., 4., 5., 6., 7., 8.$. 141
5.5 Instabilité de Kelvin Helmholtz, résultats avec la méthode particulaire pour les conditions initiales de Mulder et al., isolignes de ϕ : -0.6, -0.4, -0.2, 0., 0.2, 0.4, 0.6, de haut en bas et de gauche à droite aux temps $t = 1., 2., 3., 4., 5., 6., 7., 8.$. 142
5.6 Résultats de Mulder et al. pour l'instabilité de Kelvin-Helmholz : 128*256 points, isolignes -1/32, 0, 1/32 pour la fonction level-set. 143
5.7 Instabilité de Kelvin-Helmholtz, résultats avec la méthode particulaire pour les conditions initiales de Després et Lagoutière, isovaleurs de ϕ, à $t = 8.$, remaillage avec Λ_3, de haut en bas : 50^2, 100^2 et 200^2 particules,. 144
5.8 Résultats de Després et Lagoutière pour l'instabilité de Kelvin Helmholtz au temps $t = 8$. 145
5.9 Instabilité de Kelvin-Helmholtz, résultats avec la méthode particulaire pour les conditions initiales de Després et Lagoutière, évolution de la masse du fluide situé initialement à droite, de haut en bas : grilles de taille 50×500, 100×1000 et 200×2000. 146
5.10 Instabilité de Kelvin-Helmholtz, isovaleurs de ϕ, à $t = 8.$, de hqut en bas, 50^2 particules, 50^2 particules sur la grille grossière et 100^2 particules sur la grille fine, qui porte seulement ϕ, et 50^2 particules sur la grille grossière et 200^2 sur la grille fine, remaillage avec Λ_3. 148
5.11 Evolution de la masse du fluide situé initialement à droite, de haut en bas, grille grossière de taille 50^2, puis avec ϕ résolu sur une grille 100^2 et 200^2. 149
5.12 Interaction choc-bulle, isovaleurs de ϕ, à $t = 0., 0.5, 1., 1.5$, 50×500 particules, remaillage avec Λ_3. 151
5.13 Interaction choc-bulle, isovaleurs de ϕ, à $t = 0.5, 1., 1.5$, 50×500 particules, remaillage avec Λ_3. 152
5.14 Même légende que la figure précédente, avec 100×1000 particules. . . . 153
5.15 Même légende que la figure précédente, avec 200×2000 particules. . . . 154
5.16 Interaction choc-bulle, évolution du rapport masse/masse initiale de la bulle, pour (de haut en bas) 50×500, 100×1000 et 200×2000 particules, remaillage avec Λ_3. 155

5.17 Interaction choc-bulle, isovaleurs de ϕ, à $t = 0.5, 1., 1.5$, utilisation de deux tailles de particules portant toutes deux les variables des équations d'Euler : $dx = 1/100$ autour de la bulle et $dx = 1/50$ ailleurs, remaillage avec Λ_3. 156

5.18 Même légende que la figure précédente, avec $dx = 1/200$ autour de la bulle et $dx = 1/100$ ailleurs 157

5.19 Interaction choc-bulle, évolution de la masse de la bulle, utilisation de deux tailles de particules portant toutes deux les variables des équations d'Euler : $dx = 1/50$ et $dx = 1/100$ autour de la bulle, remaillage avec Λ_3. 158

5.20 Interaction choc-bulle, isovaleurs de ϕ, à $t = 0.5, 1., 1.5$, 50×500 particules (discrétisation N= 50) pour résoudre l'hydrodynamique, et des particules plus fines (N= 100) pour résoudre l'interface, remaillage avec Λ_3. 160

5.21 Même légende que la figure précédente, avec des grilles 100×1000 et 200×2000. 161

5.22 Interaction choc-bulle, évolution de la masse de la bulle, avec 50×500 particules pour l'hydrodynamique, et des particules plus fines ($dx = 1/100$) pour l'interface (ligne plus oscillante, située en haut au début), en bas avec 100×1000 particules pour l'hydrodynamique, et une discrétisation $dx = 1/100$ l'interface (ligne moins oscillante, située en bas au début), et remaillage avec Λ_3. 162

