



HAL
open science

Développement d'un solveur tout-couplé parallèle 3D pour la simulation des écoulements diphasiques incompressibles à forts rapports de viscosités et de masses volumiques

Mohamed El Ouafa

► **To cite this version:**

Mohamed El Ouafa. Développement d'un solveur tout-couplé parallèle 3D pour la simulation des écoulements diphasiques incompressibles à forts rapports de viscosités et de masses volumiques. Mécanique des fluides [physics.class-ph]. Université Gustave Eiffel, 2022. Français. NNT : 2022UEFL2068 . tel-04192595

HAL Id: tel-04192595

<https://theses.hal.science/tel-04192595>

Submitted on 31 Aug 2023

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

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

École doctorale n° 531 : Sciences, Ingénierie et Environnement

Doctorat Paris-Est Sup

THÈSE

pour obtenir le grade de

Docteur de l'Université Gustave Eiffel

Spécialité “Mécanique”

présentée et soutenue publiquement par

Mohamed El Ouafa

le 15 Décembre 2022

Développement d'un solveur tout-couplé parallèle 3D pour la simulation des écoulements diphasiques incompressibles à forts rapports de viscosités et de masses volumiques

Devant le jury composé de:

Olivier Desjardins	Associate professor	Univ. Cornell	Rapporteur
Jean-Luc Estivalèzes	Pr associé	IMFT - INP Toulouse	Rapporteur
Vincent Moureau	CR CNRS	CORIA	Examineur
Vincent Le Chenadec	MCF	MSME - UGE	Co-directeur
Taraneh Sayadi	CR CNRS	∂'Alembert - Sorbonne Université	Examinatrice
Anne Sergent	CR CNRS	Md, LIMSI, Univ. Paris-Saclay	Examinatrice
Benoît Trouette	MCF	MSME - UGE	Invité
Stéphane Vincent	Pr	MSME - UGE	Directeur
Stéphane Zaleski	Pr	∂'Alembert - Sorbonne Université	Président du jury

Laboratoire Modélisation et Simulation Multi Échelle (MSME)

Univ Gustave Eiffel, CNRS UMR 8208

Remerciements

Je remercie toutes celles et tous ceux qui, de près ou de loin, ont contribué à la réussite de cette thèse.

Je tiens à remercier Olivier Desjardins et Jean-Luc Estivalèzes d'avoir accepté de relire cette thèse et d'en être rapporteurs. La version finale de ce manuscrit a bénéficié de leur lecture attentive et leurs remarques pertinentes. Je tiens à remercier Stéphane Zaleski d'avoir accepté d'être président du jury. Je remercie également tous les membres du jury d'avoir accepté d'assister à la présentation de ce travail : Vincent Moureau, Taraneh Sayadi, Anne Sergent et Benoît Trouette.

Je remercie mes directeurs de thèse Stéphane Vincent et Vincent Le Chenadec qui m'ont accordé leur confiance et qui m'ont confié des missions valorisantes. Je les remercie également de m'avoir fait bénéficier de leurs compétences dans leurs domaines respectifs. Ils ont énormément contribué à ce que mes travaux de recherches se déroulent dans une bonne ambiance de travail.

Mes remerciements s'adressent également aux membres l'équipe TCM du laboratoire MSME qui m'ont beaucoup aidé pendant ma thèse. Merci à mes collègues doctorants, Samy, Mathilde, Désir-André, Georges, Dahia, Paul, Nour, Syphax, Faina, Abdelhamid, pour leur aide et leurs différents retours constructifs.

Je remercie également toutes les personnes qui m'ont permis d'oublier les soucis quotidiens de la programmation dans des soirées, repas, sorties ou matchs de foot : Amin, Mustapha, Abdellah, Abdeillah, Taha, Amine, Taha, Yassine,...

Enfin un grand merci à mon papa, à ma maman, à ma soeur et à mon frère pour le soutien qu'ils m'ont apporté, et sans lequel, ce travail n'aurait pas été possible. J'ai une pensée à mes tantes qui m'ont soutenues dès mon arrivée en France.

Résumé

Développement d'un solveur tout-couplé parallèle 3D pour la simulation des écoulements diphasiques incompressibles à forts rapports de viscosités et de masses volumiques.

Les équations de Navier-Stokes diphasiques interviennent dans plusieurs applications industrielles et environnementales. Leur résolution pose cependant de nombreux problèmes et devient un enjeu majeur pour les mathématiciens et les ingénieurs. Dans le cadre des écoulements diphasiques incompressibles, les difficultés sont multiples : d'une part, le couplage vitesse-pression induit par la contrainte d'incompressibilité et les forts rapports de masses volumiques et de viscosités conduisent à des systèmes linéaires très mal conditionnés et, d'autre part, une discrétisation inconsistante entre le transport de la quantité de mouvement et le transport de masse mènent à une divergence du code de calcul dans certaines situations. Ce travail est consacré au développement et à la mise en oeuvre d'un outil de simulation directe d'écoulements diphasiques prenant en compte toutes les échelles interfaciales afin de répondre à nos besoins. Dans ce contexte, un solveur tout-couplé original parallèle 3D permettant d'assurer le couplage direct entre la vitesse et la pression a été développé. Ce solveur a été validé sur des problèmes à l'aide de plusieurs solutions analytiques, de cas tests et d'expériences numériques, quels que soient les ordres de grandeur des discontinuités de masses volumiques et de viscosités à travers l'interface. Une bonne scalabilité associée au solveur tout-couplé a été observée, ceci pour un nombre de processeurs allant de 40 à 12800. Un schéma conservatif AMP (Algebraic Momentum Preserving) basé sur une discrétisation consistante de la masse volumique et de la quantité de mouvement a été construit et validé. Les comparaisons quantitatives et qualitatives ont permis de montrer la robustesse du notre solveur tout-couplé combiné avec le nouveau schéma AMP lorsqu'il fait face à des problèmes complexes.

Abstract

Development of a fully-coupled 3D parallel solver for the simulation of two-phase incompressible flows at large viscosity and density ratios

The two-phase Navier-Stokes equations are involved in several industrial and environmental applications. However, their resolution poses many problems and becomes a major issue for mathematicians and engineers. In the context of incompressible two-phase flows, the challenges are two-fold : on the one hand, the velocity-pressure coupling induced by the incompressibility constraint and the high density and viscosity ratios lead to very ill-conditioned linear systems and, on the other hand, an inconsistent discretization between momentum transport and mass transport lead to a divergence of the computational code in some situations. This work is devoted to the development and implementation of a direct simulation tool for two-phase flows taking into account all interfacial scales in order to meet our needs. In this context, an original 3D parallel fully-coupled solver allowing the direct coupling between velocity and pressure has been developed. This solver has been validated on problems using several analytical solutions, numerical benchmarks and experiments, regardless of the ratios of density and viscosity discontinuities across the interface. Good scalability associated with the fully-coupled solver has been observed, for a number of processors ranging from 40 to 12800. A conservative scheme AMP (Algebraic Momentum Preserving) based on a consistent discretization of density and momentum has been constructed and validated. Quantitative and qualitative comparisons have shown the robustness of our fully-coupled solver combined with the new AMP scheme when facing complex problems.

Table des matières

1	Introduction générale	31
2	Différentes approches pour la modélisation et la simulation d'un écoulement diphasique	41
2.1	Équations de Navier-Stokes pour un écoulement monophasique incompressible (NS1)	42
2.1.1	L'équation de conservation de la masse	43
2.1.2	L'équation de bilan de la quantité de mouvement	44
2.2	Le modèle 2-fluides discontinu pour les interfaces résolues	45
2.3	Représentation des écoulements diphasique incompressibles par un modèle MUF	47
2.4	Filtrage des équations diphasiques	49
2.5	Etat de l'art sur les schémas d'advection de l'interface	51
2.6	Les méthodes lagrangiennes	52
2.6.1	MAC	52
2.6.2	Front tracking	52
2.7	Les méthodes Eulériennes	53
2.7.1	Les Méthodes Level Set	53
2.7.2	Les Méthodes Volume Of Fluid (VOF)	54
2.7.3	La méthode Smooth volume of fluid (SVOF)	61
2.8	Validations de transport	62
2.8.1	Translation d'une sphère dans un champ constant	63
2.8.2	Rotation d'une sphère dans un champ statique	63
2.8.3	Advection d'une sphère dans un champ avec cisaillement	66
2.9	Conclusion	66
3	Résolution numérique du modèle à un fluide	69
3.1	Modèle d'écoulement diphasique utilisé pour la résolution des équations de Navier-Stokes	70
3.2	Intégration temporelle des équations de Navier-Stokes diphasiques	72
3.2.1	Formulation fonction de courant-vorticité $\psi-\omega$	74
3.2.2	La méthode de Prédiction-Correction	76

3.2.3	La méthode d'Uzawa	77
3.2.4	La méthode de décomposition d'opérateurs	79
3.2.5	La méthode du lagrangien augmenté standard	79
3.2.6	La méthode SIMPLE	82
3.3	Bilan des algorithmes de couplage vitesse-pression	84
3.4	Intégration spatiale des équations de Navier-Stokes diphasiques	85
3.4.1	Calcul du terme convectif	87
3.4.2	Calcul du terme Visqueux	88
3.4.3	Calcul du gradient de pression	90
3.4.4	Calcul de la divergence de la vitesse	91
3.4.5	Expression du terme capillaire	91
3.4.6	Traitement des conditions aux limites	93
3.5	Conclusion	94
4	Mise en évidence de l'intérêt d'un solveur tout-couplé original 2D	97
4.1	Développement du solveur tout-couplé en 2D	98
4.2	Préconditionnement du système $A\bar{x} = \bar{b}$	102
4.2.1	Méthodes de factorisation incomplète ILU	104
4.2.2	Méthodes multigrilles	105
4.2.3	Décomposition de domaine de Schwarz	107
4.2.4	Bilan et préconditionneur retenu	108
4.3	Les techniques de préconditionnement par blocs	109
4.3.1	Les techniques de préconditionnement pour le bloc de vitesse	110
4.3.2	Les techniques de préconditionnement pour le complément de Schur	113
4.4	Validations sur des configurations monophasiques et diphasiques	116
4.4.1	Cavité entraînée	116
4.4.2	Écoulement de Poiseuille stationnaire	120
4.4.3	Goutte statique (loi de Laplace)	129
4.4.4	Montée d'une bulle dans un liquide	131
4.4.5	Chute d'une sphère dense dans l'air	135
4.4.6	Nappe liquide cisailée périodique	140
4.4.7	Simulation de l'instabilité d'un jet visqueux en 2D	141
4.5	Conclusion	145
5	Fully-coupled parallel solver for the simulation of two-phase incompressible flows	149
5.1	Models and numerical methods	153
5.1.1	Governing equations of two-phase flows	153
5.1.2	Discretization of mass and momentum equations	153
5.1.3	Block preconditioning techniques	157

5.2	Fully coupled solver parallelization	161
5.2.1	Parallel operations	162
5.2.2	Performance evaluation	162
5.3	Applications to physical cases	164
5.3.1	Phase inversion between two incompressible liquids	164
5.3.2	3D rising bubble	167
5.3.3	Free fall of a dense sphere	171
5.4	Conclusions	172
6	Développement d'un nouveau schéma de transport consistant de la quantité de mouvement et de la masse volumique	175
6.1	Problématique et État de l'art	176
6.2	Développement d'un schéma AMP (Algebraic Momentum Preserving)	179
6.2.1	Avancement en temps de la solution	181
6.2.2	Discrétisation spatiale de la solution	181
6.2.3	Couplage du schéma AMP au modèle à un fluide	185
6.3	Validations et Résultats	186
6.4	Advection d'une goutte dense dans un fluide parfait	187
6.5	Nappe liquide cisailée périodique	190
6.6	Rupture de barrage	191
6.7	Étude paramétrique de l'inversion de phase liquide/liquide	197
6.7.1	Introduction	197
6.7.2	Géométrie, conditions initiales et paramètres physiques	199
6.7.3	Simulations et résultats	200
6.8	Caractérisation de l'instabilité 3D d'un jet visqueux à faible nombre de Reynolds	208
6.8.1	Résultats	210
6.9	Conclusion	211
7	Conclusion et perspectives	213
A	Cas B de l'inversion de phase liquide/liquide	235
B	Physical boundary conditions	239
C	Discretization with finite volumes in Fugu code	241
D	Iterative solver general structure	245
E	Velocity and pressure preconditioning	247
F	Article publié dans Communications in Computational Physics	249
G	Article publié dans Fluids MDPI	285

H	Communication issue du Congrès Français de Mécanique (CFM)	305
I	MUF compressible de fluides non miscibles	321

Table des figures

1.1	L'atomisation d'un jet d'eau dans l'air - L'injection se fait de gauche à droite . La fragmentation du jet met en présence différentes tailles de l'interface, allant des grosses structures aux gouttes les plus fines en passant par les échelles intermédiaires [112].	32
1.2	Exemples d'écoulements à caractère diphasique à phase séparée, mettant en jeu différentes échelles.	33
1.3	Exemples de dysfonctionnement des écoulements diphasiques.	34
2.1	Milieu continu de volume Ω limité par la surface $\partial\Omega$ munie d'une normale extérieure \bar{n}_{ext}	43
2.2	Définition d'un problème diphasique modèle Ω constitué de deux phases Ω_1, Ω_2 et d'une interface Γ	46
2.3	Schéma de la méthode MAC.	53
2.4	Principe de la méthode Front-tracking : (a) topologie simple (b) topologie complexe	54
2.5	Isosurfaces de la fonction Level-set ϕ . L'interface est située à $\phi = 0$. Le signe de ϕ permet de connaître la phase dans laquelle on se situe.	55
2.6	Principe de la méthode VOF, (2.6a) Interface réelle et fonction couleur associée, (2.6b) classe de méthodes SLIC pour reconstruire l'interface, (2.6c) classe de méthodes PLIC pour reconstruire l'interface.	56
2.7	Formation des volumes V1, V2 and V3	60
2.8	Principe de la méthode de VOF Ye & Weymouth	61
2.9	Translation diagonale d'une sphère dans un champ constant. Maillage 512^3 : Évolution de l'interface sphérique.	64
2.10	Translation diagonale d'une sphère dans un champ constant. De gauche à droite, erreur sur la conservation du volume et sur la forme initiale de l'interface.	64
2.11	Rotation d'une sphère dans un champ tournant sans déformation. Maillage 512^3 : Évolution de l'interface sphérique.	65
2.12	Rotation d'une sphère dans un champ tournant sans déformation. De gauche à droite, erreur sur la conservation du volume et la forme de l'interface.	66
2.13	Déformation d'une sphère dans un champ tournant avec cisaillement. Maillage 512^3 : Évolution de l'interface sphérique.	67

2.14	Déformation d'une sphère dans un champ tournant avec cisaillement. De gauche à droite, erreur sur la conservation du volume et la forme de l'interface.	67
2.15	Déformation d'une sphère dans un champ tournant avec cisaillement. Solution d'Enright et al. [65] sur un maillage 512^3	68
3.1	Maillage décalé de type MAC. Volumes de contrôle utilisés. La pression $p_{i,j}$, la viscosité d'élongation $\kappa_{i,j}$ et la fonction de couleur $C_{i,j}$ sont localisées aux nœuds de la grille eulérienne, la vitesse horizontale $u_{i,j}$ est définie sur les faces gauche et droite du volume de contrôle (en vert) associé à $p_{i,j}$ et la vitesse verticale $v_{i,j}$ est définie sur les côtés inférieur et supérieur du volume de contrôle associé à $p_{i,j}$, la vorticit�� $\Omega_{i,j}$, et les viscosit��s de cisaillement et rotation $\eta_{i,j}$ et $\zeta_{i,j}$ sont dispos��es aux sommets du volume de contr��le associ�� �� $p_{i,j}$	86
3.2	Discr��tisation des flux convectifs	88
3.3	Discr��tisation des flux de compression	89
3.4	Discr��tisation des flux de caisaillement	90
3.5	Discr��tisation des flux de rotation	90
3.6	Discr��tisation du gradient de pression	91
3.7	Discr��tisation de la divergence de la vitesse	92
3.8	Discr��tisation du gradient de la fonction couleur	93
3.9	Traitement des conditions aux limites de la fronti��re gauche en 2D	94
4.1	Structure de la matrice tout-coupl��e provenant de la discr��tisation des ��quations de conservation, 5 diagonales non nulles issues du sch��ma centr�� (en rouge pour la composante u , en vert pour la composante v), les diagonales en noir pour le couplage entre u et v , en magenta pour le gradient de pression et en cyan pour la divergence du champ de vitesse.	100
4.2	La factorisation ILU(0) pour une matrice �� un stencil de cinq points.	105
4.3	Sch��mas de V-cycle et W-cycle des m��thodes multigrilles	106
4.4	Passages entre les diff��rentes grilles : prolongation et restriction.	106
4.5	Configuration �� deux grilles, cycle en V.	107
4.6	D��coupage du domaine Ω en 2 sous-domaines disjoints Ω_1 et Ω_2 sans recouvrement.	108
4.7	Comparaison des profils des composantes de la vitesse	118
4.8	��coulement de la cavit�� entra��n��e pour un nombre de Reynolds de 1000. Les contours de la fonction courant sont donn��s, montrant l'apparition de tourbillons primaire et secondaire.	119
4.9	��coulement de la cavit�� entra��n��e pour un nombre de Reynolds de 1000. Comparaison des profils de la premi��re (gauche) et de la deuxi��me (droite) composante de la vitesse extraits sur des droites verticale et horizontale situ��es au milieu de la cavit�� avec les r��sultats de Botella et Peyret [23].	120

4.10	Écoulement de la cavité entraînée pour un nombre de Reynolds de 5000. Comparaison des composantes horizontale et verticale et de vorticité avec les valeurs de référence (https://www.acenumerics.com/the-benchmarks.html)	121
4.11	Simulation numérique sur un maillage 8×257 de l'écoulement de Poiseuille 2D par le solveur tout-couplé, avec un résidu de $\epsilon = 10^{-12}$ pour le solveur BiCGSTAB(2). Comparaison entre la solution analytique et celle issue du calcul.	123
4.12	Simulation numérique sur un maillage 8×257 de l'écoulement de Poiseuille diphasique 2D par le solveur tout-couplé, avec un résidu de $\epsilon = 10^{-12}$ pour le solveur BiCGSTAB(2). Comparaison entre la solution analytique et simulation.	124
4.13	Discrétisation de l'énergie cinétique E_k et de l'enstrophy Ens	126
4.14	Simulation numérique de l'écoulement de Taylor-Green 2D par le solveur lagrangien augmenté. Variation de l'erreur relative à la solution exacte de l'énergie cinétique E_k en fonction du maillage, et pour différentes valeurs du paramètre du lagrangien augmenté r	127
4.15	Simulation numérique de l'écoulement de Taylor-Green 2D par le solveur tout-couplé. Évolution des normes d'erreurs en fonction du maillage utilisé pour l'énergie cinétique E_k et l'enstrophie Ens , dans le cas d'un fluide parfait ($\mu = 0$).	128
4.16	Simulation numérique de l'écoulement de Taylor-Green 2D par le solveur tout-couplé. Évolution des normes d'erreurs en fonction du maillage utilisé pour l'énergie cinétique E_k et l'enstrophie Ens , dans le cas d'un fluide visqueux ($\mu = 10^{-3}$).	128
4.17	Simulation numérique de l'écoulement de Taylor-Green 2D par le solveur tout-couplé, lagrangien augmenté et la projection standard. Temps de calcul CPU requis pour simuler l'écoulement de Taylor-Green sur une grille cartésienne fine 256×256	129
4.18	Cas de la goutte statique	130
4.19	Configuration initiale du problème de la montée d'une bulle dans une colonne liquide.	131
4.20	Simulation numérique de l'écoulement d'une montée d'une bulle dans une colonne liquide par le solveur tout-couplé. Comparaison de la vitesse d'ascension 4.20a et de la position du centre de masse 4.20b pour le cas-test 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$). Comparaison entre le solveur tout-couplé et les solutions des autres groupes [98].	132
4.21	Simulation numérique de l'écoulement d'une montée d'une bulle dans une colonne d'eau par le solveur tout-couplé. Comparaison de la vitesse d'ascension 4.21a et de la position du centre de masse 4.28c pour le cas-test 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$). Les solutions ont été obtenues pour différents pas de temps : $\Delta t = 10^{-3}s$ (représenté par des triangles bleus), $\Delta t = 5 \cdot 10^{-3}s$ (représenté par des carrés rouges) et $\Delta t = 10^{-2}s$ (représenté par des croix noires).	133

- 4.22 Simulation numérique de l'écoulement d'une montée d'une bulle dans une colonne liquide par la méthode du lagrangien augmenté adaptatif. Comparaison de la vitesse d'ascension 4.22a et de la position du centre de masse 4.22b pour le cas-test 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$). Les solutions ont été obtenues pour différents pas de temps : $\Delta t = 10^{-3}s$ (représenté par des triangles bleus), $\Delta t = 5 \cdot 10^{-3}s$ (représenté par des carrés rouges) et $\Delta t = 10^{-2}s$ (représenté par des croix noires). 134
- 4.23 Simulation numérique de l'écoulement d'une montée d'une bulle dans une colonne liquide par la méthode de la projection standard. Comparaison de la vitesse d'ascension 4.23a et de la position du centre de masse 4.23b pour le cas-test 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$). Les solutions ont été obtenues pour différents pas de temps : $\Delta t = 10^{-3}s$ (représenté par des triangles bleus), $\Delta t = 5 \cdot 10^{-3}s$ (représenté par des carrés rouges) et $\Delta t = 10^{-2}s$ (représenté par des croix noires). 134
- 4.24 Simulation numérique de l'écoulement d'une montée de bulle dans une colonne liquide par le solveur tout-couplé. Formes de l'interface à l'instant final $t = 3s$ pour le cas test 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$). Plusieurs pas de temps ont été utilisé : $\Delta t = 10^{-3}s$ (représenté par des lignes en pointillés en rouge), $\Delta t = 5 \cdot 10^{-3}s$ (représenté par des lignes discontinues en bleu) et $\Delta t = 10^{-2}s$ (représenté par des lignes pleines en noir). 135
- 4.25 Simulation numérique de l'écoulement d'une montée d'une bulle dans une colonne liquide par le solveur tout-couplé, lagrangien augmenté et la projection standard. Temps de calcul CPU requis pour simuler l'écoulement de la montée de la bulle sur une grille cartésienne fine 320×640 136
- 4.26 Simulation numérique de la chute d'une sphère dense dans l'air par le solveur tout-couplé, le lagrangien augmenté et la méthode de projection standard sur une grille cartésienne 400×800 . Dans la première ligne, le solveur tout-couplé (représenté par les champs) et le lagrangien augmenté (par les isolignes). Dans la deuxième rangée, le solveur tout-couplé (représenté par les champs) et la méthode de projection standard (représentée par les isolignes avec $\Delta t = 6.25 \times 10^{-6} s$). Les différents champs présentés dans chaque ligne sont : à gauche : vitesse horizontale $[0 \text{ m}, 0.1 \text{ m}] \times [0 \text{ m}, 0.11 \text{ m}]$, au milieu : vitesse verticale $[0 \text{ m}, 0.1 \text{ m}] \times [0 \text{ m}, 0.10 \text{ m}]$, à droite : la norme de la vorticité $[0.02 \text{ m}, 0.08 \text{ m}] \times [0.034 \text{ m}, 0.089 \text{ m}]$. 138
- 4.27 Simulation numérique de la chute d'une sphère dense dans l'air par le solveur tout-couplé sur différentes grilles cartésiennes avec ($\rho_1/\rho_2 = 10^5, \mu_1/\mu_2 = 10^{10}$). . 138
- 4.28 Simulation numérique de la chute d'une sphère dense dans l'air sur différentes grilles cartésiennes avec ($\rho_1/\rho_2 = 10^5, \mu_1/\mu_2 = 10^{10}$). Évolution de l'enstrophie pour le solveur tout-couplé (a), le lagrangien augmenté adaptatif (b) et la projection standard (c). Tout est calculé avec $\Delta t = 6.2510^{-6}s$ sauf le cas jaune réalisé avec la projection standard. 141
- 4.29 Cas de la nappe liquide cisailée périodique. 142

4.30 Simulation numérique d’une nappe liquide cisailée périodique sur une grille cartésienne (256×256). Évolution temporelle de l’énergie cinétique normalisée, obtenue par différents solveurs avec un critère d’arrêt $\epsilon = 10^{-6}$ du BiCGSTAB(2). Le rapport de masses volumiques (a) est de 100 et le rapport de masses volumiques (b) est 1000. 142

4.31 Simulation numérique du remplissage d’une cavité par un jet visqueux. Résultats correspondant à trois configurations, simulées par le solveur tout-couplé, sur une grille cartésienne 320×320 . L’interface est représentée au instants $t = 0.25$ s et 0.75 s (première ligne), $t = 1.25$ s et 2 s (deuxième ligne). La colonne 1 correspond à $Re = 0.1728$, la colonne 2 correspond à $Re = 0.6$ et la colonne 3 correspond à $Re = 3.6$ 144

4.32 Simulation numérique du remplissage d’une cavité par un jet visqueux. Résultats correspondant à la première configuration, simulée par le solveur tout-couplé, sur une grille cartésienne 320×320 . L’interface est représentée aux temps $t = 0, 25$ s et $0, 75$ s (première ligne), $t = 1, 25$ s et 2 s (deuxième ligne). Le cas test correspond à un jet instable pour $Re = 0, 1728$ 145

4.33 Simulation numérique du remplissage d’une cavité par un jet visqueux. Résultats correspondant à la première configuration, simulée par différents solveurs sur une grille cartésienne 320×320 . L’interface est représentée au temps $t = 0.25$ s (première ligne), $t = 0.75$ s (deuxième ligne), $t = 1.25$ s (troisième ligne) et $t = 2$ s (quatrième ligne). Première colonne (solveur tout-couplé), deuxième colonne (méthode du lagrangien augmenté adaptatif) et troisième colonne (la méthode de projection standard). 146

5.1 Staggered u -velocity (left) and pressure (right) unknowns and their respective local indexes. Velocity control volume is centered on u -velocity (∇ symbols). Local index 0 stands for global indexes i, j and k . Directional full (± 1) or half shifts ($\pm 1/2$) allow to reach neighbours used in the 3 points discretization stencil. The horizontal velocity is linked through divergence and diffusive terms to v - and w - components (\square and \diamond) and also to the pressure (\circ). Similar notations and connectivities are adopted for the treatment of v - and w -components. All the three velocity component then involve 17 coupled variables. 155

5.2 Structure of the fully-coupled matrix, 7 diagonal form the 3 directional points stencil (in red for u -, green for v - and blue for w -component, respectively). Black lines for velocity coupling, magenta for pressure gradient and cyan for divergence. 157

5.3 Strong (filled symbols) scalability on a 138 million scalar cell mesh and weak scalability (empty ones) on meshes corresponding to $60^3 \times p$ (from 5 millions up to 1.1 billion scalar cells). 164

5.4 Kinetic energy in fluid 1 165

5.5 Kinetic energy in fluid 2 165

5.6	Potential energy in fluid 1	166
5.7	Potential energy in fluid 2	166
5.8	Enstrophy in fluid 1	166
5.9	Enstrophy in fluid 2	166
5.10	Dissipation in fluid 1	166
5.11	Dissipation in fluid 2	166
5.12	Interfacial area	166
5.13	Mass conservation	166
5.14	the effect of the residual.	167
5.15	the effect of the time step.	167
5.16	Enstrophy in fluid 1: study of the effect of the residual and the time step on the quality of the solution.	167
5.17	Bubble rise for test case 1. The interface $C = 0.5$ is plotted at times $t = 3s$ (first row) and $t = 2s$ (second row). First column holds for DROPS code, second column for NaSt3D, third column for OpenFoam and fourth column to our Fully-coupled solver.	169
5.18	Test case 1. a Rise velocity w -component as un function of time. b Center of mass position. Plain lines stand for the present work while symbols refer to benchmark results [1] from different codes.	169
5.19	Same as Fig. 5.17 for test case 2.	170
5.20	Same as Fig. 5.18 for test case 2.	170
5.21	Numerical simulation of a free fall of dense sphere at $t = 0.144$ s, obtained on a fine mesh $128 \times 128 \times 256$, by the fully coupled solver. The different fields presented in each row are: on the left, the color function, in the middle, the velocity magnitude and on the right, the vorticity magnitude.	172
5.22	Vertical velocity and center of mass for the free fall of a dense and viscous drop ($\rho_1/\rho_2 = \mu_1/\mu_2 = 10^6$) obtained by the fully coupled method with several meshes. left: Vertical velocity and right: center of mass.	172
6.1	Disposition des noeuds autour du volume de contrôle utilisé pour la discrétisation des flux ϕ sur les différentes faces.	182
6.2	Stencil du schéma CUI pour évaluer la quantité ϕ sur la face est du volume de contrôle.	184
6.3	Stencil du schéma CUI pour évaluer la quantité ϕ sur la face nord du volume de contrôle.	184
6.4	Stencil du schéma WENO pour évaluer la quantité ϕ sur la face est du volume de contrôle.	185
6.5	Disposition des noeuds autour du volume de contrôle utilisé pour la construction de la couleur C_α^n	186

6.6	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 64^2 : Forme de la goutte après une période avec la formulation non conservative. (a) : forme initiale de la goutte à $t = 0 s$, (b) : forme finale de la goutte à $t = 1 s$ et (c) : module de la vitesse à $t = 1 s$	187
6.7	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 128^2 : Forme de la goutte après une période avec la formulation non conservative. (a) : forme initiale de la goutte à $t = 0 s$, (b) : forme finale de la goutte à $t = 1 s$ et (c) : module de la vitesse à $t = 1 s$	188
6.8	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 256^2 : Forme de la goutte après une période avec la formulation conservative pour un schéma de transport de type WENO. (a) : forme initiale de la goutte à $t = 0 s$, (b) : forme finale de la goutte à $t = 1 s$ et (c) : module de la vitesse à $t = 1 s$	188
6.9	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 256^2 : Forme de la goutte après une période avec la formulation conservative pour un schéma de transport de type CUI. (a) : forme initiale de la goutte à $t = 0 s$, (b) : forme finale de la goutte à $t = 1 s$ et (c) : module de la vitesse à $t = 1 s$	189
6.10	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 256^2 : Forme de la goutte après une période avec la formulation conservative pour un schéma de transport de type Cubista. (a) : forme initiale de la goutte à $t = 0 s$, (b) : forme finale de la goutte à $t = 1 s$ et (c) : module de la vitesse à $t = 1 s$	189
6.11	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 128^2 : évolution de l'énergie cinétique en fonction du temps pour la formulation non conservative.	189
6.12	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 256^2 : évolution de l'énergie cinétique en fonction du temps pour la formulation conservative. (a) : énergie cinétique $Ek(t)$ normalisée sur sa valeur initiale $EK(0)$, (b) : erreur en norme L1 sur l'énergie cinétique $EK(t)$ par rapport à sa valeur initiale $EK(0)$	190
6.13	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 256^3 : évolution de l'énergie cinétique en fonction du temps pour la formulation conservative. (a) : énergie cinétique $Ek(t)$ normalisée sur sa valeur initiale $EK(0)$, (b) : erreur en norme L1 sur l'énergie cinétique $EK(t)$ par rapport à sa valeur initiale $EK(0)$	191
6.14	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 256^3 : forme de la goutte après une période avec la formulation conservative pour un schéma de transport de type CUBISTA. La simulation a été réalisée sur 512 processeurs du cluster HPC Irene de TGCC.	192

6.15	Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : représentation de l'évolution de l'interface de la nappe avec la formulation non conservative pour des schémas de transport de type centré. (a) : champ de vitesse, (b) : forme de l'interface.	193
6.16	Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : représentation de l'évolution de l'interface de la nappe avec la formulation conservative pour un schéma de transport de type WENO. (a) : champ de vitesse, (b) : forme de l'interface.	193
6.17	Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : représentation de l'évolution de l'interface de la nappe avec la formulation conservative pour un schéma de transport de type CUI. (a) : champ de vitesse, (b) : forme de l'interface.	194
6.18	Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : représentation de l'évolution de l'interface de la nappe avec la formulation conservative pour un schéma de transport de type Cubista. (a) : champ de vitesse, (b) : forme de l'interface.	194
6.19	Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : évolution de l'énergie cinétique $Ek(t)$ normalisée sur sa valeur initiale $EK(0)$ avec la formulation non conservative.	195
6.20	Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : évolution de l'énergie cinétique en fonction du temps pour la formulation conservative. (a) : énergie cinétique $Ek(t)$ normalisée sur sa valeur initiale $EK(0)$, (b) : erreur en norme L1 sur l'énergie cinétique $EK(t)$ par rapport à sa valeur initiale $EK(0)$	195
6.21	Rupture de barrage sur un fond sec	195
6.22	Simulation numérique de la rupture de barrage sur fond sec avec la formulation conservative. L'évolution temporelle de la surface libre est présentée. Paramètres de calcul : maillage $320 \times 100 \times 100$, $CFL = 0.4$ et résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2). La simulation a été réalisé sur 120 processeurs du cluster HPC JeanZay de l'IDRIS.	196
6.23	Simulation numérique de la rupture de barrage sur fond sec avec la formulation conservative. L'évolution temporelle de la surface libre est présentée en comparaison avec l'expérience de Kleefsman et al. aux temps $0.4s$ et $0.56s$. paramètres de calcul : maillage $320 \times 100 \times 100$, $CFL = 0.4$ et résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2).	197
6.24	Simulation numérique de la rupture de barrage sur fond sec avec la formulation conservative. L'évolution temporelle de la hauteur H4 est présentée en comparaison avec l'expérience de Kleefsman et al [106]. Paramètres de calcul : maillage $320 \times 100 \times 100$, $CFL = 0.4$ et résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2).	197
6.25	Configuration initiale pour un cas 3D de l'inversion de phase liquide/liquide.	198

6.26	Simulation numérique de l'inversion de phase liquide/liquide. L'évolution temporelle de l'interface des 2 fluides est représentée en jaune. Paramètres de calcul : un maillage $256 \times 256 \times 256$, une $CFL = 0.5$ et un résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2).	203
6.27	Simulation numérique de l'inversion de phase liquide/liquide. (a) : le ration R2, (b) : le volume de fluide léger Vol.	204
6.28	Simulation numérique de l'inversion de phase liquide/liquide. (a) : L'aire interfaciale $Aire$. (b) : L'énergie surfacique E_{surf} .	205
6.29	Simulation numérique de l'inversion de phase liquide/liquide. (a) : $Ep, 1$, énergie potentielle dans le fluide lourd. (b) : $Ep, 2$, énergie energie potentielle dans le fluide léger.	206
6.30	Simulation numérique de l'inversion de phase liquide/liquide. (a) : $Ek, 1$, énergie cinétique en coordonnées log-log dans le fluide lourd. (b) : $Ek, 2$, énergie cinétique en coordonnées log-log dans le fluide léger. L'approximation des décroissances est en ligne noire.	206
6.31	Simulation numérique de l'inversion de phase liquide/liquide. (a) : $Er, 1$, enstrophie en coordonnées log-log dans le fluide lourd. (b) : $Er, 2$, enstrophie en coordonnées log-log dans le fluide léger. L'approximation des décroissances est en ligne noire.	207
6.32	Simulation numérique de l'inversion de phase liquide/liquide. (a) : $Di, 1$, dissipation dans le fluide lourd. (b) : $Di, 2$, dissipation dans le fluide léger. Comparaison entre la dissipation calculée par le code et la dissipation calculée avec l'approximation $Di, n \approx \mu Er, n$.	207
6.33	Simulation numérique de l'inversion de phase liquide/liquide. (a) : $Ek, 1$, énergie cinétique dans le fluide lourd. (b) : $Ek, 1$, énergie cinétique dans le fluide léger.	208
6.34	Simulation numérique de l'inversion de phase liquide/liquide. (a) : $Er, 1$, enstrophie dans le fluide lourd. (b) : $Er, 1$, enstrophie dans le fluide léger.	208
6.35	Simulation numérique de l'inversion de phase liquide/liquide. (a) : bilan énergétique. (b) : comparaison entre la dissipation ϵ et la variation temporelle de l'énergie totale.	209
6.36	Configuration initiale pour l'injection d'un jet visqueux à faible nombre de Reynolds.	209
6.37	Simulation numérique du remplissage d'une cavité par un jet visqueux. Résultats correspondant à $Re = 0.6$, simulés par le solveur tout-couplé, sur une grille cartésienne 100^3 , $Re = 0.6$. L'interface des deux fluides est représentée en bleu.	210
6.38	Simulation numérique du remplissage d'une cavité par un jet visqueux. Résultats correspondant à $Re = 0.6$, simulés par le solveur tout-couplé, sur une grille cartésienne 100^3 , $Re = 0.1728$. L'interface des deux fluides est représentée en rose.	211

A.1	Simulation numérique de l'inversion de phase liquide/liquide. L'évolution temporelle de l'interface des 2 fluides est représentée en jaune. Paramètres de calcul : un maillage $256 \times 256 \times 256$, une $CFL = 0.5$ et un résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2).	236
A.2	Simulation numérique de l'inversion de phase liquide/liquide. (a) : $Ek, 1$, énergie cinétique dans le fluide lourd. (b) : $Ek, 2$, énergie cinétique dans le fluide léger.	237
A.3	Simulation numérique de l'inversion de phase liquide/liquide. (a) : $Ep, 1$, énergie potentielle dans le fluide lourd. (b) : $Ep, 2$, énergie potentielle dans le fluide léger.	237
A.4	Simulation numérique de l'inversion de phase liquide/liquide. (a) : $Er, 1$, enstrophie dans le fluide lourd. (b) : $Er, 1$, enstrophie dans le fluide léger.	237
A.5	Simulation numérique de l'inversion de phase liquide/liquide. (a) : L'aire interfaciale $Aire$. (b) : L'énergie surfacique E_{surf} .	238
A.6	Simulation numérique de l'inversion de phase liquide/liquide. (a) : le ration R2, (b) : le volume de fluide léger Vol.	238
A.7	Simulation numérique de l'inversion de phase liquide/liquide. (a) : bilan énergétique. (b) : comparaison entre la dissipation et la variation temporelle de l'énergie totale.	238

Liste des tableaux

3.1	Récapitulatif des spécificités et des performances de certains algorithmes de couplage vitesse-pression. ψ - ω : formulation fonction courant-vorticité, PC : Projection-correction, DO : Décomposition d'opérateurs, LA : lagrangien augmenté.	85
3.2	Tableau des valeurs des coefficients de pondération de la combinaison linéaire $f(\bar{u}^{n+1})$	95
4.1	Approximation de l'action de l'inverse \tilde{F}_u sur un vecteur $\bar{z} = (z_u, z_v)^T$, le vecteur solution est $\bar{y} = (r_u, r_v)^T$	112
4.2	Approximation de l'action de l'inverse $F_{vv}(I - G)$ sur le vecteur \bar{z}_v , le vecteur solution est $\bar{y} = (r_v)^T$	113
4.3	Description de l'application du préconditionneur triangulaire par blocs. $M_p^{1/\mu}$ est la matrice de masse normalisée par l'inverse de la viscosité $1/\mu$, $M_p^{(1)}$ est la matrice de masse standard, $N_p^{(1)}$ représente la partie discrète du terme convectif standard dans l'espace de pression, $A_p^{1/\rho}$ est le laplacien normalisé, qui correspond à la discrétisation du terme $\nabla \cdot \left(\frac{1}{\rho} \nabla P\right)$, et F_u est l'opérateur de convection-diffusion pour l'équation de la quantité de mouvement. Ici, $\bar{r} = (r_u, r_v, r_p)^T$ est égale à \bar{b} , \bar{q} , \bar{v} , \bar{s} , \bar{w} et \bar{t} dans le BICGSTAB(2) (voir Algo.4.2).	117
4.4	Comparaison de quelques caractéristiques de l'écoulement	118
4.5	Écoulement de la cavité entraînée pour un nombre de Reynolds de 1000. L'extrapolation de Richardson pour le tourbillon primaire : (x, y) la position du centre du tourbillon, et ω est la valeur de la vorticité à cet emplacement.	120
4.6	Écoulement de la cavité entraînée pour un nombre de Reynolds de 1000. L'extrapolation de Richardson pour le tourbillon secondaire droit : (x, y) la position du centre du tourbillon, et ω est la valeur de la vorticité à cet emplacement.	120
4.7	Écoulement de la cavité entraînée pour un nombre de Reynolds de 1000. L'extrapolation de Richardson pour le tourbillon secondaire gauche : (x, y) la position du centre du tourbillon, et ω est la valeur de la vorticité à cet emplacement.	121
4.8	Écoulement de la cavité entraînée pour un nombre de Reynolds de 5000. Extrema de la vitesse à travers les lignes centrales de la cavité, obtenus en utilisant une grille cartésienne 512×512 - Ref sont les valeurs de références données sur le site : https://www.acenumerics.com/the-benchmarks.html	122

- 4.9 Écoulement de la cavité entraînée pour un nombre de Reynolds de 5000, sur une grille cartésienne 512×512 . Carte des tourbillons de la cavité. Ref dénote les valeurs de références données sur le site : <https://www.acenumerics.com/the-benchmarks.html>. PV, BR, BL, and TL représentent les tourbillons primaire, inférieur droit, inférieur gauche et supérieur gauche, respectivement. 122
- 4.10 Simulation numérique sur des maillage $8 \times (9, 17, 33, 65, 129, 257)$ de l'écoulement de Poiseuille 2D par le solveur tout-couplé, avec un résidu de $\epsilon = 10^{-12}$ pour le solveur BiCGSTAB(2). Normes d'erreur entre profil de vitesse théorique et simulé selon le maillage. Approche VOF-PLIC avec une moyenne calculée de façon arithmétique et une méthode VOF-PLIC avec une moyenne harmonique. 124
- 4.11 Simulation numérique de l'écoulement d'une goutte statique 2D par le solveur tout-couplé. Erreur sur le saut de pression $\|E_{\Delta P_{max}}\|$ dans la goutte et évolution des nombres capillaires en fonction du nombre de Laplace (valeurs précises après 5 secondes de simulation). 131
- 4.12 Simulation numérique de la chute d'une sphère dense dans l'air. Ici, ρ_2 et μ_2 désignent la masse volumique et la viscosité du fluide à l'intérieur de la particule ρ_1 et μ_1 désignent la masse volumique et la viscosité du fluide environnant et g représente la gravité. 136
- 4.13 Simulation numérique de la chute d'une sphère dense dans l'air par le solveur tout-couplé sur différentes grilles cartésiennes avec $(\rho_1/\rho_2 = 10^5, \mu_1/\mu_2 = 10^{10})$. Normes d'erreurs $Err_2^{rel}(V_c)$ de la vitesse de chute et $Err_2^{rel}(Y_c)$ de l'évolution du centre de masse. 139
- 4.14 Simulation numérique de la chute d'une sphère dense dans l'air par le solveur tout-couplé, le lagrangien augmenté et la méthode de projection standard sur une grille fine de taille 400×800 avec $(\rho_1/\rho_2 = 10^5, \mu_1/\mu_2 = 10^{10})$. Normes d'erreurs $Err_2^{rel}(V_c)$ de la vitesse de chute et $Err_2^{rel}(Y_c)$ de l'évolution du centre de masse obtenues à l'instant final $t = 0.144s$. FC : le solveur tout-couplé, AAL : le lagrangien augmenté adaptatif et SP : la méthode de projection standard. NC (Non Communiqué), c'est à dire que la simulation s'est interrompue en raison de la divergence du solveur. 139
- 4.15 Simulation numérique de l'instabilité d'un jet visqueux en 2D par le solveur tout-couplé. Ici, ρ_2 et μ_2 correspondent respectivement à la masse volumique et à la viscosité du propergol, ρ_1 et μ_1 représentent respectivement la masse volumique et la viscosité du gaz, $Re = \frac{\rho_2 D V_0}{\mu_2}$ est le nombre de Reynolds associé au propergol, $\frac{H}{D}$ est le rapport de forme du jet et V_0 est la vitesse d'injection. Dans le présent travail, le propergol (fluide 2) est supposé se comporter comme un fluide newtonien, pour des raisons de simplicité. 143

5.1	Physical parameters describing the test cases. Here, ρ_2 and μ_2 refers to the fluid density and viscosity respectively inside the bubble whereas ρ_1 and μ_1 are related to the density and viscosity of the surrounding fluid. Also, g is the magnitude of the gravity acceleration and σ is the surface tension coefficient.	168
6.1	Synthèse des schémas de transport consistant de la quantité de mouvement et de la masse volumique.	179
6.2	Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Comparaison entre la formulation conservative (schéma WENO, CUI et CUBISTA) et la formulation non conservative (schéma centré). Normes d'erreur entre le champ de couleur à l'instant final (C^{final}) et le champ de couleur à l'instant initial ($C^{initial}$).	190
6.3	Simulation numérique de l'inversion de phase liquide/liquide par le solveur tout-couplé. Ici, ρ_2 et μ_2 correspondent respectivement à la masse volumique et à la viscosité du fluide léger, ρ_1 et μ_1 représentent respectivement la masse volumique et la viscosité du fluide lourd, σ est la tension de surface, Ar est nombre d'Archimède et Bo est le nombre de Bond.	199
6.4	Simulation numérique de l'inversion de phase liquide/liquide. Valeurs initiales et finales de l'énergie potentielle pour un maillage 512^3	205
6.5	Les paramètres physiques décrivant le cas du jet visqueux. Ici, ρ_2 et μ_2 représentent respectivement la masse volumique et la viscosité du propergol, ρ_1 et μ_1 représentent respectivement la masse volumique et la viscosité du gaz, $Re = \frac{\rho_2 D V_0}{\mu_2}$ est le nombre de Reynolds associé au propergol, $\frac{H}{D}$ est le rapport d'aspect du jet et V_0 est la vitesse d'entrée. (pour des raisons de simplicité, le propergol est supposé se comporter comme un fluide newtonien).	210

Nomenclature

$\bar{\gamma}$	Le champ d'accélération
δ_I	La distribution de Dirac
\bar{U}_I	La vitesse de l'interface
\mathcal{F}	L'opérateur de filtrage
\bar{g}	Le vecteur de l'accélération gravitationnelle
$\bar{g}(\bar{x}, t)$	Grandeur vectorielle définie dans $\Omega(t)$
$g(\bar{x}, t)$	Grandeur scalaire définie dans $\Omega(t)$
$\partial\Omega$	Surface délimitant le volume Ω
Ω	Volume à simuler
χ	L'indicatrice de phase
\varkappa	La courbure locale
$\widehat{\Phi}_k(\bar{x}, t)$	La moyenne locale de la grandeur physique $\Phi_k(\bar{x}, t)$
μ	La viscosité dynamique
Φ_f	Le flux évalué sur une face f
$\tilde{\Phi}_f$	Le flux normalisé évalué sur une face f
Φ	La quantité à transporter, $\Phi = \rho_\alpha$ ou $\Phi = \rho_\alpha \bar{u}_\alpha$
\bar{X}_i	La position du marqueur i
p	Le champ de pression
ρ	La masse volumique du fluide
$\bar{F}(t)$	La résultante des forces
$\bar{M}(t)$	La quantité de mouvement
\bar{x}	point géométrique de coordonnées cartésiennes (x, y, z)
\bar{n}_{ext}	Normale extérieure à la surface $\partial\Omega$
σ	Le coefficient de tension de surface

Γ	L'interface qui sépare deux phases
$\bar{\sigma}$	Le tenseur des contraintes
\bar{t}	Le vecteur tangent à l'interface
t	Le temps
\bar{t}	Le vecteur contrainte
$\bar{\bar{T}}$	Le tenseur des taux de déformation
$\bar{\bar{I}}$	Le tenseur identité
\bar{u}_{adv}	La vitesse advective
\bar{u}	Le champ de vitesse locale

Liste des abréviations

HPC	High Performance Computing
VOF	Volume Of Fluid
LS	Level-Set
CLSVOF	Coupled Level-Set Volume Of Fluid
FC	Fully-Coupled
SAL	Standard Augmented Lagrangian
AAAL	Algebraic Automatic Augmented Lagrangian
KSP	Kinematics Scalar Projection
LSC	Least Squares Commutator
PCD	Pressure Convection Diffusion
CFL	Courant–Friedrichs–Lewy
MOF	Moment Of Fluid
MUF	Modèle à un fluide
PFD	Principe fondamental de la dynamique
DNS	Direct Numerical Simulation
LES	Large Eddy Simulation
MAC	Marker-And-Cell
SLIC	Simple Linear Interface Calculation
PLIC	Piecewise Linear Interface Calculation
LVIRA	Least squares Volume of fluid Interface Reconstruction Algorithm
CIAM	Calcul d’Interface Affine par Morceaux
SVOF	Smooth Volume Of Fluid
CSF	Continuum Surface Force
SIMPLE	Semi-Implicit Method for Pressure Linked Equations
BiCGSTAB	Biconjugate gradient stabilized
GCR	Generalized Conjugate Residu
PCD	Pressure Convection Diffusion
AMP	Algebraic Momentum Preserving
CUI	Cubic Upwind Interpolation
CUBISTA	Convergent and Universally Bounded Interpolation Scheme for the Treatment of Advection
WENO	Weighted Essentially Non-Oscillatory

Chapitre 1

Introduction générale

Problématique et enjeux de la thèse

L’omniprésence des écoulements diphasiques rend leur prédiction nécessaire dans des domaines aussi variés que celui de l’énergie, l’hydraulique, l’océanographie, le génie chimique, l’industrie pétrolière ou encore le génie civil, où les conditions et les régimes rencontrés diffèrent grandement. Par exemple, dans l’énergie, l’atomisation des jets (Fig. 1.1) [43, 93, 112, 149] ou les sprays rencontrés lors du refroidissement de parois de bâtiments ou dans les moteurs, résultent des interactions entre un jet liquide, un plénum rempli de gaz et (potentiellement) un jet gazeux [113]. Dans le génie chimique où pour les besoins du procédé, des écoulements diphasiques sont délibérément produits où l’on met deux phases en contact, un liquide et un gaz par exemple, pour former une nouvelle espèce à leur interface commune. Les écoulements d’impact de goutte sur des surfaces (Fig. 1.2a) [195, 197, 198], la fabrication des dépôts par jet de plasma thermique (Fig. 1.2b) en particuliers métalliques, sont diphasiques et turbulents. En océanographie, le déferlement d’une vague (Fig. 1.2c) [27, 53, 127] sur une plage est un problème diphasique (eau-air), caractérisé par un fort degré de turbulence, dont les grandes échelles de la surface libre sont pilotées par les effets inertiels et gravitationnels, tandis que les structures fines apparaissent ou croissent du fait des contraintes visqueuses et de la tension de surface. Ce genre d’écoulements intéresse particulièrement l’industrie pétrolière qui reste l’un des sous-domaines les plus actifs au niveau de la mécanique des fluides. Comme exemple, nous pouvons citer le processus de la séparation des émulsions eau/huile. En ce qui concerne le domaine du génie civil, le béton frais est, par exemple, constitué de particules solides discrètes distribuées aléatoirement dans une matrice liquide [150]. Un autre exemple facile à décrire et qui montre de façon qualitative les problèmes rencontrés concerne l’instabilité 3D d’un jet visqueux (Fig. 1.2d)[101, 129, 169], pour n’en donner qu’une liste non-exhaustive.

Il est donc intéressant de pouvoir contrôler le comportement de ces écoulements, qui peuvent faire l’objet d’un dysfonctionnement comme la crise d’ébullition dans les centrales nucléaires (Fig. 1.3a), le débordement des barrages (Fig. 1.3b) ou la cavitation dans les turbomachines. Par exemple, la crise d’ébullition est un problème de sûreté nucléaire qui engendre une augmen-



FIGURE 1.1 – L’atomisation d’un jet d’eau dans l’air - L’injection se fait de gauche à droite . La fragmentation du jet met en présence différentes tailles de l’interface, allant des grosses structures aux gouttes les plus fines en passant par les échelles intermédiaires [112].

tation de la température à la paroi (flux maximal), et qui présente un risque lié aux substances radioactives. D’autres conséquences liées à ce phénomène critique pourraient conduire dans certaines situations à une dégradation brutale de l’échange thermique, suivi d’une rupture des gaines de crayon de combustibles. Le débordement de barrage est un phénomène diphasique où les ouvrages retenant de l’eau peuvent se rompre sous la pression de l’eau et dont les enjeux sont également importants. Quant à la cavitation dans les pompes, elle traduit les changements de phases liquide-vapeur suite à une diminution de pression. Elle s’accompagne de phénomènes physiques comme l’implosion des bulles de vapeur ou des ondes de choc qui détériorent le fonctionnement de la pompe.

L’approche expérimentale, bien que très largement utilisée dans plusieurs travaux de recherche, souffre parfois de certaines limitations. Elle est aussi plus délicate à appliquer à certains problèmes diphasiques. En effet, outre la complexité de ces écoulements c’est-à-dire, la topologie des interfaces (phases séparées ou dispersées), la nature des interactions (frottements, effets capillaires ...), les régimes d’écoulement (en terme de nombres de Reynolds ou de Weber, par exemple) ou encore la présence de différentes gammes d’échelles spatio temporelles (allant de la phase séparée à l’échelle des petites interfaces), il est parfois impossible de reproduire les conditions réelles à l’échelle du laboratoire et, également, de mesurer des variables intéressantes telles que la PDF des gouttelettes, la surface interfaciale ou l’intensité de la turbulence. Les difficultés rencontrées dans ces études expérimentales viennent, entre autres, des aspects mutli-échelles et multi-physiques, qui ont motivé le développement de nouvelles stratégies numériques, qui nous sont ainsi apparues comme une alternative intéressante pour surmonter les challenges en question. C’est dans cette philosophie que nous allons mettre en place une approche numérique de simulation directe capable de décrire les différents phénomènes qui interviennent dans un écoulement diphasique incompressible de deux fluides non-miscibles. Elle sera utilisée et développée sur la base de modules de programmation écrits dans le code de calcul "FUGU", qui est un



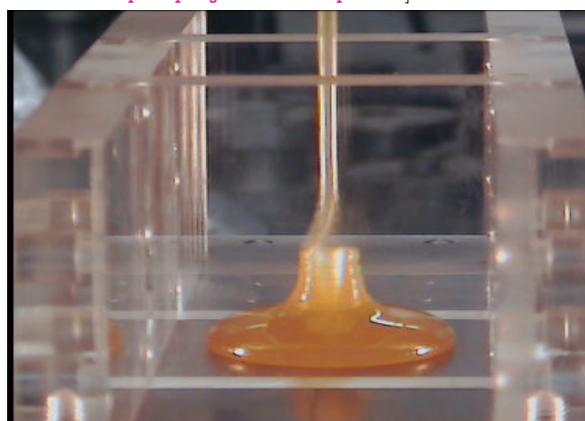
(a) L'impact d'un liquide sur une surface (Splash).
[\[http://physique.unice.fr/sem6/2012-2013/PagesWeb/PT/Splash/Html/Index.html\]](http://physique.unice.fr/sem6/2012-2013/PagesWeb/PT/Splash/Html/Index.html)



(b) Revêtement par projection de plasma.
[\[https://www.cleanpart.com/fr/Services-Produits/Revetements-par-projection-thermique/Revetement-par-projection-de-plasma\]](https://www.cleanpart.com/fr/Services-Produits/Revetements-par-projection-thermique/Revetement-par-projection-de-plasma)



(c) Formation du rouleau de déferlement d'une vague avec entraînement de sédiment (Laboratoire LEGI).
[\[https://images.cnrs.fr/en/photo/20150001_0066\]](https://images.cnrs.fr/en/photo/20150001_0066)

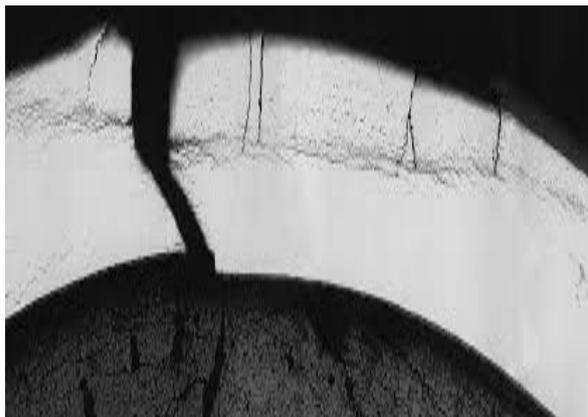


(d) Ecoulement d'un jet a faible nombre de Reynolds [129].

FIGURE 1.2 – Exemples d'écoulements à caractère diphasique à phase séparée, mettant en jeu différentes échelles.

logiciel de simulation de mécanique des fluides numérique écrit en langage Fortran et développé au laboratoire MSME. Au cours de ces dernières années, cette approche s'est imposée comme un outil incontournable d'investigation adopté par les mécaniciens des fluides pour compléter l'expérimentation afin de réduire les risques induits par certaines expériences et de répondre aux exigences fondamentales et applicatives. Elle se heurte toutefois à plusieurs difficultés qui doivent être surmontées, en termes non seulement de sensibilité des résultats numériques au choix du modèle, de la description précise de l'interface entre les deux phases et du maillage, mais également de la robustesse et la précision du solveur et d'algorithmes de résolution, permettant de simuler des problèmes diphasiques complexes à forts rapports de masses volumiques et de viscosités sans créer d'oscillations numériques ou diverger. En effet, la simulation à petite échelle de ces écoulements nécessite la prise en compte de la turbulence loin des interfaces d'une part, et le couplage entre les fluides en présence aux interfaces d'autre part (en particulier la vorticit   g  n  r  e    ces interfaces par le cisaillement et les rapports de masses volumiques et

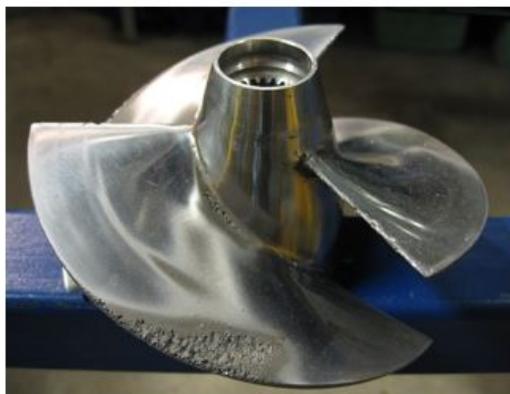
de viscosités). Or la structure mathématique des équations de Navier-Stokes incompressibles (système différentiel non-linéaire algébrique) rend sa résolution difficile en présence de forts rapports de masses volumiques et de viscosités en particulier qui détériorent le conditionnement du système et dont les distributions spatiales varient dans le temps. Ainsi, il faut pouvoir faire en sorte que ces solveurs et algorithmes d'analyse numérique s'exécutent sur des machines massivement parallèles, nécessitant une optimisation pour tirer profit de calculs intensifs répartis sur des milliers de processeurs.



(a) Comportement thermomécanique de la gaine du combustible [163].



(b) La rupture du barrage de Teton en 1976 aux États-Unis, image Wikipédia.



(c) La cavitation dans les hélices de bateaux. Ce phénomène s'accompagne de la formation de bulles de vapeur à l'intérieur du liquide dans les zones de basse pression qui se produisent dans les endroits où le liquide a été accéléré à des vitesses élevées.

[<https://vibrationdata.wordpress.com/2021/09/13/marine-propeller-cavitation/>]

FIGURE 1.3 – Exemples de dysfonctionnement des écoulements diphasiques.

Le problème révèle différents aspects. Tout d'abord, le choix du modèle décrivant le problème diphasique entre les deux phases. Ensuite, le suivi correct de l'interface en tenant compte des changements de topologies éventuels (coalescence, rupture). Puis, potentiellement, le développement d'une approche consistante de l'advection de la quantité du mouvement et la masse volumique à l'aide des schémas numériques d'ordres élevés en temps et en espace lorsqu'on fait face à des problèmes complexes à forts ratios de masses volumiques. Enfin, le dernier aspect s'ar-

ticule autour du traitement du couplage vitesse-pression nécessitant un solveur robuste et des algorithmes de résolution sur des serveurs HPC. Concernant le choix du modèle, il existe deux approches de modélisation numérique. Les modèles dits bi-fluides décrivent le mélange polyphasique à différents niveaux ([52, 99]). Dans leur version la plus complète, ces modèles prennent la forme de deux systèmes de Navier-Stokes, couplés par des termes sources supplémentaires qui représentent les échanges (masse, quantité de mouvement et énergie) entre les deux phases. Au contraire, le modèle 1-fluide consiste à modéliser le mélange polyphasique par un milieu fluide unique, mais dont les propriétés thermodynamiques sont des fonctions discontinues du temps et de l'espace. Il comprend également des termes sources additionnels (effets capillaires) ([172]). Il en résulte un système similaire aux équations de Navier–Stokes, complété par une équation scalaire d'advection qui décrit l'évolution de l'interface.

Quant à lui, le traitement numérique des interfaces (notamment le suivi des interfaces) est nécessaire pour modéliser les discontinuités des propriétés physiques à travers l'interface, pour chaque itération temporelle, en particulier en présence de grands rapports de masses volumiques et de viscosités, ainsi que la conservation de la masse. Il permettra aussi de traiter les différents aspects liés à la présence d'interfaces de forme complexe et de phénomènes capillaires. On trouve dans la littérature différentes approches pour simuler numériquement les interfaces et leurs évolutions au cours du temps. Parmi celles-ci, nous distinguons les approches eulérienne dites de front capturing (VOF, Level set,...) qui reconstruisent l'interface de manière implicite en utilisant une fonction caractéristique de phase, et celles lagrangiennes dites front tracking qui suivent explicitement l'interface. Dans la méthode VOF (Volume Of Fluid) introduite par Hirt et Nichols [96], un champ scalaire additionnel décrit la fraction du volume dans chaque maille. Le gradient de ce champ permet de reconstruire l'interface. En ce qui concerne la méthode Level-Set [132, 134], une description régulière de l'interface est utilisée, représentée par une iso-surface d'une fonction distance à l'interface. Par ailleurs, la méthode CLSVOF [166] qui implique un couplage entre les méthodes Level-Set et VOF, bénéficie des avantages de ces deux méthodes. Les méthodes de suivi de front telles que celles de Tryggvason et al [171] ou celles développées par S.Shin et D.Juric [164] reposent sur l'advection lagrangienne d'une chaîne de marqueurs reliés par des segments : la position de l'interface est ainsi représentée par ces marqueurs. Cette méthode a l'avantage de fournir une description explicite de la position de l'interface et procure des résultats précis puisqu'elle est décrite à une échelle plus petite que le maillage associé aux vitesses eulériennes (résolution sous-maille).

En outre, le traitement du couplage vitesse-pression a fait l'objet de nombreux travaux et est relativement bien maîtrisé en ce qui concerne les écoulements monophasiques. En revanche, dans le contexte de la modélisation des écoulements diphasiques à forts rapports de masses volumiques et de viscosités, des difficultés numériques apparaissent, liées à la présence de grandes déformations de l'interface et des sauts de propriétés matériels qui y sont associés. Elles conduisent à des systèmes linéaires mal conditionnés, ce qui rend la résolution beaucoup plus difficile. Pour

faire face à ce problème, de gros progrès ont été réalisés au cours des dernières décennies, et de nombreuses méthodes ont été proposées. Ces méthodes peuvent être classées en deux catégories. D'une part, les méthodes approchées dites segregated, approximent le système original en utilisant un splitting en temps, ce qui aboutit à deux équations découplées : l'une pour mettre à jour le champ de vitesse et l'autre le champ de pression. Cela soulève un certain nombre de problèmes, tels que l'apparition de couches limites numériques en raison du splitting en temps, ou la nécessité de définir des conditions aux limites de pression qui n'existent pas dans le problème de départ. Comme références, nous pouvons citer les méthodes de projection-correction introduites par Chorin [36] et Temam [167], la méthode de projection incrémentale de J-L.Guermond [87], la méthode de projection vectorielle de J-P.Caltagirone [32] et la méthode de projection scalaire cinématique (KSP) [34] introduite par Caltagirone et Vincent. D'autre part, les méthodes exactes dites monolithique, résolvent les deux champs (vitesse et pression) simultanément, préservant ainsi la consistance du système discrétisé avec les équations continues. Dans ces méthodes exactes, le problème persiste sous la forme de recherche d'un point-selle sur le couple vitesse-pression. La recherche de ce point selle peut être faite via l'utilisation d'une méthode de régularisation et lagrangien augmenté ou bien une méthode couplée (FC). Dans le cadre du lagrangien augmenté, un terme supplémentaire est ajouté à l'équation de conservation de quantité de mouvement pour pénaliser la contrainte d'incompressibilité. Nous pouvons évoquer à ce sujet la méthode du lagrangien augmenté standard (SAL) introduite par Fortin et Glowinski [71] pour des écoulements en régime de Stokes, la méthode du lagrangien augmenté adaptatif algébrique (AAAL) de Vincent et al [180, 185]. En ce qui concerne les méthodes couplées (FC), elles sont purement algébriques, utilisant ainsi des préconditionneurs spécifiques par blocs de vitesse et complément de Schur. Dans la littérature, un grand nombre de techniques de préconditionnement pour le complément de Schur ont été proposées. Dans le contexte de Stokes, Elman et Silvester [59] ont publié une approche pour les écoulements homogènes. Plus tard, un autre préconditionneur plus sophistiqué pour les écoulements hétérogènes a été développé par Bramble et al [26] en s'inspirant des travaux de Cahouet-Chabard [30]. Dans le cadre de Navier-Stokes à coefficient constant, nous pouvons trouver de nombreux développements pour construire une bonne estimation du complément de Schur. Par exemple, Elman [64] a suggéré une nouvelle technique appelée LSC (Least Squares Commutator). Plus récemment, Key et al [104] ont conçu un préconditionneur appelé PCD (Pressure Convection Diffusion). En ce qui concerne les écoulements diphasiques à fort rapports de masses volumiques et de viscosités, Bootland et ses collaborateurs [21] ont suggéré un préconditionnement efficace en étendant les préconditionneurs proposé par Kay et al [104] à la simulation des écoulements hétérogènes à coefficients variables. Pour le bloc de vitesse, il existe une variété de préconditionneurs. Parmi eux, les préconditionnements de Gauss incomplet de type ILU ou MILU [37, 107, 155, 194]. Ces préconditionnements ne sont pas facilement parallélisables car ils induisent une dépendance globale sur les inconnues, et nécessitent donc des échanges nombreux et répétés en échanges en parallélisation MPI. D'autres préconditionneurs originaux s'appuient sur les approches multigrilles, que ce soit géométrique [78, 124, 192] ou algébrique [131, 188, 191]. Ces techniques

utilisent généralement des résolutions inexactes, en effectuant une seule itération (cycle V, W ou F). Enfin, pour que ces techniques soient efficaces il est indispensable de choisir des composants multigrilles tels que : le lisseur à utiliser, le nombre d'itérations de pré-lissage et post-lissage ou encore le facteur de déraffinement de maillage entre les niveaux. D'autres préconditionneurs issus des techniques de décomposition de domaine permettent de résoudre des problèmes de plus en plus gros, en divisant le domaine d'étude en sous-domaines où la résolution est plus abordable. Dans ces techniques, nous trouvons deux grandes familles, les méthodes de décomposition de domaines avec recouvrement [146, 170] (type Schwarz), et celles sans recouvrement [118, 121] (type FETI et BDD). Au final, ces méthodes couplées sont peu utilisées en mécanique des fluides numérique pour éviter certaines difficultés liées au préconditionnement des solveurs itératifs mais elles peuvent bénéficier de récentes avancées dans le domaine des solveurs itératifs et des pré-conditionneurs, et sur leurs implémentation dans des bibliothèques parallèles open source (HYPRE [69], MUMPS [7, 8], LIS [108]). L'intérêt de ces méthodes exactes réside dans le fait qu'elles offrent une résolution très précise sans erreur de splitting temporel. Les stratégies qui en résultent, bien que plus difficiles à mettre en œuvre et plus coûteuses pour une itération temporelle de résolution, s'avèrent plus robustes et présentent de meilleures propriétés de convergence, avec peu de restrictions de CFL.

Ci-dessus les techniques de suivi d'interface et les méthodes de traitement du couplage vitesse-pression décrites offrent de très bons résultats, et permettent d'avoir une très bonne description des écoulements diphasiques incompressibles. Néanmoins, certains problèmes physiques tels que la nappe cisailée ou l'advection d'une sphère dense dans un fluide parfait, ont permis de mettre en avant l'incapacité du code de calcul basé sur ces méthodes à traiter des problèmes diphasiques à forts rapports de masses volumiques. En effet, dans le cas de l'advection d'une sphère dense dans un fluide parfait ou pour la nappe liquide cisailée, les simulations numériques deviennent instables et les vitesses sont mal évaluées, conduisant par la suite à une divergence du code de calcul. Pour contourner cette difficulté, une discrétisation consistante entre le transport de la quantité de mouvement et le transport de masse doit être considérée. Dans la littérature, différentes approches connues sous le nom de "Momentum conserving scheme", ont été mises en place afin d'améliorer la stabilité du code de calcul. L'idée génératrice de ces méthodes consiste à calculer les flux de la quantité de mouvement de façon cohérente avec les flux de la masse volumique. Autrement dit, une discontinuité de la quantité de mouvement est transportée avec la même vitesse que celle d'une discontinuité de la masse volumique, tout en utilisant les mêmes schémas de discrétisation spatio-temporelle. À ce sujet, nous pouvons évoquer différents auteurs qui ont proposé des nouvelles stratégies pour limiter ces instabilités numériques. Par exemple, Li et al. [115] pour la méthode MOF (moment of fluid), Vaudor et al. [176] et Zuzio et al. [201] pour une méthode CLS-VOF en adoptant l'approche de Rudman [151] sans la sous-grille, Le Chenadec et Pitsch [112] et Owkes et Desjardins [135] pour les méthodes VOF à interface raide (Sharp interface method), Fuster et al. [75], Arrufat et al. [10] et Sagar et al. [138]. pour les méthodes VOF conservatives, Nangia et al. [125] et Raessi et Pitsch [147] pour une méthode

Level-Set, pour n'en citer que quelques-uns.

Compte tenu des enjeux exposés précédemment, nous nous proposons dans cette thèse de développer un solveur tout-couplé parallèle 3D, capable de simuler des écoulements diphasiques à forts rapports de masses volumiques et viscosités sur plus d'un milliard de cellules. Le solveur sera évalué selon des critères à la fois théoriques (structure mathématique) et numériques (discrétisation, préconditionnement, parallélisation...). L'implémentation sera effectuée dans un code de simulation numérique directe "FUGU", développé au laboratoire MSME. Notons que le processus de développement du solveur est divisé en plusieurs étapes, dont la vérification et la validation. La vérification est guidée par la pertinence de la solution d'un modèle numérique comparée à des solutions analytiques dont les ordres de convergence spatio-temporels des schémas sont calculés. En contrepartie, la validation vise à étudier la capacité d'un code à retranscrire et caractériser des cas physiques en comparant les solutions numériques à des expériences, ou parfois à des benchmarks numériques. Par la suite, la parallélisation du code constitue un des enjeux majeurs de la thèse et qui sera déterminante pour réaliser des simulations massivement parallèles. Sur le plan technique, le développement et la validation du solveur se sont appuyés progressivement sur les machines de calcul de l'équipe et sur des projets GENCI mobilisant ainsi des millions d'heures de calcul sur différents supercalculateurs nationaux (Occigen de CINES, Irene de TGCC et JeanZay de l'IDRIS).

Plan du manuscrit

Le reste de ce mémoire est structuré comme suit :

- Le deuxième chapitre a pour objectifs de présenter les différentes approches pour la modélisation et la simulation d'un écoulement diphasique à phases séparées, et de faire un état de l'art des principales méthodes de suivi d'interface. La procédure retenue pour décrire l'advection de l'interface est ensuite analysée dans des cas-tests proposés dans la littérature.
- Le chapitre 3 est consacré à la résolution numérique du modèle à 1-fluide. Cela concerne le couplage entre la méthode de VOF et les équations de Navier-Stokes basées sur l'approximation du modèle à 1-fluide sur un maillage décallé. Le traitement numérique associé au couplage vitesse-pression est décrit dans son intégralité et deux familles de solveurs sont présentées et analysées.
- Le chapitre 4 évoque la mise en évidence d'un solveur original tout-couplé pour la simulation des équations de Navier-Stokes diphasiques. Dans un premier temps, nous allons nous attacher à la présentation des différents aspects algébriques du solveur (structure de matrice et formulation du point selle). Ensuite, nous détaillons la stratégie du préconditionnement par bloc utilisée afin d'améliorer la convergence du solveur. Puis, nous passons

quelques cas de validation du solveur tout-couplé sur des problèmes à la fois monophasiques et diphasiques. Enfin, nous nous focalisons sur des comparaisons qualitatives et quantitatives des différents solveurs dédiés au traitement du couplage vitesse-pression afin de tester leur robustesse lorsqu'ils font face à des problèmes raides.

- Le chapitre 5 aborde l'extention 3D parallèle du solveur tout-couplé afin de simuler des problèmes diphasiques incompressible sur des architectures massivement parallèles. Le solveur et les différentes méthodes numériques sont validés à l'aide de divers cas test, présentant des forts ratios de masses volumiques et de viscosités. Une étude de scalabilité forte est menée afin de mesurer les performances du solveur parallèle.
- Le chapitre 6 est consacré au couplage des équations de Navier-Stokes diphasiques avec le transport de la quantité de mouvement et le transport de la masse volumique, connu sous le nom de "Momentum conserving scheme". Les schémas de discrétisation spatio-temporelle sont détaillés. De nombreux cas-tests sont exposés afin de valider la méthode.
- Nous terminons ce manuscrit par les principales conclusions tirées des travaux de cette thèse et les perspectives à envisager pour la suite.

Chapitre 2

Différentes approches pour la modélisation et la simulation d'un écoulement diphasique

Sommaire

2.1	Équations de Navier-Stokes pour un écoulement monophasique incompressible (NS1)	42
2.1.1	L'équation de conservation de la masse	43
2.1.2	L'équation de bilan de la quantité de mouvement	44
2.2	Le modèle 2-fluides discontinu pour les interfaces résolues	45
2.3	Représentation des écoulements diphasique incompressibles par un modèle MUF	47
2.4	Filtrage des équations diphasiques	49
2.5	Etat de l'art sur les schémas d'advection de l'interface	51
2.6	Les méthodes lagrangiennes	52
2.6.1	MAC	52
2.6.2	Front tracking	52
2.7	Les méthodes Eulériennes	53
2.7.1	Les Méthodes Level Set	53
2.7.2	Les Méthodes Volume Of Fluid (VOF)	54
2.7.3	La méthode Smooth volume of fluid (SVOF)	61
2.8	Validations de transport	62
2.8.1	Translation d'une sphère dans un champ constant	63
2.8.2	Rotation d'une sphère dans un champ statique	63
2.8.3	Advection d'une sphère dans un champ avec cisaillement	66
2.9	Conclusion	66

Le processus physique que nous modélisons dans ce chapitre est la dynamique de deux fluides incompressibles (le volume est considéré comme étant constant quelle que soit la pression qu'il

subit), qui ont des propriétés physiques différentes. Nous nous intéressons en particulier aux écoulements diphasiques incompressibles de deux fluides newtoniens (le tenseur des contraintes dépend linéairement du taux de déformation) comme l'eau et l'air. De ce fait, les principales distinctions entre les deux fluides seront leurs masses volumiques et leurs viscosités. En outre, les fluides sont considérés comme étant isothermes, négligeant ainsi les variations de masses volumiques et de viscosités dues aux changements de température. De plus, on suppose aussi que les deux fluides sont non miscibles et l'interface qui les sépare est considérée comme une surface de discontinuité à travers de laquelle les propriétés physiques des deux fluides changent brusquement.

Dans la suite du chapitre, nous développons les équations de Navier-Stokes pour un écoulement monophasique incompressible. Puis, nous présentons les modèles utilisés pour la modélisation et la simulation des écoulements diphasiques, à savoir le modèle à deux fluides et le modèle à un fluide. Ces modèles sont présentés dans la section 2.1 et consistent en des systèmes d'équations différentielles partielles couplées. Ensuite, nous passons en revue des méthodes dédiées au suivi d'interface, en particulier la méthode VOF développée dans cette thèse. Enfin, des cas de validation sont présentés afin de mettre en évidence les capacités de la méthode de suivi d'interface retenue dans le cadre de ce travail. '

2.1 Équations de Navier-Stokes pour un écoulement monophasique incompressible (NS1)

Soit $\Omega(t)$ un volume matériel que l'on suit dans son mouvement, de contour $\partial\Omega(t)$ et de normale extérieure \bar{n}_{ext} à $\partial\Omega$ (Fig. 2.1). Les particules fluides qui se trouvent dans $\Omega(t = t_0)$ à l'instant $t = t_0$ sont transportées par le champ de vitesse locale \bar{u} . En un point géométrique donné \bar{x} de coordonnées cartésiennes (x, y, z) , nous notons $g(\bar{x}, t) = \rho, T, C, \dots$ (respectivement $\bar{g}(\bar{x}, t) = \rho\bar{u}, \bar{\nabla}P, \dots$) toute grandeur scalaire (respectivement vectorielle) attachée au système $\Omega(t)$ dépendant de la position $\bar{x}(x, y, z)$ et du temps t . La valeur totale prise par la grandeur $g(\bar{x}, t)$ (respectivement $\bar{g}(\bar{x}, t)$) s'écrit :

$$G(t) = \int_{\Omega(t)} g(\bar{x}, t) d\Omega \quad (2.1)$$

$$\bar{G}(t) = \int_{\Omega(t)} \bar{g}(\bar{x}, t) d\Omega \quad (2.2)$$

La dérivée particulaire de la grandeur $G(t)$ tient compte de la variation de la grandeur $g(\bar{x}, t)$ dans le temps, mais aussi de la variation par convection du flux entrant et sortant du volume $\Omega(t)$. Elle s'exprime à l'aide du théorème de Leibniz-Reynolds comme suit :

$$\frac{d}{dt}G(t) = \int_{\Omega(t)} \left(\frac{\partial g(\bar{x}, t)}{\partial t} + \nabla \cdot (g(\bar{x}, t)\bar{u}) \right) d\Omega \quad (2.3)$$

En revanche, la dérivée particulaire d'une intégrale de volume de la grandeur vectorielle \bar{G}

s'écrit :

$$\frac{d}{dt} \bar{G}(t) = \int_{\Omega(t)} \left(\frac{\partial \bar{g}(\bar{x}, t)}{\partial t} + \nabla \cdot (\bar{g}(\bar{x}, t) \otimes \bar{u}) \right) d\Omega \quad (2.4)$$

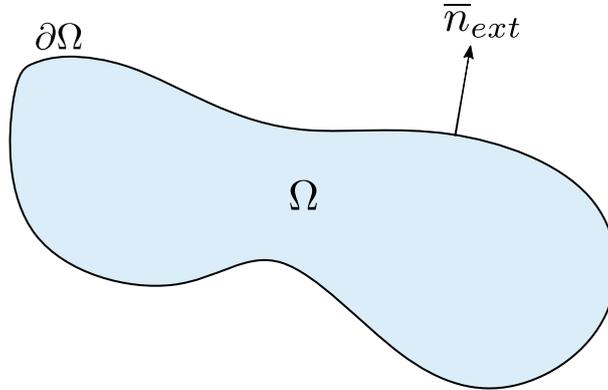


FIGURE 2.1 – Milieu continu de volume Ω limité par la surface $\partial\Omega$ munie d'une normale extérieure \bar{n}_{ext} .

2.1.1 L'équation de conservation de la masse

Tout d'abord, nous considérons le principe de conservation de masse. Soit ρ la masse volumique du fluide. Si nous prenons $g(t) = \rho$ dans l'équation (2.3), ce principe s'énonce comme suit :

$$\frac{d}{dt} \int_{\Omega(t)} \rho d\Omega = \int_{\Omega(t)} \left(\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{u}) \right) d\Omega = 0 \quad (2.5)$$

L'égalité est valable quelque soit le volume de contrôle $\Omega(t)$, et on en déduit l'équation locale de conservation de la masse sous forme conservative :

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{u}) = 0 \quad (2.6)$$

Comme $\nabla \cdot (\rho \bar{u}) = \rho \nabla \cdot \bar{u} + \bar{u} \nabla \rho$ et d'après la définition de la dérivée particulaire ($\frac{D\rho}{Dt} = \frac{\partial \rho}{\partial t} + \bar{u} \nabla \rho$), l'équation de conservation de masse (2.6) peut se réécrire comme suit :

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \bar{u} = 0, \quad (2.7)$$

Si le fluide est supposé incompressible et isotherme, alors $\rho = \text{constante}$. Cela conduit finalement à une deuxième forme de l'équation de continuité, nommée aussi l'équation de conservation de la masse,

$$\nabla \cdot \bar{u} = 0, \quad (2.8)$$

2.1.2 L'équation de bilan de la quantité de mouvement

Nous considérons maintenant le bilan de la quantité de mouvement. Pour une particule fluide, la quantité de mouvement de l'élément fluide contenu dans $\Omega(t)$ est donnée par :

$$\bar{M}(t) = \int_{\Omega(t)} \rho \bar{u} \, d\Omega(t) \quad (2.9)$$

La deuxième loi de Newton ou le principe fondamental de la dynamique (PFD), la variation temporelle de la quantité de mouvement $\bar{M}(t)$ est égale à la résultante des forces $\bar{F}(t)$ agissant sur $\Omega(t)$. Ces forces sont décomposées en une force volumique $\bar{F}_v(t)$ et une force surfacique $\bar{F}_s(t)$. Nous nous limitons au cas où la seule force volumique agissant sur le volume est la gravité :

$$\bar{F}_v(t) = \int_{\Omega(t)} \rho \bar{g} \, d\Omega \quad (2.10)$$

où \bar{g} est le vecteur de l'accélération gravitationnelle. La force surfacique $\bar{F}_s(t)$ est utilisée pour décrire les forces internes, c'est-à-dire les forces qu'un fluide exerce sur lui-même. Elles comprennent les forces de pression et la résultante des forces visqueuses exercées sur le contour $\partial\Omega(t)$ de $\Omega(t)$. Ces forces, dites contraintes, sont caractérisées par une densité superficielle \bar{t} ,

$$\bar{F}_s(t) = \int_{\partial\Omega(t)} \bar{t} \, \partial\Omega \quad (2.11)$$

appelée le vecteur contrainte. Ce vecteur est lié linéairement au tenseur des contraintes $\bar{\sigma}$ par la loi suivante (souvent appelée théorème de Cauchy), de sorte que la force surfacique $\bar{F}_s(t)$ puisse être représentée comme :

$$\bar{F}_s(t) = \int_{\partial\Omega(t)} \bar{\sigma} \cdot \bar{n}_{ext} \, \partial\Omega \quad (2.12)$$

On repart du principe fondamental de la dynamique (PFD) :

$$\frac{d\bar{M}(t)}{dt} = \bar{F}_v(t) + \bar{F}_s(t) \quad (2.13)$$

$$\frac{d\bar{M}(t)}{dt} = \int_{\Omega(t)} \rho \bar{g} \, d\Omega(t) + \int_{\partial\Omega} \bar{\sigma} \cdot \bar{n}_{ext} \, \partial\Omega \quad (2.14)$$

qui se réécrit en utilisant le théorème de la divergence :

$$\frac{d\bar{M}(t)}{dt} = \int_{\Omega(t)} \rho \bar{g} \, d\Omega(t) + \int_{\Omega(t)} \nabla \cdot \bar{\sigma} \, d\Omega(t) \quad (2.15)$$

En utilisant le théorème de Leibniz-Reynolds (2.4) dans le premier membre de (2.15), nous obtenons ainsi :

$$\frac{d}{dt} \bar{M}(t) dV = \int_{\Omega(t)} \left(\frac{\partial \rho \bar{u}}{\partial t} + \nabla \cdot (\rho \bar{u} \otimes \bar{u}) \right) d\Omega \quad (2.16)$$

$$= \int_{\Omega(t)} \rho \bar{g} d\Omega + \int_{\Omega(t)} \nabla \cdot \bar{\sigma} d\Omega(t) \quad (2.17)$$

Comme le domaine à intégrer est arbitraire et que les fonctions intégrées sont supposées continues, nous obtenons l'équation aux dérivées partielles suivante, qui représente l'équation locale du mouvement :

$$\left(\frac{\partial \rho \bar{u}}{\partial t} + \nabla \cdot (\rho \bar{u} \otimes \bar{u}) \right) = \rho \bar{g} + \nabla \cdot \bar{\sigma}. \quad (2.18)$$

En utilisant les hypothèses supplémentaires que le milieu est isotrope (c'est-à-dire que ses propriétés sont les mêmes dans toutes les directions de l'espace) et que l'action du tenseur de contrainte est indépendant du référentiel choisi (invariance sous un changement de repère), on peut montrer que le tenseur de contrainte doit avoir la forme suivante :

$$\bar{\sigma} = -p \bar{I} + \lambda \nabla \cdot \bar{u} + \bar{T} \quad (2.19)$$

où p est la pression, μ la viscosité dynamique, \bar{I} le tenseur identité et $\bar{T} = [\mu (\nabla \bar{u} + \nabla^T \bar{u})]$ le tenseur des taux de déformation. D'autres considérations physiques conduisent à des relations pour les paramètres μ et λ par exemple, $\mu > 0$ (pour un fluide visqueux) et $\lambda = -\frac{2}{3}\mu$. Dans le cas d'un fluide incompressible, c'est-à-dire $\nabla \cdot \bar{u} = 0$, l'expression du tenseur de contraintes se simplifie à :

$$\bar{\sigma} = -p \bar{I} + \bar{T} \quad (2.20)$$

On obtient ainsi les équations fondamentales de Navier-Stokes sous forme conservative pour un écoulement incompressible monophasique :

$$\nabla \cdot \bar{u} = 0, \quad (2.21)$$

$$\left(\frac{\partial \rho \bar{u}}{\partial t} + \nabla \cdot (\rho \bar{u} \otimes \bar{u}) \right) = \rho \bar{u} - \nabla p + \nabla \cdot \bar{T}. \quad (2.22)$$

2.2 Le modèle 2-fluides discontinu pour les interfaces résolues

L'aspect diphasique des écoulements n'est pas a priori pris en compte dans les équations du système (2.21) et (2.22) présentées auparavant. Dans la suite, nous allons rappeler quels sont les modèles existants pour décrire des écoulements diphasiques. Nous porterons une attention particulière sur ceux fréquemment employés. Parmi eux, on y trouve le modèle à deux fluides dans lequel les deux phases sont décrites séparément. Quant au modèle à un fluide, il remplace

le mélange diphasique par un seul fluide. Nous verrons plus en détails les différences entre ces deux modèles.

Équations de conservation de la masse et du mouvement

Nous considérons maintenant un écoulement diphasique, c'est-à-dire que $\Omega(t)$ contient un mélange de deux phases incompressibles non miscibles différentes (liquide-liquide ou liquide-gaz) qui peuvent se déplacer dans le temps et ont des propriétés physiques différentes ρ_k et μ_k ($k = 1, 2$). Pour chaque point et pour chaque instant $t \in [0, T]$, où T est un réel positif, $\Omega(t)$ est subdivisé en deux sous-domaines ouverts $\Omega_1(t)$ et $\Omega_2(t)$, chacun d'eux contenant respectivement l'une des phases. Ces phases sont séparées entre elles par une interface $\Gamma(t) = \bar{\Omega}_1(t) \cap \bar{\Omega}_2(t)$, qui correspond au bord entre les deux fluides (voir Fig. 2.2).

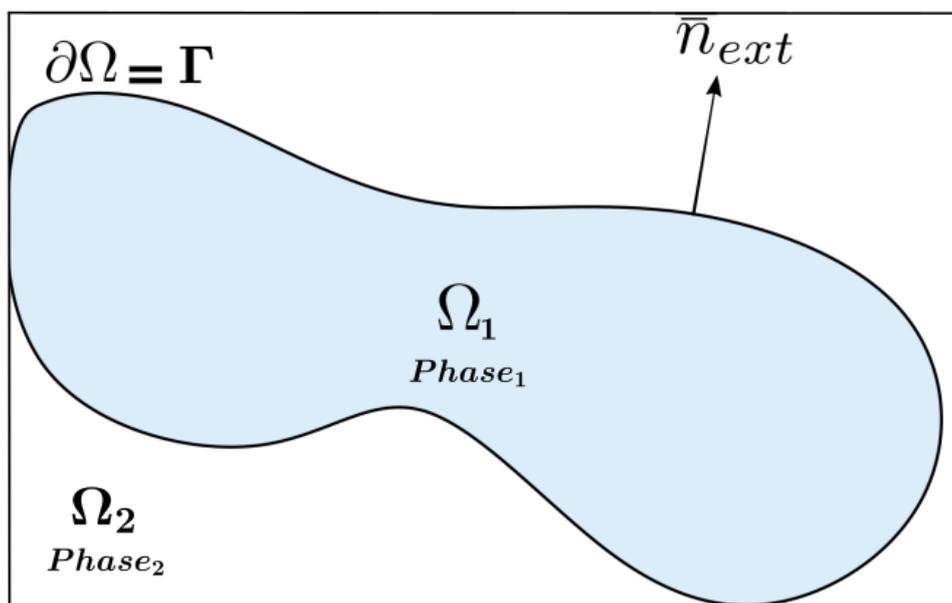


FIGURE 2.2 – Définition d'un problème diphasique modèle Ω constitué de deux phases Ω_1 , Ω_2 et d'une interface Γ .

Dans chacune des phases et à chaque instant $t \in [0, T]$, l'écoulement diphasique est décrit dans chaque sous-domaine $\Omega_k(t)$, $k = 1, 2$ par les équations de Navier-Stokes suivantes :

$$\forall t \in [0, T], \forall \bar{x} \in \Omega_k(t) \begin{cases} \nabla \cdot \bar{u}_k = 0 \\ \frac{\partial \rho_k \bar{u}_k}{\partial t} + \nabla \cdot (\rho_k \bar{u}_k \otimes \bar{u}_k) = \rho_k \bar{g} - \nabla p_k + \nabla \cdot \bar{\bar{T}}_k \end{cases} \quad (2.23)$$

avec \bar{u}_k et p_k respectivement la vitesse et la pression dans le domaine $\Omega_k(t)$, μ_k et ρ_k respectivement la viscosité et la masse volumique du fluide dans le domaine $\Omega_i(t)$.

Nous en arrivons maintenant aux conditions de saut traduisant les principes de conservation sur l'interface. Comme les phases sont visqueuses et en l'absence de changement de phase (pas de flux de matière à travers l'interface), on peut supposer que le champ de vitesse est continu à l'interface. La relation de saut dans les directions normale et tangentielle à l'interface s'écrit :

$$\begin{aligned}\bar{u}_1 \cdot \bar{n}_1 + \bar{u}_2 \cdot \bar{n}_2 &= 0 \\ \bar{u}_1 \cdot \bar{t}_1 + \bar{u}_2 \cdot \bar{t}_2 &= 0\end{aligned}\tag{2.24}$$

Par ailleurs, sans source de masse on sait que la conservation de la quantité de mouvement à la traversée d'une interface qui sépare deux fluides newtoniens s'écrit :

$$\left(-p_1 \bar{\bar{I}} + \bar{\bar{T}}_1\right) \cdot \bar{n}_1 + \left(-p_2 \bar{\bar{I}} + \bar{\bar{T}}_2\right) \cdot \bar{n}_2 = \sigma \varkappa\tag{2.25}$$

Dans les conditions de saut précédentes, \bar{n} est la normale à l'interface $\Gamma(t)$, \bar{t} le vecteur tangent à l'interface $\Gamma(t)$ σ le coefficient de tension superficielle ou encore tension de surface, \varkappa la courbure locale. Si l'on se réfère par exemple aux travaux de Drew et al. et Ishii et al. [52, 99], les équations de Navier-Stokes (2.23) associées aux relations de sauts (2.24) et (2.25) forment ce qu'on appelle le modèle 2-fluides pour les interfaces résolues.