B.1 Spectral numerical viscosity of (a) —— de-aliased spectral method, ·–·–·– 2nd-order central FD, –··–··– 4th-order central FD, –––– 2nd-order central FD with staggered grid. (b) —— EDQNM theory [**?**], –––– ALDM with optimized parameters [41]. 188

B.2 Spectral numerical viscosity of the VIC method with (a) M_4' and (b) Λ_3. 189

B.3 Spectral numerical viscosity of the VIC method with M_4. 190

B.4 Spectral numerical viscosity of the vortex method with a combination of M_4', Λ_3 and M_4 with respective coefficients (a) $c = (0.5, 0.5, 0.0)$, (b) $c = (0.6, 0.4, 0.0)$, (c) $c = (0.7, 0.3, 0.0)$, and (d) $c = (0.599, 0.4, 0.001)$. . 194

B.5 Decay of the kinetic energy of the 2D Taylor-Green vortex. 195

B.6 Rate of kinetic-energy dissipation for LES of the 3D Taylor-Green vortex at $\mathrm{Re} = 400$. —— DNS data from Ref. [10], –––– implicit LES with VIC method. 195

B.7 Instantaneous 3D energy spectra for LES with the VIC method of the Comte-Bellot – Corrsin experiment, LES with 32^3 cells/particles, $dt = 0.03$. (a) Remeshing performed with a combination of M_4' and Λ_3 $(0.6, 0.4, 0.0)$. (b) Remeshing performed with a combination of M_4'. Crosses represent DNS data of A. Wray [86]. ($+$ $t' = 42$, $\times$ $t' = 98$ and $*$ $t' = 171$), symbols denote the present LES. 196

C.1 Comparison of Lax-Wendroff (LW) and particle schemes. Top row : LW (left) and particle (right) schemes with $h = 0.01$, artificial viscosity coefficient 0.2 and $\mu = 0.84$. Bottom left : LW (blue curve) and particle (red curve) schemes without artificial viscosity at $t = 1.5$ and $h = 0.0025$. Bottom right, particle method with adaptive time-step and $h = 0.01$. . . 203

Résumé

L'objectif de cette thèse est d'évaluer la capacité d'une méthode particulaire inspirée des méthodes Vortex-In-Cell, à simuler les écoulements de la dynamique des gaz, et plus particulièrement les écoulements multifluides. Dans un premier temps nous développons donc une méthode particules-grille avec remaillage, pour la simulation d'écoulements compressibles non-visqueux. Le remaillage, conservatif, est réalisé avec des fonctions d'interpolation d'ordre élevé. Nous analysons théoriquement et testons numériquement cette méthode. Nous mettons notamment en évidence des liens forts entre la méthode particules-grilles et des schémas aux différences finies d'ordre élevé, de type Lax-Wendroff, et nous proposons un nouveau schéma d'advection des particules, simple et plus précis. Puis nous implantons une technique multi-niveaux, inspirée de l'AMR, qui permet d'adapter la précision des calculs aux caractéristiques de l'écoulement. Enfin, nous discrétisons une technique de type level-set sur les particules afin de simuler l'interface entre fluides et utilisons la technique multi-niveaux afin de résoudre plus précisément cette interface et d'améliorer la conservation des masses partielles.

Abstract

This work is devoted to the evaluation of the ability of a particle-mesh method, inspired from Vortex-In-Cell methods, to simulate gaz dynamics, especially multifluids. We firstly develop a particle method, associated with a conservative remeshing step, which is performed with high order interpolating kernels. We study theoretically and numerically this method. This analysis gives evidence of a strong relationship between the particle method and high order Lax-Wendroff-like finite difference schemes. We introduce a new scheme for the advection of particles. Then we implement a multilevel technique, inspired from AMR, which allows to increase locally the accuracy of the computations. Finally we develop a level set-like technique, discretized on the particles, to simulate the interface between compressible flows. We use the multilevel technique to improve the interface resolution and the conservation of partial masses.