2.3 Représentation des écoulements diphasique incompressibles par un modèle MUF

Pour pallier les difficultés liées au modèle à deux fluides, en l'occurrence la mise en place d'un maillage non structuré ou curviligne mobile, notre choix s'est porté vers l'utilisation du modèle à un fluide (MUF) de Kataoka [102]. Il consiste à établir une formulation de mélange dite mono fluide des écoulements diphasiques, faisant intervenir des variables spatialement filtrées. L'avantage de ce modèle est qu'il transforme un problème qui résout chaque fluide séparément en un formalisme qui permet de condenser la dynamique diphasique en une seule équation locale instantanée, faisant apparaître à la fois les bilans de chacune des deux phases et le bilan à l'interface. L'idée majeure de ce modèle selon Kataoka, consiste à introduire une nouvelle variable, l'indicatrice de phase $\chi_k(\bar{x}, t)$, afin de décrire localement l'appartenance de chaque point à une phase ou à une autre telle que :

$$\chi_k(\bar{x}, t) = \begin{cases} 1 & \text{si } \bar{x} \in \Omega_k(t) \\ 0 & \text{sinon} \end{cases}\tag{2.26}$$

Cette fonction indicatrice vérifie les propriétés suivantes :

$$\chi_k(\bar{x}, t) \chi_k(\bar{x}, t) = \chi_k(\bar{x}, t)\tag{2.27}$$

$$\chi_1(\bar{x}, t) \chi_2(\bar{x}, t) = 0\tag{2.28}$$

$$\sum_{k=1}^2 \chi_k(\bar{x}, t) = 1\tag{2.29}$$

$$\nabla \chi_k(\bar{x}, t) = -\bar{n}_k \delta_I \quad (2.30)$$

Par ailleurs, l'équation d'advection de χ_k s'écrit classiquement :

$$\frac{\partial \chi_k}{\partial t} + \bar{U}_I \cdot \nabla \chi_k = 0 \quad (2.31)$$

où \bar{U}_I est la vitesse de l'interface, \bar{n}_k la normale unitaire à l'interface sortante de la phase k et δ_I la distribution de Dirac attachée à l'interface. Ainsi, en multipliant l'équation de conservation de masse de la phase k par χ_k , nous obtenons :

$$\chi_k \nabla \cdot \bar{u}_k = 0 \quad (2.32)$$

d'où :

$$\nabla \cdot (\bar{u}_k \chi_k) - \bar{u}_k \cdot \nabla \chi_k = 0 \quad (2.33)$$

Ce qui est important, c'est que l'équation est maintenant valide sur tout le domaine, ce qui donne en utilisant l'équation (2.31) :

$$\frac{\partial \chi_k}{\partial t} + \nabla \cdot (\bar{u}_k \chi_k) + (\bar{U}_I - \bar{u}_k) \cdot \nabla \chi_k = 0, \quad \bar{x} \in \Omega(t) \quad (2.34)$$

Comme nous ne considérons que des écoulements isothermes et sans changement de phase, le champ de vitesse est continu à l'interface et, par conséquent, la vitesse de l'interface est égale à la vitesse de phase de part et d'autre de l'interface ($\bar{U}_I = \bar{u}_k$). Dans ce cas, l'équation d'advection de la fonction indicatrice χ_k devient :

$$\frac{\partial \chi_k}{\partial t} + \nabla \cdot (\bar{u}_k \chi_k) = 0 \quad (2.35)$$

Puisque ρ_k est supposée constante, l'équation précédente peut être multipliée par ρ_k de façon à obtenir une équation de bilan de masse sur tout le domaine pour la phase k :

$$\frac{\partial \rho_k \chi_k}{\partial t} + \nabla \cdot (\rho_k \bar{u}_k \chi_k) = 0 \quad (2.36)$$

En utilisant l'équation (2.33) et l'expression du gradient de χ_k (2.30). On obtient alors :

$$\nabla \cdot (\bar{u}_k \chi_k) + \bar{u}_k \cdot \bar{n}_k \delta_I = 0 \quad (2.37)$$

On procède de la même façon avec l'équation de conservation de la quantité de mouvement, on obtient l'équation de bilan de quantité de mouvement diphasique pour la phase k :

$$\chi_k \left(\frac{\partial \rho_k \bar{u}_k}{\partial t} + \nabla \cdot (\rho_k \bar{u}_k \otimes \bar{u}_k) \right) = \chi_k \rho_k \bar{g} - \chi_k \nabla p_k + \chi_k \nabla \cdot \bar{T}_k \quad (2.38)$$

En prenant soin de soustraire ce qu'on a ajouté grâce à l'équation de d'advection (2.35) et à l'expression du gradient de χ_k (2.30). On obtient l'équation de conservation de quantité de mouvement en formulation diphasique discontinue. Pour tout indice de phase k égal à 1 ou 2, il vient

$$\left(\frac{\partial \chi_k \rho_k \bar{u}_k}{\partial t} + \nabla \cdot (\chi_k \rho_k \bar{u}_k \otimes \bar{u}_k) \right) = \chi_k \rho_k \bar{g} - \nabla \chi_k p_k + \nabla \cdot (\chi_k \bar{T}_k) + (-p_k \bar{I} + \bar{T}_k) \cdot \bar{n}_k \delta_I \quad (2.39)$$

C'est ici que les descriptions à un ou deux fluides divergent : si on filtre mais sans faire la somme, on a le modèle à deux fluides d'Ishhii et al. [52, 99]. Maintenant, nous allons procéder à la sommation des équations (2.36) (2.37) et (2.39) sur les deux phases en tenant compte des bilans aux interfaces (2.24) et (2.25). On obtient :

$$\frac{\partial}{\partial t} \left(\sum_{k=1}^2 \rho_k \chi_k \right) + \nabla \cdot \left(\sum_{k=1}^2 \rho_k \bar{u}_k \chi_k \right) = 0 \quad (2.40)$$

$$\nabla \cdot \left(\sum_{k=1}^2 \bar{u}_k \chi_k \right) + \sum_{k=1}^2 \bar{u}_k \cdot \bar{n}_k \delta_I = 0 \quad (2.41)$$

$$\frac{\partial}{\partial t} \left(\sum_{k=1}^2 \chi_k \rho_k \bar{u}_k \right) + \nabla \cdot \left(\sum_{k=1}^2 \chi_k \rho_k \bar{u}_k \otimes \bar{u}_k \right) = \sum_{k=1}^2 \chi_k \rho_k \bar{g} - \nabla \sum_{k=1}^2 \chi_k p_k + \nabla \cdot \sum_{k=1}^2 \chi_k \bar{T}_k + \sigma \chi \bar{n} \delta_I \quad (2.42)$$

2.4 Filtrage des équations diphasiques

La résolution numérique des équations du milieu diphasique nécessite leur discrétisation en des points discrets du maillage. De ce fait, une épaisseur numérique de l'interface est générée à travers de laquelle les grandeurs physiques varient de façon continue, afin d'accéder à l'évolution du mélange des deux phases. Cela revient à faire un filtrage spatial pour séparer les grandes échelles des petites échelles (l'interface) de l'écoulement diphasique, analogiquement à la Simulation des Grandes Echelles (LES) [157]. Pour mettre en place ce filtrage spatial, nous définissons ainsi un noyau \mathcal{F} , $\forall \bar{x}$ et \bar{x}' :

$$\int_V \mathcal{F}(\bar{x} - \bar{x}') d\bar{x}' = 1 \quad (2.43)$$

où V est le volume caractéristique du filtre (généralement de l'ordre de la taille de la cellule). Notons que l'intérêt de l'application du filtre \mathcal{F} est de se débarrasser des relations de saut, et de passer d'une vision discontinue à une vision continue dans laquelle les phases sont vues via une concentration appelée fraction volumique notée $C_k(\bar{x}, t)$. Cette dernière représente le taux d'occupation de la phase k autour du point \bar{x} à l'instant t . Elle est définie comme étant le résultat

du filtrage de la fonction indicatrice $\chi_k(\bar{x}, t)$:

$$\overline{\chi_k} = C_k(\bar{x}, t) = \int_V \mathcal{F}(\bar{x} - \bar{x}') \chi_k(\bar{x}', t) d\bar{x}' \quad (2.44)$$

Connaissant la fraction volumique C_k de la phase k , on peut définir pour chaque grandeur physique $\Phi_k(\bar{x}, t)$ sa moyenne locale $\widehat{\Phi}_k(\bar{x}, t)$, donnée par la moyenne de Favre :

$$C_k \widehat{\Phi}_k = \overline{\chi_k \Phi_k} = \int_V \mathcal{F}(\bar{x} - \bar{x}') \chi_k(\bar{x}', t) \Phi_k(\bar{x}', t) d\bar{x}' \quad (2.45)$$

Le filtrage du système composé des équations (2.41), (2.40) et (2.42) avec l'hypothèse de commutation entre les opérations de filtrage et de dérivation conduit à :

$$\nabla \cdot \left(\sum_{k=1}^2 \widehat{u}_k C_k \right) + \sum_{k=1}^2 \overline{u_k \cdot \bar{n}_k \delta_I} = 0 \quad (2.46)$$

$$\frac{\partial}{\partial t} \left(\sum_{k=1}^2 \widehat{\rho}_k C_k \right) + \nabla \cdot \left(\sum_{k=1}^2 \widehat{\rho}_k \widehat{u}_k C_k \right) = 0 \quad (2.47)$$

$$\begin{aligned} \frac{\partial}{\partial t} \left(\sum_{k=1}^2 \widehat{\rho}_k \widehat{u}_k C_k \right) + \nabla \cdot \left(\sum_{k=1}^2 \widehat{\rho}_k \widehat{u}_k \otimes \widehat{u}_k C_k \right) &= \sum_{k=1}^2 C_k \widehat{\rho}_k \bar{g} - \nabla \cdot \sum_{k=1}^2 \widehat{p}_k C_k + \\ &\quad \nabla \cdot \sum_{k=1}^2 C_k \widehat{\widehat{T}}_k + \overline{\sigma \varkappa \bar{n} \delta_I} \end{aligned} \quad (2.48)$$

Nous définissons également les moyennes des grandeurs filtrées comme suit :

$$\bar{u} = \sum_{k=1}^2 C_k \widehat{u}_k \quad (2.49)$$

$$\bar{p} = \sum_{k=1}^2 C_k \widehat{p}_k \quad (2.50)$$

$$\bar{\rho} = \sum_{k=1}^2 C_k \widehat{\rho}_k \quad (2.51)$$

$$\overline{\rho u} = \sum_{k=1}^2 C_k \widehat{\rho}_k \widehat{u}_k \quad (2.52)$$

$$\overline{\widehat{T}} = \sum_{k=1}^2 C_k \widehat{\widehat{T}}_k \quad (2.53)$$

Considérons les définitions précédentes les équations (2.46) (2.47) et (2.48) peuvent s'écrire comme suit :

$$\nabla \cdot \bar{\bar{u}} + \sum_{k=1}^2 \overline{\bar{u}_k \cdot \bar{n}_k \delta_I} = 0 \quad (2.54)$$

$$\frac{\partial \bar{\rho}}{\partial t} + \nabla \cdot (\bar{\rho} \bar{u}) = 0 \quad (2.55)$$

$$\frac{\partial}{\partial t} (\overline{\rho \bar{u}}) + \nabla \cdot (\overline{\rho \bar{u} \otimes \bar{u}}) = \bar{\rho} \bar{g} - \nabla \bar{p} + \nabla \cdot \overline{\bar{T}} + \overline{\sigma \varkappa \bar{n} \delta_I} \quad (2.56)$$

En se servant des travaux de Girault et al.[79], nous trouvons la relation suivante :

$$\bar{\bar{u}} = \bar{\rho} \bar{u} + C_1 C_2 (\widehat{\rho}_1 - \widehat{\rho}_2) (\widehat{u}_1 - \widehat{u}_2) \quad (2.57)$$

De même selon [79], nous avons :

$$\overline{\rho \bar{u} \otimes \bar{u}} = \bar{\rho} \bar{u} \otimes \bar{u} + \frac{C_1 C_2 \widehat{\rho}_1 \widehat{\rho}_2}{\bar{p}} (\widehat{u}_1 - \widehat{u}_2) (\widehat{u}_1 - \widehat{u}_2) \quad (2.58)$$

Pour la suite, le symbole $\bar{\bar{\cdot}}$ sera omis pour alléger l'écriture. Ainsi, en supposant l'absence de glissement entre les phases $(\widehat{u}_1 - \widehat{u}_2)$, nous obtenons finalement la formulation 1-fluide pour des écoulements diphasiques

$$\nabla \cdot \bar{u} = 0 \quad (2.59)$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{u}) = 0 \quad (2.60)$$

$$\frac{\partial \rho \bar{u}}{\partial t} + \nabla \cdot (\rho \bar{u} \otimes \bar{u}) = \rho \bar{g} - \nabla p + \nabla \cdot \overline{\bar{T}} + \sigma \varkappa \bar{n} \delta_I \quad (2.61)$$

2.5 Etat de l'art sur les schémas d'advection de l'interface

Dans la section précédente, nous venons de dégager le système d'équations à résoudre pour simuler les écoulements multiphasiques incompressibles à phases séparées. Dans le modèle à un fluide, nous avons constaté l'apparition d'un nouveau terme source de tension de surface. Par ailleurs, la masse volumique et la viscosité ne sont plus constantes. En effet, dans les mailles coupées par l'interface, la masse volumique et la viscosité varient tandis qu'ailleurs elles sont constantes. En utilisant cette formulation, on doit connaître la position de l'interface afin de calculer les propriétés physiques utilisées dans l'équation (2.61) à partir des propriétés individuelles de chaque fluide, et de calculer les quantités interfaciales telles que la normale à l'interface et la courbure interfaciale. Ceci est réalisé en utilisant une fonction de marqueur qui identifiera chaque fluide dans le système. Dans la section suivante, nous passons brièvement en revue les différentes possibilités d'utiliser une fonction marqueur afin de calculer toutes les quantités liées

à l'advection de masse et à la déformation de l'interface, puis nous expliquons les méthodes mises en œuvre dans le code FUGU utilisé dans ce travail.

Comme nous l'avons indiqué précédemment, nous utilisons une fonction marqueur afin de décrire et suivre l'évolution de l'interface. Dans la littérature, il existe deux classes de méthodes pour assurer le suivi spatio-temporel de l'interface, différenciées entre les méthodes lagrangiennes et les méthodes dites eulériennes. Dans les méthodes lagrangiennes, au lieu d'utiliser directement une fonction marqueur, on utilise des marqueurs, qui sont transportés de manière lagrangienne par le champ de vitesse local comme suit :

$$\frac{d\bar{X}_i}{dt} = \bar{u}(t, \bar{X}_i(t)) \quad (2.62)$$

où \bar{X}_i la position du marqueur i . La fonction marqueur est ensuite reconstruite à partir des positions de marqueurs. Dans les méthodes eulériennes, la fonction marqueur est directement advectée. Une telle fonction peut être, par exemple, une fonction discrète qui prend la valeur 1 dans un fluide et 0 dans un autre et une valeur intermédiaire dans les mailles coupées par l'interface, de sorte qu'elle peut être utilisée comme fonction de pondération dans une interpolation linéaire ou harmonique utilisée pour calculer les propriétés physiques dans la formulation à un fluide. Par exemple, si $M(\bar{x}, t)$ est une fonction marqueur qui est égale à 1 dans le fluide 1 et à 0 dans le fluide 2, la masse volumique dans l'équation (2.60) peut être calculée comme $\rho(\bar{x}, t) = M(\bar{x}, t)\rho_1 + (1 - M(\bar{x}, t))\rho_2$.

2.6 Les méthodes lagrangiennes

2.6.1 MAC

Une des premières méthodes utilisant une formulation lagrangienne est la méthode Marker-And-Cell (MAC) développée par Harlow et Welch [92]. Dans cette méthode, le fluide est identifié par des particules marqueurs (Fig. 2.3) distribuées dans toute la région du fluide, et les équations de conservation sont résolues sur une grille régulière qui couvre à la fois la partie remplie de fluide et la partie vide du domaine. L'un de ses avantages remarquables est la conservation exacte de la masse. Cependant, plusieurs inconvénients se présentent : tout d'abord, elle se restreint aux écoulements bidimensionnels à surface libre, où seul un fluide est en mouvement. Ensuite, elle est très coûteuse en temps de calcul, puisqu'elle nécessite un nombre important de marqueurs par rapport au nombre de points résolus dans le maillage eulérien. De plus, l'interface n'est pas précisément définie, de sorte que le calcul des caractéristiques topologiques comme la normale ou la courbure est complexe.

2.6.2 Front tracking

Dans les méthodes de Front-Tracking, telle que celle développée par Unverdi et Tryggvason [171], l'interface est remplie de points marqueurs (Fig. 2.4) qui sont ensuite advectés en utilisant la

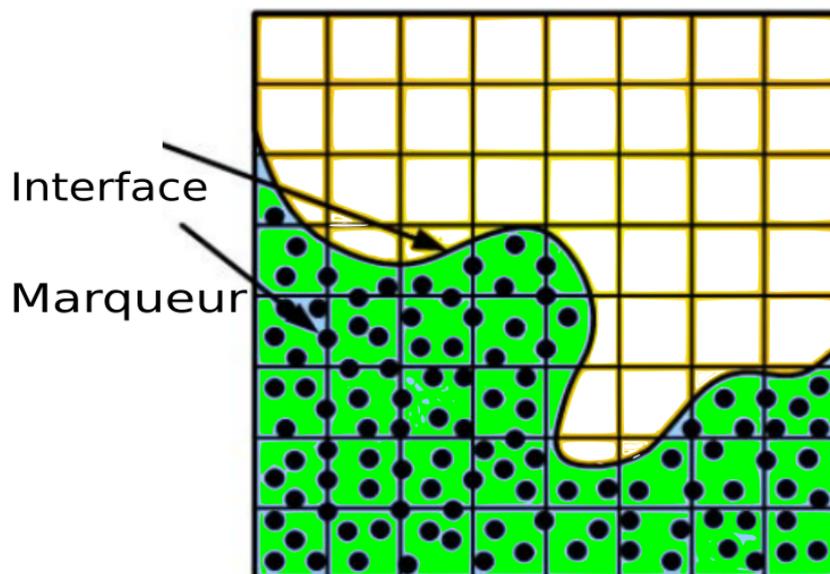


FIGURE 2.3 – Schéma de la méthode MAC.

vitesse du fluide interpolée à l'emplacement des points marqueurs (Eq. 2.62). L'intérêt de ces méthodes, comme l'ont démontré par ailleurs Popinet et Zaleski [143], est qu'elles permettent de suivre une interface avec une description sous-maille de celle-ci, puisqu'elle repose sur un maillage lagrangien de marqueurs placés sur l'interface. Une fois que la position des marqueurs est connue, on peut calculer une fonction marqueur utilisée pour évaluer les propriétés physiques. De plus, la courbure de l'interface peut être calculée précisément si nécessaire pour inclure les forces de tension de surface, qui sont déterminantes pour appréhender la physique à l'interface des écoulements diphasiques. Ces forces de tension superficielle sont déterminées via la relation de Frenet appliquée aux tensions localisées aux points d'intersection entre le maillage de l'interface et les volumes de contrôle du maillage. Malgré les avantages présentés précédemment, des limites apparaissent à nouveau lorsque des topologies d'interface complexes se développent pendant la simulation, comme la coalescence et la rupture de la surface de l'interface. En outre, l'advection de l'interface ne garantit pas la conservation du volume qui est une propriété fondamentale pour les écoulements incompressibles. Rajoutons à ces inconvénients l'extension au cas tridimensionnel qui n'est pas immédiate pour ce qui est de la gestion du maillage.

2.7 Les méthodes Eulériennes

2.7.1 Les Méthodes Level Set

Dans les méthodes Level set [132, 134], la fonction marqueur est définie comme étant la distance signée entre les points \bar{x} de la grille et l'interface la plus proche,

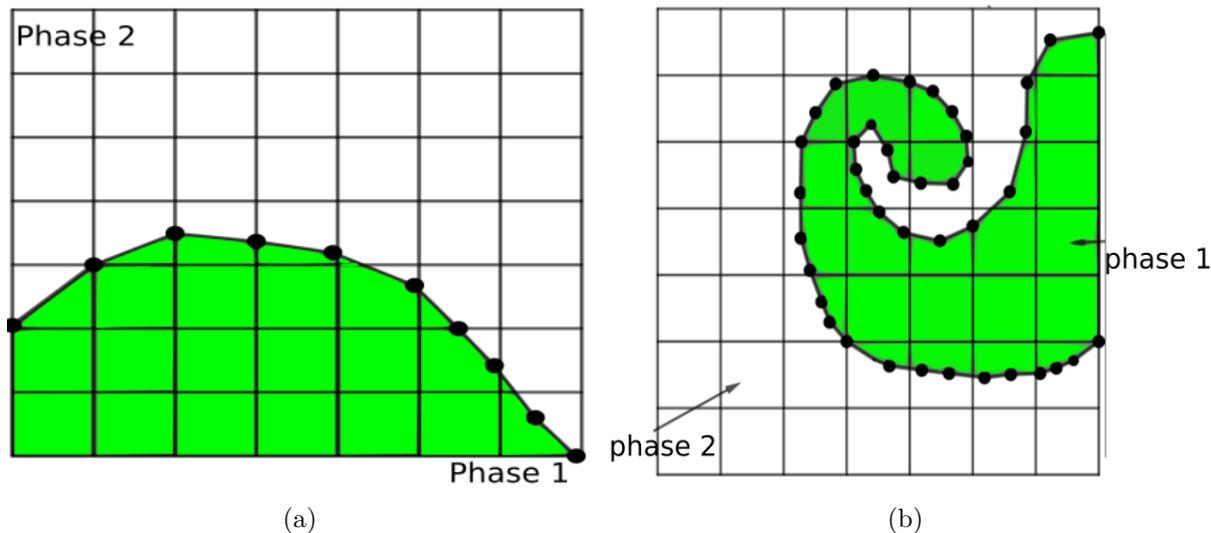


FIGURE 2.4 – Principe de la méthode Front-tracking : (a) topologie simple (b) topologie complexe

$$\phi(\bar{x}, t) = \pm \min_{\bar{y} \in \Gamma(t)} \|\bar{x} - \bar{y}\|_2 \quad (2.63)$$

où l'interface correspond à l'isocontour $\phi = 0$ (Fig. 2.5). Pour suivre l'évolution de l'interface, la fonction marqueur ϕ est directement advectée par le champ de vitesse \bar{u} suivant l'équation d'advection suivante :

$$\frac{\partial \phi}{\partial t} + \bar{u} \cdot \nabla \phi = 0 \quad (2.64)$$

La définition de la fonction distance signée implique que $\|\nabla \phi\|_2 = 1$, ce n'est pas le cas après advection par l'équation (2.64). Il serait donc nécessaire de rajouter une procédure de réinitialisation qui modifiera ϕ pour que la condition $\|\nabla \phi\|_2 = 1$ soit satisfaite. Néanmoins, ni l'étape d'advection, ni la procédure de réinitialisation ne garantissent la conservation de masse, ce qui constitue le principal inconvénient de la méthode Level Set. Pour améliorer les performances de cette méthode, de nombreuses évolutions ont été apportées. Par exemple, nous pouvons citer la méthode de Olsson et Kreiss [133] intitulée "Conservative Level-Set", la méthode proposée par Desjardins et al [44] nommée "Accurate Conservative Level-Set" et leurs éventuelles modifications. Enfin, la méthode CLSVOF [166] qui combine les méthodes Level-Set et VOF, tire profit des avantages de ces deux méthodes. En effet, la conservation de la masse est bien garantie, et les propriétés géométriques peuvent être facilement calculées.

2.7.2 Les Méthodes Volume Of Fluid (VOF)

Il s'agit de la première méthode basée sur une approche eulérienne à champ discontinu. Elle a été initialement introduite par Hirt et Nichols en 1981 [96] pour représenter un milieu diphasique par l'intermédiaire d'une variable scalaire représentant la fraction volumique d'un fluide sur chaque cellule. Cette fonction, appelée fonction couleur C , peut varier de 0 à 1, chacune

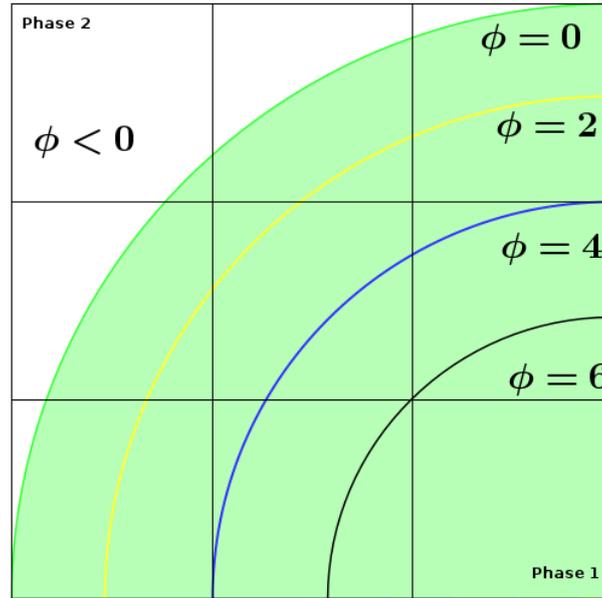


FIGURE 2.5 – Isosurfaces de la fonction Level-set ϕ . L'interface est située à $\phi = 0$. Le signe de ϕ permet de connaître la phase dans laquelle on se situe.

des deux valeurs extrêmes représentant la présence d'un seul des deux fluides dans la cellule. Si la valeur de la fraction volumique est comprise entre ces deux valeurs, la cellule en question contient un mélange et l'interface est présente.

La méthode est principalement constituée de deux étapes. La première est l'advection de la fonction de couleur, la seconde est la reconstruction de l'interface à partir de la fonction de couleur. La fonction couleur C est advectée ainsi par le champ de vitesse local \bar{u} et est la solution de l'équation d'advection suivante :

$$\frac{\partial C}{\partial t} + \bar{u} \cdot \nabla C = 0 \quad (2.65)$$

qui peut s'écrire sous forme conservative comme suit :

$$\frac{\partial C}{\partial t} + \nabla \cdot (\bar{u}C) = C \nabla \cdot \bar{u}. \quad (2.66)$$

2.7.2.1 Reconstruction géométrique de l'interface

La résolution numérique de l'équation (2.66) doit garantir la conservation exacte de la masse. Néanmoins, elle reste délicate et constitue un des enjeux majeurs de la simulation des écoulements diphasiques. En effet, comme la fonction couleur C est discontinue, la diffusion numérique peut par exemple induire un lissage de la discontinuité au cours du temps. Pour contourner ce problème, l'interface est reconstruite dans le but de calculer les flux convectifs de façon géométrique. Pour ce faire, différentes techniques ont été développées. La figure 2.6 présente un exemple

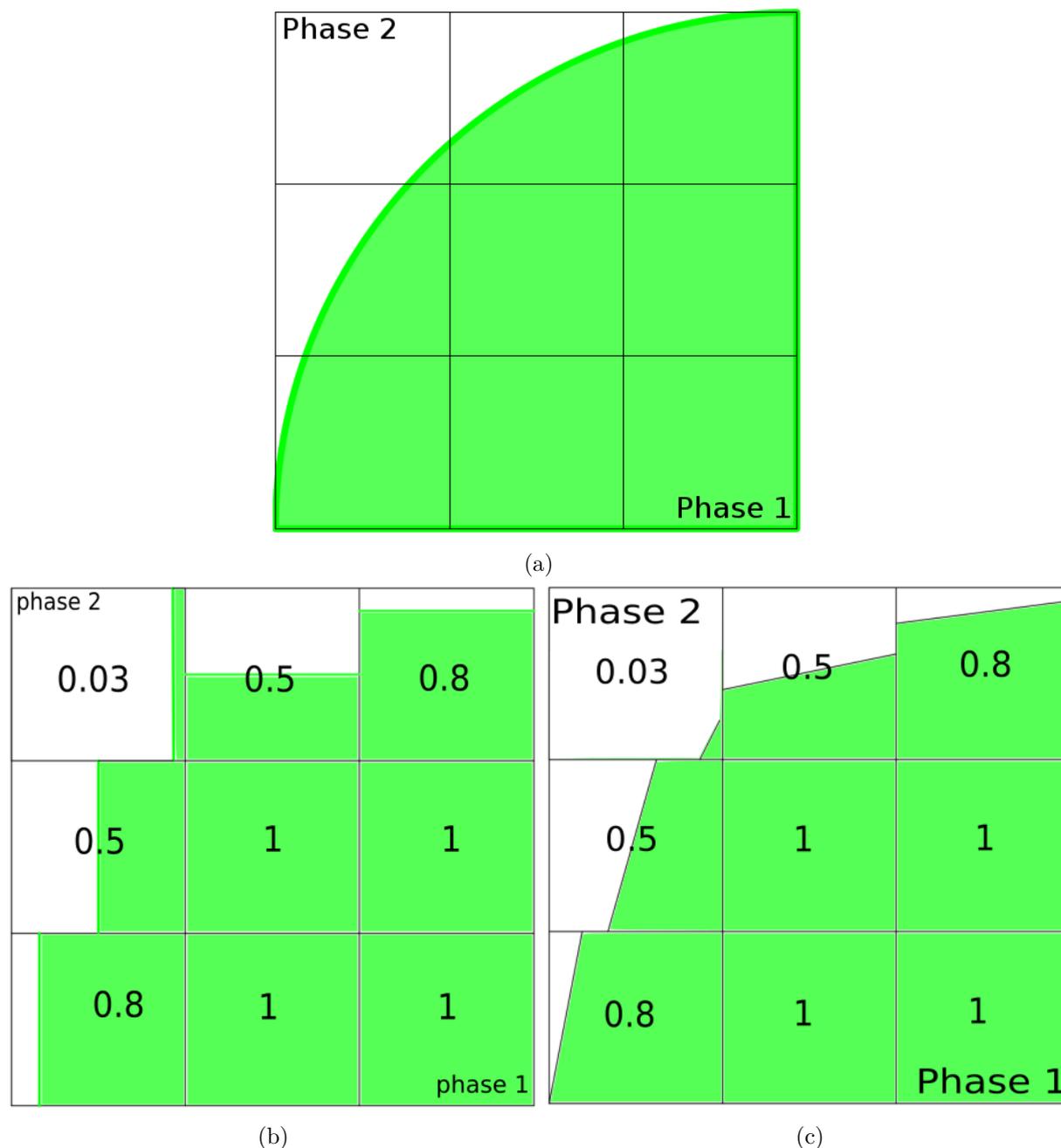


FIGURE 2.6 – Principe de la méthode VOF, (2.6a) Interface réelle et fonction couleur associée, (2.6b) classe de méthodes SLIC pour reconstruire l'interface, (2.6c) classe de méthodes PLIC pour reconstruire l'interface.

de la méthode VOF et de deux des principales techniques de reconstruction de l'interface. La méthode SLIC (Simple Line Interface Calculation) (voir Fig. 2.6b) introduite par Noh et Woodward [130], représente l'interface dans chaque cellule du maillage à l'aide de segments de droite alignés avec la grille eulerienne. La position et l'orientation des segments dépend des valeurs de C dans la cellule où sera reconstruite l'interface, ainsi que dans deux cellules voisines. Dans la pratique, cette méthode s'avère insuffisante pour décrire l'évolution de l'interface. En effet, des morceaux de l'interface se détachent d'une manière non physique, puisque les interfaces générées sont discontinues. Enfin on notera que la méthode ne considère qu'un bloc 3×1 autour de la

cellule d'intérêt, alors que l'interface reconstruite dans la cellule peut avoir une représentation différente par direction. Une amélioration directe de cet algorithme est la méthode PLIC (Piecewise Linear Interface Calculation) (voir Fig. 2.6c) introduite par DeBar [41] et Youngs [199]. Elle consiste à approximer l'interface par des segments de droite dans chaque cellule, qui peut être orientée arbitrairement par rapport à l'axe des coordonnées. L'orientation des segments est déterminée par la normale à l'interface, qui est trouvée en considérant la valeur de la couleur C à la fois dans la cellule considérée et dans les cellules adjacentes. D'autres méthodes de reconstruction d'interface utilisent toutes un bloc 3×3 de cellules voisines pour déterminer la normale en son centre mais elles sont au mieux du premier ordre. À cette classe appartient la méthode "center of mass" de Saltzman et Puckett [144], qui utilisent ce stencil de différentes manières pour obtenir le centre de masse du bloc, et par la suite la normale \bar{n} pointant du centre de la cellule vers ce centre de masse. Une autre manière pour déterminer la normale est d'utiliser directement le gradient de la fonction couleur comme dans les travaux de Parker et Young. Dans le cadre des méthodes de second ordre, on trouve les méthodes LVIRA (Least-squares Volume-of-fluid Interface Reconstruction Algorithm) et ELVIRA (E pour Efficient) développées respectivement par Puckett et al [145] et Pilliod [142]. Dans ces méthodes, une norme de l'erreur (norme $\|\cdot\|_2$) entre la fonction couleur et l'interface reconstruite est minimisée. Une autre approche a été présentée par Aulisa et al [13], où plusieurs interpolations linéaires (planes en trois dimensions) ont été ainsi sélectionnées par une procédure d'ajustement via une méthode des moindres carrés. Cette méthode est considérée comme étant du second ordre lorsqu'elle est appliquée de manière itérative.

2.7.2.2 Les algorithmes d'advection

Une fois que l'interface de chaque cellule est construite à partir de la fonction couleur C à l'instant t^n , un algorithme d'advection est utilisé pour assurer l'advection de l'interface. La position de l'interface et les fractions volumiques C au temps t^{n+1} sont ainsi déterminées. Dans la littérature, on trouve deux méthodes d'advection basées sur un splitting de directions : la méthode CIAM (Calcul d'Interface Affine par Morceaux) [117, 160] et l'algorithme de Weymouth et Yue [77], nous décrivons d'abord leurs aspects communs.

Après addition et soustraction d'un terme proportionnel à la divergence de la vitesse, l'équation (2.65) conduit à

$$\frac{\partial C}{\partial t} + \nabla \cdot (\bar{u}C) = C \nabla \cdot \bar{u} \quad (2.67)$$

La partie discrète de cette équation est obtenue en faisant une intégration spatio-temporelle de l'équation (2.67)

$$\frac{C_{i,j,k}^{n+1} - C_{i,j,k}^n}{\Delta t} \Delta \Omega = - \sum_{facesf} F_f^{(C)} + \Delta t \int_{\Omega} C \nabla \cdot \bar{u} \, d\Omega \quad (2.68)$$

où le premier terme du second membre est la somme des flux de la VOF sur la face de la cellule i, j, k . Bien évidemment, le terme de "compression" du côté droit disparaît pour un écoulement incompressible, mais il est utile de le conserver en raison de son utilité dans le transport par splitting de directions. Dans l'équation précédente, les flux $F_f^{(C)}$ de la VOF, seront calculés via des reconstructions géométriques à cause de la discontinuité de la fonction couleur C . Ils s'écrivent comme suit :

$$F_f^{(C)} = \Delta t \int_{\Omega} \bar{u} \cdot \bar{n}_f C \, dS \quad (2.69)$$

Avec \bar{n}_f la normale à la face f du volume de contrôle Ω .

Une fois l'interface de chaque cellule construite à partir de la fonction couleur C , l'équation (2.68) est utilisée pour mettre à jour les fractions de volume durant l'étape d'advection. Comme indiqué plus haut, les méthodes VOF standard utilisent un splitting temporel dans lequel le problème d'advection est divisé en plusieurs mises à jour séquentielles de la fraction volumique dans chacune des dimensions spatiales. Le splitting de l'équation (2.68) est formulé comme suit :

$$\frac{C_{i,j,k}^{n,1} - C_{i,j,k}^{n,0}}{\Delta t} \Delta \Omega = F_{i+1/2,j,k} - F_{i-1/2,k} + \int_{\Omega} C \frac{\partial u}{\partial x} \, d\Omega \quad (2.70)$$

$$\frac{C_{i,j,k}^{n,2} - C_{i,j,k}^{n,1}}{\Delta t} \Delta \Omega = G_{i,j+1/2,k} - G_{i,j-1/2,k} + \int_{\Omega} C \frac{\partial v}{\partial y} \, d\Omega \quad (2.71)$$

$$\frac{C_{i,j,k}^{n,n+1} - C_{i,j,k}^{n,2}}{\Delta t} \Delta \Omega = H_{i,j,k+1/2} - H_{i,j,k-1/2} + \int_{\Omega} C \frac{\partial w}{\partial z} \, d\Omega \quad (2.72)$$

où F , G et H sont les flux de la fonction couleur C advectés dans les première, deuxième et troisième directions ; c'est à dire que le fluide est initialement transporté le long de x , puis de y , et enfin (pour les écoulements 3D) de z . Pour alléger la notation, nous utilisons une nouvelle réécriture de (2.70)-(2.72) sous la forme suivante :

écrite de manière discrète comme suit :

$$\frac{\Delta C}{\Delta t} \Delta \Omega = \Delta_m F_m + \int_{\Omega} C \frac{\partial u_m}{\partial x_m} \, d\Omega \quad \text{Pour } m = 1, 2, 3; \quad (2.73)$$

Ici, m représente l'indice cartésien et $\Delta_m F_d$ décrit le flux net dans cette direction. Dans l'équation précédente, nous remettons à l'échelle les variables d'espace et de temps de sorte que le volume de la cellule et le pas de temps soient égaux à 1. Toute composante de vitesse u devient $u' = u\Delta t/h$ et la valeur $C_{i,j,k}$ est également la fraction volumique de fluide de référence dans la cellule i, j, k . Après ces transformations, l'équation (2.73) devient :

$$C_{i,j,k}^{n,l+1} = C_{i,j,k}^{n,l} - F_{m-}^{(C)} - F_{m+}^{(C)} + c_m \frac{\partial u_m}{\partial x_m} \quad (2.74)$$

avec l'exposant $l = 0, 1, 2$ qui est l'indice de sous-étape, i.e. $C_{i,j,k}^{n,0} = C_{i,j,k}^n$ et $C_{i,j,k}^{n,3} = C_{i,j,k}^{n+1}$. La face avec l'exposant $m-$ (resp $m+$) est la face "gauche" (resp droite) dans la direction m et

$F_m^{(C)}$ sont les flux de la VOF provenant d'une cellule à l'autre dans chaque direction. Ces flux sont ainsi estimés par des considérations géométriques à partir de la fonction couleur C et de la reconstruction de l'interface. L'expression $\frac{\partial u_m}{\partial x_m}$ est une approximation par volumes finis de la dérivée spatiale qui correspond à la m ième composante du vecteur vitesse le long de la direction m , et le coefficient de compression c_m représente une approximation de la fonction de couleur.

À chaque application de l'équation (2.74) correspond une sous-étape d'advection. Après chaque sous-étape d'advection, une nouvelle reconstruction de l'interface est effectuée à partir des fractions volumiques $C_{i,j,k}^{n,l+1}$ mises à jour en calculant la normale à l'interface \bar{n} . Une fois que la normale à l'interface \bar{n} est déterminée, un nouveau vecteur normal colinéaire noté \bar{m} et de norme L_1 unitaire est alors déduit de \bar{n} , de sorte que $|m_x| + |m_y| + |m_z| = 1$. Considérant le volume $V = C_{i,j,k}$ dans la cellule (i, j, k) la constante du plan α (déterminée comme dans [161]) est définie de façon à ce que le plan coupe exactement un volume V dans la cellule.

$$\bar{m} \cdot \bar{n} = \alpha \quad (2.75)$$

VOF CIAM (Calcul d'Interface Affine par Morceaux)

La méthode CIAM fait référence à une méthode VOF générique avec une étape de reconstruction linéaire par morceaux, comme cela est décrit dans la littérature. Toutefois l'interface est reconstruite dans chaque cellule. Celle-ci est advectée via un schéma lagrangien qui consiste à déplacer chaque point de cette interface par le champ de vitesse u' en plusieurs sous-étapes. Dans chaque sous-étape temporelle m de l'advection, le champ de vitesse est orienté dans la direction m , et ne dépend que de la composante $x_m \in [0, 1]$ de la variable spatiale normalisée. La vitesse sur l'interface est calculée par interpolation linéaire, à partir des vitesses euleriennes u_{m-} sur la face gauche et u_{m+} sur la face droite de la cellule de sorte que :

$$u'_m(x_m) = -u_{m-}(1 - x_m) + u_{m+}x_m \quad (2.76)$$

où l'origine de la coordonnée x_m est choisie sur la face gauche.

Ensuite, les points de l'interface sont advectés de manière lagrangienne pour décrire l'évolution de l'interface comme le montre l'équation :

$$\frac{dx_m}{dt} = u'_m \quad (2.77)$$

Afin de suivre l'évolution de l'interface, un schéma explicite d'ordre 1 est utilisé. Comme le temps a été remis à l'échelle, le pas de temps est égal à l'unité et il disparaît des équations. Ainsi, nous obtenons la position des points de l'interface à l'instant $t + \Delta t$ comme suit :

$$x_m^{n+1} = x_m^n + u'_m(x_m) = -u_{m-} + (1 + u_{m+} + u_{m-})x_m^n \quad (2.78)$$

Chaque advection donne la position des points x_m^{n+1} en fonction de la position d'origine x_m^n , et

comprime les distances le long de la direction m par un facteur $(1 + u_{m+1} + u_{m-})$. Il en résulte au plus trois segments de l'interface dans chaque cellule. Les volumes résultants sont représentés sur la figure 2.7 et sont intitulés V_1 , V_2 et V_3 . Finalement, la fonction couleur $C_{i,j,k}^{n+1}$ est obtenue en additionnant trois contributions.

$$C_{i,j,k}^{n+1} = V_1 + V_2 + V_3 \quad (2.79)$$

Comme il existe une corrélation entre l'advection des points x_m^{n+1} de l'interface et la définition (2.69) de $F_f^{(C)}$, alors $V_1 = -F_{m+}^{(C)}$, $V_3 = F_{m+}^{(C)}$ et $V_2 = (1 + u_{m+1} + u_{m-}) C_{i,j,k}^{n,l}$. L'expression finale de la sous-étape du splitting temporel s'écrit comme suit :

$$C_{i,j,k}^{n,l+1} = (1 + u_{m+1} + u_{m-}) C_{i,j,k}^{n,l} - F_{m+}^{(C)} - F_{m-}^{(C)} \quad (2.80)$$

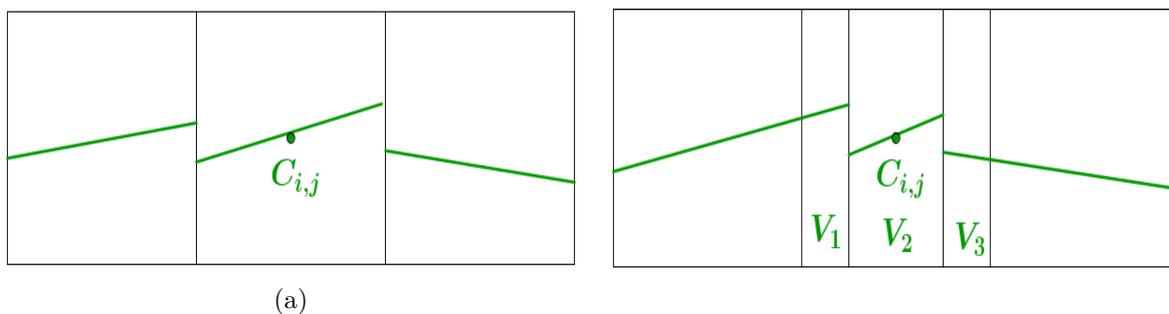


FIGURE 2.7 – Formation des volumes V_1 , V_2 and V_3

Ce qui montre que l'approximation de la dérivée $\frac{\partial u_m}{\partial x_m} = u_{m+} + u_{m-}$ et le coefficient de compression $c_m = C_{i,j,k}^{n,l}$. Nous notons que la séquence de trois sous-étapes lagrangiennes de l'équation (2.80) n'aboutit pas à la conservation du volume à la précision machine.

$$C_{i,j,k}^{n+1} - C_{i,j,k}^n = - \sum_{facesf} F_f^{(C)} + \sum_{m=1}^3 c_m (u_{m+1} + u_{m-}) \quad (2.81)$$

Bien que les termes de flux s'annulent lors de l'intégration sur les faces de chaque cellule, la somme des termes de compression ne s'annule pas puisque c_m change à chaque sous-étape l .

VOF de Weymouth et Yue

Pour pallier au problème de conservation de masse de la méthode VOF-CIAM, Weymouth et Yue ont proposé un nouvel algorithme qui préserve la masse à la précision machine. Dans cet algorithme, le coefficient de compression est indépendant de la direction m , de sorte que $c_m = c$, et il est défini comme suit :

$$c = H \left(C_{i,j,k}^n - 1/2 \right) \quad (2.82)$$

où H est une fonction de Heaviside, c'est-à-dire que $c = 0$ lorsque $C_{i,j,k}^n < 1/2$ et $c = 1$ pour

$C_{i,j,k}^n > 1/2$. Les flux $F_f^{(C)}$ dans l'équation (2.81) sont par ailleurs définis différemment via une procédure géométrique, comme étant la quantité de matière entrant ou sortant à travers les faces d'un volume de contrôle donné, comme le montre la figure 2.8. En utilisant ces définitions simples, le schéma de l'advection conserve le volume à la précision machine. En effet, la sommation des trois sous-étapes de l'équation (2.74) aboutit à

$$C_{i,j,k}^{n+1} - C_{i,j,k}^n = - \sum_{\text{faces } f} F_f^{(C)} + c \sum_{m=1}^3 (u_{m+} + u_{m-}) \quad (2.83)$$

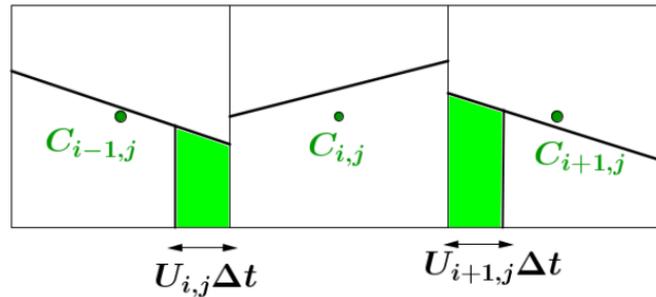


FIGURE 2.8 – Principe de la méthode de VOF Ye & Weymouth

Nous notons que l'algorithme précédent requiert une étape supplémentaire de clipping (borner C entre 0 et 1) pour éviter les conséquences des erreurs d'arrondi. Ce clipping appliqué à la fin de chaque étape d'advection directionnelle consiste à réinitialiser les valeurs de $C_{i,j,k}$ de sorte que $C_{i,j,k}$ est prise égale à 0 lorsque $C_{i,j,k} < \epsilon_c$ ou à 1 pour $C_{i,j,k} > 1 - \epsilon_c$. Quand la tension de surface est nulle, le choix de $\epsilon_c = 10^{-12}$ est satisfaisant. Sinon, $\epsilon_c = 10^{-8}$ donne des résultats plus stables en présence d'une interface plus lisse.

Avec ces principes simples des méthodes VOF, nous obtenons une procédure robuste et efficace, qui permet de gérer naturellement les changements de topologie de l'interface, les ruptures et connexions, ainsi que la conservation de masse. Toutefois, ces méthodes présentent des difficultés qui résident dans l'évaluation des grandeurs qui caractérisent l'interface, telles que la normale et la courbure surtout en 3D à cause des formes d'interface raides. Un autre défaut connu concerne la génération de brisures ou fragments numériques dès lors que les échelles des structures interfaciales sont de taille proche de celle de la maille.

2.7.3 La méthode Smooth volume of fluid (SVOF)

Les palliatifs aux inconvénients de la méthode VOF peuvent être contournés en utilisant une autre classe de méthodes permettant de régulariser la VOF. Les méthodes de profil d'interpolation sous contrainte (CIP) [158] consistent à résoudre une équation de transport pour la fonction de couleur. Basée sur le CIP, la fonction couleur et son gradient sont calculés pour aboutir à une fonction d'interpolation cubique. Cette méthode est connue pour sa faible diffusivité numérique et sa bonne stabilité. Cependant, son principal inconvénient réside dans l'épaisseur de l'interface qui n'est pas contrôlée. Une méthode VOF régularisée récente consiste à introduire

une fonction lisse de volume de fluide (SVOF) obtenue en résolvant une équation de diffusion à faible coût comme décrit dans [76, 90]. Cette procédure numérique présente à priori deux avantages. D'une part, la diffusion contrôlée de points d'interface à travers desquels les propriétés physiques varient fortement mais continuellement. D'autre part, comme les propriétés physiques sont continues, les propriétés de convergence pour le solveur deviennent meilleures, même en présence de grands rapports de viscosités et densités. En outre, l'application de la modélisation de type CSF [24] (Continuum Surface Tension) à SVOF améliore considérablement la représentation de l'approximation des forces de tension superficielle.

La fonction couleur lissée C_s (Smooth VOF) varie entre 0 et 1 comme C (obtenue via l'Eq. (2.83)), et dont la diffusion sur un pas de temps fictif $\Delta t = 1s$ est contrôlée autour de $C = 0.5$. Cette fonction est obtenue en procédant à une opération de lissage sur la fonction de couleur C par résolution de l'équation de Helmholtz suivante (le lecteur est invité à se reporter à [76, 90] pour plus d'informations),

$$C^{s,n+1} + \nabla \cdot D \nabla C^{s,n+1} = C^{n+1} \quad (2.84)$$

où $D = L_i \Delta h^2$ est le coefficient de diffusion qui est alors défini par le produit du volume de contrôle Δh^2 et d'un paramètre numérique L_i . On peut constater que ce paramètre L_i permet de faire en sorte que la fonction couleur lissée $C^{s,n+1}$ s'étende sur une distance $L_i = k \Delta h$ de part et d'autre de l'interface, où k est compris entre 0.2 et 2. On notera que l'opération de lissage décrite ci-dessus dans l'équation (2.84), revient à remplacer la discontinuité de l'interface par une zone d'épaisseur ajustable, à travers duquel la transition entre les deux phases a eu lieu. Par conséquent, cette méthode ne génère pas de déformation artificielle près de l'interface et fournit des stencils suffisamment larges pour mieux estimer la normale et la courbure de l'interface, comme dans les méthodes Level Set.

2.8 Validations de transport

Nous avons fait précédemment un état de l'art des différentes techniques de suivi d'interface, classées en deux catégories : lagrangiennes et eulériennes. Notre choix s'est porté sur l'utilisation des méthodes eulériennes, et particulièrement la méthode VOF de Yue et Weymouth.

Afin de montrer les capacités de la méthode retenue à décrire correctement l'interface, même lorsque celle-ci est fortement étirée et déformée, trois configurations classiques ont été choisies : la translation d'une sphère dans un champ constant, la rotation d'une sphère et la déformation d'une sphère dans un champ tournant. Dans chaque configuration, nous résolvons l'équation de transport suivante :

$$\frac{\partial C}{\partial t} + \nabla \cdot (\bar{u} C) = C \nabla \cdot \bar{u} \quad (2.85)$$

Pour évaluer les erreurs commises lors du transport, nous avons choisi deux types d'erreur. D'une part, l'erreur de conservation de masse intégrée sur le domaine de calcul. Elle est définie comme suit :

$$\epsilon_{mass} = \frac{\int_V c(\bar{x}, t) dV - \int_V c(\bar{x}, 0) dV}{\int_V c(\bar{x}, 0) dV} \quad (2.86)$$

D'autre part, l'erreur géométrique qui mesure la différence entre l'interface calculée et l'interface exacte :

$$\epsilon_{geo} = \int_V [c(\bar{x}, T) - c_{exacte}(\bar{x}, T)] dV \quad (2.87)$$

Pour chaque configuration, nous réalisons une étude de convergence en calculant l'ordre de précision numérique induit grâce à la relation suivante :

$$ordre = \frac{\log\left(\frac{\epsilon_{h1}}{\epsilon_{h2}}\right)}{\log\left(\frac{h1}{h2}\right)} \quad (2.88)$$

où ϵ_{h1} et ϵ_{h2} sont respectivement les erreurs associées aux maillages de pas h_1 et h_2 .

2.8.1 Translation d'une sphère dans un champ constant

Pour commencer, nous considérons la translation diagonale d'une sphère de rayon $r = 0.15m$, centrée au point $(0.5, 0.5, 0.5)$ dans un domaine cubique $\Omega = [0, 1]^3$. Le champ de vitesse utilisé est $\bar{u} = (1, 1, 1)$. Le pas de temps est constant et est fixé à $\Delta t = 10^{-3}s$. L'advection est arrêtée après un tour complet et donc lorsque $t = 1s$.

Comme présenté sur la figure 2.9, après un tour complet la sphère retrouve sa position initiale. Qualitativement, nous pouvons facilement vérifier que l'advection de la sphère est parfaitement calculée grâce au schéma de Weymouth & Yue. En parallèle, nous avons mené une étude de convergence en maillage pour évaluer la capacité du schéma à conserver la forme de l'interface et son volume. Pour ces simulations (voir Fig. 2.10), nous remarquons que l'algorithme conserve bien le volume à la précision machine. De plus, l'erreur géométrique sur la forme de la sphère diminue en fonction du maillage à l'ordre 1.

2.8.2 Rotation d'une sphère dans un champ statique

Nous nous intéressons maintenant à l'étude de la rotation d'une sphère dans un champ statique. Dans un domaine cubique de coté $L = 1m$, une sphère est initialisée, centrée au point $(0.5, 0.5, 0.75)$, avec un rayon de $0.2m$. Ce test est important car il ne privilégie aucune direction, et par la suite c'est un bon moyen pour vérifier la validité du schéma de transport.

Pour les simulations numériques, nous choisissons un pas de temps $\Delta t = 5 \cdot 10^{-4}s$ sur une grille 512^3 , et nous les adaptions pour les autres maillages afin de respecter la condition CFL. Ainsi le

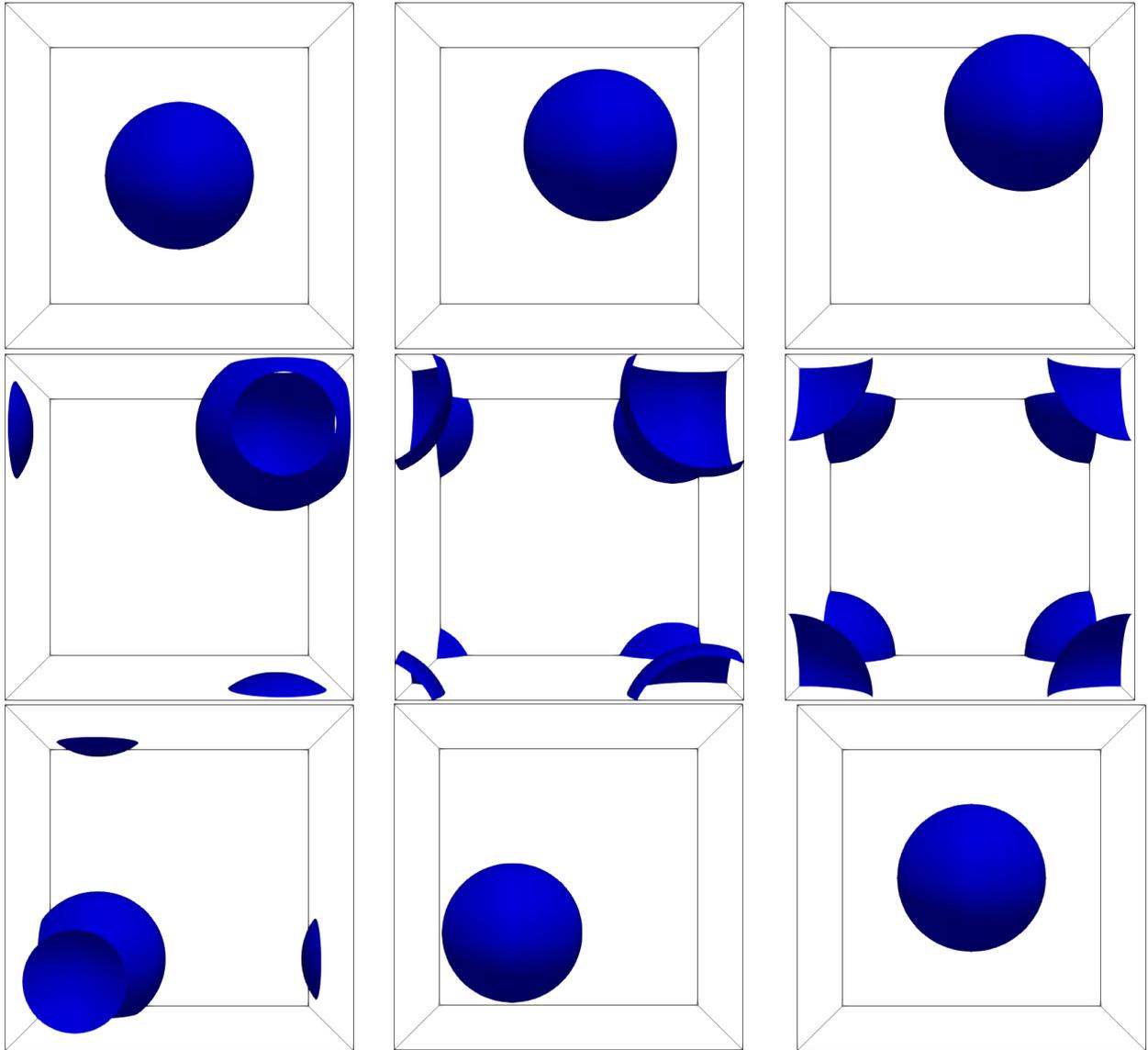


FIGURE 2.9 – Translation diagonale d’une sphère dans un champ constant. Maillage 512^3 : Évolution de l’interface sphérique.

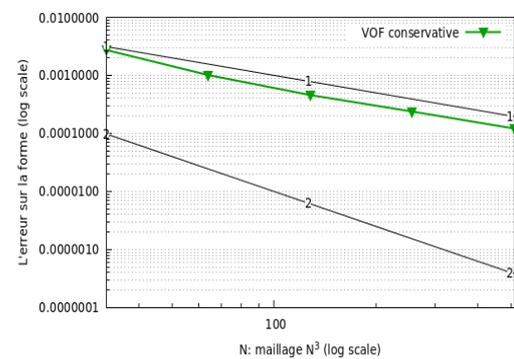
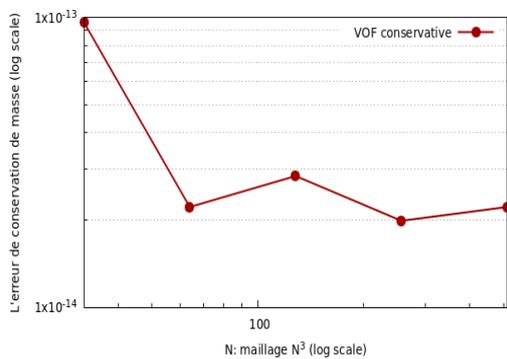


FIGURE 2.10 – Translation diagonale d’une sphère dans un champ constant. De gauche à droite, erreur sur la conservation du volume et sur la forme initiale de l’interface.

champ de vitesse est défini afin que la sphère tourne dans le sens horaire.

$$\begin{aligned} u(x, y, z) &= -\frac{\pi}{2} \left(y - \frac{L}{2} \right) \\ v(x, y, z) &= \frac{\pi}{2} \left(x - \frac{L}{2} \right) \\ w(x, y, z) &= 0 \end{aligned} \tag{2.89}$$

Comme l'illustre la figure 2.11, nous obtenons des résultats très satisfaisants, avec une conservation de masse à la précision machine, et un taux de convergence d'ordre 1 pour l'erreur sur la forme (voir Fig. 2.12). En effet, pendant le transport, l'interface reste circulaire et au bout du temps physique $t = 3s$ elle revient à sa place initiale (sans subir aucune déformation).

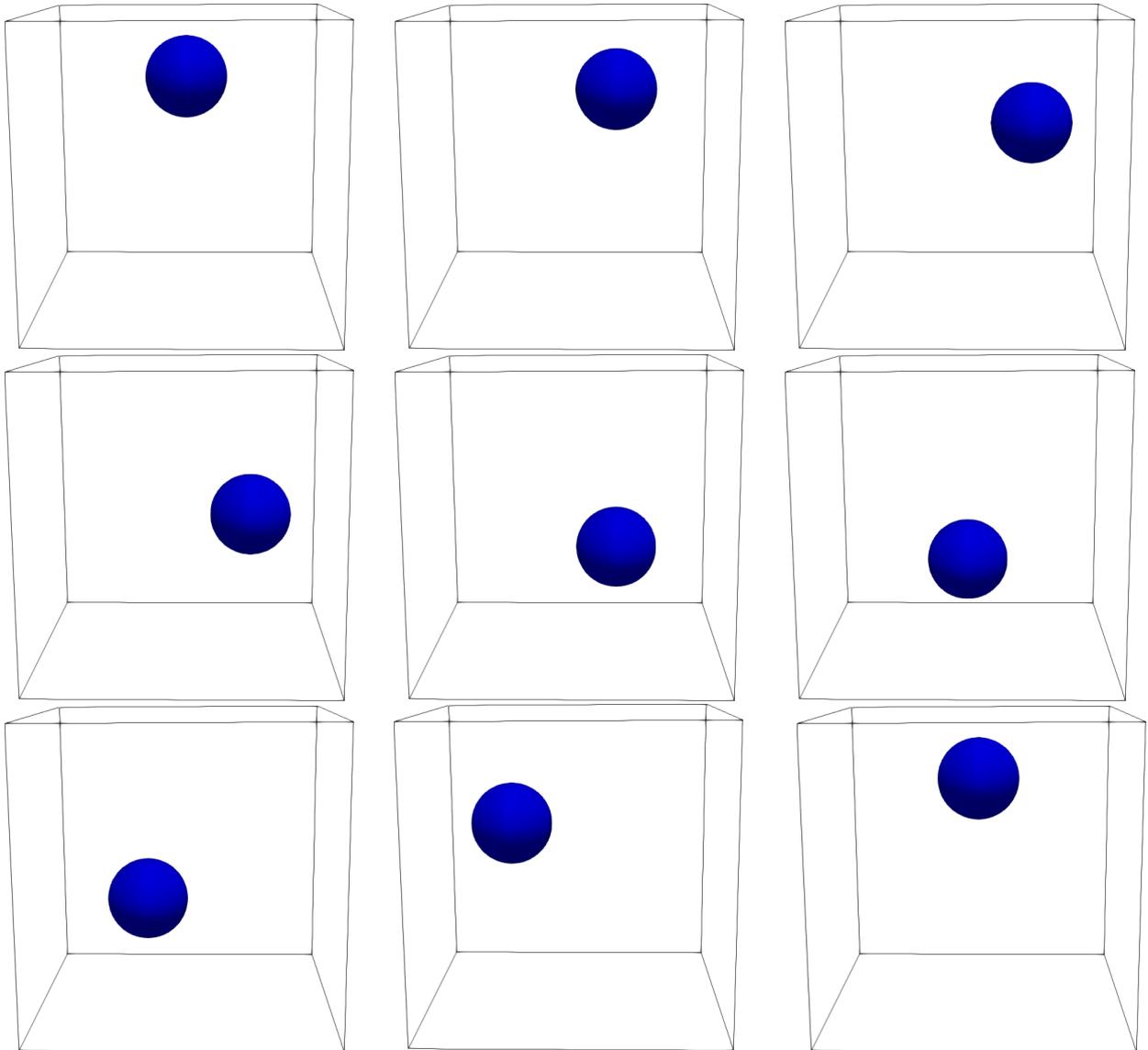


FIGURE 2.11 – Rotation d'une sphère dans un champ tournant sans déformation. Maillage 512^3 : Évolution de l'interface sphérique.

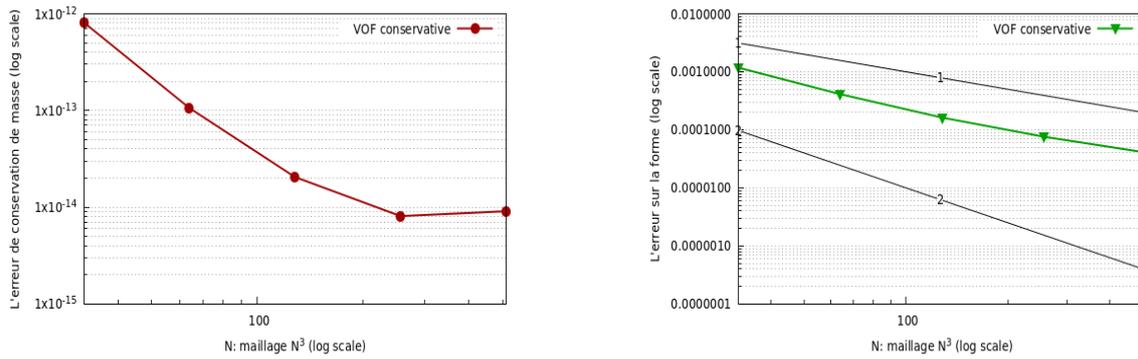


FIGURE 2.12 – Rotation d’une sphère dans un champ tournant sans déformation. De gauche à droite, erreur sur la conservation du volume et la forme de l’interface.

2.8.3 Advection d’une sphère dans un champ avec cisaillement

Ce cas test a été proposé par Enright et al. [65] à partir des travaux de Leveque [114]. Initialement, une goutte sphérique de rayon $r = 0.15$ est placée aux coordonnées $(0.35, 0.35, 0.35)$ dans une boîte périodique de côté $L = 1$. Dans cette configuration, l’interface est transportée avec le champ de vitesse suivant :

$$\begin{aligned}
 u(x, y, z) &= 2 \sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) \\
 v(x, y, z) &= -\sin(2\pi x) \sin^2(\pi y) \sin(2\pi z) \\
 w(x, y, z) &= -\sin(2\pi x) \sin(2\pi y) \sin^2(\pi z)
 \end{aligned}
 \tag{2.90}$$

Ce champ de vitesse est modulé par une fonction de temps $f(t)$ définie par $f(t) = \cos(\pi t/T)$, avec $T = 3s$ la période. Ainsi, le champ de vitesse est réversible, en d’autres termes, la sphère revient à sa position initiale en $t = T$.

La figure 2.13 présente les résultats que nous avons obtenus pour une interface sphérique avec un maillage de 512^3 . Nous pouvons voir que nous arrivons à des interfaces qui sont similaires à celles proposées dans les travaux d’Enright (voir Figs. 2.15). Nous avons également réalisé une étude de convergence en calculant l’ordre de précision spatial. Nous dressons les résultats de l’étude de convergence dans la figure 2.14. Nous pouvons constater une conservation de la masse à la précision machine quelque soit le maillage. En outre, un ordre 2 est obtenu sur l’erreur géométrique, ce qui valide notre méthode de transport.

2.9 Conclusion

Nous avons présenté dans ce chapitre les différentes approches pour la modélisation des écoulements diphasiques à phases séparées, à savoir le modèle à deux fluides et le modèle à un fluide. De plus, nous avons récapitulé les différentes techniques de suivi d’interface (les méthodes eulériennes et lagrangiennes). Notre choix s’est porté sur l’utilisation des méthodes eulériennes, et

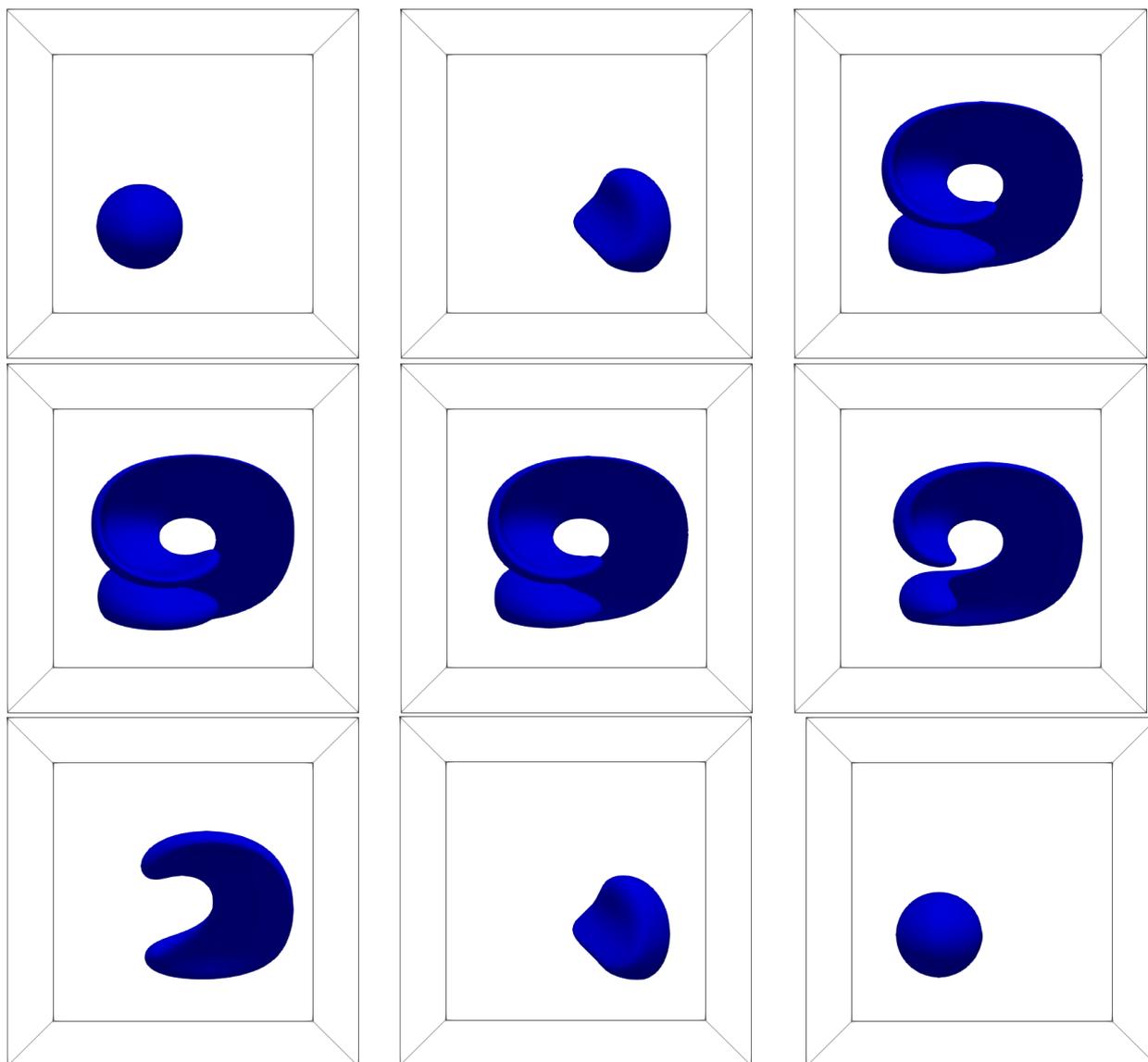


FIGURE 2.13 – Déformation d’une sphère dans un champ tournant avec cisaillement. Maillage 512^3 : Évolution de l’interface sphérique.

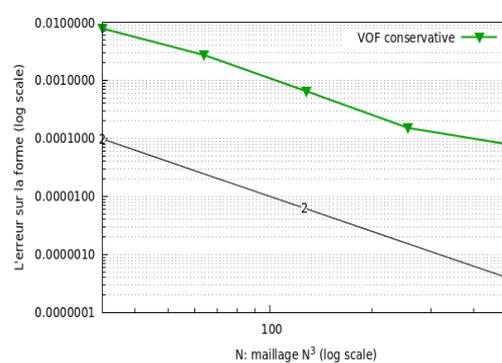
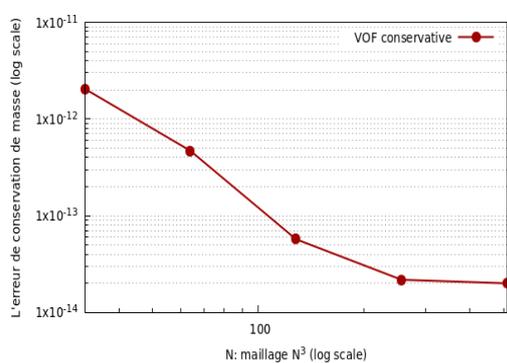


FIGURE 2.14 – Déformation d’une sphère dans un champ tournant avec cisaillement. De gauche à droite, erreur sur la conservation du volume et la forme de l’interface.

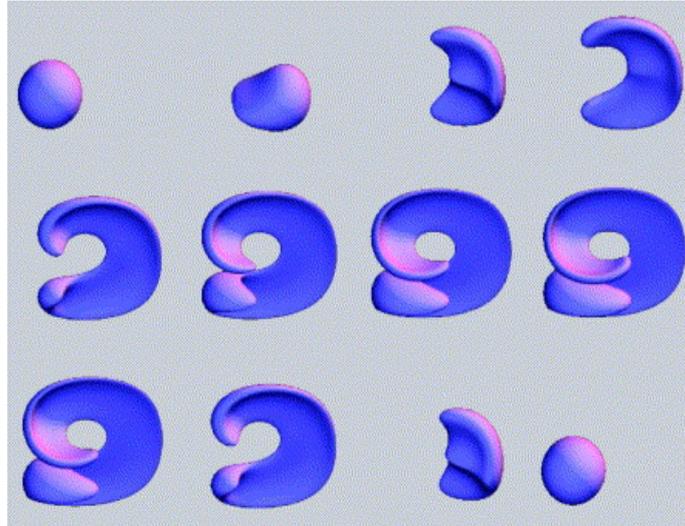


FIGURE 2.15 – Déformation d’une sphère dans un champ tournant avec cisaillement. Solution d’Enright et al. [65] sur un maillage 512^3 .

en particulier, la méthode VOF de Yue et Weymouth qui gère intrinsèquement le changement topologique de l’interface tout en assurant aussi la conservation du volume. À la fin du chapitre, nous avons pu valider la méthode VOF sur plusieurs cas-tests académiques. Nous avons obtenu un ordre 1 ou 2 de convergence spatiale pour la méthode et une précision machine sur la conservation du volume.

Chapitre 3

Résolution numérique du modèle à un fluide

Sommaire

3.1	Modèle d'écoulement diphasique utilisé pour la résolution des équations de Navier-Stokes	70
3.2	Intégration temporelle des équations de Navier-Stokes diphasiques	72
3.2.1	Formulation fonction de courant-vorticité $\psi-\omega$	74
3.2.2	La méthode de Prédiction-Correction	76
3.2.3	La méthode d'Uzawa	77
3.2.4	La méthode de décomposition d'opérateurs	79
3.2.5	La méthode du lagrangien augmenté standard	79
3.2.6	La méthode SIMPLE	82
3.3	Bilan des algorithmes de couplage vitesse-pression	84
3.4	Intégration spatiale des équations de Navier-Stokes diphasiques	85
3.4.1	Calcul du terme convectif	87
3.4.2	Calcul du terme Visqueux	88
3.4.3	Calcul du gradient de pression	90
3.4.4	Calcul de la divergence de la vitesse	91
3.4.5	Expression du terme capillaire	91
3.4.6	Traitement des conditions aux limites	93
3.5	Conclusion	94

Dans le chapitre 2 de ce manuscrit nous avons mis en œuvre les différentes procédures numériques retenues pour simuler un écoulement diphasique : le modèle à un fluide qui remplace le mélange diphasique par un seul fluide, et la méthode VOF qui décrit l'évolution de l'interface dans un champ de vitesse donné \bar{u} . Avec le champ de pression p , le champ de vitesse \bar{u} devient à présent une deuxième inconnue et représente la vitesse de l'écoulement qui doit satisfaire à la fois l'équation de conservation de quantité de mouvement et l'équation de conservation de la masse. Ainsi, le chapitre 3 est consacré aux techniques de résolution des équations de Navier-Stokes

diphasiques incompressibles. Dans un premier temps, nous rappelons le modèle d'écoulement diphasique utilisé pour la résolution des équations de Navier-Stokes. Ensuite, nous décrivons les différents algorithmes dédiés au traitement du couplage vitesse-pression, à savoir les méthodes découplées (la formulation fonction courant-vorticité et projection-correction) et les méthodes couplées (l'algorithme d'Uzawa, la décomposition d'opérateur, le lagrangien augmenté et l'algorithme SIMPLE). Enfin, nous détaillons les schémas de discrétisation spatiale des différents termes des équations de conservation, ainsi qu'une technique de pénalisation qui gère de façon naturelle les conditions aux limites.

3.1 Modèle d'écoulement diphasique utilisé pour la résolution des équations de Navier-Stokes

Dans le cadre de ce travail, la simulation numérique des écoulements diphasiques incompressibles non miscibles sera faite à l'aide d'un modèle de mélange appelé "modèle à un fluide" de Katoaka [102]. Les équations à résoudre sont : l'équation de conservation du volume ($\nabla \cdot \bar{u} = 0$) et le bilan de quantité de mouvement du mélange, augmenté par les effets capillaires qui agissent proche de l'interface. Nous négligerons les phénomènes de transfert de chaleur, et donc nous n'aurons pas à résoudre l'équation de l'énergie. Une équation supplémentaire sera également résolue : l'équation d'advection d'un champ scalaire aussi appelé la fonction couleur, utilisée pour caractériser la présence de l'interface, ainsi que les propriétés de l'écoulement au travers d'un fluide équivalent (masse volumique et viscosité). La fonction C , qui peut prendre des valeurs comprises entre 0 et 1, représente le taux de présence d'un des deux fluides dans un volume de contrôle. En effet, si cette fonction prend des valeurs différentes de 0 et de 1, le volume de contrôle en question contient un mélange des deux fluides, ce que signifie que l'interface y est présente. L'originalité du modèle à un fluide réside sur le fait qu'il tient implicitement compte des relations de saut à travers l'interface induites par la tension superficielle, ce qui revient aux équations classiques de Navier-Stokes à une phase dès que $C = 1$ ou $C = 0$ partout. Les équations du modèle à un fluide gouvernant le mouvement de l'écoulement diphasique s'écrivent comme suit :

$$\frac{\partial \rho \bar{u}}{\partial t} + \nabla \cdot (\rho \bar{u} \otimes \bar{u}) + \overline{B}_{penu} (f(\bar{u}) - \bar{u}_\infty) = -\nabla p + \nabla \cdot \overline{\overline{T}} + \rho \bar{g} + \overline{F}_s, \quad (3.1)$$

$$\nabla \cdot \bar{u} = 0, \quad (3.2)$$

$$\frac{\partial C}{\partial t} + \nabla \cdot (\bar{u} C) = C \nabla \cdot \bar{u}, \quad (3.3)$$

L'équation (5.2a) décrit la conservation de la quantité de mouvement, l'équation (5.2b) traduit la conservation de masse (les deux phases sont immiscibles et incompressibles) et l'équation (5.2c) représente l'advection de la fonction de couleur C . Dans les équations précédentes, \bar{u} est le champ

de vitesse, p le champ de pression, t le temps, ρ et μ la masse volumique et la viscosité du fluide de mélange, calculées via des moyennes arithmétiques, géométriques ou discontinues, à partir de ρ_1 , ρ_2 , μ_1 et μ_2 qui sont les masses volumiques et les viscosités des fluides 1 et 2 respectivement. En outre, $\overline{\overline{T}}$ représente le tenseur des taux de déformations pour un fluide Newtonien [32], tandis que $\overline{\overline{B}}_{penu}$ est un tenseur d'ordre 2 pour spécifier les conditions aux limites [137], dont les composantes diagonales tendent vers l'infini le long de la frontière et sont identiquement nulles à l'intérieur du domaine de fluide. En outre, $\overline{\overline{F}}_s$ est le terme capillaire agissant sur l'interface, modélisé dans cette étude par le modèle CSF de Brackbill [24]. La masse volumique et la viscosité du mélange peuvent être définies selon une moyenne arithmétique :

$$\rho(C) = \rho_1 C + (1 - C)\rho_2 \quad (3.4)$$

$$\mu(C) = \mu_1 C + (1 - C)\mu_2. \quad (3.5)$$

La moyenne harmonique peut aussi être utilisée, elle s'écrit :

$$\rho(C) = \frac{\rho_1 \rho_2}{\rho_2 C + (1 - C)\rho_1}, \quad (3.6)$$

$$\mu(C) = \frac{\mu_1 \mu_2}{\mu_2 C + (1 - C)\mu_1}. \quad (3.7)$$

D'autres choix sont possibles pour calculer les lois de mélange du modèle 1 fluide, et correspondent à l'utilisation des moyennes discontinues comme suit :

$$si \ C \geq 0.5 \ \text{alors} \ \rho = \rho_1 \ \text{et} \ \mu = \mu_1 \quad (3.8)$$

$$si \ C < 0.5 \ \text{alors} \ \rho = \rho_2 \ \text{et} \ \mu = \mu_2 \quad (3.9)$$

où ρ_1 , ρ_2 , μ_1 et μ_2 sont les masses volumiques et les viscosités des fluides 1 et 2. Par principe du mélange, les propriétés physiques mentionnées ci-dessus sont constantes dans le temps par phase. C (et donc ρ et μ) ne varient dans l'espace qu'à proximité de l'interface, c'est-à-dire dans les cellules dans lesquelles les volumes de contrôle sont coupés par l'interface [76, 90].

En ce qui concerne le tenseur des taux de déformations $\overline{\overline{T}}$, Caltagirone et Vincent [32] ont proposé une nouvelle technique de second ordre, intitulée Viscous Penalty Method. L'intérêt majeur de cette formulation est qu'elle permet de distinguer toutes les contributions physiques reflétant les effets de la compressibilité, de l'élongation, du cisaillement et de la rotation. Le fractionnement de ces contributions permet alors d'agir différemment sur chaque terme pour imposer fortement la contrainte associée. Après une décomposition appropriée, nous supposons que le tenseur des taux de déformation est donné par :

$$\bar{\bar{T}} = \kappa \begin{pmatrix} \frac{\partial u}{\partial x} & 0 \\ 0 & \frac{\partial v}{\partial y} \end{pmatrix} + \zeta \begin{pmatrix} 0 & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & 0 \end{pmatrix} - \eta \begin{pmatrix} 0 & \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} & 0 \end{pmatrix} \quad (3.10)$$

avec $\kappa = 2\mu$ la viscosité d'élongation, $\zeta = 2\mu$ la viscosité de cisaillement, et $\eta = \mu$ la viscosité de rotation.

Pour traiter la force capillaire \bar{F}_s définie dans l'équation (5.2a), nous utilisons la méthode de la force continue CFS (Continuum Surface Force) introduite par Brackbill et al. [24]. Dans cette méthode, nous procédons à une opération de lissage de la fonction couleur (comme cela est décrit dans la subsection (2.7.3)) afin de régulariser sa valeur à l'interface. Basé sur CFS, la force capillaire s'écrit comme suit :

$$\bar{F}_s = -\sigma \kappa(\bar{x}) \nabla C_s(\bar{x}) = -\sigma \nabla \cdot \bar{n}(\bar{x}) \nabla C_s(\bar{x}) \quad (3.11)$$

où σ est le coefficient de tension de surface supposé constant ici, \bar{x} étant un point quelconque de la région fictive remplaçant l'interface réelle, \bar{n} est la normale à l'interface et $\kappa(\bar{x})$ la courbure locale.

En considérant que les quantités ρ et μ sont localement constantes, et en tenant compte de l'équation de transport de la masse volumique $\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \bar{u} = 0$, les équations de conservation (5.2a-5.2b) peuvent être réécrites :

$$\rho \left[\frac{\partial \bar{u}}{\partial t} + \nabla \cdot (\bar{u} \otimes \bar{u}) \right] + \bar{B}_{penu} \cdot (f(\bar{u}) - \bar{u}_\infty) = -\nabla p + \nabla \cdot \bar{\bar{T}} + \rho \bar{g} + \bar{F}_s, \quad (3.12)$$

$$\nabla \cdot \bar{u} = 0, \quad (3.13)$$

3.2 Intégration temporelle des équations de Navier-Stokes diphasiques

Les problèmes que nous étudions étant des problèmes instationnaires, le pas de temps Δt est donc un paramètre numérique très important pour la qualité de la solution numérique. Il est donc nécessaire de choisir un schéma d'intégration en temps, avant que les équations de Navier-Stokes incompressibles (eq. 3.12-3.13) soient résolues. Cette intégration temporelle est réalisée par un schéma d'Euler d'ordre 1 ou 2, et les termes convectifs sont linéarisés en utilisant l'extrapolation d'Adams-Bashforth. Le système semi-discret qui en résulte s'écrit d'une manière généralisée comme suit :

$$\begin{aligned} \frac{\rho^{n+1}}{\Delta t} (\alpha \bar{u}^{n+1} - \beta \bar{u}^n + \gamma \bar{u}^{n-1}) + \rho^{n+1} \nabla \cdot \left((2\bar{u}^n - \bar{u}^{n-1}) \otimes \bar{u}^{n+1} \right) \\ + \bar{B}_{penu} \cdot (f(\bar{u}^{n+1}) - \bar{u}_\infty) = -\nabla p^{n+1} + \nabla \cdot \bar{T}^{n+1} + \rho^{n+1} \bar{g} + \bar{F}_s, \end{aligned} \quad (3.14)$$

$$\nabla \cdot \bar{u}^{n+1} = 0, \quad (3.15)$$

Pour finir, les valeurs des paramètres α , β et γ dépendent du schéma temporel utilisé :

- $\alpha = 1$, $\beta = -1$ et $\gamma = 0$: pour le schéma d'Euler d'ordre 1.
- $\alpha = 3/2$, $\beta = -2$ et $\gamma = 1/2$: pour le schéma de Gear ou Euler d'ordre 2.

Ainsi, après une discrétisation appropriée, le système générique qui en résulte est un système linéaire non symétrique de type point de selle qui prend la forme suivante :

$$\begin{pmatrix} F_u & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \bar{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \bar{f} \\ 0 \end{pmatrix} \quad (3.16)$$

On peut réécrire (3.16) sous la forme $Ax = b$ avec $F_u = M_u^{(\rho)} + N_u^{(\rho)} + L_u^{(\mu)}$, B_u représente la matrice de la divergence, B_p^T l'opérateur de gradient de pression discret, $M_u^{(\rho)}$ la matrice de masse agissant sur la vitesse, $L_u^{(\mu)}$ l'opérateur des contributions visqueuses, $N_u^{(\rho)}$ désigne la partie linéarisée du terme convectif et \bar{f} représente le second membre associé à la vitesse.

La résolution du système d'équations de Navier-Stokes (3.16) nécessite un traitement spécifique du couplage vitesse-pression, dans la mesure où l'équation de conservation de quantité de mouvement (3.14) doit être satisfaite en même temps que la contrainte d'incompressibilité (3.15). Une étude bibliographique a permis de constater l'existence d'un certain nombre d'approches par lesquelles les équations peuvent être formulées, ainsi qu'un grand nombre de méthodes de discrétisation pour relever le défi du couplage pression-vitesse. Les techniques populaires traitant ce couplage peuvent être classées en deux catégories : les méthodes découplées de type projection-correction de Chorin et Temam [36, 167] dites approchées, approximent le système original via un splitting d'opérateurs, il en résulte deux équations découplées : l'une pour mettre à jour le champ de vitesse et l'autre le champ de pression. Ces méthodes requièrent l'imposition de conditions aux limites sur la pression qui ne sont pas claires à établir et elles induisent la résolution d'une équation de type Poisson à coefficients variables qui est très délicate et difficile dès lors qu'il existe un fort rapport de masses volumiques à l'interface. D'autres techniques de découplage s'appuient sur l'utilisation de la formulation fonction de courant-vorticité [73] ayant pour but d'éliminer la pression des équations et par la suite réduire le nombre de variables à résoudre. En outre, les méthodes couplées dites exactes, résolvent simultanément les deux champs (vitesse et pression) en incluant la contrainte dans le système linéaire, contrairement aux méthodes approchées qui induisent une erreur de splitting qui n'est pas consistante avec

les équations du modèle continu. Dans les méthodes couplées, le système à résoudre est connu sous le nom de point selle sur le couple vitesse-pression. Ce système de point selle peut être résolu soit directement via la méthode d'Uzawa, une méthode de type décomposition d'opérateurs, soit au moyen d'une méthode de pénalité (lagrangien augmenté (AL)). À ce propos, la méthode d'Uzawa permet la résolution du système linéaire (3.16) par un algorithme itératif de minimisation, dans lequel chaque itération nécessite la résolution d'une équation sur le champ de vitesse. Une mise à jour ultérieure du champ de pression est effectuée de manière explicite. Quant à la méthode de décomposition d'opérateurs, elle consiste à décomposer les opérateurs différentiels en une somme d'opérateurs simples (sous-problèmes) du problème général, tout en assurant le couplage entre la vitesse et la pression. La première étape consiste donc à résoudre le problème d'advection-diffusion. La deuxième étape repose sur la résolution du problème de Stokes dégénéré. En ce qui concerne la méthode du lagrangien augmenté (AL), elle a été proposée initialement par Fortin et Glowinski [71] principalement pour les écoulements monophasiques en régime de Stokes. Cette approche a été récemment étendue à la simulation des écoulements diphasiques par Vincent et al. [180, 185]. Elle consiste à introduire un paramètre numérique r qui pénalise la contrainte d'incompressibilité, qui doit être fixé par l'utilisateur afin de résoudre au mieux le problème proposé. D'autres algorithmes de couplage vitesse-pression utilisés dans tous les codes de CFD commerciaux ont connu leur succès depuis la mise en place de la méthode SIMPLE de Patankar [139] et de nombreuses variantes (SIMPLER [139], SIMPLESC [50], PISO [100]). Basées sur un procédé itératif, ces techniques simplifient énormément le formalisme et aussi l'implémentation des équations de conservation. Le détail de ces différentes approches couplées et découplées est donné dans les sections (3.2.1) à (3.2.5). Pour simplifier le formalisme, nous considérons un schéma d'Euler d'ordre 1 pour l'intégration temporelle.

3.2.1 Formulation fonction de courant-vorticité ψ - ω

La formulation de fonction courant et celle de la vorticité a été l'un des premiers algorithmes pour résoudre les équations de Navier-Stokes incompressibles. L'algorithme original a été développé par Froom [73] afin d'éliminer le terme de pression, qui complique la résolution de la quantité de mouvement. Pour contourner cette difficulté, Froom s'est appuyé sur d'autres variables du mouvement que la vitesse, à savoir la vorticité $\omega = \nabla \times \bar{u}$:

$$\omega = \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) \quad (3.17)$$

et la fonction courant ψ , définie en 2D par les relations :

$$u = \frac{\partial \psi}{\partial y} \quad v = -\frac{\partial \psi}{\partial x} \quad (3.18)$$

Cette fonction de courant satisfait l'équation de conservation de masse puisque :

$$\nabla \cdot \bar{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = \frac{\partial^2 \psi}{\partial x \partial y} - \frac{\partial^2 \psi}{\partial y \partial x} = 0 \quad (3.19)$$

Considérons un écoulement bidimensionnel incompressible, et plaçons nous dans l'hypothèse où $\mu = \text{cte}$ et $\rho = \text{cte}$. Alors, d'après l'équation de quantité de mouvement écrite sous forme non conservative, et en l'absence de termes sources on a les deux équations suivantes :

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\mu}{\rho} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (3.20)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \frac{\mu}{\rho} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (3.21)$$

Pour éliminer la pression p , dérivons les équations (3.20)-(3.21) par rapport à y et x respectivement,

$$\frac{\partial^2 u}{\partial y \partial t} + \frac{\partial u}{\partial y} \frac{\partial u}{\partial x} + u \frac{\partial^2 u}{\partial y \partial x} + \frac{\partial v}{\partial y} \frac{\partial u}{\partial y} + v \frac{\partial^2 u}{\partial y^2} = -\frac{1}{\rho} \frac{\partial^2 p}{\partial y \partial x} + \frac{\mu}{\rho} \left(\frac{\partial^3 u}{\partial y \partial x^2} + \frac{\partial^3 u}{\partial y^3} \right) \quad (3.22)$$

$$\frac{\partial^2 v}{\partial x \partial t} + \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + u \frac{\partial^2 v}{\partial x^2} + \frac{\partial v}{\partial x} \frac{\partial v}{\partial y} + v \frac{\partial^2 v}{\partial x \partial y} = -\frac{1}{\rho} \frac{\partial^2 p}{\partial x \partial y} + \frac{\mu}{\rho} \left(\frac{\partial^3 v}{\partial x^3} + \frac{\partial^3 v}{\partial x \partial y^2} \right) \quad (3.23)$$

En permutant les opérateurs de différentiation et en soustrayant (3.23) de (3.22), on montre que :

$$\begin{aligned} \frac{\partial}{\partial t} \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) + u \frac{\partial}{\partial x} \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) + v \frac{\partial}{\partial y} \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) + \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) = \\ \frac{\mu}{\rho} \left[\frac{\partial^2}{\partial x^2} \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) + \frac{\partial^2}{\partial y^2} \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) \right] \end{aligned} \quad (3.24)$$

Notons que le quatrième terme du premier membre est nul selon l'équation de conservation de masse ($\nabla \cdot \bar{u} = 0$). En outre, en substituant la vorticit   d  finie par (3.17), nous obtenons l'  quation de transport de la vorticit   :

$$\frac{\partial \omega}{\partial t} + u \frac{\partial \omega}{\partial x} + v \frac{\partial \omega}{\partial y} = \frac{\mu}{\rho} \left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} \right) = \frac{\mu}{\rho} \Delta \omega \quad (3.25)$$

D'autre part, nous pouvons exprimer facilement la vorticit   ω en fonction de ψ :

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} = -\frac{\partial^2 \psi}{\partial x^2} - \frac{\partial^2 \psi}{\partial y^2} = -\Delta \psi \quad (3.26)$$

Ainsi le syst  me    r  soudre est alors compos   des   quations (3.25) et (3.26) :

$$\frac{\partial \omega}{\partial t} + u \frac{\partial \omega}{\partial x} + v \frac{\partial \omega}{\partial y} = \frac{\mu}{\rho} \Delta \omega \quad (3.27)$$

$$\omega + \Delta \psi = 0 \quad (3.28)$$

La r  solution du syst  me (3.27) et (3.28) n  cessite l'impl  mentation des conditions aux limites

pour chaque variable. Elle donne successivement ω , puis ψ et enfin les champs de vitesse u et v par différenciation de la fonction courant. Si le champ de pression est souhaité, il peut être obtenu a posteriori en résolvant l'équation de Poisson suivante (voir [73] pour plus de détails) :

$$\Delta p = 2\rho \left[\frac{\partial^2 \psi}{\partial x^2} \frac{\partial^2 \psi}{\partial y^2} - \left(\frac{\partial^2 \psi}{\partial x \partial y} \right) \right] \quad (3.29)$$

La formulation fonction de courant-vorticité a été largement utilisée pour des écoulements incompressibles monophasiques et bidimensionnels. En revanche, elle est devenue moins populaire parce que son extension aux écoulements tridimensionnels est délicate. En effet, la fonction de courant et la vorticité deviennent des vecteurs à trois composantes en trois dimensions. On a donc un système de six EDP au lieu de deux nécessaires. En outre, la formulation possède des difficultés liées à la présence des propriétés physiques variables, et des conditions aux limites qui ne sont pas aisées à imposer.

3.2.2 La méthode de Prédiction-Correction

La littérature sur les méthodes de projection est vaste et de nombreuses variantes et formulations existent. Dans le cadre de ce travail, la méthode de projection proposée par Goda [81] est considérée, où le gradient de pression est maintenu dans l'étape de prédiction. Contrairement à l'approche de Goda, où l'étape de prédiction est effectuée explicitement, une résolution implicite est choisie ici afin de s'affranchir de la condition de stabilité de type CFL. Le principe de la méthode de projection repose essentiellement sur le découplage des champs de vitesse et de pression, et l'algorithme qui en résulte comporte deux étapes. D'abord, une étape de prédiction où une vitesse intermédiaire \bar{u}^* qui ne satisfait pas la contrainte d'incompressibilité, est déterminée comme solution de l'équation suivante :

$$\rho^{n+1} \left[\frac{(\bar{u}^* - \bar{u}^n)}{\Delta t} + \nabla \cdot (\bar{u}^n \otimes \bar{u}^*) \right] + \bar{\bar{B}}_{penu} \cdot (f(\bar{u}^*) - \bar{u}_\infty) = -\nabla p^n + \nabla \cdot [\mu^{n+1}(\nabla \bar{u}^* + \nabla \bar{u}^{*T})] + \rho^{n+1} \bar{g} + \bar{F}_s \quad (3.30)$$

La deuxième étape de correction, constitue la projection de la vitesse prédite \bar{u}^* sur un champ à divergence nulle. Elle consiste ainsi à calculer l'incrément de pression p^* comme étant la solution de l'équation de Helmholtz suivante, avec des conditions aux limites homogènes de Neumann :

$$\nabla \cdot \left(\frac{\Delta t}{\rho^{n+1}} \nabla p^* \right) + \alpha_p (f(p^*) - p_\infty) = \nabla \cdot \bar{u}^* \quad (3.31)$$

où le terme de pénalisation α_p tend vers zéro à l'intérieur du domaine Ω , et l'infini sur sa frontière Γ . Après l'étape de prédiction et celle de projection, la mise à jour de la vitesse et de la pression est effectuée comme suit :

$$\bar{u}^{n+1} = \bar{u}^* - \frac{\Delta t}{\rho^{n+1}} \nabla p^*, \quad (3.32)$$

$$p^{n+1} = p^n + p^*. \quad (3.33)$$

L'intérêt de la méthode de projection réside dans sa simplicité et sa facilité de mise en œuvre. Cependant, cette approche souffre de nombreuses limitations : l'opérateur de splitting induit une perte de précision temporelle sur la solution numérique, et impose l'utilisation des conditions aux limites appropriées pour le champ de pression dans l'étape de projection qui n'existent pas dans le système original (couplé). En effet, l'inconsistance des conditions aux limites induit des erreurs numériques dans le champ de pression qui donnent lieu à des couches limites artificielles comme décrit dans [85, 89]. En outre, lorsque des écoulements multiphasiques sont simulés avec de grands rapports de masses volumiques, une attention particulière sur le solveur itératif est indispensable à l'inversion du système matriciel résultant de la discrétisation de l'équation de Helmholtz Eq. (3.31), qui devient très mal conditionnée en raison des discontinuités dans l'opérateur laplacien $\nabla \cdot \left(\frac{\Delta t}{\rho^{n+1}} \nabla p^* \right)$. Dans le contexte de la modélisation des écoulements diphasiques à très grands rapports de masses volumiques et de viscosités, de nombreuses variantes de cette technique ont été proposées depuis une quarantaine d'années. On peut citer par exemple la méthode à pas fractionnaires proposée par Guermond et Salgado [84], et qui consiste à extraire la masse volumique de l'équation de Poisson en pénalisant la divergence de la vitesse. Plus récemment Caltagirone et Breil [32] ont adopté une méthode qui porte sur une projection vectorielle du champ de vitesse, en utilisant uniquement la décomposition de Hodge-Helmholtz et une correction de vitesse. Une autre approche intitulée Fast pressure-correction a été développée par Dodd et Ferrante [47]. Elle repose sur la décomposition du gradient de pression en deux termes, l'un avec un terme de densité variable traité de façon explicite et l'autre avec un terme de densité constant traité implicitement.

3.2.3 La méthode d'Uzawa

La méthode d'Uzawa est une méthode itérative de minimisation dans laquelle chaque itération nécessite la résolution d'une équation sur le champ de vitesse. L'algorithme part d'une valeur initiale et itère en la corrigeant pour se rapprocher de la solution du problème. Enfin, les champs de vitesse et de pression sont résolus de manière couplée, en même temps que la contrainte d'incompressibilité. Les résolutions secondaires requises peuvent être remplacées par des résolutions inexactes et la recherche du point selle sur le couple (\bar{u}^{n+1}, p^{n+1}) peut se faire grâce à l'algorithme d'Uzawa.

Une fois que le système est discrétisé, nous avons avec les notations précédentes

$$\begin{pmatrix} F_u & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \bar{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \bar{f} \\ 0 \end{pmatrix}. \quad (3.34)$$

En développant le système (4.24), nous obtenons :

$$F_u \bar{u}^{n+1} + B_p^T p^{n+1} = \bar{f} \quad (3.35)$$

$$B_u \bar{u}^{n+1} = 0 \quad (3.36)$$

De l'équation (3.35), il vient :

$$\bar{u}^{n+1} = F_u^{-1} (\bar{f} - B_p^T p^{n+1}) \quad (3.37)$$

et alors de la seconde équation (3.36) :

$$(B_u F_u^{-1} B_p^T) p^{n+1} = (B_u F_u^{-1}) \bar{f} \quad (3.38)$$

En posant $S = B_u F_u^{-1} B_p^T$ le complément de schur et $C = (B_u F_u^{-1}) \bar{f}$, ceci s'écrit encore :

$$S p^{n+1} = C \quad (3.39)$$

On peut montrer que S est une matrice bien conditionnée et donc (3.39) peut être résolu par des méthodes itératives bien établies. Cependant, la difficulté de cette approche est que le complément de Schur est dense et donc très coûteux à former. Suite à cela, la recherche du point selle sur le couple (\bar{u}^{n+1}, p^{n+1}) peut se faire grâce à l'algorithme d'Uzawa suivant :

```

1
2 ! Initialisation
3  $k = 0$ :  $p^{(0)} \in R^m$  étant donné, trouver  $u^{(0)} \in R^n$  tel que
4  $F_u u^{(0)} = \bar{f} - B_p^T p^{(0)}$ 
5
6 !  $k \geq 0$ :  $(u^{(k)}, p^{(k)})$  étant connus, calculer successivement :
7  $r^{(k)} = B^T u^{(k)}$ 
8  $p^{(k+1)} = p^{(k)} - \lambda r^{(k)}$ 
9 ! puis trouver  $u^{(k)} \in R^n$  tel que
10  $F_u u^{(k)} = f - B_p^T p^{(k+1)}$ 

```

Listing 3.1 – L'algorithme d'Uzawa

où le pas de descente $\lambda > 0$ est un paramètre de cet algorithme. Le test d'arrêt peut être effectué sur le résidu : $\|r^{(k)}\| < \epsilon$, où ϵ est le seuil de convergence.

L'approche d'Uzawa permet d'éviter les effets indésirables des couches limites numériques, qui peuvent être causées par le découplage vitesse pression et qui sont responsables de la limitation de la précision des champs de vitesse et de pression. Elle s'avère utile pour le problème de Stokes en régime stationnaire, puisqu'elle converge indépendamment du maillage. Toutefois, les conditions pour assurer la convergence sont souvent fortes et la convergence est trop lente pour le cas de

Stokes généralisé ainsi que pour le cas de Navier-Stokes, en particulier avec des écoulements à grand nombre de Reynolds. En outre, il faut noter que l'algorithme ne converge pas tout le temps. Par ailleurs, il peut se produire un phénomène d'oscillations, où la convergence est atteinte après une suite infinie d'oscillations.

3.2.4 La méthode de décomposition d'opérateurs

Les méthodes de décomposition d'opérateurs portent sur la résolution de problèmes du type :

$$(\mathbb{P}) \begin{cases} \frac{d\phi}{dt} + A(\phi) = 0 \\ \phi(0) = \phi_0 \end{cases}$$

où $A = A_1 + A_2$ est la somme de deux opérateurs plus simples que A . En utilisant le schéma de Marchuk-Yanenko [80], le problème initial (\mathbb{P}) est subdivisé en deux sous problèmes plus abordables :

$$(\mathbb{P}) \begin{cases} \frac{\phi^{n+1/2} - \phi^n}{\Delta t} + A_1(\phi^{n+1/2}) = 0 \\ \frac{\phi^{n+1} - \phi^{n+1/2}}{\Delta t} + A_2(\phi^{n+1}) = 0 \end{cases}$$

En appliquant ce schéma de décomposition au système (3.14) et (3.15), nous obtenons deux sous-problèmes, le problème d'advection-diffusion (3.40) et le problème de Stokes dégénéré (3.41). La première étape est moins coûteuse et consiste donc à résoudre le problème d'advection-diffusion :

$$\rho^{n+1} \left[\frac{(\bar{u}^* - \bar{u}^n)}{\Delta t} + \nabla \cdot (\bar{u}^n \otimes \bar{u}^*) \right] + \bar{\bar{B}}_{penu} \cdot (f(\bar{u}^*) - \bar{u}_\infty) = -\nabla p^n + \nabla \cdot [\mu^{n+1}(\nabla \bar{u}^* + \nabla \bar{u}^{*T})] + \rho^{n+1} \bar{g} + \bar{F}_s \quad (3.40)$$

La deuxième étape est la résolution du problème de Stokes dégénéré via un algorithme d'Uzawa :

$$\frac{\rho^{n+1} (\bar{u}^{n+1} - \bar{u}^*)}{\Delta t} + \nabla p^{n+1} = 0 \quad (3.41)$$

$$\nabla \cdot \bar{u}^{n+1} = 0 \quad (3.42)$$

La méthode de décomposition d'opérateurs présente un avantage majeur, qui réside dans sa haute précision numérique. En effet, la propriété d'incompressibilité est toujours garantie grâce au processus d'Uzawa. Cependant, il présente un coût énorme en temps de calcul, limitant par la suite son utilisation.

3.2.5 La méthode du lagrangien augmenté standard

Pour pallier les difficultés liées aux méthodes citées précédemment, une méthode exacte appelée lagrangien augmenté standard (SAL) a été proposée. Initialement développée par Fortin

et Glowinski [71] pour étudier des écoulements monophasiques en régime de Stokes, cette méthode a été récemment étendue à la simulation des écoulements diphasiques par Vincent et ses collaborateurs [180, 185] pour tirer avantage de sa robustesse, sa simplicité et son efficacité. Contrairement à la méthode de projection, l'approche du lagrangien augmenté est une méthode d'optimisation qui consiste à remplacer le problème initial en un problème de minimisation sous contrainte, où on aboutit à la recherche d'un point selle sur le couple vitesse-pression. Ce qui permet de satisfaire les deux équations du système en même temps.

Le principe de cette méthode repose sur le fait que l'équation de conservation de masse n'est pas utilisée dans sa forme originale, elle est transformée en une équation de pression qui sera couplée à celle de la vitesse. En effet, si l'on se place dans le cadre d'un écoulement compressible entre deux instants successifs t et $t + dt$, la différentielle de la masse volumique par rapport à la pression et la température s'écrit :

$$\frac{d\rho}{dt} = \left(\frac{\partial\rho}{\partial p}\right)_T \frac{dp}{dt} + \left(\frac{\partial\rho}{\partial T}\right)_p \frac{dT}{dt} \quad (3.43)$$

Comme l'équation de conservation de masse pour un fluide compressible s'écrit :

$$\frac{d\rho}{dt} + \rho \nabla \cdot \bar{u} = 0 \quad (3.44)$$

On trouve :

$$\left(\frac{\partial\rho}{\partial p}\right)_T \frac{dp}{dt} + \left(\frac{\partial\rho}{\partial T}\right)_p \frac{dT}{dt} + \rho \nabla \cdot \bar{u} = 0 \quad (3.45)$$

Soit :

$$\chi_T \frac{dp}{dt} + \beta \frac{dT}{dt} + \nabla \cdot \bar{u} = 0 \quad (3.46)$$

Où : $\chi_T = \frac{1}{\rho} \left(\frac{\partial\rho}{\partial p}\right)_T$ et $\beta = \frac{1}{\rho} \left(\frac{\partial\rho}{\partial T}\right)_p$ sont respectivement, les coefficient de compressibilité isotherme, et de dilatation thermique isobare.

Pour un écoulement isotherme, on écrit la variation thermodynamique de la pression (3.46) entre deux instants d'équilibre du système comme suit :

$$p^{n+1} = p^n - r \nabla \cdot \bar{u}^{n+1} \quad (3.47)$$

Ici, n se réfère à un indice associé au temps physique $n\Delta t$, Δt est le pas de temps supposé constant à chaque itération résolue et $r = \frac{\chi_T}{\Delta t}$ représente le paramètre du lagrangien augmenté utilisé pour imposer la contrainte d'incompressibilité. Ainsi, connaissant les champs initiaux de vitesse $\bar{u}^{*,0} = \bar{u}^n$ et de pression $p^{*,0} = p^n$, la solution est donnée par l'algorithme d'Uzawa [173] : tant que $|\nabla \cdot \bar{u}^{*,m}| > \epsilon$, résoudre :

$$\begin{aligned}
\rho^{n+1} \left[\frac{(\bar{u}^{*,m} - \bar{u}^{*,0})}{\Delta t} + \nabla \cdot (\bar{u}^{*,0} \otimes \bar{u}^{*,m}) \right] + \bar{B}_{penu} \cdot (f(\bar{u}^{*,m}) - \bar{u}_\infty) \\
= -\nabla p^{*,0} + r \nabla (\nabla \cdot \bar{u}^{*,m}) + \nabla \cdot [\mu^{n+1} (\nabla \bar{u}^{*,m} + \nabla \bar{u}^{(*,m)T})] + \rho^{n+1} \bar{g} + \bar{F}_s, \quad (3.48)
\end{aligned}$$

$$p^{*,m} = p^{*,0} - r \nabla \cdot \bar{u}^{*,m} \quad (3.49)$$

avec, ϵ le seuil de précision basé sur le niveau de divergence souhaité et m l'indice d'itération. Ainsi pour $m = 1, 2 \dots n$ le processus itératif corrige simultanément la pression et la vitesse jusqu'à la convergence vers la solution (\bar{u}^{n+1}, p^{n+1}) . En pratique, une itération du lagrangien augmenté (AL) avec une valeur adéquate du paramètre r suffit pour obtenir une bonne solution. Un choix approprié de r doit garantir que la contrainte d'incompressibilité est satisfaite, tout en assurant que la valeur du terme lagrangien augmenté (AL) soit comparable au plus grand terme des équations de Navier-Stokes. En prenant des valeurs très faibles de r , l'importance est donnée aux équations de Navier-Stokes alors que lorsque r est très grand, le champ de vitesse sera à divergence nulle mais ne satisfera pas les équations de Navier-Stokes. Généralement, des valeurs de r 10 à 1000 fois plus grandes que les autres termes adimensionnels dans les équations de bilan de quantité de mouvement donnent des résultats satisfaisants.

La méthode du lagrangien augmenté présente de nombreux avantages. D'une part, elle évite d'imposer des conditions aux limites sur le champ de pression. D'autre part, cette méthode est très robuste, dans la mesure où elle permet de simuler des écoulements diphasiques à très grands rapports de masses volumiques et de viscosités ($\frac{\rho_2}{\rho_1} \ll 1$ ou $\frac{\rho_2}{\rho_1} \gg 1$ ainsi que $\frac{\mu_2}{\mu_1} \ll 1$ ou $\frac{\mu_2}{\mu_1} \gg 1$), contrairement à la méthode de projection. D'autres avantages sont liés à l'utilisation de grands pas de temps, ce qui présente un intérêt majeur lorsqu'on veut obtenir rapidement une solution pilotée par un pas de temps supérieur à la condition de stabilité. Néanmoins, cette méthode possède des inconvénients, qui résident sur le fait qu'elle coûte cher numériquement. En effet, la simulation des écoulements diphasiques avec des grands rapports de masses volumiques et de viscosités requiert l'utilisation de grandes valeurs du paramètre r , ce qui conduit à des systèmes linéaires mal conditionnés dont le coût d'inversion est cher. En outre, il est important de préciser que la solution numérique dépend fortement du choix du paramètre r , lui-même très sensible aux paramètres du problème tels que les rapports de masses volumiques et de viscosités, mais aussi au maillage.

Des améliorations de la méthode du lagrangien augmenté ont été proposées par Vincent et ses collaborateurs. Contrairement à la méthode standard du lagrangien augmenté, où le paramètre de pénalité fixé par l'utilisateur est constant, la méthode du lagrangien augmenté adaptative et algébrique (AALL) automatise le calcul du paramètre r qui devient une fonction du temps et de l'espace $r(t, M)$ qui s'adapte localement. Ainsi, l'estimation de $r(t, M)$ repose sur l'analyse

de la matrice issue de la discrétisation de l'équation de vitesse pénalisée.

3.2.6 La méthode SIMPLE

La méthode SIMPLE de Patankar [139] fait partie des familles de solveur qui traitent le couplage vitesse-pression via un processus itératif. Dans la littérature, il existe de nombreuses variantes de cette méthode, comme SIMPLER, SIMPLEC ou encore PISO. Nous décrivons par la suite la méthode SIMPLE. Pour avoir un aperçu des autres variantes, nous invitons le lecteur à se référer à [50, 100, 139].

Dans un premier temps, nous rappelons les équations de Navier-Stokes à résoudre :

$$\rho^{n+1} \left[\left(\frac{\partial \bar{u}}{\partial t} \right)^{n+1} + (\nabla \cdot (\bar{u} \otimes \bar{u}))^{n+1} \right] = -(\nabla p)^{n+1} + \left(\nabla \cdot \left[\mu^{n+1} (\nabla \bar{u} + \nabla^T \bar{u}) \right] \right)^{n+1} + \rho^{n+1} \bar{g}, \quad (3.50)$$

$$\nabla \cdot \bar{u}^{n+1} = 0. \quad (3.51)$$

Ensuite, nous linéarisons le terme convectif en prenant la vitesse advective à l'instant n :

$$\rho^{n+1} \left[\left(\frac{\partial \bar{u}}{\partial t} \right)^{n+1} + \nabla \cdot (\bar{u}^n \otimes \bar{u}^{n+1}) \right] = -(\nabla p)^{n+1} + \left(\nabla \cdot \left[\mu^{n+1} (\nabla \bar{u} + \nabla^T \bar{u}) \right] \right)^{n+1} + \rho^{n+1} \bar{g}. \quad (3.52)$$

Dans l'équation précédente, la pression p^{n+1} à l'instant $n+1$ n'est pas connue. Il est donc intéressant de considérer la pression à l'instant n tout en résolvant une vitesse intermédiaire \bar{u}^* (qui ne satisfait pas l'équation de conservation de masse), il vient :

$$\rho^{n+1} \left[\left(\frac{\partial \bar{u}}{\partial t} \right)^* + \nabla \cdot (\bar{u}^n \otimes \bar{u}^*) \right] = -(\nabla p)^n + \left(\nabla \cdot \left[\mu^{n+1} (\nabla \bar{u} + \nabla^T \bar{u}) \right] \right)^* + \rho^{n+1} \bar{g}, \quad (3.53)$$

Les équations du mouvement (3.52)-(3.53) discrétisées via des volumes finis, peuvent se mettre sous les formes suivantes :

$$A_P^{\bar{u}} \bar{u}^{n+1} + \sum_{K=E,W,N,S} A_K^{\bar{u}} \bar{u}_K^{n+1} = \bar{b}_u^n - (\nabla p^{n+1}) V_P \quad (3.54)$$

$$A_P^{\bar{u}^*} \bar{u}^* + \sum_{K=E,W,N,S} A_K^{\bar{u}^*} \bar{u}_K^* = \bar{b}_u^n - (\nabla p^n) V_P$$

(3.55)

avec \bar{b}_u^n est le second membre dont on aura soustrait les gradients de pression ∇p^{n+1} et ∇p^n

aux instants $n + 1$ et n , V_P le volume de contrôle, P le nœud au centre de chaque volume de contrôle, E et W les nœuds Est et Ouest et N et S les nœuds Nord et Sud.

A ce niveau, aucune des deux variables n'est correcte. Le champ de vitesse intermédiaire \bar{u}^* devra être corrigé par une vitesse correctrice \bar{u}' , telle que :

$$\bar{u}^{n+1} = \bar{u}^* + \bar{u}' . \quad (3.56)$$

La pression p^{n+1} , tel que le couple (\bar{u}^{n+1}, p^{n+1}) vérifie les équations de conservations (3.50) et (3.51) est obtenue en introduisant une pression correctrice comme suit :

$$p^{n+1} = p^n + p' . \quad (3.57)$$

Pour obtenir le champ correctif \bar{u}' , il faut soustraire l'équation (3.55) de l'équation (3.54). On obtient alors :

$$A_P^{\bar{u}} \bar{u}' + \sum_{K=E,W,N,S} A_K^{\bar{u}} \bar{u}'_K = - (\nabla p') V_P . \quad (3.58)$$

Il s'en suit :

$$\bar{u}' = - \sum_{K=E,W,N,S} \frac{A_K^{\bar{u}} \bar{u}'_K}{A_P^{\bar{u}}} - \frac{V_P}{A_P^{\bar{u}}} (\nabla p') \quad (3.59)$$

À priori, le premier terme du second membre de (3.59) n'est pas connu. L'approche à suivre sera donc de négliger ce terme. Cette hypothèse simplificatrice n'influence pas le résultat final, puisque \bar{u}' représente une correction qui tend vers zéro une fois l'algorithme convergé. Ainsi, la relation entre la vitesse correctrice et la pression correctrice s'exprime par :

$$\bar{u}' = - \frac{V_P}{A_P^{\bar{u}}} (\nabla p') \quad (3.60)$$

L'introduction de l'expression corrigée (3.60) dans l'équation de conservation de la masse (3.51), donne l'équation de correction de pression. Elle s'écrit sous la forme suivante :

$$\nabla \cdot \bar{u}^* = \nabla \cdot \left[\frac{V_P}{A_P^{\bar{u}}} (\nabla p') \right] \quad (3.61)$$

Enfin, l'algorithme SIMPLE sera résumé comme suit :

- 1 ! Initialisation des champs de vitesse et de pression
- 2 Choisir $p^m = p^n$ et $\bar{u}^m = \bar{u}^n$
- 3 ! Résolution de l'équation de quantité de mouvement pour obtenir le champ prédicteur \bar{u}^m à
- 4 ! l'itération m entre deux itérations temporelles n et $n+1$
- 5 $A_P^{\bar{u}} \bar{u}^m + \sum_K A_K^{\bar{u}} \bar{u}_K^m = \bar{b}_u^n - (\nabla p^n) V_P$
- 6 ! Résolution de l'équation de correction de pression pour obtenir la pression correctrice p'

7	$\nabla \cdot \left[\frac{V_P}{A_P^u} (\nabla p') \right] = \nabla \cdot \bar{u}^m$
8	! Détermination de la correction de vitesse \bar{u}'
9	$\bar{u}' = -\frac{V_P}{A_P^u} (\nabla p')$
10	! Mise à jour de la vitesse et de la pression
11	$\bar{u}^{m+1} = \bar{u}^m + \bar{u}'$
12	$p^{m+1} = p^m + p'$

Listing 3.2 – L’algorithme SIMPLE

En résumé, l’algorithme SIMPLE permet de simplifier l’écriture des équations de conservation. Cependant, des difficultés de convergence apparaissent et peuvent conduire à une divergence de l’algorithme. En effet, la simplification effectuée dans l’équation (3.59) affecte le taux de convergence et surestime la pression corrective p' . Pour pallier cette difficulté, il est intéressant d’introduire un coefficient de sous-relaxation aussi bien pour la composante de vitesse que celle de la pression. Les équations (3.56)-(3.57) deviennent :

$$\bar{u}^{n+1} = \bar{u}^* + \alpha_u \bar{u}' \quad (3.62)$$

$$p^{n+1} = p^n + \alpha_p p' \quad (3.63)$$

Certains auteurs préconisent que $\alpha_u + \alpha_p = 1$, où α_u et α_p représentent les coefficients de sous-relaxation associés respectivement à la vitesse et la pression. Toutefois, l’ajustement des facteurs de sous-relaxation reste difficile, et présente une forte dépendance à la nature du problème. Le remède à ce problème se fait via l’utilisation de nombreuses autres variantes, à savoir SIMPLER, SIMPLEC et PISO.

3.3 Bilan des algorithmes de couplage vitesse-pression

Le tableau (3.1) regroupe les propriétés des différentes familles de solveurs présentées plus haut. Pour résumer, la principale difficulté des méthodes de projection-correction réside dans le fait qu’elles sont beaucoup plus sensibles au rapport de masses volumiques au travers des interfaces. Rajoutons à cela l’inconsistance entre les équations continues et les équations discrètes. Quant à la formulation fonction de courant-vorticité, elle est limitée aux écoulements monophasiques bidimensionnels, et nécessite des conditions aux limites assez difficiles à mettre en place. Dans les méthodes de décomposition d’opérateur ou celle d’Uzawa, nous avons pu voir que le processus itératif rend ces méthodes plus coûteuses en temps de calcul, malgré leur haute précision numérique. En ce qui concerne la méthode du lagrangien augmenté, elle présente beaucoup d’avantages et reste très robuste même en présence de forts rapports de masses volumiques et de viscosités. Cependant, cette méthode coûte cher numériquement à cause du paramètre numérique r , qui reste difficile à fixer. En effet, un choix judicieux du paramètre r permet d’accélérer

la convergence mais c'est une opération qui n'est pas évidente. Les autres algorithmes de couplage vitesse-pression, tel que SIMPLE, restent robustes mais ils présentent l'inconvénient d'être itératifs.

Compte tenu des difficultés citées ci-dessus, nous avons fait le choix d'aborder la résolution du point selle du point de vue algébrique par des pré-conditionneurs spécifiques afin d'améliorer les propriétés spectrales du système algébrique (3.16). Cette approche intitulée solveur tout-couplé, permet d'assurer le couplage direct, sans perturbation des champs de vitesse et de pression. Une telle stratégie, aussi préconditionnée par des préconditionneurs triangulaires par bloc et des estimations du complément de Schur, s'est avérée être efficace et robuste malgré la présence de grands rapports de viscosités et de masses volumiques. Ses performances dépendent en effet fortement de celles du solveur de systèmes linéaires. Nous reviendrons plus en détails sur cette approche lorsque nous aborderons la mise en évidence de l'intérêt d'un solveur tout-couplé original (voir chapitre 4).

TABLE 3.1 – Récapitulatif des spécificités et des performances de certains algorithmes de couplage vitesse-pression. ψ - ω : formulation fonction courant-vorticité, PC : Projection-correction, DO : Décomposition d'opérateurs, LA : lagrangien augmenté.

Couplage	Consistance	Coût de calcul	$\nabla \cdot \bar{u} = 0$	CL sur la pression	$\frac{\rho_2}{\rho_1} \gg 1$
ψ - ω	Non	Raisnable	Exacte	Oui	Impossible
PC	Non	Raisnable	Exacte	Oui	Très sensible
Uzawa	Oui	Cher	Exacte	Non	Sensible
DO	Non	Cher	Exacte	Non	Sensible
LA	Oui	Cher	Exacte	Non	Sensible
SIMPLE	Non	Cher	Exacte	Oui	Sensible

3.4 Intégration spatiale des équations de Navier-Stokes diphasiques

Nous avons fait le choix de résoudre numériquement les équations de conservation du modèle à un fluide (voir Eqs. 3.14-3.15) sur un maillage cartésien uniforme de type MAC, introduit initialement par Harlow et Welch. Les variables discrètes de vitesse et de pression sont disposées de façon décalée, elles possèdent chacune leur propre volume de contrôle comme cela est schématisé sur la figure 3.1. Ce choix permet de garantir une discrétisation naturelle des opérateurs différentiels tels que la divergence et le gradient, et évite également les oscillations sur le champ de pression. Il a aussi l'avantage de traiter naturellement les conditions aux limites, tout en engendrant des nœuds fictifs à l'extérieur du domaine physique. La discrétisation spatiale que nous allons présenter en 2D ici a été généralisée en 3D.

Avant d'être résolues numériquement, les équations du modèle à un fluide doivent être discrétisées spatialement. Il existe ainsi une grande variété de méthodes numériques qui permettent de

discrétiser ce système d'équations, comme la méthode des différences finies, la méthode des éléments finis et celle des volumes finis. Ici, nous adoptons une approche volumes finis, qui assure la conservation de la masse et de la quantité de mouvement sur chaque volume de contrôle. Cette méthode consiste à intégrer le système d'équations de conservation (voir Eqs. 3.14-3.15) sur chaque volume de contrôle $\Omega_{i,j}$ délimité par la surface $\Sigma_{i,j}$. Ainsi, si nous intégrons la formulation locale du modèle à un fluide avec un schéma d'Euler d'ordre 1, nous obtenons une description globale de l'écoulement diphasique comme suit :

$$\int_{\Omega_{\bar{u}(i,j)}} \frac{\rho^{n+1}}{\Delta t} (\bar{u}^{n+1} - \bar{u}^{n-1}) dV + \int_{\Omega_{\bar{u}(i,j)}} \rho^{n+1} \nabla \cdot (\bar{u}^n \otimes \bar{u}^{n+1}) dV + \int_{\Omega_{\bar{u}(i,j)}} \bar{B}_{penu} (f(\bar{u}^{n+1}) - \bar{u}_\infty) dV = - \int_{\Omega_{\bar{u}(i,j)}} \nabla p^{n+1} dV + \int_{\Omega_{\bar{u}(i,j)}} \nabla \cdot \bar{T}^{n+1} dV + \int_{\Omega_{\bar{u}(i,j)}} \rho^{n+1} \bar{g} dV + \int_{\Omega_{\bar{u}(i,j)}} \bar{F}_s dV, \quad (3.64)$$

$$\int_{\Omega_p(i,j)} \nabla \cdot \bar{u}^{n+1} dV = 0. \quad (3.65)$$

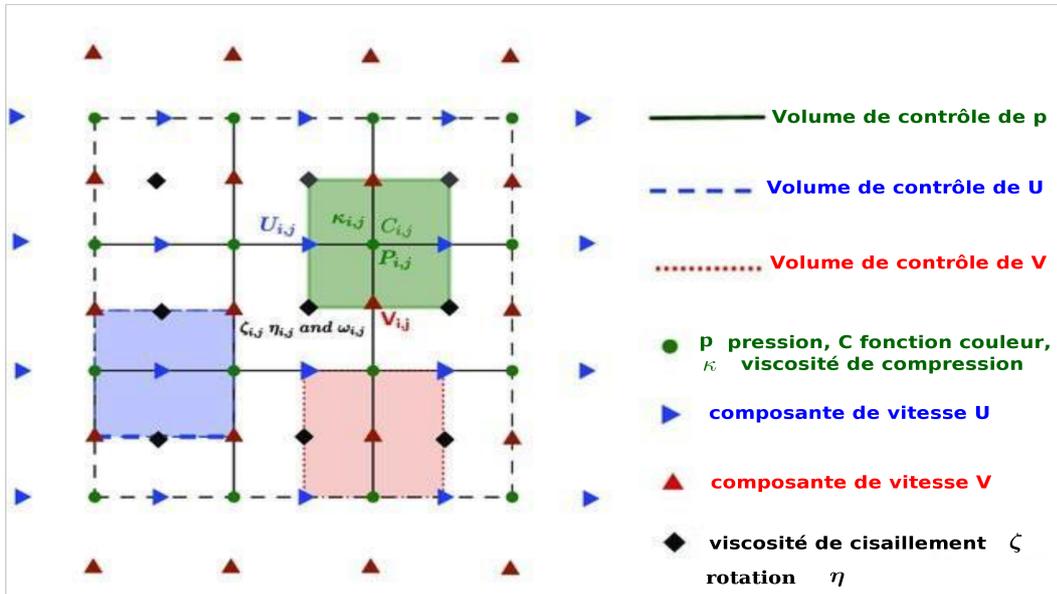


FIGURE 3.1 – Maillage décalé de type MAC. Volumes de contrôle utilisés. La pression $p_{i,j}$, la viscosité d'élongation $\kappa_{i,j}$ et la fonction de couleur $C_{i,j}$ sont localisées aux nœuds de la grille eulérienne, la vitesse horizontale $u_{i,j}$ est définie sur les faces gauche et droite du volume de contrôle (en vert) associé à $p_{i,j}$ et la vitesse verticale $v_{i,j}$ est définie sur les côtés inférieur et supérieur du volume de contrôle associé à $p_{i,j}$, la vorticité $\Omega_{i,j}$, et les viscosités de cisaillement et rotation $\eta_{i,j}$ et $\zeta_{i,j}$ sont disposées aux sommets du volume de contrôle associé à $p_{i,j}$.

Nous utilisons ensuite le théorème de la divergence pour transformer les intégrales sur les volumes de contrôle $\Omega_{i,j}$, en intégrales surfaciques sur le contour $\Sigma_{i,j}$ de normale \bar{n} . Nous obtenons :

$$\int_{\Omega_{\bar{u}(i,j)}} \frac{\rho^{n+1}}{\Delta t} (\bar{u}^{n+1} - \bar{u}^n) dV + \int_{\Sigma_{\bar{u}(i,j)}} \rho^{n+1} \bar{u}^{n+1} \bar{u}^n \cdot \bar{n} dS + \int_{\Omega_{\bar{u}(i,j)}} \bar{B}_{penu} (f(\bar{u}^{n+1}) - \bar{u}_\infty) dV = - \int_{\Omega_{\bar{u}(i,j)}} \nabla p^{n+1} dV + \int_{\Sigma_{\bar{u}(i,j)}} \bar{T}^{n+1} \cdot \bar{n} dS + \int_{\Omega_{\bar{u}(i,j)}} \rho^{n+1} \bar{g} dV + \int_{\Omega_{\bar{u}(i,j)}} \bar{F}_s dV, \quad (3.66)$$

$$\int_{\Omega_{p(i,j)}} \nabla \cdot \bar{u}^{n+1} dV = 0. \quad (3.67)$$

Pour estimer les flux à travers le contour $\Sigma_{i,j}$, nous définissons les pas d'espace Δx et Δy respectivement dans la direction x et y . Par souci de simplicité, nous nous limitons au cas où le maillage est régulier, c'est-à-dire $\Delta x = \Delta y = h$.

3.4.1 Calcul du terme convectif

Nous discrétisons le terme convectif de l'équation (3.66) en premier ordre en temps, en supposant que la masse volumique ρ^{n+1} est constante dans le volume de contrôle $\Omega_{u(i,j)}$:

$$\int_{\Omega_{\bar{u}(i,j)}} \rho^{n+1} \nabla \cdot (\bar{u}^n \otimes \bar{u}^{n+1}) d\Omega \approx \rho^{n+1} \int_{\Sigma_{\bar{u}(i,j)}} \bar{u}^n (\bar{u}^{n+1} \cdot \bar{n}) d\Sigma \quad (3.68)$$

L'approximation des flux de l'équation (C.1), à chaque face de volume de contrôle $\Omega_{u(i,j)}$ (Fig. 3.2a) se fait via un schéma centré d'ordre 2. Elle s'écrit dans la direction x comme suit :

$$\rho^{n+1} \int_{\Sigma_{u(i,j)}} \bar{u}^n (\bar{u}^{n+1} \cdot \bar{n}) d\Sigma_{u(i,j)} \approx \rho_{i,j}^{n+1} \left(\frac{u_{i,j}^n + u_{i+1,j}^n}{2} \frac{u_{i,j}^{n+1} + u_{i+1,j}^{n+1}}{2} - \frac{u_{i-1,j}^n + u_{i,j}^n}{2} \frac{u_{i-1,j}^{n+1} + u_{i,j}^{n+1}}{2} \right) h + \rho_{i,j}^{n+1} \left(\frac{v_{i,j}^n + v_{i,j+1}^n}{2} \frac{u_{i,j}^{n+1} + u_{i,j+1}^{n+1}}{2} - \frac{v_{i,j-1}^n + v_{i,j}^n}{2} \frac{u_{i,j-1}^{n+1} + u_{i,j}^{n+1}}{2} \right) h \quad (3.69)$$

De la même façon, le calcul des flux à chaque face du volume de contrôle $\Omega_{v(i,j)}$ (Fig. 3.2b) est défini comme suit :

$$\rho^{n+1} \int_{\Sigma_{v(i,j)}} \bar{v}^n (\bar{u}^{n+1} \cdot \bar{n}) d\Sigma_{v(i,j)} \approx \rho_{i,j}^{n+1} \left(\frac{u_{i+1,j}^n + u_{i+1,j-1}^n}{2} \frac{v_{i,j}^{n+1} + v_{i+1,j}^{n+1}}{2} - \frac{u_{i,j}^n + u_{i,j-1}^n}{2} \frac{v_{i-1,j}^{n+1} + v_{i,j}^{n+1}}{2} \right) h + \rho_{i,j}^{n+1} \left(\frac{v_{i,j}^n + v_{i,j+1}^n}{2} \frac{v_{i,j}^{n+1} + v_{i,j+1}^{n+1}}{2} - \frac{v_{i,j-1}^n + v_{i,j}^n}{2} \frac{v_{i,j-1}^{n+1} + v_{i,j}^{n+1}}{2} \right) h \quad (3.70)$$

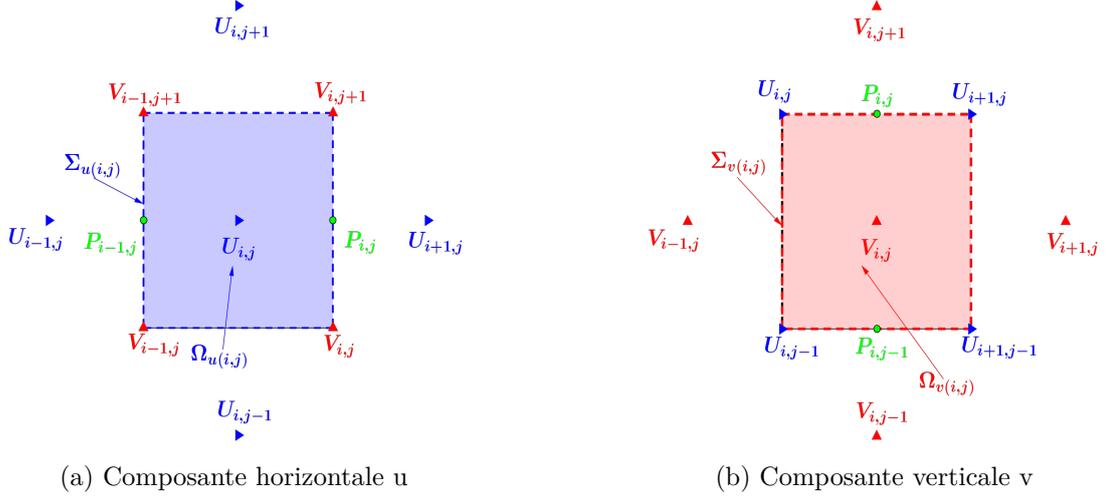


FIGURE 3.2 – Discrétisation des flux convectifs

3.4.2 Calcul du terme Visqueux

Le tenseur des taux de déformation est donné par :

$$T = \kappa \begin{pmatrix} \frac{\partial u}{\partial x} & 0 \\ 0 & \frac{\partial v}{\partial y} \end{pmatrix} + \zeta \begin{pmatrix} 0 & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & 0 \end{pmatrix} - \eta \begin{pmatrix} 0 & \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} & 0 \end{pmatrix} \quad (3.71)$$

avec $\kappa = 2\mu$ la viscosité d'élongation, $\zeta = 2\mu$ la viscosité de cisaillement et $\eta = \mu$ la viscosité de rotation.

Cet opérateur est approximé implicitement, en utilisant un schéma centré d'ordre 2. Son approximation s'écrit pour la composante u de vitesse comme étant une somme de trois contributions (élongation, cisaillement et rotation). En utilisant les points de maillage présentés sur les figures 3.3a-3.4a-3.5a et nécessaires pour l'approximation en volume finis, le terme visqueux pour la composante horizontale u se poursuit de la façon suivante :

- Le terme de compression dans direction x s'écrit :

$$\begin{aligned} \int_{\Sigma_u(i,j)} \frac{\partial}{\partial x} \left(\kappa \frac{\partial u}{\partial x} \right) dV &\approx \left[\kappa \frac{\partial u}{\partial x} \Delta y \right]_{b_x}^{f_x} \\ &\approx \left(\kappa_{i,j} \frac{u_{i+1,j}^{n+1} - u_{i,j}^{n+1}}{\Delta x} - \kappa_{i-1,j} \frac{u_{i,j}^{n+1} - u_{i-1,j}^{n+1}}{\Delta x} \right) \Delta y \end{aligned} \quad (3.72)$$

avec f_x (b_x) la face avant (arrière) dans la direction x .

- Le terme de cisaillement dans direction x s'écrit :

$$\begin{aligned} \int_{\Sigma_u(i,j)} \frac{\partial}{\partial y} \left(\zeta \frac{\partial u}{\partial y} \right) dV &\approx \left[\zeta \frac{\partial u}{\partial y} \Delta x \right]_{b_y}^{f_y} \\ &\approx \left(\zeta_{i,j+1} \frac{u_{i,j+1}^{n+1} - u_{i,j}^{n+1}}{\Delta y} - \zeta_{i,j} \frac{u_{i,j}^{n+1} - u_{i,j-1}^{n+1}}{\Delta y} \right) \Delta x \end{aligned} \quad (3.73)$$

avec f_y (b_y) la face avant (arrière) dans la direction y .

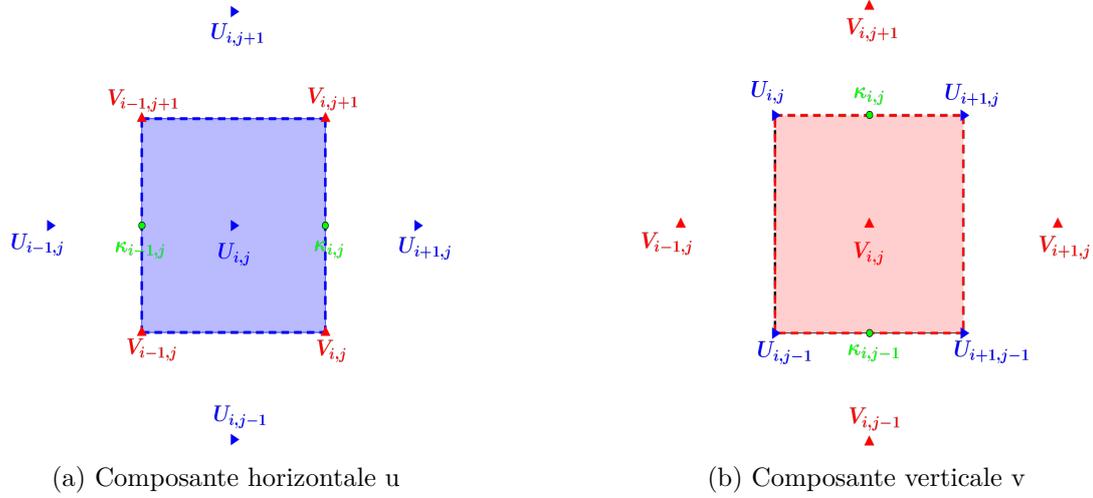


FIGURE 3.3 – Discretisation des flux de compression

- Le terme de rotation dans direction x s'écrit :

$$\begin{aligned} \int_{\Sigma_u(i,j)} \frac{\partial}{\partial y} \left(-\eta \frac{\partial u}{\partial y} + \eta \frac{\partial v}{\partial x} \right) dV &\approx \left[-\eta \frac{\partial u}{\partial y} + \eta \frac{\partial v}{\partial x} \right]_{b_y}^{f_y} \Delta x \\ &\approx \left(-\eta_{i,j+1} \frac{u_{i,j+1}^{n+1} - u_{i,j}^{n+1}}{\Delta y} + \eta_{i,j+1} \frac{v_{i,j+1}^{n+1} - v_{i-1,j+1}^{n+1}}{\Delta x} \right) \Delta x \\ &\quad + \left(\eta_{i,j} \frac{u_{i,j}^{n+1} - u_{i,j-1}^{n+1}}{\Delta y} - \eta_{i,j} \frac{v_{i,j}^{n+1} - v_{i-1,j}^{n+1}}{\Delta x} \right) \Delta x \end{aligned} \quad (3.74)$$

Pour la composante v verticale de vitesse, le terme visqueux est approximé en utilisant les points de maillage illustrés sur les figures 3.3b-3.4b-3.5b.

- Le terme de compression dans direction y s'écrit :

$$\begin{aligned} \int_{\Sigma_v(i,j)} \frac{\partial}{\partial y} \left(\kappa \frac{\partial v}{\partial y} \right) dV &\approx \left[\kappa \frac{\partial v}{\partial y} \Delta x \right]_{b_x}^{f_x} \\ &\approx \left(\kappa_{i,j} \frac{v_{i,j+1}^{n+1} - v_{i,j}^{n+1}}{\Delta y} - \kappa_{i-1,j} \frac{v_{i,j}^{n+1} - v_{i-1,j-1}^{n+1}}{\Delta x} \right) \Delta x \end{aligned} \quad (3.75)$$

- Le terme de cisaillement dans direction y s'écrit :

$$\begin{aligned} \int_{\Sigma_v(i,j)} \frac{\partial}{\partial x} \left(\zeta \frac{\partial v}{\partial x} \right) dV &\approx \left[\zeta \frac{\partial v}{\partial x} \Delta y \right]_{b_y}^{f_y} \\ &\approx \left(\zeta_{i+1} \frac{v_{i+1,j}^{n+1} - v_{i,j}^{n+1}}{\Delta x} - \zeta_{i,j} \frac{v_{i,j}^{n+1} - v_{i-1,j}^{n+1}}{\Delta x} \right) \Delta y \end{aligned} \quad (3.76)$$

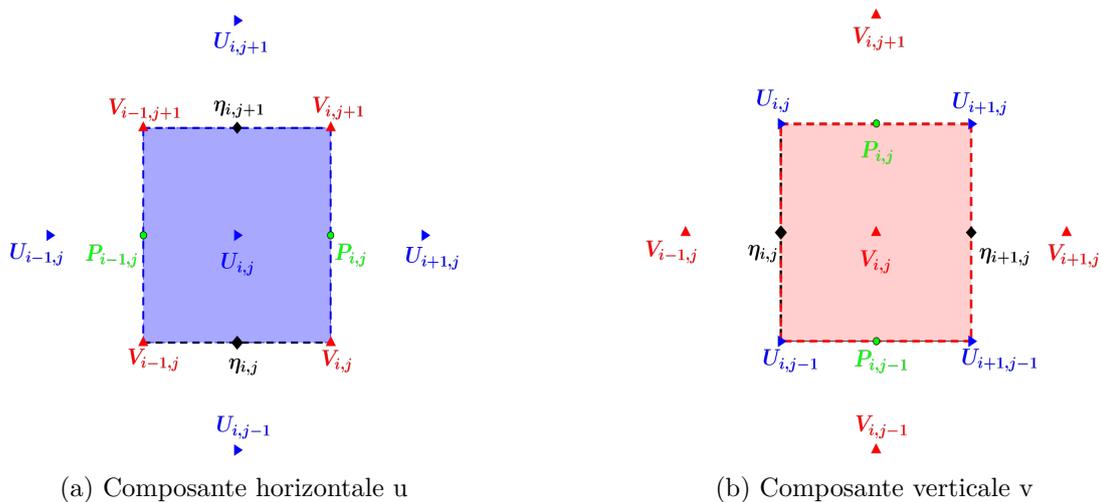


FIGURE 3.4 – Discretisation des flux de cisailage

- Le terme de rotation dans direction y s'écrit :

$$\begin{aligned}
 & \int_{\Sigma_v(i,j)} \frac{\partial}{\partial x} \left(\eta \frac{\partial v}{\partial x} - \eta \frac{\partial u}{\partial y} \right) dV \\
 & \approx \left[\eta \frac{\partial v}{\partial x} - \eta \frac{\partial u}{\partial y} \right]_{b_y}^{f_y} \Delta y \\
 & \approx \left(\eta_{i+1,j} \frac{v_{i+1,j}^{n+1} - v_{i,j}^{n+1}}{\Delta x} - \eta_{i,j} \frac{u_{i+1,j}^{n+1} - u_{i+1,j-1}^{n+1}}{\Delta x} \right) \Delta y \\
 & - \left(\eta_{i,j} \frac{v_{i,j}^{n+1} - v_{i-1,j}^{n+1}}{\Delta x} - \eta_{i,j} \frac{u_{i,j}^{n+1} - u_{i,j-1}^{n+1}}{\Delta y} \right) \Delta y
 \end{aligned} \tag{3.77}$$

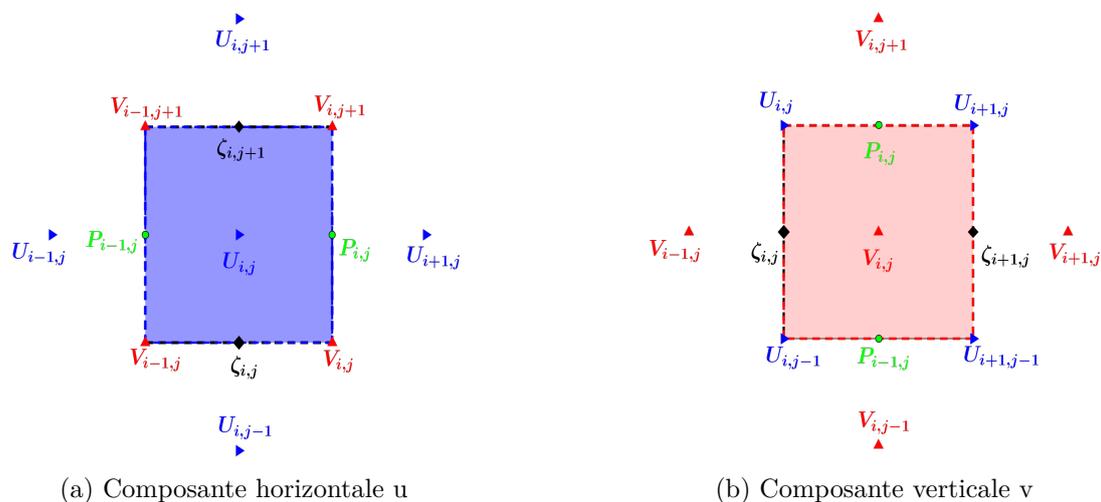


FIGURE 3.5 – Discretisation des flux de rotation

3.4.3 Calcul du gradient de pression

Si on se réfère à la figure 3.6a,

- le gradient de pression pour la composante horizontale de la vitesse u s'écrit :

$$\begin{aligned} \int_{\Sigma_u(i,j)} p^{n+1} n_x dS &\approx \left[p^{n+1} \Delta y \right]_{b_x}^{f_x} \\ &\approx \left(p_{i,j}^{n+1} - p_{i-1,j}^{n+1} \right) \Delta y \end{aligned} \quad (3.78)$$

De façon analogue (Fig. 3.6a),

- la composante verticale du gradient est discrétisée de la façon suivante :

$$\begin{aligned} \int_{\Sigma_v(i,j)} p^{n+1} n_y dS &\approx \left[p^{n+1} \Delta x \right]_{b_y}^{f_y} \\ &\approx \left(p_{i,j}^{n+1} - p_{i,j-1}^{n+1} \right) \Delta x \end{aligned} \quad (3.79)$$

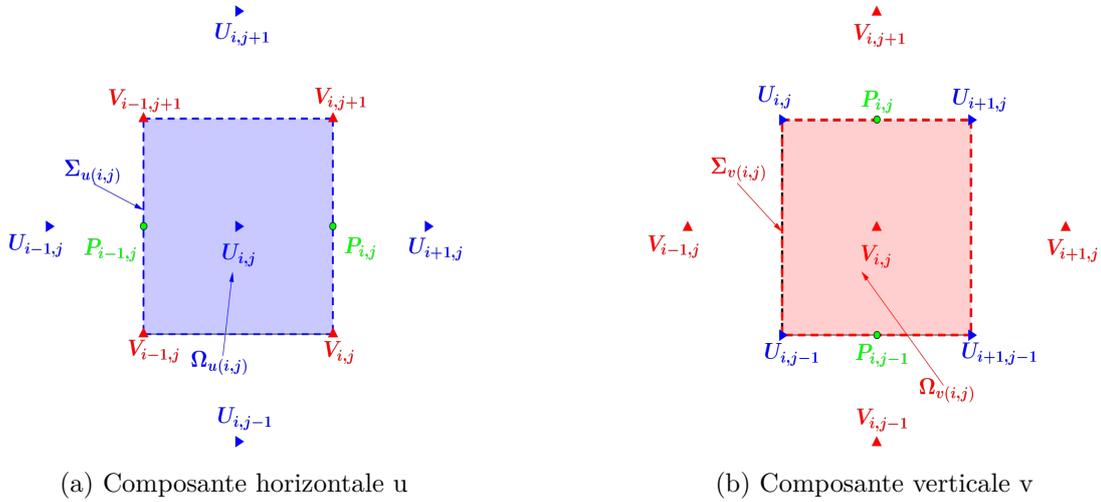


FIGURE 3.6 – Discrétisation du gradient de pression

3.4.4 Calcul de la divergence de la vitesse

La divergence $\nabla \cdot \bar{u}^{n+1}$ du champ de vitesse $\bar{u}^{n+1} = (u^{n+1}, v^{n+1})$ est approximée au centre des cellules (Fig. 3.7) par :

$$\begin{aligned} \int_{\Omega_p(i,j)} \nabla \cdot \bar{u}^{n+1} dV &\approx \left[\left(\frac{\partial u}{\partial x} \right)^{n+1} \Delta x \Delta y \right]_{b_x}^{f_x} + \left[\left(\frac{\partial v}{\partial y} \right)^{n+1} \Delta x \Delta y \right]_{b_y}^{f_y} \\ &\approx \left(\frac{u_{i+1,j}^{n+1} - u_{i,j}^{n+1}}{\Delta x} + \frac{v_{i,j+1}^{n+1} - v_{i,j}^{n+1}}{\Delta x} \right) \Delta x \Delta y \end{aligned} \quad (3.80)$$

3.4.5 Expression du terme capillaire

Comme nous l'avons annoncé ci-dessus, le terme capillaire \bar{F}_s de l'équation (3.12) est approximé de façon explicite via la méthode CFS. L'intégration de ce terme dans un volume de contrôle $\Omega_{i,j}$ se poursuit de la façon suivante :

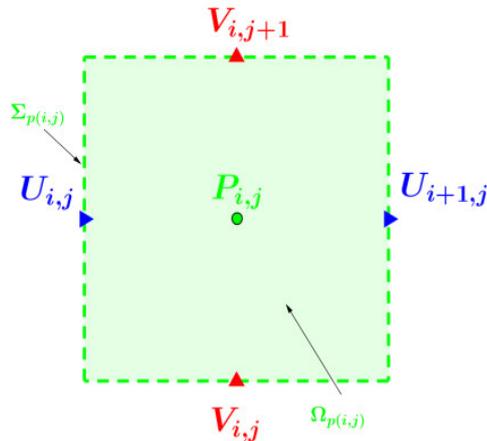


FIGURE 3.7 – Discrétisation de la divergence de la vitesse

$$F_{s,(i,j)} \approx \int_{\Omega_{\bar{u}(i,j)}} \sigma \nabla \cdot \bar{n} \nabla C \, d\Omega \quad (3.81)$$

L'inconvénient majeur de cette approche est la génération de courants parasites à proximité de l'interface. Quand la dynamique est dominée par la viscosité ou la gravité, les effets de la tension superficielle sont négligeables et les courants parasites sont insignifiants. Cependant, dans certaines situations, comme les instabilités de bulles, les effets capillaires sont dominants et les courants parasites générés peuvent perturber la solution numérique, et générer des déformations artificielles de l'interface. Pour contourner ce problème, une fonction couleur lissée notée C^s (voir Eq. (2.84)) est utilisée pour estimer les propriétés géométriques de l'interface, à savoir la normale \bar{n} et la courbure \varkappa .

Pour pouvoir discrétiser correctement le terme capillaire en utilisant la méthode CSF nous avons besoin de calculer la normale \bar{n} . Pour ce faire, nous utilisons directement la fonction couleur lissée C^s :

$$\bar{n} = \nabla \cdot \left(\frac{\nabla C^s}{\|\nabla C^s\|} \right) \quad (3.82)$$

Le gradient de la fonction couleur lissée $\frac{\partial C^s}{\partial x}$ est calculé sur les mêmes points de discrétisation que le champ de vitesse. La première composante est discrétisée comme schématisé sur la figure 3.8a :

$$\frac{\partial C^s}{\partial x} \approx \frac{C_{i,j}^s - C_{i-1,j}^s}{h} \quad (3.83)$$

De façon analogue, l'approximation de la deuxième composante $\frac{\partial C^s}{\partial y}$ est présentée sur la figure 3.8b :

$$\frac{\partial C^s}{\partial y} \approx \frac{C_{i,j}^s - C_{i,j-1}^s}{h} \quad (3.84)$$

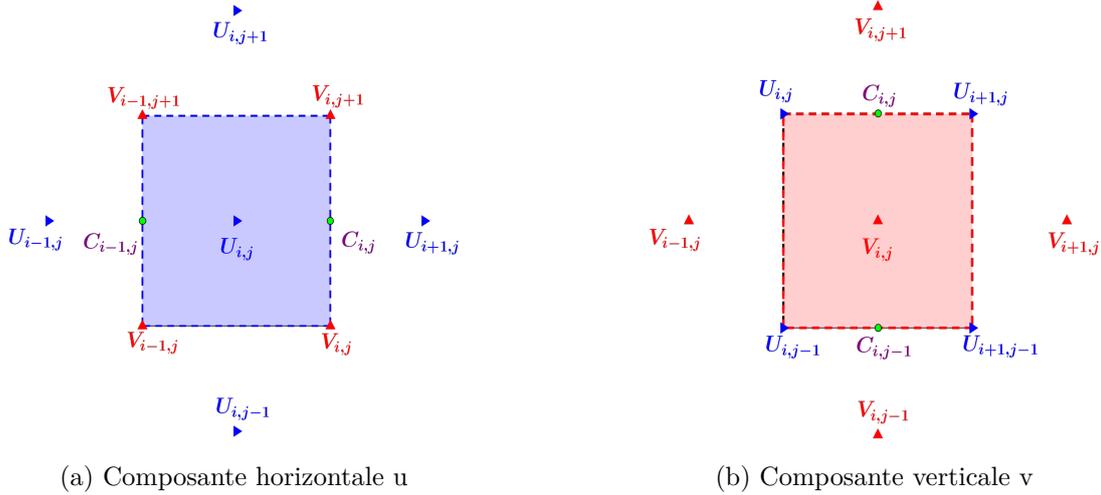


FIGURE 3.8 – Discrétisation du gradient de la fonction couleur

3.4.6 Traitement des conditions aux limites

Pour compléter la formulation du modèle à un fluide, les conditions aux limites associées au champ de vitesse \bar{u} sur la frontière du domaine peuvent être spécifiées en utilisant la méthode de pénalisation [137]. Cela offre un traitement de manière naturelle pour imposer différents types de conditions aux limites, telles que les conditions de Dirichlet ou/et de Neumann. Numériquement, cette méthode est basée sur l'utilisation d'un terme de contrôle $\bar{\bar{B}}_{penu} \cdot (f(\bar{u}^{n+1}) - \bar{u}_\infty)$, qui est ajouté à l'équation de bilan de la quantité de mouvement :

$$NS(\bar{u}^{n+1}) + \bar{\bar{B}}_{penu} \cdot (f(\bar{u}^{n+1}) - \bar{u}_\infty) = 0 \quad (3.85)$$

avec

$$\bar{\bar{B}}_{penu} = \begin{pmatrix} \alpha_u & 0 \\ 0 & \alpha_v \end{pmatrix} \quad (3.86)$$

qui est un tenseur d'ordre deux dont les composantes diagonales tendent vers l'infini le long de la frontière du domaine et sont identiquement nulles à l'intérieur du domaine. Ici, $f(\bar{u}^{n+1})$ est une fonction discrète de u^{n+1} et v^{n+1} , qui s'écrit sous la forme d'une combinaison linéaire des vitesses résolues $u_{i,j}^{n+1}$ et $v_{i,j}^{n+1}$.

$$f(u^{n+1}) = a_0 u_{i,j}^{n+1} + a_1 u_{i-1,j}^{n+1} + a_2 u_{i+1,j}^{n+1} + a_3 u_{i,j-1}^{n+1} + a_4 u_{i,j+1}^{n+1}, \quad (3.87)$$

$$f(v^{n+1}) = a_0 v_{i,j}^{n+1} + a_1 v_{i-1,j}^{n+1} + a_2 v_{i+1,j}^{n+1} + a_3 u_{i,j-1}^{n+1} + a_4 v_{i,j+1}^{n+1}. \quad (3.88)$$

Le traitement des conditions aux limites de Dirichlet, appliquées à la limite gauche par exemple (voir Fig. 3.9) est piloté par les coefficients a_i qui sont alors définis comme suit : $a_0 = a_2 = \frac{1}{2}$ et $a_1 = a_3 = a_4 = 0$. Ainsi, l'équation de la quantité de mouvement discrète à la limite gauche devient :

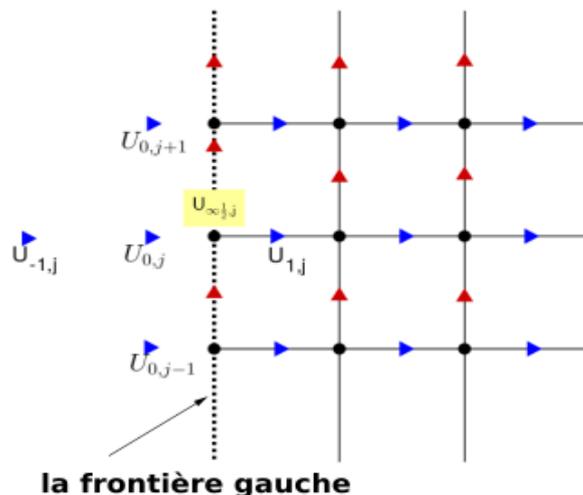


FIGURE 3.9 – Traitement des conditions aux limites de la frontière gauche en 2D

$$NS(u_{i,j}^{n+1}) + \alpha_u \left(\frac{1}{2}u_{0,j}^{n+1} + \frac{1}{2}u_{1,j}^{n+1} - u_{\infty\frac{1}{2},j} \right) = 0. \quad (3.89)$$

avec $u_{\infty\frac{1}{2},j} = 0$.

Pour imposer des conditions aux limites homogènes de type Neumann, les coefficients sont fixés à $a_0 = -1$, $a_2 = 1$ et $a_1 = a_3 = a_4 = 0$. L'équation résultante sous forme discrète s'écrit :

$$NS(u_{i,j}^{n+1}) + \alpha_u \left(-u_{0,j}^{n+1} + u_{1,j}^{n+1} - u_{\infty\frac{1}{2},j} \right) = 0. \quad (3.90)$$

avec $u_{\infty\frac{1}{2},j} = 0$.

Pour les autres parois (à gauche, à droite, en haut ou en bas du domaine), les coefficients a_i associés aux directions relatives (normale ou tangentielle) à la paroi sont donnés dans le tableau 3.2.

3.5 Conclusion

Le long de ce chapitre, nous avons retracé en détail les différents algorithmes dédiés au traitement du couplage vitesse-pression, en particulier, la méthode de projection-correction et celle du lagrangien augmenté, implémentées dans le code "FUGU".

Nous nous sommes ensuite attachés à la discrétisation des équations de conservation sur une grille décalée de type MAC. Une attention particulière a été portée à la pénalisation de l'équation de bilan de la quantité de mouvement afin d'imposer les conditions aux limites.

En perspective de ce chapitre, nous souhaitons développer une approche exacte monolithique qui

TABLE 3.2 – Tableau des valeurs des coefficients de pondération de la combinaison linéaire $f(\bar{u}^{n+1})$.

Condition limite	Position de la paroi							
	À gauche		À droite		En haut		En Bas	
	$\bar{u} \cdot \bar{n}$	$\bar{u} \cdot \bar{t}$						
Dirichlet	$a_0 = \frac{1}{2}$	$a_0 = 1$						
	$a_1 = 0$	$a_1 = 0$	$a_1 = \frac{1}{2}$	$a_1 = 0$				
	$a_2 = \frac{1}{2}$	$a_2 = 0$						
	$a_3 = 0$	$a_3 = 0$	$a_3 = 0$	$a_3 = 0$	$a_3 = \frac{1}{2}$	$a_3 = 0$	$a_3 = 0$	$a_3 = 0$
	$a_4 = 0$	$a_4 = \frac{1}{2}$	$a_4 = 0$					
Neumann	$a_0 = -1$		$a_0 = 1$		$a_0 = 1$		$a_0 = -1$	
	$a_1 = 0$		$a_1 = -1$		$a_1 = 0$		$a_1 = 0$	
	$a_2 = 1$		$a_2 = 0$		$a_2 = 0$		$a_2 = 0$	
	$a_3 = 0$		$a_3 = 0$		$a_3 = -1$		$a_3 = 0$	
	$a_4 = 0$		$a_4 = 0$		$a_4 = 0$		$a_4 = 1$	

fera l'objet du chapitre suivant. Cette approche nous permettra de traiter le couplage vitesse-pression ainsi que la simulation des problèmes raides à forts rapports de masses volumiques et de viscosités.

Chapitre 4

Mise en évidence de l'intérêt d'un solveur tout-couplé original 2D

Sommaire

4.1 Développement du solveur tout-couplé en 2D	98
4.2 Préconditionnement du système $A\bar{x} = \bar{b}$	102
4.2.1 Méthodes de factorisation incomplète ILU	104
4.2.2 Méthodes multigrilles	105
4.2.3 Décomposition de domaine de Schwarz	107
4.2.4 Bilan et préconditionneur retenu	108
4.3 Les techniques de préconditionnement par blocs	109
4.3.1 Les techniques de préconditionnement pour le bloc de vitesse	110
4.3.2 Les techniques de préconditionnement pour le complément de Schur	113
4.4 Validations sur des configurations monophasiques et diphasiques	116
4.4.1 Cavité entraînée	116
4.4.2 Écoulement de Poiseuille stationnaire	120
4.4.3 Goutte statique (loi de Laplace)	129
4.4.4 Montée d'une bulle dans un liquide	131
4.4.5 Chute d'une sphère dense dans l'air	135
4.4.6 Nappe liquide cisailée périodique	140
4.4.7 Simulation de l'instabilité d'un jet visqueux en 2D	141
4.5 Conclusion	145

Dans le chapitre 3 de ce manuscrit, nous avons porté une attention particulière aux algorithmes du traitement du couplage vitesse-pression dédiés à la résolution des équations de Navier-Stokes. Le code de calcul maison (FUGU) de l'équipe, qui existait déjà, repose uniquement sur deux méthodes de résolution : la projection standard de type GODA [81] et le lagrangien augmenté [180, 185]. À ce niveau, notre apport majeur réside essentiellement dans la mise en place d'une approche exacte. Elle sera basée sur le développement d'un solveur tout-couplé d'une part, et d'une

stratégie de préconditionnement par bloc d'autre part.

Au début de ce chapitre, nous allons décrire le solveur tout-couplé développé dans le cadre de cette thèse, ainsi que quelques préconditionneurs classiques et spécifiques par bloc destinés à la résolution des systèmes à point selle. Dans la suite du chapitre, nous allons nous attacher à la validation du solveur tout-couplé à travers différents cas test monophasiques et diphasiques. Ceci nous permettra de juger des capacités du solveur et de la qualité du couplage entre les différentes méthodes numériques utilisées dans le cadre de ce travail. Nous allons aussi comparer les performances du solveur par rapport aux méthodes déjà existantes, à savoir la méthode de projection standard et la méthode du lagrangien augmenté. L'apport majeur du solveur tout-couplé sur des écoulements diphasiques à forts rapports de masses volumiques et viscosités est également abordé.

4.1 Développement du solveur tout-couplé en 2D

À ce jour, le solveur tout-couplé, qui est très largement utilisé par la communauté des éléments finis dans le contexte de la simulation des écoulements stationnaires et monophasiques. À ce jour, cette approche n'a pas suscité beaucoup d'intérêt de la part de la communauté des différences finies et des volumes finis. L'objectif de ce travail est de développer un solveur tout-couplé efficace pour résoudre les équations de Navier-Stokes diphasiques et incompressibles. Ainsi, après une discrétisation appropriée via une approche de type volumes finis pénalisés, le système générique qui en résulte est un système linéaire non symétrique de type point selle qui prend la forme suivante :

$$\begin{pmatrix} F_u & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \bar{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \bar{f} \\ 0 \end{pmatrix} \quad (4.1)$$

où $F_u = M_u^{(\rho)} + N_u^{(\rho)} + L_u^{(\mu)}$, et enfin :

- $B_u \bar{u}^{n+1} \approx \int_{\Omega_{i,j,k}} \nabla \cdot \bar{u}^{n+1} dV$ est la partie discrète de la divergence,
- $B_p^T p^{n+1} \approx \int_{\Omega_{i,j,k}} \nabla p^{n+1} dV$ est l'opérateur de gradient de pression discret,
- $M_u^{(\rho)} \bar{u}^{n+1} \approx \int_{\Omega_{i,j,k}} \left(\frac{\rho^{n+1}}{\Delta t} \bar{u}^{n+1} + \overline{B_{penu} f}(\bar{u}^{n+1}) \right) dV$ est la matrice de masse,
- $L_u^{(\mu)} \bar{u}^{n+1} \approx - \int_{\Sigma_{i,j,k}} \overline{\mathbf{T}^{n+1}} \cdot \bar{n} dS$ est l'opérateur de contraintes visqueuses,
- $N_u^{(\rho)} \bar{u}^{n+1} \approx \int_{\Sigma_{i,j,k}} \rho \bar{u}^{n+1} \bar{u}^n \cdot \bar{n} dS$ est la partie discrète du terme convectif,
- $\bar{f} \approx \int_{\Omega_{i,j,k}} \left(\frac{\rho^n}{\Delta t} \bar{u}^n + \overline{B_{penu} \bar{u}_\infty} + \bar{F}_s \right) dV$ est le second membre associé à la vitesse.

Le système réduit peut être réécrit sous la forme $A\bar{x} = \bar{b}$, de N équations à N inconnues à coefficients dans \mathbb{R} . Dans ce système, A est une matrice carée d'ordre N , qui est associée aux

coefficients de discrétisation en temps et en espace et \bar{b} le second membre à valeurs dans \mathbb{R} associé à chaque composante du vecteur solution \bar{x} . Ainsi, la solution du problème diphasique s'écrit :

$$\bar{x} = A^{-1} \times \bar{b}. \quad (4.2)$$

Pour calculer $\bar{x} = (\bar{u}^{n+1}, p^{n+1})^T$, nous distinguons deux classes de méthodes de résolutions d'un système linéaire, les méthodes directes et les méthodes itératives, chacune adaptée à des types de matrices particulières (matrices symétriques, matrices pleines, matrices creuses, ...). Les méthodes directes basées sur une élimination de Gauss ou une décomposition LU [3, 7, 8, 15, 19, 20, 39] sont suffisamment robustes pour divers problèmes, puisqu'elles nécessitent un nombre fini d'opérations pour trouver la solution exacte. En revanche, elles sont plus coûteuses en temps et en mémoire. Lorsque la taille du système augmente elles deviennent impraticables pour des problèmes 3D, car elles nécessitent des matrices qui deviennent pleines lors de la résolution même si elles sont initialement creuses. L'alternative aux méthodes directes consiste en l'utilisation de solveurs itératifs, pour lesquels une approximation de la solution est générée. Parmi elles, on peut citer l'approche GMRES [154] (ou une de ses variantes, par exemple le RGMRES [174] ou le FGMRES [46]), les solveurs multigrilles géométriques [11, 14, 162], algébriques [131, 153, 175] ou bien hybrides [119] (géométrique-algébrique), les méthodes de décomposition de domaines [118, 121, 146, 170] et les méthodes de type gradient conjugué et ses variantes [49, 83, 94]. Il est donc nécessaire d'estimer la structure de notre système linéaire afin de choisir la méthode la plus judicieuse suivant la nature du système matriciel. Si on se réfère aux différents schémas de discrétisation proposés dans le chapitre précédent (schéma centré) pour évaluer les flux sur les faces des volumes de contrôle, on remarque que chaque équation du système linéaire ne couple que quelques composantes de la solution (5 pour le schéma centré en 2D, 4 pour le couplage entre u et v , 2 pour le gradient de pression et 4 pour la divergence du champ de vitesse). Pour un maillage de dimension $N_x \times N_y$, le vecteur solution comportera $(N_x + 1) \times N_y$ composantes u , $N_x \times (N_y + 1)$ composantes v et $N_x \times N_y$ composantes p , soit $N = N_x \times (3N_y + 1) + N_y$ composantes et A sera une matrice carée $N \times N$ coefficients. Il apparaît alors clairement que la matrice A générée par la discrétisation des équations de Navier-Stokes en formulation à un fluide sera très creuse, et seules 11 diagonales non-nulles interviendront dans la matrice si on discrétise par exemple les termes non-linéaires et les termes visqueux via un schéma centré (voir figure 4.1). Nous aurons donc intérêt à utiliser des solveurs itératifs, qui s'avèrent très utiles pour optimiser le stockage et limiter les opérations.

Nous avons opté pour l'utilisation d'une méthode itérative, dans laquelle on construit une suite $(x^{(k)})_{k \in \mathbb{N}}$ d'éléments de \mathbb{R}^N qui va converger (sous certaines conditions) vers la solution \bar{x} du système. Nous avons choisi d'utiliser une méthode itérative de type gradient conjugué BiCG-STAB(2) qui s'est montré plus efficace que le GMRES (10 % en moyenne) pour résoudre des systèmes vectoriels couplés de type lagrangien augmenté ou tout-couplé. Cette méthode a été privilégiée en dépit de ses propriétés de convergence non-monotone. Ces méthodes furent intro-

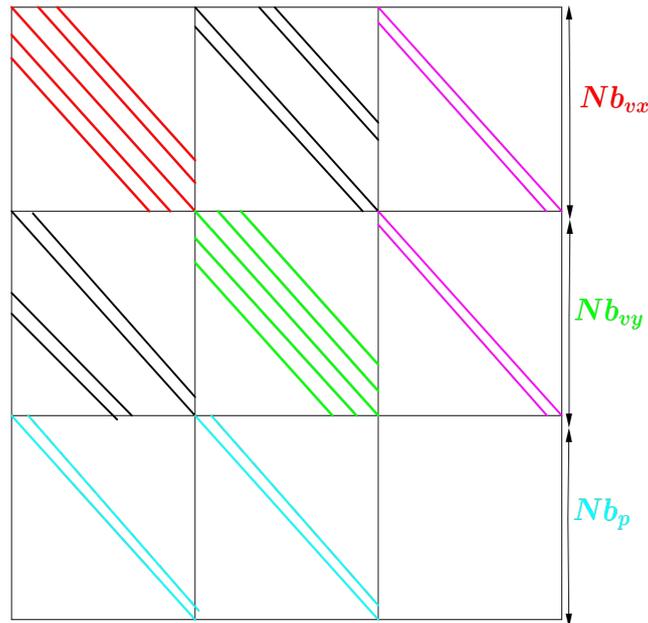


FIGURE 4.1 – Structure de la matrice tout-couplée provenant de la discrétisation des équations de conservation, 5 diagonales non nulles issues du schéma centré (en rouge pour la composante u , en vert pour la composante v), les diagonales en noir pour le couplage entre u et v , en magenta pour le gradient de pression et en cyan pour la divergence du champ de vitesse.

duites en 1952 par Hestenes et Stiefel [94], pour les matrices symétriques définies positives, et furent étendues à des matrices régulières quelconques par Van Der Vost [49]. Si on suppose que les conditions initiales (\bar{u}^0, p^0) du problème diphasique considérées sont définies, l'état initial est décrite par l'indice 0, l'itération à travers laquelle les valeurs des différents paramètres et vecteurs sont connus par k et l'étape courante par $k + 1$, l'algorithme de résolution correspondant est alors le suivant (Van Der Vorst [49]) :

```

1  ! L'initialisation
2       $k = 0$    !  $k \in \mathbb{N}^+$ , le nombre d'itération
3       $\bar{x}^{(0)} \in \mathbb{R}^N$ 
4       $\bar{q} = A\bar{x}^{(0)}$  !  $\bar{x}^{(0)} \in \mathbb{R}$  est la solution initiale
5       $\bar{r}^{(0)} = \bar{b} - \bar{q}$  ! le résidu initial du système préconditionné
6       $\tilde{r} = \bar{r}^{(0)}$  !  $\tilde{r}$  est un vecteur arbitraire, tel que  $\tilde{r} \cdot \bar{r}^{(0)} \neq 0$ 
7       $\xi_0 = \omega_2 = 1$ ;  $\alpha = 0$ ;  $\bar{u} = \bar{0}$ 
8       $\xi_0 = \omega_2 = 1$ ;  $\alpha = 0$ ;  $\bar{u} = \bar{0}$ 
9  while  $\frac{\|\bar{r}^{(k)}\|_2}{\|\bar{r}^{(0)}\|_2} > \varepsilon$  do
10      $k = k + 1$ 
11      $\xi_0 = -\omega_2 \xi_0$ 
12  ! Étape paire
13      $\xi_1 = \tilde{r} \cdot \bar{r}^{(k-1)}$ ;  $\beta = \alpha \xi_1 / \xi_0$ ;  $\xi_0 = \xi_1$  ! produit scalaire #1
14      $\bar{v} = A\bar{u}$  ! produit matrice-vecteur #1
15      $\gamma = \tilde{r} \cdot \bar{v}$ ;  $\alpha = \xi / \gamma$  ! produit scalaire #2

```

```

16       $\bar{r}^{(k)} = \bar{r}^{(k-1)} - \alpha \bar{v}$  ! combinaison linéaire #1
17       $\bar{s} = A\bar{r}^{(k)}$  ! produit matrice-vecteur #2
18       $\bar{x}^{(k)} = \bar{x}^{(k-1)} + \alpha \bar{u}$  ! combinaison linéaire #2
19 ! Étape impaire
20       $\xi_1 = \tilde{\bar{r}} \cdot \bar{s}$ ;  $\beta = \alpha \xi_1 / \xi_0$  ! produit scalaire #3
21       $\bar{v} = \bar{s} - \beta \bar{v}$  ! combinaison linéaire #3
22       $\bar{w} = A\bar{v}$  ! produit matrice-vecteur #3
23       $\gamma = \tilde{\bar{r}} \cdot \bar{w}$ ;  $\alpha = \xi_0 / \gamma$  ! produit scalaire #4
24       $\bar{u} = \bar{r}^{(k)} - \beta \bar{u}$  ! combinaison linéaire #4
25       $\bar{r}^{(k)} = \bar{r}^{(k)} - \alpha \bar{v}$  ! combinaison linéaire #5
26       $\bar{s} = \bar{s} - \alpha \bar{w}$  ! combinaison linéaire #6
27       $\bar{t} = A\bar{s}$  ! produit matrice-vecteur #4
28 ! GCR(2)
29       $\omega_1 = \bar{r}^{(k)} \cdot \bar{s}$  ! produit scalaire #5
30       $\lambda = \bar{s} \cdot \bar{s}$  ! produit scalaire #6
31       $\nu = \bar{s} \cdot \bar{t}$  ! produit scalaire #7
32       $\tau = \bar{t} \cdot \bar{t}$  ! produit scalaire #8
33       $\omega_2 = \bar{r}^{(k)} \cdot \bar{t}$  ! produit scalaire #9
34       $\tau = \tau - \nu^2 / \lambda$ ;  $\omega_2 = (\omega_2 - \nu \omega_1 / \lambda) / \tau$ ;  $\omega_1 = (\omega_1 - \nu \omega_2) / \lambda$ 
35       $\bar{x}^{(k)} = \bar{x}^{(k)} + \omega_1 \bar{r}^{(k)} + \omega_2 \bar{s} + \alpha \bar{u}$  ! combinaison linéaire #7
36       $\bar{r}^{(k)} = \bar{r}^{(k)} - \omega_1 \bar{s} - \omega_2 \bar{t}$  ! combinaison linéaire #8
37       $\bar{u} = \bar{u} - \omega_1 \bar{v} - \omega_2 \bar{w}$  ! combinaison linéaire #9
38      End While

```

Listing 4.1 – L’algorithme du BiCGSTAB(2)

où ϵ est un paramètre de contrôle de la convergence de l’algorithme BiCGSTAB(2) relatif au résidu de la solution à la $k^{\text{ième}}$ itération. Le défaut majeur de ce critère est qu’il n’est pas normalisé et par conséquent ne permet pas de comparer la précision des calculs dans des situations différentes. Une autre option efficace et très répandue pour imposer le critère d’arrêt, consiste à normaliser le résidu \bar{r}^k par la norme du résidu initial $\bar{r}^{(0)}$ et à définir le critère d’arrêt comme suit :

$$\frac{\|\bar{r}^k\|_2}{\|\bar{r}^{(0)}\|_2} \leq \epsilon \quad (4.3)$$

où encore par une adimensionalisation par la norme du seconde membre $\|\bar{b}\|$ tel que :

$$\frac{\|\bar{r}^k\|_2}{\|\bar{b}\|_2} \leq \epsilon. \quad (4.4)$$

Du fait de l’indépendance du nombre d’itérations pour converger en fonction de la valeur initiale de la solution $\bar{x}^{(0)}$, nous avons retenu le critère d’arrêt qui correspond à la normalisation de $\|\bar{r}^k\|_2$ par $\|\bar{b}\|_2$.

En pratique, on peut montrer que l'algorithme converge en au plus N itérations (N la dimension du système linéaire) et que le nombre d'itérations est proportionnel à la racine carrée du conditionnement de la matrice A . Ce conditionnement mesure la sensibilité du système aux erreurs d'arrondis et aux petites perturbations des données de problème (le second membre b et la matrice A) qui pilotent la stabilité de la solution, ce qui traduit la difficulté à résoudre le système $A\bar{x} = \bar{b}$. Dans les problèmes qui nous concernent, les coefficients de A , qui dépendent entre autre de la masse volumique ρ et de la viscosité μ , admettront des valeurs hétérogènes à l'interface entre les deux fluides. La matrice A sera donc très mal conditionnée, avec des coefficients extra-diagonaux qui pourront être grands et des coefficients diagonaux d'ordre de grandeur très différents. Finalement, l'efficacité du solveur BICSGTAB(2) peut être mise à mal par un mauvais conditionnement de la matrice A , d'où la nécessité d'introduire un préconditionneur pour accélérer la convergence du solveur [16, 189]. Le principe du preconditionnement ainsi que les différentes techniques issues de la littérature seront abordées plus en détail dans la prochaine section.

4.2 Préconditionnement du système $A\bar{x} = \bar{b}$

Comme nous l'avons déjà mentionné, la performance des solveurs itératifs dépend fortement du conditionnement du système matriciel $A\bar{x} = \bar{b}$ à résoudre, que l'on peut lier à la nature du problème étudié. Dans le contexte de la simulation des écoulements diphasiques instationnaire, de forts rapports de masses volumiques et de viscosités couplent fortement les composantes de vitesse, et compliquent sérieusement les choses pour obtenir un taux de convergence élevé. Il est donc nécessaire d'employer un preconditionnement pour accélérer la convergence du solveur itératif et d'éviter les éventuelles instabilités numériques. Une telle technique consiste à transformer le système original $A\bar{x} = \bar{b}$ en un autre équivalent en utilisant un preconditionnement à gauche comme suit :

$$\mathcal{P}^{-1}A\bar{x} = \mathcal{P}^{-1}\bar{b} \quad (4.5)$$

ou à droite par :

$$A\mathcal{P}^{-1}\bar{y} = \bar{b}, \quad \bar{y} = \mathcal{P}\bar{x}. \quad (4.6)$$

On peut aussi combiner ces deux types pour aboutir à un preconditionnement mixte, à gauche et à droite :

$$\mathcal{P}_G^{-1}A\mathcal{P}_D^{-1}\bar{x} = \mathcal{P}_G^{-1}\bar{b} \quad (4.7)$$

qui ont la même solution, où les matrices de preconditionnement \mathcal{P} , \mathcal{P}_G et \mathcal{P}_D sont des approximations de A ($\mathcal{P} \approx A$) et dont l'inverse peut être appliqué de manière efficace. Dans le cas idéal où $P = A$, le solveur BICGSTAB(2) converge en une itération. Mais, en pratique cela nécessite un coût en temps CPU trop important : c'est une méthode directe. Par ailleurs, prendre $\mathcal{P} = I$ permet de calculer \mathcal{P}^{-1} plus facilement, mais dans ce cas la méthode itérative est appliquée à la

matrice originale et donc, encore une fois, rien n'a été gagné. En pratique, le préconditionneur \mathcal{P} est choisit pour que le conditionnement de $\mathcal{P}^{-1}A$ soit meilleur que celui de A , et que l'application de l'inverse de P ne soit pas trop onéreuse en temps de calcul et en stockage mémoire. Pour le solveur BICGSTAB(2) préconditionné, chaque produit matrice-vecteur $A\bar{x} = \bar{z}$ sera suivi par le calcul de $\mathcal{P}^{-1}\bar{z} = \bar{y}$. La même transformation doit être appliquée au terme source $\mathcal{P}^{-1}\bar{b}$. Alors, pour résoudre le système $\mathcal{P}^{-1}A\bar{x} = \mathcal{P}^{-1}\bar{b}$ avec un critère d'arrêt choisi ϵ et en se donnant une solution initiale $\bar{x}^{(0)}$, le BiCGSTAB(2) préconditionné procède comme suit :

```

1  ! L'initialisation
2       $k = 0$  !  $k \in \mathbb{N}^+$ , le nombre d'itération
3       $\bar{x}^{(0)} \in \mathbb{R}^N$ 
4       $\bar{b} = \mathcal{P}^{-1}\bar{b}$  ! Préconditionnement est appliqué au second membre
5       $\bar{z} = A\bar{x}^{(0)}$  !  $\bar{x}^{(0)} \in \mathbb{R}$  est la solution initiale
6       $\bar{q} = \mathcal{P}^{-1}\bar{z}$ 
7       $\bar{r}^{(0)} = \bar{b} - \bar{q}$  ! le résidu initial du système préconditionné
8       $\tilde{\bar{r}} = \bar{r}^{(0)}$  !  $\tilde{\bar{r}}$  est un vecteur arbitraire, tel que  $\tilde{\bar{r}} \cdot \bar{r}^{(0)} \neq 0$ 
9       $\xi_0 = \omega_2 = 1$ ;  $\alpha = 0$ ;  $\bar{u} = \bar{0}$ 
10      $\xi_0 = \omega_2 = 1$ ;  $\alpha = 0$ ;  $\bar{u} = \bar{0}$ 
11 while  $\frac{\|\bar{r}^{(k)}\|_2}{\|\bar{b}\|_2} > \epsilon$  do
12      $k = k + 1$ 
13      $\xi_0 = -\omega_2\xi_0$ 
14 ! Étape paire
15      $\xi_1 = \tilde{\bar{r}} \cdot \bar{r}^{(k-1)}$ ;  $\beta = \alpha\xi_1/\xi_0$ ;  $\xi_0 = \xi_1$  ! produit scalaire #1
16      $\bar{z} = A\bar{u}$  ! produit matrice-vecteur #1
17      $\bar{v} = \mathcal{P}^{-1}\bar{z}$  ! preconditionneur #1
18      $\gamma = \tilde{\bar{r}} \cdot \bar{v}$ ;  $\alpha = \xi/\gamma$  ! produit scalaire #2
19      $\bar{r}^{(k)} = \bar{r}^{(k-1)} - \alpha\bar{v}$  ! combinaison linéaire #1
20      $\bar{z} = A\bar{r}^{(k)}$  ! produit matrice-vecteur #2
21      $\bar{s} = \mathcal{P}^{-1}\bar{z}$  ! preconditionneur #2
22      $\bar{x}^{(k)} = \bar{x}^{(k-1)} + \alpha\bar{u}$  ! combinaison linéaire #2
23 ! Étape impaire
24      $\xi_1 = \tilde{\bar{r}} \cdot \bar{s}$ ;  $\beta = \alpha\xi_1/\xi_0$  ! produit scalaire #3
25      $\bar{v} = \bar{s} - \beta\bar{v}$  ! combinaison linéaire #3
26      $\bar{z} = A\bar{v}$  ! produit matrice-vecteur #3
27      $\bar{w} = \mathcal{P}^{-1}\bar{z}$  ! preconditionneur #3
28      $\gamma = \tilde{\bar{r}} \cdot \bar{w}$ ;  $\alpha = \xi_0/\gamma$  ! produit scalaire #4
29      $\bar{u} = \bar{r}^{(k)} - \beta\bar{u}$  ! combinaison linéaire #4
30      $\bar{r}^{(k)} = \bar{r}^{(k)} - \alpha\bar{v}$  ! combinaison linéaire #5
31      $\bar{s} = \bar{s} - \alpha\bar{w}$  ! combinaison linéaire #6
32      $\bar{z} = A\bar{s}$  ! produit matrice-vecteur #4

```

```

33       $\bar{t} = \mathcal{P}^{-1}\bar{z}$  ! preconditionneur #4
34 ! GCR(2)
35       $\omega_1 = \bar{r}^{(k)} \cdot \bar{s}$  ! produit scalaire #5
36       $\lambda = \bar{s} \cdot \bar{s}$  ! produit scalaire #6
37       $\nu = \bar{s} \cdot \bar{t}$  ! produit scalaire #7
38       $\tau = \bar{t} \cdot \bar{t}$  ! produit scalaire #8
39       $\omega_2 = \bar{r}^{(k)} \cdot \bar{t}$  ! produit scalaire #9
40       $\tau = \tau - \nu^2/\lambda$ ;  $\omega_2 = (\omega_2 - \nu\omega_1/\lambda)/\tau$ ;  $\omega_1 = (\omega_1 - \nu\omega_2)/\lambda$ 
41       $\bar{x}^{(k)} = \bar{x}^{(k)} + \omega_1\bar{r}^{(k)} + \omega_2\bar{s} + \alpha\bar{u}$  ! combinaison linéaire #7
42       $\bar{r}^{(k)} = \bar{r}^{(k)} - \omega_1\bar{s} - \omega_2\bar{t}$  ! combinaison linéaire #8
43       $\bar{u} = \bar{u} - \omega_1\bar{v} - \omega_2\bar{w}$  ! combinaison linéaire #9
44 End While

```

Listing 4.2 – L’algorithme du BiCGSTAB(2) preconditionné

Nous en arrivons au coeur du problème qui est le choix du preconditionneur \mathcal{P} . Trouver un bon preconditionneur est une question fort délicate et reste une tâche non triviale. Néanmoins, l’investigation mathématique et une analyse approfondie sont des outils indispensables pour l’élaboration d’un bon preconditionneur. Un passage en revue de la littérature sur ce sujet a permis de mettre en exergue l’existence de plusieurs techniques. Parmi elles, on peut citer les méthodes de factorisation de Gauss incomplète LU ou MILU et leur généralisation ILU(p) [155], les méthodes multigrilles algébriques ou géométriques [11, 14, 119, 131, 153, 162, 175] en encore les méthodes de décomposition de domaines [118, 121, 146, 170]. Nous donnons ci-dessous une brève description de ces techniques.

4.2.1 Méthodes de factorisation incomplète ILU

La méthode de factorisation de Gauss incomplète LU compte parmi les méthodes de preconditionnement les plus classiques pour les matrices creuses. Celles-ci reposent sur une décomposition de la forme $A = LU - R$ où L est la matrice creuse triangulaire inférieure et U la matrice creuse triangulaire supérieure ont des coefficients non nuls aux mêmes emplacements que la matrice A (remplissage prédéfini), et R est le résidu ou l’erreur de la factorisation. Pour les matrices avec un stencil à 5 points, issues de la discrétisation des équations aux dérivées partielles elliptiques par exemple, la figure 4.2 illustre une factorisation ILU(0) de la matrice A .

D’une manière générale, l’algorithme ci-dessous représente la construction des matrices triangulaires L et U à partir de A :

```

1 ! Données :  $A$  la matrice à factoriser,  $N$  la taille de la matrice
2 Pour  $k = 1$  à  $N$  faire
3     Pour  $i = k + 1$  à  $N$  faire
4          $A(i, k) = A(i, k) / A(k, k)$ 
5         Pour  $j = k + 1$  à  $N$  faire

```

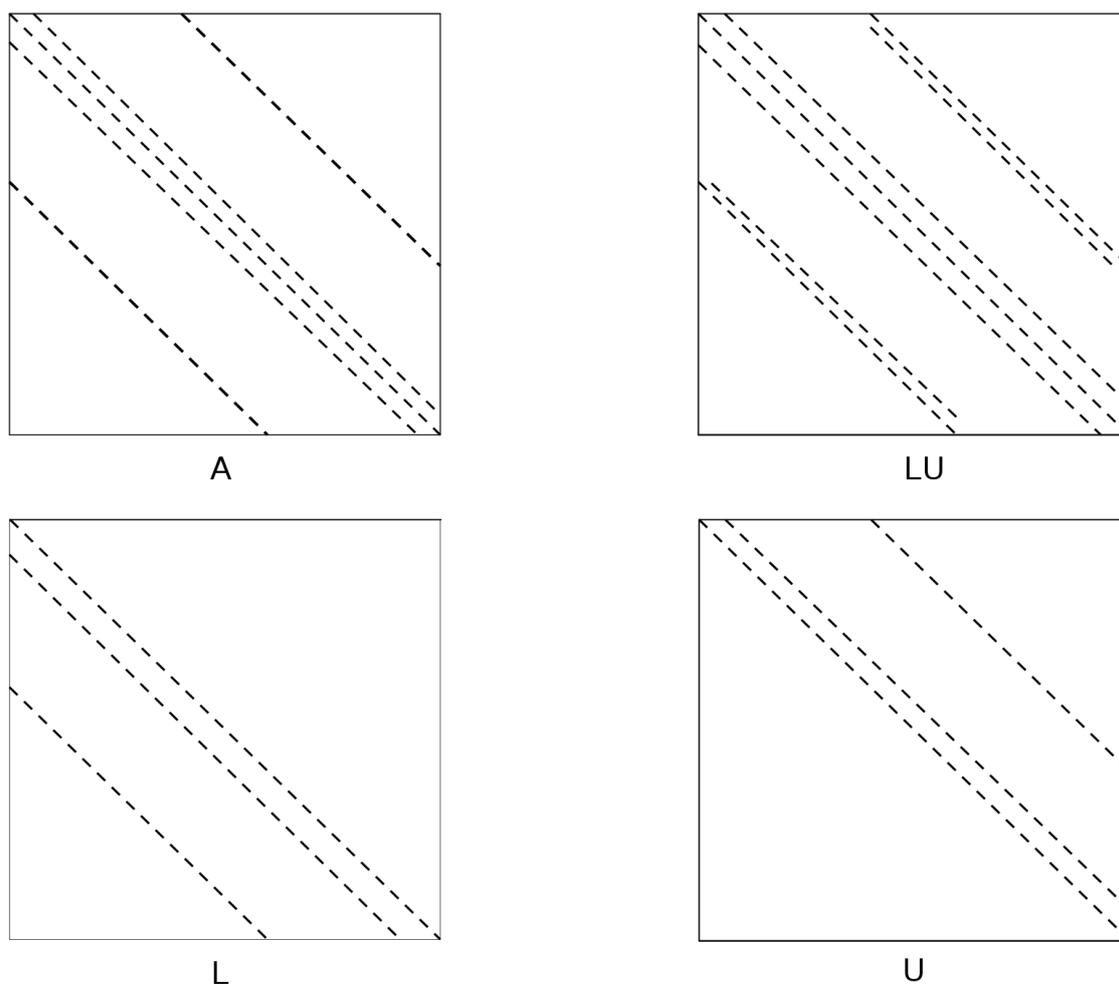


FIGURE 4.2 – La factorisation $ILU(0)$ pour une matrice à un stencil de cinq points.

```

6           A(i , j)=A(i , j)-A(i , k)×A(k , j)
7           end
8     end
9 end

```

Listing 4.3 – Factorisation Incomplète LU

4.2.2 Méthodes multigrilles

Les méthodes multigrilles exploitent la discrétisation des équations aux dérivées partielles en utilisant une hiérarchie de grilles. L'objectif principal est d'obtenir une convergence optimale du solveur itératif, essentiellement avec un cycle V ou W . Ces cycles multigrilles sont schématisés sur la figure 4.3 par itération du préconditionneur multigrille.

Dans la littérature, nous trouvons deux grandes familles de multigrilles : d'une part, les multigrilles algébriques qui reposent uniquement sur la matrice du système à résoudre. D'autre part, les multigrilles géométriques qui nécessitent de générer des maillages grossiers, souffrent de la

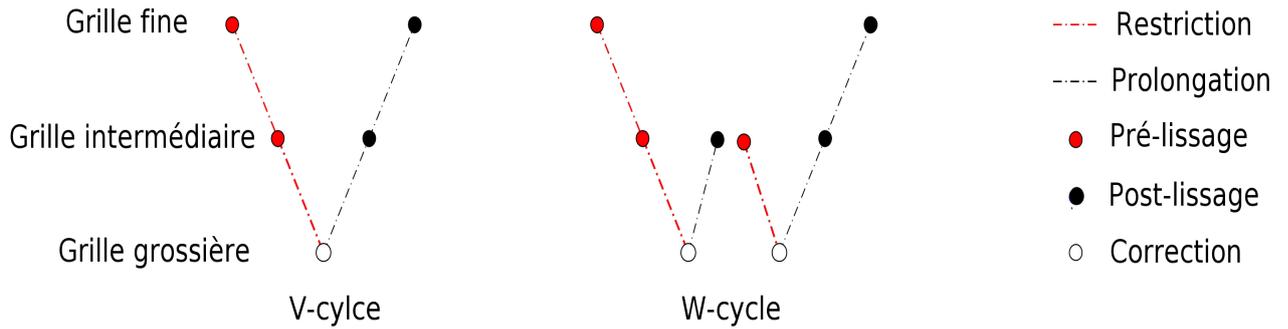


FIGURE 4.3 – Schémas de V-cycle et W-cycle des méthodes multigrilles

construction de ces maillages notamment quand ils sont non structurés ou encore lorsque la géométrie du problème est complexe. Hormis ces différences, il existe des caractéristiques communes entre ces deux techniques multigrilles. La figure 4.3 en présente certaines telles que le lissage, en l'occurrence la méthode Jacobi ou le Red-Black Gauss-Seidel qui réduit rapidement les hautes fréquences, l'opération de restriction R , qui consiste à projeter le résidu sur la grille grossière, l'opération de prolongation P , qui calcule le résidu sur la grille fine et la correction pour obtenir la solution améliorée.

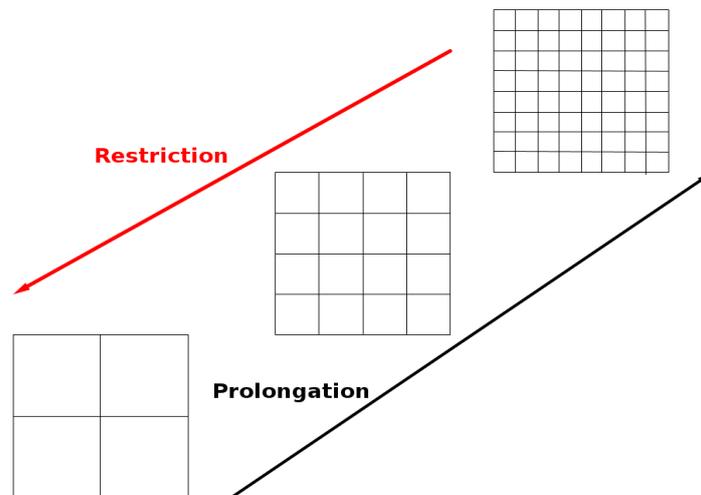


FIGURE 4.4 – Passages entre les différentes grilles : prolongation et restriction.

En pratique, lorsqu'on souhaite résoudre le système linéaire $P\bar{y} = \bar{z}$, la mise en oeuvre de la méthode multigrille à deux niveaux s'établit comme ci-dessous :

- 1 ! Données : P_h la matrice du système sur la grille fine de taille h
- 2 ! 1) Pré-lissage
- 3 on résout $P_h \bar{y}_h = \bar{z}_h$ sur la grille fine de taille h , en itérant
- 4 quelques itérations de Jacobi ou Red-Black Gauss-Seidel pour
- 5 obtenir une approximation \hat{y}_h
- 6 ! 2) Calcul du résidu \bar{r}_h

```

7    $\bar{r}_h = \bar{z}_h - P_h \bar{y}_h$ 
8   ! 3) Restriction
9   on restreint le résidu sur la grille grossière de taille  $\frac{h}{2}$  pour
10  obtenir  $\bar{z}_{h/2}$ . En parallèle, nous définissons la matrice  $P_{h/2}$  sur la
    grille grossière de taille  $\frac{h}{2}$ .
11  ! 4) Pré-lissage
12  on résout  $P_{h/2} \bar{y}_{h/2} = \bar{z}_{h/2}$  sur la grille grossière de taille  $\frac{h}{2}$  avec
13  précision, en itérant quelques itérations de Jacobi ou Red-Black
14  Gauss-Seidel.
15  ! 6) Prolongation
16  On prolonge  $\bar{y}_{h/2}$  sur la grille fine de taille  $h$  pour
17  obtenir la correction  $\bar{y}^C$ .
18  ! 7) Correction
19  On corrige la solution  $\bar{y}'_h = \hat{y}_h + \bar{y}^C$ 
20  ! 8) Post-lissage
21  On termine le cycle, par quelques itérations de Jacobi ou
22  Red-Black Gauss-Seidel, en prenant  $\bar{y}'_h$  comme condition initiale
23  pour trouver  $\bar{y}_h$ , tel que  $P_h \bar{y}_h = \bar{z}_h$ .

```

Listing 4.4 – Mise en oeuvre pratique d'une méthode multigrille à deux niveaux

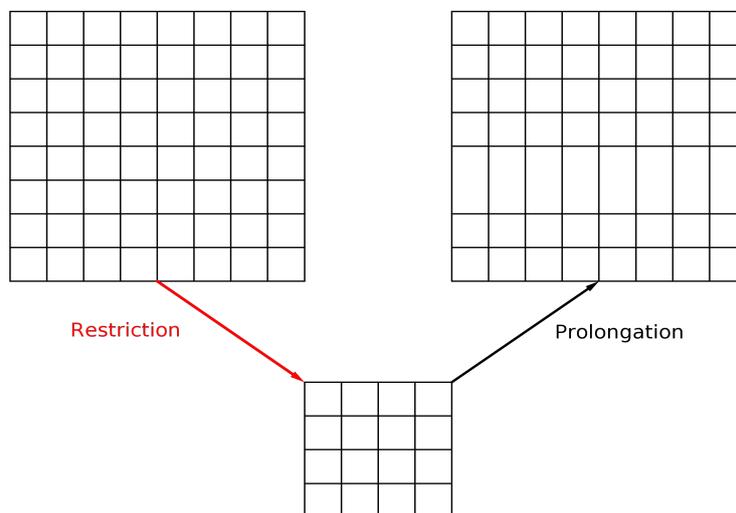


FIGURE 4.5 – Configuration à deux grilles, cycle en V.

4.2.3 Décomposition de domaine de Schwarz

Les méthodes de décomposition de domaine type Schwarz font référence à un ensemble de techniques qui sont articulées autour du principe de partitionnement. Ces méthodes ont été principalement développées pour résoudre des équations aux dérivées partielles sur des domaines en deux ou trois dimensions.

Considérons le problème de la résolution de l'équation de Laplace $\Delta T = 0$ sur un domaine Ω partitionné en deux sous-domaines disjoints Ω_1 et Ω_2 , séparés par une interface commune Γ comme le montre la figure 4.6. Les méthodes de décomposition de domaine tentent de résoudre le problème sur l'ensemble du domaine Ω à partir de solutions des problèmes sur les sous-domaines Ω_1 et Ω_2 en faisant un raccord à l'interface Γ .

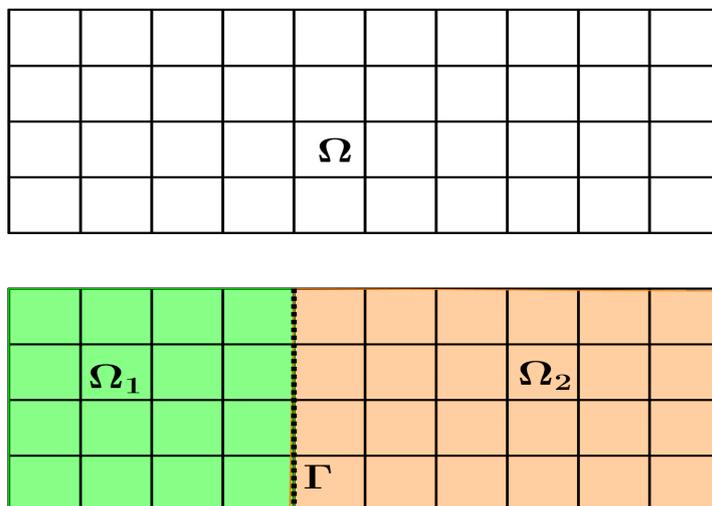


FIGURE 4.6 – Découpage du domaine Ω en 2 sous-domaines disjoints Ω_1 et Ω_2 sans recouvrement.

L'algorithme de Schwarz le plus fondamental consiste en un procédé itératif. Il repose sur la génération d'une suite de solutions T_1^n sur Ω_1 et T_2^n sur Ω_2 de la façon suivante :

```

1  ! Initialisations : Choisir  $T_1^0 \subset \Omega_1$  et  $T_2^0 \subset \Omega_2$ 
2  ! Pour tout  $n \geq 0$ , effectuer de façon successive :
3  ! Calcul de  $T_1^{n+1} \subset \Omega_1$ 
4       $\Delta T_1^{n+1} = 0$  sur  $\Omega_1$ 
5       $T_1 = 0$  sur  $\partial\Omega \cap \partial\Omega_1$ 
6       $T_1^{n+1} = T_2^n$  sur  $\Gamma$ 
7  ! Calcul de  $T_2^{n+1} \subset \Omega_2$ 
8       $\Delta T_2^{n+1} = 0$  sur  $\Omega_2$ 
9       $T_2 = 0$  sur  $\partial\Omega \cap \partial\Omega_2$ 
10      $T_2^{n+1} = T_1^{n+1}$  sur  $\Gamma$ 
11  ! Le critère d'arrêt est satisfait
12     Lorsque  $T_1^n$  et  $T_2^n$  sont suffisamment proches sur  $\Gamma$ 

```

Listing 4.5 – Procédé itératif de décomposition de domaine

4.2.4 Bilan et préconditionneur retenu

Malgré leur coût faible, l'inconvénient majeur des algorithmes de factorisation incomplète ILU réside dans leur robustesse en parallèle. En effet, ces méthodes sont souvent utilisées en tant

que préconditionneurs locaux et parallélisent mal vu leur dépendance globale à la structure algébrique du système. Les autres techniques de préconditionnements à savoir les méthodes multi-grilles et celles de décomposition de domaine possèdent un taux de convergence indépendant de la discrétisation et se parallélisent mieux, conduisant à des coûts de calcul qui se rapprochent de la limite théorique (proportionnels à la taille du problème). Cependant, elles sont très utilisées lorsque la matrice A est issue d'une discrétisation d'EDP scalaire. Dans le cadre des équations de Navier-Stokes, et comme une résolution tout-couplée est visée, nous serons amenés à aborder le problème du point de vue algébrique par des préconditionnements spécifiques, destinés à la résolution de points selles. Cette voie purement algébrique consiste à construire un préconditionneur par bloc, c'est à dire qui agit indépendamment sur les variables du problème (u , v et p). Nous nous orientons donc vers une l'utilisation de ces techniques, inspirées des travaux de Niall [21] que nous enrichissons par le développement d'un nouveau préconditionneur original pour le bloc de vitesse en se basant sur des considérations algébriques. Nous verrons dans la section suivante 4.3 plus de détails sur le choix et l'implémentation de ces techniques.

4.3 Les techniques de préconditionnement par blocs

Dans cette section, on anticipe la présentation des techniques du préconditionnement par blocs, reposant sur une factorisation algébrique de la matrice A . L'objectif sera donc se ramener à un système résolvable $\mathcal{P}^{-1}A\bar{x} = \mathcal{P}^{-1}\bar{b}$, qui allie rapidité et précision. Nous allons donc en premier lieu présenter la décomposition par blocs de la matrice A . Puis, nous déduisons la construction exacte du préconditionneur \mathcal{P} en se basant sur cette factorisation. Ensuite, nous exposons les différents choix pour élaborer une estimation du bloc de vitesse, en particulier l'approche que nous avons développée. Enfin, nous attachons aux différentes approximations utilisées dans la littérature pour approcher le complément de Schur, lorsqu'on fait face à la résolution des équations de Navier-Stokes.

Afin de construire un préconditionneur \mathcal{P} approprié, nous considérons la décomposition \mathcal{LDU} par blocs de la matrice initiale A , donnée par :

$$\begin{pmatrix} F_u & B_p^T \\ B_u & 0 \end{pmatrix} = \begin{pmatrix} I_u & 0 \\ B_u F_u^{-1} & I_p \end{pmatrix} \begin{pmatrix} F_u & 0 \\ 0 & S_p \end{pmatrix} \begin{pmatrix} I_u & F_u^{-1} B_p^T \\ 0 & I_p \end{pmatrix} = \mathcal{LDU} \quad (4.8)$$

Ici $S_p = -B_u F_u^{-1} B_p^T$ est le complément de Schur du bloc de pression et F_u est l'opérateur de convection-diffusion augmenté de la matrice de masse. Dans ce travail, nous utilisons le préconditionneur triangulaire par blocs qui figure parmi les préconditionneurs les plus populaires pour les problèmes de point selle. Pour ce faire, nous prenons le produit \mathcal{DU} de l'équation (I.8) :

$$\mathcal{P} = \begin{pmatrix} F_u & B_p^T \\ 0 & S_p \end{pmatrix}. \quad (4.9)$$

Par conséquent, l'application du préconditionneur de l'équation (I.9) implique la résolution du système suivant, où les composantes sont trouvées une par une par des substitutions :

$$\begin{pmatrix} F_u & B_p^T \\ 0 & S_p \end{pmatrix} \begin{pmatrix} \bar{r}_u \\ \bar{r}_p \end{pmatrix} = \begin{pmatrix} \bar{z}_u \\ \bar{z}_p \end{pmatrix} \quad (4.10)$$

où $(\bar{r}_u, \bar{r}_p)^T = \bar{b}, \bar{q}, \bar{v}^k, \bar{s}^k, \bar{t}^k$ et \bar{w}^k sont une séquence de vecteurs générés à chaque itération du BiCGSTAB(2) (pour plus de détails, le lecteur est invité à se reporter à l'algorithme 4.2).

Ainsi, lorsque l'inverse du préconditionneur triangulaire P^{-1} est appliqué à la matrice A , le système préconditionné $P^{-1}A$ possède une seule valeur propre $\lambda = 1$ de multiplicité $Nb_{vx} + Nb_{vy} + Nb_p$. Le polynôme caractéristique de $P^{-1}A$ est de degré 2, ce qui garantit la convergence d'un solveur de sous-espace de Krylov tel que GMRES en deux itérations. En effet, en considérant le problème des valeurs propres pour le système préconditionné $P^{-1}A$, ces valeurs peuvent facilement être obtenues à partir du problème généralisé des valeurs propres suivant :

$$\begin{pmatrix} F_u & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \bar{u} \\ p \end{pmatrix} = \lambda \begin{pmatrix} F_u & B_p^T \\ 0 & S_p \end{pmatrix} \begin{pmatrix} \bar{u} \\ p \end{pmatrix} \quad (4.11)$$

A partir de la première ligne de l'équation (4.11) on obtient $(1 - \lambda)(F_u \bar{u} + B_p^T p) = 0$, admettant deux solutions possibles. La première solution est une valeur propre $\lambda = 1$ de multiplicité $Nb_{vx} + Nb_{vy}$, et la seconde correspond à $\bar{u} = -F_u^{-1} B_p^T p$. Ensuite, la substitution de \bar{u} dans la deuxième ligne de l'équation (4.11) conduit à $-B_u F_u^{-1} B_p^T p = \lambda S_p p$, ce qui donne la deuxième valeur propre $\lambda = 1$ de multiplicité Nb_p . De même, si à la place du préconditionneur triangulaire supérieur $\mathcal{P} = \mathcal{DU}$, on se sert du préconditionneur bloc-diagonal D , $\lambda = 1$ apparaît comme une valeur propre de multiplicité au moins $n_u - n_p$ pour tout vecteur propre propre $[\bar{u}^T, \bar{0}^T]^T$ avec $B\bar{u} = 0$, alors que $\lambda = \frac{1}{2} \pm \frac{\sqrt{5}}{2}$ de multiplicité n_p comme cela est décrit dans [190].

Puisque la convergence dépend des valeurs propres λ du système préconditionné (plus les valeurs propres sont petites, plus la convergence est rapide), on s'attend à ce que le préconditionneur triangulaire par blocs soit plus performant et il est par conséquent retenu pour ce travail. Cette approche a cependant un inconvénient majeur, qui est que l'inversion des blocs S_p et F_u ne peut pas être effectuée dans des calculs réalistes. Elle est en effet plus coûteuse que la résolution du problème du point selle (5.6) par des méthodes directes. Par conséquent, l'inversion des blocs S_p et F_u doit être approximée par des matrices \hat{S}_p et \hat{F}_u . Cette approximation est discutée dans la section suivante.

4.3.1 Les techniques de préconditionnement pour le bloc de vitesse

Dans le cadre des équations de Navier-Stokes, F_u est l'opérateur de convection-diffusion pour lequel des méthodes connues de résolution approchée peuvent être appliquées. Les bon choix de \hat{F}_u pour approcher F_u s'avère décisif pour construire un préconditionneur robuste. En effet, dans

le cadre du lagrangien augmenté ou tout-couplé, ce bloc de vitesse est mal conditionné, ce qui soulève des difficultés considérables, tel que le couplage entre les composantes de la vitesse et les forts rapports de masses volumiques et de viscosités lorsque des écoulements multiphasiques sont en jeu.

Dans la littérature il existe une variété de préconditionneurs, par exemple le préconditionnement de type ILU [37, 107, 155, 194], également appelé factorisation incomplète de Gauss, qui permet d’atteindre de faibles résidus lorsque plusieurs millions d’inconnues sont impliquées. Ces préconditionnements ne sont pas facilement parallélisables car ils induisent une dépendance globale sur les inconnues, et nécessitent donc des échanges nombreux et répétés dans un parallélisme distribué. Il est donc possible d’utiliser les variantes d’ILU dans une implémentation par blocs, c’est-à-dire que le préconditionnement est tronqué et n’utilise que les variables disponibles sur le processeur. Néanmoins, dans la plupart des cas représentatifs, les préconditionneurs classiques de la littérature tels que ILU et leurs variantes ne permettent pas de résoudre efficacement les équations de Navier-Stokes entièrement couplée. D’autres techniques multigrilles [78, 124, 131, 188, 191, 192] spécialisées ont été proposées pour les systèmes de points de selle couplés dans les écoulements incompressibles (ainsi que pour d’autres applications). Dans ces méthodes multigrilles, une séquence de grilles est utilisée : en partant de la grille fine, sur laquelle on souhaite calculer la solution, jusqu’à une grille plus grossière, où une solution exacte ou approximative est beaucoup moins coûteuse. Une autre classe de méthodes à plusieurs niveaux est celle des méthodes de décomposition de domaine [118, 121, 146, 170], qui sont le plus souvent utilisées comme préconditionneurs pour accélérer la convergence du solveur itératif. Ces approches multi-niveaux, que ce soit multigrille ou décomposition de domaine sont largement utilisées, surtout dans le cas des équations elliptiques. L’intérêt de leur utilisation réside dans le fait de procurer un taux de convergence indépendant de la discrétisation, conduisant à des coûts de calcul proportionnels à la taille du problème.

4.3.1.1 Préconditionneur triangulaire pour le bloc de vitesse F_u

Notre objectif ici est de développer des approximations plus efficaces, essentiellement basées sur des considérations algébriques, et qui utilisent le complément de Schur. Pour ce faire, nous allons aborder les différentes manières d’approximer le bloc de vitesse F_u . Ainsi, nous détaillons également les aspects techniques incluant la factorisation du bloc de vitesse \hat{F}_u mais aussi le calcul de l’action de \hat{F}_u^{-1} sur un vecteur quelconque \bar{z} .

Considérons la matrice creuse et non symétrique F_u , résultant de la discrétisation des équations de bilan de quantité de mouvement. Cette matrice peut être scindée en blocs de matrices 2×2 comme suit :

$$F_u = \begin{pmatrix} F_{uu} & F_{uv} \\ F_{vu} & F_{vv} \end{pmatrix} \quad (4.12)$$

Le préconditionnement envisagé dans notre cas exploite la décomposition LU de l’équation (4.12),

en prenant le bloc triangulaire supérieur U :

$$\tilde{F}_u = \begin{pmatrix} F_{uu} & F_{uv} \\ 0 & F_{vv} - F_{vu}F_{uu}^{-1}F_{uv} \end{pmatrix} \quad (4.13)$$

où $F_{vv} - F_{vu}F_{uu}^{-1}F_{uv}$ est le complément idéal de Schur de F_{uu} dans F_u , qui peut être réécrit comme ci-dessous en définissant une nouvelle matrice $G = F_{vv}^{-1}F_{vu}F_{uu}^{-1}F_{uv}$:

$$F_{vv} - F_{vu}F_{uu}^{-1}F_{uv} = F_{vv} \left(I - F_{vv}^{-1}F_{vu}F_{uu}^{-1}F_{uv} \right) = F_{vv} (I - G). \quad (4.14)$$

Par conséquent, l'action de l'inverse de \tilde{F}_u sur un vecteur standard $\bar{z} = (z_u, z_v)^T$ doit être effectuée par une substitution élémentaire. Ceci peut être réalisé en résolvant deux systèmes linéaires et en effectuant un produit matrice-vecteur. L'algorithme correspondant 4.1 est le suivant :

Le preconditionneur triangulaire par bloc pour le bloc de vitesse F_u	
1	!Solve $F_{vv}(I - G)\bar{r}_v = \bar{z}_v$
2	!Update $\bar{z}_u \leftarrow \bar{z}_u - F_{uv}\bar{r}_v$
3	!Solve $F_{uu}\bar{r}_u = \bar{z}_u$

TABLE 4.1 – Approximation de l'action de l'inverse \tilde{F}_u sur un vecteur $\bar{z} = (z_u, z_v)^T$, le vecteur solution est $\bar{y} = (r_u, r_v)^T$.

Concernant l'application de l'inverse du bloc $F_{vv}(I - G)$ sur un vecteur \bar{z} , on suppose que les matrices $(I - G)$ et F_{vv} sont inversibles, et que l'inverse $(I - G)^{-1}$ peut être développé comme une série infinie des différents termes qui sont présents dans l'opérateur $(I - G)$. En utilisant le développement en série de Neumann suivant [123] :

$$(I - G)^{-1} = I + G + G^2 + \dots + G^m \approx \sum_{k=0}^{\infty} G^k \quad (4.15)$$

qui peut s'écrire comme suit :

$$(I - G)^{-1} \approx \sum_{k=0}^{\infty} \left(F_{vv}^{-1}F_{vu}F_{uu}^{-1}F_{uv} \right)^k \quad (4.16)$$

Ainsi, l'inverse du bloc $F_{vv}(I - G)$ peut s'écrire comme :

$$[F_{vv}(I - G)]^{-1} = (I - G)^{-1}F_{vv}^{-1} \approx \left[\sum_{k=0}^{\infty} (F_{vv}^{-1}F_{vu}F_{uu}^{-1}F_{uv})^k \right] F_{vv}^{-1} \quad (4.17)$$

À l'ordre 1, un preconditionneur triangulaire par bloc se décline comme suit

$$[F_{vv}(I - G)]^{-1} \approx \left(I + F_{vv}^{-1}F_{vu}F_{uu}^{-1}F_{uv} \right) F_{vv}^{-1} \quad (4.18)$$

En remplaçant (5.16) dans la première étape de l'algorithme 4.1, l'action de l'inverse du bloc

$F_{vv}(I - G)$ sur le vecteur \bar{z}_v s'effectuera suivant l'algorithme 4.2.

Le preconditionneur triangulaire par bloc pour le bloc de vitesse F_u	
1	!Solve $F_{vv}\bar{r}_{v1} = \bar{z}_v$
2	!Compute $\bar{r}_{r1}^* = F_{uv}\bar{r}_{v1}$
3	!Solve $F_{uu}\bar{r}_{u1} = \bar{r}_{r1}^*$
4	!Compute $\bar{r}_{v1}^* = F_{vu}\bar{r}_{u1}$
5	!Solve $F_{vv}\bar{r}_{v2} = \bar{r}_{v1}^*$
6	!Compute $\bar{r}_v = \bar{r}_{v1} + \bar{r}_{v2}$

TABLE 4.2 – Approximation de l'action de l'inverse $F_{vv}(I - G)$ sur le vecteur \bar{z}_v , le vecteur solution est $\bar{y} = (r_v)^T$.

À l'ordre 0, on obtient le bloc de Gauss-Seidel (bloc de Jacobi si $A_{uv} = 0$). Dans notre problème principal, la matrice de (I.13) est obtenue en tronquant par exemple le développement à l'ordre 0 ce qui conduit à :

$$[F_{vv}(I - G)]^{-1} \approx F_{vv}^{-1} \quad (4.19)$$

Dans le cadre de cette étude, à l'ordre 0, le bloc de Gauss-Seidel s'est révélé légèrement plus performant que le bloc de Jacobi et le preconditionneur triangulaire. Il a donc été préféré. En effet, nous avons réalisé une comparaison en 2D de différents preconditionneurs dédiés au bloc de vitesse, pour un problème diphasique de la montée d'une bulle à différentes conditions de CFL. En outre, aucun gain de temps CPU n'est obtenu lorsque l'ordre de l'approximation est augmenté. Finalement, la forme de notre preconditionneur de vitesse à l'ordre 0 sera la suivante :

$$\tilde{F}_u \approx \hat{F}_u = \begin{pmatrix} F_u & F_{uv} \\ 0 & F_{vv} \end{pmatrix}. \quad (4.20)$$

4.3.2 Les techniques de preconditionnement pour le complément de Schur

Dans cette section, nous décrivons deux méthodes d'approximation de l'opérateur de Schur $S_p = -B_u F_u^{-1} B_p^T$ qui peuvent être utilisées dans le preconditionneur (4.10). Pour l'écoulement de Stokes stationnaire, où F_u contient seulement le terme de diffusion A_μ , le complément de Schur est équivalent à la matrice de masse [60] :

$$S_p^{-1} \approx M_p = \int_{\Omega_{p(i,j)}} dV \quad (4.21)$$

Pour des écoulements de Stokes instationnaires, le terme de diffusion est contrebalancé par le terme temporel. Dans ce cas, le preconditionneur de Cahouet-Chabard [30] présente un bon candidat pour approximer l'inverse du Schur, via l'introduction d'une matrice de masse M_p et un laplacien $A_p = B_u M_u^{-1} B_p^T$ comme suit :

$$S_p^{-1} \approx \mu M_p^{-1} + \frac{1}{\Delta t} \left(B_u (M_u^{-1}) B_p^T \right)^{-1} \quad (4.22)$$

où M_u est la matrice de masse dans l'espace de vitesse.

En ce qui concerne les écoulements incompressibles de type Navier-Stokes, nous trouvons une variété de méthodes de factorisation approximant le complément de Schur S_p . Bien qu'elles n'aient pas été envisagées dans cette thèse, ces techniques incluent le préconditionneur Simple [148] où le Schur est approché comme suit :

$$S_p \approx B_u(\text{diag}(F_u))^{-1}B_p^T \quad (4.23)$$

Une autre approche consiste à augmenter le nouveau système linéaire afin que le nouveau système admette une approximation plus simple de son complément de Schur. Ces idées sont à l'origine du lagrangien augmenté [17, 18], pour lequel nous remplaçons le système d'origine (5.6) par :

$$\begin{pmatrix} F_u + rB_p^T D^{-1}B_u & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \bar{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \bar{f} \\ 0 \end{pmatrix} \quad (4.24)$$

où D est une matrice symétrique définie positive et $r > 0$ est un paramètre positif. Notons que, par construction, les solutions \bar{u}^{n+1} et p^{n+1} restent inchangées. De plus, si D est une matrice diagonale, alors $F_u^{(r)} = F_u + rB_p^T D^{-1}B_u$ sera creuse. Par la suite, le préconditionneur triangulaire supérieur par bloc est alors utilisé sur le système augmenté (4.24)

$$P_r = \begin{pmatrix} \hat{F}_u^{(r)} & B_p^T \\ 0 & \hat{S}_p^{(r)} \end{pmatrix} \quad (4.25)$$

où, comme précédemment, $\hat{F}_u^{(r)}$ approche $F_u^{(r)}$ et $\hat{S}_p^{(r)}$ approche $S_p^{(r)} = -B_u(F_u^{(r)})^{-1}B_p^T$. En pratique, sous un choix approprié de r l'approximation du complément de Schur, définie par son inverse se lit comme suit :

$$(S_p^{(r)})^{-1} \approx \mu M_u^{-1} + rD^{-1}. \quad (4.26)$$

D'autres techniques reposent sur l'utilisation de commutateurs approchés, avec l'utilisation du commutateur entre l'opérateur de divergence B_u et l'opérateur de convection-diffusion F_u . Parmi ces techniques, nous trouvons le préconditionneur PCD (Pressure Convection Diffusion) utilisé dans les travaux de Kay et al [104], où l'inverse du complément de Schur est approximé par :

$$S_p^{-1} \approx A_p^{-1}F_p M_p^{-1}. \quad (4.27)$$

où $A_p = B_u B_p^T$ qui représente un laplacien discret, et F_p qui est l'opérateur de convection-diffusion agissant dans l'espace de pression. On peut également utiliser d'autres types de ces techniques, de telle sorte que le commutateur discret soit petit au sens des moindres carrés. Ainsi, l'approximation inverse du complément de Schur intitulée préconditionneur LCS (Least-

Squares Commutator) [56, 57] prend alors la forme suivante :

$$S_p^{-1} \approx \left(B_u M_u^{-1} B_p^T \right)^{-1} \left(B_u M_u^{-1} F M_u^{-1} B_p^T \right) \left(B_u M_u^{-1} B_p^T \right)^{-1}. \quad (4.28)$$

Pour des écoulements diphasiques à forts rapports de masses volumiques et de viscosités, la matrice de masse appropriée est redimensionnée par la viscosité μ de la façon suivante :

$$M_p^{(1/\mu)} \approx \int_{\Omega_p(i,j)} \mu^{-1} dV. \quad (4.29)$$

Si on l'admet que le PCD est toujours sous la forme de (4.27), mais avec une mise à l'échelle supplémentaire de la matrice de masse $M_p^{(1/\mu)}$, alors, nous avons besoin de redimensionner le laplacien discret de sorte que :

$$A_p^{(\mu)} \approx \int_{\Omega_p(i,j)} \mu \Delta \phi dV. \quad (4.30)$$

Ces choix donnent l'approximation du complément de Schur suivante :

$$S_p^{-1} \approx \left(A_p^{(\mu)} \right)^{-1} F_p \left(M_p^{(1/\mu)} \right)^{-1}. \quad (4.31)$$

Cependant, d'après les travaux de Bootland et al [21], la performance de ce préconditionneur dépend fortement du rapport de masse volumique entre les deux fluides. En particulier, l'approximation du complément de Schur décrite via l'équation (4.31) se révèle moins efficace dans les cas diphasique instationnaire à forts rapports de masses volumiques. Pour étendre l'utilisation du PCD à la simulation des écoulements diphasiques à forts rapports de masses volumiques et de viscosités, nous devons subdiviser l'opérateur F_p et traiter les termes séparément et de manière appropriée. Pour un écoulement diphasique l'opérateur F_p est séparé en trois parties :

$$F_p = A_p^{(\mu)} + N_p^{(\rho)} + \Delta t^{-1} M_p^{(\rho)}. \quad (4.32)$$

Après un splitting approprié de l'équation (4.32) et une scaling adéquat des différents opérateurs, Bootland et al [21] arrivent à construire une approximation de l'inverse de Schur par la relation suivante :

$$\hat{S}_{PCD}^{-1} = (M_p^{1/\mu})^{-1} + (A_p^{1/\rho})^{-1} \left(N_p^{(1)} + \frac{1}{\Delta t} M_p^{(1)} \right) (M_p^{(1)})^{-1}. \quad (4.33)$$

Ici, l'exposant entre parenthèses est donné comme 1 pour bien montrer qu'il s'agit de termes non adimensionnés, c'est à dire dont la définition ne comprend ni ρ ni μ . Nous avons donc :

$$N_p^{(1)} \approx \int_{\Omega_p(i,j)} \nabla \cdot (\bar{u} \nabla \phi) dV. \quad (4.34)$$

En revanche le terme $A_p^{1/\rho}$ correspond à un laplacien à masse volumique variable, qui est donné par :

$$A_p^{(1/\rho)} \approx \int_{\Omega_p(i,j)} \nabla \cdot \left(\frac{1}{\rho} \nabla \phi \right) dV. \quad (4.35)$$

Afin d'approximer l'inverse du complément de Schur \hat{S}_{PCD}^{-1} , nous utilisons le solveur multigrille PFMG ou SMG de la bibliothèque HYPRE [69], pour lequel l'action de $(A_p^{1/\rho})^{-1}$ est requise. Ce solveur a été conçu spécifiquement pour les équations elliptiques à coefficients variables. Les matrices de masse diagonales à savoir $(M_p^{1/\mu})^{-1}$ et $(M_p^{(1)})^{-1}$ sont alors inversées, en appliquant un redimensionnement aux vecteurs appropriés suivi d'un produit matrice-vecteur pour l'opérateur $N_p^{(1)} + \frac{1}{\Delta t} M_p^{(1)}$. En outre, le préconditionneur PCD requiert certains choix de conditions aux limites (le lecteur se réfère à [61] pour plus d'informations), qui doivent être compatibles avec les conditions physiques aux limites appliquées à la vitesse \bar{u} . En effet, le choix de conditions aux limites de Dirichlet pour la vitesse \bar{u} implique l'application de conditions aux limites de Neumann pour les opérateurs discrets $A_p^{(1/\rho)}$ et $N_p^{(1)} + \frac{1}{\Delta t} M_p^{(1)}$. D'autre part, l'utilisation des conditions aux limites d'entrée ou de sortie respectivement pour la vitesse \bar{u} entraîne l'utilisation des conditions de Robin pour les opérateurs discrets de l'approximation de Schur. En ce qui concerne le bloc de vitesse, il nécessite la résolution de deux systèmes linéaires F_{vv} et F_{uu} , qui sont tous deux effectués à l'aide du solveur multigrille de la bibliothèque HYPRE [69]. Ces résolutions sont complétées par deux produits matrice-vecteur et deux mises à jour du second membre. Finalement, l'algorithme appliqué au préconditionneur triangulaire par bloc est résumé dans le tableau 4.3.

4.4 Validations sur des configurations monophasiques et diphasiques

Dans les sections précédentes, nous avons développé un solveur tout-couplé implicite pour traiter le couplage vitesse-pression issu des équations de Navier-Stokes en 2D. Nous avons également mis en place quelques préconditionneurs pour le système à point selle. Dans cette partie, nous allons nous attacher à la validation du solveur tout-couplé, à travers des cas monophasiques et diphasiques afin de prouver sa robustesse. L'apport majeur du nouveau solveur tout-couplé sur la simulation des écoulements diphasiques avec forts rapports de propriétés physiques à l'interface entre deux fluides sera également étudié.

4.4.1 Cavité entraînée

Le premier cas test porte sur l'étude numérique de l'écoulement incompressible et bidimensionnel au sein d'une cavité carrée de côté $L = 1m$, entièrement remplie d'un fluide visqueux. Dans cette configuration, les conditions aux limites à appliquer sont de type Dirichlet homogène, à l'exception de la paroi supérieure qui se déplace à une vitesse $u_0 = -1m.s^{-1}$. Pour valider le solveur tout-couplé, nous considérons les résultats numériques obtenus pour trois nombres de Reynolds ($Re = 100, 1000$ and 5000). Nous les comparons à ceux issus des travaux de Botella et Peyret [23]. Pour nous assurer de la qualité de nos résultats et leurs convergences numériques vers ceux de références, nous estimons l'ordre de convergence via une extrapolation de Richardson (pour plus de détails, le lecteur est invité à se reporter à [128]). Ainsi, nous utilisons différents

Le préconditionneur triangulaire par bloc	
1	Solve $\begin{pmatrix} F_{uu} & F_{uv} & B_{pu}^T \\ 0 & F_{vv} & B_{pv}^T \\ 0 & 0 & \hat{S}_{PCD} \end{pmatrix} \begin{pmatrix} \bar{r}_u \\ \bar{r}_v \\ \bar{r}_p \end{pmatrix} = \begin{pmatrix} \bar{z}_u \\ \bar{z}_v \\ \bar{z}_p \end{pmatrix}$
2	!1 Approximate the inverse of Schur \hat{S}_{PCD}
3	–Two mass matrix solve $(M_p^{(1/\mu)})^{-1}$ and $(M_p^{(1)})^{-1}$
4	$\bar{r}_{p1} = (M_p^{(1)})^{-1} \bar{z}_p$ $\bar{r}_{p2} = (M_p^{(1/\mu)})^{-1} \bar{z}_p$
5	–Matrix–vector product for the operator $F_P^{(1)}$
6	$\bar{r}_{p3} = F_P^{(1)} \bar{r}_{p2}$
7	–Solve Laplace operator $A_P^{1/\rho}$
8	$\bar{r}_{p4} = (A_P^{1/\rho})^{-1} \bar{r}_{p3}$!with multigrid solver
9	$\bar{r}_p = \bar{r}_{p4} + \bar{r}_{p1}$
10	!2 Update \bar{z}_u and \bar{z}_v
11	$\bar{z}_u = \bar{z}_u - B_{pu}^T \bar{r}_p$
12	$\bar{z}_v = \bar{z}_v - B_{pv}^T \bar{r}_p$
13	!3 Approximate the velocity block \hat{F}_u
14	–Matrix solve F_{vv}
15	$\bar{r}_v = F_{vv}^{-1} \bar{z}_v$!with multigrid solver
16	–Matrix–vector product for the operator F_{uv}
17	$\hat{r}_u = F_{uv} \bar{r}_v$
18	–Update \bar{z}_u
19	$\bar{z}_u = \bar{z}_u - \hat{r}_u$
20	–Matrix solve F_{uu}
21	$\bar{r}_u = F_{uu}^{-1} \bar{z}_u$!with multigrid solver

TABLE 4.3 – Description de l’application du préconditionneur triangulaire par blocs. $M_p^{1/\mu}$ est la matrice de masse normalisée par l’inverse de la viscosité $1/\mu$, $M_p^{(1)}$ est la matrice de masse standard, $N_p^{(1)}$ représente la partie discrète du terme convectif standard dans l’espace de pression, $A_p^{1/\rho}$ est le laplacien normalisé, qui correspond à la discrétisation du terme $\nabla \cdot \left(\frac{1}{\rho} \nabla P\right)$, et F_u est l’opérateur de convection-diffusion pour l’équation de la quantité de mouvement. Ici, $\bar{r} = (r_u, r_v, r_p)^T$ est égale à \bar{b} , \bar{q} , \bar{v} , \bar{s} , \bar{w} et \bar{t} dans le BiCGSTAB(2) (voir Algo.4.2).

maillages cartésiens uniformes, tout en maintenant le pas de temps constant et égal à $\Delta t = 10^{-3}s$ et un résidu de $\epsilon = 10^{-6}$ pour le solveur BiCGSTAB(2).

Comme nous nous intéressons seulement aux écoulements permanents ici, les itérations temporelles sont achevées si le critère stationnaire est satisfait (10^{-6}).

$$\max \left(\|u^{n+1} - u^n\|, \|v^{n+1} - v^n\| \right) \leq 10^{-6} \max \left(\|u^{n+1}\|, \|v^{n+1}\| \right) \quad (4.36)$$

ici, le nombre de Reynolds est défini comme suit :

$$Re = \frac{\rho u_0 L}{\mu} \quad (4.37)$$

où ρ est la masse volumique, μ est la viscosité dynamique, L est la hauteur de la cavité, et u_0 est la vitesse caractéristique de l’écoulement.

Les figures 4.7a et 4.7b montrent l'évolution de la vitesse horizontale u et verticale v le long de la ligne médiane de la cavité, obtenues par le solveur tout-couplé pour $Re = 100$ avec différents maillages cartésiens. Il apparaît clairement de ces figures que les résultats sont qualitativement similaires dès le maillage 32×32 pour les deux composantes de vitesse u et v . En outre, le tableau 4.4 fournit les valeurs numériques correspondant au maximum de la vitesse u sur la ligne médiane verticale (notée u_{max}), au maximum de la vitesse v sur la ligne médiane horizontale (v_{max}), à la vorticité ω_{center} , aux vitesses u_{center} et v_{center} au centre de la cavité. Ces valeurs sont comparées à la solution de référence. Comme nous pouvons le constater dans le tableau 4.4, tous les résultats de calcul sont en bon accord avec la référence et les écarts ne dépassent pas 0.006%. Comme nous pouvions nous y attendre, les taux de convergence estimés pour les grandeurs u_{center} , v_{center} , ω_{center} , u_{max} et v_{max} sont 2.00, 1.94, 2.05, 2.05 et 2.04, correspondant à la précision du schéma centré utilisé.

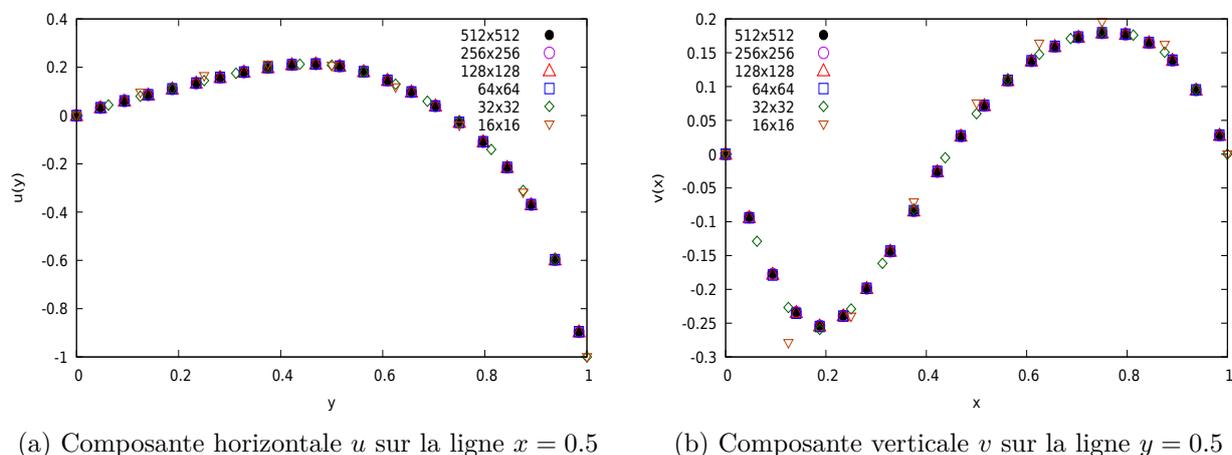


FIGURE 4.7 – Écoulement de la cavité entraînée pour un nombre de Reynolds de 100. Comparaison des profils de la première (gauche) et de la deuxième (droite) composante de la vitesse extraits sur des droites verticale et horizontale situées au milieu de la cavité avec les résultats de Botella et Peyret [23].

N	u_{center}	order	v_{center}	order	ω_{center}	order	u_{max}	order	v_{max}	order
64	0.20876	1.39	0.05804	3.03	1.17746	5.01	0.21375	-	0.17991	3.72
128	0.20904	1.41	0.05767	2.34	1.17461	5.97	0.21395	1.65	0.17956	2.09
256	0.20912	1.85	0.05757	1.89	1.17446	1.29	0.21402	1.56	0.17959	2.10
512	0.20914	1.98	0.05754	1.82	1.17440	1.47	0.21403	2.53	0.17957	1.91
1024	0.20914	2.00	0.05753	1.94	1.17439	2.05	0.21403	2.05	0.17957	2.04
Extra	-	-	-	-	1.17439	-	0.21403	-	0.179560	-
Ref [23]	-	-	-	-	1.17441	-	0.21404	-	0.179572	-

TABLE 4.4 – Écoulement de la cavité entraînée pour un nombre de Reynolds de 100. Comparaison de quelques caractéristiques de l'écoulement le long de la ligne médiane avec les valeurs de référence de Botella et Peyret, [23]. Ici, Extra est la valeur interpolée via une extrapolation de Richardson.

Le deuxième cas test est étudié pour $Re = 1000$, où la dynamique de l'écoulement est prin-

cipalement dominée par les forces d’inertie. L’objectif de ce cas est de vérifier la capacité du solveur tout-couplé à capturer les différentes structures présentées dans l’écoulement, telles que les tourbillons primaire, secondaire droit et secondaire gauche, et à estimer la vorticit  au niveau de ces endroits. De nouveau, les r sultats num riques sont confront s   ceux de r f rence [23], en conservant les m mes param tres num riques que dans l’ tude pr c dente.

  $Re = 1000$, la fonction courant de l’ coulement est illustr e   la figure 4.8. L’existence de trois tourbillons diff rents est  vidente. Le tourbillon primaire not  PV est de grande taille et centr  dans la cavit , les autres tourbillons secondaires repr sent s par LSV et RSV sont de petite taille et situ s respectivement pr s des coins inf rieurs gauche et droit de la cavit . La figure 4.9 pr sente la vitesse u extraite de la ligne centrale verticale et la vitesse v le long de la ligne centrale horizontale. Les r sultats num riques pr sentent une bonne concordance avec ceux de r f rence, ce qui confirme la pertinence du solveur tout-coupl . En plus de cela, les tableaux 4.5, 4.6, et 4.7 synth tisent les valeurs de la vorticit  au centre de chaque tourbillon et leurs positions. Ces r sultats pr sentent tous une pr cision de second ordre dans la zone de convergence asymptotique. En ce qui concerne les positions des tourbillons, la diff rence est toujours inf rieure   0.11%, et pour la vorticit , l’ cart le plus important est de l’ordre de 0.06%.

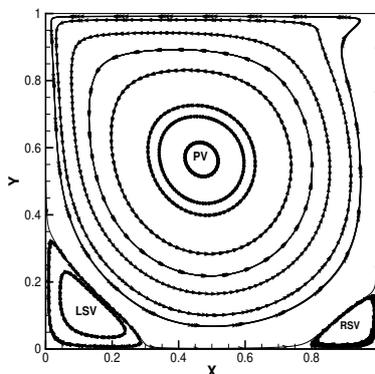


FIGURE 4.8 –  coulement de la cavit  entra n e pour un nombre de Reynolds de 1000. Les contours de la fonction courant sont donn s, montrant l’apparition de tourbillons primaire et secondaire.

Pour valider la capacit  du solveur tout-coupl    simuler des probl mes stationnaires plus complexes, le cas de $Re = 5000$ est  galement  tudi . La figure 4.10 illustre les profils des composantes horizontale (u) et verticale (v) et de la vorticit  (ω) le long des lignes centrales $x = 0.5$ et $y = 0.5$. Les extrema de vitesse sur les lignes centrales de la cavit , les intensit s du tourbillon et leurs emplacements sont respectivement indiqu s dans les tableaux 4.8 et 4.9. Il ressort de ces tableaux que l’accord entre les solutions num riques et les valeurs de r f rence est tr s bon.

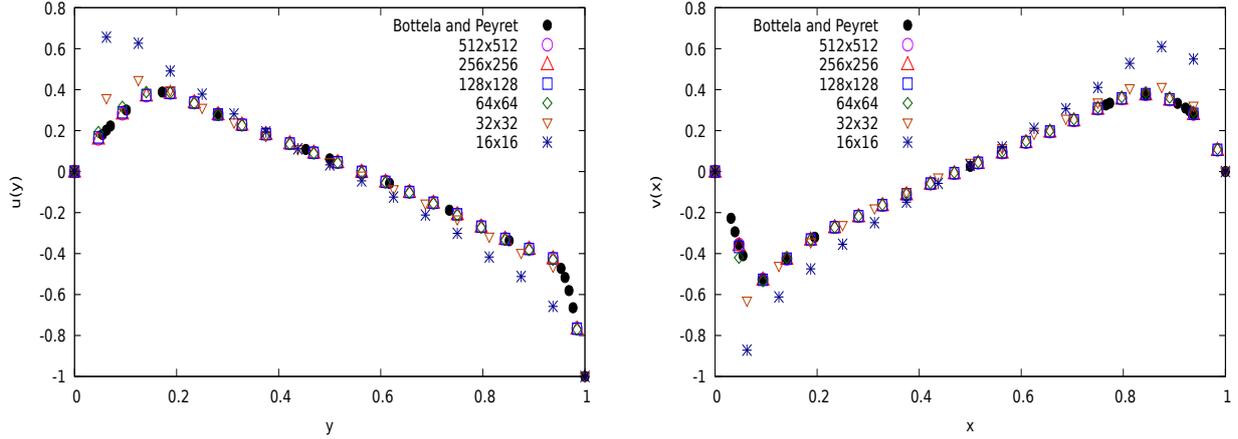
(a) Composante horizontale u sur la ligne $x = 0.5$ (b) Composante verticale v sur la ligne $y = 0.5$

FIGURE 4.9 – Écoulement de la cavité entraînée pour un nombre de Reynolds de 1000. Comparaison des profils de la première (gauche) et de la deuxième (droite) composante de la vitesse extraits sur des droites verticale et horizontale situées au milieu de la cavité avec les résultats de Botella et Peyret [23].

N	x	order	y	order	ω	order
32	0.460674		0.545429		2.16344	
64	0.467474		0.560757		2.07752	
128	0.468837	1.31	0.564257	2.13	2.06684	3.01
256	0.469113	2.30	0.565009	2.21	2.06693	-
512	0.469185	1.93	0.565183	2.11	2.06737	2.29
Extra	0.469179	-	0.56518	-	2.067369	-
Ref [23]	0.4692	-	0.5652	-	2.067753	-

TABLE 4.5 – Écoulement de la cavité entraînée pour un nombre de Reynolds de 1000. L'extrapolation de Richardson pour le tourbillon primaire : (x, y) la position du centre du tourbillon, et ω est la valeur de la vorticité à cet emplacement.

N	x	order	y	order	ω	order
32	0.08940		0.08617		-1.50012	
64	0.12906		0.10731		-1.23281	
128	0.13438	2.89	0.11081	2.59	-1.13287	1.41
256	0.13556	2.16	0.11157	2.20	-1.11431	2.42
512	0.13586	1.96	0.11174	2.12	-1.11053	2.30
Extra	0.13584	-	0.11173	-	-1.1105301	-
Ref [23]	0.1360	-	0.1118	-	-1.109789	-

TABLE 4.6 – Écoulement de la cavité entraînée pour un nombre de Reynolds de 1000. L'extrapolation de Richardson pour le tourbillon secondaire droit : (x, y) la position du centre du tourbillon, et ω est la valeur de la vorticité à cet emplacement.

4.4.2 Écoulement de Poiseuille stationnaire

Dans ce cas test, nous souhaitons simuler numériquement au moyen du solveur tout-couplé, l'écoulement de Poiseuille généré par un gradient de pression. Nous faisons les simulations pour deux types d'écoulements, un écoulement monophasique et un écoulement diphasique pour lequel

N	x	order	y	order	ω	order
32	0.9073		0.0985712		-0.736348	
64	0.9152		0.0808984		-0.401598	
128	0.916427	2.68	0.0786425	2.96	-0.361584	3.06
256	0.916657	2.41	0.07821	2.38	-0.354138	2.42
512	0.916709	2.14	0.0781233	2.31	-0.352588	2.26
Extra	0.916708	-	0.078116	-	-0.3525867	-
Ref [23]	0.9167	-	0.0781	-	-0.3522861	-

TABLE 4.7 – Écoulement de la cavité entraînée pour un nombre de Reynolds de 1000. L'extrapolation de Richardson pour le tourbillon secondaire gauche : (x,y) la position du centre du tourbillon, et ω est la valeur de la vorticité à cet emplacement.

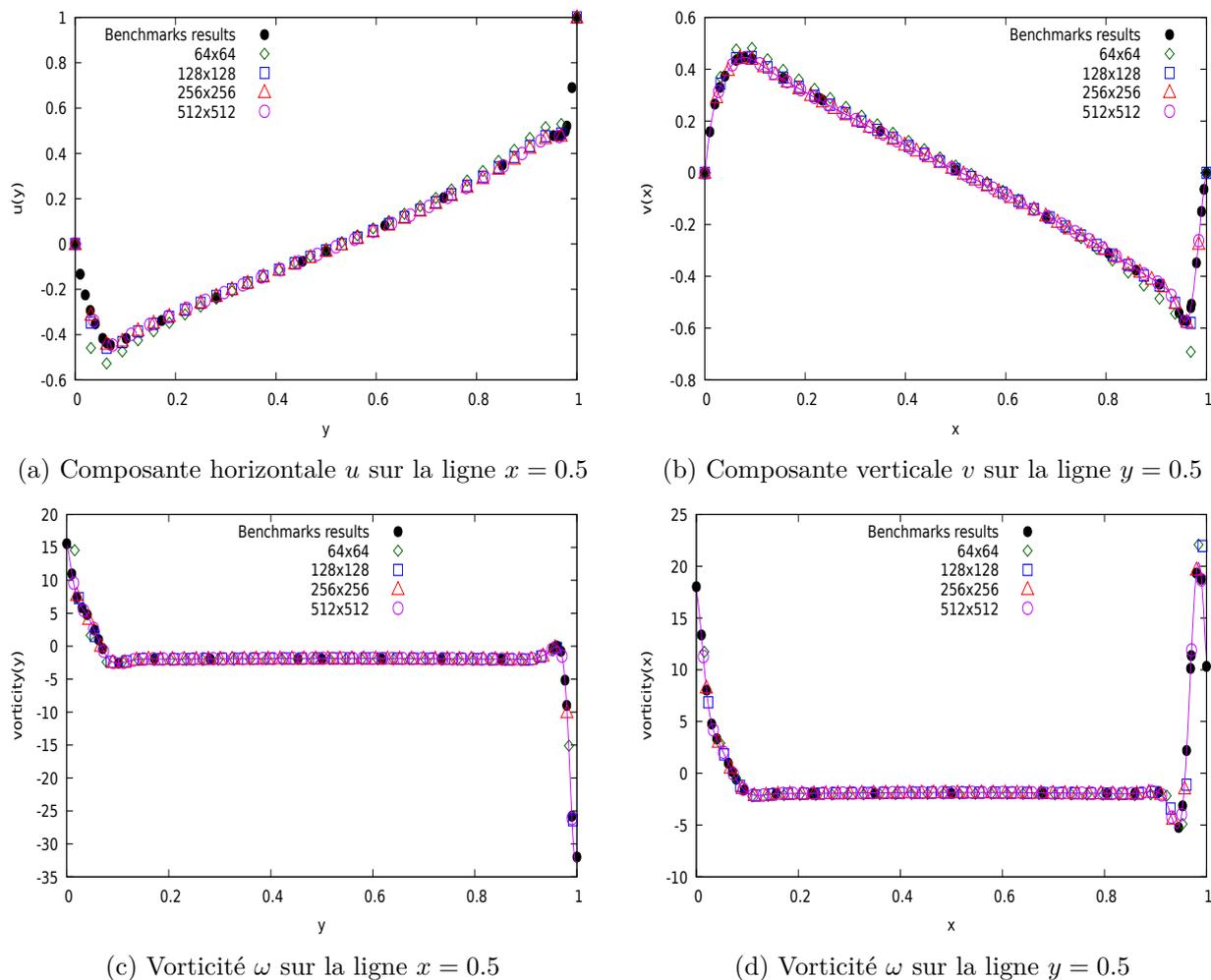


FIGURE 4.10 – Écoulement de la cavité entraînée pour un nombre de Reynolds de 5000. Comparaison des composantes horizontale et verticale et de vorticité avec les valeurs de référence (<https://www.acenumerics.com/the-benchmarks.html>)

le rapport de viscosité est de 10. Pour cette étude, l'objectif principal est de comparer les résultats obtenus par la simulation numérique avec les résultats exactes. Ces comparaisons permettront de valider notre solveur tout-couplé et de vérifier l'implémentation du terme visqueux.

L'écoulement de fluide se fait dans une cavité carrée de $1m$ de côté. Pour la dynamique, nous

	y_{min}	u_{min}	x_{max}	v_{max}	x_{min}	v_{min}
Present work	0.0742196	-0.44616	0.079103	0.445349	0.958007	-0.574624
Ref	0.0742967	-0.4473136	0.0796881	0.4475084	0.9574605	-0.5761359

TABLE 4.8 – Écoulement de la cavité entraînée pour un nombre de Reynolds de 5000. Extrema de la vitesse à travers les lignes centrales de la cavité, obtenus en utilisant une grille cartésienne 512×512 - Ref sont les valeurs de références données sur le site : <https://www.acenumerics.com/the-benchmarks.html>.

	x	x_{ref}	y	y_{ref}	ω	ω_{ref}
PV	0.515265	0.5150938	0.535095	0.5352620	-1.94649	-1.940636
TL1	0.0630822	0.06335432	0.909118	0.9092568	2.08812	2.091562
BR1	0.805512	0.8046258	0.072499	0.07274724	2.76827	2.752178
BR2	0.978491	0.9783737	0.018725	0.01877709	-0.0350574	-0.0349262
BL1	0.0727551	0.07285072	0.137474	0.1370630	1.52971	1.518861

TABLE 4.9 – Écoulement de la cavité entraînée pour un nombre de Reynolds de 5000, sur une grille cartésienne 512×512 . Carte des tourbillons de la cavité. Ref dénote les valeurs de références données sur le site : <https://www.acenumerics.com/the-benchmarks.html>. PV, BR, BL, and TL représentent les tourbillons primaire, inférieur droit, inférieur gauche et supérieur gauche, respectivement.

supposons qu'il y a adhérence des fluides aux parois supérieure et inférieure, c'est à dire que la vitesse du fluide sur les parois est nulle. En revanche, des conditions de périodicité sont imposées sur la vitesse à l'entrée et à la sortie de la cavité. En considérant un temps suffisamment long et en négligeant les effets de gravité, nous pouvons comparer la solution numérique stationnaire à la solution analytique définie comme suit :

$$u(y) = \begin{cases} \frac{\Delta P}{2\mu_1} \left(y^2 + y \frac{\mu_1(d^2 - H^2) - \mu_2 d^2}{\mu_2 d - \mu_1(d - H)} \right), & \text{si } y \leq d \\ \frac{\Delta P}{2\mu_1} \left(y^2 + y \frac{(\mu_1(d^2 - H^2) - \mu_2 d^2) + H(d^2 - Hd)(\mu_2 - \mu_1)}{\mu_2 d - \mu_1(d - H)} \right), & \text{si } y > d \end{cases} \quad (4.38)$$

avec H la distance entre les deux plans horizontaux, d la position de l'interface entre les deux fluides et $\Delta P = p_{out} - p_{in}$ la différence entre les pressions à la sortie et à l'entrée de la cavité. Par définition, $0 < d < H$.

Pour quantifier les niveaux d'erreurs, nous définissons l'erreur $\|E_u\|$ commise sur la vitesse horizontale u entre la solution numérique et la solution analytique en norme $\|\cdot\|_2$ et $\|\cdot\|_\infty$.

$$\|E_u\|_2 = \sqrt{\frac{\sum_{i=1 \dots N_y} \|u_{ref}(i) - u_{calc}(i)\|_2}{\sum_{i=1 \dots N_y} \|u_{ref}(i)\|_2}} \quad (4.39a)$$

$$\|E_u\|_\infty = \frac{\max |u_{ref}(i) - u_{calc}(i)|}{\max |u_{ref}(i)|} \quad (4.39b)$$

où N_y représente le nombre de mailles de la grille dans la direction verticale, $u_{ref}(i)$ la vitesse exacte calculée au nœud i du maillage et $u_{calc}(i)$ la solution numérique en i .

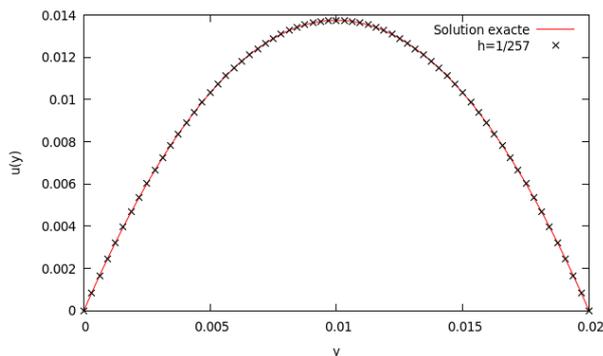


FIGURE 4.11 – Simulation numérique sur un maillage 8×257 de l'écoulement de Poiseuille 2D par le solveur tout-couplé, avec un résidu de $\epsilon = 10^{-12}$ pour le solveur BiCGSTAB(2). Comparaison entre la solution analytique et celle issue du calcul.

4.4.2.1 Résultats obtenus en écoulement monophasique

Dans cette partie, nous présentons les résultats de la simulation que nous avons obtenus en écoulement monophasique. Le tracé 4.11 de la composante u du champ de vitesse montre que les simulations numériques tendent vers les solutions exactes. Quel que soit le maillage, l'erreur entre solution numérique et analytique est de l'ordre de la précision machine, conformément à ce qui est attendu avec un schéma centré sur un maillage régulier.

4.4.2.2 Résultats obtenus en écoulement diphasique

Comme précédemment énoncé, l'étude de l'écoulement diphasique se fait avec un rapport de viscosité de 10. Les comparaisons des solutions numériques aux solutions exactes permettent non seulement de valider notre solveur tout-couplé, mais aussi de connaître le choix optimal pour évaluer les viscosités (moyennes harmoniques ou arithmétiques).

Dans cette partie, nous faisons une étude de convergence en maillage pour un écoulement de Poiseuille diphasique avec un rapport de viscosité de 10. Pour ce faire, nous étudions l'évolution de la vitesse horizontale u sur l'axe $x = H/2$ pour différents maillages. Nous obtenons les résultats présentés dans la figure 4.12. Les solutions sont confondues quelles que soient les moyennes utilisées pour approximer la viscosité.

Le tableau 4.10 donne les erreurs 4.39a et 4.39b pour le solveur tout-couplé et la méthode VOF avec une viscosité obtenue soit avec une moyenne arithmétique, soit avec une moyenne harmonique. Nous constatons que la méthode VOF/PLIC avec une viscosité obtenue via une moyenne harmonique est toujours plus précise et converge à l'ordre 2 avec des niveaux d'erreur $\|L\|_2$ et $\|L\|_\infty$ identiques. Quant à elle, la méthode VOF/PLIC avec une moyenne arithmétique est moins précise et converge à l'ordre 1.

VOF-PLIC μ arithmétique				
h	$\ E_u\ _2$	Ordre	$\ E_u\ _\infty$	Ordre
1/9	5,44E-02		9,14E-02	
1/17	3,01E-02	0,85	5,16E-02	0,82
1/33	1,58E-02	0,92	2,73E-02	0,92
1/65	8,14E-03	0,95	1,40E-02	0,96
1/129	4,12E-03	0,98	7,12E-03	0,98
1/257	2,08E-03	0,99	3,58E-03	1,0
VOF-PLIC μ harmonique				
h	$\ E_u\ _2$	Ordre	$\ E_u\ _\infty$	Ordre
1/9	8,48E-03		1,17E-02	
1/17	2,40E-03	1,82	3,31E-03	1,82
1/33	6,45E-04	1,90	8,75E-04	1,91
1/65	1,67E-04	1,94	2,25E-04	1,96
1/129	4,27E-05	1,97	5,76E-05	1,97
1/257	1,09E-05	1,97	1,49E-05	1,95

TABLE 4.10 – Simulation numérique sur des maillage $8 \times (9, 17, 33, 65, 129, 257)$ de l'écoulement de Poiseuille 2D par le solveur tout-couplé, avec un résidu de $\epsilon = 10^{-12}$ pour le solveur BiCG-STAB(2). Normes d'erreur entre profil de vitesse théorique et simulé selon le maillage. Approche VOF-PLIC avec une moyenne calculée de façon arithmétique et une méthode VOF-PLIC avec une moyenne harmonique.

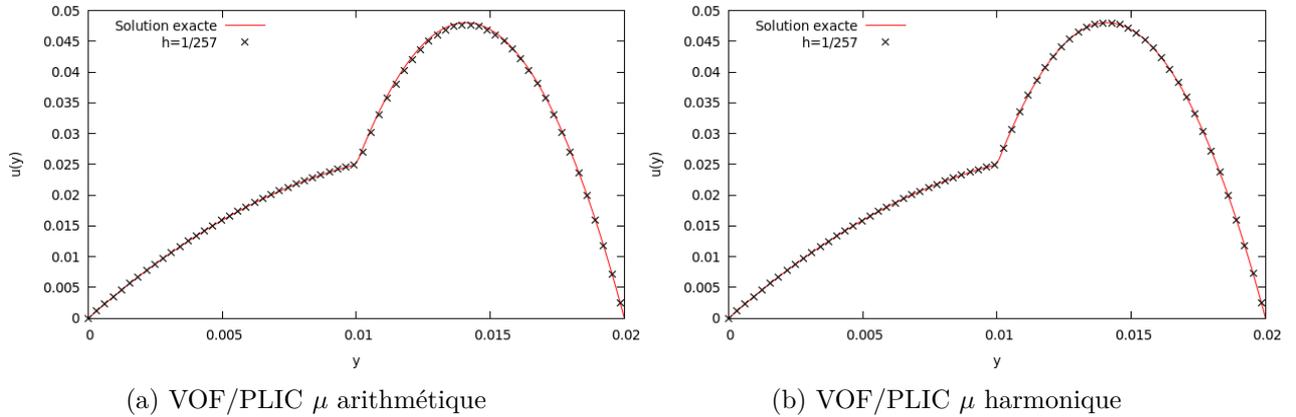


FIGURE 4.12 – Simulation numérique sur un maillage 8×257 de l'écoulement de Poiseuille diphasique 2D par le solveur tout-couplé, avec un résidu de $\epsilon = 10^{-12}$ pour le solveur BiCGSTAB(2). Comparaison entre la solution analytique et simulation.

4.4.2.3 Tourbillons de Taylor-Green

Les tourbillons de Taylor-Green sont une solution analytique périodique des équations de Navier-Stokes incompressible en deux dimensions. Ils décrivent un ensemble de tourbillons qui s'amortissent avec le temps. Ils sont utilisés comme benchmark pour valider des solutions numériques. La configuration géométrique est simple, il s'agit de prendre une boîte périodique $\Omega = (0, 1) \times (0, 1)$. Dans ce cas, les solutions analytiques s'écrivent avec $\nu = \frac{\mu}{\rho}$:

$$u(x, y, t) = V_0 \cos(2\pi x) \sin(2\pi y) e^{-8\pi^2 \nu t}, \quad (4.40)$$

$$v(x, y, t) = -V_0 \sin(2\pi x) \cos(2\pi y) e^{-8\pi^2\nu t}. \quad (4.41)$$

En nous servant des équations (4.40)-(4.41), deux quantités physiques intéressantes peuvent être évaluées : l'énergie cinétique E_k et l'enstrophie Ens :

$$E_k = \int_0^1 \int_0^1 \rho \frac{(u^2 + v^2)}{2} dx dy = \frac{\rho V_0^2 F^2(t)}{4} \quad (4.42)$$

$$Ens = \int_0^1 \int_0^1 \rho \frac{\omega^2}{2} dx dy = 2\pi^2 \rho V_0^2 F^2(t) \quad (4.43)$$

où $\omega = \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right)$ dénote la vortacité en deux dimensions et $F(t) = e^{-8\pi^2\nu t}$.

L'énergie cinétique E_k et l'enstrophie ζ sont deux grandeurs scalaires. Elles seront donc calculées au nœuds de pression comme suit :

$$E_k \simeq \sum_{\Omega_p} \frac{\rho}{2} (u_p^2 + v_p^2) \Omega_p, \quad (4.44)$$

$$\zeta \simeq \sum_{\Omega_p} \frac{\rho}{2} \omega_p^2 \Omega_p. \quad (4.45)$$

Les carrés des composantes de vitesses u_p^2 et v_p^2 sont obtenus par les interpolations suivantes :

$$u_p^2 = \frac{1}{2} (u_{i,j}^2 + u_{i+1,j}^2), \quad (4.46)$$

$$v_p^2 = \frac{1}{2} (v_{i,j}^2 + v_{i,j+1}^2). \quad (4.47)$$

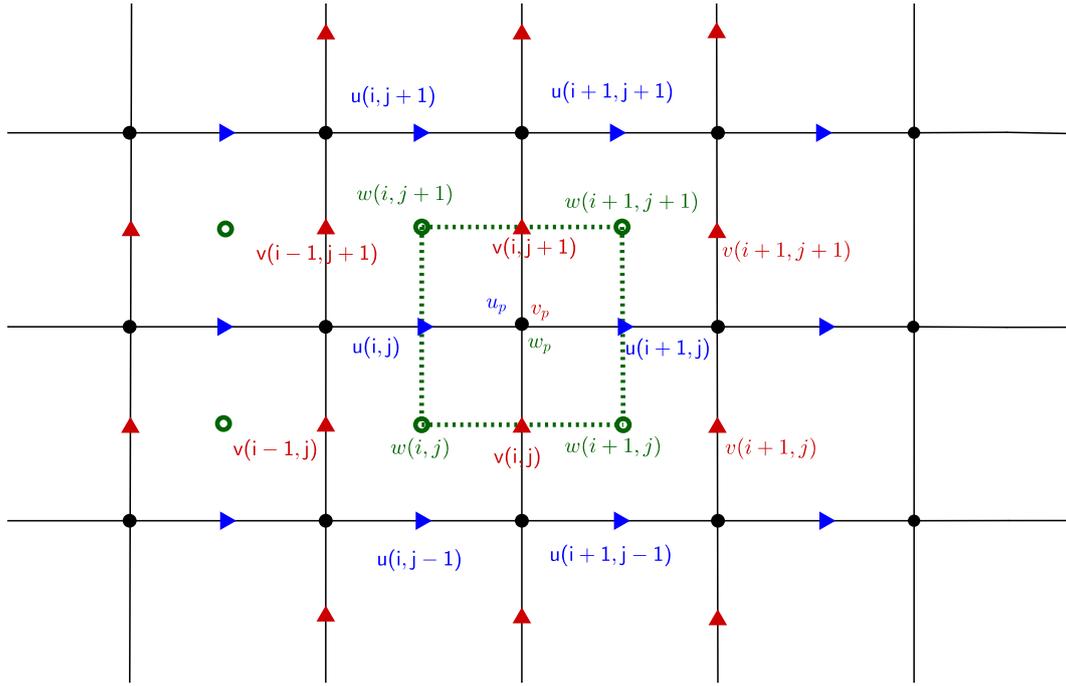
La vortacité est définie comme étant le rotationnel de la vitesse ($\bar{\omega} = \nabla \times \bar{u}$). En 2D, la vortacité se réduit à un scalaire qui sera exprimé aux nœuds de viscosité (voir Fig. 4.13) :

$$\omega(i, j) = \frac{v(i, j) - v(i-1, j)}{\Delta x} - \frac{u(i, j) - u(i, j-1)}{\Delta y}. \quad (4.48)$$

Le carré de la vortacité ω_p^2 est obtenu en faisant la moyenne des carrés des quatres vorticités qui viennent d'être exprimées. Nous aurons donc :

$$\omega_p^2 = \frac{\omega^2(i, j) + \omega^2(i+1, j) + \omega^2(i, j+1) + \omega^2(i+1, j+1)}{4} \quad (4.49)$$

Afin de mesurer la performance et la robustesse des solveurs implementés dans le code FUGU, les erreurs relatives à l'énergie cinétique E_k et à l'enstrophie Ens sont calculées pour un écoulement d'un fluide visqueux et parfait. Les résultats numériques sont comparés aux valeurs analytiques obtenues à partir des équations (4.42) et (4.62). Considérons donc q_t^{exacte} la quantité exacte, q_t

FIGURE 4.13 – Discrétisation de l'énergie cinétique E_k et de l'enstrophy Ens

la quantité numérique et n_t le nombre de pas de temps. L'erreur relative en norme L^2 sur la quantité q_t est estimée comme suit :

$$Err_2^{rel}(q_t) = \left(\frac{\sum_{k=1}^{n_t} \|q_t^{exacte} - q_t\|^2}{\sum_{k=1}^{n_t} \|q_t^{exacte}\|^2} \right)^{1/2} \quad (4.50)$$

Dans la suite, nous utilisons trois solveurs : le solveur tout-couplé, le lagrangien augmenté standard (avec $r = 10^6$ dans le cas d'un fluide parfait et $r = 10^2$ dans le cas d'un fluide visqueux) et la méthode de projection-correction. Pour la discrétisation spatiale, cinq maillages ont été simulés, allant de 16×16 à 256×256 , avec un résidu de $\epsilon = 10^{-12}$ pour le solveur BiCGSTAB(2). En ce qui concerne la discrétisation temporelle, nous avons imposé un pas de temps $\Delta t = 10^{-3}s$, pour lequel un temps physique de 0.3 seconde a été réalisé. Par ailleurs, la masse volumique est prise égale à $1kg \cdot m^{-3}$, et la viscosité a été fixée à $10^{-3}Pa \cdot s$ pour le cas visqueux.

Afin de mieux comparer les trois solveurs et comprendre l'effet du paramètre r du lagrangien augmenté, nous avons effectué une analyse quantitative à travers différentes valeurs du paramètre r allant de 10^3 à 10^7 (voir Fig. 4.14). On peut observer sur cette figure la sensibilité de l'erreur au paramètre du lagrangien augmenté. En revanche, on peut constater qu'une valeur de $r = 10^6$ semble être un bon choix afin de conserver l'énergie cinétique E_k à la précision machine. Notons qu'au-delà de cette valeur, le coût du calcul avec cette méthode devient trop élevé.

Pour un écoulement avec un fluide parfait ($\mu = 0Pa \cdot s$), les figures 4.15a-4.15b donnent les erreurs L^2 pour l'énergie cinétique E_k et l'enstrophie Ens entre la solution exacte et les solutions numériques, obtenues avec différents maillages. Pour l'enstrophie Ens , comme les approximations

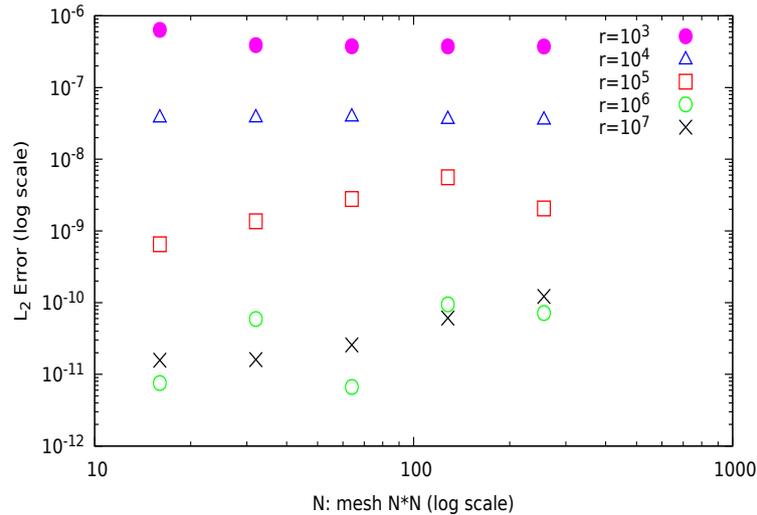


FIGURE 4.14 – Simulation numérique de l’écoulement de Taylor-Green 2D par le solveur lagrangien augmenté. Variation de l’erreur relative à la solution exacte de l’énergie cinétique E_k en fonction du maillage, et pour différentes valeurs du paramètre du lagrangien augmenté r .

des dérivées sont du second ordre, tous les solveurs ont une précision du second ordre, sauf la projection standard avec $\Delta t = 10^{-3}s$ qui est dans l’incapacité d’atteindre la précision attendue. En ce qui concerne l’énergie cinétique E_k , les résultats montrent que le solveur tout-couplé et le lagrangien augmenté sont plus efficaces que la projection standard, car ils conservent l’énergie cinétique E_k à la précision machine. Par conséquent, on peut clairement observer l’impact du pas de temps Δt sur la méthode de projection standard, causé par le splitting des opérateurs qui introduit une dissipation numérique. En effet, lorsque le pas de temps est divisé par 10, l’erreur sur l’énergie cinétique est divisée par 100.

Le même cas a été simulé pour un temps physique de 20 secondes, en gardant les mêmes paramètres numériques que dans le premier cas. Dans ce cas, tous les solveurs convergent avec la précision machine, à l’exception de la projection standard qui diverge après 6 secondes à cause de l’accumulation d’erreurs, y compris pour de petits pas de temps de l’ordre de $\Delta t = 10^{-6}s$. En outre, il a été démontré que les schémas skew-symétriques préservent l’énergie cinétique à la précision machine [74]. Nous notons également que les méthodes spectrales sont également capables de préserver cette quantité à la précision machine [165]. Dans ce travail, le schéma centré, combiné avec les techniques du lagrangien augmenté ou solveur tout-couplé, conserve donc également l’énergie cinétique de façon discrète.

La figure 4.16 montre les résultats obtenus pour le cas visqueux avec $\mu = 10^{-3}Pa \cdot s$. Pour le solveur tout-couplé et le lagrangien augmenté, on peut facilement remarquer que l’ensrophie Ens et l’énergie cinétique E_k convergent avec une précision du second ordre. L’impact du pas de temps est remarquable pour la méthode de projection standard. En effet, les résultats avec $\Delta t = 10^{-4}s$ sont significativement meilleurs que ceux obtenus avec $\Delta t = 10^{-3}s$. On peut en

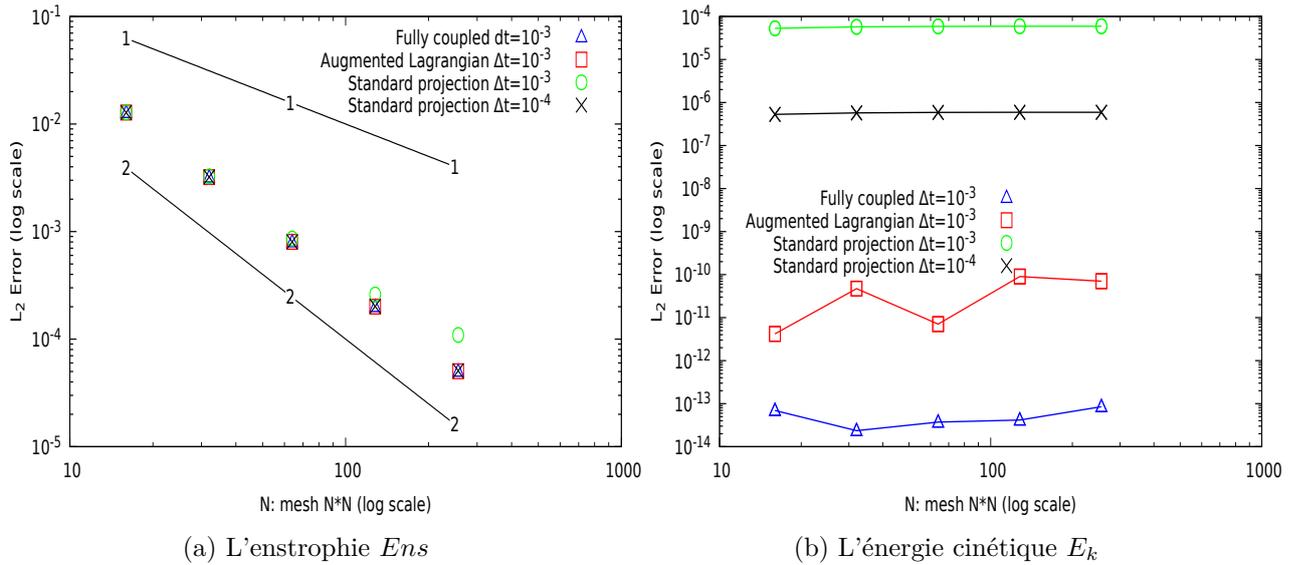


FIGURE 4.15 – Simulation numérique de l'écoulement de Taylor-Green 2D par le solveur tout-couplé. Évolution des normes d'erreurs en fonction du maillage utilisé pour l'énergie cinétique E_k et l'ensstrophie Ens , dans le cas d'un fluide parfait ($\mu = 0$).

conclure que la méthode de projection standard n'est pas en mesure de procurer de bons résultats avec de grands pas de temps, puisque le splitting introduit une erreur dans les champs de vitesses, et conduit donc à des erreurs numériques importantes. Par ailleurs, les effets de la dissipation numérique disparaissent pour la projection standard, lorsque de petits pas de temps sont utilisés, ceci rend la méthode plus coûteuse en temps de calcul et peut limiter son utilisation.

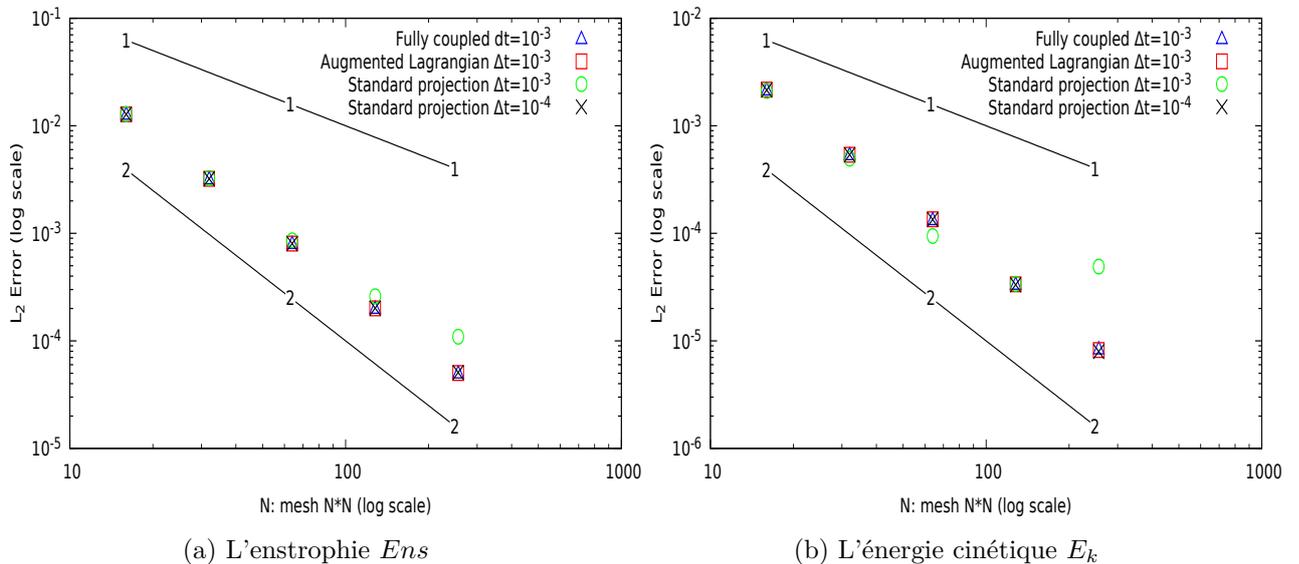


FIGURE 4.16 – Simulation numérique de l'écoulement de Taylor-Green 2D par le solveur tout-couplé. Évolution des normes d'erreurs en fonction du maillage utilisé pour l'énergie cinétique E_k et l'ensstrophie Ens , dans le cas d'un fluide visqueux ($\mu = 10^{-3}$).

Pour finir, figure 4.17 présente le temps d'exécution (en secondes) pour les trois différents solveurs dans les cas visqueux et non visqueux, sur le maillage le plus fin 256×256 . Pour obtenir la

même qualité de solutions numériques quel que soit le cas étudié (visqueux et non visqueux), nous constatons que la méthode de projection standard est la plus coûteuse en temps CPU par rapport aux deux autres méthodes. Nous observons également pour les deux cas que la méthode du lagrangien augmenté est plus rapide que la méthode tout-couplé. Ceci peut s'expliquer par le fait que les deux solveurs ont besoin d'une itération pour converger vers la solution exacte, et que la taille du système à résoudre dans le cas de la méthode tout-couplé est plus grande que celle de la méthode du lagrangien augmenté. En plus de cela, les résultats montrent que la méthode tout-couplé permet d'économiser 84% (cas non visqueux) et 63% (cas visqueux) du temps CPU par rapport à la projection standard, tout en consommant plus de 75% (cas non visqueux) et 22% (cas visqueux) du temps CPU par rapport à la méthode du lagrangien augmenté.

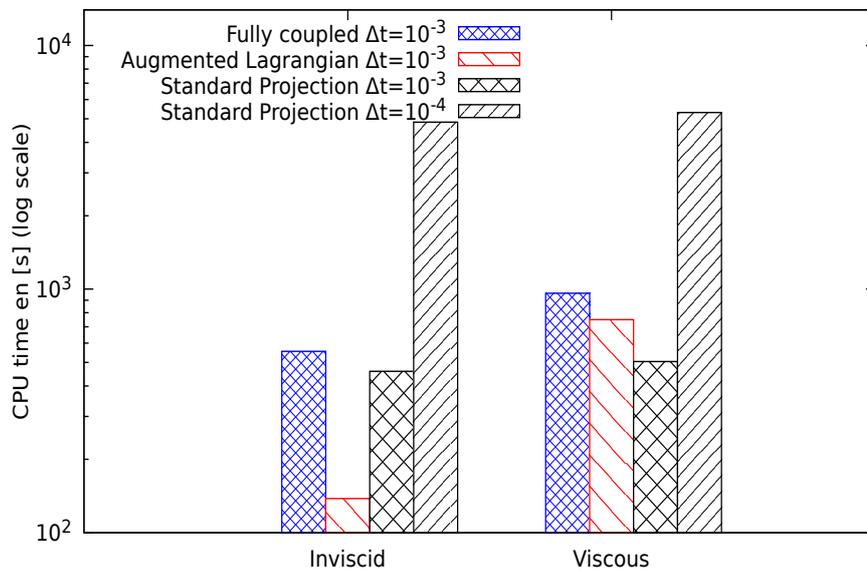


FIGURE 4.17 – Simulation numérique de l'écoulement de Taylor-Green 2D par le solveur tout-couplé, lagrangien augmenté et la projection standard. Temps de calcul CPU requis pour simuler l'écoulement de Taylor-Green sur une grille cartésienne fine 256×256 .

4.4.3 Goutte statique (loi de Laplace)

Afin de vérifier le traitement de la condition de saut à travers l'interface, un cas d'une goutte statique est étudié. Il consiste à définir une interface circulaire de rayon $r = 0.25m$ centrée en $(0.5m, 0.5m)$, et placée dans une cavité carré de coté $L = 1m$, comme cela est schématisé sur la figure 4.18. Les nombres caractéristiques qui pilotent ce cas (le nombre capillaire Ca et le nombre de Laplace La) sont définis par les équations (4.51) et (4.52) respectivement :

$$Ca = \frac{\|u\|_{\infty} \mu_1}{\sigma}, \quad (4.51)$$

$$La = \frac{\rho_1 D \sigma}{\mu_1^2}. \quad (4.52)$$

où ρ_1 et μ_1 correspondent aux masse volumique et viscosité de la goutte, σ est le coefficient de tension de surface et D le diamètre de la goutte.

Initialement, les deux fluides à l'intérieur et à l'extérieur de la goutte sont au repos, et en

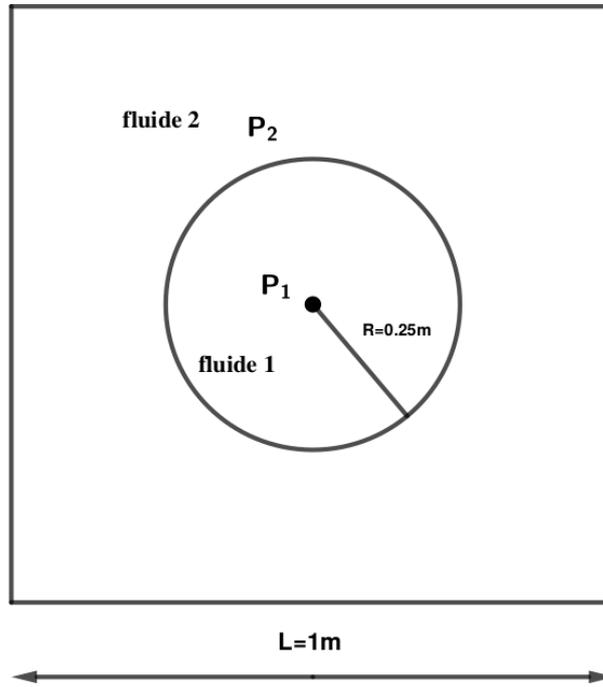


FIGURE 4.18 – Cas de la goutte statique

l'absence de la gravité, la solution théorique de ce cas-test est régie par la loi de Laplace. Cette loi s'exprime comme suit :

$$[p] = \sigma \varkappa = \frac{\sigma}{R} \quad (4.53)$$

où $[p] = p_2 - p_1$ représente le saut de pression entre les fluides 1 et 2, σ est le coefficient de tension de surface et \varkappa la courbure locale. Pour quantifier les erreurs commises sur la pression, nous avons calculé l'erreur relative du saut de pression par rapport à σ/R . Pour avoir zéro courant parasite, nous avons remplacé la courbure \varkappa dans les équations de bilan de quantité de mouvement par sa valeur exacte $\frac{1}{R}$. Nous avons ainsi défini l'erreur :

$$\|E_{\Delta P_{max}}\| = \frac{|\Delta P_{max} - \sigma/R|}{\sigma/R} \quad (4.54)$$

où ΔP_{max} représente la différence maximale entre les pressions à l'extérieur et à l'intérieur de la goutte.

le tableau 4.11 regroupe les résultats de ce test pour différents nombres de Laplace, pour un maillage 128×128 , avec un résidu de $\epsilon = 10^{-12}$ pour le solveur BiCGSTAB(2). Nous constatons bien que l'erreur sur la pression est de l'ordre de l'erreur machine, quel que soit le nombre de Laplace. Nous faisons les mêmes remarques pour l'évolution du nombre capillaire en fonction du nombre de Laplace.

La	$\ E_{\Delta P_{max}}\ $	Ca
0.5	2,57E-14	5,63E-14
5	2,78E-13	4,78E-16
50	4,40E-15	1,52E-16
500	3,29E-15	1,11E-16
5000	8,54E-15	2,83E-17
50000	3,20E-15	3,25E-17

TABLE 4.11 – Simulation numérique de l'écoulement d'une goutte statique 2D par le solveur tout-couplé. Erreur sur le saut de pression $\|E_{\Delta P_{max}}\|$ dans la goutte et évolution des nombres capillaires en fonction du nombre de Laplace (valeurs précises après 5 secondes de simulation).

4.4.4 Montée d'une bulle dans un liquide

Le problème de la montée d'une bulle dans un liquide au repos est classique dans la littérature. Il met en jeu une physique très variée et des changements de topologie. Afin de valider les résultats de nos simulations numérique, notre attention a été portée vers l'étude proposée par Hysing et al [98], en utilisant l'approche tout-couplé, le lagrangien augmenté adaptatif avec $r = 10^3$ et la méthode de projection standard. L'état initial de la simulation est simple, elle se compose d'une boîte rectangulaire ($L_x \times L_y$) remplie du liquide dans laquelle une bulle de diamètre $D = 0.5m$ est placée au point $(0.5m, 0.5m)$, comme cela est schématisé sur la figure 4.19.

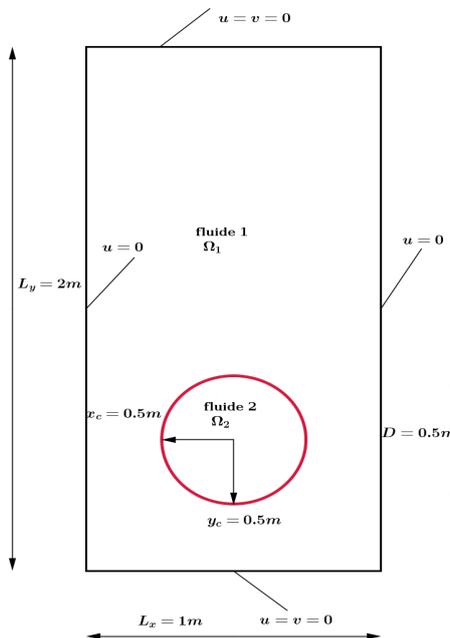


FIGURE 4.19 – Configuration initiale du problème de la montée d'une bulle dans une colonne liquide.

La bulle est moins dense que le fluide environnant ($\rho_2 < \rho_1$). Elle est entraînée vers le haut par la force d'Archimède ($g = 9.81m \cdot s^{-2}$), tandis que sa déformation est contrebalancée par la tension de surface $\sigma = 1.98N \cdot m^{-1}$. Aux limites, nous imposons des conditions de non glissement sur les parois horizontales, alors que sur les parois verticales, nous fixons une condition de glissement.

Nous choisissons ici de traiter la seconde configuration d’Hysing, qui est caractérisée par des forts rapports de masses volumiques et de viscosités, respectivement ($\rho_1/\rho_2 = 10^3$ et $\mu_1/\mu_2 = 10^2$). L’intérêt principal de cette configuration est de comparer le temps de résolution des trois solveurs. Pour pouvoir effectuer des comparaisons détaillées, trois pas de temps ont été choisis : $\Delta t = 10^{-3}s$, $5 \cdot 10^{-3}s$ et $10^{-2}s$ tout en maintenant le résidu constant ($\epsilon = 10^{-7}$ pour le solveur BiCGSTAB(2)).

Pour l’analyse qualitative des résultats, nous avons considéré l’évolution temporelle de deux quantités :

— Le centre de masse Y_c

$$Y_c = \frac{\int_{\Omega_2} y \, dx}{\int_{\Omega_2} 1 \, dx}, \quad (4.55)$$

— La vitesse de montée V_c

$$V_c = \frac{\int_{\Omega_2} V \, dx}{\int_{\Omega_2} 1 \, dx}. \quad (4.56)$$

Ici, Y_c correspond au centre de masse de la bulle, V_c désigne la vitesse de montée de la bulle, et Ω_2 représente le sous-domaine occupé par la bulle.

Avant de comparer les trois solveurs, les solutions numériques sur la grille la plus fine ($h = 1/320$) avec le solveur tout-couplé sont confrontées aux solutions de références, obtenues grâce aux codes de calcul de différentes équipes à savoir TP2D (groupe 1), FreeLIFE (groupe 2) et MoonMD (groupe 3). Comme nous pouvons le voir sur la Fig. 4.20b, l’estimation du centre de masse par le solveur tout-couplé est en bon accord avec les valeurs de référence. Quant à la vitesse de montée de la bulle, les solutions semblent s’écarter les unes des autres à partir de $t = 1s$. Pour des temps longs ($t = 2.5s$), la solution du solveur tout-couplé et la référence du groupe 2 semblent converger, mais malgré tout il est difficile de faire une discrimination entre les différents codes.

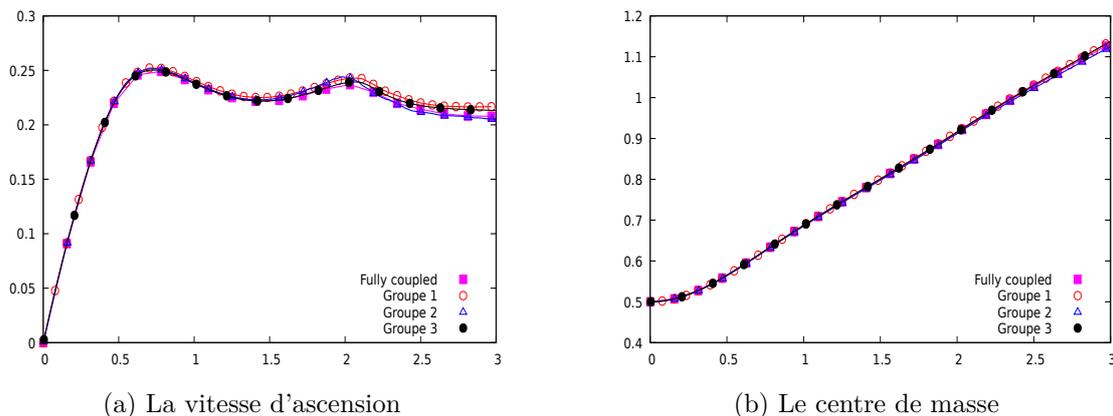


FIGURE 4.20 – Simulation numérique de l’écoulement d’une montée d’une bulle dans une colonne liquide par le solveur tout-couplé. Comparaison de la vitesse d’ascension 4.20a et de la position du centre de masse 4.20b pour le cas-test 2 ($\rho_1/\rho_2 = 10^3$, $\mu_1/\mu_2 = 10^2$). Comparaison entre le solveur tout-couplé et les solutions des autres groupes [98].

Les figures 4.21, 4.22 et 4.23 illustrent l’évolution temporelle de la vitesse de montée et le

centre de masse de la bulle, obtenue sur un maillage fin 320×640 avec les solveurs tout-couplé, lagrangien augmenté et la projection standard. Les solutions procurent des résultats en bon accord avec ceux de Hysing et al.[98], bien qu'ils ne soient pas présentés ici. Les simulations montrent également que les résultats obtenus par les solveurs tout-couplé et lagrangien augmenté sont insensibles au pas de temps. En effet, pour ces deux solveurs, la vitesse de montée et l'évolution du centre de masse sont qualitativement similaires quel que soit le pas de temps utilisé. Cela diffère donc de la méthode de projection standard, pour laquelle de grands pas de temps induisent de toute évidence des erreurs significatives sur la vitesse de montée (voir Fig.4.23).

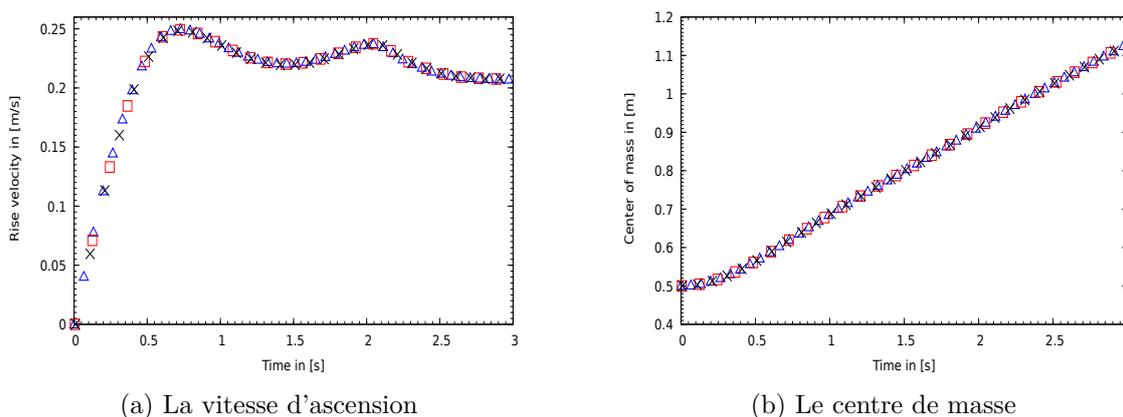


FIGURE 4.21 – Simulation numérique de l'écoulement d'une montée d'une bulle dans une colonne d'eau par le solveur tout-couplé. Comparaison de la vitesse d'ascension 4.21a et de la position du centre de masse 4.28c pour le cas-test 2 ($\rho_1/\rho_2 = 10^3$, $\mu_1/\mu_2 = 10^2$). Les solutions ont été obtenues pour différents pas de temps : $\Delta t = 10^{-3} s$ (représenté par des triangles bleus), $\Delta t = 5 \cdot 10^{-3} s$ (représenté par des carrés rouges) et $\Delta t = 10^{-2} s$ (représenté par des croix noires).

La figure 4.24 compare la forme de la bulle à $t = 3s$ pour les trois solveurs en utilisant différents pas de temps. Il ressort de ces figures que les solveurs tout-couplé et lagrangien augmenté sont plus précis que la méthode de projection standard. En effet, dans les régions résolues de la bulle, la forme globale de la bulle est convergée, et les régions filamentaires ressemblent aux résultats de référence [98]. Ici, les solutions n'ont pas été précisément comparées à celles de Hysing, car même dans le cas de Hysing, il existe une dispersion importante entre les méthodes, et aucune ne peut être considérée comme une solution de référence. Mais malgré tout, la forme globale de la bulle est similaire à celle de Hysing, alors que pour la méthode de projection standard, nous notons une grande différence lorsque le pas de temps est modifié, dans les régions résolues mais aussi au niveau des ligaments. Pour le deuxième cas ($\rho_1/\rho_2 = 10^3$, $\mu_1/\mu_2 = 10^2$), nous concluons que les méthodes tout-couplé et lagrangien augmenté sont en mesure de procurer des meilleurs résultats avec de grands pas de temps que la méthode de projection.

Dans ce qui suit, nous considérons les temps CPU nécessaires à chacune des méthodes pour simuler la montée de la bulle. Ces temps CPU sont illustrés dans la figure 4.25. Pour les trois pas de temps, la méthode du lagrangien augmenté est la plus lente à converger. D'autre part, la différence entre la méthode tout-couplé et la méthode de projection standard est significative, ainsi

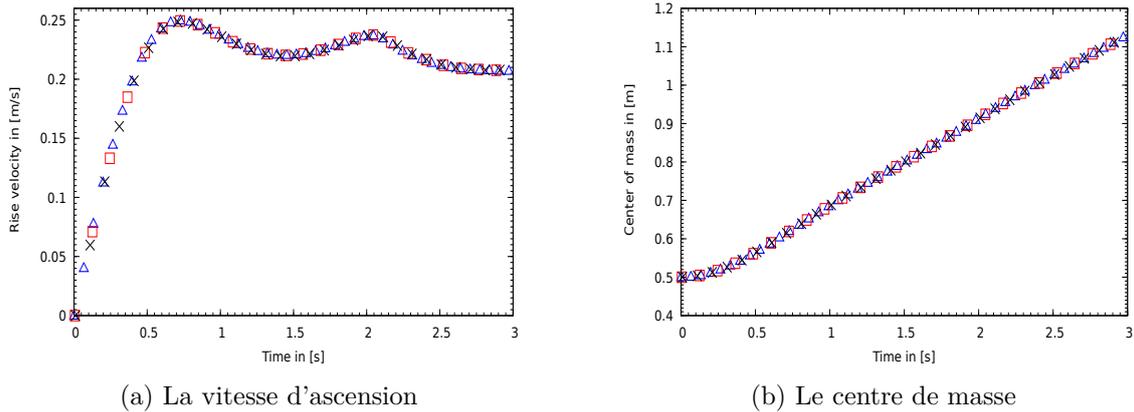


FIGURE 4.22 – Simulation numérique de l'écoulement d'une montée d'une bulle dans une colonne liquide par la méthode du lagrangien augmenté adaptatif. Comparaison de la vitesse d'ascension 4.22a et de la position du centre de masse 4.22b pour le cas-test 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$). Les solutions ont été obtenues pour différents pas de temps : $\Delta t = 10^{-3}s$ (représenté par des triangles bleus), $\Delta t = 5 \cdot 10^{-3}s$ (représenté par des carrés rouges) et $\Delta t = 10^{-2}s$ (représenté par des croix noires).

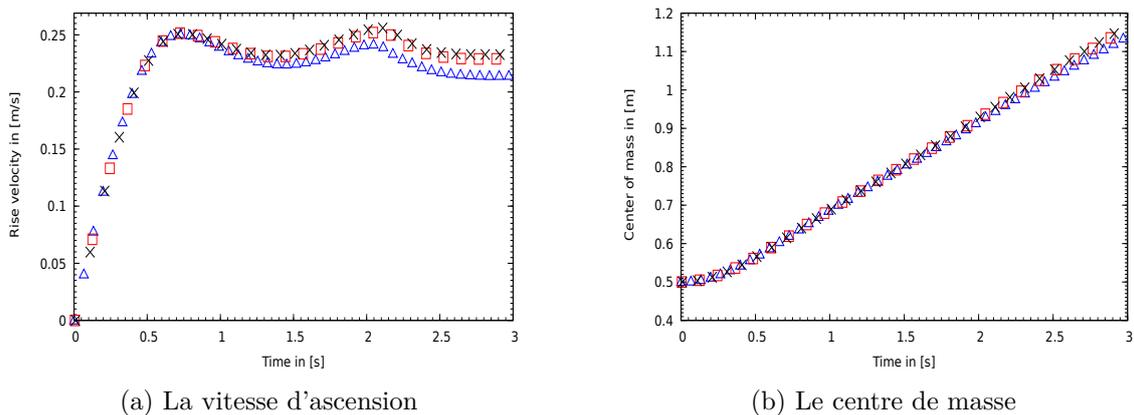


FIGURE 4.23 – Simulation numérique de l'écoulement d'une montée d'une bulle dans une colonne liquide par la méthode de la projection standard. Comparaison de la vitesse d'ascension 4.23a et de la position du centre de masse 4.23b pour le cas-test 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$). Les solutions ont été obtenues pour différents pas de temps : $\Delta t = 10^{-3}s$ (représenté par des triangles bleus), $\Delta t = 5 \cdot 10^{-3}s$ (représenté par des carrés rouges) et $\Delta t = 10^{-2}s$ (représenté par des croix noires).

pour le petit pas de temps $\Delta t = 10^{-3}s$ le solveur tout-couplé économise 80% du temps CPU par rapport au lagrangien augmenté, alors qu'elle économise 57% de temps par rapport à la méthode de projection standard. Ces écarts importants sont liés au fait que le lagrangien augmenté et projection standard nécessitent un grand nombre d'itérations pour converger. Lorsque des forts rapports de masses volumiques et de viscosités sont considérés, l'utilisation d'une grande valeur du paramètre r est nécessaire pour satisfaire la contrainte d'incompressibilité. Cela conduit à un système linéaire très mal conditionné, d'où le coût élevé de la méthode du lagrangien augmenté. Pour la projection standard, le coût du calcul est lié à la résolution de l'équation de Poisson à coefficients variables.

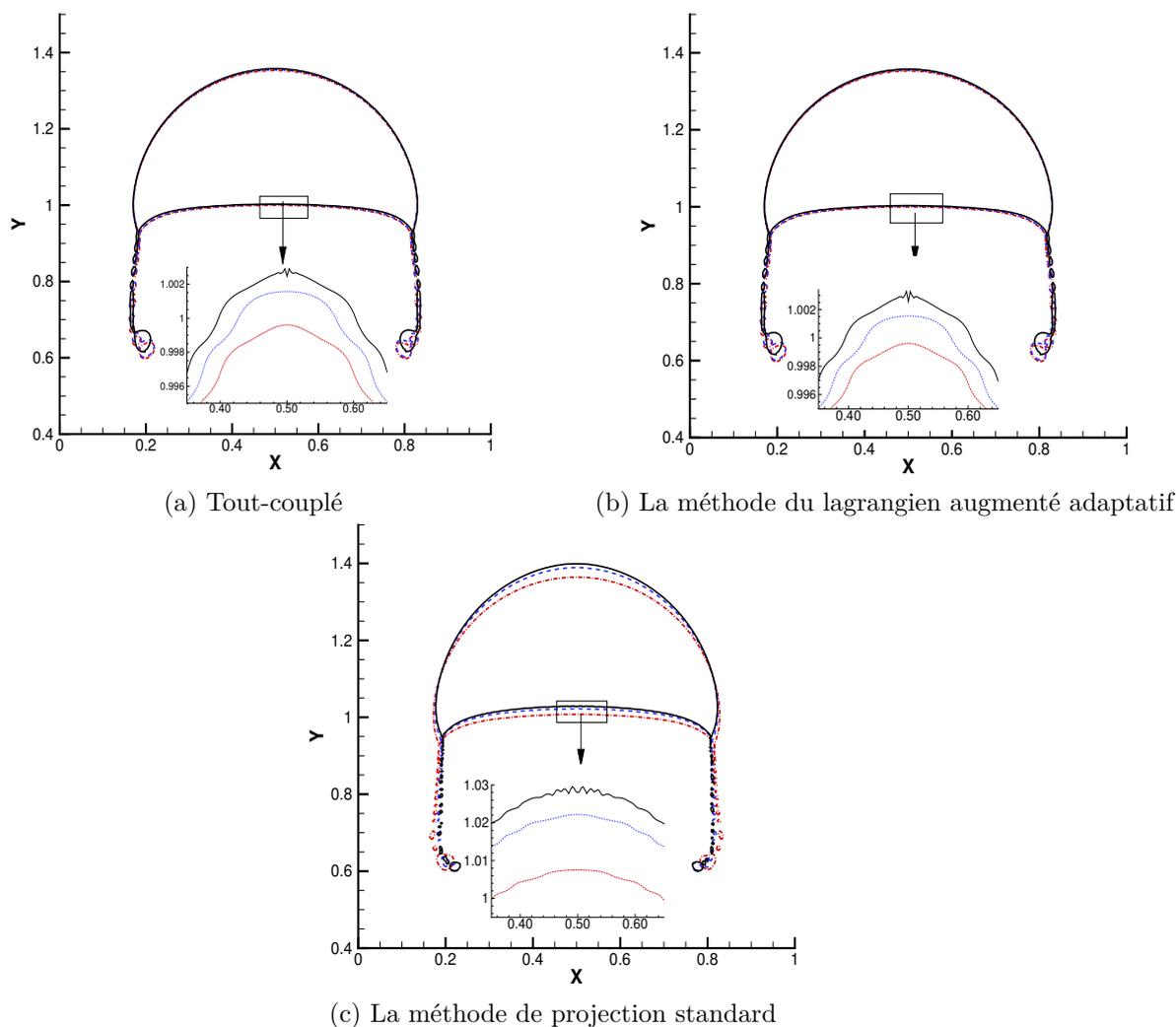


FIGURE 4.24 – Simulation numérique de l’écoulement d’une montée de bulle dans une colonne liquide par le solveur tout-couplé. Formes de l’interface à l’instant final $t = 3s$ pour le cas test 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$). Plusieurs pas de temps ont été utilisé : $\Delta t = 10^{-3}s$ (représenté par des lignes en pointillés en rouge), $\Delta t = 5 \cdot 10^{-3}s$ (représenté par des lignes discontinues en bleu) et $\Delta t = 10^{-2}s$ (représenté par des lignes pleines en noir).

4.4.5 Chute d’une sphère dense dans l’air

Le présent cas test a pour objectif de tester la capacité du solveur tout-couplé à simuler des écoulements diphasiques avec des forts rapports de masses volumiques et de viscosités, et à comparer ce solveur à la méthode de projection standard et la méthode du lagrangien augmenté adaptatif. La configuration étudiée correspond à un cylindre dense de masse volumique ρ_2 , de viscosité dynamique μ_2 et de rayon $r = 0.0125m$. Initialement centré à $(0.05 m, 0.15 m)$, le cylindre est lâché sans vitesse initiale dans une cavité rectangulaire remplie d’air, ayant pour densité et viscosité respectivement ρ_1 et μ_1 . Les dimensions de la cavité sont 0.2 m de hauteur et 0.1 m de largeur. Pour les conditions aux limites, nous imposons des conditions de non glissement sur les parois horizontales, alors que sur les parois verticales, nous fixons une condition de glissement. L’intérêt de ce cas découle de la prise en compte des forts rapports de masses volumiques et de viscosités. De plus, si on suppose que l’air n’a pas d’effet sur la chute de la sphère dense, une

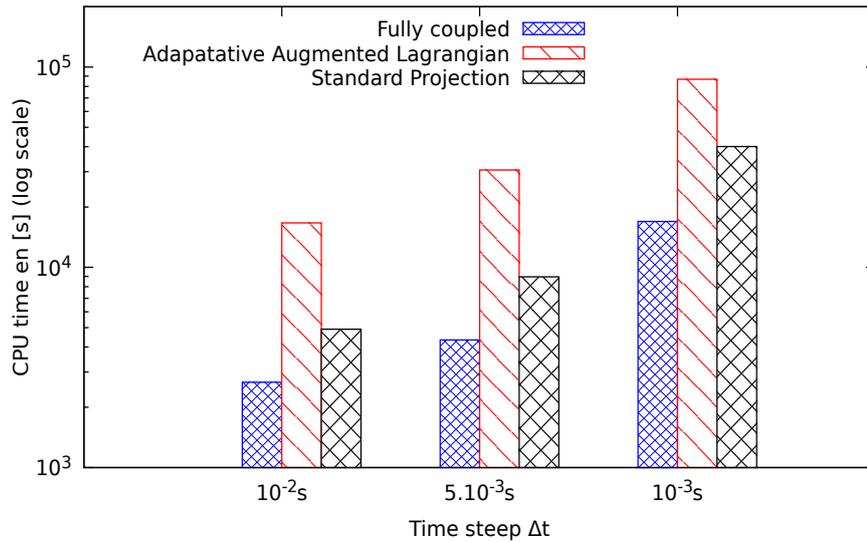


FIGURE 4.25 – Simulation numérique de l’écoulement d’une montée d’une bulle dans une colonne liquide par le solveur tout-couplé, lagrangien augmenté et la projection standard. Temps de calcul CPU requis pour simuler l’écoulement de la montée de la bulle sur une grille cartésienne fine 320×640 .

solution analytique de la vitesse de cette bille V_C peut être facilement trouvée : $V_c = -gt$.

Au regard des paramètres numériques, les simulations sont effectuées sur cinq grilles cartésiennes (50×100 , 100×200 , 200×400 , et 400×800), avec un résidu $\epsilon = 10^{-6}$ pour le solveur BiCGSTAB(2). Pour les dérivées temporelles, un pas de temps constant $\Delta t = 6.25 \times 10^{-5} s$ est choisi, et 2300 pas de temps ont été effectués, correspondant à un temps physique de l’ordre de 0.144 secondes. Les paramètres physiques du cas-test sont fournis dans le tableau 4.12.

Test case	ρ_1	ρ_2	μ_1	μ_2	g	ρ_2/ρ_1	μ_2/μ_1
1	1.1768	10^5	1.85×10^{-5}	10^5	-9.81	10^5	10^{10}

TABLE 4.12 – Simulation numérique de la chute d’une sphère dense dans l’air. Ici, ρ_2 et μ_2 désignent la masse volumique et la viscosité du fluide à l’intérieur de la particule ρ_1 et μ_1 désignent la masse volumique et la viscosité du fluide environnant et g représente la gravité.

En analysant les résultats numériques (Fig. 4.26), nous pouvons constater que les solveurs tout-couplé et le lagrangien augmenté adaptatif restent robustes et précis, même en présence de forts rapports de masses volumiques et viscosités. Ainsi, un bon accord est observé entre les deux méthodes, avec de légères différences sur les champs de vorticit . Dans les m mes conditions, la m thode de projection standard ne parvient pas   converger   la fin de la simulation vers la solution souhait e. En effet, le solveur BiCGStab(2) n’a pas pu atteindre le r sidu souhait  pour r soudre l’ quation de Poisson   coefficients variables et par cons quent, l’ tape de correction ne permettait pas de r duire le niveau de divergence  lev  de la vitesse pr dite. Les r sultats pr sent s pour la projection utilisent un pas de temps 10 fois plus petit que les 2 autres solveurs

pour qu'elle converge.

Commentaires sur la vitesse de chute et le centre de masse

L'objectif de cette partie est d'évaluer la précision du solveur tout-couplé et de comparer les solutions numériques aux solutions exactes. Pour ce faire, nous avons observé au cours du temps deux quantités distinctes. Ces quantités sont définies comme suit :

— Centre de masse Y_c

$$Y_c = \frac{\int_{\Omega_2} y \, dx}{\int_{\Omega_2} 1 \, dx}, \quad (4.57)$$

— Vitesse de chute v_c

$$V_c = \frac{\int_{\Omega_2} v \, dx}{\int_{\Omega_2} 1 \, dx}. \quad (4.58)$$

Ici, Y_c correspond au centre de masse de la bille, V_c désigne la vitesse de chute de la bille, et Ω_2 représente le sous-domaine occupé par la bille. Sur le plan quantitatif nous avons comparé nos résultats numériques (4.57) et (4.58) aux solutions exactes qui peuvent être réécrites sous la forme :

$$V_c^{exacte} = -gt \quad (4.59)$$

$$Y_c^{exacte} = -\frac{gt^2}{2} + 0.15 \quad (4.60)$$

Pour évaluer les erreurs numériques commises, nous avons utilisé l'erreur relative en norme L^2 , qui s'écrit comme suit :

$$Err_2^{rel}(q_t) = \left(\frac{\sum_{k=1}^{nt} \|q_t^{exacte} - q_t\|^2}{\sum_{k=1}^{nt} \|q_t^{exacte}\|^2} \right)^{1/2} \quad (4.61)$$

Ici, q_t^{exacte} est la quantité exacte, q_t est la quantité numérique et n_t représente le nombre de pas de temps.

Sur la figure 4.27 sont présentés les résultats des simulations de la chute d'une particule dense. Les solutions obtenues sur les différents maillages sont confrontées aux solutions exactes. De cette figure, il apparait clairement que les solutions obtenues convergent bien vers la solution exacte, y compris pour le maillage grossier. Le tableau 4.13 regroupe les normes d'erreurs calculées pour la position du centre de masse et pour la vitesse de la chute de la particule. Il ressort de ce tableau que le niveau d'erreur est toujours faible pour le solveur tout-couplé. Même si les erreurs sont saturées (du fait que les simulations sont effectuées avec un fluide dense et visqueux autour de la particule alors que la théorie suppose que le cas se fait dans le vide), l'écart relatif entre la théorie et la simulation numérique n'excède pas 0.01%.

Afin de caractériser l'influence des méthodes numériques et de leur discrétisation, nous allons

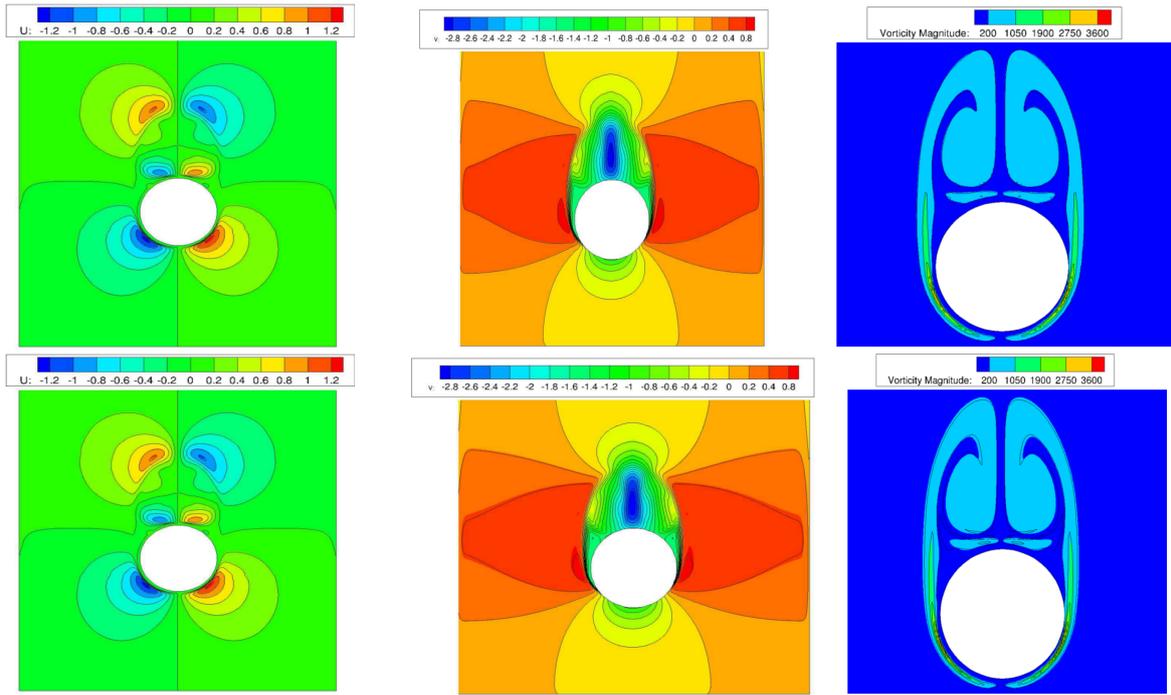
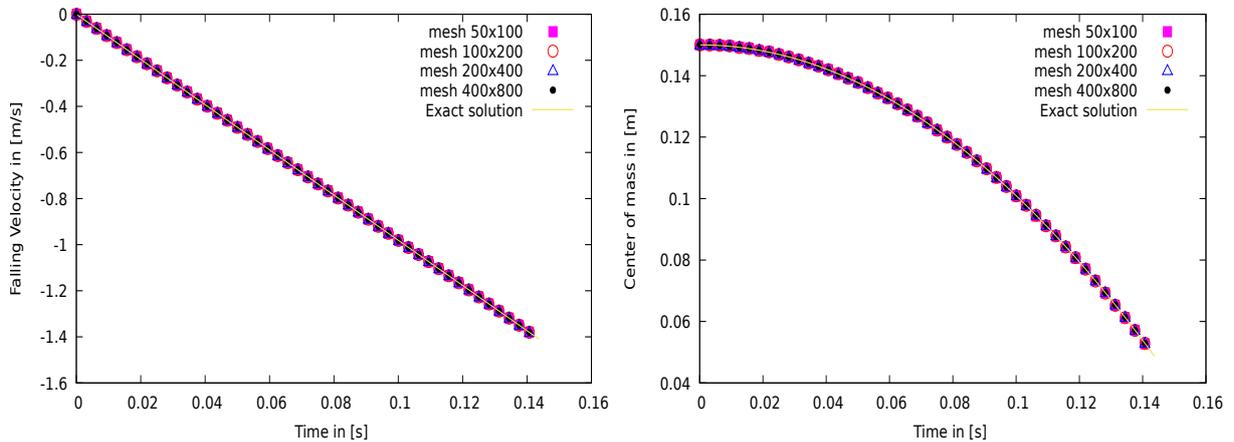


FIGURE 4.26 – Simulation numérique de la chute d’une sphère dense dans l’air par le solveur tout-couplé, le lagrangien augmenté et la méthode de projection standard sur une grille cartésienne 400×800 . Dans la première ligne, le solveur tout-couplé (représenté par les champs) et le lagrangien augmenté (par les isolignes). Dans la deuxième rangée, le solveur tout-couplé (représenté par les champs) et la méthode de projection standard (représentée par les isolignes avec $\Delta t = 6.25 \times 10^{-6}$ s). Les différents champs présentés dans chaque ligne sont : à gauche : vitesse horizontale $[0 \text{ m}, 0.1 \text{ m}] \times [0 \text{ m}, 0.11 \text{ m}]$, au milieu : vitesse verticale $[0 \text{ m}, 0.1 \text{ m}] \times [0 \text{ m}, 0.10 \text{ m}]$, à droite : la norme de la vorticité $[0.02 \text{ m}, 0.08 \text{ m}] \times [0.034 \text{ m}, 0.089 \text{ m}]$.



(a) La vitesse verticale de la chute

(b) L’évolution du centre de masse

FIGURE 4.27 – Simulation numérique de la chute d’une sphère dense dans l’air par le solveur tout-couplé sur différentes grilles cartésiennes avec $(\rho_1/\rho_2 = 10^5, \mu_1/\mu_2 = 10^{10})$.

comparer les résultats issus du solveur tout-couplé sur la grille la plus fine à ceux fournis par la projection standard et lagrangien augmenté adaptatif, les performances de chaque solveur sont aussi évaluées. L’ensemble des résultats est présenté dans le tableau 4.14. Comme illustré dans ce dernier, les niveaux d’erreur sont comparables entre le solveur tout-couplé et la méthode

Maillage	Δt	$Err_2^{rel}(V_c)$	$Err_2^{rel}(Y_c)$
50×100	$6.25 \times 10^{-5} s$	5.05×10^{-5}	8.41×10^{-4}
100×200	$6.25 \times 10^{-5} s$	2.37×10^{-5}	5.74×10^{-4}
200×400	$6.25 \times 10^{-5} s$	2.62×10^{-5}	2.38×10^{-4}
400×800	$6.25 \times 10^{-5} s$	3.07×10^{-5}	2.21×10^{-4}

TABLE 4.13 – Simulation numérique de la chute d’une sphère dense dans l’air par le solveur tout-couplé sur différentes grilles cartésiennes avec ($\rho_1/\rho_2 = 10^5, \mu_1/\mu_2 = 10^{10}$). Normes d’erreurs $Err_2^{rel}(V_c)$ de la vitesse de chute et $Err_2^{rel}(Y_c)$ de l’évolution du centre de masse.

Method	$Err_2^{rel}(V_c)$	$Err_2^{rel}(Y_c)$	CPU Time in [s]
FC avec $\Delta t = 6 \cdot 25 \times 10^{-5} s$	5.05×10^{-5}	2.21×10^{-4}	44986
AAL avec $\Delta t = 6 \cdot 25 \times 10^{-5} s$	6.63×10^{-5}	1.25×10^{-4}	101933
SP avec $\Delta t = 6 \cdot 25 \times 10^{-5} s$	NC	NC	-
SP avec $\Delta t = 6 \cdot 25 \times 10^{-6} s$	3.36×10^{-5}	7.05×10^{-5}	486941

TABLE 4.14 – Simulation numérique de la chute d’une sphère dense dans l’air par le solveur tout-couplé, le lagrangien augmenté et la méthode de projection standard sur une grille fine de taille 400×800 avec ($\rho_1/\rho_2 = 10^5, \mu_1/\mu_2 = 10^{10}$). Normes d’erreurs $Err_2^{rel}(V_c)$ de la vitesse de chute et $Err_2^{rel}(Y_c)$ de l’évolution du centre de masse obtenues à l’instant final $t = 0.144s$. FC : le solveur tout-couplé, AAL : le lagrangien augmenté adaptatif et SP : la méthode de projection standard. NC (Non Communiqué), c’est à dire que la simulation s’est interrompue en raison de la divergence du solveur.

du lagrangien augmenté adaptatif pour le même pas de temps $\Delta t = 6.25 \times 10^{-5} s$. Cependant, le solveur tout-couplé se révèle plus rapide car il économise 56% du temps CPU par rapport au lagrangien augmenté adaptatif. Ces grandes différences peuvent être expliquées par le fait que la méthode du lagrangien augmenté adaptative nécessite un grand nombre d’itérations pour converger, tout cela avec une saturation du solveur BICGStab(2). En effet en présence de forts rapports de masses volumiques et de viscosités, des grandes valeurs du paramètre r du lagrangien augmenté seront nécessaires afin d’imposer la contrainte d’incompressibilité. Par conséquent, la discrétisation des équations de conservation donne lieu à des systèmes linéaires très mal conditionnés, et rend la convergence du solveur itératif délicate. En ce qui concerne la méthode de projection standard, des difficultés numériques apparaissent liées à la présence d’un fort saut de masse volumique et plus particulièrement à la résolution de l’équation de Poisson. Enfin, pour procurer la même qualité de solution que les autres solveurs, la méthode de projection requiert une réduction du pas de temps. Ce choix apparaît judicieux pour réduire la dissipation numérique causée par le splitting temporel. Néanmoins, elle rend la méthode de projection standard extrêmement coûteuse en temps de calcul.

Commentaires sur l’Enstrophie

L’enstrophie est une quantité intégrale qui est associée à la dissipation et aux effets de la turbulence à petite échelle. Il est important de relever qu’il s’agit d’une étude bidimensionnelle et qu’en deux dimensions, la cascade d’énergie qui pourrait être liée à la turbulence et à l’enstrophie est différente de celle qui se produit en trois dimensions. Mathématiquement, l’enstrophie est définie

comme étant l'intégrale du carré de la vorticité sur le domaine, et elle s'écrit Ens_1 pour le fluide 1 :

$$Ens_1 = \int_V \frac{\omega^2}{2} dV \quad (4.62)$$

où $\omega = \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right)$ est la vorticité en deux dimensions. La partie discrète de l'enstrophie Ens_1 , dans laquelle $C_{i,j}$ est la fonction de couleur située aux nœuds de viscosité ($C_{i,j} = 0$ dans le fluide 1 et $C_{i,j} = 1$ dans le fluide 2) s'écrit :

$$Ens_1 \simeq \sum_{i,j} (1 - C_{i,j}) \frac{\omega_{i,j}^2}{2} \Delta V \quad (4.63)$$

Dans cette partie, nous étudions l'évolution de l'enstrophie dans le fluide 1. Les résultats sont illustrés dans la figure 4.28 en échelle logarithmique, pour différents maillages. Comme nous pouvons le voir sur cette figure, l'enstrophie augmente avec le temps de façon monotone. En ce qui concerne la convergence de l'enstrophie, la simulation sur une grille de 200×400 est suffisante pour faire converger l'enstrophie pour les solveurs tout-couplé et lagrangien augmenté adaptatif. En revanche, l'enstrophie diverge sur la grille la plus fine pour la projection standard, pour les mêmes raisons que celles évoquées précédemment.

4.4.6 Nappe liquide cisailée périodique

Afin de montrer la capacité du solveur tout-couplé à conserver l'énergie cinétique, un cas-test académique est étudié. Ce cas consiste à définir une nappe liquide périodique, d'épaisseur $\delta = 3.10^{-4}$ m, et à la placer au centre d'un domaine carré, comme cela est schématisé sur la figure 4.29. Deux rapports de masses volumiques de 10 et 10^3 sont considérés. Les viscosités du liquide et du gaz ainsi que la tension de surface sont prises égales à zéro afin de conserver l'énergie cinétique initiale. Le champ de vitesse initiale est définie comme suit :

$$U = \begin{cases} u = A - 0.04 \cos\left(\frac{2\pi x}{L}\right) \frac{L}{2\pi} \frac{-2}{\delta} e^{-\frac{2y}{\delta}} \\ v = 0.04 \sin\left(\frac{2\pi x}{L}\right) e^{-\frac{2y}{\delta}} \end{cases} \quad (4.64)$$

$$A = \begin{cases} 30 \text{ m.s}^{-1} & \text{dans le gaz} \\ 2 \text{ m.s}^{-1} & \text{dans le liquide} \end{cases} \quad (4.65)$$

Les résultats obtenus pour les deux rapports de masse volumique avec le solveur tout-couplé et le lagrangien augmenté adaptatif sont reportés sur la figure 4.30. Dans les deux cas, l'énergie cinétique obtenue par le solveur tout-couplé est conservée dans le temps avec une erreur à la précision machine, ce qui est en bon accord avec la théorie. Dans le cas du lagrangien augmenté adaptatif, on note une légère augmentation de l'énergie cinétique mais les résultats sont plutôt encourageants. Au contraire, la projection standard subit une soudaine divergence et par la suite l'énergie cinétique diverge, y compris pour de petits pas de temps. De plus, il a été démontré qu'une discrétisation consistante de la quantité de mouvement et de la masse volumique améliore

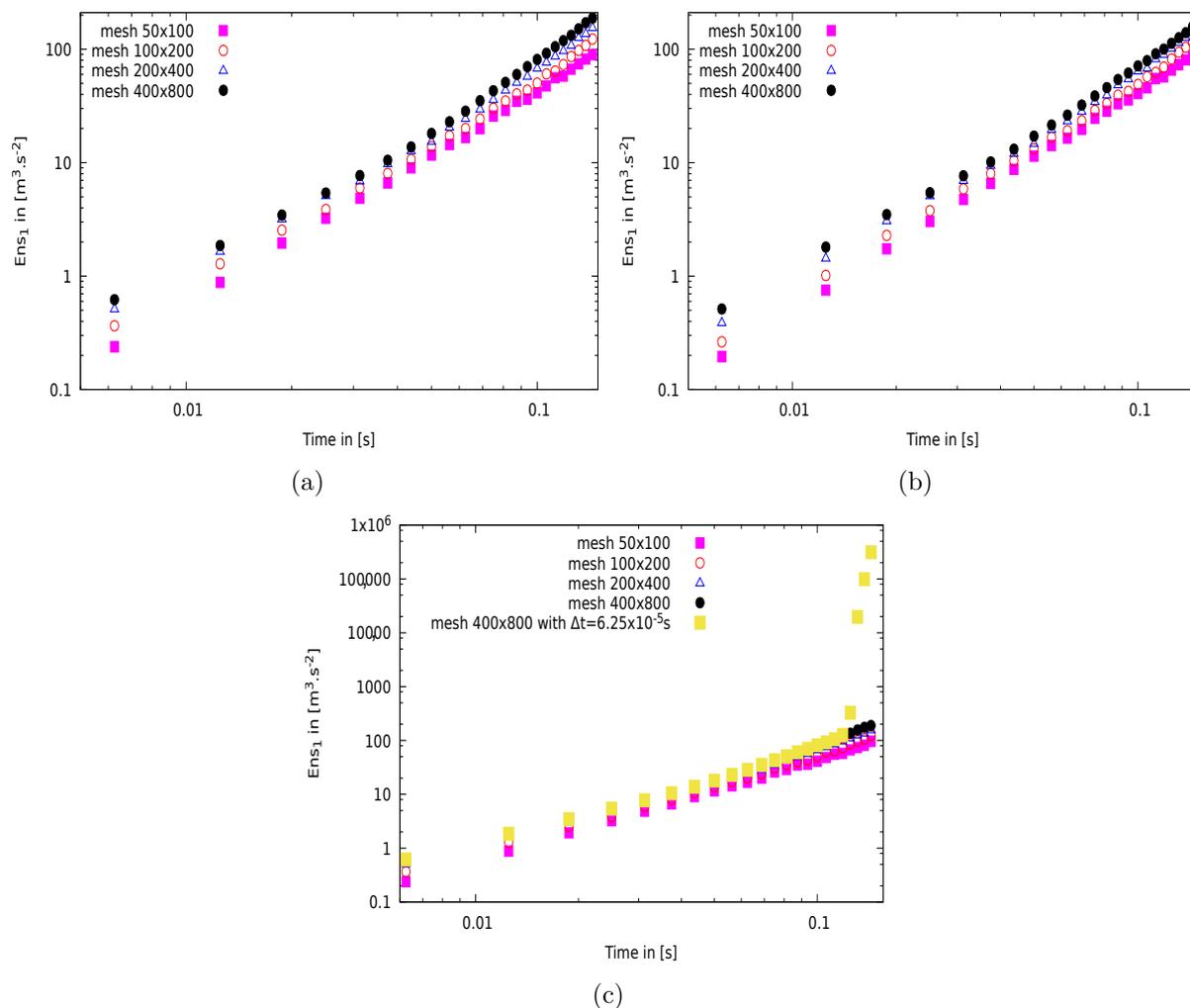


FIGURE 4.28 – Simulation numérique de la chute d’une sphère dense dans l’air sur différentes grilles cartésiennes avec $(\rho_1/\rho_2 = 10^5, \mu_1/\mu_2 = 10^{10})$. Évolution de l’ensrophie pour le solveur tout-couplé (a), le lagrangien augmenté adaptatif (b) et la projection standard (c). Tout est calculé avec $\Delta t = 6.2510^{-6}s$ sauf le cas jaune réalisé avec la projection standard.

significativement la conservation de l’énergie cinétique jusqu’à ce que l’interface commence à se déformer. Après cet instant, l’énergie cinétique décroît, comme cela est décrit par exemple dans les travaux de Mukundan et al [12]. En revanche, des instabilités interfaciales apparaissent plus rapidement pour les schémas WENO classique et Weno conservatif. Dans cette étude, un schéma centré utilisé dans le cadre du solveur tout-couplé ou du lagrangien augmenté conserve l’énergie cinétique pendant des temps plus longs sans l’apparition d’instabilités interfaciales.

4.4.7 Simulation de l’instabilité d’un jet visqueux en 2D

L’instabilité d’un jet visqueux à faible nombre de Reynolds est un phénomène fondamental, car il intervient dans de nombreuses applications industrielles telles que l’industrie alimentaire comme la confection de gâteaux, les applications aéronautiques à savoir la mise au point de fusées ou encore aérospatiales lors du remplissage des boosters par du propergol. Ce problème a été étudié expérimentalement par Tomé et al. [168]. qui ont permis entre autre de caractériser

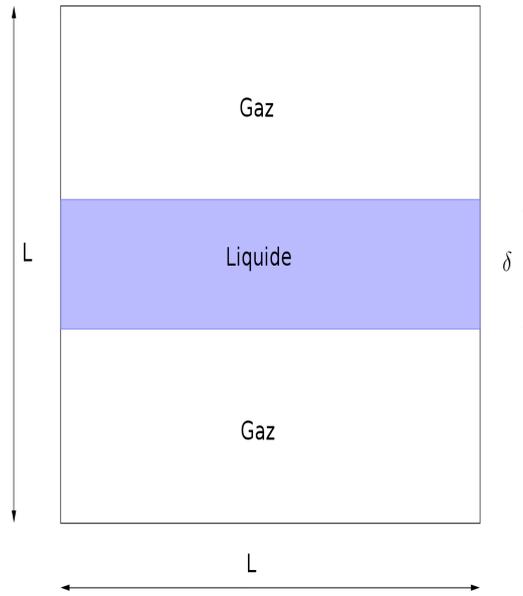
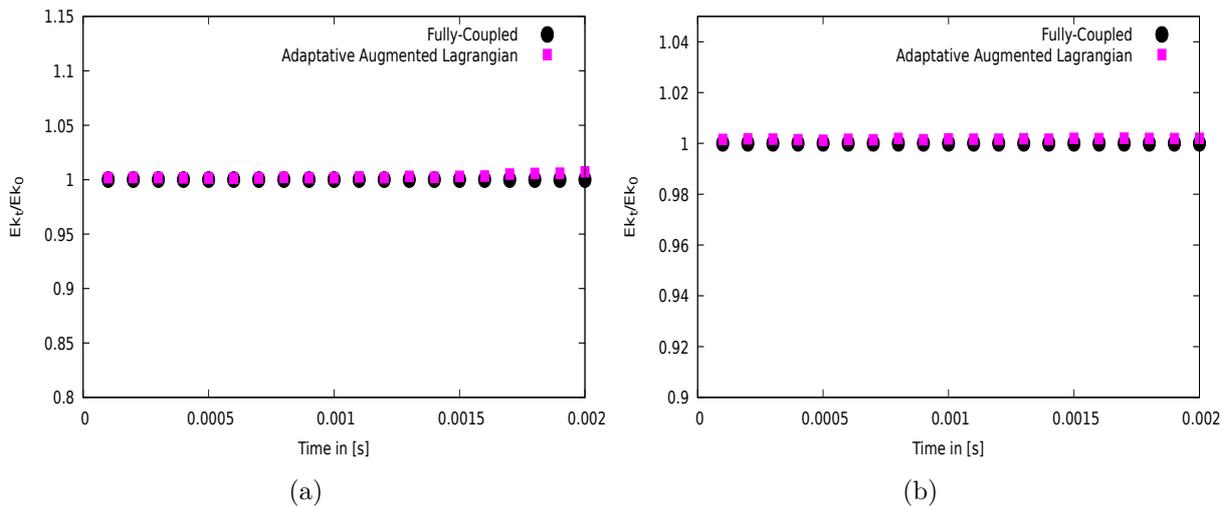


FIGURE 4.29 – Cas de la nappe liquide cisillée périodique.

FIGURE 4.30 – Simulation numérique d’une nappe liquide cisillée périodique sur une grille cartésienne (256×256). Évolution temporelle de l’énergie cinétique normalisée, obtenue par différents solveurs avec un critère d’arrêt $\epsilon = 10^{-6}$ du BiCGSTAB(2). Le rapport de masses volumiques (a) est de 100 et le rapport de masses volumiques (b) est 1000.

et d’établir des critères pour le déclenchement de l’instabilité. Ensuite, de nombreuses études numériques [179, 196] ont été menées pour simuler ce problème dans des configurations 2D et 3D, pour lesquelles les résultats de calcul ont été comparés à des expériences. Le présent cas est doublement intéressant : d’une part, la possibilité de comparer les solutions numériques avec les solutions expérimentales. D’autre part, les forts rapports de viscosités et de masses volumiques près de l’interface conduisent à des systèmes linéaires mal conditionnés, ce qui constitue un véritable challenge pour notre solveur tout-couplé.

La configuration à simuler consiste à injecter un fluide visqueux sous l’action de la gravité dans une cavité carrée de côté $L = 1m$ et remplie d’air. L’injection se fait à débit constant, avec une

vitesse égale à $V = 1 \text{ m}\cdot\text{s}^{-1}$ et sur une ouverture de diamètre $D = 0.08 \text{ m}$. Enfin, pour la prise en compte des conditions aux limites, nous avons utilisé des conditions limites de glissement sur les parois solides de la cavité et des conditions limites de flux nuls en sortie, dans la partie supérieure de la cavité. Les paramètres physiques de ces calculs sont explicités dans le tableau 4.15.

Test case	$\rho_1[\text{kg}\cdot\text{m}^{-3}]$	$\rho_2[\text{kg}\cdot\text{m}^{-3}]$	$\mu_1[\text{Pa}\cdot\text{s}]$	$\mu_2[\text{Pa}\cdot\text{s}]$	Re	$\frac{H}{D}$	$V_0[\text{m}\cdot\text{s}^{-1}]$
1	1.1768	1800	1.85×10^{-3}	500	0.1728	12.5	0.6
2	1.1768	1800	1.85×10^{-3}	300	0.6	10	1
3	1.1768	1800	1.85×10^{-3}	300	3.6	10	6

TABLE 4.15 – Simulation numérique de l’instabilité d’un jet visqueux en 2D par le solveur tout-couplé. Ici, ρ_2 et μ_2 correspondent respectivement à la masse volumique et à la viscosité du propergol, ρ_1 et μ_1 représentent respectivement la masse volumique et la viscosité du gaz, $Re = \frac{\rho_2 D V_0}{\mu_2}$ est le nombre de Reynolds associé au propergol, $\frac{H}{D}$ est le rapport de forme du jet et V_0 est la vitesse d’injection. Dans le présent travail, le propergol (fluide 2) est supposé se comporter comme un fluide newtonien, pour des raisons de simplicité.

Dans leur étude, Tomé et al. présentent deux paramètres importants afin de caractériser les limites de stabilité du jet en fonction du nombre de Reynolds Re et du rapport de forme H/D entre la hauteur de la cavité et le diamètre d’injection. D’après cette étude, des instabilités se développent pour des rapports de H/D supérieurs à 3π et pour un nombre de Reynolds inférieur à 0.52. Ainsi, l’objectif de cette étude est de comparer la performance de trois méthodes (tout-couplé, lagrangien augmenté adaptatif et la projection standard) pour traiter ce type de problème et de fournir une bonne prédiction de leur comportement physique associé. Les systèmes linéaires issus de la discrétisation seront résolus avec le même solveur, le BiCGSTAB(2) avec un critère d’arrêt $\epsilon = 10^{-6}$.

La figure 4.31 présente différents résultats de simulation d’injection qui sont caractéristiques de la topologie du jet et du remplissage de la cavité. L’évolution du propergol dans le temps est considérée pour trois nombres de Reynolds $Re = 0.1728$, $Re = 0.6$ et $Re = 3.6$ pour un maillage 320×320 et un pas de temps $\Delta t = 5 \cdot 10^{-4} \text{ s}$. De ces figures, il ressort que les simulations numériques avec le solveur tout-couplé donnent d’excellents résultats pour la physique du jet, qui sont en bon accord avec le critère théorique établi par Tomé et al. En effet, le premier cas $Re = 0.1728$ correspond bien à une instabilité, alors que dans les deuxième et troisième cas, $Re = 0.6$ et $Re = 3.6$, les oscillations du jet n’apparaissent pas. Dans le premier cas, nous observons une diminution du diamètre du jet dans la partie centrale de la cavité, mais le jet reste symétrique, ceci parce qu’il n’y a pas d’instabilité. Au contraire, dès que le propergol touche la paroi inférieure de la cavité, le jet se déstabilise, et une instabilité se développe et emprisonne les bulles d’air. Enfin, nous présentons les lignes de courant sur la figure 4.32 pour le premier cas, ce qui permet d’évaluer l’interaction entre les deux fluides. Sur ces figures, nous observons des recirculations dans l’air, dont la taille augmente puis diminue en alternance avec les oscillations du jet.

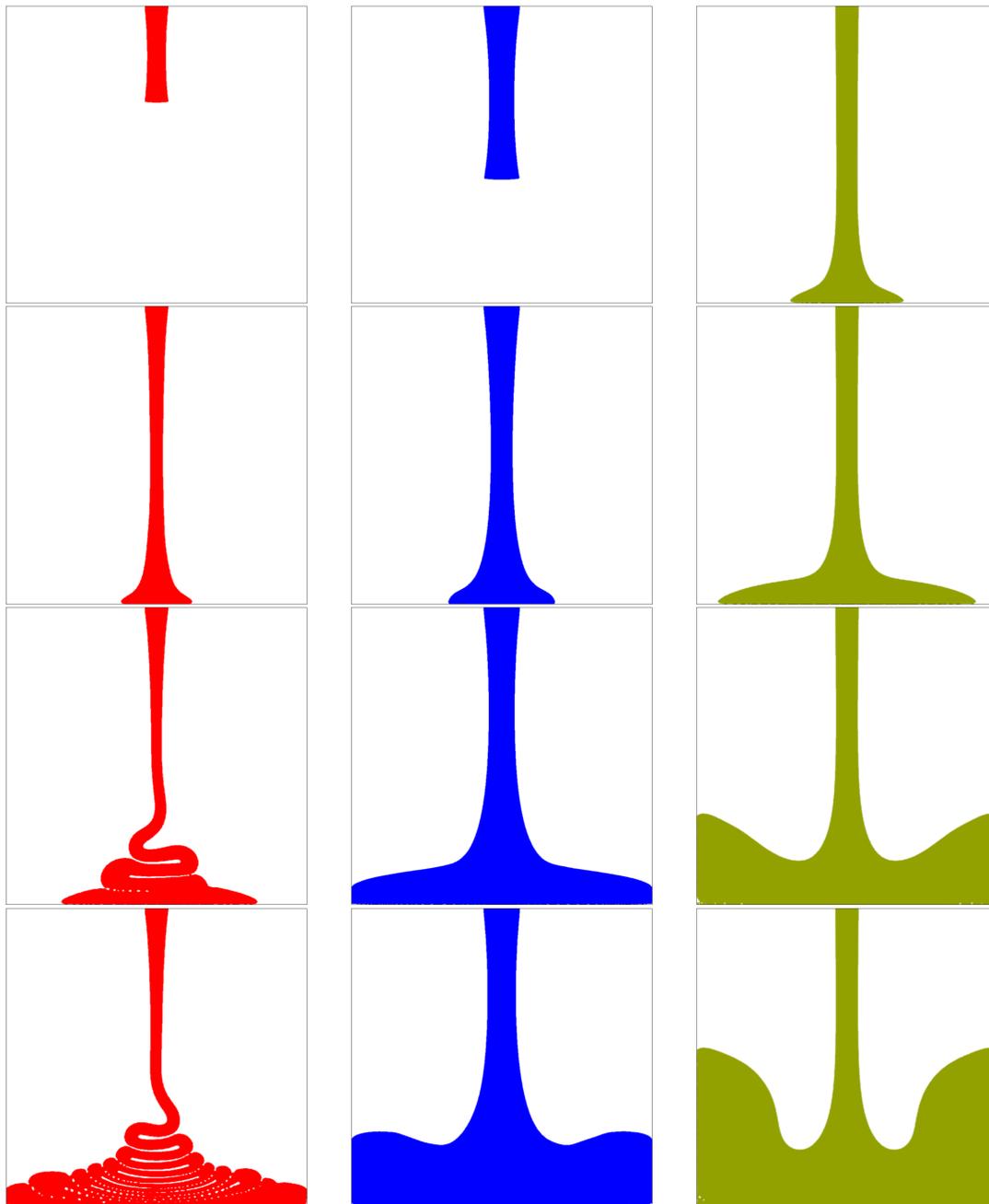


FIGURE 4.31 – Simulation numérique du remplissage d’une cavité par un jet visqueux. Résultats correspondant à trois configurations, simulées par le solveur tout-couplé, sur une grille cartésienne 320×320 . L’interface est représentée aux instants $t = 0.25$ s et 0.75 s (première ligne), $t = 1.25$ s et 2 s (deuxième ligne). La colonne 1 correspond à $Re = 0.1728$, la colonne 2 correspond à $Re = 0.6$ et la colonne 3 correspond à $Re = 3.6$.

Les résultats ainsi obtenus avec la méthode du lagrangien augmenté adaptatif sont présentés dans la figure 4.33 pour $Re = 0.1728$. Les résultats montrent bien une cohérence entre le solveur tout-couplé et la méthode du lagrangien augmenté adaptatif aux temps $t = 1$ s et $t = 2$ s. Ensuite, nous relevons des différences significatives entre ces deux solveurs puisque l’instabilité ne se produit pas au même moment, ceci étant dû à la nature du paramètre r du lagrangien augmenté, qui se comporte comme une viscosité. Ces écarts sont également causés par la nature instable de

l'écoulement qui amplifie différemment les erreurs numériques en fonction de la différence des solveurs qui ne sont pas pareillement sensibles aux erreurs de troncature. En dépit de tout, le plus important est que ces méthodes déclenchent une instabilité pour le même Re et dans la même situation d'injection. Toutefois, comme nous l'observons sur la figure 4.33, les simulations obtenues avec la méthode de projection divergent pour $Re = 0.1728$ après 2800 itérations. Ceci est encore une conséquence directe de l'étape de correction, qui ne parvient pas à réduire le niveau de divergence de la vitesse prédite.

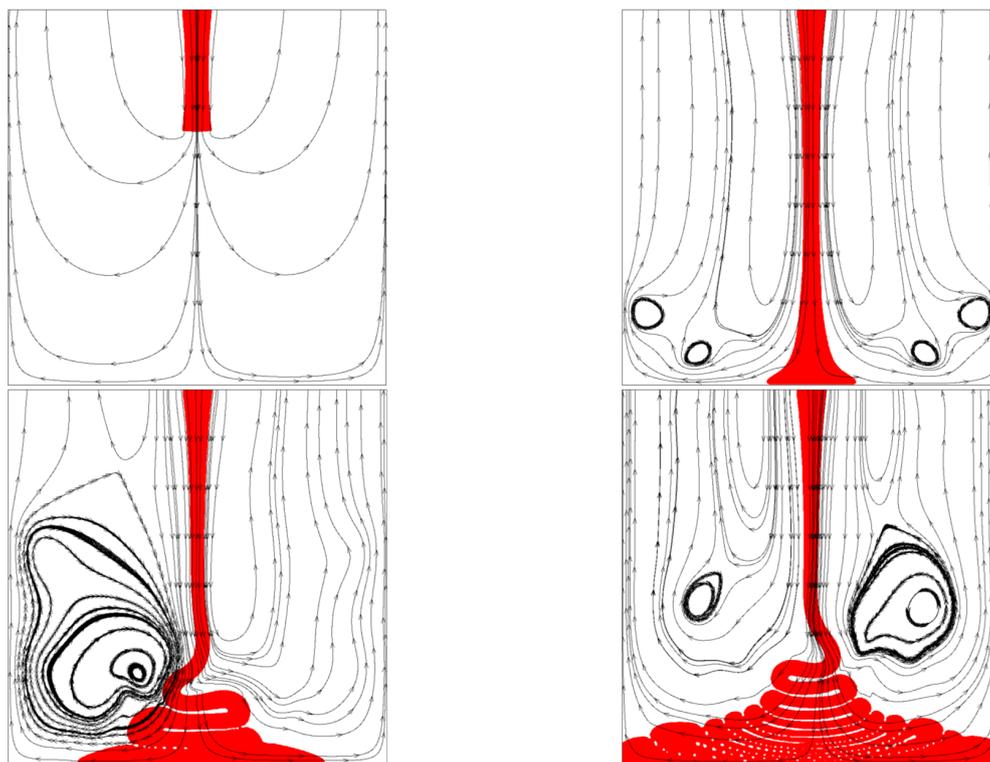


FIGURE 4.32 – Simulation numérique du remplissage d'une cavité par un jet visqueux. Résultats correspondant à la première configuration, simulée par le solveur tout-couplé, sur une grille cartésienne 320×320 . L'interface est représentée aux temps $t = 0, 25$ s et $0, 75$ s (première ligne), $t = 1, 25$ s et 2 s (deuxième ligne). Le cas test correspond à un jet instable pour $Re = 0, 1728$.

4.5 Conclusion

Dans ce chapitre, nous avons proposé un solveur tout-couplé original pour la simulation numérique des équations de Navier-Stokes diphasiques et incompressibles 2D. Ce solveur repose sur une résolution simultanée des inconnues du problème diphasique, à savoir la pression p et le champ de vitesse \bar{u} . Une attention particulière a été portée aux différentes techniques de préconditionnement, lorsque une résolution tout-couplée est mise en place.

Dans le reste du chapitre, nous avons donc présenté différents cas test afin de vérifier l'implémentation du solveur tout-couplé. Ils ont été validés en se référant soit à des solutions analytiques comme dans le cas des tourbillons de Green-Taylor, l'écoulement de Poiseuille, la goutte statique

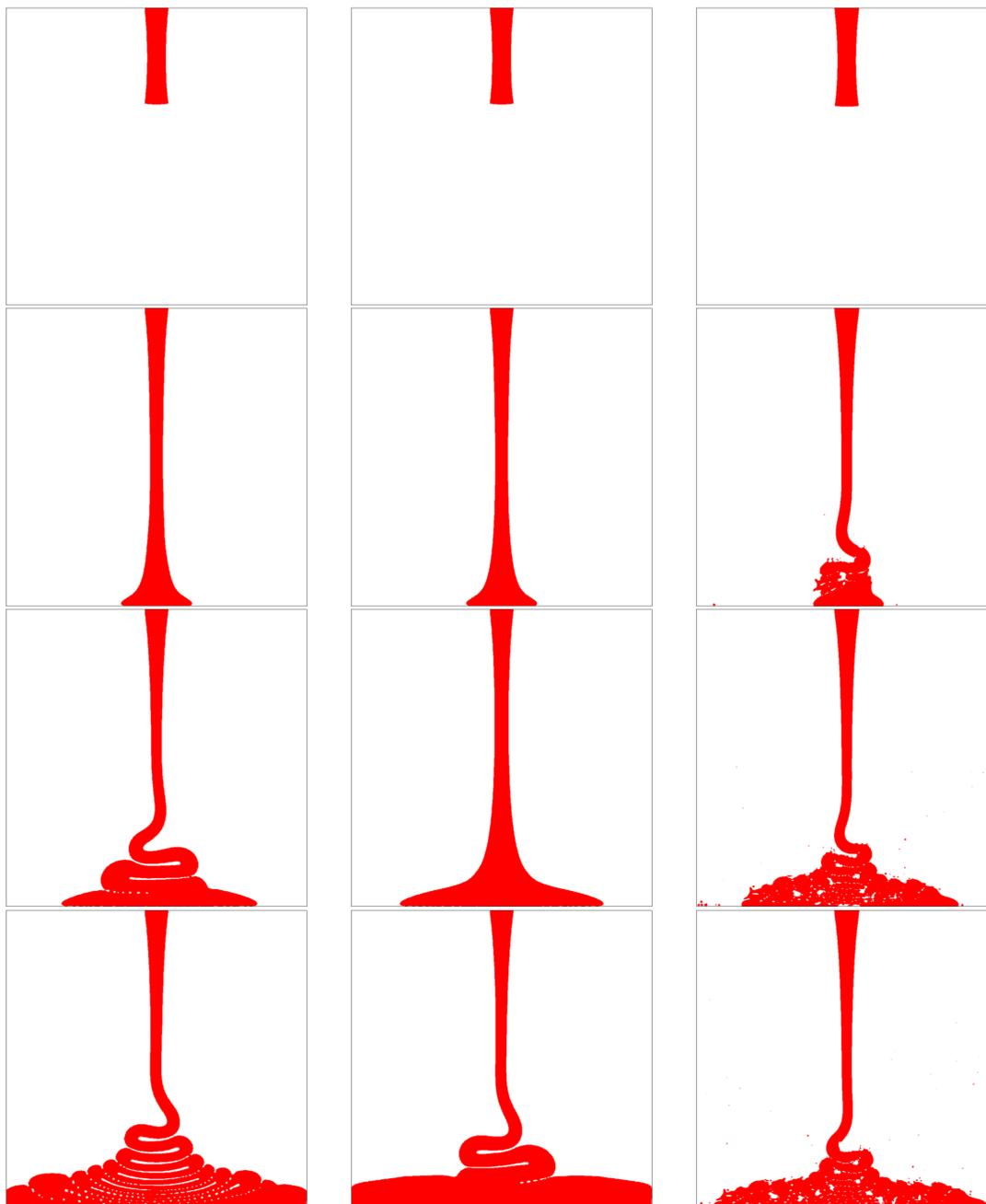


FIGURE 4.33 – Simulation numérique du remplissage d’une cavité par un jet visqueux. Résultats correspondant à la première configuration, simulée par différents solveurs sur une grille cartésienne 320×320 . L’interface est représentée au temps $t = 0.25$ s (première ligne), $t = 0.75$ s (deuxième ligne), $t = 1.25$ s (troisième ligne) et $t = 2$ s (quatrième ligne). Première colonne (solveur tout-couplé), deuxième colonne (méthode du lagrangien augmenté adaptatif) et troisième colonne (la méthode de projection standard).

et la sphère dense, soit à un benchmark numérique tel que le cas de la cavité entraînée ou encore la montée d’une bulle dans une colonne liquide, soit par des expériences comme dans le cas d’instabilité d’un jet visqueux.

Après cette campagne de validation, nous pouvons conclure que le solveur tout-couplé nous a

permis de simuler des problèmes complexes à forts rapports de masses volumiques et de viscosités. Nous avons également pu constater que le solveur tout-couplé et le lagrangien augmenté permettent d'utiliser des pas de temps plus grands que la méthode de projection standard, qui ne parvient pas à converger dans certains cas. De ce fait, une prochaine étape consiste à étendre l'utilisation du solveur tout-couplé à la simulation des problèmes diphasiques 3D sur des architectures massivement parallèles.

Chapitre 5

Fully-coupled parallel solver for the simulation of two-phase incompressible flows

Sommaire

5.1	Models and numerical methods	153
5.1.1	Governing equations of two-phase flows	153
5.1.2	Discretization of mass and momentum equations	153
5.1.3	Block preconditioning techniques	157
5.2	Fully coupled solver parallelization	161
5.2.1	Parallel operations	162
5.2.2	Performance evaluation	162
5.3	Applications to physical cases	164
5.3.1	Phase inversion between two incompressible liquids	164
5.3.2	3D rising bubble	167
5.3.3	Free fall of a dense sphere	171
5.4	Conclusions	172

**Mohamed El Ouafa, Stéphane Vincent,
Vincent Le Chenadec, Benoît Trouette**
MSME, Univ Gustave Eiffel, CNRS UMR 8208,
Univ Paris Est Creteil,
F-77454 Marne-la-Vallée, France

Abstract :

In the framework of the in-house code Fugu, a fully-coupled solver is developed for massively parallel simulations of three-dimensional incompressible multiphase flows. The linearized momentum and continuity equations arising from the implicit solution of the fluid velocities and pressure are solved simultaneously. The method uses a BiCGStab(2) [48] iterative solver with PFMG or SMG preconditioners and an efficient sparse matrix-vector multiplication using the CSR storage format. The construction and the tracking of the interface separating the different involved phases is based on a conservative VOF method [193]. Test cases, such as spherical bubble rising in quiescent liquid and free fall of a dense sphere are carried out to validate the models, especially in the presence of strong density and viscosity ratios between fluid phases. Other cases, like the phase inversion, demonstrate the ability of the new fully-coupled solver to solve two-phase problems with more than 1 billion degree of freedom, with excellent scalability.

Keywords : Fully coupled solver ; linear system preconditionning, two-phase flows, large density and viscosity ratios, HPC

Multiphase flows with separated phases are ubiquitous in nature and in industrial applications. Energy [40, 70], environment [120], chemical engineering [2, 136], hydrology, material processes [111, 184], petroleum engineering [35], vehicle design [187] or civil engineering, are examples of applications where the conditions and two-phase flow regimes encountered spread wide and far in terms of density and viscosity ratios, inertial, viscous, gravity or capillary effects. As the physics of these ubiquitous flows are, however, very complex, it is therefore interesting to be able to characterize their behaviour, in order to control and understand malfunctions such as for boiling in nuclear power plants, overflow in dams or cavitation in turbomachinery, to mention just a few.

The experimental study, although widely developed in many research works, sometimes suffers from certain limitations or implementation difficulties. It is even more difficult to apply to some two-phase problems. Indeed, in addition to the complexity of these flows, i.e. the topology of

the interfaces (separated or dispersed phases), the nature of the interactions (friction, capillary effects, etc.), the flow regimes (in terms of Reynolds or Weber numbers, for example) or the presence of different scales in space and time (from the separated phase to the scale of small interfaces), it is sometimes impossible to simultaneously reproduce the real conditions at the laboratory scale and to measure quantities of interest such as droplet size distributions, interfacial area density or turbulence intensity. Consequently, numerical simulation has become an indispensable tool to better understand and therefore control the bearing of complex two-phase flows. Over the past few decades, this approach has become a central tool for investigation, adopted by the fluid mechanics community to complement experiments.

However, numerical simulation faces several challenges that need to be overcome, not only in terms of the sensitivity of the numerical results to the choice of model and mesh, but also in terms of the robustness and accuracy of the solver and solution algorithms, allowing the simulation of complex two-phase problems with high density or viscosity ratios without causing numerical oscillations or divergence and with acceptable time-to-solution. Certainly, the simulation of these flows at a small scale requires taking into account the turbulence far from the interfaces on the one hand, and the coupling between the fluids present at the interfaces on the other hand (in particular the vorticity generated at these interfaces by the shear and the density or viscosity ratios). Due to the mathematical structure of the Navier-Stokes incompressible equations considered in the recent work (algebraic non-linear differential system), it is difficult to solve them, especially in the presence of high density and viscosity ratios which deteriorate the conditioning of the system and whose spatial distributions vary in space and time. Thus, it is necessary to make numerical solvers and related algorithms run on massively parallel machines, and to optimise them in order to take advantage of intensive computing distributed over thousands of processors.

The problem reveals two aspects: the first is related to the choice of the model and the second to the efficiency of the solver and resolution algorithms on HPC architectures. Concerning the choice of model, there are two approaches to numerical modelling of two-phase flows. The so-called two-fluid models describe multiphase mixing at different levels [51, 99]. In their most complete version, these models take the form of two Navier-Stokes systems, coupled by additional source terms that represent the exchanges (mass, momentum and energy) between the two phases. An important aspect is that both sets of equations hold over the entirety of the domain, which challenges the model in the void regions (where only one phase is present). In contrast, the one-fluid approach consists in modelling the two-phase flow as a single fluid, possessing discontinuous thermodynamic properties as well as source terms representing the capillary effects [172]. This results in a single set of Navier-Stokes equations, supplemented by a scalar advection equation to characterize the interface presence. The one-fluid model is devoted to fully resolved interfaces without any further closure requirements whereas two-fluid models are generally adapted to two-phase dispersed flows laying on a scale separation assumption. In the present work, the one-fluid model will be considered.

In addition, the treatment of velocity-pressure coupling in incompressible flows has been the subject of numerous documented studies and is relatively well controlled as far as single-phase flows are concerned. On the contrary, in the context of two-phase flows with high density and viscosity ratios, numerical difficulties appear, linked to the presence of large interface deformations and associated material fluid property jumps. They lead to ill-conditioned linear systems, which yields a much more challenging problem. At the discrete level, the problem persists in the form of a saddle point system which is costly to solve. Upon discretization and linearisation, it leads to a system of the form

$$\begin{bmatrix} F_u & B_p^T \\ B_u & 0 \end{bmatrix} \begin{pmatrix} \bar{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \bar{f} \\ 0 \end{pmatrix} \quad (5.1)$$

where F_u includes the convective and viscous transport operators, B_p^T the pressure gradient operator, B_u the velocity divergence and \bar{u}^{n+1} and p^{n+1} denote the velocity and the pressure resolved variables at time $(n + 1)\Delta t$, with Δt denoting the time step used for the temporal integration. There are two approaches to solve this large system of equations: either the use of specialized solvers (e.g. iterative with preconditioning), known as exact or coupled method, or the transformation into an approximate problem whose solution is more affordable (Chorin-Temam type projection [36] and its variants [86, 88], known as the approximate or segregated method).

Segregated method do not solve for all of the unknowns at the same time. Instead, it approximates the original system via operator splitting, resulting in two decoupled equations: one to update the velocity field and the other the pressure field. Instead, coupled methods solve both fields (velocity and pressure) simultaneously, thus preserving the consistency of the discretized system with the continuous equations. In coupled methods, the original saddle point system is inverted, thus keeping the velocity-pressure coupling at each time step. This saddle point system can be solved either directly by a fully-coupled solver [22, 54, 55, 126] or with an augmented Lagrangian (AL) method [72, 186, 187]. The interest of these methods is to allow a very accurate and robust resolution, without splitting errors (except the one related to the linearization of the inertial term). This work develops a coupled method, in particular block preconditioners dedicated to the resolution of the saddle point systems from the discretization of the one-fluid model. They rely on recent advances in the field of iterative solvers and preconditioners, and their implementation with open source parallel libraries (HYPRE [68], MUMPS [5, 6] and LIS [109]). The objective is to include the massively parallel computing component in the 3D numerical modeling of a physical problem in order to reduce the execution time of computationally expensive applications.

In the following, the models and the numerical algorithms are described. Then different test cases are presented in order to verify the accuracy and robustness of the developed fully-coupled

solver : 3D rising bubble, shearing of a periodic planar liquid sheet, free fall of a dense sphere and the phase inversion between two incompressible liquids.

5.1 Models and numerical methods

All the developments described and validated in the present work are part of the Fugu code developed by the Heat and Mass Transfer team of MSME laboratory at Gustave Eiffel University.

5.1.1 Governing equations of two-phase flows

In this work, the dynamics of the two immiscible phases is described by the One Fluid Model (OFM) [103]. The incompressible Navier-Stokes equations are solved for an equivalent single fluid with variable material properties, with an additional source term in the momentum equations that introduces the capillary force. An additional advection equation describes the evolution of the phase function C , which is an indicator field transported by the incompressible fluid velocity \bar{u} . One defines $C(t, \bar{x}) = 1$ ($C(t, \bar{x}) = 0$) if \bar{x} belongs to the domain spanned by fluid 1 (fluid 2) at time t . The OFM is flexible in the sense that it can be solved on a fixed mesh, and can be used to simulate various geometries or interface conditions. With mixing rules for the effective density ρ and viscosity μ , the governing equations read

$$\begin{cases} \rho \frac{\partial \bar{u}}{\partial t} + \nabla \cdot (\rho \bar{u} \otimes \bar{u}) + \bar{\mathbf{B}} \cdot (\bar{f}(\bar{u}) - \bar{u}_\infty) = -\bar{\nabla} p + \nabla \cdot \bar{\mathbf{T}} + \rho \bar{g} + \bar{F}_s & (5.2a) \\ \nabla \cdot \bar{u} = 0 & (5.2b) \\ \frac{\partial C}{\partial t} + \bar{u} \cdot \bar{\nabla} C = 0 & (5.2c) \\ \rho(C) = \rho_1 C + (1 - C)\rho_2 & (5.2d) \\ \mu(C) = \mu_1 C + (1 - C)\mu_2 & (5.2e) \end{cases}$$

where $\bar{u} = (u, v, w)^T$ is the fluid velocity, p the pressure field, t the time, ρ and μ are the density and viscosity of the equivalent fluid given by an arithmetic average, with ρ_1 , ρ_2 , μ_1 and μ_2 the densities and the viscosities of fluids 1 and 2, respectively. In addition, the viscous stress tensor for a newtonian fluid is $\bar{\mathbf{T}} = [\mu (\nabla \bar{u} + (\nabla \bar{u})^T)]$, whereas $\bar{F}_s = \sigma \kappa \bar{n} \delta_I$ is the capillary term acting on the interface, modelled in this study by the Continuum Surface tension Force model (CSF) [25]. Finally, $\bar{\mathbf{B}} \cdot (\bar{f}(\bar{u}) - \bar{u}_\infty)$ is a darcy-like penalty term used for specifying domain boundary conditions or immersed boundaries [9]. For example, at bound Γ , the tensor $\bar{\mathbf{B}}$ has diagonal components that tend to infinity while they are identically zero inside the fluid domain Ω . The user-specified function $\bar{f}(\bar{u})$ can vary over space and specify problem dependent boundary conditions (Dirichlet, Neumann, Robin).

5.1.2 Discretization of mass and momentum equations

Without loss of generality, a conservative VOF approach from [193] is used to solve Eq. (5.2c) beforehand rewritten in its conservative form. As the interface tracking technique is not the main

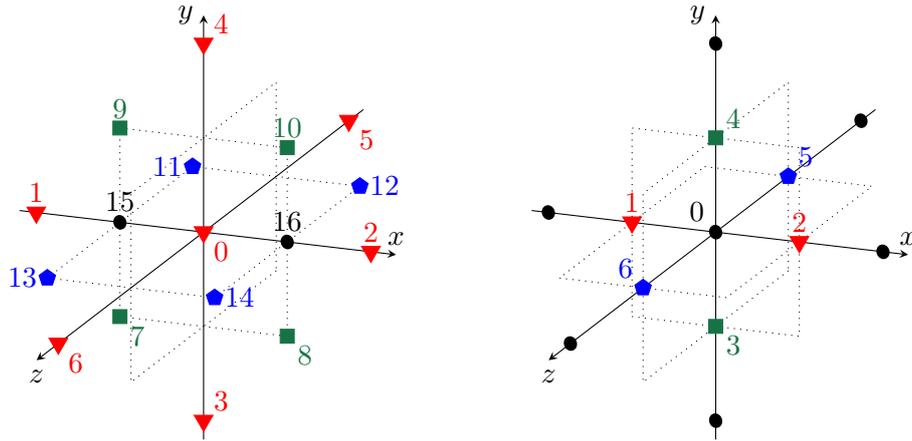


FIGURE 5.1 – Staggered u -velocity (left) and pressure (right) unknowns and their respective local indexes. Velocity control volume is centered on u -velocity (∇ symbols). Local index 0 stands for global indexes i , j and k . Directional full (± 1) or half shifts ($\pm 1/2$) allow to reach neighbours used in the 3 points discretization stencil. The horizontal velocity is linked through divergence and diffusive terms to v - and w - components (\square and \diamond) and also to the pressure (\circ). Similar notations and connectivities are adopted for the treatment of v - and w -components. All the three velocity component then involve 17 coupled variables.

where

$$B_u \bar{u}^{n+1} \approx \int_{\Sigma_p(i,j,k)} \bar{u}^{n+1} \cdot \bar{n} dS \quad (5.5a)$$

$$M_u^{(\rho)} \bar{u}^{n+1} \approx \int_{\Omega_{\bar{u}}(i,j,k)} \left[\frac{\rho \bar{u}^{n+1}}{\Delta t} + \bar{\mathbf{B}} \cdot \bar{\mathbf{f}}(\bar{u}^{n+1}) \right] dV \quad (5.5b)$$

$$N_u^{(\rho)} \bar{u}^{n+1} \approx \int_{\Sigma_{\bar{u}}(i,j,k)} (\rho \bar{u}^{n+1} \otimes \bar{u}^n) \cdot \bar{n} dS \quad (5.5c)$$

$$L_u^{(\mu)} \bar{u}^{n+1} \approx - \int_{\Sigma_{\bar{u}}(i,j,k)} \bar{\mathbf{T}}^{n+1} \cdot \bar{n} dS \quad (5.5d)$$

$$B_p^T \bar{p}^{n+1} \approx \int_{\Sigma_{\bar{u}}(i,j,k)} \bar{p}^{n+1} \bar{n} dS \quad (5.5e)$$

$$\bar{\mathbf{f}} \approx \int_{\Omega_{\bar{u}}(i,j,k)} \left[\frac{\rho}{\Delta t} \bar{u}^n + \bar{\mathbf{B}} \cdot \bar{u}_\infty + \rho \bar{\mathbf{g}} + \bar{\mathbf{F}}_s \right] dV \quad (5.5f)$$

with \bar{n} the outward pointing normal with respect to the control volume $\Sigma_{i,j,k}$. To characterize fluxes across surfaces $\Sigma_{i,j,k}$ of control volume $\Omega_{i,j,k}$, mesh spacing Δx , Δy and Δz are defined, respectively in the x -, y - and z -directions. For the sake of simplicity, constant mesh spacing are presented. Moreover, the integrals are explicitly written for the velocity component u in the x -direction (the extension to the other components is carried out in the same way) and for each control volume. Details are given in appendix B with particular focus on the treatment of the viscous stress tensor. The penalty terms used for handling boundary conditions are discussed in appendix C.

5.1.2.3 Linear system solving

Finite volumes and penalty methods on a staggered mesh, together with an implicit temporal discretization of Eqs. (5.2a) and (5.2b), result in large and non-symmetric linear saddle point systems. The unknowns of the problem, the discrete velocity \bar{u} and pressure p fields, are coupled by the incompressibility constraint $\nabla \cdot \bar{u} = 0$. As presented in the introduction, this saddle point system can be reformulated as follows :

$$\begin{bmatrix} F_u & B_p^T \\ B_u & 0 \end{bmatrix} \begin{pmatrix} \bar{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \bar{f} \\ 0 \end{pmatrix} \quad (5.6)$$

with $F_u = M_u^{(\rho)} + N_u^{(\rho)} + L_u^{(\mu)}$. The reduced system can be rewritten as $A\bar{x} = \bar{b}$ with A the matrix associated to discretization coefficients in time and space, \bar{b} the second member associated with each component of the solution vector and $\bar{x} = (\bar{u}^{n+1}, p^{n+1})^T$ the searched flow solution.

In order to compute \bar{x} , it is necessary to choose a method for solving a linear system. Many techniques exist to solve Eq. (5.6). A wide range of methods, known as direct methods, include Gaussian elimination, factorisation techniques (LU, QR, Cholesky, ...) and multifrontal methods. These techniques are very robust with respect to various problems but typically do not scale well with problem size and are impractical in three-dimensional configurations. Alternative approaches are iterative methods, such as fixed-point (Jacobi, Gauss-Seidel, relaxation, ...), Krylov or multigrid methods. They are however usually effective for specific kind of matrices (symmetric, dense or sparse matrices for example). It is therefore necessary to estimate the structure of the linear system. According to the discretization schemes proposed in the previous paragraph, with the evaluations of fluxes on faces of each control volumes, each equation of the linear system couples only a few components of the solution. With the choice of 3 points stencil per direction, if u -component equation is for example considered, 7 degree of freedom are coupled by the diffusive or inertial terms, 8 more are used for the coupling between u and v as well as between u and w and 2 more are needed for the pressure gradient (see Fig. 5.1). On the whole, 17 degrees of freedom are coupled for each control volume. In addition, 6 additional degrees of freedom are used for the velocity divergence. The same reads for equations v - and w -velocity components. With the example of a mesh composed of $N_x \times N_y \times N_z$ control volumes, and $N = N_x = N_y = N_z$, the solution vector has $4N^3 + 3N^2$ unknowns, the additional $3N^2$ coming from the choice of staggered meshes. The matrix A will be a square sparse matrix of $(4N^3 + 3N^2)^2$ coefficients. Thus, only 17 non-zero diagonals (see Fig. 5.2) acting on each velocity component and 6 non-zero diagonals coming from the discretization of the velocity divergence will be involved in the matrix. In order to save memory space, only the non-zero coefficients arising from the discretization of the incompressible Navier-Stokes equations (Eqs. (5.2a)-(5.2b)) have to be considered and the Compressed Storage Row (CSR) storage format is used.

It is then clear that the matrix generated by the discretization of the Navier-Stokes equations in a one-fluid formulation will be very sparse. There is therefore no interest in using direct methods.

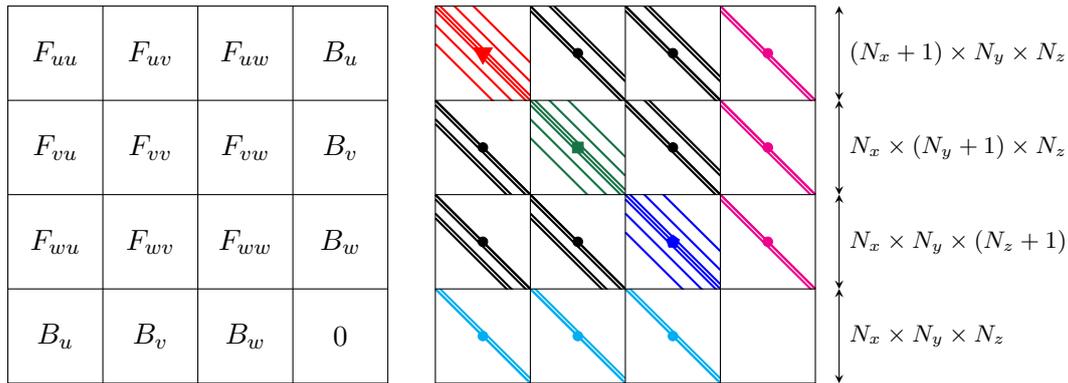


FIGURE 5.2 – Structure of the fully-coupled matrix, 7 diagonal from the 3 directional points stencil (in red for u -, green for v - and blue for w -component, respectively). Black lines for velocity coupling, magenta for pressure gradient and cyan for divergence.

The proposed strategy to solve this sparse linear system is to employ a Krylov solver, here the BiCGStab(2) [48] solver, on the entire system. This involves building an efficient preconditioner $\mathcal{P} \approx A$ to stabilize and accelerate the iterative solver convergence. Indeed, for difficult problems corresponding to ill conditioned matrices, it is essential to combine the BiCGStab(2) algorithm with a suitable preconditioner to avoid possible numerical instabilities. Then, instead of solving $A\bar{x} = \bar{b}$, the left-preconditioned system $\mathcal{P}^{-1}A\bar{x} = \mathcal{P}^{-1}\bar{b}$ is preferred, for which $\text{cond}(\mathcal{P}^{-1}A) < \text{cond}(A)$ is expected. Therefore, in the BiCGStab(2) algorithm, every matrix-vector product $\bar{z} = A\bar{x}$ is followed by the computation of $\mathcal{P}^{-1}\bar{z}$. The same transformation must be applied to the second member $\bar{b} \leftarrow \mathcal{P}^{-1}\bar{b}$. Then, before describing the construction of the preconditioner \mathcal{P} , solving the system $A\bar{x} = \bar{b}$ with a chosen threshold ε and by giving an initial guess $\bar{x}^{(0)}$, is given with the pseudo code given in appendix D.

5.1.3 Block preconditioning techniques

To build the preconditioning matrix \mathcal{P} required in subsection 5.1.2.3, a large variety of preconditioners exist. Among them, ILU type preconditioning, also called incomplete Gauss factorisation, allows to reach low residuals when several millions of unknowns are involved. However, these preconditioners are not easily scalable as they induce a global dependence on the unknowns, and therefore require numerous and repeated exchanges in the domain decomposition framework. It is possible to use ILU techniques and their derived versions in a block implementation, i.e. the preconditioning is performed in a decomposed fashion where the linear operators, decomposed along rows, neglect the contributions from the out-of-core degrees-of-freedom. Nevertheless, in most of the representative cases, the classical preconditioners of the literature such as ILU fail to provide an efficient solution of the fully-coupled Navier-Stokes equations. This was highlighted in a number of documented studies [182], in the context of the augmented Lagrangian method for example with applications to both unsteady laminar and turbulent two-phase flows, but it was limited to 3D cases around 100 million cells. The strategy implemented in this work is to build the preconditioner of the problem by taking a LU block decomposition of the original

matrix A and to introduce a second spatial discretization of the momentum equation in the discrete pressure space. Applied on the linear system (5.6), the LU block factorization gives

$$\begin{bmatrix} F_u & B_p^T \\ B_u & 0 \end{bmatrix} = \begin{bmatrix} I_u & 0 \\ B_u F_u^{-1} & I_p \end{bmatrix} \begin{bmatrix} F_u & B_p^T \\ 0 & S_p \end{bmatrix} \quad (5.7)$$

where $S_p = -B_u F_u^{-1} B_p^T$ is the Schur complement of the pressure block. Thus, if the upper-triangular block U is considered as preconditioner, the preconditioning operator \mathcal{P} directly reads :

$$\mathcal{P} = \begin{bmatrix} F_u & B_p^T \\ 0 & S_p \end{bmatrix} \quad (5.8)$$

According to this choice, the iterative solver would need exactly two iterations to compute the solution. However, it is not feasible to use the exact Schur complement S_p and the velocity block F_u as a part of the preconditioning operator, as they require the knowledge of their inverses, respectively S_p^{-1} and F_u^{-1} , two dense matrices that are time-dependent and impractical to compute. Due to this difficulty, an approximation of the action of the inverse of the velocity block F_u and the Schur complement S_p on any vector has to be considered. The way to achieve this is described in next sections.

5.1.3.1 For the Velocity block

It is important to note here that an appropriate resolution of the Navier-Stokes equations entails the use of an efficient preconditioning of the velocity blocks. This is especially true when an augmented Lagrangian or a fully-coupled solver is used. These velocity blocks are sparse, large and ill-conditioned in the presence of large viscosity ratios. They therefore require resolution by an efficient and robust preconditioning, which must address the challenge raised by the coupling between velocity components due to the large viscosity ratios typically encountered in multiphase flows applications. In this context, this section describes some techniques for the preconditioning of the coupling between velocity components, essentially based on an approximate algebraic preconditioner for the Schur complement.

Let us consider the sparse and non-symmetric matrix F_u , arising from the discretization of the linearized momentum equations (Eqs. (5.5)). This matrix can be split into a 3×3 blocks matrix, rewritten in the following compact form

$$F_u = \begin{bmatrix} A_{uw} & A_{uvw} \\ A_{wvu} & F_{ww} \end{bmatrix} \quad (5.9)$$

with

$$A_{uw} = \begin{bmatrix} F_{uu} & F_{uv} \\ F_{vu} & F_{vv} \end{bmatrix}, \quad A_{uvw} = \begin{bmatrix} F_{uw} \\ F_{vw} \end{bmatrix} \quad \text{and} \quad A_{wvu} = \begin{bmatrix} F_{wu} & F_{wv} \end{bmatrix} \quad (5.10)$$

The preconditioning studied in our case takes advantage of the block LU decomposition of Eq. (5.9), by carefully considering the specific block upper triangular preconditioner given by :

$$\tilde{F}_u = \begin{bmatrix} A_{uw} & A_{uvw} \\ 0 & F_{ww} - A_{wvu}A_{uv}^{-1}A_{uvw} \end{bmatrix} \quad (5.11)$$

where $A_{wvu}A_{uv}^{-1}A_{uvw}$ is the exact Schur complement of F_u , which can be rewritten as below by introducing a new matrix G :

$$\begin{aligned} F_{ww} - A_{wvu}A_{uv}^{-1}A_{uvw} &= F_{ww} \left(I - F_{ww}^{-1}A_{wvu}A_{uv}^{-1}A_{uvw} \right) \\ &= F_{ww} (I - G) \end{aligned} \quad (5.12)$$

As a result, the direct action of the inverse of \tilde{F}_u on a standard vector $(z_u, z_v, z_w)^T$ should be done by a backward elementary substitution. This can be achieved by solving two linear systems and realizing a matrix-vector product. In the corresponding algorithm given in appendix E, approximations of the inverses of $F_{ww}(I - G)$ and A_{uw} are required.

In terms of application of the inverse of the block $F_{ww}(I - G)$ to the vector \bar{z} , it is assumed that the matrices $(I - G)$ and F_{ww} are invertible, and that the inverse $(I - G)^{-1}$ can be developed as an infinite series of expansions of the different terms that are present in the operator $(I - G)$. Using the following Neumann series expansion [122] and the definition of G (Eq. (I.10)), it comes

$$\begin{aligned} (I - G)^{-1} &\approx \sum_{k=0}^{\infty} G^k \\ &\approx \sum_{k=0}^{\infty} \left(F_{ww}^{-1}A_{wvu}A_{uv}^{-1}A_{uvw} \right)^k \end{aligned} \quad (5.13)$$

Thus, the inverse of the block $F_{ww}(I - G)$ writes :

$$[F_{ww}(I - G)]^{-1} \approx \left[\sum_{k=0}^{\infty} (F_{ww}^{-1}A_{wvu}A_{uv}^{-1}A_{uvw})^k \right] F_{ww}^{-1}. \quad (5.14)$$

When a single term (order 0) of the infinite series is kept, \tilde{F}_u from Eq. (5.11) becomes the block Gauss-Seidel preconditioner. In the case where $A_{uvw} = 0$ is assumed, \tilde{F}_u is a block Jacobi preconditioner.

Now in the addressed problem, the matrix from Eq. (I.13) is obtained by truncating for example the expansion at the order 0 which leads to the following :

$$[F_{ww}(I - G)]^{-1} \simeq F_{ww}^{-1} \quad (5.15)$$

When two terms are retained (order 1), a block triangular preconditioner is built, which reads

$$[F_{ww}(I - G)]^{-1} \simeq \left(I + F_{ww}^{-1}A_{wvu}A_{uv}^{-1}A_{uvw} \right) F_{ww}^{-1} \quad (5.16)$$

Now, regardless of the truncation order, the inverse of the block A_{uw} is not known explicitly in Eq. (5.11). So, an approximation of this block has to be found. The latter is done following the same reasoning as above from Eq. (5.11) but considering the A_{uv} block defined in Eq. (5.10)

instead of F_u . This leads, at order 0, to :

$$\tilde{A}_{uv}^{(0)} = \begin{bmatrix} F_{uu} & F_{uv} \\ 0 & F_{vv} \end{bmatrix} \quad (5.17)$$

and at order 1 to

$$\tilde{A}_{uv}^{(1)} = \begin{bmatrix} F_{uu} & F_{uv} \\ 0 & F_{vv} - F_{vu}F_{uu}^{-1}F_{uv} \end{bmatrix} \quad (5.18)$$

Eventually, the form of our velocity preconditioner at orders 0 is :

$$\tilde{F}_u^{(0)} \approx \begin{bmatrix} F_{uu} & F_{uv} & F_{uw} \\ 0 & F_{vv} & F_{vw} \\ 0 & 0 & F_{ww} \end{bmatrix} \quad (5.19)$$

In this study, the Gauss-Seidel block was found to perform slightly better than Jacobi block and triangular block preconditioners. It was therefore preferred. Indeed, 2D comparison of the different preconditioners dedicated to the velocity block, for a two-phase flow problem of the rising of a bubble at different CFL conditions has been conducted. In addition, it appears that no improvement in time-to-solution was obtained when increasing the order of the approximation.

5.1.3.2 For the Schur complement

In the literature, a number of preconditioning techniques for the Schur complements S_p have been proposed. In the context of Stokes flow, ELMAN et al. [58] proposed an approach for homogenous flows. Later, a more sophisticated preconditionner for heterogeneous flows was developed by CAHOUEY et al. [29]. In the context of the incompressible Navier-Stokes equations with constant coefficient, we can find numerous issues to construct a good approximation of the Schur complement S_p . For example, [62] suggested a new technique referred to as LSC (Least Squares Commutator). More recently, KAY et al. [105] have studied a preconditionner coined PCD (Pressure Convection Diffusion). A few studies have also been dedicated to two-phase flows at large density and viscosity ratios. The projection preconditionner, which has been developed by CAI et al. [31] for variable coefficient problems, and adapted by NANGIA et al. [126], is one example. In addition, BOOTLAND et al. [22] have proposed extensions of the PCD and LSC techniques to two-phase flows. To improve performances, the PCD preconditioning with suitable scalability are retained in this work. This preconditioner is given by $S_p^{-1} = \hat{S}_{PCD}$ with

$$\hat{S}_{PCD}^{-1} = (M_p^{(1/\mu)})^{-1} + (A_p^{(1/\rho)})^{-1}(N_p^{(1)} + \Delta t^{-1}M_p^{(1)})(M_p^{(1)})^{-1} \quad (5.20)$$

with the following operators defined for the pressure space discretization

$$\begin{cases} A_p^{(1/\rho)}\phi \approx \int_{\Omega_p} \nabla \cdot (\rho^{-1}\nabla\phi) dV & (5.21a) \end{cases}$$

$$\begin{cases} M_p^{(1/\mu)}\phi \approx \int_{\Omega_p} \frac{1}{\mu} dV & (5.21b) \end{cases}$$

$$\begin{cases} N_p^{(1)}\phi \approx \int_{\Omega_p} \nabla \cdot (\bar{u}\phi) dV & (5.21c) \end{cases}$$

where $M_p^{(1/\mu)}$ is the diagonal pressure mass matrix scaled by the inverse of the viscosity $1/\mu$, $M_p^{(1)}$ is the diagonal standard pressure mass matrix, $N_p^{(1)}$ represents the convective term in the pressure space, and $A_p^{(1/\rho)}$ is the scaled Laplacian that corresponds to the discretization of the term $\nabla \cdot (\rho^{-1}\nabla\phi)$. In order to approximate the inverse of the Schur complement \hat{S}_{PCD}^{-1} , the multigrid solver of the HYPRE library [68] is used, for which the action of $A_p^{(1/\rho)}$ is required. This solver was indeed designed specifically for elliptic equations with variable coefficients. The two diagonal mass matrix are then inverted, namely $(M_p^{(1/\mu)})^{-1}$ and $(M_p^{(1)})^{-1}$, by applying a rescaling to suitable vectors followed by a matrix-vector product for the operator $F_p^{(1)} = N_p^{(1)} + \Delta t^{-1}M_p^{(1)}$. Furthermore, the pressure convection-diffusion preconditioner requires the construction of a Laplace $A_p^{(1/\rho)}$ and a convection-diffusion operator $N_p^{(1)} + \Delta t^{-1}M_p^{(1)}$, together with some choices of boundary conditions (the reader is referred to [63] for more information). In relation to the velocity block, it requires the resolution of three linear systems, involving velocity blocks F_{ww} , F_{vv} and F_{uu} , which are all performed using the aforementioned multigrid solver. These resolutions are completed by two matrix vector products and two updates of the second member. The algorithm applied to block triangular preconditionner is summarized in the algorithm presented in appendix E.

5.2 Fully coupled solver parallelization

The numerical simulation of an unsteady two-phase flow, possibly in turbulent regime, requires very important resources in terms of mesh size, computational time and memory storage. The decomposition domain based parallelization of the different steps, resulting from the equations discretization, is essential to achieve calculations on grids with more than a billion points. Indeed, parallelization has multiple objectives. On the one hand, to obtain reasonable computational times, ideally inversely proportional to the computational resources. And on the other hand, to be able to deal with realistic simulations in terms of dimensionless numbers and physical phenomena solved.

Numerical tests have revealed that about 90% of the total CPU time is dedicated to the linear solver for serial applications. Therefore, the focus for optimization concerns the parallel algorithmic processing of this specific part of the code, by leveraging the combined use of iterative solver BiCGStab(2), block and multigrid preconditioners. The solver requires a large number of matrix-vector multiplication, dot products, linear combinations and several applications of the multigrid preconditioners. These operations are the most important in terms of execution times.

A domain decomposition approach is used to share the spatial data across processes, using a Cartesian decomposition of the global mesh, with MPI exchanges across subdomain boundaries. This parallelization is adapted to massively parallel computers with distributed memory.

5.2.1 Parallel operations

In the BiCGStab(2) algorithm (see appendix D), for each iteration, are performed :

- 4 matrix-vector multiplications

The strategy used involves partitioning the rows of the matrix, as well as the vectors between the processors. Each processor P_i performs locally the product of a block A_i and a vector X_i . The resulting vector $(A \times X)_i$ has the dimension of the number of rows of the matrix A_i . Therefore, no communication is needed in the local computation of the matrix-vector product, instead, this distribution implements short communication messages before the beginning of each calculation.

- 9 dot products.

The 9 dot products of this algorithm represent global values using distributed components of local vectors. Thus, each processor calculates a partial dot product of two corresponding local vectors by multiplying these two vectors and summing the components of the partial result. Then each processor sends its partial result to all the others and receives the products of the other processors that it sums to its own result. Therefore, a global communication is necessary in this scalar product.

- 9 linear combinations.

Each processor contains a part of each global vector included in the operation $X_i \leftarrow X_i + \alpha Y_i$, called local vector. The result vector of the linear combination obtained is also a local vector. Therefore, no communication is necessary in this operation.

- 16 multigrid preconditioner applications, corresponding to 4 multigrid per component (u , v , w and p).

To apply the preconditioner \mathcal{P} which is involved in the Bicgstab(2) algorithm, we will need to solve 16 matrix subsystems per iteration. This is done by using the HYPRE parallel computation library, which offers a large choice of interfaces (structured, semi-structured or algebraic) and multigrid preconditioners (algebraic as BoomerAMG and geometric like PFMG or SMG). In order to take advantage of the higher efficiency of PFMG, this solution strategy has been implemented in the code as it reveals to be the most efficient multigrid solver for our multiphase problems and fully-coupled approach.

5.2.2 Performance evaluation

The classical approach to qualifying a solver in the high-performance computing domain is to check its scalability. It allows to evaluate the performance of a parallel solver when the number of cores is increased. Two metrics, the speed-up S and the efficiency E are defined and two kinds of scalability achieved within two different contexts are distinguished:

- Strong scalability: for a fixed problem size, the number of processes is increased. Ideally, one hopes for linear scalability. In the case of an application that performs a large number of computations, the aim is to find the point at which a reasonable computational time is achieved but still limits the overhead induced by a parallel process.

The speed-up $S(p)$ is defined as the gain of a parallel computation with p processes compared to the same algorithm over a number of reference processes p_{ref} .

$$S(p) = \frac{t_{ref}}{t_p} p_{ref} \quad (5.22)$$

where t_{ref} is the time cost for p_{ref} processes (note that t_{ref} not necessary equals to t_{seq} as the reference might not be the sequential case), and t_p the amount of time to complete the same unit of work with p processes.

- Weak scalability: both number of processes and problem size are increased proportionally, such that the quantity of data to be processed per process holds constant. In the case of ideal weak scalability, a constant execution time of the program should be achieved. This corresponds to a constant efficiency which is therefore independent of the number of processes.

In this case the speed-up is defined by

$$S(p) = \frac{t_{ref}}{t_p} p \quad (5.23)$$

For both cases, the efficiency provides an assessment of the "performance" of the parallel computation. In theory it is less than 1, but for superscalar algorithms the efficiency can exceed this conventional value. The following expression is used to define the efficiency :

$$E(p) = \frac{S(p)}{p} \quad (5.24)$$

In this section, the parallel performance of the in-house code Fugu and specifically the new fully-coupled solver are evaluated. For this purpose, we will consider the strong and weak scalability of the solver as defined above, by checking the evolution of two criteria : the speed-up and the efficiency.

The weak and strong scalability tests are carried out on the simulation of the liquid-liquid phase inversion, on two French supercomputers : SKL Irene, from CEA's Very Large Computing Centre (TGCC) in Bruyères-le-Châtel and for the Jean Zay : HPE SGI 8600 computer from the Institute for Development and Resources in Intensive Scientific Computing (IDRIS-CNRS). The number of processors is increased until 12800. Figure 5.3 presents the speed-up and efficiency for strong and weak approaches. The previous definitions (Eqs. (5.22)-(5.24)) allow a direct comparison of metrics.

Both scalability analysis have been performed over 30 time iterations of the phase inversion problem tackled in Sec. 5.3.1 at fixed residual $\varepsilon = 10^{-4}$ of the iterative solver. Performances have been evaluated using complete computing nodes, i.e. the number of cores/processors is a

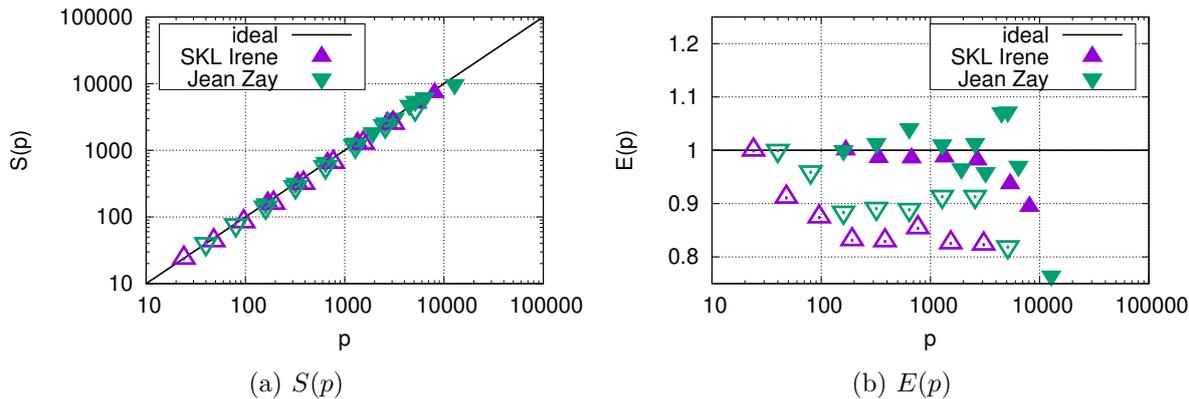


FIGURE 5.3 – Strong (filled symbols) scalability on a 138 million scalar cell mesh and weak scalability (empty ones) on meshes corresponding to $60^3 \times p$ (from 5 millions up to 1.1 billion scalar cells).

multiple of 24 (40) for the SKL Irene (Jean Zay) supercalculator.

First, a good scalability S associated with the fully-coupled solver is obtained (Fig. 5.3a) with a speed-up close to the ideal one. No significant differences are visible between the two supercomputers.

Thereafter, it can be seen that the code is able to exploit the available computing resources on these supercomputers. The parallel efficiency E is 96% for a calculation on 6400 cores with 27^3 cells/core on Jean Zay and 89% for a calculation on 8064 cores with 25^3 cells/core on Irene. In the worst case, the weak scalability study (Fig. 5.3b) shows that the parallel efficiency obtained with 60^3 cells/core loading does not deteriorate that much when increasing the problem size. Indeed, the computational overhead is less than 20% up to a 5120 cores on Jean Zay and 3072 cores on SKL Irene.

5.3 Applications to physical cases

This part is addressed to the validation of the fully-coupled solver, and more specifically the numerical methodologies developed, specifically on two-phase flow test cases. Single phase validations on standard flow problems of the literature have previously been considered in [55] for the 2D fully coupled method. The contribution of the fully-coupled method when tackling complex problems with high density and viscosity ratio will also be studied.

5.3.1 Phase inversion between two incompressible liquids

A two-phase flow, water-oil for instance, where one phase is dispersed in the other, can correspond to a phase inversion, i.e., the dispersed phase becoming the continuous phase. This unsteady phenomenon is then accompanied by several small and large scale interfacial processes, such as deformation, ligament formation, interface break-up and coalescence [66]. In this work, a three-dimensional phase inversion case recently studied by SAEEDIPOUR et al. [156], is investigated. In the following, the geometry and the physical parameters of the problem are

briefly presented before a discussion about the numerical results as well as their comparisons with the reference [156]. An initial cubic blob of light liquid, referred to as fluid 1, is placed at a bottom corner of a cubic box filled with a heavier liquid, called fluid 2. The size of the box is (H, H, H) , while the size of the blob of light fluid is $(H/2, H/2, H/2)$. All box walls are considered as slip walls, so that the normal components as well as normal derivatives of tangential components are zero. Gravitational acceleration g is applied in the vertical direction. The dimensionless numbers governing the dynamics of this problem are therefore the Reynolds number $Re = \rho_1 H U_g / \mu_1$ and the Weber number $We = \rho_1 H U_g^2 / \sigma$. The velocity scale used in previous studies [66] is $U_g = \sqrt{gH/2}(\rho_2 - \rho_1) / \rho_1$. The liquids properties are $\rho_1 = 1000 \text{ kg/m}^3$, $\rho_2 = 900 \text{ kg/m}^3$, $\mu_1 = \mu_2 = 1 \text{ Pa}\cdot\text{s}$, $g = -9.81 \text{ m/s}^2$ and $\sigma = 0.45 \text{ kg/s}^2$. The height $H = 1 \text{ m}$, leads to $Re = 211$ and $We = 121$.

The macroscopic quantities of interest are defined and widely discussed in [66]. In this paper, the focus is on the enstrophy time evolution. Concerning the numerical parameters, the simulations are carried out on four grid ranging from 128^3 to 1024^3 cells. The residual of the iterative BiCGStab(2) solver is initially fixed to $\varepsilon = 10^{-5}$. As far as the time derivatives, a constant time step $\Delta t = 10^{-3} \text{ s}$ is chosen, and a time corresponding to 10 s of the flow motion is crossed. For the smaller mesh 1024^3 , it should be noted that the simulations were performed on 6696 processors of the HPC Occigen cluster of CINES and have taken about 125h of computing time.

Before discussions about the numerical properties of the simulations, it should be noted that all quantities of interest presented in Figures (5.4, 5.5, 5.6, 5.7, 5.8, 5.9, 5.10, 5.11, 5.12, 5.13) (kinetic and potential energy, enstrophy, dissipation, interfacial area and volume of light fluid in the top of the cavity) and defined in [66] well converge, for all times, through the references solution from a mesh size of 256^3 . The exception for convergence is the interfacial area which is very dependent on the computation strategy. Moreover, in line with the VOF used method [193], the volume of light fluid remains constant to almost machine accuracy along the simulation, regardless of the mesh size.

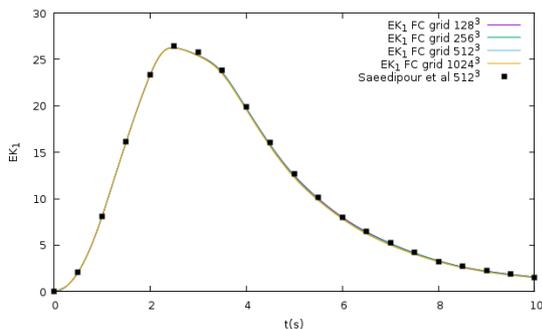


FIGURE 5.4 – Kinetic energy in fluid 1

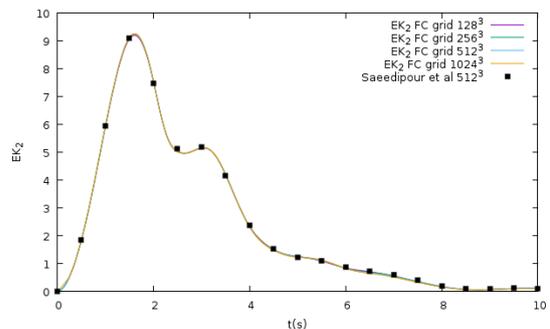


FIGURE 5.5 – Kinetic energy in fluid 2

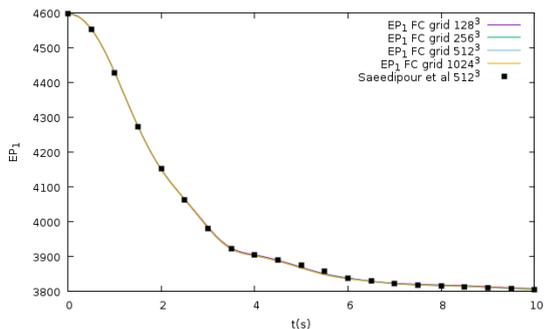


FIGURE 5.6 – Potential energy in fluid 1

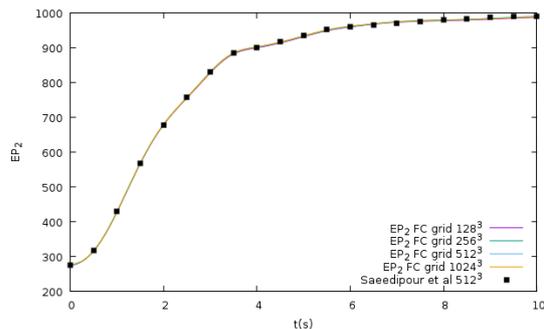


FIGURE 5.7 – Potential energy in fluid 2

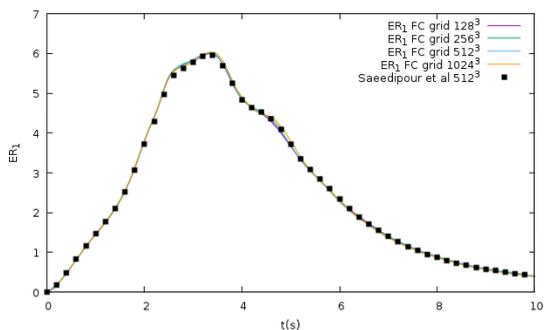


FIGURE 5.8 – Enstrophy in fluid 1

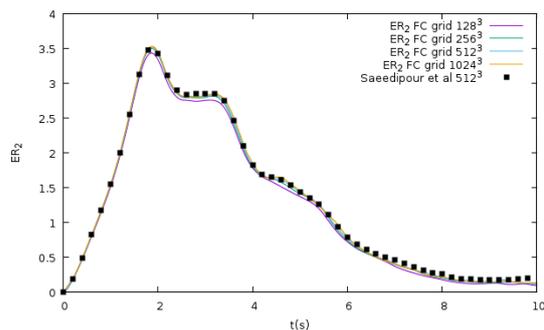


FIGURE 5.9 – Enstrophy in fluid 2

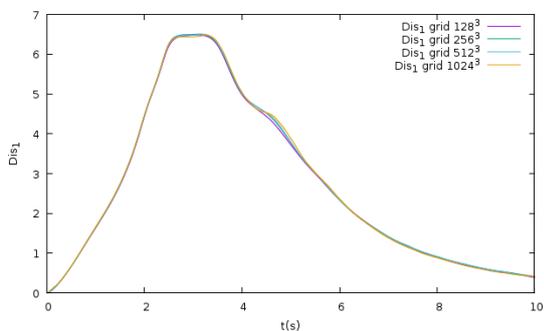


FIGURE 5.10 – Dissipation in fluid 1

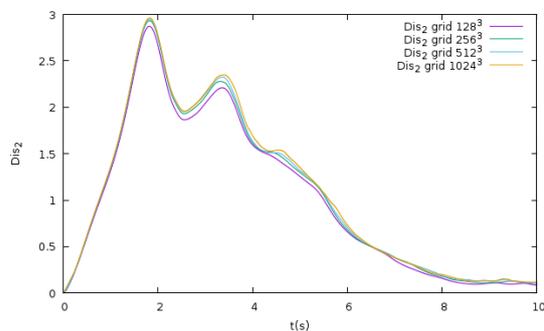


FIGURE 5.11 – Dissipation in fluid 2

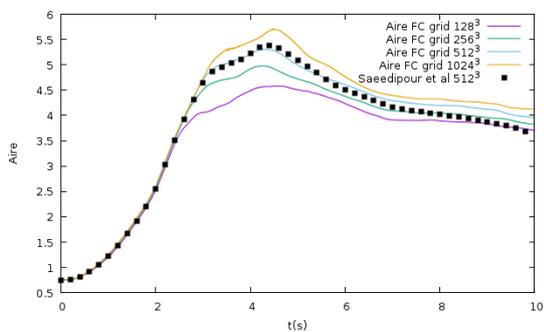


FIGURE 5.12 – Interfacial area

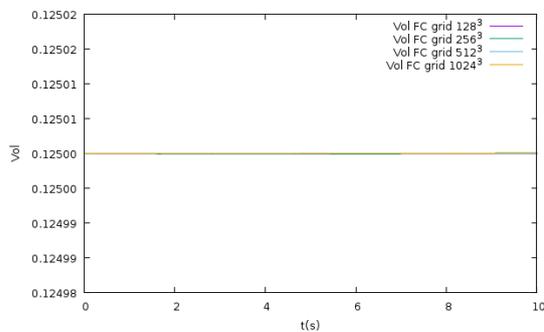


FIGURE 5.13 – Mass conservation

Figure 5.16 presents the time evolution of the enstrophy integrated for both fluids over the

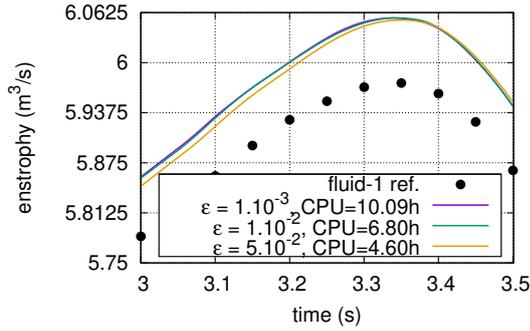


FIGURE 5.14 – the effect of the residual.

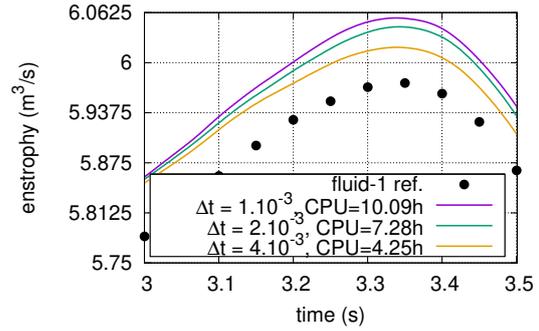


FIGURE 5.15 – the effect of the time step.

FIGURE 5.16 – Enstrophy in fluid 1: study of the effect of the residual and the time step on the quality of the solution.

whole domain. Results are compared with those of [156]. A very relevant parameter to analyze in numerical simulation in general and for our case in particular, is the effect of the residual. For this, three different residuals are considered : $\varepsilon = 10^{-3}$, 10^{-2} and 5×10^{-3} . The results obtained on the enstrophy of the heavy fluid for example, always being compared with the reference data, are shown in the figure 5.14. The mesh used for such simulations is 512^3 , with a reference time step $\Delta t = 10^{-3}$ s. We can show the effect of the residual values on the enstrophy solution near the first peak. The time step is fixed, $\Delta t = 10^{-3}$ s. The initial residual is $\varepsilon_0 = 10^{-3}$. With a 10 times larger value of residual, $\varepsilon = 10\varepsilon_0$, the computational cost is reduced by 30% and by 50% with a residual threshold $\varepsilon = 50\varepsilon_0$. On the plotted interval, the difference between simulations are about 2‰ while a robust 1‰ difference is again observed from the reference solution [156]. The same quantity, the enstrophy of the heavy fluid, is considered, but this time we carry out a comparison based on the CFL condition (or time step). For a grid 512^3 and a residual threshold value $\varepsilon = 10^{-3}$, the results are exported on the figure 5.15. Plotting the enstrophy curves for this second comparison first confirms insensitivity of the calculation to CFL condition, contrary to what we may be expected. This may be justified by the way the equations are solved. Indeed, the use of an implicit scheme via a fully-coupled solver, allows to simulate problems with CFL higher than 1 and to obtain an expected physical meaning of the simulations. With a fixed residual, $\varepsilon = 10^{-3}$, and a reference time step $\Delta t = 10^{-3}$ s, the computational cost is reduced by almost 30% with a time step $\Delta t/2$ and about 60% with $\Delta t/4$. On this range of time steps, the obtained peak intensity varies less than 1%.

5.3.2 3D rising bubble

This first case is based on a numerical benchmark [1] investigating rising bubbles in a liquid column. The studied cases allow to evaluate the robustness of the complete algorithm when the interface is in motion and undergoes relatively large deformations. The authors [1] compare different solutions of two configurations, achieved using numerical codes developed by different teams. Three groups participated in these two test cases : group 1 (IGPM, RWTH Aachen) with the DROPS code Finite element/Level-set, group 2 (INS, University of Bonn) with the code

NaSt3D finite difference/Level set, and group 3 with the code OpenFOAM finite volume/VOF. A qualitative analysis of the characteristic parameters of the rising bubble, such as the time evolution of the centre of mass or the rising velocity of the bubble, allowed the authors to establish reference solutions for chosen parameters. Here are compared the simulation results with the reference solutions from [1]².

A brief description of the problem set-up and characteristics is presented. Initially, a bubble of radius $R_0 = 0.25$ m and centre (0.5 m, 0.5 m, 0.5 m) is placed inside a rectangular tank of dimensions $[0, 1] \times [0, 1] \times [0.2]$ m³. Under the effect of buoyancy force, the bubble rises in the z -direction. The bubble density, ρ_2 , is lower than that of the surrounding liquid (ρ_1). No slip boundary conditions at the walls is assumed, i.e., $\bar{u} = \bar{0}$ on all boundaries. The dimensionless numbers of this flow are the Reynolds number $Re = \rho_1 U_g D / \mu_1$, and the less common Eötvös number $EO = \rho_1 U_g^2 D / \sigma$ yielding the ratio between gravitational forces and surface tension effects. The terms ρ_1 and μ_1 are the density and viscosity of the carrier fluid, D the bubble diameter, $U_g = \sqrt{gD}$ the gravitational velocity and σ the surface tension coefficient. In both tackled configurations, the evolution of the bubble is studied over 3 s. Table 5.1 gives the physical properties and dimensionless parameters associated with the two configurations.

Case	ρ_1 (kg/m ³)	ρ_1/ρ_2	μ_1 (Pa·s)	μ_1/μ_2	g (m/s ²)	σ (N/m ²)	Re	EO
1	1000	10	10	10	0.981	24.5	35	10
2	1000	1000	10	100	0.981	1.96	35	125

TABLE 5.1 – Physical parameters describing the test cases. Here, ρ_2 and μ_2 refers to the fluid density and viscosity respectively inside the bubble whereas ρ_1 and μ_1 are related to the density and viscosity of the surrounding fluid. Also, g is the magnitude of the gravity acceleration and σ is the surface tension coefficient.

For the qualitative analysis of the results, the following characteristics are studied : the barycenter \mathbf{x}_c of the bubble and the rise velocity \bar{u}_c . Note that these quantity definitions can be found in [1]. To validate our solver, we have defined a reference solution on a fine mesh according to the authors [1].

For both test cases, the simulations were carried out on three grids with a mesh size $h = 1/32$, $1/64$ and $1/128$ and time step size $\Delta t = 10^{-3}$ s. As far as the residual of the BiCGSTAB(2), a constant value $\epsilon = 10^{-4}$ is chosen. It has to be noted that the simulation were performed on 144 processors of the HPC cluster Occigen of CINES and required about 4h of computing time. Our reference solution is defined as the one achieved on the finest grid $128 \times 128 \times 256$, i.e. $h = 1/128$ with the fully-coupled solver. It will be further compared with those of the three groups from the benchmark. It should be noted that the reference solution of the first group (with DROPS code) is provided on a regular initial grid containing $4 \times 8 \times 4$ cells, each being subdivided into 6 tetrahedra. This grid is then dynamically and adaptively refined near the interface, leading to a mesh size of $h = 1/32$ within the refinement zone. The time step size $\Delta t = 2.5 \times 10^{-4}$ s in the benchmark simulations. The results of the second group (NaSt3DGPf) is achieved with

². The references data of all quantities of interest are available in the website : <http://wissrech.ins.uni-bonn.de/research/projects/risingbubblebenchmark>.

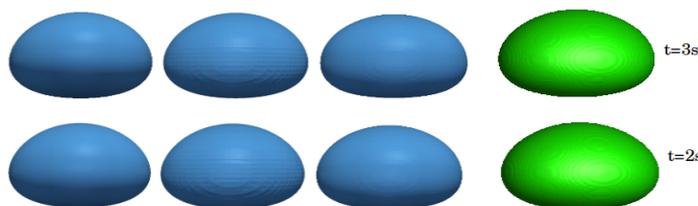


FIGURE 5.17 – Bubble rise for test case 1. The interface $C = 0.5$ is plotted at times $t = 3s$ (first row) and $t = 2s$ (second row). First column holds for DROPS code, second column for NaSt3D, third column for OpenFoam and fourth column to our Fully-coupled solver.

$121 \times 121 \times 241$ grid cells and time step size $\Delta t = 10^{-4}$ s. Lastly, group 3 (OpenFOAM) has constructed a reference solution on a regular grid with $128 \times 128 \times 256$ cells and a time step $\Delta t = 10^{-4}$ s.

Test case 1

The simulations of the rising bubble obtained at instant $t = 2$ s and $t = 3$ s are shown in Fig. 5.17. It can be observed that the overall shape of the bubble is similar with the fully-coupled approach to that of others groups. Indeed, as for the references, the bubble reaches a stable ellipsoidal shape by extending itself in directions perpendicular to the flow.

Time evolution of the rising velocity are shown in figure 5.18a for different grid refinements.

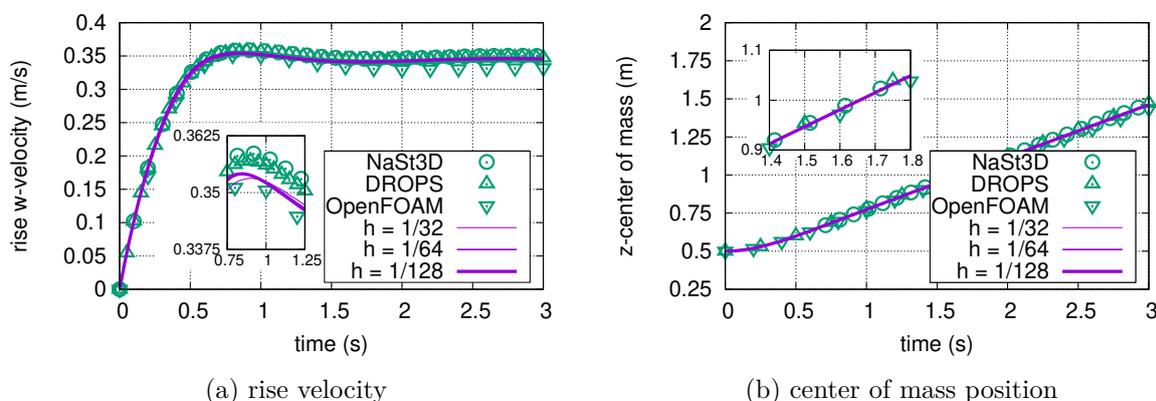


FIGURE 5.18 – Test case 1. **a** Rise velocity w -component as un function of time. **b** Center of mass position. Plain lines stand for the present work while symbols refer to benchmark results [1] from different codes.

Also, the state of the simulation between 0.75 s and 1.25 s is visualized in a zoomed extracted in Fig. 5.18a. It can be concluded that a very nice convergence of the solutions is obtained for the fully-coupled solver from grid $h = 1/32$. Furthermore, the solutions provide results in good agreement with those of the DROPS and NaSt3D codes. Furthermore, in the final part of the simulation, significant differences are noted between the fully-coupled solver and the OpenFOAM code. With respect to the evolution of the position of the centre of mass, we can clearly see on the Fig. 5.18b that the solutions converge from the grid $h = 1/32$, they are in accordance with the references.

Test case 2

This test was also carried out on grids with a mesh size $h = 1/32$, $1/64$ and $1/128$ and time step size $\Delta t = 10^{-3}$ s. As for the reference solutions, the same numerical parameters are used as in the previous case. According to the diagram of CLIFT et al. [38], the bubble in test 2 is in a dimpled ellipsoidal-cap regime where its shape can be strongly deformed with the interface dug in its wake. This is in particular due to the value of the surface tension that is lower than in test case 1. In Figure 5.19, the evolution of the interface of test 2 is reported at $t = 2$ s and $t = 3$ s. The numerical shapes corresponding to $C = 0.5$ iso-surface correlates well with experimental result predicted by the diagram of [38]. As illustrated in this figure, the Fully-coupled solver provides results comparable to the one of the benchmark. In fact, in the resolved regions of the bubble, the overall shape of the bubble is converged as for the other groups in the benchmark. However, the bubble shapes strongly differ at the bottom edge between the codes.

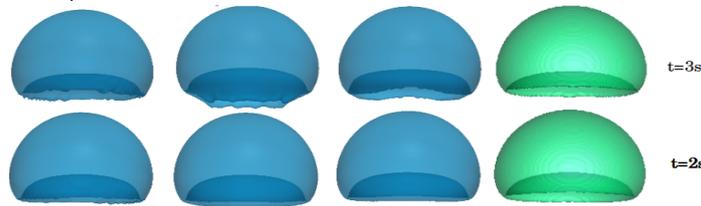


FIGURE 5.19 – Same as Fig. 5.17 for test case 2.

The rising velocity over time is visualized in Figure 5.20a. We notice that all the codes have a velocity peak which is of the order of 0.37 at $t = 0.5$. After this time, the bubble velocity decreases to reach a stabilized state. Also, our simulations are in better agreement with the results of DROPS and OpenFOAM codes than the simulations of NaSt3D code. Finally, Fig. 5.20b shows the evolution of the barycenter of the bubble. Here, the numerical results show good agreement between all codes with less sensitivity of the results regarding the numerical approaches. On a global point of view, the Fugu solver has been favourably compared to the results of benchmark [1], which validates the accuracy and the physical relevancy of the fully-coupled solver.

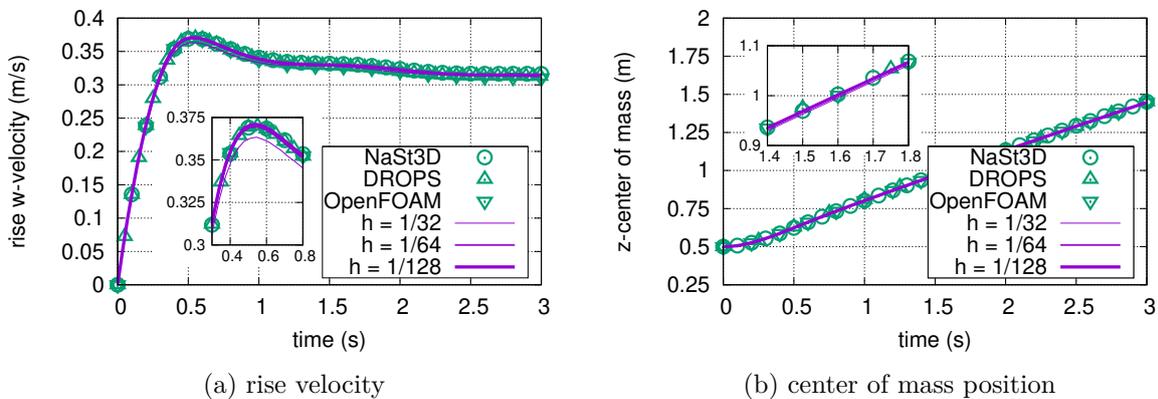


FIGURE 5.20 – Same as Fig. 5.18 for test case 2.

5.3.3 Free fall of a dense sphere

The free fall of a dense sphere in air may seem like a simple process at first sight. However, predicting the behaviour of this two-phase system perfectly is not easy on a numerical point of view. Indeed, the high density and viscosity ratios lead to ill-conditioned linear systems and difficulties in solving issues. Consequently, this test case has a great interest to us for two reasons. On the one hand, it allows checking whether the resolution is robust, when complex unsteady two-phases problems are undertaken, in which density and viscosity ratio may exceed 10^6 . On the other hand, since the exact solution of the falling velocity ($w_c = -gt$ m/s in a void medium) and center of mass ($z_c = -gt^2/2 + z_0$ m) are known, thereafter we can evaluate the accuracy of the fully-coupled solver.

The initial configuration consists of highly viscous liquid droplet of radius $r = 0.0125$ m, density $\rho = \rho_2$ and dynamic viscosity $\mu = \mu_2$ which is released without initial velocity in air. Gravity is set at $g = -9.81$ m/s⁻² in the z -direction, while the surface tension coefficient σ is set to zero. The sphere is centered at $(x_0, y_0, z_0) = (0.05$ m, 0.05 m, 0.15 m) in a parallelepiped cavity full of air whose density and viscosity are $\rho = \rho_1$ and $\mu = \mu_1$ respectively. The cavity is 0.2 m high, 0.1 m long and 0.1 m wide.

Concerning the numerical parameters, the simulations are carried out on a Cartesian grid containing $128 \times 128 \times 256$ cells, with a residual of $\varepsilon = 10^{-4}$ for the BiCGStab(2). As far as the time derivatives, a constant time step $\Delta t = 1 \times 10^{-4}$ s is chosen, and 1500 time steps are computed, corresponding to 0.15 second of the flow motion. It has to be noted that the simulation were performed on 144 processors of the HPC cluster Occigen of CINES and required about 5h of computing time. The physical parameters for the test case are $\rho_1 = 1$ kg/m³, $\mu_1 = 1$ Pa·s, $\rho_2/\rho_1 = \mu_2/\mu_1 = 10^6$ and $g = -9.81$ m/s².

As in the case of the rising bubble, the qualitative analysis of the results is focused on the barycenter \mathbf{x}_c of the bubble and the rise velocity \bar{u}_c . The simulation results of the fully-coupled solver after 1500 time iterations are illustrated in Fig. 5.21. The corresponding colour function at the interface, vertical velocity and vorticity magnitude are presented. It can be seen that the fully coupled solver accurately predicts the fall of the sphere when neglecting the drag force. Unquestionably, the computation shows that the strain rate tensor inside the ball vanishes as $\|\nabla \cdot \bar{\mathbf{T}}\| \approx 0$. Fig. (5.22) reports the evolution over time of the position of the falling velocity and the centre of mass of the sphere, obtained by the fully-coupled solver when the reference grid $128 \times 128 \times 256$ is considered as well as the corresponding analytical solutions. The numerical solution is in good agreement with the reference solution.

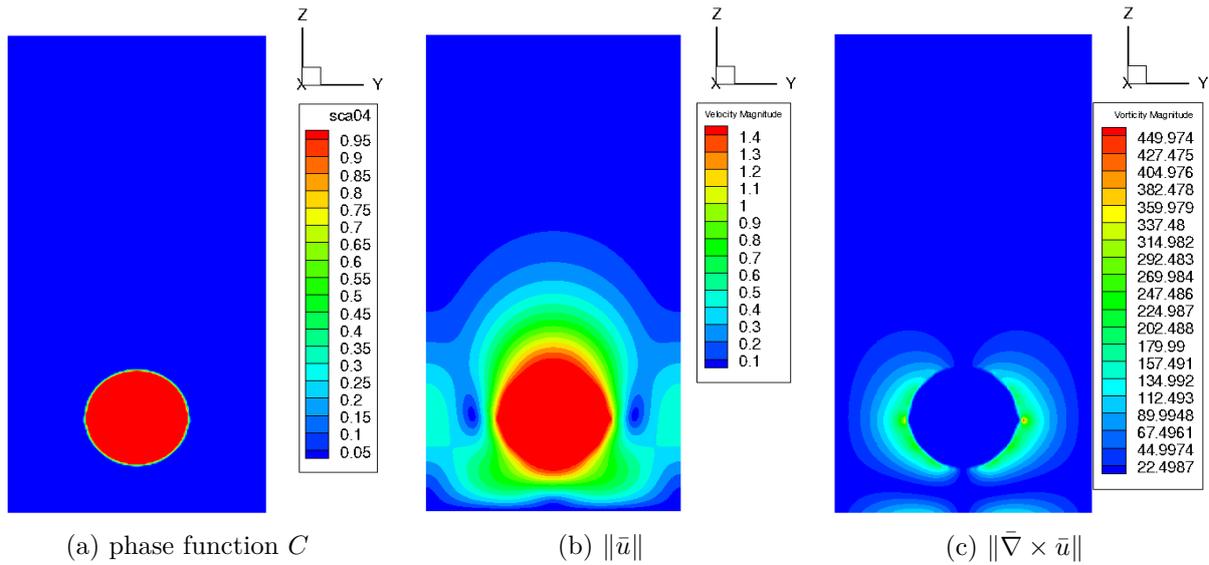


FIGURE 5.21 – Numerical simulation of a free fall of dense sphere at $t = 0.144$ s, obtained on a fine mesh $128 \times 128 \times 256$, by the fully coupled solver. The different fields presented in each row are: on the left, the color function, in the middle, the velocity magnitude and on the right, the vorticity magnitude.

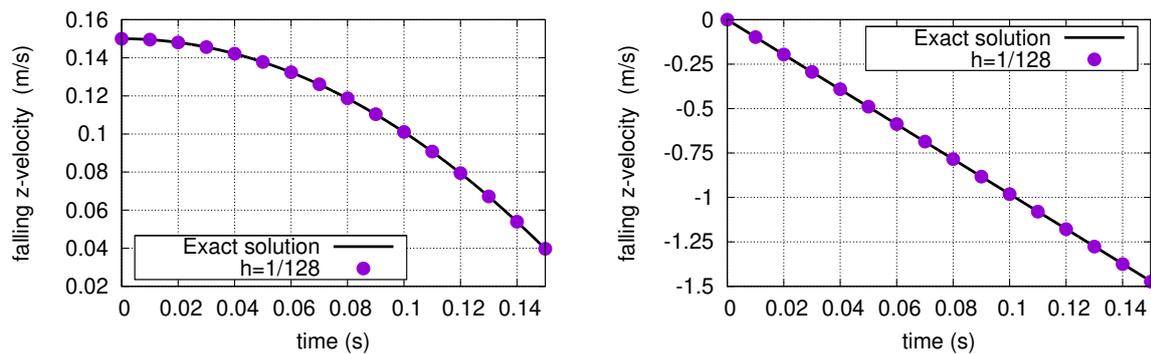


FIGURE 5.22 – Vertical velocity and center of mass for the free fall of a dense and viscous drop ($\rho_1/\rho_2 = \mu_1/\mu_2 = 10^6$) obtained by the fully coupled method with several meshes. left: Vertical velocity and right: center of mass.

5.4 Conclusions

An original fully-coupled solver has been presented for computing three-dimensional unsteady and incompressible two phase flows, in which the velocity-pressure coupling is kept at each time step. The tracking of the interface has been modelled by solving an advection equation of phase function by means of a conservative VOF scheme. The Navier-Stokes equations have been successfully solved using our new fully-coupled solver thanks to the development of new block preconditioning techniques for the velocity block. The latter has been done by means of an infinite series development of the different terms of the velocity block. Further, we demonstrated the ability of the solver to deal with complex cases, for high density and viscosity ratios, such as the case of the free fall of dense sphere or rising bubble in a water column. In addition, we have

shown that fully-coupled solver is able to resolve two-phase problems on more than 1 billions cells, with excellent scalability. In addition, the fully coupled solver is very insensitive to the time step and the residual level of the iterative method, which is a nice feature of the approach, that is not true for time splitting or segregated methods such as the projection technique.

Acknowledgements

We are grateful for access to the computational facilities of the French CINES (National computing center for higher education) and TGCC granted by GENCI under project numbers A0092B06115 and A0112B061165. We thank the technical and administrative teams of these supercomputer centers and agencies for their kind and efficient help.

Chapitre 6

Développement d'un nouveau schéma de transport consistant de la quantité de mouvement et de la masse volumique

Sommaire

6.1	Problématique et État de l'art	176
6.2	Développement d'un schéma AMP (Algebraic Momentum Preserving)	179
6.2.1	Avancement en temps de la solution	181
6.2.2	Discrétisation spatiale de la solution	181
6.2.3	Couplage du schéma AMP au modèle à un fluide	185
6.3	Validations et Résultats	186
6.4	Advection d'une goutte dense dans un fluide parfait	187
6.5	Nappe liquide cisailée périodique	190
6.6	Rupture de barrage	191
6.7	Étude paramétrique de l'inversion de phase liquide/liquide	197
6.7.1	Introduction	197
6.7.2	Géométrie, conditions initiales et paramètres physiques	199
6.7.3	Simulations et résultats	200
6.8	Caractérisation de l'instabilité 3D d'un jet visqueux à faible nombre de Reynolds	208
6.8.1	Résultats	210
6.9	Conclusion	211

Dans le chapitre précédent, nous avons présenté le solveur tout-couplé 3D parallèle, pour lequel les équations de Navier-Stokes diphasiques et incompressibles sont résolues sous forme conservative. Cependant, un cas particulier comme l'advection d'une sphère dense dans un fluide parfait

affecte la stabilité du code, et mène à une perte non contrôlée de l'énergie cinétique. Ces instabilités sont liées à la discrétisation des termes convectifs sur le maillage décalé, et en particulier à l'inconsistance dans le transport de la masse volumique et de la quantité de mouvement.

Dans ce chapitre, nous développons une nouvelle stratégie capable de surmonter ces instabilités numériques. Pour ce faire, nous allons conjuguer le transport de la quantité de mouvement avec le transport de la masse volumique afin d'aboutir à une maîtrise de la consistance, de la stabilité et de la précision du code de calcul.

Nous faisons en premier lieu un état de l'art des différentes méthodes existantes permettant d'assurer la consistance entre le transport de la masse volumique et de la quantité de mouvement. La section 6.2 est consacrée au nouveau schéma de transport consistant que nous avons développé dans le code FUGU, et que nous nommerons schéma AMP (Algebraic Momentum Preserving) ou formulation conservative. Enfin, nous présentons quelques cas de validation et de comparaison afin de montrer les améliorations significatives apportées par le nouveau schéma, notamment en termes de stabilité du code.

6.1 Problématique et État de l'art

Nous nous intéressons ici à la modélisation et à la simulation des écoulements diphasiques incompressibles. L'objectif est de proposer un schéma numérique consistant pour le modèle à un fluide. Tout d'abord, nous rappelons les équations du mouvement directement établies à partir de la loi fondamentale de la dynamique, la seconde loi de Newton qui se présente sous la forme suivante :

$$\rho \bar{\gamma} = \rho \frac{d\bar{u}}{dt} = \sum \bar{F} \quad (6.1)$$

où ρ est la masse volumique du mélange variable et discontinue (au niveau de l'interface), $\bar{\gamma}$ et \bar{u} sont respectivement les composantes d'accélération et de vitesse, t le temps et \bar{F} la résultante des forces qui agissent sur un volume matériel Ω .

Pour un écoulement diphasique, si p représente le champ de pression, \bar{g} l'accélération gravitationnelle, σ la tension de surface, \varkappa la courbure locale à l'interface et C la fonction couleur, le modèle à un fluide dans sa forme standard s'écrit :

$$\frac{dC}{dt} = \frac{\partial C}{\partial t} + \bar{u} \cdot \nabla C = 0 \quad (6.2)$$

$$\rho \left(\frac{\partial \bar{u}}{\partial t} + \bar{u} \cdot \nabla \bar{u} \right) = -\nabla p + \nabla \cdot [\mu(\nabla \bar{u} + \nabla \bar{u}^T)] + \rho \bar{g} + \sigma \kappa \nabla C \quad (6.3)$$

$$\nabla \cdot \bar{u} = 0 \quad (6.4)$$

$$\rho = \rho_1 C + \rho_2 (1 - C) \quad (6.5)$$

$$\mu = \mu_1 C + \mu_2 (1 - C) \quad (6.6)$$

Notons que la combinaison des équations (6.2), (6.4) et (6.5) permet de vérifier l'équation de conservation de masse $\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{u}) = 0$. En effet,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{u}) = \frac{\partial \rho}{\partial t} + \bar{u} \cdot \nabla \rho + \rho \nabla \cdot \bar{u} = (\rho_1 - \rho_2) \left[\frac{\partial C}{\partial t} + \bar{u} \cdot \nabla C \right] = 0. \quad (6.7)$$

La formulation à un fluide présentée ci-dessus est non conservative dans sa forme, car appliquée à des variables non conservées (pression p et vitesse \bar{u}). D'un autre côté, la consistance entre la masse volumique et la quantité de mouvement n'est pas garantie en raison, entre autre, de l'utilisation de maillage décalé, et spécialement la discontinuité de la masse volumique dans les volumes de contrôles coupés par l'interface. Pour des écoulements incompressibles monophasiques ou des écoulements incompressibles dilatables avec une masse volumique constante dans le terme inertiel comme les équations considérées pour la convection naturelle sous l'approximation de Boussinesq, il est équivalent de considérer la formulation non conservative $\rho \left(\frac{\partial \bar{u}}{\partial t} + \bar{u} \cdot \nabla \bar{u} \right)$ ou la forme conservative $\frac{\partial \rho \bar{u}}{\partial t} + \nabla \cdot (\rho \bar{u} \otimes \bar{u})$ du terme inertiel. Pour les écoulements diphasiques avec interfaces et masses volumiques variables, ceci n'est plus valable, même lorsque chaque phase séparée est incompressible, en raison de la présence d'une masse volumique moyenne et variable dans les volumes de fluide locaux coupés par l'interface. Lorsque des forts rapports de masse volumique sont considérés, l'une des questions les plus importantes de la communauté est le problème de consistance entre la masse volumique et la quantité de mouvement. Pour faire une étude rigoureuse sur cette problématique, nous nous proposons d'abord de passer en revue les connaissances déjà acquises sur ce problème et les méthodes qui ont été utilisées.

Au cours des dernières décennies, la communauté diphasique s'est intéressée à la raison fondamentale et à la résolution des instabilités numériques associées aux écoulements diphasiques dont les rapports de masses volumiques sont de l'ordre de 100 à 1000 et plus. Dans le cadre des méthodes VOF, l'une des premières méthodes développées pour surmonter le problème de la consistance entre la masse volumique et la quantité de mouvement a été proposée par Rudman [152]. Dans son approche, Rudman calcule les flux de quantité de mouvement et de masse directement à partir des flux de la VOF. Ceci est réalisé via l'utilisation d'une sous-grille fine pour le transport de l'interface, lorsque un maillage décalé est mis en place. Afin de contourner le coût de l'utilisation d'une sous-grille fine, Sussman et al [28] utilisent un maillage colocalisé (toutes les variables résident au centre des cellules) pour assurer plus facilement la consistance entre la conservation de la masse et de la quantité de mouvement. Des avancées récentes ont été proposées par Le Chenadec et Pitsch [112], et plus tard par Owkes et Desjardins [135] en ce qui concerne les schémas basés sur les VOF-unsplit, qui utilisent des reconstructions géométriques (conservatives) des flux. Par exemple, Le Chenadec et Pitsch emploient une technique intitulée "Lagrangian remap", permettant de construire des flux consistants de masse volumique

et quantité de mouvement, tandis que Owkes et Desjardins advectent la masse volumique sur une grille deux fois plus fine (même principe que Rudman) pour assurer la consistance. D'autres méthodes ont également été abordées par Fuster et al. [75], Arrufat et al [10] et Sagar et al [138]. Dans ces méthodes, les équations de bilan de la quantité de mouvement et de masse volumique sont résolues sur chaque volume de contrôle, en reconstruisant dans chacun d'eux les fractions volumiques associées. Par rapport à la méthode Level set et à toutes les méthodes équivalentes qui reposent sur l'utilisation d'une fonction distance signée, le concept de transport consistant à initialement été proposé par Raessi et Pitsch [147]. Le point clé de leur approche est la définition des flux de masse de façon géométrique à partir de la fonction Level set, mais aussi l'utilisation des mêmes flux de masse pour évaluer les flux de la quantité de mouvement. À ce problème, une solution plus efficace a été établie par Desmons et al [45]. Dans leur méthode HOMP (high-order momentum preserving), une résolution d'une équation d'advection supplémentaire est mise en place pour la prédiction de la masse, de manière cohérente avec le terme inertiel de l'équation de la quantité de mouvement. L'utilisation la plus récente de ces techniques pour la simulation d'écoulements diphasiques à forts rapports de masses volumiques a été menée par Nangia et al [125]. Dans leurs travaux récents, Nangia et ces collaborateurs ont proposé une approche similaire à celle établie par Desmons et al [45] pour évaluer les flux de masse volumique. Une fois connus, les flux sont ensuite introduits dans le calcul du terme convectif. Dans le cadre des méthodes CLS-VOF, Vaudor et al [177] reprennent les différents développements proposés par Rudman sans l'utilisation d'une sub-grille. Plus récemment, une méthode CMOM (Consistent Mass-Momentum transport) a été développée par Zuzio et al [201]. L'idée génératrice de cette méthode est de résoudre sur chaque volume de contrôle de vitesse une équation de transport de moment ainsi qu'une équation supplémentaire de transport de la masse pour obtenir une masse prédite cohérente avec la solution du transport de quantité de mouvement. Dans ce schéma, les flux de masse sont évalués à partir de la fonction de couleur VOF et sont ensuite utilisés pour calculer le terme convectif de la quantité de mouvement. Pour avoir une vue d'ensemble des spécificités des nombreuses méthodes dans la littérature, nous renvoyons le lecteur au tableau 6.1, qui donne une synthèse de ces approches basées sur les méthodes VOF et Level-set.

	Rudman [152]	Bussman et al [28]	Le chenadec et Pitsch [112]	Oweks et Desjardins [135]	Fuster et al [75]
Maillage	2D décalé	3D hexaédrique colocalisé	3D décalé	3D décalé	3D décalé
Advection de l'interface	VOF	VOF	VOF	VOF	VOF
Calcul des flux	Split, flux algébriques, euler	Unsplit, flux géométriques, euler	Unsplit, flux géométriques, semi-lagrangien	Unsplit, flux géométriques, semi-lagrangien	split, flux géométriques, euler
Tension de surface	CSF	Non spécifiée	VOF convolution	Fonctions de hauteur	Fonctions de hauteur
Couplage vitesse-pression	Projection-correction	Projection-correction	Projection-correction	Projection-correction	Projection-correction
	Raessi et Pitsch [147]	Vaudor et al [177]	Nangia et al [126]	Desmons et al [45]	Zuzio et al [202]
Maillage	2D décalé	3D décalé	3D décalé	3D décalé	3D décalé
Advection de l'interface	Level Set	Level Set	Level Set	Level Set	Level Set
Calcul des flux	Level Set, semi-lagrangien	split, flux géométriques, euler	split, flux algébriques, euler	split, flux géométriques, euler	split, flux géométriques, euler
Tension de surface	Ghost Fluid	Ghost Fluid	CSF	CSF	Ghost Fluid
Couplage vitesse-pression	Projection-correction	Projection-correction	Tout-couplé	Projection-correction	Projection-correction

TABLE 6.1 – Synthèse des schémas de transport consistant de la quantité de mouvement et de la masse volumique.

6.2 Développement d'un schéma AMP (Algebraic Momentum Preserving)

Nous nous intéressons maintenant au nouveau schéma basé sur les travaux de Nangia et al [125]. Contrairement à l'approche de Nangia et al portant sur un transport de la masse volumique et une déduction du terme convectif des équations de Navier-Stokes à partir du transport de la masse volumique, nous avons choisi de conjuguer le transport de la masse volumique avec la quantité de mouvement pour maintenir la stabilité du code, notamment lorsque des instabilités interfaciales se développent. L'idée majeure repose sur la reformulation et la discrétisation consistante du terme inertiel dans les équations de bilan de la quantité de mouvement. Dans le

contexte du MUF, le terme inertiel peut être formulé sous une forme conservative comme suit :

si ρ est variable :

$$\begin{aligned} \rho \left(\frac{\partial \bar{u}}{\partial t} + \bar{u} \cdot \nabla \bar{u} \right) &= \frac{\partial (\rho \bar{u})}{\partial t} - \bar{u} \frac{\partial \rho}{\partial t} + \rho \bar{u} \cdot \nabla \bar{u} = \frac{\partial \rho \bar{u}}{\partial t} + \nabla \cdot (\rho \bar{u} \otimes \bar{u}) - \bar{u} \frac{\partial \rho}{\partial t} - \bar{u} \nabla \cdot (\rho \bar{u}) \\ &= \frac{\partial \rho \bar{u}}{\partial t} + \nabla \cdot (\rho \bar{u} \otimes \bar{u}) - \bar{u} \left[\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{u}) \right]. \end{aligned} \quad (6.8)$$

Considérons maintenant l'équation de transport de masse et le bilan de quantité de mouvement (pour plus de clarté, nous supposons qu'il n'y a pas de viscosité, ni de gravité, ni de tension superficielle). $\forall \alpha \in (x, y, z)$, les équations de conservation AMP peuvent être exprimées comme suit :

$$\frac{\partial \rho_\alpha}{\partial t} + \nabla \cdot (\rho_\alpha \bar{u}) = 0 \quad (6.9)$$

$$\frac{\partial \rho_\alpha u_\alpha}{\partial t} + \nabla \cdot (\rho_\alpha u_\alpha \otimes \bar{u}) = -\nabla_\alpha p \quad (6.10)$$

$$\nabla \cdot \bar{u} = 0 \quad (6.11)$$

Les équations de Navier-Stokes (6.9)-(6.10) (sous forme conservative) peuvent être reformulées sous forme de notation d'opérateur comme suit :

$$\frac{\partial \rho_\alpha}{\partial t} + N(\rho_\alpha, \bar{u}) = 0 \quad (6.12)$$

$$\frac{\partial \rho_\alpha \bar{u}_\alpha}{\partial t} + N(\rho_\alpha \bar{u}_\alpha, \bar{u}) = -\nabla_\alpha p \quad (6.13)$$

En faisant un splitting temporel de l'équation (6.13), nous trouvons :

$$\frac{\rho_\alpha^{n+1} - \rho_\alpha^n}{\Delta t} + N(\rho_\alpha^{*,n+1}, \bar{u}^{*,n+1}) = 0 \quad (6.14)$$

$$\frac{\rho_\alpha^{**} \bar{u}_\alpha^{**} - \rho_\alpha^n \bar{u}_\alpha^n}{\Delta t} + N(\rho_\alpha^{*,n+1} \bar{u}_\alpha^{*,n+1}, \bar{u}^{*,n+1}) = 0 \quad (6.15)$$

$$\frac{\rho_\alpha^{n+1} \bar{u}_\alpha^{n+1} - \rho_\alpha^{**} \bar{u}_\alpha^{**}}{\Delta t} = -\nabla_\alpha p \quad (6.16)$$

$$\nabla \cdot \bar{u} = 0 \quad (6.17)$$

où les quantités $(\bar{u}_\alpha^{*,n+1}, \rho_\alpha^{*,n+1})$ sont les extrapolations de la vitesse eulérienne et de la masse volumique à l'instant $n+1$. L'idée précédemment évoquée consiste à résoudre l'équation de continuité auxiliaire (6.14) en parallèle avec l'équation de bilan de quantité de mouvement (6.15). Ceci est réalisé pour chaque composante de vitesse et dans chaque volume de contrôle, d'une

manière qui assure la consistance discrète des flux de masse volumique et de quantité de mouvement. Dans la suite du chapitre, nous contractons la masse volumique ρ_α^{n+1} et la quantité de mouvement $\rho_\alpha^{**}\bar{u}_\alpha^{**}$ dans une seule variable Φ , avec $\Phi = \rho_\alpha$ ou $\Phi = \rho_\alpha\bar{u}_\alpha$. Pour plus de simplicité dans les notations, les équations (6.14)-(6.15) deviennent :

$$\frac{\Phi^{n+1} - \Phi^n}{\Delta t} + N(\Phi, \bar{u}^{*,n+1}) = 0 \quad (6.18)$$

ou encore :

$$\frac{\partial \Phi}{\partial t} + N(\Phi, \bar{u}) = 0 \quad (6.19)$$

6.2.1 Avancement en temps de la solution

Concernant l'intégration temporelle de l'équation (6.19), nous utilisons un schéma d'ordre trois en temps de type Runge-Kutta à trois étapes (SSP-RK3) [82]. Ce schéma d'avancement en temps peut être vu comme une combinaison convexe de pas d'un schéma d'Euler d'ordre 1. Pour une CFL (condition de Courant-Friedrichs-Lewy) où $\Delta t < \min\left(\frac{u}{\Delta x}, \frac{v}{\Delta y}, \frac{w}{\Delta z}\right)$, l'ensemble du schéma d'avancement s'articule donc de la façon suivante :

$$\phi_1 = \phi^n - \Delta t \nabla \cdot (\phi^n \bar{u}_{adv}^n) \quad \text{avec } \bar{u}_{adv}^n = \bar{u}^n \quad (6.20)$$

$$\phi_2 = \frac{3}{4}\phi^n + \frac{1}{4}\phi^1 - \frac{1}{4}\Delta t \nabla \cdot (\phi^1 \bar{u}_{adv}^1) \quad (6.21)$$

$$\phi^{n+1} = \frac{1}{3}\phi^n + \frac{2}{3}\phi^2 - \frac{2}{3}\Delta t \nabla \cdot (\phi^2 \bar{u}_{adv}^2) \quad (6.22)$$

où \bar{u}_{adv}^1 et \bar{u}_{adv}^2 sont les vitesses extrapolées de manière appropriée pour maintenir la précision du schéma. Elle s'écrivent sous la forme :

$$\bar{u}_{adv}^1 = 2\bar{u}^n - \bar{u}^{n-1} \quad (6.23)$$

$$\bar{u}_{adv}^2 = \frac{3}{2}\bar{u}^n - \frac{1}{2}\bar{u}^{n-1} \quad (6.24)$$

6.2.2 Discrétisation spatiale de la solution

Nous venons d'exposer une discrétisation temporelle de l'équation de transport de la fonction Φ écrite sous sa forme conservative (6.20),(6.21),(6.22). IL reste alors à discrétiser la partie convective de l'équation, c'est à dire les variations spatiales de la fonction Φ . Dans le code FUGU, la discrétisation spatiale des équations de Navier-Stokes est basée sur une formulation en volumes finis. La partie convective est donc intégrée sur chaque volume élémentaire pour

donner :

$$\int_{\Omega_{i,j,k}} \nabla \cdot (\Phi \bar{u}_{adv}) dV = \int_{\Sigma_{i,j,k}} \Phi \bar{u}_{adv} \cdot \bar{n} dS. \quad (6.25)$$

Par souci de simplicité de mise en œuvre des schémas numériques, nous décrivons la discrétisation du terme $\nabla \cdot (\Phi \bar{u}_{adv})$, dans laquelle Φ est définie sur les faces x et est advectée par une vitesse \bar{u}_{adv} ,

$$\int_{\Omega_{i,j,k}} \nabla \cdot (\Phi \bar{u}_{adv}) dV \approx [(\phi_e u_e - \phi_w u_w) \Delta y \Delta z + (\phi_n v_n - \phi_s v_s) \Delta x \Delta z + (\phi_f w_f - \phi_b w_b) \Delta x \Delta y] \quad (6.26)$$

les vitesse advectives u_e , u_w , v_n , v_s , w_f , et w_b sur les faces du volume de contrôle sont obtenues en faisant la moyenne des composantes de vitesses adjacentes. Comme le montre la figure 6.1, ces composantes de vitesse se présentent comme suit :

$$\begin{aligned} u_e &= \frac{u_{i+1,j,k} + u_{i,j,k}}{2}, \\ u_w &= \frac{u_{i-1,j,k} + u_{i,j,k}}{2}, \\ v_n &= \frac{v_{i,j+1,k} + v_{i,j,k}}{2}, \\ v_s &= \frac{v_{i,j,k} + v_{i,j-1,k}}{2}, \\ w_f &= \frac{w_{i,j,k+1} + w_{i,j,k}}{2}, \\ w_b &= \frac{w_{i,j,k} + w_{i,j,k-1}}{2}. \end{aligned}$$

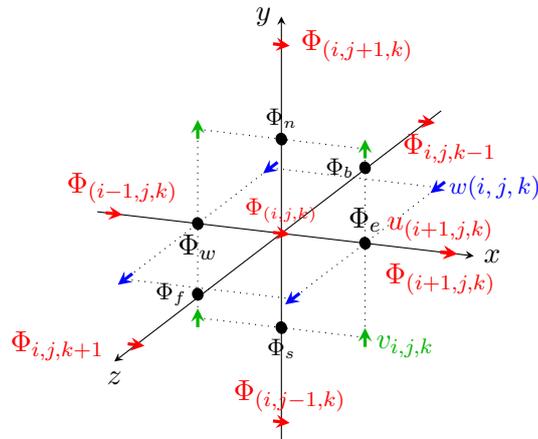


FIGURE 6.1 – Disposition des noeuds autour du volume de contrôle utilisé pour la discrétisation des flux ϕ sur les différentes faces.

Pour prendre en compte les variations spatiales de la fonction Φ et ses valeurs sur les faces du

volume de contrôle ($\phi_e, \phi_w, \phi_n, \phi_s, \phi_f$ et ϕ_w) nous nous orientons naturellement vers l'utilisation des schémas d'ordres élevés. Ce choix est en parfait accord avec l'utilisation d'un schéma d'ordre élevé en temps, afin de générer un schéma stable. C'est à ce niveau que l'application d'un schéma WENO5 [200], CUI [141] ou encore CUBISTA [4] prend toute son importance pour calculer avec une plus grande précision les variations spatiales de Φ . Ces schémas sont consistants et seront donc convergents à condition de respecter la condition CFL.

6.2.2.1 Intégration spatiale : schéma CUI et CUBISTA

Pour trouver une approximation des flux numériques, nous présentons un schéma upwind d'ordre 3 avec un limiteur qui respecte les critères de variation totale décroissante (TVD) et de consistance pour éviter toute oscillation de la solution. Pour une face décalée $f \in e, w, n, s, f, b$, la valeur en amont Φ_C , la valeur décentrée en amont Φ_U et la valeur en aval Φ_D sont repérées en fonction de la direction de la vitesse advective. Par exemple, la Fig.6.2a illustre le cas où $u_e > 0$ pour lequel le volume de contrôle à droite de la face e est identifié comme aval (Φ_D), et les volumes de contrôles situés à gauche de la face e sont attribués à l'amont (Φ_C) et à l'amont décentré Φ_U . De manière analogue, les stencils à trois points sont utilisés pour les autres faces du volume de contrôle. Finalement, l'approximation du flux Φ_f sur une face f peut s'écrire sous forme de variable normalisée comme suit :

$$\text{Pour le schéma CUI : } \quad \tilde{\Phi}_f = \begin{cases} 3\tilde{\Phi}_C & \text{si } 0 < \tilde{\Phi}_C < \frac{2}{13} \\ \frac{5}{6}\tilde{\Phi}_C + \frac{1}{3} & \text{si } \frac{2}{13} < \tilde{\Phi}_C < \frac{4}{5} \\ 1 & \text{si } \frac{4}{5} < \tilde{\Phi}_C < 1 \\ \tilde{\Phi}_C & \text{sinon} \end{cases} \quad (6.27)$$

$$\text{Pour le schéma CUBISTA : } \quad \tilde{\Phi}_f = \begin{cases} \frac{7}{4}\tilde{\Phi}_C & \text{si } 0 < \tilde{\Phi}_C < \frac{3}{8} \\ \frac{3}{4}\tilde{\Phi}_C + \frac{3}{8} & \text{si } \frac{3}{8} < \tilde{\Phi}_C < \frac{3}{4} \\ \frac{1}{4}\tilde{\Phi}_C + \frac{3}{4} & \text{si } \frac{3}{4} < \tilde{\Phi}_C < 1 \\ \tilde{\Phi}_C & \text{sinon} \end{cases} \quad (6.28)$$

où la valeur normalisée $\tilde{\Phi}_f$ est définie par

$$\tilde{\Phi}_f = \frac{\Phi - \Phi_U}{\Phi_D - \Phi_U}. \quad (6.29)$$

Une procédure similaire peut être utilisée pour évaluer une discrétisation des quantités advectées sur les volumes de contrôles de la variable v et w .

6.2.2.2 Intégration spatiale : schéma Weno5

Afin de compléter la discrétisation spatiale du terme convectif, nous introduisons un nouveau schéma de 5ème ordre de type Weno. Basé sur une formulation conservative, ce schéma s'appuie sur l'utilisation d'un stencil de 5 points pour évaluer au mieux les flux. Par exemple pour obtenir une meilleure approximation du flux $f \in e, w, n, s, f, b$ à la face f , le schéma WENO5 utilise la



FIGURE 6.2 – Stencil du schéma CUI pour évaluer la quantité ϕ sur la face est du volume de contrôle.



FIGURE 6.3 – Stencil du schéma CUI pour évaluer la quantité ϕ sur la face nord du volume de contrôle.

valeur en amont Φ_C , les valeurs décentrées en amont Φ_U et Φ_{UU} , et les valeurs décentrées en aval Φ_D et Φ_{DD} . Les figures 6.4a et 6.4b illustrent les cas où $u_f > 0$ et $u_f < 0$ pour lesquels les flux sur la face f sont discrétisés. Etant donné ce stencil, nous construisons ainsi une discrétisation du flux Φ_f comme suit :

$$\Phi_f = \frac{\alpha_1 q_1 + \alpha_2 q_2 + \alpha_3 q_3}{\alpha_1 + \alpha_2 + \alpha_3} \quad (6.30)$$

où les termes q_1 , q_2 et q_3 sont les suivants,

$$q_1 = \frac{1}{3}\Phi_{UU} + \frac{1}{3}\Phi_U + \frac{1}{3}\Phi_C$$

$$q_2 = \frac{1}{6}\Phi_U + \frac{5}{6}\Phi_C + \frac{1}{3}\Phi_U$$

$$q_3 = \frac{1}{3}\Phi_C + \frac{5}{6}\Phi_D + \frac{1}{6}\Phi_{DD}$$

avec

$$\alpha_1 = \frac{\frac{1}{10}}{(IS_1 + \epsilon_{WENO})^2} \quad (6.31)$$

$$\alpha_2 = \frac{\frac{6}{10}}{(IS_2 + \epsilon_{WENO})^2} \quad (6.32)$$

$$\alpha_3 = \frac{\frac{3}{10}}{(IS_3 + \epsilon_{WENO})^2} \quad (6.33)$$

où ϵ_{WENO} assure que le dénominateur soit non nul et les termes IS, dénotant l'indicateur de régularité, sont définis par :

$$IS_1 = \frac{13}{12} (\Phi_{UU} - 2\Phi_U + \Phi_C)^2 + \frac{3}{12} (\Phi_{UU} - 4\Phi_U + 3\Phi_C)^2 + \epsilon_{WENO} \quad (6.34)$$

$$IS_2 = \frac{13}{12} (\Phi_U - 2\Phi_C + \Phi_U)^2 + \frac{3}{12} (\Phi_D - \Phi_U)^2 + \epsilon_{WENO} \quad (6.35)$$

$$IS_3 = \frac{13}{12} (\Phi_C - 2\Phi_D + \Phi_{DD})^2 + \frac{3}{12} (3\Phi_C - 4\Phi_D + \Phi_{DD})^2 + \epsilon_{WENO} \quad (6.36)$$

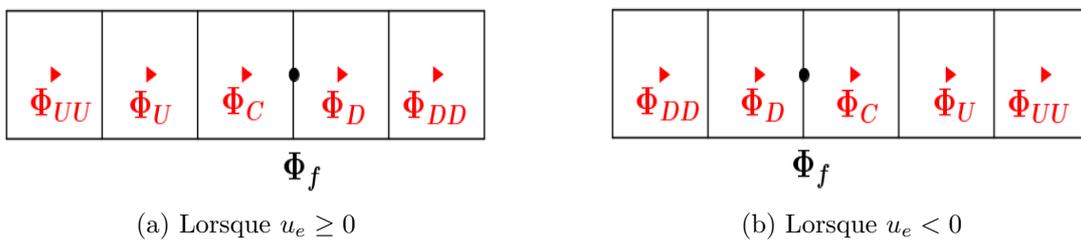


FIGURE 6.4 – Stencil du schéma WENO pour évaluer la quantité ϕ sur la face est du volume de contrôle.

6.2.3 Couplage du schéma AMP au modèle à un fluide

Le point de départ de notre réflexion sur la stabilisation du code repose sur la conjugaison du transport de la masse volumique et la quantité de mouvement de façon consistante. Une fois que les équations de continuité et de quantité de mouvement sont résolues, des valeurs "intermédiaires" de la quantité de mouvement et une mise à jour de la masse volumique sont obtenues. Enfin, le couplage du nouveau schéma AMP avec le MUF se résume ainsi :

- **Étape 0** : Sur chaque volume de contrôle (Voir Fig.6.5), nous calculons C_α^n par interpolation géométrique de C centrée
- **Étape 1** : Sur chaque volume de contrôle, nous calculons la masse volumique ρ_α^n à partir de la fonction couleur C_α^n

$$\rho_\alpha^n = C_\alpha^m \rho_{\alpha 2} + (1 - C_\alpha^m) \rho_{\alpha 1} \quad (6.37)$$

- **Étape 2** : Sur chaque volume de contrôle, nous calculons le moment $\rho_\alpha^n \bar{u}_\alpha^n$ à partir de la masse volumique ρ_α^n et la vitesse Eulerienne \bar{u}_α^n .
- **Étape 3** : Nous mettons ensuite à jour la masse volumique ρ_α^{n+1} et le moment intermédiaire $\rho_\alpha^{**} \bar{u}_\alpha^{**}$ de manière à assurer la consistance discrète des flux de masse et de quantité de

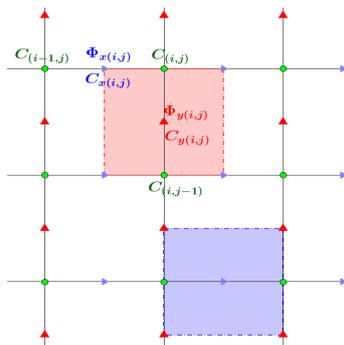


FIGURE 6.5 – Disposition des noeuds autour du volume de contrôle utilisé pour la construction de la couleur C_α^n .

mouvement. Dans ce qui suit, pour alléger l'écriture le symbole Φ désigne à la fois la masse volumique ρ_α^{n+1} et le moment intermédiaire $\rho_\alpha^{**}\bar{u}_\alpha^{**}$.

$$\phi^1 = \phi^n - \Delta t \nabla \cdot (\phi^n \bar{u}_{adv}^n) \quad \text{avec } \bar{u}_{adv}^n = \bar{u}^n \quad (6.38)$$

$$\phi^2 = \frac{3}{4}\phi^n + \frac{1}{4}\phi^1 - \frac{1}{4}\Delta t \nabla \cdot (\phi^1 \bar{u}_{adv}^1) \quad (6.39)$$

$$\phi^{n+1} = \frac{1}{3}\phi^n + \frac{2}{3}\phi^2 - \frac{2}{3}\Delta t \nabla \cdot (\phi^2 \bar{u}_{adv}^2) \quad (6.40)$$

avec :

$$\bar{u}_{adv}^1 = 2\bar{u}^n - \bar{u}^{n-1} \quad (6.41)$$

$$\bar{u}_{adv}^2 = \frac{3}{2}\bar{u}^n - \frac{1}{2}\bar{u}^{n-1} \quad (6.42)$$

Étape 4 : Nous résolvons les équations de Navier-Stokes incompressibles afin d'obtenir les champs de vitesse \bar{u}^{n+1} et de pression p^{n+1} . Ceci est réalisé en utilisant la masse volumique ρ^{n+1} et le moment intermédiaire $\rho^{**}\bar{u}^{**}$.

$$\frac{\rho_\alpha^{n+1}\bar{u}_\alpha^{n+1} - \rho_\alpha^{**}\bar{u}_\alpha^{**}}{\Delta t} = -\nabla_\alpha p \quad (6.43)$$

$$\nabla \cdot \bar{u} = 0 \quad (6.44)$$

6.3 Validations et Résultats

Dans cette section, nous présentons différents cas-tests prouvant l'importance de la mise en place d'un schéma consistant dans la stabilité et la précision du code pour des problèmes diphasiques, en particulier pour des cas raides à forts rapports de masse volumique de l'ordre 10^3 à 10^6 . À ce

point, nous considérons deux formulations du modèle MUF, une conservative (schéma AMP) et l'autre non conservative (schéma centré). Nous comparons également les résultats issus de chaque formulation à des solutions analytiques ou à des problèmes de référence tirés de la littérature.

6.4 Advection d'une goutte dense dans un fluide parfait

Dans un premier temps, nous allons nous intéresser à une configuration académique dans laquelle on considère le déplacement d'une goutte à masse volumique élevée [42, 112, 126, 202]. L'objectif principal sera de vérifier la bonne implémentation numérique du nouveau schéma AMP, en comparant les résultats obtenus par la simulation numérique avec ceux analytiques. Cette comparaison permettra non seulement de valider nos simulations numériques mais aussi de tester les propriétés du nouveau schéma lorsque des écoulements diphasiques à forts rapports de masses volumiques sont considérés.

L'interface est initialement une goutte de rayon $R = 0.2 m$, centrée dans une boîte carrée de côté $L = 1 m$. Au début du calcul, la goutte possède une vitesse initiale $U = 1 m s^{-1}$ dans un gaz au repos. Les masses volumiques de la goutte et du gaz sont : $\rho_{goutte} = 10^6 kg m^{-3}$, $\rho_{gaz} = 1 kg m^{-3}$. Les viscosités de la goutte et du gaz, la gravité ainsi que la tension de surface sont considérées nulles. En l'absence de ces effets, l'énergie cinétique devrait se conserver, et le phénomène devrait être purement périodique. Pour cette raison, nous avons choisi d'utiliser des conditions aux limites périodiques.

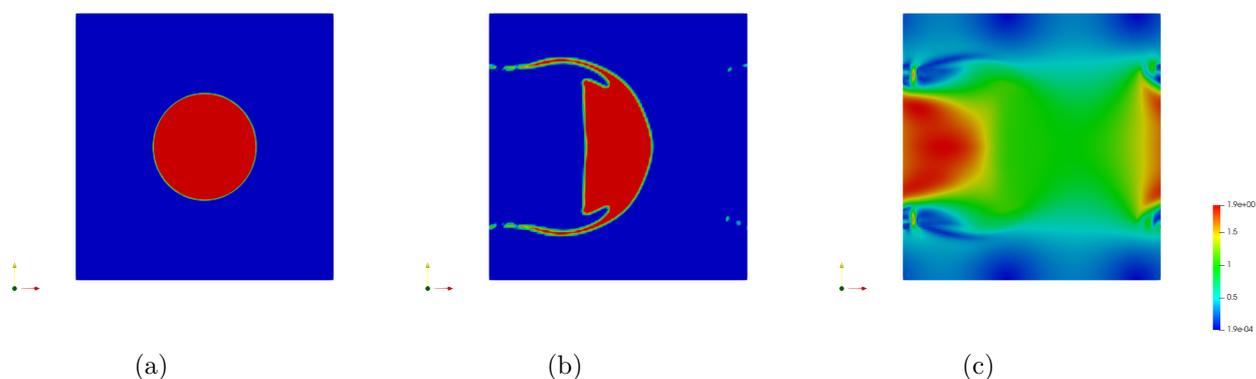


FIGURE 6.6 – Simulation numérique de l'advection d'une goutte dense dans un fluide parfait. Maillage 64^2 : Forme de la goutte après une période avec la formulation non conservative. (a) : forme initiale de la goutte à $t = 0 s$, (b) : forme finale de la goutte à $t = 1 s$ et (c) : module de la vitesse à $t = 1 s$

Dans un premier temps, nous démontrons l'importance d'un schéma de transport consistant de la masse volumique et de la quantité de mouvement pour des écoulements diphasiques à fort rapport de masse volumique. Les figures (6.6-6.7) montrent la forme de la goutte sur deux maillages différents 64^2 et 128^2 , lorsque les simulations numériques sont effectuées avec un schéma inconsistant, c'est-à-dire en utilisant une formulation non conservative pour le modèle MUF.

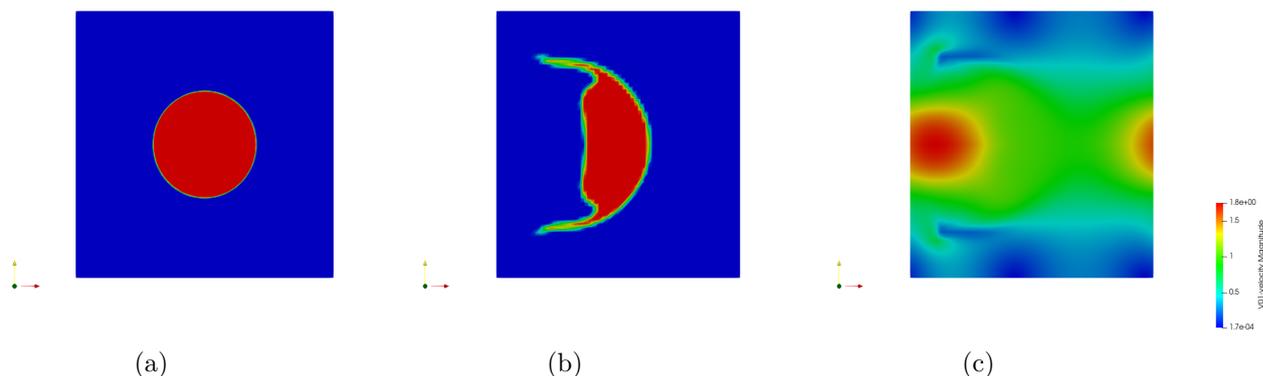


FIGURE 6.7 – Simulation numérique de l’advection d’une goutte dense dans un fluide parfait. Maillage 128^2 : Forme de la goutte après une période avec la formulation non conservative. (a) : forme initiale de la goutte à $t = 0\text{ s}$, (b) : forme finale de la goutte à $t = 1\text{ s}$ et (c) : module de la vitesse à $t = 1\text{ s}$

Après un tour complet au temps $t = 1\text{ s}$, la goutte est fortement déformée et aucune convergence en maillage n’est observée, ce qui montre bien que la formulation non conservative n’est pas robuste pour une configuration à fort rapport de masse volumique. Dans ces conditions, nous constatons une perte de l’ordre de 8% de l’énergie cinétique due aux erreurs sur la quantité de mouvement à l’interface.

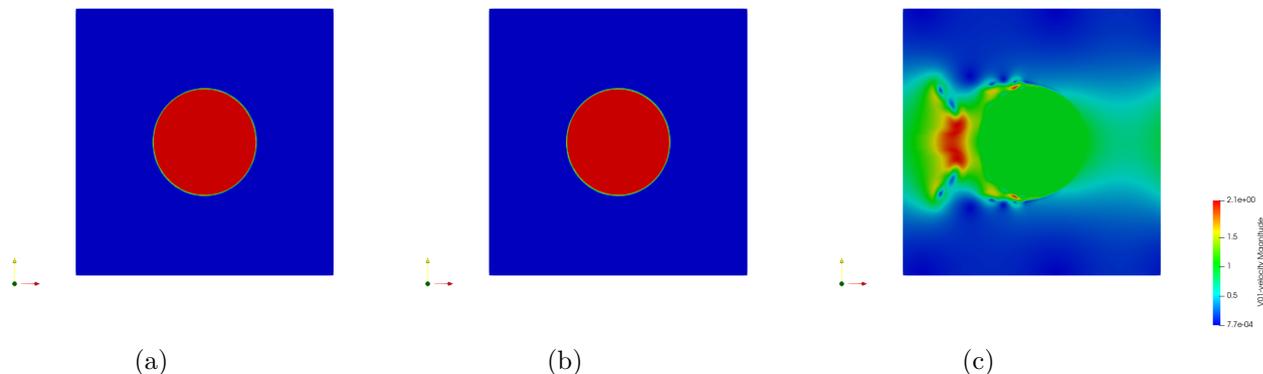


FIGURE 6.8 – Simulation numérique de l’advection d’une goutte dense dans un fluide parfait. Maillage 256^2 : Forme de la goutte après une période avec la formulation conservative pour un schéma de transport de type WENO. (a) : forme initiale de la goutte à $t = 0\text{ s}$, (b) : forme finale de la goutte à $t = 1\text{ s}$ et (c) : module de la vitesse à $t = 1\text{ s}$

Sur les figures (6.8-6.9-6.10) nous présentons les résultats obtenus avec le nouveau schéma de transport (AMP). Ces figures montrent clairement les améliorations apportées par le nouveau schéma. En effet, la sphère reste circulaire et nous n’apercevons aucune différence entre sa forme initiale et sa forme finale. Quant à l’évolution de l’énergie cinétique, le nouveau schéma conserve bien cette grandeur à une précision de l’ordre de 10^{-7} contrairement à la formulation non conservative qui introduit une dissipation non négligeable (voir Figs.6.12 et 6.11)

Pour finir avec le cas de la sphère dense en 2D, le tableau 6.2 présente les erreurs géométriques

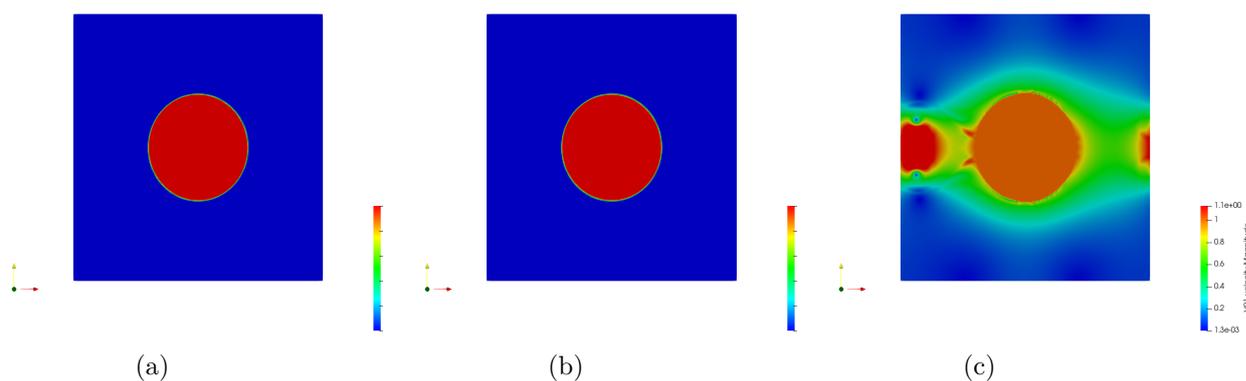


FIGURE 6.9 – Simulation numérique de l’advection d’une goutte dense dans un fluide parfait. Maillage 256^2 : Forme de la goutte après une période avec la formulation conservative pour un schéma de transport de type CUI. (a) : forme initiale de la goutte à $t = 0$ s, (b) : forme finale de la goutte à $t = 1$ s et (c) : module de la vitesse à $t = 1$ s

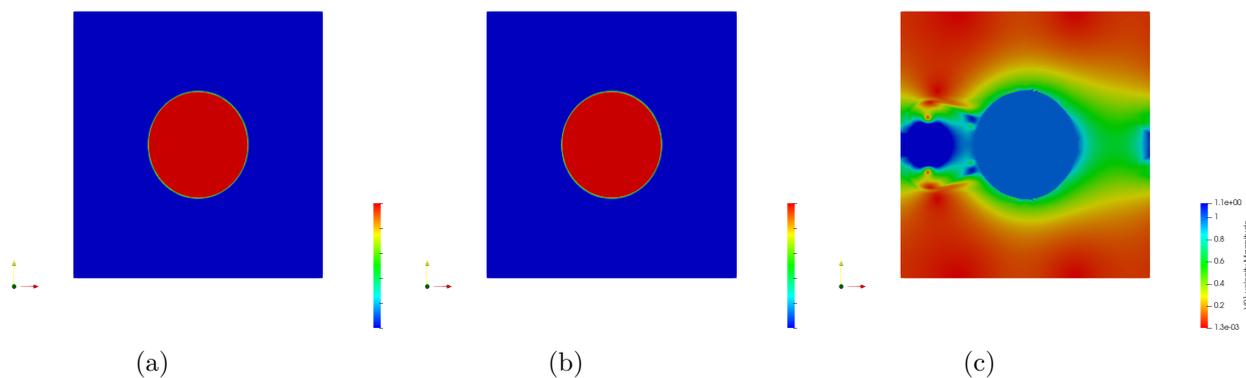


FIGURE 6.10 – Simulation numérique de l’advection d’une goutte dense dans un fluide parfait. Maillage 256^2 : Forme de la goutte après une période avec la formulation conservative pour un schéma de transport de type Cubista. (a) : forme initiale de la goutte à $t = 0$ s, (b) : forme finale de la goutte à $t = 1$ s et (c) : module de la vitesse à $t = 1$ s

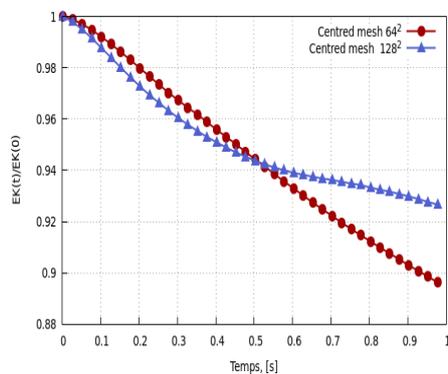


FIGURE 6.11 – Simulation numérique de l’advection d’une goutte dense dans un fluide parfait. Maillage 128^2 : évolution de l’énergie cinétique en fonction du temps pour la formulation non conservative.

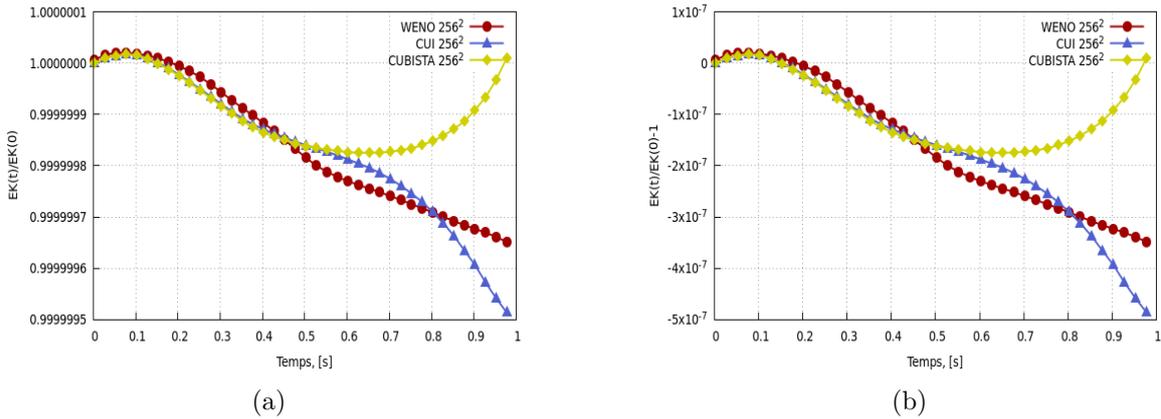


FIGURE 6.12 – Simulation numérique de l’advection d’une goutte dense dans un fluide parfait. Maillage 256^2 : évolution de l’énergie cinétique en fonction du temps pour la formulation conservative. (a) : énergie cinétique $EK(t)$ normalisée sur sa valeur initiale $EK(0)$, (b) : erreur en norme L1 sur l’énergie cinétique $EK(t)$ par rapport à sa valeur initiale $EK(0)$

en norme L_2 qui mesurent la différence entre le champ de couleur à l’instant final (C^{final}) et le champ de couleur à l’instant initial ($C^{initial}$). Pour chacun des trois schémas (WENO, CUI et CUBISTA) de la formulation conservative, l’erreur géométrique sur l’interface diminue en fonction du maillage à l’ordre 1. Par contre, nous remarquons une divergence de la formulation non conservative avec le schéma centré.

Maillage	WENO		CUI		CUBISTA		CENTRÉ	
32×32	2,17E-03	-	2,17E-03	-	2,17E-03	-	4,19E-02	-
64×64	7,53E-04	1.52	7,54E-04	1.52	7,54E-04	1.52	6,67E-02	-
128×128	4,32E-04	0.81	4,19E-04	0.84	4,18E-04	0.85	7,22E-02	-
256×256	2,52E-04	0.77	2,50E-04	0.75	2,51E-04	0.74	Divergence	-

TABLE 6.2 – Simulation numérique de l’advection d’une goutte dense dans un fluide parfait. Comparaison entre la formulation conservative (schéma WENO, CUI et CUBISTA) et la formulation non conservative (schéma centré). Normes d’erreur entre le champ de couleur à l’instant final (C^{final}) et le champ de couleur à l’instant initial ($C^{initial}$).

Ce cas test a également été réalisé en 3D. La solution de référence est obtenue à l’aide du schéma CUBISTA pour un maillage 256^3 . La figure 6.14 illustre une comparaison de la solution issue de la formulation conservative avec la solution exacte. Nous constatons encore une fois une conservation de l’énergie cinétique de la sphère, avec une précision de l’ordre de 10^{-7} . Cela se confirme en observant l’évolution temporelle de la sphère présentée sur la figure 6.14. En effet, grâce au schéma conservatif, la sphère préserve sa forme initiale sans subir aucune déformation, conservant par la suite son énergie cinétique initiale.

6.5 Nappe liquide cisailée périodique

Dans ce cas, on considère une nappe liquide périodique [12], d’épaisseur $\delta = 300 \mu m$, sur un domaine carré de côté $L = 10 \delta$. Ce cas test permet d’identifier les problèmes de stabilité et

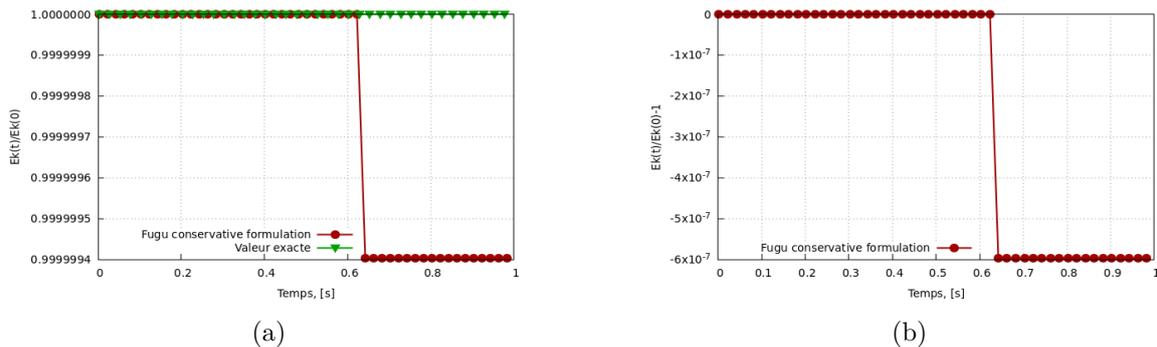


FIGURE 6.13 – Simulation numérique de l’advection d’une goutte dense dans un fluide parfait. Maillage 256^3 : évolution de l’énergie cinétique en fonction du temps pour la formulation conservative. (a) : énergie cinétique $EK(t)$ normalisée sur sa valeur initiale $EK(0)$, (b) : erreur en norme L1 sur l’énergie cinétique $EK(t)$ par rapport à sa valeur initiale $EK(0)$.

de dissipation des schémas numériques. Les masses volumiques du liquide et du gaz sont : $\rho_{liquide} = 10^3$ et $\rho_{gaz} = 1$. Les viscosités du liquide et du gaz, la gravité ainsi que la tension de surface sont nulles. À l’état initial, la nappe liquide évolue à une vitesse $U_l = 2 \text{ m s}^{-1}$, et est entourée par un courant gazeux possédant une vitesse $U_g = 15 \text{ m s}^{-1}$. Par ailleurs, une perturbation (6.45) est introduite afin de déclencher une instabilité,

$$v(x, y) = 10^{-2} U_l \sin\left(\frac{2\pi x}{L}\right) \exp\left[-\left(\frac{2y}{\delta}\right)^2\right]. \quad (6.45)$$

Du fait des conditions aux limites périodiques, l’énergie cinétique totale Ek (somme des énergies cinétiques des phases liquide et gazeuse) doit rester constante au cours du temps, jusqu’au déclenchement de l’instabilité. L’interface à l’instant où l’instabilité se développe est illustré sur les figures 6.15-6.16-6.17-6.18. Nous constatons que la formulation non conservative entraîne un développement plus tôt de l’instabilité avec le schéma centré. Alors qu’avec la formulation conservative (schémas WENO, CUI et CUBISTA), l’interface reste peu perturbée.

Pour comprendre l’impact de ces instabilités, les figures 6.19 et 6.20 illustrent l’évolution de l’énergie cinétique obtenue avec le nouveau schéma AMP (à l’aide des schéma WENO, CUI et CUBISTA) et la formulation non conservative (à l’aide du schéma centré et upwind). Pour le schéma AMP, nous observons que le niveau d’énergie cinétique totale reste constant, avec une légère diminution qui est provoquée par la formation de tourbillons localisés à l’interface. En revanche, une diminution significative de l’énergie cinétique est constatée vers la fin de simulation pour la formulation non conservative. Cette diminution est due à la dissipation numérique induite par les schémas et à l’inconsistance entre le transport de la masse volumique et la quantité de mouvement.

6.6 Rupture de barrage

Le problème de rupture de barrage compte parmi les plus anciens problèmes rencontrés en hydraulique. Ce problème a été largement étudié dans la littérature par plusieurs approches ex-

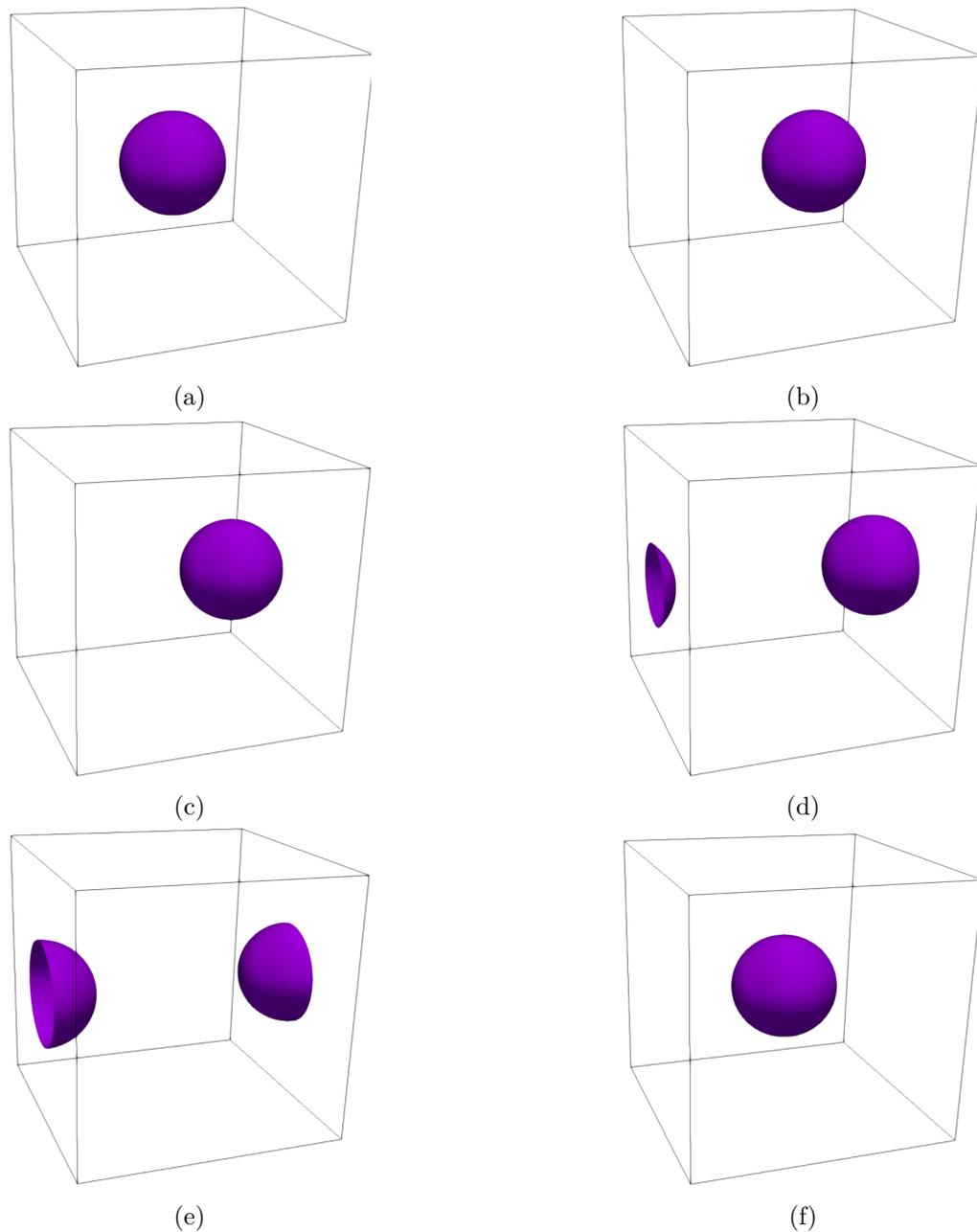


FIGURE 6.14 – Simulation numérique de l’advection d’une goutte dense dans un fluide parfait. Maillage 256^3 : forme de la goutte après une période avec la formulation conservative pour un schéma de transport de type CUBISTA. La simulation a été réalisée sur 512 processeurs du cluster HPC Irene de TGCC.

périmentales, théoriques et numériques. C’est pour cette raison que nous avons jugé nécessaire de passer ce cas afin de valider le nouveau schéma de transport et de nous assurer de la qualité des résultats obtenus.

Pour la résolution du problème, on s’intéressera au cas pour lequel la position initiale du barrage est représentée sur la figure 6.21. Ce choix correspond aux expériences décrites par Kleefsman et al. [106]. Ainsi, la colonne d’eau située en amont du barrage se propage directement sur un sol sec après la rupture du barrage. On appellera ce cas : rupture sur fond sec en présence d’un

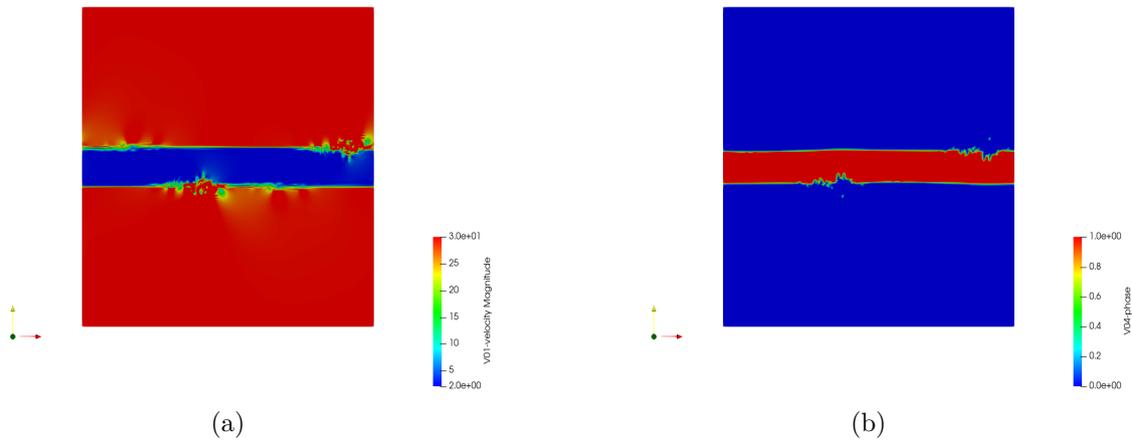


FIGURE 6.15 – Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : représentation de l'évolution de l'interface de la nappe avec la formulation non conservative pour des schémas de transport de type centré. (a) : champ de vitesse, (b) : forme de l'interface.

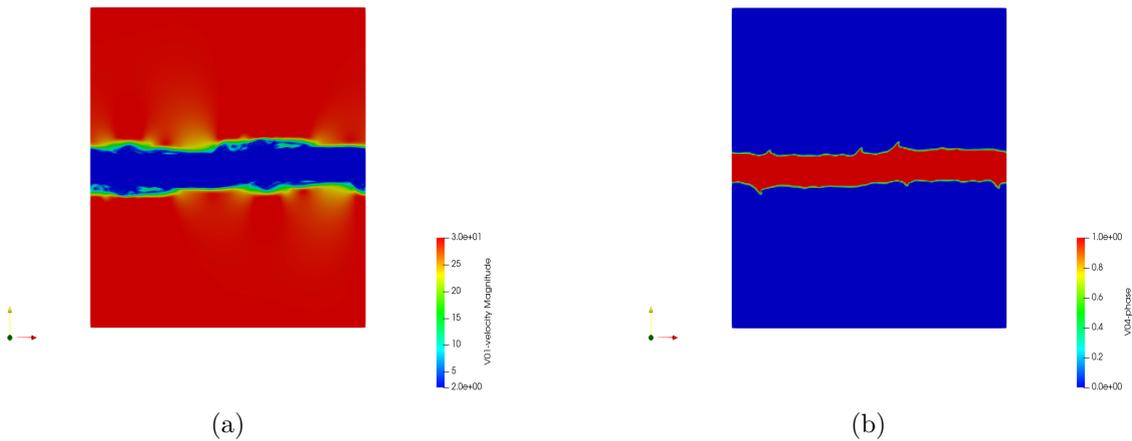


FIGURE 6.16 – Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : représentation de l'évolution de l'interface de la nappe avec la formulation conservative pour un schéma de transport de type WENO. (a) : champ de vitesse, (b) : forme de l'interface.

obstacle. Nous notons que la rupture du barrage sera supposée être instantanée. La colonne d'eau en amont du barrage s'effondrera sous l'effet de la gravité.

Les paramètres physiques utilisés lors de la simulation sont les suivants :

$$\begin{cases} \rho = 1000 \text{kg} \cdot \text{m}^{-3} \text{ et } \mu = 10^{-3} \text{kg} \cdot \text{m}^{-3} \cdot \text{s}^{-1} \text{ dans l'eau} \\ \rho = 1 \text{kg} \cdot \text{m}^{-3} \text{ et } \mu = 10^{-5} \text{kg} \cdot \text{m}^{-3} \cdot \text{s}^{-1} \text{ dans l'air} \\ \sigma = 0.72 \text{N} \cdot \text{m}^{-1}. \end{cases} \quad (6.46)$$

En utilisant la même définition que dans l'article [178], la vitesse à laquelle se déplace l'onde de

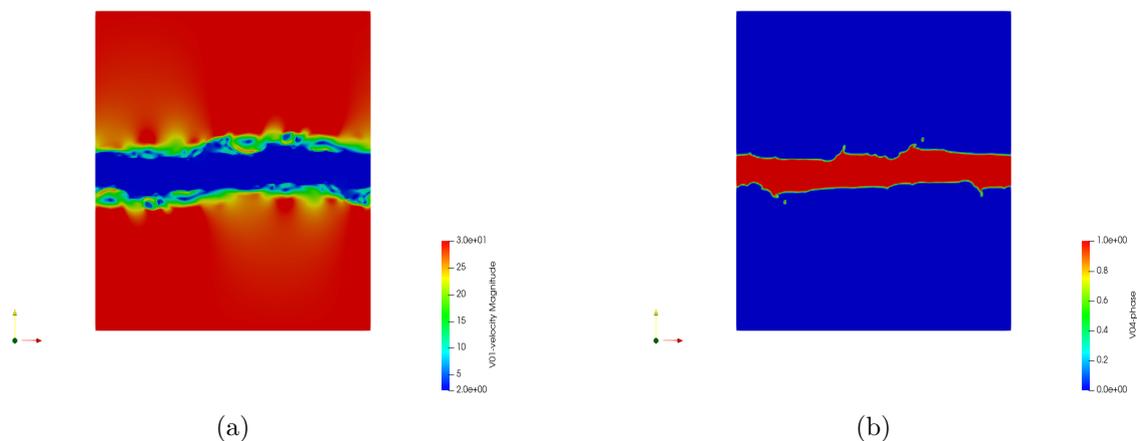


FIGURE 6.17 – Simulation numérique de la nappe liquide cisillée périodique. Maillage 2562 : représentation de l'évolution de l'interface de la nappe avec la formulation conservative pour un schéma de transport de type CUI. (a) : champ de vitesse, (b) : forme de l'interface.

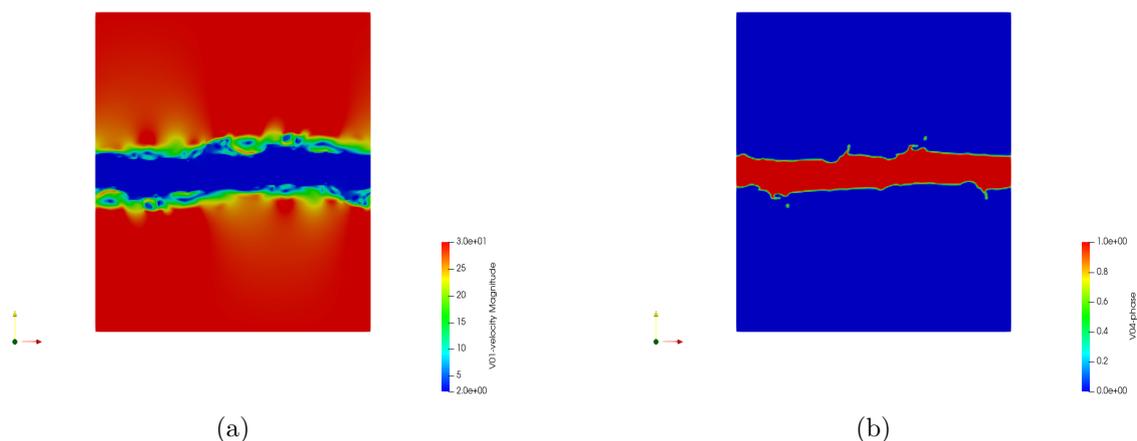


FIGURE 6.18 – Simulation numérique de la nappe liquide cisillée périodique. Maillage 2562 : représentation de l'évolution de l'interface de la nappe avec la formulation conservative pour un schéma de transport de type Cubista. (a) : champ de vitesse, (b) : forme de l'interface.

choc après la rupture du barrage est de l'ordre :

$$U_0 = \sqrt{gH} \quad (6.47)$$

avec H la hauteur du barrage avant rupture et g la gravité. Cette vitesse caractéristique du problème est utilisée pour évaluer les nombres adimensionnels qui pilotent la dynamique du problème :

$$Re = \frac{\rho H U_0}{\mu} \approx 3.13 \times 10^6 \quad (6.48)$$

le nombre de Reynolds, qui est très grand par rapport à 1. Cela signifie que les effets de viscosité peuvent être négligés par rapport aux effets d'inertie. L'eau avance donc sans aucune résistance.

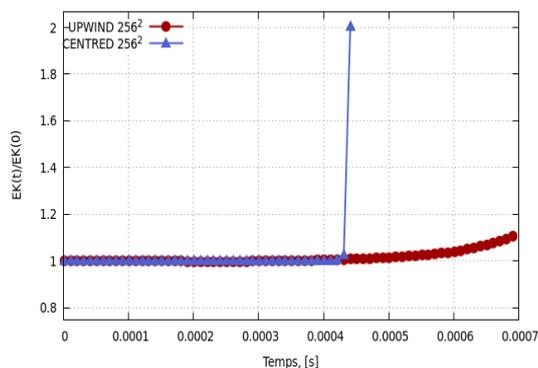


FIGURE 6.19 – Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : évolution de l'énergie cinétique $Ek(t)$ normalisée sur sa valeur initiale $EK(0)$ avec la formulation non conservative.

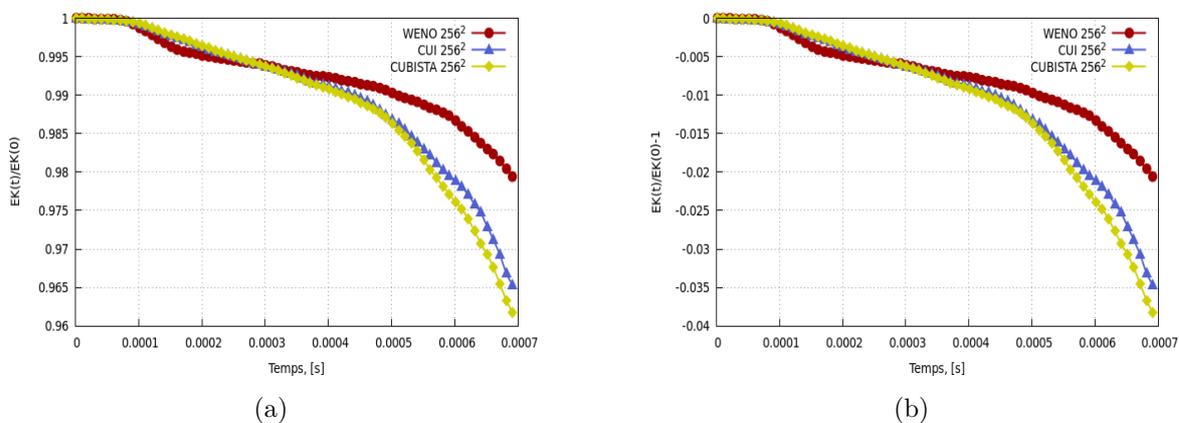


FIGURE 6.20 – Simulation numérique de la nappe liquide cisailée périodique. Maillage 256^2 : évolution de l'énergie cinétique en fonction du temps pour la formulation conservative. (a) : énergie cinétique $Ek(t)$ normalisée sur sa valeur initiale $EK(0)$, (b) : erreur en norme L1 sur l'énergie cinétique $EK(t)$ par rapport à sa valeur initiale $EK(0)$.

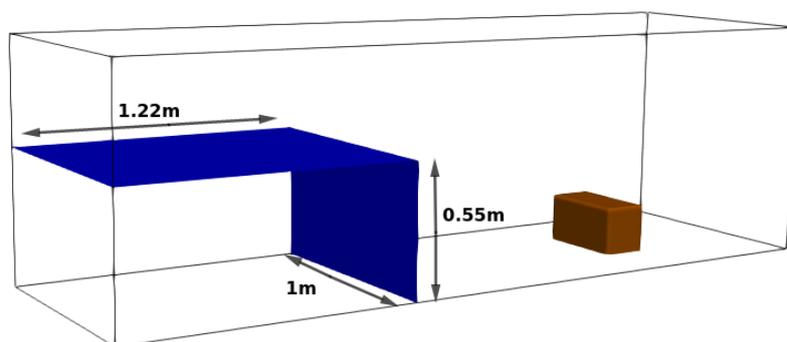


FIGURE 6.21 – Rupture de barrage sur un fond sec

$$W_e = \frac{\rho H V}{\sigma} \approx 4347.22 \quad (6.49)$$

le nombre de Weber est lui aussi très grand devant 1, ce qui nous permet de dire que les

effets de tension de surface sont négligeables par rapport à ceux d'inertie. Nous travaillons donc sur un problème de surface libre.

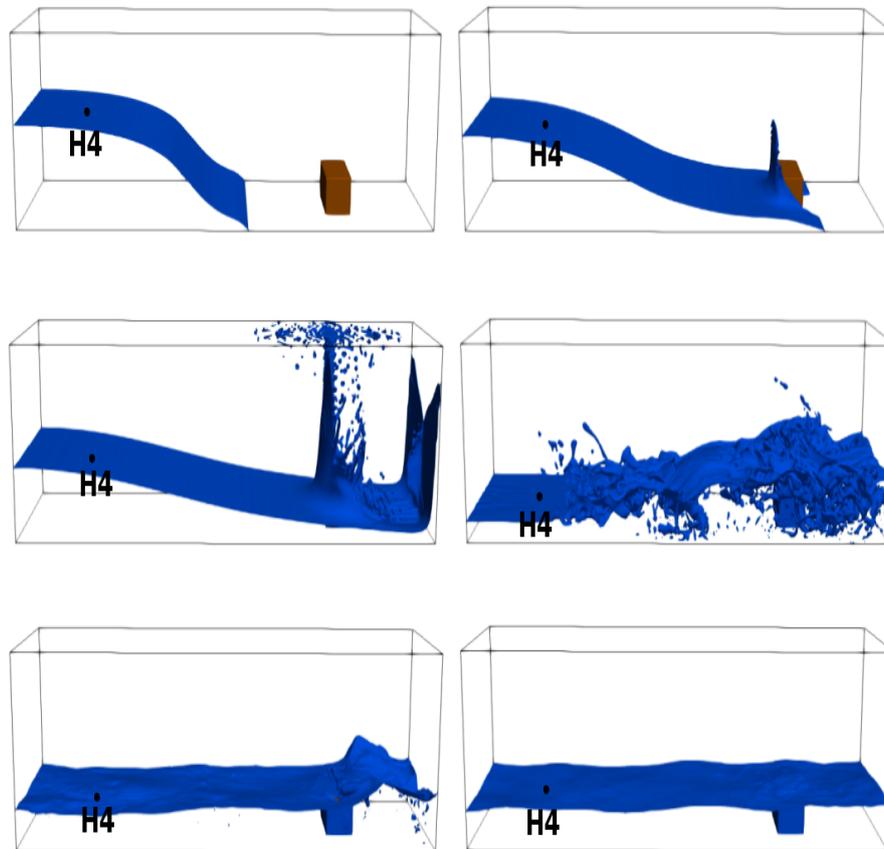


FIGURE 6.22 – Simulation numérique de la rupture de barrage sur fond sec avec la formulation conservative. L'évolution temporelle de la surface libre est présentée. Paramètres de calcul : maillage $320 \times 100 \times 100$, $CFL = 0.4$ et résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2). La simulation a été réalisée sur 120 processeurs du cluster HPC JeanZay de l'IDRIS.

Nous présentons, sur la figure 6.22, l'évolution de la surface libre après la rupture de barrage. Il ressort de ces figures que l'évolution temporelle de la surface libre, issue de la simulation numérique sur un maillage $320 \times 100 \times 100$, montre bien les capacités du code avec le nouveau schéma conservatif à simuler la physique du problème d'impact d'une surface libre sur un obstacle. Cependant, nous notons une divergence du code avec l'ancienne formulation non conservative, à cause de l'inconsistance entre le transport de la masse volumique et la quantité de mouvement. Pour faire une analyse qualitative, les figures 6.23a-6.23b-6.23c et 6.23d montrent deux instants de début de la simulation ainsi que des images de la vidéo de l'expérience (aux mêmes moments). Nous pouvons constater une cohérence entre les résultats issus de la simulation numérique et ceux de l'expérience. Toutefois, pour appuyer cette analyse et valider notre approche, nous proposons d'analyser la figure 6.24 qui indique l'évolution de la hauteur $H4$ au cours du temps. Nous notons une décroissance temporelle de la hauteur de liquide dans la retenue, qui est en

bon accord avec les expériences.

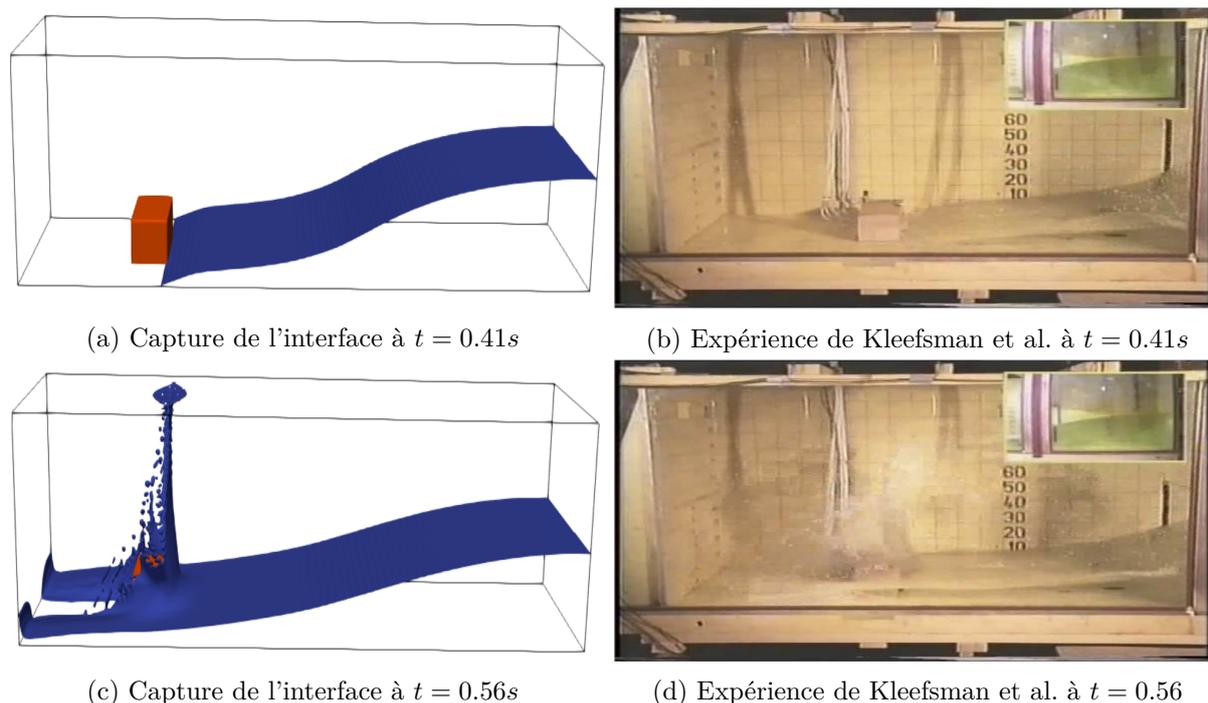


FIGURE 6.23 – Simulation numérique de la rupture de barrage sur fond sec avec la formulation conservative. L'évolution temporelle de la surface libre est présentée en comparaison avec l'expérience de Kleefsman et al. aux temps 0.4s et 0.56s. paramètres de calcul : maillage $320 \times 100 \times 100$, $CFL = 0.4$ et résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2).

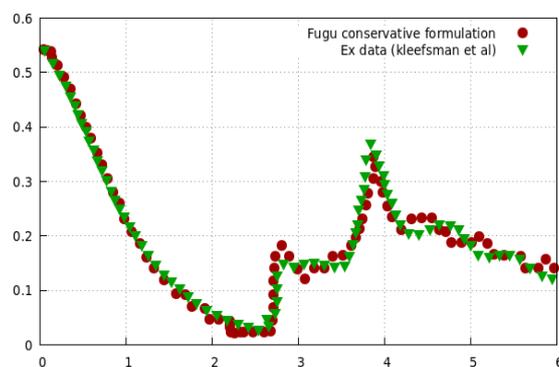


FIGURE 6.24 – Simulation numérique de la rupture de barrage sur fond sec avec la formulation conservative. L'évolution temporelle de la hauteur H_4 est présentée en comparaison avec l'expérience de Kleefsman et al [106]. Paramètres de calcul : maillage $320 \times 100 \times 100$, $CFL = 0.4$ et résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2).

6.7 Étude paramétrique de l'inversion de phase liquide/liquide

6.7.1 Introduction

Le problème d'inversion de phase d'un fluide léger dans un fluide lourd dans une boîte close est un sujet d'étude qui a été initialement proposé par Vincent et Caltagirone [183]. La figure 6.25

représente la configuration 3D qui est généralement utilisée. Les premières études réalisées sont passées de l'étude de faisabilité de ce genre de problème au calcul de grandeurs physiques plus complexes. Les études suivantes [67, 181] ont progressivement ajouté des variables d'études. Tout d'abord, l'étude de l'énergie cinétique et de l'énergie potentielle puis, plus récemment, l'étude de l'enstrophie, de la dissipation et de l'aire interfaciale. L'un des objectifs de ces études est d'arriver à faire converger les résultats lorsque le maillage est raffiné. Cela permettrait de mettre en évidence la convergence des résultats et donc que l'usage d'un maillage plus grossier serait suffisant pour simuler ce genre de problème. Ainsi il serait possible de diminuer les temps et les coûts de calcul. L'autre intérêt est de définir une configuration simple et reproductible de type atomisation numérique.

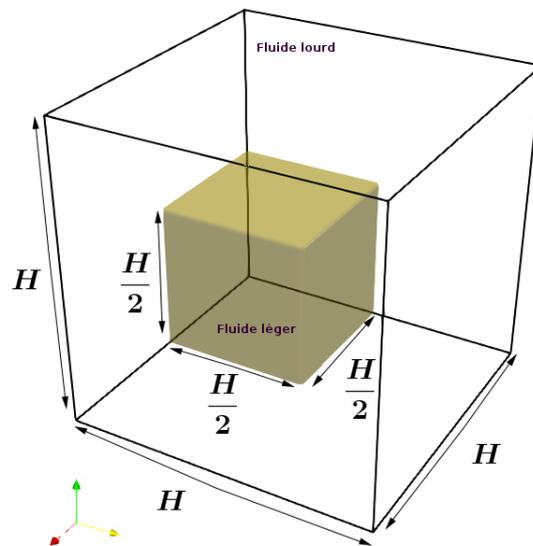


FIGURE 6.25 – Configuration initiale pour un cas 3D de l'inversion de phase liquide/liquide.

Comme nous le verrons plus tard, les nombres de Bond et d'Archimède sont les nombres adimensionnels qui pilotent la dynamique de ce problème. Ces deux nombres dépendent des propriétés physiques des fluides et de la dimension du domaine. Les masses volumiques sont généralement celle de l'huile pour le fluide léger et celle de l'eau pour le fluide lourd. Ce sont les valeurs des viscosités et de la tension de surface qui sont généralement modifiées afin d'imposer les valeurs des nombres de Bond et d'Archimède (ou des nombres de Reynolds et de Weber pour les études les moins récentes).

Nous avons constaté que les premières études menées avaient des nombres de Bond et d'Archimède très élevés. Les forces de volume hydrostatique ou de gravité et les effets d'Archimède étaient donc très importants. Les études suivantes ont progressivement choisi d'imposer des propriétés physiques permettant de diminuer ces deux nombres et donc de diminuer la dispersion et la fragmentation. Au cours des précédentes études nous avons constaté que l'étude de l'enstrophie et de l'aire interfaciale n'étaient pas systématiques, de plus, leurs convergences n'étaient

pas atteintes lorsque le maillage de calcul était raffiné. Une récente étude [159] a montré que la divergence de l'ensrophie se trouvait dans le haut de la boîte proche de l'interface où un grand nombre de changements de topologies se produisent. Une des clés serait donc d'après cette étude de contrôler les instants pour lesquels la topologie change dans cette zone. La dernière étude publiée en avril 2021, [156], est la seule étude que nous avons relevé qui a réussi une convergence en maillage de l'ensrophie. C'est d'ailleurs cette étude que nous avons réalisée dans le chapitre 5, où une convergence de l'ensrophie a été atteinte. Dans l'article [156] deux cas ont été réalisés afin de voir si la position du fluide léger a une influence sur la convergence en maillage. Le premier cas avec le fluide léger placé dans un coin du domaine et le second avec le fluide léger qui est décalé d'une longueur h dans les 3 directions par rapport au premier cas. Les deux cas ont réussi à atteindre une convergence en maillage de l'ensrophie.

6.7.2 Géométrie, conditions initiales et paramètres physiques

Comme décrit dans la figure 6.25, un volume cubique d'un fluide léger (identifié comme le fluide1) est placé dans la partie inférieure d'une boîte remplie d'un fluide lourd (le fluide 2). Nous pourrions assimiler les fluides 1 et 2 comme étant respectivement de l'huile et de l'eau. Nous imposerons une viscosité égale pour les 2 fluides ($\mu_1 = \mu_2 = \mu$). La boîte est de dimension (H, H, H) et le cube de fluide 1 $(H/2, H/2, H/2)$. Nous considérerons un glissement libre et un mouillage partiel sur les murs extérieurs. Nous considérerons également que les angles de contact de 90° sont équivalents à une symétrie de réflexion. L'accélération gravitationnelle est choisie comme étant $g = (0, 0, -9.81)$. Les différents paramètres physiques des cas d'étude A et B sont donnés dans le tableau 6.3, les nombres adimensionnels associés sont aussi présentés dans ce tableau. Ces paramètres ont été choisis afin d'avoir un cas amenant à une situation dans laquelle les nombres de Bond et d'Archimède sont plus faibles que les études existantes : le cas A, et une situation intermédiaire, le cas B. Nous supposons que le cas A, du fait de ses paramètres, aura moins de dispersion et de fragmentation et qu'il sera donc plus simple à faire converger sur les grandeurs étudiées. Le cas B étant proche d'études ayant déjà été réalisées, nous ne pouvons pas affirmer que les résultats convergeront.

	ρ_1	ρ_2	μ_1	μ_2	σ	H	$Ar^{1/2}$	Bo
	$(kg.m^{-3})$	$(kg.m^{-3})$	$(Pa.s)$	$(Pa.s)$	$(kg.s^{-2})$	(m)		
Case A	1000	900	0.1566046	0.1566046	0.122625	0.1	200	80
Case B	1000	900	0.01957557	0.01957557	0.015328	0.1	1600	640

TABLE 6.3 – Simulation numérique de l'inversion de phase liquide/liquide par le solveur tout-couplé. Ici, ρ_2 et μ_2 correspondent respectivement à la masse volumique et à la viscosité du fluide léger, ρ_1 et μ_1 représentent respectivement la masse volumique et la viscosité du fluide lourd, σ est la tension de surface, Ar est nombre d'Archimède et Bo est le nombre de Bond.

Pour, cette nouvelle étude, il a été proposé d'utiliser les nombres de Bond et d'Archimède au lieu de nombres de Reynolds et de Weber afin d'utiliser des nombres adimensionnels qui ne dépendent pas directement de la vitesse. Les nombres adimensionnels qui pilotent la dynamique

de ce problème sont donc la racine carrée du nombre d'Archimède ($Ar^{1/2}$) et le nombre de Bond (Bo) définis de la manière suivante :

$$Ar^{1/2} = \sqrt{\frac{\rho_2(\rho_2 - \rho_1)gH^3}{\mu^3}} \quad (6.50)$$

$$Bo = \frac{(\rho_2 - \rho_1)gH^2}{\sigma} \quad (6.51)$$

Le nombre de Bond compare les forces de volume hydrostatiques ou de gravité aux forces de tensions superficielles.

- Si $Bo < 1$, les forces de tensions superficielles sont plus importantes que les forces de volume hydrostatique ou de gravité.
- Si $Bo > 1$, les forces de volume hydrostatique ou de gravité sont plus importantes que les forces de tensions

Le nombre d'Archimède quant à lui compare les effets d'Archimède à la dissipation visqueuse

- Si $Ar < 1$, la dissipation visqueuse est plus importante que les effets d'Archimède.
- Si $Ar > 1$, les effets d'Archimède sont plus importants que la dissipation visqueuse.

Nous pouvons exprimer ces équations en fonction de la vitesse dont la formule sera donnée par la suite :

$$Ar^{1/2} = \frac{\rho_2 U_g H}{\mu} \quad (6.52)$$

$$Bo = \frac{2\rho_2 U_g^2 H}{\sigma} \quad (6.53)$$

Dans cette étude, nous utiliserons la définition suivante pour la vitesse :

$$U_g = \sqrt{\frac{(\rho_2 - \rho_1)gH}{2\rho_2}} \quad (6.54)$$

6.7.3 Simulations et résultats

Concernant les paramètres numériques, les simulations ont été réalisées sur une grille cartésienne allant de 128^3 à 513^3 cellules, avec un résidu de $\epsilon = 10^{-5}$ du solveur BiCGStab(2). Pour le cas 513^3 il faut noter que les simulations ont été réalisées sur 4096 processeurs du cluster HPC Irene de TGCC. Comme nous l'avons mentionné précédemment, nous avons réalisé des simulations sur les cas A et B pour différents maillages, nous ne présentons dans cette section que les résultats du cas 1. Les résultats du cas B seront présentés dans l'annexe A. Pour les deux cas, nous montrerons des modélisations à différents instants afin de voir comment évoluent les deux fluides au cours du temps. Nous tracerons également les différentes grandeurs que nous avons calculées

au cours du temps.

Dans cette partie, il faudra faire attention aux notations employées. Le code FUGU utilisé pour les simulations est implémenté de manière à ce que les indices 1 et 2 correspondent respectivement au fluide lourd et au fluide léger. Les grandeurs et les graphes qui suivront utiliseront la logique du code comme notations. Au cours des simulations, 9 grandeurs ont été calculées :

- R2, représente la quantité de fluide léger dans le huitième supérieur du domaine. R2=0 s'il n'y a que du fluide lourd et R2=1 s'il y a que du fluide léger. Compte tenu du volume initial de goutte de liquide $V_2 = H^3/8$, il est défini comme :

$$R2 = \frac{\int_{\Omega_{sup}} C dV}{V_2}. \quad (6.55)$$

- Vol2, représente le volume de fluide léger, il permet de vérifier que le volume de fluide léger reste constant au cours du temps. Il se calcule comme :

$$Vol2 = \int_{\Omega_{sup}} C dV. \quad (6.56)$$

- EK1, représente l'énergie cinétique du fluide lourd (1) :

$$Ek1 = \frac{1}{2} \int_{\Omega} (1 - C) \rho_1 \bar{u}^2 dV. \quad (6.57)$$

- EK2, représente l'énergie cinétique du fluide léger (2) :

$$Ek2 = \frac{1}{2} \int_{\Omega} C \rho_2 \bar{u}^2 dV. \quad (6.58)$$

- EP1, représente l'énergie potentielle du fluide lourd (1) :

$$EP1 = \frac{1}{2} \int_{\Omega} (1 - C) \rho_1 \bar{g} z dV. \quad (6.59)$$

- EP2, représente l'énergie potentielle du fluide léger(2) :

$$EP2 = \frac{1}{2} \int_{\Omega} C \rho_2 \bar{g} z dV. \quad (6.60)$$

- ER1, représente l'enstrophie du fluide lourd (1) :

$$ER1 = \int_{\Omega} (1 - C) (\nabla \times \bar{u})^2 dV. \quad (6.61)$$

- ER2, représente l'enstrophie du fluide léger (2) :

$$ER2 = \int_{\Omega} C (\nabla \times \bar{u})^2 dV. \quad (6.62)$$

- Di1, représente la dissipation du fluide lourd (1) :

$$Di1 = \int_{\Omega} (1 - C) 2\mu_1 \overline{\overline{S}} : \overline{\overline{S}} dV. \quad (6.63)$$

- Di2, représente la dissipation du fluide léger (2) :

$$Di2 = \int_{\Omega} C\mu_2 \overline{\overline{S}} : \overline{\overline{S}} dV \quad (6.64)$$

avec $\overline{\overline{S}} = 1/2 (\nabla \bar{u} + (\nabla \bar{u})^T)$.

- Aire, représente l'aire interfaciale. Contrairement aux études précédentes, où elle est approximée comme étant $\sum_{int} S_{cell}$ où $S_{cell} = \Delta x \Delta y = \Delta y \Delta z = \Delta x \Delta z$, dans cette étude, elle est évaluée à partir de la normale à l'interface \vec{n} , en calculant dans un cube unitaire, la surface coupée par le plan $\vec{n} \cdot \vec{x} = \alpha$ conformément à l'approche PLIC de la méthode VOF [116]
- E_{surf} qui est l'énergie interfaciale.

$$E_{surf} = \sigma \times Aire. \quad (6.65)$$

- ϵ est la dissipation issue du bilan d'énergie qui s'écrit :

$$\epsilon = \frac{d}{dt} (Ek_1 + EK_2 + Ep_1 + Ep_2 + E_{surf}). \quad (6.66)$$

6.7.3.1 Simulations

Dans nos simulations présentées sur la figure 6.26, nous observons les mêmes phénomènes que dans la figure A.1. Tout d'abord la montée du fluide léger vers le haut de la boîte, environs 1 s pour le cas A. Ensuite, le fluide léger recircule avec plus au moins de fragmentation et dispersion. Dans le case A, les nombres d'Archimèdes sont moins élevés que pour le cas B, il y a donc moins de dispersion et de fragmentation. Les 2 fluides vont ensuite se stabiliser progressivement avec la remontée des dernières goutte du fluide léger vers le haut avec un mouvement oscillant de type ballotement dans la cavité.

6.7.3.2 Résultats

Dans un premier temps nous avons effectué une analyse de convergence pour évaluer l'effet de la taille du maillage sur la solution physique. L'étude est réalisée sur différents maillage structurés de taille 128^3 , 256^3 et 512^3 . Comme cité précédemment l'énergie cinétique, l'énergie potentielle, l'énergie interfaciale, l'ensrophie, l'aire interfaciale, la dissipation, le volume de l'huile, le ratio et le bilan énergétique sont les caractéristiques physiques étudiées. La solution physique semble converger quand la taille du maillage devient plus fine.

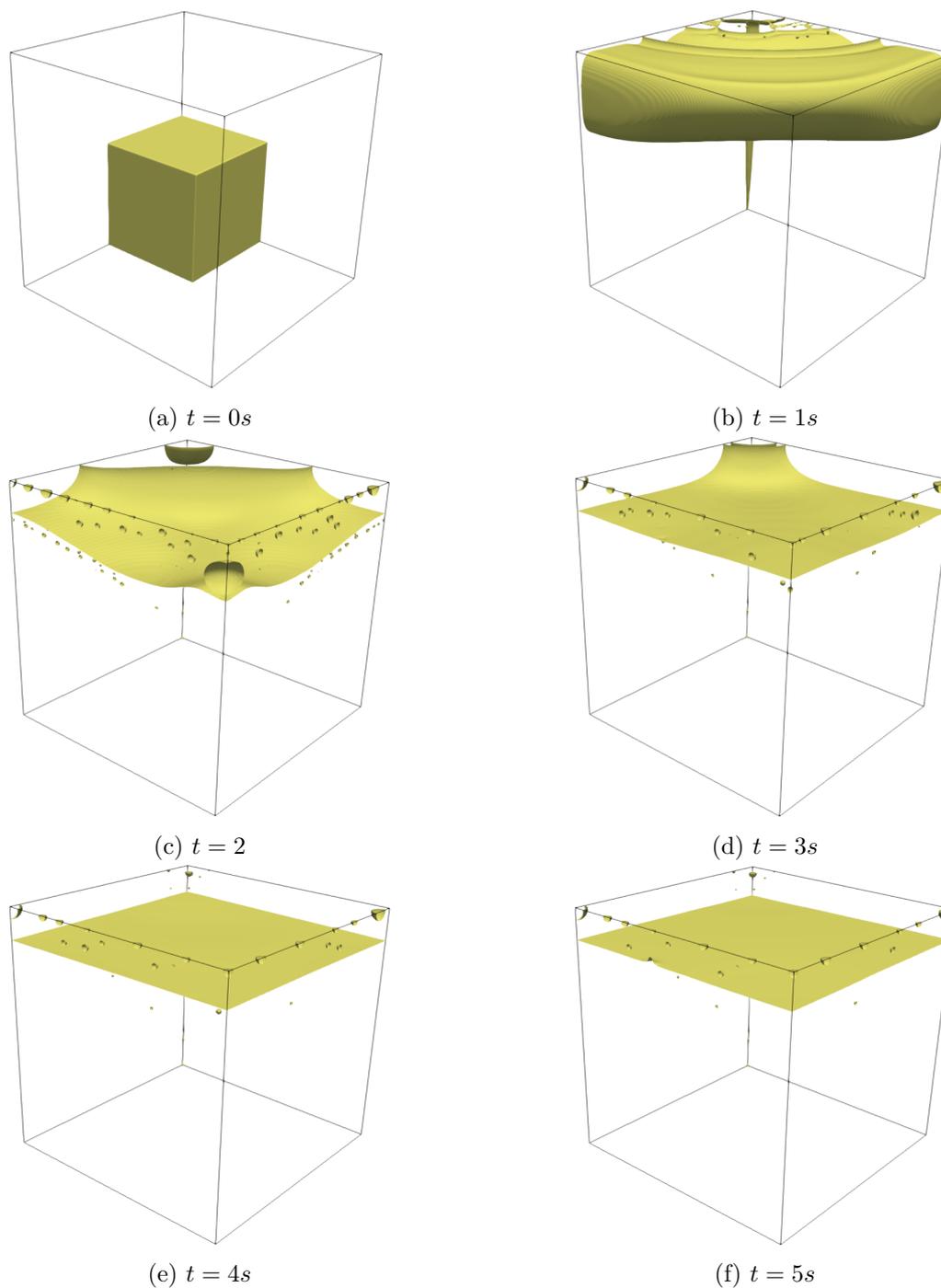


FIGURE 6.26 – Simulation numérique de l'inversion de phase liquide/liquide. L'évolution temporelle de l'interface des 2 fluides est représentée en jaune. Paramètres de calcul : un maillage $256 \times 256 \times 256$, une $CFL = 0.5$ et un résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2).

6.7.3.3 R2 et Vol2

Pour le cas A, la figure 6.27 illustre l'évolution du ratio R2 et du volume Vol. Nous pouvons observer que R2 tend vers 1 à la fin des simulations. R2 étant un coefficient nous indiquant la quantité de fluide léger dans le haut du domaine d'étude. Si $R2 = 1$, cela signifie qu'il y a 100% de fluide léger dans le 8^{ème} supérieur de la boîte, ce qui correspond à la quantité de fluide léger.

L'inversion de phase s'est donc bien produite. Les courbes traçant Vol étant constants au cours du temps, nous vérifions bien que les volumes de fluides léger sont bien conservés quelque soit la grille utilisée.

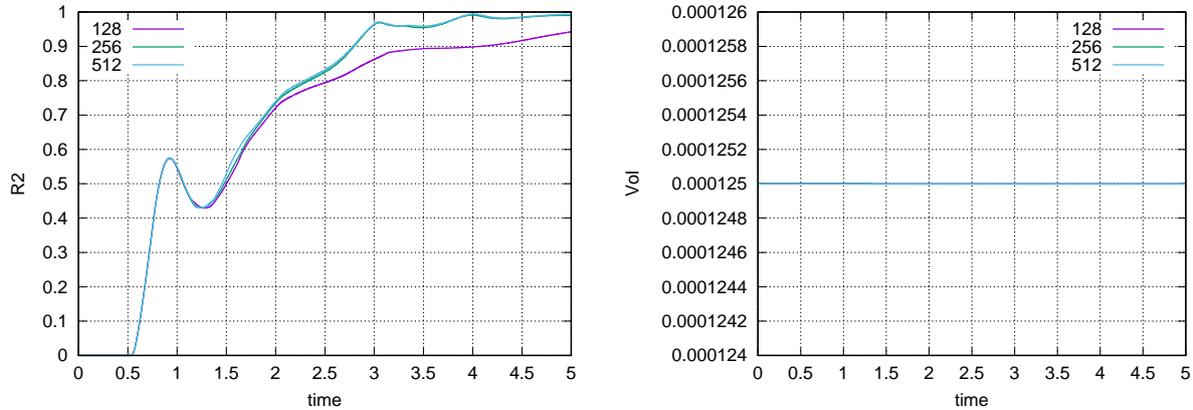


FIGURE 6.27 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : le ration R2, (b) : le volume de fluide léger Vol.

6.7.3.4 L'aire interfaciale et l'énergie surfacique

Connaissant les topologies initiales et finales nous pouvons calculer facilement l'aire interfaciale et l'énergie surfacique à ces instants. À l'instant initial, le fluide léger a la forme d'un cube de hauteur $H/2$ avec 3 faces en contact avec le fluide lourd. Les autres faces sont en contact avec les parois et ne sont donc pas comptabilisées. Nous avons donc $Aire = 3 \times 0,05^3 m^2 = 7,5 \cdot 10^{-3} m^2$ et $E_{surf} = 9.196875 \cdot 10^{-4} J$. Lorsque t tends vers l'infini, la surface est plane et est de dimension $H^2 = 10^{-2} m^2$. Quant à elle, l'énergie surfacique E_{surf} est de l'ordre $1.122625 \cdot 10^{-4} J$.

À partir de la figure 6.28, nous pouvons constater que les valeurs initiales de l'aire interfaciale et de l'énergie surfacique correspondent bien à ce que l'on vient de calculer. Les différences que nous pouvons observer sont de l'ordre de 0.25%. Concernant les valeurs finales de l'aire interfaciale et de l'énergie surfacique, nous avons un écart de l'ordre de 4%. Cela s'explique par le fait que sur la durée de nos simulations, les fluides n'ont pas encore atteint un état stable et l'interface des 2 fluides n'est donc pas plane. Nous pouvons néanmoins observer que les graphes convergent progressivement vers la valeur attendue.

6.7.3.5 L'énergie potentielle

Connaissant les topologies à l'instant initial et à l'instant final, nous pouvons déterminer facilement l'énergie potentielle des 2 fluides à ces instants et ainsi vérifier que l'énergie potentielle de nos simulations est cohérente. A l'instant initial, t tend vers 0 seconde :

$$E_{p,1}^{init} = \rho_1 g H^4 / 32 \quad (6.67)$$

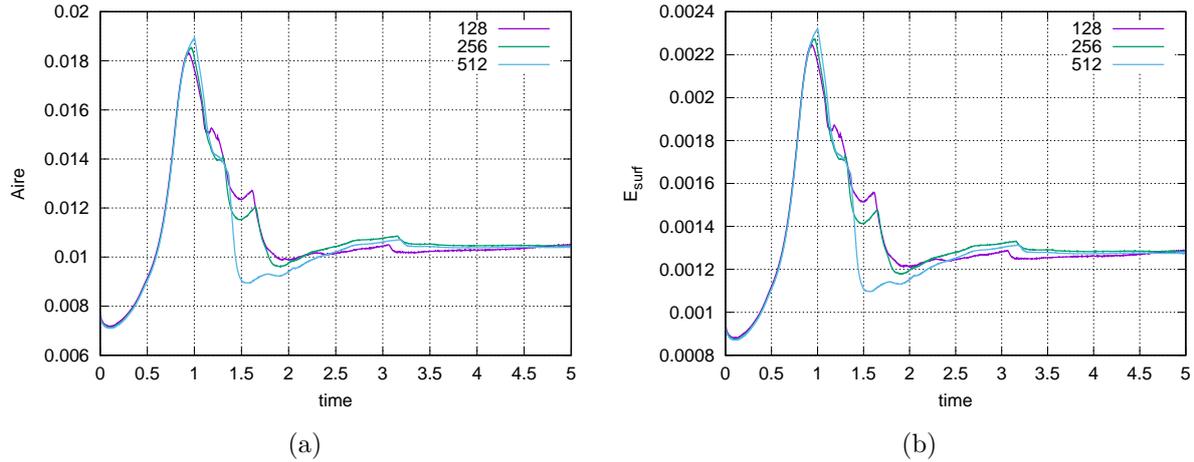


FIGURE 6.28 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : L'aire interfaciale $Aire$. (b) : L'énergie surfacique E_{surf} .

$$E_{p,2}^{init} = \rho_2 g H^4 15/32 \quad (6.68)$$

Lorsque l'inversion de phase est achevée :

$$E_{p,1}^{final} = \rho_1 g H^4 15/128 \quad (6.69)$$

$$E_{p,2}^{final} = \rho_2 g H^4 49/128 \quad (6.70)$$

Nous regroupons les différentes valeurs obtenues dans le tableau 6.4.

	Valeur théorique	Cas A	Écart
$E_{p,1}^{init}$	0.45984375	0.4598433	$9.87 \times 10^{-5} \%$
$E_{p,2}^{init}$	0.027590625	0.0275909	$1.32 \times 10^{-3} \%$
$E_{p,1}^{final}$	0.3755390625	0.3755524	$3.55 \times 10^{-3} \%$
$E_{p,2}^{final}$	0.10346484375	0.1034528	$1.16 \times 10^{-2} \%$

TABLE 6.4 – Simulation numérique de l'inversion de phase liquide/liquide. Valeurs initiales et finales de l'énergie potentielle pour un maillage 512^3 .

Si l'on compare les valeurs obtenues, nous remarquons que la différence entre les valeurs théoriques et numériques est comprise entre 0 et $1.16 \times 10^{-2} \%$. Nous pouvons donc considérer que les valeurs que nous obtenons pour l'énergie potentielle sont correctes.

6.7.3.6 L'énergie cinétique

Si on se fie à l'Appendix A de [67], nous pouvons approximer la décroissance de l'énergie cinétique selon l'écoulement des fluides. Dans le fluide léger nous pouvons approximer la décroissance de l'énergie cinétique par une équation de la forme Kt^{-2} et dans le fluide lourd par une équation de la forme Ke^{-ct} , ou K et c sont des constantes. L'approximation en Kt^{-2} rappelle l'approximation

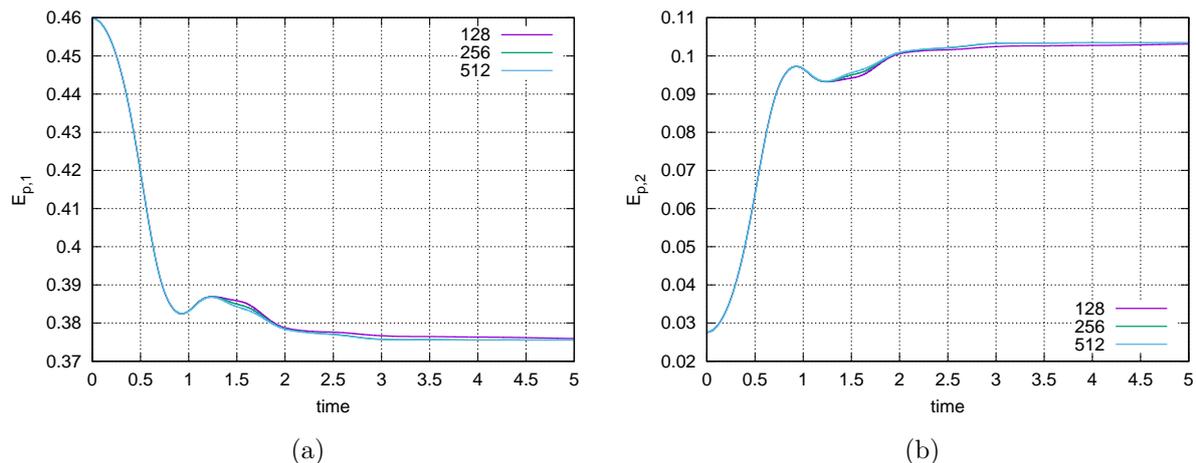


FIGURE 6.29 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : $E_{p,1}$, énergie potentielle dans le fluide lourd. (b) : $E_{p,2}$, énergie énergie potentielle dans le fluide léger.

mation de la turbulence isotrope homogène. Lorsque la turbulence est isotrope homogène, cela signifie que toutes les caractéristiques de la turbulence sont invariantes par rotation. Lorsque la décroissance est de la forme Ke^{-ct} , le problème d'inversion de phase est dans une configuration dans laquelle il y a peu de rupture et de coalescence dans le fluide léger. Pour le cas A, figure 6.30, nous pouvons voir que pour le fluide lourd l'équation n'est pas respectée. La décroissance est plutôt de la forme Kt^{-4} . En revanche, pour le fluide léger, l'approximation de la forme Ke^{-ct} se vérifie.

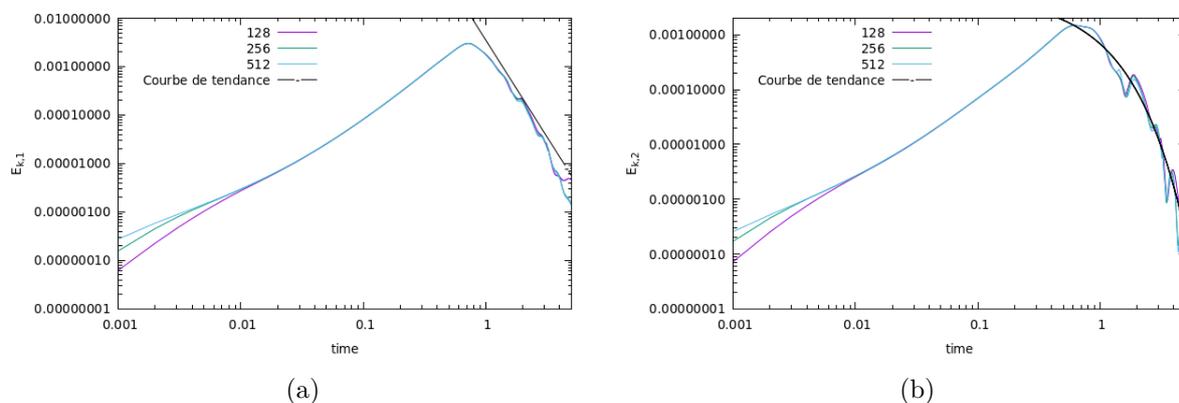


FIGURE 6.30 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : $E_{k,1}$, énergie cinétique en coordonnées log-log dans le fluide lourd. (b) : $E_{k,2}$, énergie cinétique en coordonnées log-log dans le fluide léger. L'approximation des décroissances est en ligne noire.

6.7.3.7 L'entrophie

Si on se fie toujours à l'Appendix de [67], la décroissance de l'entrophie est de la forme Kt^{-3} . Dans les deux fluides nous avons bien une décroissance de la forme Kt^{-3} . Concernant l'entrophie, nous pouvons également observer l'amplitude maximale à $t \approx 1s$.

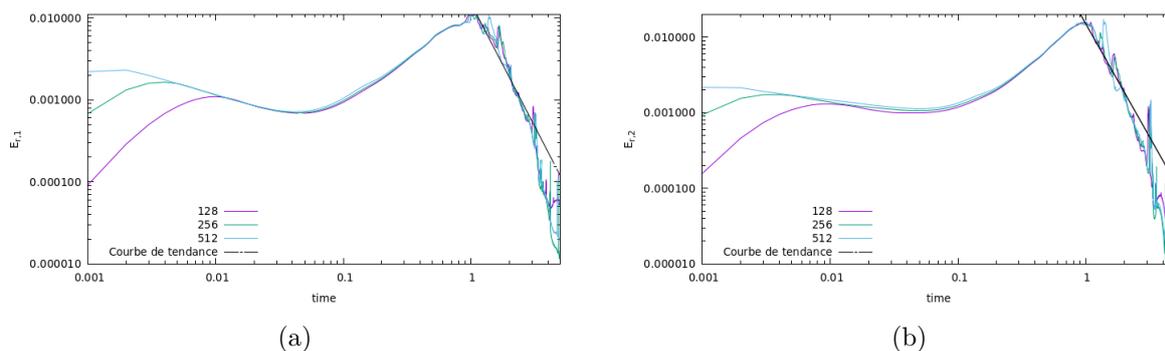


FIGURE 6.31 – Simulation numérique de l’inversion de phase liquide/liquide. (a) : $E_{r,1}$, enstrophie en coordonnées log-log dans le fluide lourd. (b) : $E_{r,2}$, enstrophie en coordonnées log-log dans le fluide léger. L’approximation des décroissances est en ligne noire.

6.7.3.8 La dissipation

Bien que les formules de l’enstrophie et de la dissipation soient différentes, il existe une relation entre ces deux grandeurs. En effet, si on se réfère à l’article [110] la dissipation peut être approximée à partir de l’enstrophie comme suit $D_{i,n} \approx \mu_n E_{r,n}$

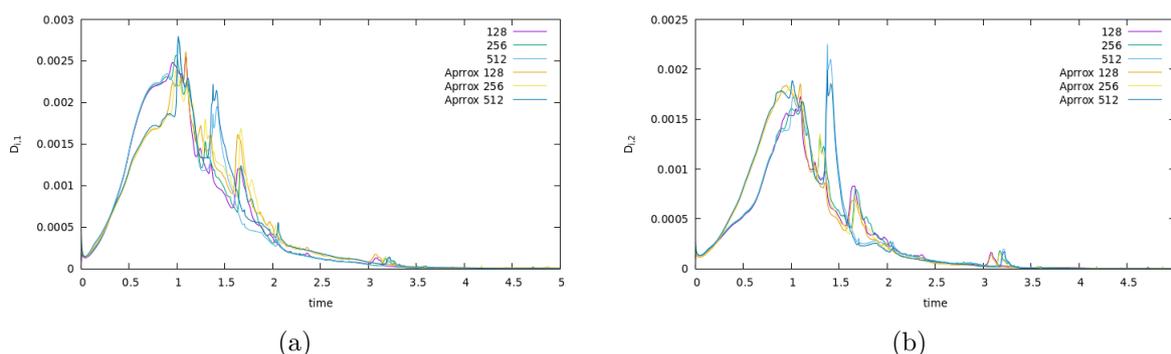


FIGURE 6.32 – Simulation numérique de l’inversion de phase liquide/liquide. (a) : $D_{i,1}$, dissipation dans le fluide lourd. (b) : $D_{i,2}$, dissipation dans le fluide léger. Comparaison entre la dissipation calculée par le code et la dissipation calculée avec l’approximation $D_{i,n} \approx \mu E_{r,n}$.

Nous allons tracer dans un graphique la dissipation calculée par le code et la dissipation calculée par cette équation $D_{i,n} \approx \mu E_{r,n}$ afin de comparer les résultats que nous obtenons. Nous pouvons voir dans les figures 6.32a et 6.32b que sur chaque graphique les courbes approximatives déterminées avec l’équation suivent bien les mêmes variations que les courbes déterminées par le code. Nous pouvons donc considérer l’enstrophie comme étant un marqueur de la dissipation.

6.7.3.9 Commentaires généraux

Pour l’ensemble des grandeurs que nous avons calculées, nous remarquons que les convergences en maillage sont atteintes. Nous avons également remarqué que les courbes présentant des pics, comme l’énergie cinétique (Voir Fig.6.33) et l’enstrophie (Voir Fig.6.34), se produisent au même instant pour un cas donné. Nous pouvons prendre pour exemple le cas A. Ces maxima d’enstrophie et d’énergie cinétique se produisent vers 1s. Si on se réfère à la figure 6.26, nous pouvons

observer que ces maxima ont lieu lorsque le fluide léger atteint le haut de la boîte. C'est après avoir atteint le haut de la boîte que le fluide va se fragmenter et recirculer avant de se stabiliser. Il semble donc normal que l'énergie cinétique et l'enstrophie diminuent ensuite.

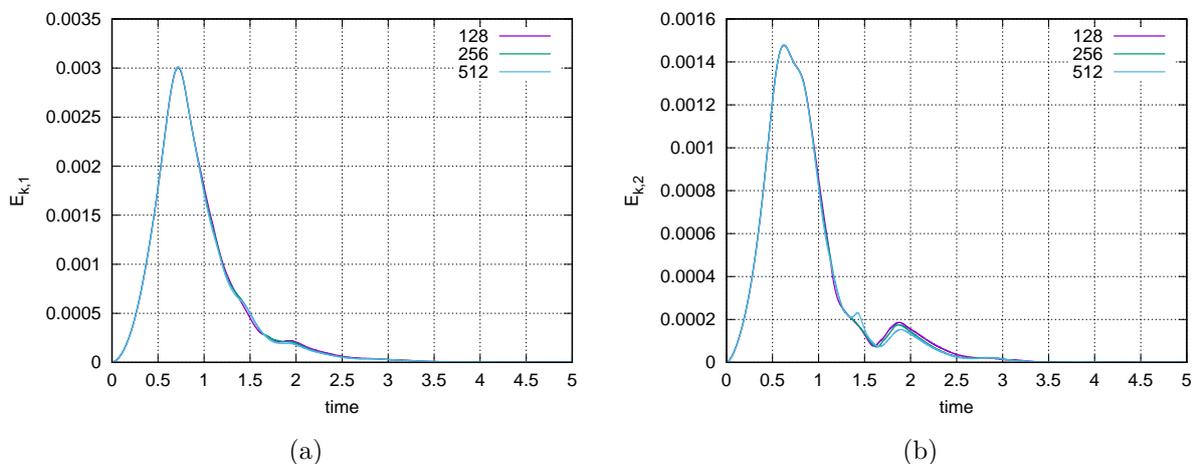


FIGURE 6.33 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : $E_{k,1}$, énergie cinétique dans le fluide lourd. (b) : $E_{k,2}$, énergie cinétique dans le fluide léger.

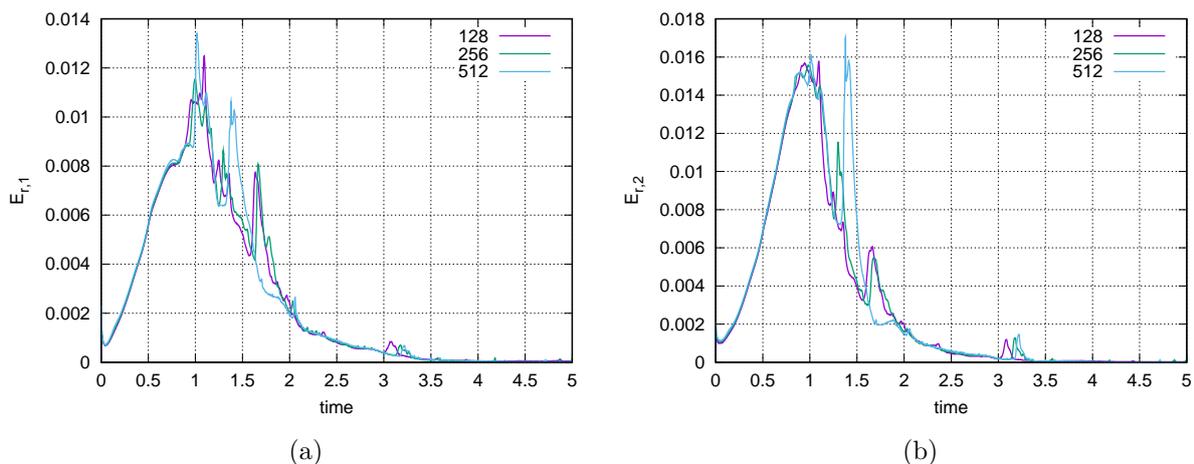


FIGURE 6.34 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : $E_{r,1}$, enstrophie dans le fluide lourd. (b) : $E_{r,2}$, enstrophie dans le fluide léger.

Nous pouvons également constater une conservation de l'énergie totale, comme c'est illustré sur la figure 6.35. En effet, la variation de l'énergie totale est égale à la dissipation malgré une légère différence de l'ordre 10^{-3} .

6.8 Caractérisation de l'instabilité 3D d'un jet visqueux à faible nombre de Reynolds

La caractérisation de l'instabilité 3D d'un jet visqueux à faible nombre de Reynolds est un phénomène classique, aussi bien dans la nature que dans l'industrie. Afin de bien comprendre ce phénomène, des expériences, des simulations numériques et des analyses dimensionnelles sont

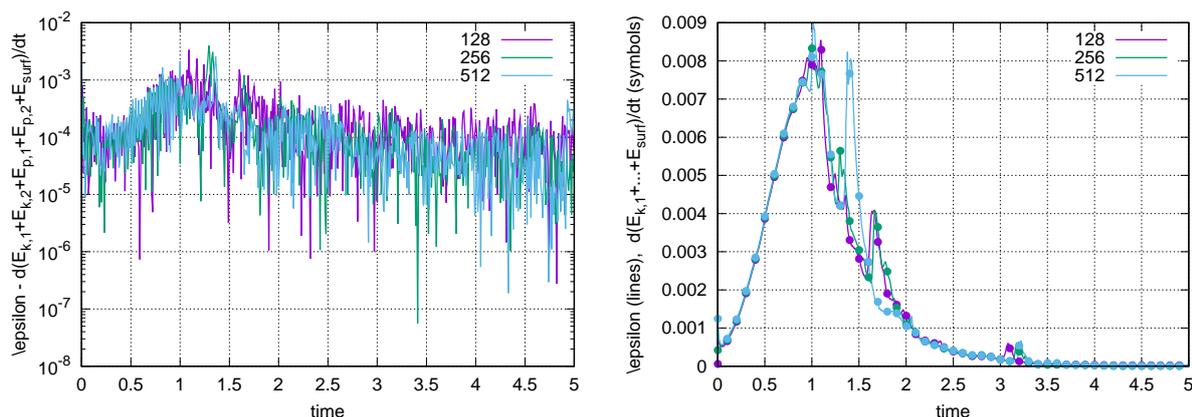


FIGURE 6.35 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : bilan énergétique. (b) : comparaison entre la dissipation ϵ et la variation temporelle de l'énergie totale.

utilisée dans la littérature. Dans cette étude, le problème étudié est le suivant : une cavité cubique de côté H , remplie d'air est étudiée. Comme cela est schématisé sur la figure 6.36, nous injectons un fluide très visqueux à un débit constant dans la partie supérieure de la cavité, tandis que la force gravitationnelle agit avec une intensité constante g dans le sens négatif de la direction z .

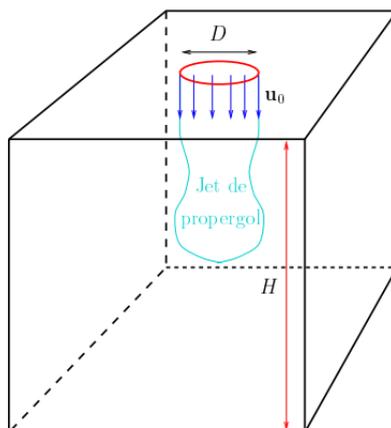


FIGURE 6.36 – Configuration initiale pour l'injection d'un jet visqueux à faible nombre de Reynolds.

Pour les bords du domaine, nous avons utilisé des conditions limites de glissement sur les parois solides, et des conditions limites de sortie, dans la partie supérieure de la cavité. Les caractéristiques de la simulation numérique sont données dans le tableau 6.5.

En outre, Tomé et al [168] donnent deux paramètres déterminants pour l'apparition de l'instabilité : le rapport d'aspect du jet $H/D > 3\pi$ et $Re < 0.52$, où Re est le nombre de Reynolds basé sur le diamètre d'entrée D et la vitesse d'entrée V_0 . Ainsi, le but de cette étude est de vérifier la capacité de la méthode entièrement couplée à traiter ce type de problème et de fournir une bonne prédiction du comportement physique associé. En ce qui concerne les simulations numériques, les matrices résultant de la discrétisation seront résolues avec le même solveur, le BiCGStab(2)

Test case	ρ_1	ρ_2	μ_1	μ_2	Re	$\frac{H}{D}$	V_0	ρ_2/ρ_1	μ_2/μ_1
1	1.1768	1800	1.85×10^{-3}	500	0.1728	12.5	0.6	1529	1.6×10^5
2	1.1768	1800	1.85×10^{-3}	300	0.6	10	1	1529	2.7×10^5

TABLE 6.5 – Les paramètres physiques décrivant le cas du jet visqueux. Ici, ρ_2 et μ_2 représentent respectivement la masse volumique et la viscosité du propergol, ρ_1 et μ_1 représentent respectivement la masse volumique et la viscosité du gaz, $Re = \frac{\rho_2 D V_0}{\mu_2}$ est le nombre de Reynolds associé au propergol, $\frac{H}{D}$ est le rapport d'aspect du jet et V_0 est la vitesse d'entrée. (pour des raisons de simplicité, le propergol est supposé se comporter comme un fluide newtonien).

avec un résidu de $\epsilon = 10^{-5}$.

6.8.1 Résultats

Les figures 6.37-6.38 présentent l'évolution du liquide au cours du temps pour deux nombres de Reynolds : $Re = 0.6$ et $Re = 0.1728$, en utilisant un maillage de 100^3 raffiné au niveau de l'injecteur. De ces figures, nous pouvons constater que nos simulations numériques avec le solveur tout-couplé procurent de très bons résultats quant à la physique du jet, qui sont en bon accord avec le critère théorique donné par Tomé et al [168]. En effet, le premier cas $Re = 0.1728$ correspond à une instabilité du jet, alors que dans le deuxième, $Re = 0.6$ l'oscillation du jet n'apparaît pas.

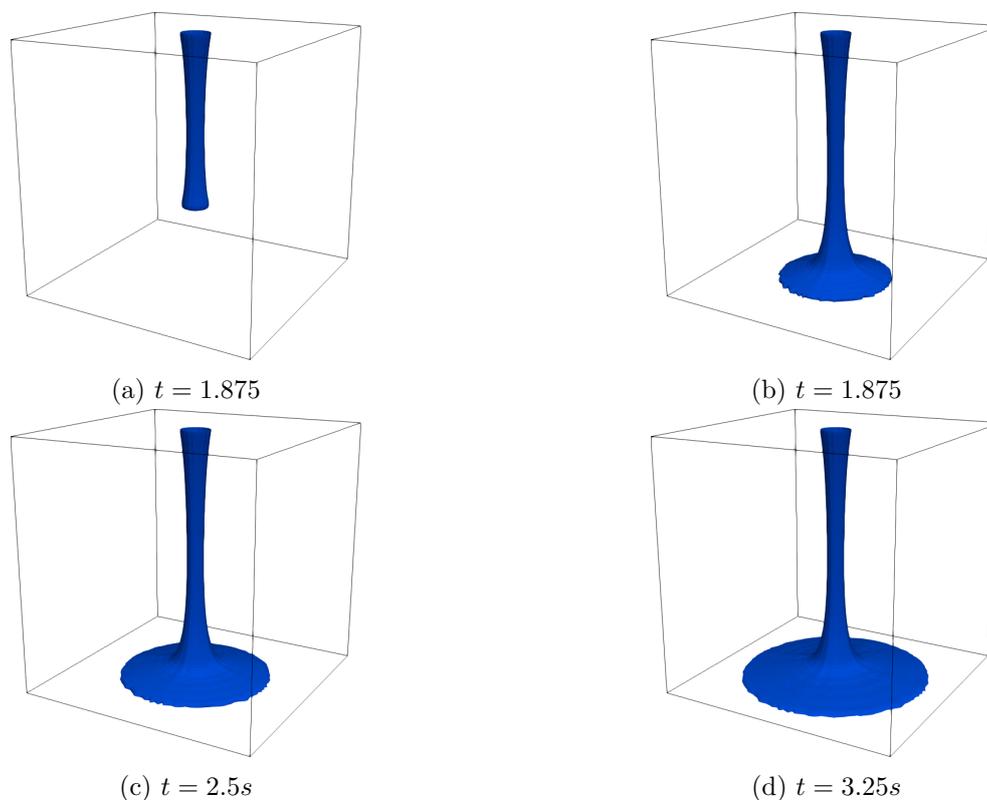


FIGURE 6.37 – Simulation numérique du remplissage d'une cavité par un jet visqueux. Résultats correspondant à $Re = 0.6$, simulés par le solveur tout-couplé, sur une grille cartésienne 100^3 , $Re = 0.6$. L'interface des deux fluides est représentée en bleu.

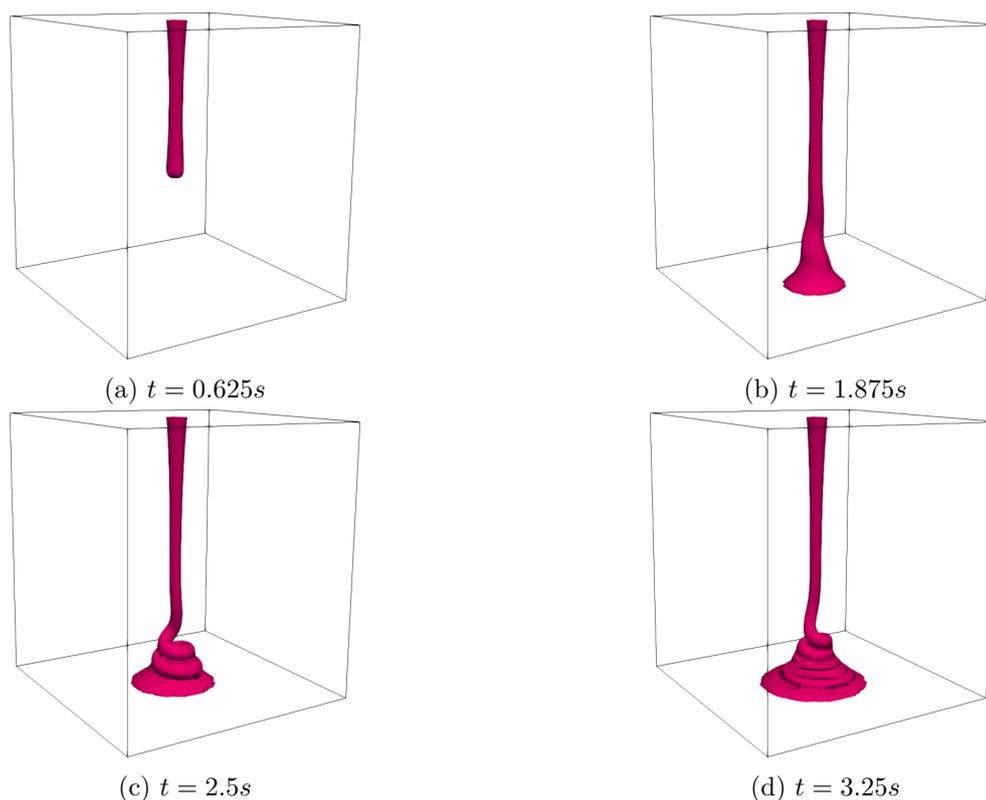


FIGURE 6.38 – Simulation numérique du remplissage d’une cavité par un jet visqueux. Résultats correspondant à $Re = 0.6$, simulés par le solveur tout-couplé, sur une grille cartésienne 100^3 , $Re = 0.1728$. L’interface des deux fluides est représentée en rose.

6.9 Conclusion

Dans ce chapitre nous avons décrit une formulation conservative (AMP) telle qu’elle est implémentée dans le code de résolution d’écoulements diphasiques FUGU. Grâce à l’intégration de cette formulation, nous avons proposé une discrétisation consistante de la masse volumique et de la quantité de mouvement permettant de stabiliser le code lorsqu’on a des instabilités numériques. Pour le cas de sphère dense, le nouveau schéma a permis de réduire la dissipation numérique au niveau de la précision machine. En ce qui concerne la nappe cisailée, nous avons montré également que le nouveau schéma permettait d’éliminer la formation de tourbillons non physiques localisés à l’interface. Ceux-ci sont à l’origine de la diminution significative de l’énergie cinétique. De plus, cette méthode s’est révélée efficace dans le cas de la rupture de barrage avec des résultats en accord avec les références de la littérature. Nous avons ensuite réalisé deux nouveaux cas d’inversion de phase avec des paramètres physiques différents. Nous avons obtenu des résultats très encourageants pour nos deux cas d’étude pour des maillages allant jusqu’à 512^3 . Enfin à travers le dernier cas du jet visqueux, nous avons pu montrer la robustesse de notre solveur tout-couplé combiné avec le nouveau schéma de transport, lorsqu’il fait face à des problèmes raides de forts rapports de masses volumiques et de viscosités.

Chapitre 7

Conclusion et perspectives

Cette thèse s'est articulée autour du développement et de la validation d'une approche monolithique destinée à simuler des écoulements diphasiques incompressibles et complexes (au sens des forts rapports de masses volumiques et de viscosités), et de l'application de cette approche à l'étude de quelques problèmes physiques. Pour ce faire, nous avons construit un solveur tout-couplé original parallèle 3D permettant d'assurer le couplage direct entre la vitesse et la pression par la contrainte d'incompressibilité $\nabla \cdot \bar{u} = 0$, et d'aboutir à une résolution très précise et robuste, sans erreur de splitting en temps, des écoulements diphasiques incompressibles.

Après avoir établi les équations du modèle à un fluide (MUF) à partir des équations locales monophasiques, nous avons construit une méthode de suivi d'interface permettant de gérer l'évolution de l'interface par le biais de l'équation du taux de présence. Notre choix s'est porté vers l'utilisation d'une méthode eulérienne, et en particulier, la méthode VOF de Yue et Weymouth [77] qui gère intrinséquement le changement topologique de l'interface. Cette méthode a ainsi été validée avec succès sur plusieurs cas-tests académiques. Nous avons obtenu un ordre 1 ou 2 de convergence spatiale pour la méthode et une précision machine sur la conservation du volume.

Le couplage entre la méthode de suivi d'interface et la résolution des équations de Navier-Stokes représente une étape déterminante dans la simulation des écoulements diphasiques. En effet, le terme capillaire et les propriétés physiques qui apparaissent dans le bilan de la quantité de mouvement nécessitent généralement l'utilisation de techniques de suivi d'interfaces et la mise en place d'un solveur pour le calcul du champ de vitesse requis pour gérer l'évolution de l'interface. Le calcul de ce champ de vitesse implique un traitement spécifique du couplage vitesse-pression et une satisfaction de la contrainte d'incompressibilité. Une très large littérature a été consacrée au traitement du couplage vitesse-pression. En ce sens, nous avons présenté les deux grandes catégories d'approches, avec tout d'abord l'utilisation d'approches découplées comme la méthode de projection-correction, puis l'utilisation d'approches couplées telles que le lagrangien augmenté. L'utilisation d'une méthode de type projection-correction pour traiter le couplage vitesse-pression comporte de sérieuses difficultés, à savoir l'imposition de conditions limites sur

la pression et le fait qu'elles sont beaucoup plus sensibles au rapport de masses volumiques. La méthode du lagrangien augmenté présente, quant à elle, une résolution très robuste puisqu'elle permet de passer des cas diphasiques à forts rapports de masses volumiques. Néanmoins, elle présente un défaut majeur qui réside dans son coût d'utilisation.

La mise en place des approches citées précédemment présente des difficultés et peut devenir rapidement coûteuse. Nous nous sommes donc orientés vers le développement d'un solveur tout-couplé original et exact afin de palier à ces défauts. Ce solveur permet de satisfaire les équations du modèle à 1 fluide en même temps, sans passer par l'opérateur de splitting temporel, ni par l'introduction d'un paramètre numérique de pénalisation. C'est d'ailleurs ce qui justifie l'appellation "méthode exacte" si les systèmes linéaires creux issus de la discrétisation des équations de conservation sont résolus avec une précision machine. Notre stratégie pour résoudre ces systèmes linéaires creux est de faire appel au solveur BiCGStab(2) sur la totalité du système, en choisissant la structure CSR (compressed storage raw) afin de ne stocker que les coefficients non nuls de la matrice, et ensuite de construire un préconditionneur efficace pour accélérer la convergence du solveur itératif. L'établissement du préconditionneur a été fait en utilisant l'opérateur pression-convection-diffusion (PCD) pour le bloc pression, et le préconditionneur triangulaire par bloc pour le bloc de vitesse. La méthodologie développée a consisté en l'utilisation des solveurs multi-grille SMG et PFMG scalaires (pour chaque composante de vitesse et la pression) de la librairie HYPRE, couramment utilisée dans la communauté du HPC. Le solveur tout-couplé s'est ainsi montré robuste dans des cas de validations diphasiques à forts rapports de masses volumiques et de viscosités. La montée d'une bulle dans une colonne d'eau, la chute d'une sphère dense dans l'air ou encore l'instabilité d'un jet visqueux furent utilisés pour montrer et tester l'efficacité du solveur tout-couplé. De plus, nous avons mesuré les temps passés strictement dans le solveur tout-couplé et ceux passés dans la méthode du lagrangien augmenté et celle de la projection standard, nous avons relevé une différence significative. En effet, le solveur tout-couplé permet de réduire le temps CPU de 50% à 80% par rapport à la méthode du lagrangien augmenté, alors que cette réduction est comprise entre 57% et 90% par rapport à la méthode de projection standard.

L'extension du solveur tout-couplé pour la résolution de problèmes diphasiques tridimensionnels autour du calcul massivement parallèle a été achevée avec succès et constitue une contribution majeure de cette thèse. Cette contribution nous a permis de répondre à nos attentes, dans le sens où la contrainte d'incompressibilité est satisfaite avec un seuil situé entre 10^{-4} et 10^{-8} , quels que soient les rapports de masses volumiques ($\frac{\rho_2}{\rho_1} \geq 10^6$) et de viscosités ($\frac{\mu_2}{\mu_1} \geq 10^6$) à la traversée de l'interface. En ce qui concerne le développement du solveur tout-couplé parallèle, nous avons constaté que sa mise en œuvre nécessite d'employer un préconditionneur pour obtenir une convergence rapide du solveur itératif. Pour ce faire, nous nous sommes appuyés sur les travaux de Niall Bottland pour préconditionner le bloc de pression (complément de Schur) que nous avons enrichi par le développement d'un nouveau préconditionneur original pour le bloc de vitesse. Nous avons

parallélisé ces préconditionneurs via l'utilisation de multigrilles géométriques (SMG et PFMG) de la bibliothèque gratuite HYPRE dont la précision et l'efficacité sont avérées. De très bonnes courbes de performance sont obtenues si on utilise un nombre minimum de points par sous-bloc parallèle (32^3 cellules par proc). Nous avons réussi à conserver des performances très correctes, une efficacité entre 96% et 100% jusqu'à 6400 processeurs sur la machine SKL IRENE du CEA, et une efficacité entre 89% et 100% jusqu'à 8064 processeurs sur la machine JEAN ZAY de l'IDRIS. Dans la suite, nous avons pu mener tout un ensemble de simulations parallèles sur des problèmes physiques variés et des maillages contenant de 4 millions (cas de la sphère dense et la montée d'une bulle dans une collone d'eau) à 1 milliard de points (cas de l'inversion de phase).

Notre seconde contribution dans ce travail concerne les aspects spécifiquement diphasiques des équations du modèle, c'est-à-dire le traitement du terme convectif dans le bilan de quantité de mouvement via un schéma AMP, en particulier lorsque le rapport des masses volumiques de deux fluides est élevé. Au cours de cette démarche, nous avons réécrit l'équation de la quantité de mouvement sous forme conservative ou divergentielle. Ensuite, nous avons conjugué le transport de la quantité de mouvement avec le transport de la masse volumique sur le volume de contrôle de chaque composante de vitesse. Lors du transport, les flux des deux quantités (masse volumique et quantité de mouvement) ont été évalués de manière consistante en utilisant des schémas algébriques TVD type CUI et CUBISTA. Concernant l'intégration temporelle des équations, un schéma d'ordre trois de Runge-Kutta (SSP-RK3) a été implémenté. Au final, nous avons validé ce nouveau schéma AMP sur des problèmes d'écoulements diphasiques pour les cas à forts rapports de masses volumiques (de 10^3 à 10^6), notamment pour le cas de la sphère dense et la nappe liquide cisailée, le schéma AMP a permis de mettre en exergue la dissipation numérique causée par les erreurs de quantité de mouvement à l'interface. Sur le plan physique, nous avons pu valider nos simulations en les comparant à des expériences, comme dans le cas de la rupture de barrage et l'injection d'un fluide visqueux dans une cavité. Nous avons également abordé le problème de l'inversion de phase, où des phénomènes de coalescence ou de rupture sont mis en jeu. L'étude de convergence menée est très prometteuse et nous poursuivons l'étude détaillée afin de réaliser un benchmark numérique avec plusieurs codes CFD développés dans des laboratoires français, sous la supervision de Stéphane Zaleski.

Au niveau de la valorisation scientifique, le développement du solveur tout-couplé 2D a permis la publication d'un article portant sur la résolution des équations de Navier-Stokes pour des écoulements incompressibles monophasiques et diphasiques dans le journal CiCP. L'article a été soumis en décembre 2020 et accepté en mars 2021. Un deuxième article portant sur le développement d'un solveur tout-couplé pour la simulation des écoulements diphasiques à forts rapports de masses volumiques et de viscosités a été accepté dans le journal Fluids MDPI. En juin 2022, un acte de congrès sur le développement d'un solveur tout-couplé parallèle 3D pour la simulation des écoulements diphasiques incompressibles a été rédigé et soumis pour un symposium au Congrès Mécanique Français. Une étude de performance (scalabilité forte et faible)

du solveur tout-couplé est en cours, et fera l'objet d'une publication quand elle sera finalisée. Quand il sera abouti, un nouveau Benchmark numérique sur le problème d'inversion de phases fera aussi l'objet d'une publication.

Concernant les perspectives, ce travail de thèse peut être amélioré notamment sur le plan numérique en :

- analysant l'impact des conditions aux limites dans le préconditionneur PCD en particulier lorsque des conditions d'entrée/sortie (inflow/outflow) sont utilisées pour le champ de vitesse.
- mettant en œuvre une linéarisation de Newton qui s'avère plus efficace pour pouvoir résoudre correctement le terme convectif du bilan de la quantité de mouvement.
- évaluant de manière consistante les flux de masse et de quantité de mouvement en utilisant la reconstruction géométrique de la VOF.

La poursuite de la réflexion sur l'extension du solveur tout-couplé que nous avons mis en œuvre à des écoulements compressibles tridimensionnelles est également envisagée et a été en partie abordée dans la thèse (voir l'annexe I), même si elle n'est pas présentée ici. Une autre piste de travail consiste à impliquer les termes capillaires, en utilisant des méthodes de type frontières immergées. D'autres voies possibles portent sur la simulation numérique des configurations réelles d'écoulements diphasiques comme le cisaillement d'une nappe liquide et la comparaison des solutions à des expériences. A plus long terme, nous pouvons rajouter l'aspect anisotherme et les transferts de masse et de chaleur avec ou sans changement de phase.

Bibliographie

- [1] J. ADELSBERGER, P. ESSER, M. GRIEBEL, S. GROSS, M. KLITZ et A. RÜTTGERS. “3D incompressible two-phase flow benchmark computations for rising droplets”. In : *Proceedings of the 11th world congress on computational mechanics (WCCM XI), Barcelona, Spain*. T. 179. 2014 (cf. p. [167-170](#)).
- [2] G. C. AGBANGLA, P. BACCHIN et E. CLIMENT. “Collective dynamics of flowing colloids during pore clogging”. In : *Soft Matter* 10.33 (2014), p. 6303-6315 (cf. p. [150](#)).
- [3] C. ALAPPAT, A. BASERMANN, A. R. BISHOP, H. FEHSKE, G. HAGER, O. SCHENK, J. THIES et G. WELLEIN. “A Recursive Algebraic Coloring Technique for Hardware-Efficient Symmetric Sparse Matrix-Vector Multiplication”. In : *ACM Trans. Parallel Comput.* 7.3 (juin 2020) (cf. p. [99](#)).
- [4] M. ALVES, P. OLIVEIRA et F. PINHO. “A convergent and universally bounded interpolation scheme for treatment of advection”. In : *International Journal for Numerical Methods in Fluids* 41 (jan. 2003), p. 47-75 (cf. p. [183](#)).
- [5] P. R. AMESTOY, A. BUTTARI, J.-Y. L’EXCELLENT et T. MARY. “Performance and scalability of the block low-rank multifrontal factorization on multicore architectures”. In : *ACM Transactions on Mathematical Software (TOMS)* 45.1 (2019), p. 1-26 (cf. p. [152](#)).
- [6] P. R. AMESTOY, I. S. DUFF, J.-Y. L’EXCELLENT et J. KOSTER. “A fully asynchronous multifrontal solver using distributed dynamic scheduling”. In : *SIAM Journal on Matrix Analysis and Applications* 23.1 (2001), p. 15-41 (cf. p. [152](#)).
- [7] P. R. AMESTOY, A. BUTTARI, J.-Y. L’EXCELLENT et T. MARY. “Performance and Scalability of the Block Low-Rank Multifrontal Factorization on Multicore Architectures”. In : *ACM Trans. Math. Softw.* 45.1 (fév. 2019) (cf. p. [37, 99](#)).
- [8] P. R. AMESTOY, I. S. DUFF, J.-Y. L’EXCELLENT et J. KOSTER. “A Fully Asynchronous Multifrontal Solver Using Distributed Dynamic Scheduling”. In : *SIAM Journal on Matrix Analysis and Applications* 23.1 (2001), p. 15-41 (cf. p. [37, 99](#)).
- [9] P. ANGOT, C.-H. BRUNEAU et P. FABRIE. “A penalization method to take into account obstacles in incompressible viscous flows”. In : *Numerische Mathematik* 81.4 (1999), p. 497-520 (cf. p. [153](#)).

- [10] T. ARRUFAT, M. CRIALESI ESPOSITO, D. FUSTER, Y. LING, L. MALAN, S. PAL, R. SCARDOVELLI, G. TRYGGVASON et S. ZALESKI. “A mass-momentum consistent, Volume-of-Fluid method for incompressible flow on staggered grids”. In : *Computers & Fluids* 215 (jan. 2021), p. 104785 (cf. p. 37, 178).
- [11] S. F. ASHBY et R. D. FALGOUT. “A Parallel Multigrid Preconditioned Conjugate Gradient Algorithm for Groundwater Flow Simulations”. In : *Nuclear Science and Engineering* 124.1 (1996), p. 145-159 (cf. p. 99, 104).
- [12] A. ASURI MUKUNDAN, T. MÉNARD, J. C. BRÄNDLE DE MOTTA et A. BERLEMONT. “A 3D Moment of Fluid method for simulating complex turbulent multiphase flows”. In : *Computers Fluids* 198 (2020), p. 104364 (cf. p. 141, 190).
- [13] E. AULISA, S. MANSERVISI, R. SCARDOVELLI et S. ZALESKI. “Interface reconstruction with least-squares fit and split advection in three-dimensional Cartesian geometry”. In : *Journal of Computational Physics* 225.2 (2007), p. 2301-2319 (cf. p. 57).
- [14] A. BAKER, R. FALGOUT, T. KOLEV et U. YANG. “Scaling Hypre’s Multigrid Solvers to 100,000 Cores”. In : oct. 2012, p. 261-279 (cf. p. 99, 104).
- [15] E. BAVIER, M. HOEMMEN, S. RAJAMANICKAM et H. THORNQUIST. “Amesos2 and Belos : Direct and Iterative Solvers for Large Sparse Linear Systems”. In : *Scientific Programming* 20 (jan. 2012), p. 241-255 (cf. p. 99).
- [16] M. BENZI, G. H. GOLUB et J. LIESEN. “Numerical solution of saddle point problems”. In : *Acta Numerica* 14 (2005), p. 1-137 (cf. p. 102).
- [17] M. BENZI, M. NG, Q. NIU et Z. WANG. “A Relaxed Dimensional Factorization preconditioner for the incompressible Navier–Stokes equations”. In : *Journal of Computational Physics* 230.16 (2011), p. 6185-6202 (cf. p. 114).
- [18] M. BENZI et M. OLSHANSKII. “An Augmented Lagrangian-Based Approach to the Oseen Problem”. In : *SIAM Journal on Scientific Computing* 28 (jan. 2006), p. 2095-2113 (cf. p. 114).
- [19] M. BOLLHÖFER, A. EFTEKHARI, S. SCHEIDEGGER et O. SCHENK. “Large-scale Sparse Inverse Covariance Matrix Estimation”. In : *SIAM Journal on Scientific Computing* 41.1 (2019), A380-A401 (cf. p. 99).
- [20] M. BOLLHÖFER, O. SCHENK, R. JANALIK, S. HAMM et K. GULLAPALLI. “State-of-the-Art Sparse Direct Solvers”. In : (2020). Sous la dir. d’A. GRAMA et A. H. SAMEH, p. 3-33 (cf. p. 99).
- [21] N. BOOTLAND, A. BENTLEY, C. KEES et A. WATHEN. “Preconditioners for Two-Phase Incompressible Navier–Stokes Flow”. In : *SIAM Journal on Scientific Computing* 41.4 (jan. 2019), B843-B869 (cf. p. 36, 109, 115).

- [22] N. BOOTLAND, A. BENTLEY, C. KEES et A. WATHEN. “Preconditioners for Two-Phase Incompressible Navier–Stokes Flow”. In : *SIAM Journal on Scientific Computing* 41.4 (2019), B843-B869 (cf. p. [152](#), [160](#)).
- [23] O. BOTELLA et R. PEYRET. “Benchmark spectral results on the lid-driven cavity flow”. In : *Computers Fluids* 27 (mai 1998), p. 421-433 (cf. p. [116](#), [118-121](#)).
- [24] J. BRACKBILL, D. KOTHE et C. ZEMACH. “A continuum method for modeling surface tension”. In : *Journal of Computational Physics* 100.2 (1992), p. 335-354 (cf. p. [62](#), [71](#), [72](#)).
- [25] J. U. BRACKBILL, D. B. KOTHE et C. ZEMACH. “A continuum method for modeling surface tension”. In : *Journal of computational physics* 100.2 (1992), p. 335-354 (cf. p. [153](#)).
- [26] J. BRAMBLE et J. PASCIAK. “Iterative techniques for time dependent Stokes problems”. In : *Computers Mathematics with Applications* 33.1 (1997), p. 13-30 (cf. p. [36](#)).
- [27] A. C. BROWN et R. K. PAASCH. “The Accelerations of a Wave Measurement Buoy Impacted by Breaking Waves in the Surf Zone”. In : *Journal of Marine Science and Engineering* 9.2 (2021) (cf. p. [31](#)).
- [28] M. BUSSMANN, D. KOTHE et J. SICILIAN. “Modeling High Density Ratio Incompressible Interfacial Flows”. In : t. 257. Jan. 2002 (cf. p. [177](#), [179](#)).
- [29] J. CAHOUE et J.-P. CHABARD. “Some fast 3D finite element solvers for the generalized Stokes problem”. In : *International Journal for Numerical Methods in Fluids* 8.8 (1988), p. 869-895 (cf. p. [160](#)).
- [30] J. CAHOUE et J.-P. CHABARD. “Some fast 3D finite element solvers for the generalized Stokes problem”. In : *International Journal for Numerical Methods in Fluids* 8.8 (1988), p. 869-895 (cf. p. [36](#), [113](#)).
- [31] M. CAI, A. NONAKA, J. B. BELL, B. E. GRIFFITH et A. DONEV. “Efficient variable-coefficient finite-volume Stokes solvers”. In : *Communications in Computational Physics* 16.5 (2014), p. 1263-1297 (cf. p. [160](#)).
- [32] J.-P. CALTAGIRONE et J. BREIL. “Sur une méthode de projection vectorielle pour la résolution des équations de Navier-Stokes”. In : *Comptes Rendus de l’Académie des Sciences - Series IIB - Mechanics-Physics-Astronomy* 327.11 (1999), p. 1179-1184 (cf. p. [36](#), [71](#), [77](#)).
- [33] J.-P. CALTAGIRONE et S. VINCENT. “Sur une méthode de pénalisation tensorielle pour la résolution des équations de Navier–Stokes”. In : *Comptes Rendus de l’Académie des Sciences-Series IIB-Mechanics* 329.8 (2001), p. 607-613 (cf. p. [241](#)).
- [34] J.-P. CALTAGIRONE et S. VINCENT. “A Kinematics Scalar Projection Method (KSP) for Incompressible Flows with Variable Density”. In : *Open Journal of Fluid Dynamics* 05 (mars 2015) (cf. p. [36](#)).

- [35] R. F. CERQUEIRA, E. E. PALADINO, F. EVRARD, F. DENNER et B. van WACHEM. “Multiscale modeling and validation of the flow around Taylor bubbles surrounded with small dispersed bubbles using a coupled VOF-DBM approach”. In : *International Journal of Multiphase Flow* 141 (2021), p. 103673 (cf. p. 150).
- [36] A.-J. CHORIN. “Numerical Solution of the Navier-Stokes Equations”. In : *Mathematics of Computation* 22.104 (1968), p. 745-762 (cf. p. 36, 73, 152).
- [37] E. CHOW et Y. SAAD. “Experimental Study of ILU Preconditioners for Indefinite Matrices”. In : *J. COMPUT. APPL. MATH* 86 (1997), p. 387-414 (cf. p. 36, 111).
- [38] R. CLIFT, J. R. GRACE et M. E. WEBER. “Bubbles, drops, and particles”. In : (2005) (cf. p. 170).
- [39] T. A. DAVIS, S. RAJAMANICKAM et W. M. SID-LAKHDAR. “A survey of direct methods for sparse linear systems”. In : *Acta Numerica* 25 (2016), p. 383-566 (cf. p. 99).
- [40] G. DAVY, E. REYSSAT, S. VINCENT et S. MIMOUNI. “Euler–Euler simulations of condensing two-phase flows in mini-channel : Combination of a sub-grid approach and an interface capturing approach”. In : *International Journal of Multiphase Flow* 149 (2022), p. 103964 (cf. p. 150).
- [41] R. B. DEBAR. “Eulerian hydrodynamics code for compressible nonviscous flow of several fluids in two-dimensional”. In : (mars 1974) (cf. p. 57).
- [42] O. DESJARDINS et V. MOUREAU. “Methods for multiphase flows with high density ratio By”. In : 2010 (cf. p. 187).
- [43] O. DESJARDINS, V. MOUREAU et H. PITSCHE. “An accurate conservative level set/ghost fluid method for simulating turbulent atomization”. In : *Journal of Computational Physics* 227 (sept. 2008), p. 8395-8416 (cf. p. 31).
- [44] O. DESJARDINS, V. MOUREAU et H. PITSCHE. “An accurate conservative level set/ghost fluid method for simulating turbulent atomization”. In : *Journal of Computational Physics* 227.18 (sept. 2008), p. 8395-8416 (cf. p. 54).
- [45] F. DESMONS et M. COQUERELLE. “A generalized high-order momentum preserving (HOMP) method in the one-fluid model for incompressible two phase flows with high density ratio”. In : *Journal of Computational Physics* 437 (2021), p. 110322 (cf. p. 178, 179).
- [46] B. DEVRIES, J. IANNELLI, C. TREFFTZ, K. A. O’HEARN et G. WOLFFE. “Parallel Implementations of FGMRES for Solving Large, Sparse Non-symmetric Linear Systems”. In : *Procedia Computer Science* 18 (2013). 2013 International Conference on Computational Science, p. 491-500 (cf. p. 99).
- [47] M.-S. DODD et A. FERRANTE. “fast pressure-correction method for incompressible two-fluid flows”. In : *Journal of Computational Physics* 273 (2014), p. 416-434 (cf. p. 77).
- [48] J. J. DONGARRA, I. S. DUFF, D. C. SORESENSEN et H. A. VAN DER VORST. *Numerical linear algebra for high-performance computers*. SIAM, 1998 (cf. p. 150, 157).

- [49] J. J. DONGARRA, I. S. DUFF, D. C. SORESENSEN et H. A. van der VORST. *Numerical Linear Algebra for High-Performance Computers*. Society for Industrial et Applied Mathematics, 1998 (cf. p. 99, 100).
- [50] J. P. V. DOORMAAL et G. D. RAITHY. “ENHANCEMENTS OF THE SIMPLE METHOD FOR PREDICTING INCOMPRESSIBLE FLUID FLOWS”. In : *Numerical Heat Transfer* 7.2 (1984), p. 147-163 (cf. p. 74, 82).
- [51] D. A. DREW et S. L. PASSMAN. *Theory of multicomponent fluids*. T. 135. Springer Science & Business Media, 2006 (cf. p. 151).
- [52] D. A. (A. DREW et S. L. PASSMAN. *Theory of multicomponent fluids*. New York : Springer, 1999 (cf. p. 35, 47, 49).
- [53] C. DULOU et M. BELZONS. “Laboratory study of wave bottom interaction in the bar formation on an erodible sloping bed”. In : *Journal of Geophysical Research* 105 (août 2000), p. 19745-21850 (cf. p. 31).
- [54] M. EL OUAFA, S. VINCENT et V. LE CHENADEC. “Monolithic solvers for incompressible two-phase flows at large density and viscosity ratios”. In : *Fluids* 6.1 (2021), p. 23 (cf. p. 152).
- [55] M. EL OUAFA, S. VINCENT et V. LE CHENADEC. “Navier-stokes solvers for incompressible single- and two-phase flows”. In : *Communications in Computational Physics* 29.4 (2021), p. 1213-1245 (cf. p. 152, 164).
- [56] H. ELMAN, V. HOWLE, J. SHADID, D. SILVESTER et R. TUMINARO. “Least Squares Preconditioners for Stabilized Discretizations of the Navier–Stokes Equations”. In : *SIAM J. Sci. Comput* 30 (jan. 2007), p. 290-311 (cf. p. 115).
- [57] H. ELMAN, V. E. HOWLE, J. SHADID, R. SHUTTLEWORTH et R. TUMINARO. “Block Preconditioners Based on Approximate Commutators”. In : *SIAM Journal on Scientific Computing* 27.5 (2006), p. 1651-1668 (cf. p. 115).
- [58] H. ELMAN et D. SILVESTER. “Fast nonsymmetric iterations and preconditioning for Navier–Stokes equations”. In : *SIAM Journal on Scientific Computing* 17.1 (1996), p. 33-46 (cf. p. 160).
- [59] H. ELMAN et D. SILVESTER. “Fast nonsymmetric iterations and preconditioning for Navier-Stokes equations”. English. In : *SIAM Journal on Scientific Computing* 17.1 (jan. 1996), p. 33-46 (cf. p. 36).
- [60] H. ELMAN et D. SILVESTER. “Fast Nonsymmetric Iterations and Preconditioning for Navier–Stokes Equations”. In : *SIAM Journal on Scientific Computing* 17.1 (1996), p. 33-46 (cf. p. 113).
- [61] H. ELMAN et R. TUMINARO. “Boundary conditions in approximate commutator preconditioners for the Navier-Stokes equations”. In : *Sci. Comput.J. Comput. Appl. Math.SIAM J. Sci. Comput.SIAM J. Sci. Comput* 35 (jan. 2009), p. 257-280 (cf. p. 116).

- [62] H. C. ELMAN. “Preconditioning for the steady-state Navier–Stokes equations with low viscosity”. In : *SIAM Journal on Scientific Computing* 20.4 (1999), p. 1299-1316 (cf. p. 160).
- [63] H. C. ELMAN et R. S. TUMINARO. “Boundary conditions in approximate commutator preconditioners for the Navier-Stokes equations”. In : *Electronic Transactions on Numerical Analysis* 35 (2009), p. 257-280 (cf. p. 161).
- [64] H. C. ELMAN. “Preconditioning for the Steady-State Navier–Stokes Equations with Low Viscosity”. In : *SIAM Journal on Scientific Computing* 20.4 (1999), p. 1299-1316 (cf. p. 36).
- [65] D. ENRIGHT, F. LOSASSO et R. FEDKIW. “A fast and accurate semi-Lagrangian particle level set method”. In : *Computers & Structures* 83.6 (2005). Frontier of Multi-Phase Flow Analysis and Fluid-Structure, p. 479-490 (cf. p. 66, 68).
- [66] J.-L. ESTIVALEZES, W. ANISZEWSKI, F. AUGUSTE, Y. LING, L. OSMAR, J.-P. CALTAGIRONE, L. CHIRCO, A. PEDRONO, S. POPINET, A. BERLEMONT et al. “A phase inversion benchmark for multiscale multiphase flows”. In : *Journal of Computational Physics* 450 (2022), p. 110810 (cf. p. 164, 165).
- [67] J.-L. ESTIVALEZES, W. ANISZEWSKI, F. AUGUSTE, Y. LING, L. OSMAR, J.-P. CALTAGIRONE, L. CHIRCO, A. PEDRONO, S. POPINET, A. BERLEMONT, J. MAGNAUDET, T. MÉNARD, S. VINCENT et S. ZALESKI. “A phase inversion benchmark for multiscale multiphase flows”. In : *Journal of Computational Physics* 450 (2022), p. 110810 (cf. p. 198, 205, 206).
- [68] R. D. FALGOUT et U. M. YANG. “hypre : A library of high performance preconditioners”. In : *International Conference on computational science*. Springer. 2002, p. 632-641 (cf. p. 152, 161).
- [69] R. D. FALGOUT et U. M. YANG. “hypre : A Library of High Performance Preconditioners”. In : *International Conference on Computational Science*. 2002 (cf. p. 37, 116).
- [70] S. FLEAU, S. MIMOUNI, N. MÉRIGOUX et S. VINCENT. “Validation of a multifield approach for the simulations of two-phase flows”. In : *Computational Thermal Sciences : An International Journal* 7.5-6 (2015) (cf. p. 150).
- [71] M. FORTIN et R. GLOWINSKI. “Augmented Lagrangian Methods : Applications to the Numerical Solution of Boundary-Value Problems”. In : *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik* 65.12 (1985), p. 622-622 (cf. p. 36, 74, 80).
- [72] M. FORTIN et R. GLOWINSKI. *Augmented Lagrangian methods : applications to the numerical solution of boundary-value problems*. Elsevier, 2000 (cf. p. 152).
- [73] J. FROMM. “The time dependent flow of an incompressible viscous fluid”. In : *Methods Comput. Phys.* 3 (jan. 1964) (cf. p. 73, 74, 76).

- [74] D. FUSTER. “An energy preserving formulation for the simulation of multiphase turbulent flows”. In : *Journal of Computational Physics* 235 (2013), p. 114-128 (cf. p. [127](#)).
- [75] D. FUSTER, T. ARRUFAT, M. CRIALESI ESPOSITO, Y. LING, L. MALAN, S. PAL, R. SCARDOVELLI, G. TRYGGVASON et S. ZALESKI. *A momentum-conserving, consistent, Volume-of-Fluid method for incompressible flow on staggered grids*. Nov. 2018 (cf. p. [37](#), [178](#), [179](#)).
- [76] G. PIANET, S. VINCENT, J. LEBOI, J.-P. CALTAGIRONE et M. ANDERHUBER. “Simulating compressible gas bubbles with a smooth volume tracking 1-Fluid method”. In : *International Journal of Multiphase Flow* 36 (avr. 2010), p. 273-283 (cf. p. [62](#), [71](#)).
- [77] G. WEYMOUTH et D. YUE. “Conservative Volume-of-Fluid method for free-surface simulations on Cartesian-grids”. In : *J. Comput. Physics* 229 (avr. 2010), p. 2853-2865 (cf. p. [57](#), [213](#)).
- [78] F. GASPAR, C. RODRIGO et E. HEIDENREICH. “Geometric multigrid methods on structured triangular grids for incompressible Navier-Stokes equations at low reynolds numbers”. In : *International Journal of Numerical Analysis and Modeling* 11 (jan. 2014), p. 400-411 (cf. p. [36](#), [111](#)).
- [79] I. GIRAULT, A. CHADIL, E. MASI, S. VINCENT et O. SIMONIN. “A one-fluid formulation for Particle-Resolved Direct Numerical Simulation of gas-solid reactive flow with surface reactions”. In : *to be submitted to international journal of multi phase flows* (), p. 50 (cf. p. [51](#)).
- [80] R. GLOWINSKI, T.-W. PAN et J. PERIAUX. “A Lagrange multiplier/fictitious domain method for the numerical simulation of incompressible viscous flow around moving rigid bodies : (I) case where the rigid body motions are known a priori”. In : *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics* 324.3 (1997), p. 361-369 (cf. p. [79](#)).
- [81] K. GODA. “A Multistep physics Technique with implicit Difference Schemes for Calculating Two- or Three-Dimensional Cavity Flows”. In : *Journal of computational physics* 30 (1979) (cf. p. [76](#), [97](#)).
- [82] S. GOTTLIEB, C.-W. SHU et E. TADMOR. “Strong Stability-Preserving High-Order Time Discretization Methods”. In : *SIAM Review* 43.1 (2001), p. 89-112 (cf. p. [181](#)).
- [83] P. GRAVES-MORRIS. “BiCGStab, VPASab and an adaptation to mildly nonlinear systems”. In : *Journal of Computational and Applied Mathematics* 201.1 (2007), p. 284-299 (cf. p. [99](#)).
- [84] J.-L. GUERMOND et A. SALGADO. “A splitting method for incompressible flows with variable density based on a pressure Poisson equation”. In : *Journal of Computational Physics* 228 (mai 2009), p. 2834-2846 (cf. p. [77](#)).

- [85] J. GUERMOND, P. MINEV et J. SHEN. “An overview of projection methods for incompressible flows”. In : *Computer Methods in Applied Mechanics and Engineering* 195 (sept. 2006), p. 6011-6045 (cf. p. 77).
- [86] J.-L. GUERMOND. “Remarques sur les méthodes de projection pour l’approximation de séquations de Navier–Stokes”. In : *Numerische Mathematik* 67.4 (1994), p. 465-473 (cf. p. 152).
- [87] J.-L. GUERMOND. “Un résultat de convergence d’ordre deux en temps pour l’approximation des équations de Navier-Stokes par une technique de projection incrémentale”. fr. In : *ESAIM : Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique* 33.1 (1999), p. 169-189 (cf. p. 36).
- [88] J.-L. GUERMOND, P. MINEV et J. SHEN. “An overview of projection methods for incompressible flows”. In : *Computer methods in applied mechanics and engineering* 195.44-47 (2006), p. 6011-6045 (cf. p. 152).
- [89] J.-L. GUERMOND et J. SHEN. “Velocity-Correction Projection Methods for Incompressible Flows”. In : *SIAM J. Numerical Analysis* 41 (fév. 2003), p. 112-134 (cf. p. 77).
- [90] R. GUILLAUMENT, S. VINCENT et J.-P. CALTAGIRONE. “An original algorithm for VOF based method to handle wetting effect in multiphase flow simulation”. In : *Mechanics Research Communications* 63 (jan. 2015), p. 26-32 (cf. p. 62, 71).
- [91] F. H. HARLOW et J. E. WELCH. “Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface”. In : *The physics of fluids* 8.12 (1965), p. 2182-2189 (cf. p. 154).
- [92] F. H. HARLOW et J. E. WELCH. “Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface”. In : *Physics of Fluids* 8.12 (déc. 1965), p. 2182-2189 (cf. p. 52).
- [93] M. HERRMANN. “On simulating primary atomization using the refined level set grid method”. English (US). In : *Atomization and Sprays* 21.4 (2011), p. 283-301 (cf. p. 31).
- [94] M. R. HESTENES et E. STIEFEL. “Methods of conjugate gradients for solving linear systems”. In : *Journal of research of the National Bureau of Standards* 49 (1952), p. 409-435 (cf. p. 99, 100).
- [95] C. HIRSCH. *Numerical computation of internal and external flows : The fundamentals of computational fluid dynamics*. Elsevier, 2007 (cf. p. 154).
- [96] C. HIRT et B. NICHOLS. “Volume of fluid (VOF) method for the dynamics of free boundaries”. In : *Journal of Computational Physics* 39.1 (1981), p. 201-225 (cf. p. 35, 54).
- [97] M. HOSEA et L. SHAMPINE. “Efficiency comparisons of methods for integrating ODEs”. In : *Computers & Mathematics with Applications* 28.6 (1994), p. 45-55 (cf. p. 154).

- [98] S. HYSING, S. TUREK, D. KUZMIN, N. PAROLINI, E. BURMAN, S. GANESAN et L. TOBISKA. “Quantitative benchmark computations of two-dimensional bubble dynamics”. In : *International Journal for Numerical Methods in Fluids* 60.11 (2009), p. 1259-1288 (cf. p. [131-133](#)).
- [99] M. ISHII et T. HIBIKI. *Thermo-fluid dynamics of two-phase flow*. New York : Springer-Verlag, 2010 (cf. p. [35](#), [47](#), [49](#), [151](#)).
- [100] R. ISSA, A. GOSMAN et A. WATKINS. “The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme”. In : *Journal of Computational Physics* 62.1 (1986), p. 66-82 (cf. p. [74](#), [82](#)).
- [101] T. JINGXUAN, N. RIBE, X. WU et H. C. SHUM. “Steady and Unsteady Buckling of Viscous Capillary Jets and Liquid Bridges”. In : *Physical Review Letters* 125 (sept. 2020) (cf. p. [31](#)).
- [102] I. KATAOKA. “Local instant formulation of two-phase flow”. In : *International Journal of Multiphase Flow* 12.5 (1986), p. 745-758 (cf. p. [47](#), [70](#)).
- [103] I. KATAOKA. “Local instant formulation of two-phase flow”. In : *International Journal of Multiphase Flow* 12.5 (1986), p. 745-758 (cf. p. [153](#)).
- [104] D. KAY, D. LOGHIN et A. WATHEN. “A Preconditioner for the Steady-State Navier–Stokes Equations”. In : *SIAM Journal on Scientific Computing* 24.1 (2002), p. 237-256 (cf. p. [36](#), [114](#)).
- [105] D. KAY, D. LOGHIN et A. WATHEN. “A preconditioner for the steady-state Navier–Stokes equations”. In : *SIAM Journal on Scientific Computing* 24.1 (2002), p. 237-256 (cf. p. [160](#)).
- [106] K. KLEEFSMAN, G. FEKKEN, A. VELDMAN, B. IWANOWSKI et B. BUCHNER. “A Volume-of-Fluid based simulation method for wave impact problems”. In : *Journal of Computational Physics* 206.1 (2005), p. 363-393 (cf. p. [192](#), [197](#)).
- [107] I. N. KONSHIN, M. A. OLSHANSKII et Y. V. VASSILEVSKI. “ILU Preconditioners for Nonsymmetric Saddle-Point Matrices with Application to the Incompressible Navier–Stokes Equations”. In : *SIAM Journal on Scientific Computing* 37.5 (2015), A2171-A2197 (cf. p. [36](#), [111](#)).
- [108] H. KOTAKEMORI, H. HASEGAWA et A. NISHIDA. “Performance evaluation of a parallel iterative method library using OpenMP”. In : *Eighth International Conference on High-Performance Computing in Asia-Pacific Region (HPCASIA’05)*. 2005, 5 pp.-436 (cf. p. [37](#)).
- [109] H. KOTAKEMORI, H. HASEGAWA et A. NISHIDA. “Performance evaluation of a parallel iterative method library using openmp”. In : *Eighth International Conference on High-Performance Computing in Asia-Pacific Region (HPCASIA’05)*. IEEE. 2005, 5-pp (cf. p. [152](#)).

- [110] E. LABOURASSE, D. LACANETTE, A. TOUTANT, P. LUBIN, S. VINCENT, O. LEBAGUE, J.-P. CALTAGIRONE et P. SAGAUT. “Towards large eddy simulation of isothermal two-phase flows : Governing equations and a priori tests”. In : *International Journal of Multiphase Flow* 33.1 (2007), p. 1-39 (cf. p. 207).
- [111] D. LACANETTE, S. VINCENT, E. ARQUIS et P. GARDIN. “Numerical simulation of gas-jet wiping in steel strip galvanizing process”. In : *ISIJ international* 45.2 (2005), p. 214-220 (cf. p. 150).
- [112] V. LE CHENADEC et H. PITSCH. “A monotonicity preserving conservative sharp interface flow solver for high density ratio two-phase flows”. In : *Journal of Computational Physics* 249 (sept. 2013), p. 185-203 (cf. p. 31, 32, 37, 177, 179, 187).
- [113] A. LEFEBVRE et V. MCDONELL. *Atomization and Sprays*. Mars 2017, p. 1-284 (cf. p. 31).
- [114] R. J. LEVEQUE. “High-resolution conservative algorithms for advection in incompressible flow”. In : *SIAM J. Numer. Anal.* 33 (1996), p. 627-665 (cf. p. 66).
- [115] G. LI, Y. LIAN, Y. GUO, M. JEMISON, M. SUSSMAN, T. HELMS et M. ARIENTI. “Incompressible multiphase flow and encapsulation simulations using the moment-of-fluid method”. In : *International Journal for Numerical Methods in Fluids* 79.9 (2015), p. 456-490 (cf. p. 37).
- [116] J. LI. “Calcul d’Interface Affine par Morceaux”. In : 1995 (cf. p. 202).
- [117] J. LI. “Piecewise linear interface calculation”. French. In : *C. R. Acad. Sci., Paris, Sér. II, Fasc. b* 320.8 (1995), p. 391-396 (cf. p. 57).
- [118] P. L. LIONS. “On the Schwarz Alternating Method III : A Variant for Nonoverlapping Subdomains”. In : *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*. Sous la dir. de T. F. CHAN, R. GLOWINSKI, J. PÉRIAUX et O. B. WIDLUND. Society for Industrial et Applied Mathematics, 1990. Chap. 11, p. 202-223 (cf. p. 37, 99, 104, 111).
- [119] C. LU, X. JIAO et N. MISSIRLIS. “A hybrid geometric + algebraic multigrid method with semi-iterative smoothers”. In : *Numerical Linear Algebra with Applications* 21 (mars 2014) (cf. p. 99, 104).
- [120] P. LUBIN, S. VINCENT, S. ABADIE et J.-P. CALTAGIRONE. “Three-dimensional large eddy simulation of air entrainment under plunging breaking waves”. In : *Coastal engineering* 53.8 (2006), p. 631-655 (cf. p. 150).
- [121] J. MANDEL. “Balancing domain decomposition”. In : *Communications in Numerical Methods in Engineering* 9.3 (1993), p. 233-241 (cf. p. 37, 99, 104, 111).
- [122] K. S. MILLER. “On the inverse of the sum of matrices”. In : *Mathematics magazine* 54.2 (1981), p. 67-72 (cf. p. 159).
- [123] K. S. MILLER. “On the Inverse of the Sum of Matrices”. In : *Mathematics Magazine* 54.2 (1981), p. 67-72 (cf. p. 112, 323).

- [124] K. MUZHINJI, S. SHATEYI et S. MOTSA. “The Mixed Finite Element Multigrid Method for Stokes Equations”. In : *The Scientific World Journal* 2015 (avr. 2015), p. 1-12 (cf. p. [36](#), [111](#)).
- [125] N. NANGIA, B.-E. GRIFFITH, N.-A. PATANKAR et A.-P.-S. BHALLA. “A robust incompressible Navier-Stokes solver for high density ratio multiphase flows”. In : *Journal of Computational Physics* 390 (2019), p. 548-594 (cf. p. [37](#), [178](#), [179](#)).
- [126] N. NANGIA, B. E. GRIFFITH, N. A. PATANKAR et A. P. S. BHALLA. “A robust incompressible Navier-Stokes solver for high density ratio multiphase flows”. In : *Journal of Computational Physics* 390 (2019), p. 548-594 (cf. p. [152](#), [160](#), [179](#), [187](#)).
- [127] M. K. NEZHAD, H. MICHALLET et É. BARTHELEMY. “Equilibre morphologique de barres de déferlement : expériences”. In : 2004 (cf. p. [31](#)).
- [128] X. NICOLAS, S. GOUNAND, M. MEDALE et S. GLOCKNER. “Benchmark Solution for a Three-Dimensional Mixed-Convection Flow, Part 2 : Analysis of Richardson Extrapolation in the Presence of a Singularity”. In : *Numerical Heat Transfer, Part B : Fundamentals* 60.5 (2011), p. 346-369 (cf. p. [116](#)).
- [129] J. NOBREGA, O. CARNEIRO, F. PINHO, G. PAULO, M. TOMÉ, A. CASTELO et J. CUMINATO. “THE PHENOMENON OF JET BUCKLING : EXPERIMENTAL RESULTS AND NUMERICAL PREDICTIONS”. In : (juill. 2008) (cf. p. [31](#), [33](#)).
- [130] W. NOH et P. WOODWARD. “SLIC (Simple Line Interface Calculation)”. In : 59 (1976), p. 330 (cf. p. [56](#)).
- [131] Y. NOTAY. “An aggregation-based algebraic multigrid method”. In : *Electronic Transactions on Numerical Analysis* 37 (mars 2008) (cf. p. [36](#), [99](#), [104](#), [111](#)).
- [132] E. OLSSON et G. KREISS. “A conservative level set method for two phase flow”. In : *Journal of Computational Physics* 210.1 (2005), p. 225-246 (cf. p. [35](#), [53](#)).
- [133] E. OLSSON et G. KREISS. “A conservative level set method for two phase flow”. In : *Journal of Computational Physics* 210 (nov. 2005), p. 225-246 (cf. p. [54](#)).
- [134] E. OLSSON, G. KREISS et S. ZAHEDI. “A conservative level set method for two phase flow II”. In : *Journal of Computational Physics* 225.1 (2007), p. 785-807 (cf. p. [35](#), [53](#)).
- [135] M. OWKES et O. DESJARDINS. “A mass and momentum conserving unsplit semi-Lagrangian framework for simulating multiphase flows”. In : *Journal of Computational Physics* 332 (mars 2017), p. 21-46 (cf. p. [37](#), [177](#), [179](#)).
- [136] A. OZEL, J. B. de MOTTA, M. ABBAS, P. FEDE, O. MASBERNAT, S. VINCENT, J.-L. ESTIVALEZES et O. SIMONIN. “Particle resolved direct numerical simulation of a liquid–solid fluidized bed : comparison with experimental data”. In : *International Journal of Multiphase Flow* 89 (2017), p. 228-240 (cf. p. [150](#)).
- [137] C.-H. B. P. ANGOT et P. FABRIE. “A penalization method to take into account obstacles in viscous flows”. In : *Numerische Mathematik* 81 (fév. 1999), p. 497-520 (cf. p. [71](#), [93](#)).

- [138] S. PAL, D. FUSTER et S. ZALESKI. *A novel momentum-conserving, mass-momentum consistent method for interfacial flows involving large density contrasts*. Jan. 2021 (cf. p. 37, 178).
- [139] S. V. PATANKAR. *Numerical Heat Transfer and Fluid Flow*. Computational Methods in Mechanics and Thermal Sciences. Washington : Hemisphere, 1980 (cf. p. 74, 82).
- [140] S. V. PATANKAR. *Numerical heat transfer and fluid flow*. CRC press, 2018 (cf. p. 154).
- [141] J. K. PATEL et G. NATARAJAN. “A generic framework for design of interface capturing schemes for multi-fluid flows”. In : *Computers Fluids* 106 (2015), p. 108-118 (cf. p. 183).
- [142] J. E. PILLIOD et E. G. PUCKETT. “Second-Order Accurate Volume-of-Fluid Algorithms for Tracking Material Interfaces”. In : *J. Comput. Phys.* 199.2 (sept. 2004), p. 465-502 (cf. p. 57).
- [143] S. POPINET et S. ZALESKI. “A front-tracking algorithm for accurate representation of surface tension”. In : *International Journal for Numerical Methods in Fluids* 30.6 (1999), p. 775-793 (cf. p. 53).
- [144] E. G. PUCKETT et J. S. SALTZMAN. “A 3D adaptive mesh refinement algorithm for multimaterial gas dynamics”. In : *Physica D Nonlinear Phenomena* 60.1-4 (nov. 1992), p. 84-93 (cf. p. 57).
- [145] E. PUCKETT. “A volume-of-fluid interface reconstruction algorithm that is second-order accurate in the max norm”. In : *Communications in Applied Mathematics and Computational Science* 5 (oct. 2010), p. 199-220 (cf. p. 57).
- [146] A. QUARTERONI et A. VALLI. *Domain Decomposition Methods for Partial Differential Equations*. Oxford, UK : Oxford University Press, 1999 (cf. p. 37, 99, 104, 111).
- [147] M. RAESSI et H. PITSCH. “Consistent mass and momentum transport for simulating incompressible interfacial flows with large density ratios using the level set method”. In : *Computers Fluids* 63 (2012), p. 70-81 (cf. p. 37, 178, 179).
- [148] M. REHMAN, C. VUIK et G. SEGAL. “SIMPLE-type preconditioners for the Oseen problem”. In : *Reports of the Department of Applied Mathematical Analysis, 08-09* 61 (oct. 2009) (cf. p. 114).
- [149] R. D. REITZ et F. V. BRACCO. “Mechanism of atomization of a liquid jet”. In : *The Physics of Fluids* 25.10 (1982), p. 1730-1742 (cf. p. 31).
- [150] N. ROUSSEL, P. COUSSOT, G. OVARLEZ, C. FERRARIS, N. MARTYS, S. GAUFFINET-GARRAULT, X. CHATEAU, R. FLATT, I. SCHÖBER, K. KHAYAT, S. GRÜNEWALD, L. N. THRANE, P. BILLBERG et D. FEYS. *Contributor contact details*. Sous la dir. de N. ROUSSEL. Woodhead Publishing Series in Civil and Structural Engineering. Woodhead Publishing, 2012, p. ix-x (cf. p. 31).

- [151] M. RUDMAN. “A volume-tracking method for incompressible multifluid flows with large density variations”. In : *International Journal for Numerical Methods in Fluids* 28.2 (août 1998), p. 357-378 (cf. p. 37).
- [152] M. RUDMAN. “A volume-tracking method for incompressible multifluid flows with large density variations”. In : *International Journal for Numerical Methods in Fluids* 28.2 (1998), p. 357-378 (cf. p. 177, 179).
- [153] J. W. RUGE et K. STÜBEN. “4. Algebraic Multigrid”. In : *Multigrid Methods*, p. 73-130 (cf. p. 99, 104).
- [154] Y. SAAD et M. H. SCHULTZ. “GMRES : A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems”. In : *SIAM Journal on Scientific and Statistical Computing* 7.3 (1986), p. 856-869 (cf. p. 99).
- [155] Y. SAAD. “Iterative Methods for Sparse Linear System”. In : *Studies in Computational Mathematics* 8 (déc. 2001), p. 423-440 (cf. p. 36, 104, 111).
- [156] M. SAEEDIPOUR, S. VINCENT et J.-L. ESTIVALÈZES. “Toward a fully resolved volume of fluid simulation of the phase inversion problem”. In : *Acta Mechanica* 232 (juill. 2021) (cf. p. 164, 165, 167, 199).
- [157] P. SAGAUT et D. DRIKAKIS. “Large Eddy Simulation”. In : *Encyclopedia of Aerospace Engineering*. John Wiley & Sons, Ltd, 2010 (cf. p. 49).
- [158] Y. SATO et B. NIČENO. “A conservative local interface sharpening scheme for the constrained interpolation profile method”. In : *International Journal for Numerical Methods in Fluids* 70.4 (2012), p. 441-467 (cf. p. 61).
- [159] T. SAYADI, S. ZALESKI, S. POPINET, V. LE CHENADEC et S. VINCENT. “A Convergence Study of the One-Fluid Formulation in a Phase Inversion Application at Moderate Reynolds and Weber Numbers”. In : *Turbulence and Interactions*. Sous la dir. de M. DEVILLE, C. CALVIN, V. COUAILLIER, M. DE LA LLAVE PLATA, J.-L. ESTIVALÈZES, T. H. LÊ et S. VINCENT. Cham : Springer International Publishing, 2021, p. 80-90 (cf. p. 199).
- [160] R. SCARDOVELLI et S. ZALESKI. “Interface Reconstruction with Least-Square Fit and Split Eulerian-Lagrangian Advection”. In : *International Journal for Numerical Methods in Fluids* 41 (jan. 2003), p. 251-274 (cf. p. 57).
- [161] R. SCARDOVELLI et S. ZALESKI. “Analytical Relations Connecting Linear Interfaces and Volume Fractions in Rectangular Grids”. In : *Journal of Computational Physics* 164.1 (2000), p. 228-237 (cf. p. 59).
- [162] S. SCHAFFER. “A Semicoarsening Multigrid Method for Elliptic Partial Differential Equations with Highly Discontinuous and Anisotropic Coefficients”. In : *SIAM Journal on Scientific Computing* 20.1 (1998), p. 228-242 (cf. p. 99, 104).
- [163] V. SCHEIFF. “Etude expérimentale et modélisation du transfert de chaleur de l'ébullition transitoire”. Déc. 2018 (cf. p. 34).

- [164] S. SHIN et D. JURIC. “Modeling Three-Dimensional Multiphase Flow Using a Level Contour Reconstruction Method for Front Tracking without Connectivity”. In : *Journal of Computational Physics* 180.2 (2002), p. 427-470 (cf. p. 35).
- [165] C.-W. SHU, W. S. DON, D. GOTTLIEB, O. SCHILLING et L. JAMESON. “Numerical Convergence Study of Nearly Incompressible, Inviscid Taylor–Green Vortex Flow”. In : *J. Sci. Comput.* 24 (juill. 2005), p. 1-27 (cf. p. 127).
- [166] M. SUSSMAN et E.-G. PUCKETT. “A Coupled Level Set and Volume-of-Fluid Method for Computing 3D and Axisymmetric Incompressible Two-Phase Flows”. In : *Journal of Computational Physics* 162.2 (2000), p. 301-337 (cf. p. 35, 54).
- [167] R. TÉMAM. “Sur l’approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (I)”. In : *Archive for Rational Mechanics and Analysis* 32.2 (jan. 1969), p. 135-153 (cf. p. 36, 73).
- [168] M. F. TOMÉ, B. DUFFY et S. MCKEE. “A numerical technique for solving unsteady non-Newtonian free surface flows”. In : *Journal of Non-Newtonian Fluid Mechanics* 62.1 (1996), p. 9-34 (cf. p. 141, 209, 210).
- [169] M. TOMÉ, S. MCKEE, L. BARRATT, D. JARVIS et A. PATRICK. “An experimental and numerical investigation of container filling with viscous liquids”. In : *International Journal for Numerical Methods in Fluids* 31.8 (1999), p. 1333-1353 (cf. p. 31).
- [170] A. TOSELLI et O. WIDLUND. *Domain Decomposition Methods – Algorithms and Theory*. T. 34. Jan. 2005 (cf. p. 37, 99, 104, 111).
- [171] G. TRYGGVASON, B. BUNNER, A. ESMAEELI, D. JURIC, N. AL-RAWAHI, W. TAUBER, J. HAN, S. NAS et Y.-J. JAN. “A Front-Tracking Method for the Computations of Multiphase Flow”. In : *Journal of Computational Physics* 169.2 (2001), p. 708-759 (cf. p. 35, 52).
- [172] G. TRYGGVASON, R. SCARDOVELLI et S. ZALESKI. *Direct Numerical Simulations of Gas–Liquid Multiphase Flows*. Cambridge University Press, 2011 (cf. p. 35, 151).
- [173] S. VAJDA. “Studies in Linear and Non-Linear Programming. (Stanford Mathematical Studies in the Social Sciences.) By K. J. Arrow, L. Hurwicz, and H. Uzawa. Pp. 229. 60s. 1958. (Stanford Univ. Press)”. In : *The Mathematical Gazette* 44 (1960), p. 151-151 (cf. p. 80).
- [174] H. A. VAN DER VORST et C. VUIK. “GMRESR : a family of nested GMRES methods”. In : *Numerical Linear Algebra with Applications* 1.4 (1994), p. 369-386 (cf. p. 99).
- [175] P. VANEK, J. MANDEL et M. BREZINA. “Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems”. In : *Computing* 56 (2005), p. 179-196 (cf. p. 99, 104).

- [176] G. VAUDOR, T. MÉNARD, W. ANISZEWSKI, M. DORING et A. BERLEMONT. “A consistent mass and momentum flux computation method for two phase flows. Application to atomization process”. In : *Computers & Fluids* 152 (avr. 2017) (cf. p. 37).
- [177] G. VAUDOR, T. MÉNARD, W. ANISZEWSKI, M. DORING et A. BERLEMONT. “A consistent mass and momentum flux computation method for two phase flows. Application to atomization process”. In : *Computers & Fluids* 152 (avr. 2017) (cf. p. 178, 179).
- [178] S. VINCENT, J.-P. CALTAGIRONE et P. BONNETON. “Numerical modelling of bore propagation and run-up on sloping beaches using a MacCormack TVD scheme”. In : *Journal of Hydraulic Research* 39.1 (2001), p. 41-49 (cf. p. 193).
- [179] S. VINCENT et J.-P. CALTAGIRONEL. “Efficient solving method for unsteady incompressible interfacial flow problems”. In : *International Journal for Numerical Methods in Fluids* 30.6 (1999), p. 795-811 (cf. p. 142).
- [180] S. VINCENT, A. SARTHOU, J. CALTAGIRONE, F. SONILHAC, P. FÉVRIER, C. MIGNOT et G. PIANET. “Augmented Lagrangian and penalty methods for the simulation of two-phase flows interacting with moving solids. Application to hydroplaning flows interacting with real tire tread patterns”. English. In : *Journal of Computational Physics* 230.4 (2011), p. 956-983 (cf. p. 36, 74, 80, 97).
- [181] S. VINCENT, M. TAVARES, S. FLEAU, S. MIMOUNI, M. OULD-ROUISS et J.-L. ESTIVALEZES. “A priori filtering and LES modeling of turbulent two-phase flows application to phase separation”. In : *Computers & Fluids* 176 (2018), p. 245-259 (cf. p. 198).
- [182] S. VINCENT. “Contribution à la modélisation et à la simulation numérique d’écoulements diphasiques de fluides non miscibles”. Thèse de doct. Habilitation à Diriger des Recherches de l’Université Bordeaux 1, 2010 (cf. p. 157).
- [183] S. VINCENT, J.-P. CALTAGIRONE et D. JAMET. “TEST-CASE NO 15 : PHASE INVERSION IN A CLOSED BOX (PC)”. In : *Multiphase Science and Technology* 16.1-3 (2004), p. 101-104 (cf. p. 197).
- [184] S. VINCENT, G. BALMIGERE, C. CARUYER, E. MEILLOT et J.-P. CALTAGIRONE. “Contribution to the modeling of the interaction between a plasma flow and a liquid jet”. In : *Surface and Coatings Technology* 203.15 (2009), p. 2162-2171 (cf. p. 150).
- [185] S. VINCENT, J.-P. CALTAGIRONE, P. LUBIN et T. N. RANDRIANARIVELO. “An adaptive augmented Lagrangian method for three-dimensional multimaterial flows”. In : *Computers & Fluids* 33 (2004), p. 1273-1289 (cf. p. 36, 74, 80, 97).
- [186] S. VINCENT, J.-P. CALTAGIRONE, P. LUBIN et T. N. RANDRIANARIVELO. “An adaptive augmented Lagrangian method for three-dimensional multimaterial flows”. In : *Computers & Fluids* 33.10 (2004), p. 1273-1289 (cf. p. 152).

- [187] S. VINCENT, A. SARTHOU, J.-P. CALTAGIRONE, F. SONILHAC, P. FÉVRIER, C. MIGNOT et G. PIANET. “Augmented Lagrangian and penalty methods for the simulation of two-phase flows interacting with moving solids. Application to hydroplaning flows interacting with real tire tread patterns”. In : *Journal of computational physics* 230.4 (2011), p. 956-983 (cf. p. [150](#), [152](#)).
- [188] M. WABRO. “Coupled algebraic multigrid methods for the Oseen problem”. In : *Computing and Visualization in Science* 7 (oct. 2004), p. 141-151 (cf. p. [36](#), [111](#)).
- [189] A. J. WATHEN. “Preconditioning”. In : *Acta Numerica* 24 (2015), p. 329-376 (cf. p. [102](#)).
- [190] A. WATHEN et D. SILVESTER. “Fast Iterative Solution of Stabilised Stokes Systems. Part I : Using Simple Diagonal Preconditioners”. In : *SIAM Journal on Numerical Analysis* 30.3 (1993), p. 630-649 (cf. p. [110](#)).
- [191] R. WEBSTER. “An algebraic multigrid solver for Navier-Stokes problems”. In : *International Journal for Numerical Methods in Fluids* 18.8 (1994), p. 761-780 (cf. p. [36](#), [111](#)).
- [192] P. WESSELING et C. OOSTERLEE. “Geometric multigrid with applications to computational fluid dynamics”. In : *Journal of Computational and Applied Mathematics* 128.1 (2001). Numerical Analysis 2000. Vol. VII : Partial Differential Equations, p. 311-334 (cf. p. [36](#), [111](#)).
- [193] G. D. WEYMOUTH et D. K.-P. YUE. “Conservative volume-of-fluid method for free-surface simulations on cartesian-grids”. In : *Journal of Computational Physics* 229.8 (2010), p. 2853-2865 (cf. p. [150](#), [153](#), [165](#)).
- [194] S. Ø. WILLE, Ø. STAFF et A. F. D. LOULA. “Parallel ILU preconditioning, a priori pivoting and segregation of variables for iterative solution of the mixed finite element formulation of the Navier–Stokes equations”. In : *International Journal for Numerical Methods in Fluids* 41.9 (2003), p. 977-996 (cf. p. [36](#), [111](#)).
- [195] L. XU, W. W. ZHANG et S. R. NAGEL. “Drop Splashing on a Dry Smooth Surface”. In : *Phys. Rev. Lett.* 94 (18 mai 2005), p. 184505 (cf. p. [31](#)).
- [196] X. XU, J. OUYANG, T. JIANG et Q. LI. “Numerical simulation of 3D-unsteady viscoelastic free surface flows by improved smoothed particle hydrodynamics method”. In : *Journal of Non-Newtonian Fluid Mechanics* s 177–178 (juin 2012), p. 109-120 (cf. p. [142](#)).
- [197] A. L. YARIN et D. A. WEISS. “Impact of drops on solid surfaces : self-similar capillary waves, and splashing as a new type of kinematic discontinuity”. In : *Journal of Fluid Mechanics* 283 (1995), p. 141-173 (cf. p. [31](#)).
- [198] A. YARIN. “DROP IMPACT DYNAMICS : Splashing, Spreading, Receding, Bouncing. . .” In : *Annual Review of Fluid Mechanics* 38.1 (2006), p. 159-192 (cf. p. [31](#)).
- [199] D. YOUNGS. “Time-Dependent Multi-material Flow with Large Fluid Distortion”. In : *Num. Method Fluid Dyn.* 24 (jan. 1982), p. 273-285 (cf. p. [57](#)).

- [200] J. ZHU et J. QIU. “A new fifth order finite difference WENO scheme for solving hyperbolic conservation laws”. In : *Journal of Computational Physics* 318 (mai 2016) (cf. p. [183](#)).
- [201] D. ZUZIO, A. ORAZZO, J.-L. ESTIVALÈZES et I. LAGRANGE. “A new efficient momentum preserving Level-Set/VOF method for high density and momentum ratio incompressible two-phase flows”. In : *Journal of Computational Physics* 410 (2020), p. 109342 (cf. p. [37](#), [178](#)).
- [202] D. ZUZIO, A. ORAZZO, J.-L. ESTIVALÈZES et I. LAGRANGE. “A new efficient momentum preserving Level-Set/VOF method for high density and momentum ratio incompressible two-phase flows”. In : *Journal of Computational Physics* 410 (fév. 2020), p. 109342 (cf. p. [179](#), [187](#)).

Annexe A

Cas B de l'inversion de phase
liquide/liquide

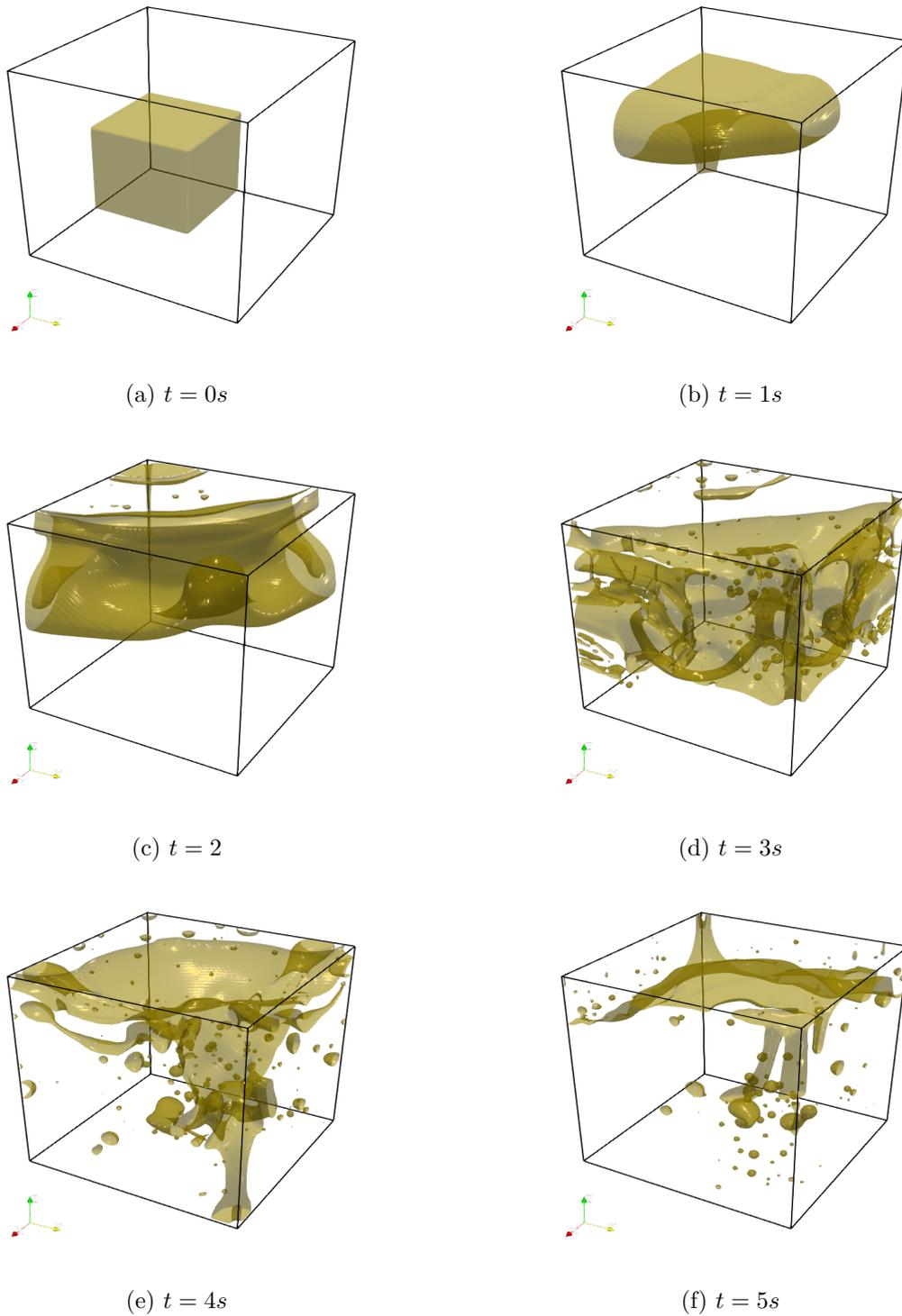


FIGURE A.1 – Simulation numérique de l'inversion de phase liquide/liquide. L'évolution temporelle de l'interface des 2 fluides est représentée en jaune. Paramètres de calcul : un maillage $256 \times 256 \times 256$, une $CFL = 0.5$ et un résidu $\epsilon = 10^{-5}$ du solveur BiCGStab(2).

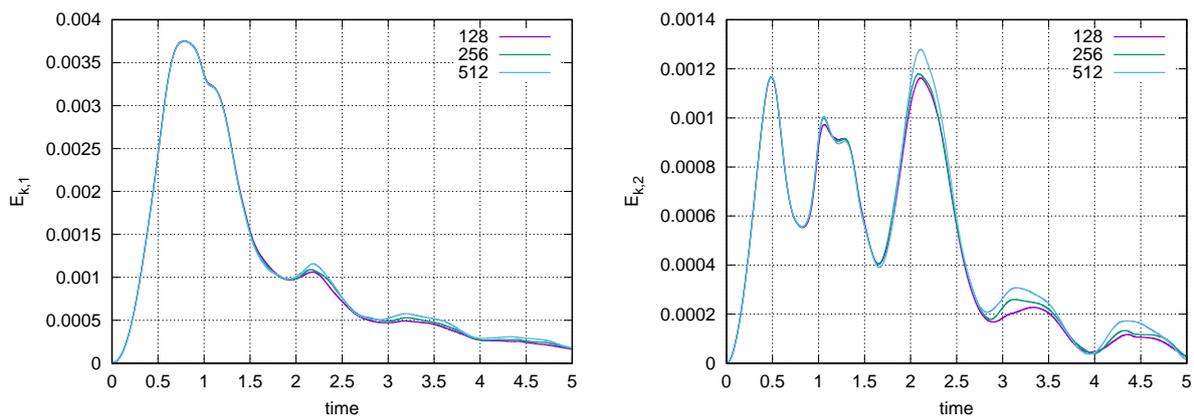


FIGURE A.2 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : $E_{k,1}$, énergie cinétique dans le fluide lourd. (b) : $E_{k,2}$, énergie cinétique dans le fluide léger.

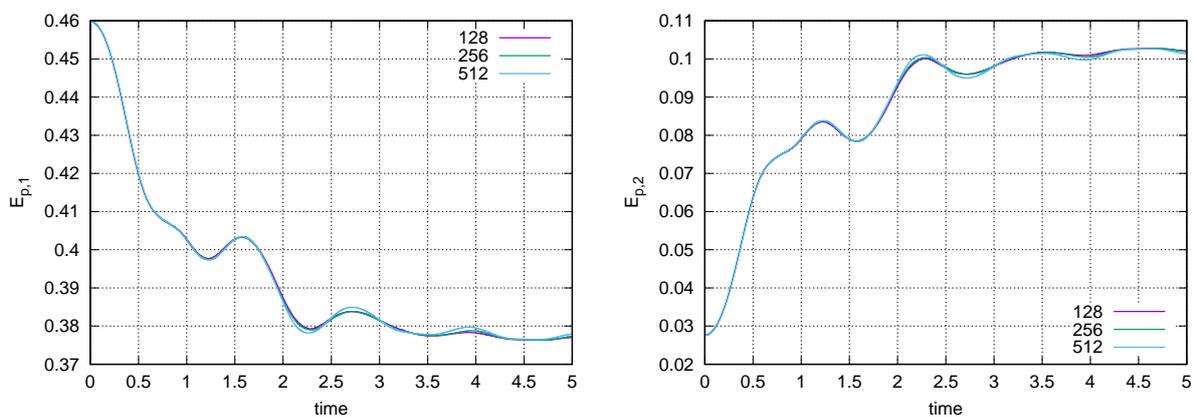


FIGURE A.3 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : $E_{p,1}$, énergie potentielle dans le fluide lourd. (b) : $E_{p,2}$, énergie potentielle dans le fluide léger.

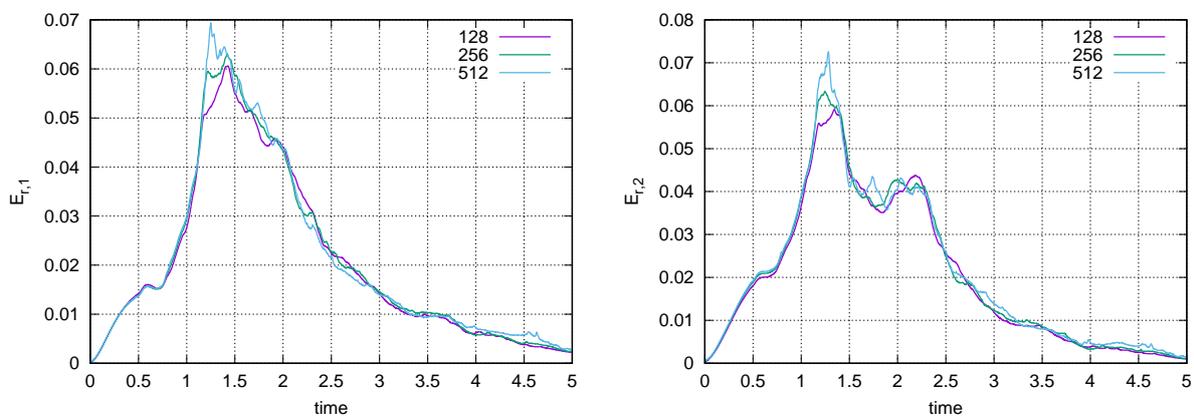


FIGURE A.4 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : $E_{r,1}$, entropie dans le fluide lourd. (b) : $E_{r,2}$, entropie dans le fluide léger.

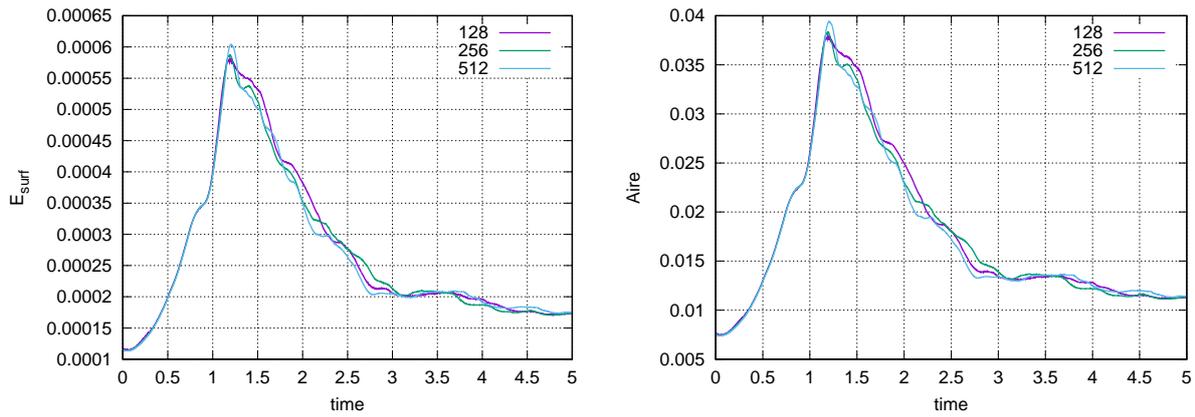


FIGURE A.5 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : L'aire interfaciale $Aire$. (b) : L'énergie surfacique E_{surf} .

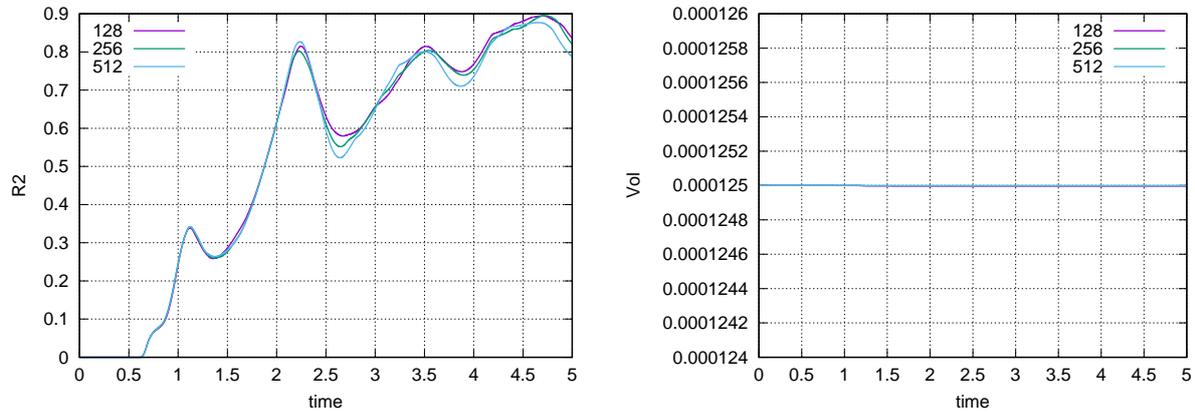


FIGURE A.6 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : le ration R2, (b) : le volume de fluide léger Vol.

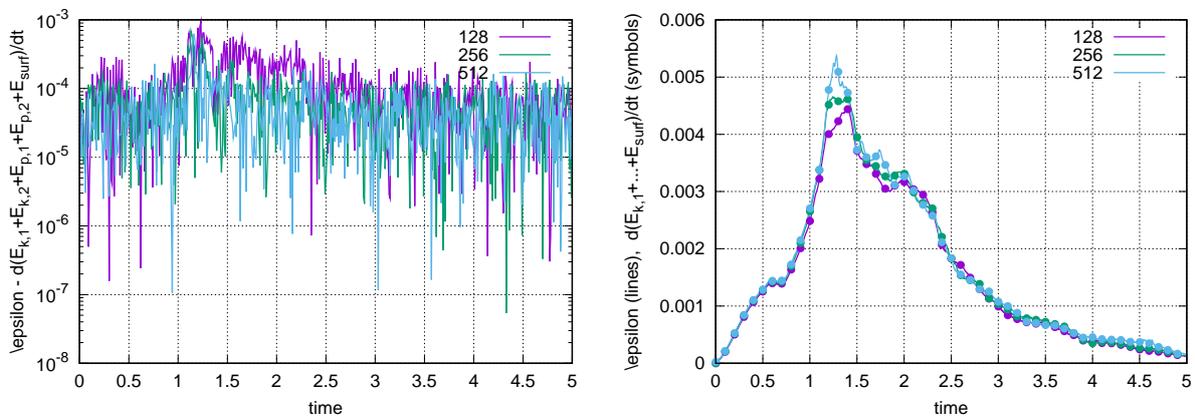


FIGURE A.7 – Simulation numérique de l'inversion de phase liquide/liquide. (a) : bilan énergétique. (b) : comparaison entre la dissipation et la variation temporelle de l'énergie totale.

Annexe B

Physical boundary conditions

To complete the one-fluid formulation, the boundary conditions of the fluid domain are imposed via the penalty term $\bar{\bar{B}} \cdot (f(\bar{u}) - \bar{u}_\infty)$ on $\Gamma = \partial\Omega$. Boundary conditions such as Dirichlet or Neumann can be enforced on the domain borders. Since the physical boundaries of the fluid domain coincide with the nodes of the scalar meshes, it is no longer the variable \bar{u}^{n+1} that has to be penalised, but a function $f(\bar{u}^{n+1})$ of this variable. where

$$\bar{\bar{B}} = \begin{bmatrix} \alpha_u & 0 & 0 \\ 0 & \alpha_v & 0 \\ 0 & 0 & \alpha_w \end{bmatrix} \quad (\text{B.1})$$

is a tensor field whose diagonal components tend to infinity along the boundary Γ and are identically zero inside the fluid domain Ω . Here, $f(\bar{u}^{n+1})$ is a discrete function of u^{n+1} , v^{n+1} and w^{n+1} , which is written as a linear combination of resolved velocities $u_{i,j,k}^{n+1}$, $v_{i,j,k}^{n+1}$, $w_{i,j,k}^{n+1}$ and their neighbors :

$$f(u) = a_0 u_{i,j,k}^{n+1} + a_1 u_{i-1,j,k}^{n+1} + a_2 u_{i+1,j,k}^{n+1} + a_3 u_{i,j-1,k}^{n+1} + a_4 u_{i,j+1,k}^{n+1} + a_5 u_{i,j,k-1}^{n+1} + a_6 u_{i,j,k+1}^{n+1} \quad (\text{B.2})$$

$$f(v) = a_0 v_{i,j,k}^{n+1} + a_1 v_{i-1,j,k}^{n+1} + a_2 v_{i+1,j,k}^{n+1} + a_3 v_{i,j-1,k}^{n+1} + a_4 v_{i,j+1,k}^{n+1} + a_5 v_{i,j,k-1}^{n+1} + a_6 v_{i,j,k+1}^{n+1}, \quad (\text{B.3})$$

$$f(w) = a_0 w_{i,j,k}^{n+1} + a_1 w_{i-1,j,k}^{n+1} + a_2 w_{i+1,j,k}^{n+1} + a_3 w_{i,j-1,k}^{n+1} + a_4 w_{i,j+1,k}^{n+1} + a_5 w_{i,j,k-1}^{n+1} + a_6 w_{i,j,k+1}^{n+1}. \quad (\text{B.4})$$

The treatment of the Dirichlet boundary conditions, applied to the left boundary for example is controlled by the coefficients a_i which are then set to the following values : $a_0 = a_2 = \frac{1}{2}$ and $a_1 = a_3 = a_4 = a_5 = a_6 = 0$. Thus, the discrete momentum equation on the left boundary becomes :

$$NS(u_{i,j,k}^{n+1}) + \alpha_u \left(\frac{1}{2} u_{i,j,k}^{n+1} + \frac{1}{2} u_{i+1,j,k}^{n+1} - u_\infty \right) = 0. \quad (\text{B.5})$$

Annexe C

Discretization with finite volumes in Fugu code

Convective term

One recalls the convective term discretized to first order in time, and integrated over the control volume Ω , bounded by the surface Σ :

$$\int_{\Omega} \nabla \cdot (\rho \bar{u} \otimes \bar{u}) dV = \int_{\Sigma} \rho u_{\ell} u_m n_m dS \approx \int_{\Sigma} \rho^{n+1} u_m^n u_{\ell}^{n+1} n_m dS \quad (\text{C.1})$$

The discretization on the 3D staggered mesh of the convective term is performed with a centered scheme in a semi-implicit way. It is written for the velocity component u :

$$\begin{aligned} (\text{C.1}) \Rightarrow & \rho_{i,j,k}^{n+1} \left(\frac{u_{i,j,k}^n + u_{i+1,j,k}^n}{2} \frac{u_{i,j,k}^{n+1} + u_{i+1,j,k}^{n+1}}{2} - \frac{u_{i-1,j,k}^n + u_{i,j,k}^n}{2} \frac{u_{i-1,j,k}^{n+1} + u_{i,j,k}^{n+1}}{2} \right) \Delta y \Delta z \\ & + \rho_{i,j,k}^{n+1} \left(\frac{v_{i,j,k}^n + v_{i,j+1,k}^n}{2} \frac{u_{i,j,k}^{n+1} + u_{i,j+1,k}^{n+1}}{2} - \frac{v_{i,j-1,k}^n + v_{i,j,k}^n}{2} \frac{u_{i,j-1,k}^{n+1} + u_{i,j,k}^{n+1}}{2} \right) \Delta x \Delta z \\ & + \rho_{i,j,k}^{n+1} \left(\frac{w_{i,j,k}^n + w_{i,j,k+1}^n}{2} \frac{u_{i,j,k}^{n+1} + u_{i,j,k+1}^{n+1}}{2} - \frac{w_{i,j,k-1}^n + w_{i,j,k}^n}{2} \frac{u_{i,j,k-1}^{n+1} + u_{i,j,k}^{n+1}}{2} \right) \Delta x \Delta y \end{aligned} \quad (\text{C.2})$$

with Δx , Δy and Δz the sizes of the parallelepipedic control volume in the x -, y - and z -direction, respectively.

Viscous terms

The viscous stress tensor is decomposed as $T_{\ell m} = \kappa \Lambda_{\ell m} + \zeta \Xi_{\ell m} - \eta \Gamma_{\ell m}$ where Λ , Ξ and Γ pseudo-tensors are defined in CALTAGIRONE et al. [33]. The coefficient κ , ζ and η are elongational, shear and rotational viscosities. They are linked to the dynamical viscosity by $\kappa = \zeta = 2\mu$ and $\eta = \mu$. Pseud-tensors Λ , Ξ and Γ respectively read

$$\Lambda = \begin{pmatrix} \frac{\partial u}{\partial x} & 0 & 0 \\ 0 & \frac{\partial v}{\partial y} & 0 \\ 0 & 0 & \frac{\partial w}{\partial z} \end{pmatrix}, \quad \Xi = \begin{pmatrix} 0 & \frac{\partial u}{\partial y} & \frac{\partial u}{\partial z} \\ \frac{\partial v}{\partial x} & 0 & \frac{\partial v}{\partial z} \\ \frac{\partial w}{\partial x} & \frac{\partial w}{\partial y} & 0 \end{pmatrix}, \quad \Gamma = \begin{pmatrix} 0 & \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x} & \frac{\partial u}{\partial z} - \frac{\partial w}{\partial x} \\ \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} & 0 & \frac{\partial v}{\partial z} - \frac{\partial w}{\partial y} \\ \frac{\partial w}{\partial x} - \frac{\partial u}{\partial z} & \frac{\partial w}{\partial y} - \frac{\partial v}{\partial z} & 0 \end{pmatrix} \quad (\text{C.3})$$

The viscous operator is approximated implicitly using standard second-order, centered finite-differences. It is written explicitly for the u -velocity component :

C.0.0.1 Elongation viscosity $(\kappa\Lambda_{\ell m})_{,m}$, u -component

$$\int \frac{\partial}{\partial x} \left(\kappa \frac{\partial u}{\partial x} \right) dV \approx \left(\kappa_{i+\frac{1}{2},j,k} \frac{u_{i+1,j,k}^{n+1} - u_{i,j,k}^{n+1}}{\Delta x} - \kappa_{i-\frac{1}{2},j,k} \frac{u_{i,j,k}^{n+1} - u_{i-1,j,k}^{n+1}}{\Delta x} \right) \Delta y \Delta z \quad (\text{C.4})$$

Shear viscosity $(\zeta\Xi_{\ell m})_{,m}$, u -component

$$\begin{aligned} \int \frac{\partial}{\partial y} \left(\zeta \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial z} \left(\zeta \frac{\partial u}{\partial z} \right) dV \approx & \left(\zeta_{i,j+\frac{1}{2},k} \frac{u_{i,j+1,k}^{n+1} - u_{i,j,k}^{n+1}}{\Delta y} - \zeta_{i,j-\frac{1}{2},k} \frac{u_{i,j,k}^{n+1} - u_{i,j-1,k}^{n+1}}{\Delta y} \right) \Delta x \Delta z \\ & + \left(\zeta_{i,j,k+\frac{1}{2}} \frac{u_{i,j,k+1}^{n+1} - u_{i,j,k}^{n+1}}{\Delta z} - \zeta_{i,j,k-\frac{1}{2}} \frac{u_{i,j,k}^{n+1} - u_{i,j,k-1}^{n+1}}{\Delta z} \right) \Delta x \Delta y \end{aligned} \quad (\text{C.5})$$

Rotational viscosity $(-\eta\Gamma_{\ell m})_{,m}$, u -component

$$\begin{aligned} \int \frac{\partial}{\partial y} \left(-\eta \frac{\partial u}{\partial y} + \eta \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial z} \left(-\eta \frac{\partial u}{\partial z} + \eta \frac{\partial w}{\partial x} \right) dV \\ \approx \left(-\eta_{i,j+\frac{1}{2},k} \frac{u_{i,j+1,k}^{n+1} - u_{i,j,k}^{n+1}}{\Delta y} + \eta_{i-\frac{1}{2},j+1,k} \frac{v_{i,j+1,k}^{n+1} - v_{i-1,j+1,k}^{n+1}}{\Delta x} \right) \Delta x \Delta z \\ + \left(\eta_{i,j-\frac{1}{2},k} \frac{u_{i,j,k}^{n+1} - u_{i,j-1,k}^{n+1}}{\Delta y} - \eta_{i-\frac{1}{2},j,k} \frac{v_{i,j,k}^{n+1} - v_{i-1,j,k}^{n+1}}{\Delta x} \right) \Delta x \Delta z \\ + \left(-\eta_{i,j,k+\frac{1}{2}} \frac{u_{i,j,k+1}^{n+1} - u_{i,j,k}^{n+1}}{\Delta z} + \eta_{i-\frac{1}{2},j,k+1} \frac{w_{i,j,k+1}^{n+1} - w_{i-1,j,k+1}^{n+1}}{\Delta x} \right) \Delta x \Delta y \\ + \left(\eta_{i,j,k-\frac{1}{2}} \frac{u_{i,j,k}^{n+1} - u_{i,j,k-1}^{n+1}}{\Delta z} - \eta_{i-\frac{1}{2},j,k} \frac{w_{i,j,k}^{n+1} - w_{i-1,j,k}^{n+1}}{\Delta x} \right) \Delta x \Delta y \end{aligned} \quad (\text{C.6})$$

Pressure Gradient ∇p , u -component

For the horizontal velocity component u , the pressure gradient is discretized as follows :

$$\int \frac{\partial p}{\partial x} dV \approx (p_{i+\frac{1}{2},j,k} - p_{i-\frac{1}{2},j,k}) \Delta x \Delta y \Delta z \quad (\text{C.7})$$

Divergence $\nabla \cdot \bar{u}^{n+1}$

The divergence $\nabla \cdot \bar{u}$ of the velocity field $\bar{u}^{n+1} \approx (u^{n+1}, v^{n+1}, w^{n+1})^T$ is approximated at cell centers by

$$\int \nabla \cdot \bar{u}^{n+1} dV \approx \left(\frac{u_{i+\frac{1}{2},j,k}^{n+1} - u_{i-\frac{1}{2},j,k}^{n+1}}{\Delta x} + \frac{v_{i,j+\frac{1}{2},k}^{n+1} - v_{i,j-\frac{1}{2},k}^{n+1}}{\Delta y} + \frac{w_{i,j,k+\frac{1}{2}}^{n+1} - w_{i,j,k-\frac{1}{2}}^{n+1}}{\Delta z} \right) \Delta x \Delta y \Delta z \quad (\text{C.8})$$

Annexe D

Iterative solver general structure

Algorithm 1 Solving strategy with BiCGStab(2) solver assuming preconditioner \mathcal{P} is known.

```

1: procedure INITIALISATION
2:    $k = 0$  ▷  $k \in \mathbb{N}^+$ , iteration number
3:    $\bar{b} = \mathcal{P}^{-1}\bar{b}$  ▷ Precond is applied on r.h.s.
4:    $\bar{z} = A\bar{x}^{(0)}$  ▷  $\bar{x}^{(0)} \in \mathbb{R}$  is an initial guess
5:    $\bar{q} = \mathcal{P}^{-1}\bar{z}$ 
6:    $\bar{r}^{(0)} = \bar{b} - \bar{q}$  ▷ Initial residual of the preconditioned system
7:    $\tilde{r} = \bar{r}^{(0)}$  ▷  $\tilde{r}$  is an arbitrary vector, such that  $\tilde{r} \cdot \bar{r}^{(0)} \neq 0$ 
8:    $\xi_0 = \omega_2 = 1; \alpha = 0; \bar{u} = \bar{0}$ 
9: end procedure
10: while  $\frac{\|\bar{r}^{(k)}\|_2}{\|\bar{r}^{(0)}\|_2} > \varepsilon$  do
11:    $k = k + 1$ 
12:    $\xi_0 = -\omega_2\xi_0$ 
      EVEN STEP
13:    $\xi_1 = \tilde{r} \cdot \bar{r}^{(k-1)}; \beta = \alpha\xi_1/\xi_0; \xi_0 = \xi_1$  ▷ dot product #1
14:    $\bar{z} = A\bar{u}$  ▷ matrix-vector #1
15:    $\bar{v} = \mathcal{P}^{-1}\bar{z}$  ▷ preconditionner #1
16:    $\gamma = \tilde{r} \cdot \bar{v}; \alpha = \xi/\gamma$  ▷ dot product #2
17:    $\bar{r}^{(k)} = \bar{r}^{(k-1)} - \alpha\bar{v}$  ▷ linear combination #1
18:    $\bar{z} = A\bar{r}^{(k)}$  ▷ matrix-vector #2
19:    $\bar{s} = \mathcal{P}^{-1}\bar{z}$  ▷ preconditionner #2
20:    $\bar{x}^{(k)} = \bar{x}^{(k-1)} + \alpha\bar{u}$  ▷ linear combination #2
      ODD STEP
21:    $\xi_1 = \tilde{r} \cdot \bar{s}; \beta = \alpha\xi_1/\xi_0$  ▷ dot product #3
22:    $\bar{v} = \bar{s} - \beta\bar{v}$  ▷ linear combination #3
23:    $\bar{z} = A\bar{v}$  ▷ matrix-vector #3
24:    $\bar{w} = \mathcal{P}^{-1}\bar{z}$  ▷ preconditionner #3
25:    $\gamma = \tilde{r} \cdot \bar{w}; \alpha = \xi_0/\gamma$  ▷ dot product #4
26:    $\bar{u} = \bar{r}^{(k)} - \beta\bar{u}$  ▷ linear combination #4
27:    $\bar{r}^{(k)} = \bar{r}^{(k)} - \alpha\bar{v}$  ▷ #5
28:    $\bar{s} = \bar{s} + \alpha\bar{w}$  ▷ #6
29:    $\bar{z} = A\bar{s}$  ▷ matrix-vector #4
30:    $\bar{t} = \mathcal{P}^{-1}\bar{z}$  ▷ preconditionner #4
      GENERALIZED CONJUGATE GRADIENT PART
31:    $\omega_1 = \bar{r}^{(k)} \cdot \bar{s}$  ▷ dot product #5
32:    $\lambda = \bar{s} \cdot \bar{s}$  ▷ #6
33:    $\nu = \bar{s} \cdot \bar{t}$  ▷ #7
34:    $\tau = \bar{t} \cdot \bar{t}$  ▷ #8
35:    $\omega_2 = \bar{r}^{(k)} \cdot \bar{t}$  ▷ #9
36:    $\tau = \tau - \nu^2/\lambda; \omega_2 = (\omega_2 - \nu\omega_1/\lambda)/\tau; \omega_1 = (\omega_1 - \nu\omega_2)/\lambda$ 
37:    $\bar{x}^{(k)} = \bar{x}^{(k)} + \omega_1\bar{r}^{(k)} + \omega_2\bar{s} + \alpha\bar{u}$  ▷ linear combination #7
38:    $\bar{r}^{(k)} = \bar{r}^{(k)} - \omega_1\bar{s} - \omega_2\bar{t}$  ▷ #8
39:    $\bar{u} = \bar{u} - \omega_1\bar{v} - \omega_2\bar{w}$  ▷ #9
40: end while

```

Annexe E

Velocity and pressure preconditioning

Velocity preconditioning

Algorithm 1 Block triangular block preconditionner for the velocity block. Approximation of the inverse of the velocity block \tilde{F}_u on a vector $(z_u, z_v, z_w)^T$, the solution vector is $(r_u, r_v, r_w)^T$

- 1: $F_{ww}(I - G)\bar{r}_w = \bar{z}_w$ ▷ Solve
 - 2: $\bar{r}_{uv} \leftarrow \bar{r}_{uv} - A_{uvw}\bar{z}_w$ ▷ Update
 - 3: $A_{uv}\bar{r}_{uv} = \bar{z}_{uv}$ with $\bar{z}_{uv} = (z_u, z_v)^T$ ▷ Solve
-

Pressure preconditioning

Algorithm 2 Representation of the application of the block triangular preconditioner. $M_p^{(1/\mu)}$ is the diagonal pressure mass matrix scaled by the inverse of the viscosity $1/\mu$, $M_p^{(1)}$ is the diagonal standard pressure mass matrix, $N_p^{(1)}$ represents the standard convective matrix in the pressure space, $A_p^{(1/\rho)}$ is the scaled Laplacian, which corresponds to the discretisation of the term $\nabla \cdot (\rho^{-1} \nabla \phi)$, and F is the convection-diffusion-reaction operator for the momentum equation.

$$\text{SOLVE } \begin{bmatrix} F_{uu} & F_{uv} & F_{uw} & B_{pu}^T \\ 0 & F_{vv} & F_{vw} & B_{pv}^T \\ 0 & 0 & F_{ww} & B_{pw}^T \\ 0 & 0 & 0 & \hat{S}_{PCD} \end{bmatrix} \begin{pmatrix} \bar{r}_u \\ \bar{r}_v \\ \bar{r}_w \\ \bar{r}_p \end{pmatrix} = \begin{pmatrix} \bar{z}_u \\ \bar{z}_v \\ \bar{z}_w \\ \bar{z}_p \end{pmatrix}$$

APPROXIMATE THE INVERSE OF SCHUR \hat{S}_{PCD}

- 1: $\bar{r}_{p1} = (M_p^{(1)})^{-1} \bar{z}_p$ ▷ $2 \times$ mass matrix solve $(M_p^{(1/\mu)})^{-1}$ and $(M_p^{(1)})^{-1}$
- 2: $\bar{r}_{p2} = (M_p^{(1/\mu)})^{-1} \bar{z}_p$
- 3: $\bar{r}_{p3} = F_P^{(1)} \bar{r}_{p2}$ ▷ Matrix-vector product for the operator $F_P^{(1)}$
- 4: $\bar{r}_{p4} = (A_P^{(1/\rho)})^{-1} \bar{z}_{p3}$ ▷ Solve Laplace operator $A_P^{(1/\rho)}$
- 5: $\bar{r}_p = \bar{r}_{p4} + \bar{r}_{p1}$

UPDATE \bar{z}_u, \bar{z}_v AND \bar{z}_w

- 6: $\bar{z}_u = \bar{z}_u - B_{pu}^T \bar{r}_p$
- 7: $\bar{z}_v = \bar{z}_v - B_{pv}^T \bar{r}_p$
- 8: $\bar{z}_w = \bar{z}_w - B_{pw}^T \bar{r}_p$

APPROXIMATE THE VELOCITY BLOCK \hat{F}_u

- 9: $\bar{r}_w = F_{ww}^{-1} \bar{z}_w$ ▷ Matrix solve F_{ww}
 - 10: $\hat{\bar{z}}_{uv} = F_{uw} \bar{r}_w + F_{vw} \bar{r}_w$ ▷ Matrix-vector product for the operator F_{uw} and F_{vw}
 - 11: $\bar{z}_{uv} = \bar{z}_{uv} - \hat{\bar{z}}_{uv}$ ▷ Update \bar{r}_{uv}
 - 12: $\bar{r}_v = F_{vv}^{-1} \bar{z}_v$ ▷ Matrix solve F_{vv}
 - 13: $\hat{\bar{z}}_u = F_{uv} \bar{r}_v$ ▷ Matrix-vector product for the operator F_{uv}
 - 14: $\bar{z}_u = \bar{z}_u - \hat{\bar{z}}_u$ ▷ Update \bar{r}_{u_u}
 - 15: $\bar{r}_u = F_{uu}^{-1} \bar{z}_u$ ▷ Matrix solve F_{uu}
-

Annexe F

**Article publié dans Communications
in Computational Physics**

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/348565970>

Navier–Stokes Solvers for Incompressible Single–and Two–Phase Flows

Article in *Communications in Computational Physics* · January 2021

DOI: 10.4208/cicp.OA-2020-0044

CITATIONS

2

READS

277

3 authors:



Elouafa Mohamed
Université Gustave Eiffel

5 PUBLICATIONS 3 CITATIONS

[SEE PROFILE](#)



Stéphane Vincent
Université Gustave Eiffel

258 PUBLICATIONS 2,949 CITATIONS

[SEE PROFILE](#)



Vincent Le Chenadec
CentraleSupélec

27 PUBLICATIONS 332 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Thermoacoustic instabilities [View project](#)



sediment and particle flow interactions [View project](#)

Navier-Stokes Solvers for Incompressible Single- and Two-Phase Flows

Mohamed El Ouafa^{1,*}, Stéphane Vincent¹ and Vincent Le Chenadec¹

¹ MSME, Univ. Gustave Eiffel, UMR CNRS 8208, Univ. Paris-Est Créteil,
5 bd Descartes, 77454 Marne-la-Vallée, France.

Received 3 March 2020; Accepted (in revised version) 23 July 2020

Abstract. The presented work is dedicated to the mathematical and numerical modeling of unsteady single- and two-phase flows using finite volume and penalty methods. Two classes of Navier-Stokes solvers are considered. Their accuracy and robustness are compared to identify their respective strengths and weaknesses. Exact (also referred to as monolithic) solvers such as the Augmented Lagrangian and the Fully Coupled methods address the saddle-point structure on the pressure-velocity couple of the discretized system by means of a penalization term or even directly, whereas approximate (segregated) solvers such as the Standard Projection method rely on operator splitting to break the problem down to decoupled systems. The objective is to compare all approaches in the context of two-phase flows at high viscosity and density ratios. To characterize the interface location, a volume-of-fluid (VOF) approach is used based on a Piecewise Linear Interface Construction (PLIC). Various 2D simulations are performed on single- and two-phase flows to characterize the behavior and performances of the various solvers.

AMS subject classifications: 76M12

Key words: Monolithic solvers, Augmented Lagrangian, two-phase flows, saddle point, projection method.

1 Introduction

The Navier-Stokes equations play a crucial role in several industrial and environmental applications that include weather forecast, combustion and propulsion, civil engineering, aerodynamics, and hydrodynamics. However, their resolution causes many problems and remains a challenging task for mathematicians as well as engineers. In the context of incompressible two-phase flows, the difficulties are two-fold: on the one hand, the

*Corresponding author. *Email addresses:* mohamed.elouafa@u-pem.fr (M. El Ouafa), stephane.vincent@u-pem.fr (S. Vincent), vincent.lechenadec@u-pem.fr (V. Le Chenadec)

velocity-pressure coupling induced by the incompressibility constraint, which must hold at every instant, gives rise to a saddle-point problem for which dedicated solvers must be designed. On the other hand, large density and viscosity ratios in the presence of large interface distortions yield ill-conditioned linear systems. Multiple strategies have been devised over the years to solve these ill-conditioned linear systems (the reader is referred to [7,9] for more information).

The devised solutions the pressure-velocity coupling challenge can be classified in two categories. On the one hand, segregated methods, introduced by Chorin [8] and Temam [36], approximate the original system using time splitting, thereby resulting in two decoupled equations: one to update the velocity field and the other the pressure field. This raises a number of issues, such as the appearance of numerical artifacts as a consequence of the splitting error, or the need for pressure boundary conditions that do not exist in the original problem. On the other hand, coupled methods solve both fields (velocity and pressure) simultaneously, hence preserving the consistency of the discretized system with the continuous equations (as opposed to segregated methods, which introduce a splitting error that is inconsistent with the continuous system). The system to be solved in coupled methods is known as a saddle point system on the pressure-velocity couple. This saddle point system can in turn be solved either directly (this strategy will be referred to as the Fully Coupled, or FC, method) or by means of a penalty method (Augmented Lagrangian, AL). The AL method was proposed initially by Fortin and Glowinski [14] primarily for single-phase Stokes flows. This approach was recently applied to two-phase flows by Vincent and coworkers [24, 38]. This is an iterative method, where each iteration requires the solution of the velocity equation modified to include an additional term that penalizes the incompressibility constraint. A subsequent update of the pressure field is performed explicitly. FC methods on the other hand are purely algebraic, using specific block preconditioners so as to improve the spectral properties of the original saddle point system. In recent years, significant efforts have been dedicated to this approach, and several techniques have been proposed in the context of single-phase flows, in the Stokes or steady regimes primarily, using finite element methods [11,26].

In the context of two-phase flow modeling at high density and viscosity ratios, several techniques have been investigated to tackle the induced stiffness. For projection methods (PR), Guermond and Salgado [18] have studied a new fractional time-step technique for variable density flows, which consist in extracting the density from the Poisson equation by penalizing the divergence of the velocity. The validity of this method is demonstrated for Rayleigh-Taylor instability at small density ratio. More recently, Dodd and Ferrante [10] have proposed a fast pressure-correction method for simulating incompressible two-phase flows with large density and viscosity ratios. The method is based on the splitting of the pressure gradient into two terms, one with a variable density which is explicitly treated and the other one with a constant density treated implicitly. The results obtained for a capillary wave show that this method is able to treat two-phase flow problems with density and viscosity ratios up to 10^5 . Nevertheless, the explicit treatment of the velocity prediction imposes a restriction on the time-step size to maintain numer-

ical stability. Another approach followed by Caltagirone and Breil [6] relies on a vector projection of the velocity field, using only a velocity correction and Hodge-Helmholtz decomposition. Three-dimensional tests show that this method is very effective for simulating two-phase flows with very large ratios of density and viscosity ($\rho_2/\rho_1 > 1000$ and $\mu_2/\mu_1 > 1000$). However, the solution of large sparse linear systems on all velocity components is complex, and as in the scalar projection method, it also has a time splitting error. In addition to these techniques, Guo and Xu [19] have proposed an accurate method for phase field models of two-phase incompressible flows, using the convex splitting principle. The numerical results presented for a rising bubble and a phase separation configuration of a binary fluid at low Reynolds show the high order accuracy and the stability of this method.

Regarding the Augmented Lagrangian method, Vincent et al. [38] have proposed the Algebraic Adaptive Augmented Lagrangian (AAAL) for simulating two-phase flows. As opposed to the Standard Augmented Lagrangian (SAL), where the penalty parameter is constant, AAAL automates the parameter selection locally by analysing the matrix arising from the discretization of the penalized velocity equation. The robustness of this method has been demonstrated by simulating the free fall of a dense cylinder, with large density and viscosity ratios (10^5 and 10^{10} , respectively). Although this method is very effective for two-phase flows with large contrasts, it is computationally expensive, because the linear systems to be solved are ill-conditioned and the only preconditioner identified by the authors as sufficiently robust (ILU [32]) does not scale well in parallel implementation with distributed memory. For the FC method, studies on two-phase flows have been documented as well. In the case of steady Stokes motions, Cai et al. [28] have investigated the rise of a bubble at high contrasts ($r_\rho = r_\mu = 10^3$). For unsteady Navier-Stokes equations, Bootland et al. [2] have proposed a new preconditioner (termed Pressure Convection-Diffusion, or PCD). Numerical results for a dynamic dam-break problem (air-water) showed that the PCD can be an effective approach for two-phase flows. A distinctive feature of the aforementioned FC methods is that they all rely on specific block-triangular preconditioners. Other authors, including Bosch et al. [22], model the flow differently by coupling the Navier-Stokes equation and the Cahn-Hilliard [4] equation to also yield coupled scheme. Such an approach, also preconditioned by block-triangular preconditioners and Schur complement approximations, has proved to be accurate and robust in the presence of large viscosity and density ratios.

The main objective of the present work is to further investigate Fully Coupled methods, the robustness of which can ultimately unlock new applications such as the simulation of two-phase flows at large density and viscosity ratios. The challenge effectively lies at the algebraic level, and the poor conditioning of the linear systems must be addressed in order to challenge the widespread use of segregated methods. In this context, the main objective of the current study is to review robust parallel preconditioners and to compare their performance with well-established alternatives. The paper is organized as follows: in Section 2 the Navier-Stokes equations are formulated for two-phase incompressible flows, and the temporal and spatial discretizations are presented. The solution

strategies used in our simulations are delineated in Section 3, with a particular emphasis for the block preconditioners used on the saddle point systems. Section 4 is dedicated to numerical results for single- and two-phase flows. This section also provides detailed comparisons with respect to various documented solvers for incompressible flows.

2 Governing equations

The one-fluid formulation, developed by Kataoka [25] for multiphase flows, is considered. In its incompressible form, the one-fluid model consists of the Navier-Stokes equation augmented with the capillary effect, supplemented by the divergence-free condition and a transport equation for a color function, denoted C , used to characterize the interface location and the fluid properties (mass density and viscosity). In the proposed work, C will be chosen as the volume fraction occupied by a reference phase. The originality of the one-fluid model is that it takes implicitly into account the jumps relations across the interface induced by surface tension, which reverts to the classical single phase Navier-Stokes equations away from the interface.

Let $\Omega \subset \mathbb{R}^d$ be an open bounded domain, with smooth boundary $\Gamma = \partial\Omega$ and \mathbf{n} the unit normal to Γ . The one-fluid model equations governing the multiphase flow motion read:

$$\frac{\partial \rho(C) \mathbf{u}}{\partial t} + \nabla \cdot (\rho(C) \mathbf{u} \otimes \mathbf{u}) = -\nabla p + \nabla \cdot [\mu(C) (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + \rho(C) \mathbf{g} + \mathbf{F}_s, \quad (2.1)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (2.2)$$

$$\frac{\partial C}{\partial t} + \mathbf{u} \cdot \nabla C = 0. \quad (2.3)$$

Eq. 2.1 represents the conservation of linear momentum, Eq. 2.2 the conservation of volume (the phases are immiscible and incompressible), and Eq. 2.3 the advection of the color function C . In Eqs. 2.1, 2.2, and 2.3, \mathbf{u} is the velocity field, p the pressure field, t the time, \mathbf{g} the gravity field, and ρ and μ the mixture density and dynamic viscosity, respectively. \mathbf{F}_s is the capillary term, modeled in this study by the continuum surface tension force model [3]. The mixture density and viscosity are defined according to the following constitutive laws :

$$\rho(C) = \rho_1 C + (1-C) \rho_2, \quad (2.4)$$

$$\mu(C) = \mu_1 C + (1-C) \mu_2, \quad (2.5)$$

where ρ_1 , ρ_2 , μ_1 and μ_2 are the densities and the viscosities of fluids 1 and 2. By mixture density or viscosity, it is understood that the aforementioned material properties are phase-wise constant. C (as well as ρ and μ , therefore) vary spatially in the interface vicinity only, that is in the vicinity of the cells or control volumes cut by the interface.

The current study focuses on solution techniques for the incompressible two-phase Navier-Stokes equations (2.1) and (2.2). The solution of the advection equation (2.3) is

performed with by the Volume-of-Fluid (VOF) method, using the Piecewise Linear Interface Construction (PLIC) technique [39]. The spatial discretization of the Navier-Stokes equations (2.1) and (2.2) is performed on a staggered mesh based on the MAC scheme of Harlow and Welch [20]. This approach guarantees the consistency of the differential operators such as the divergence and the gradient and it also avoids oscillations on the pressure field. A centered approximation of the convective term alleviates any numerical viscosity. The resulting semi-discrete system is advanced in time implicitly by means of a second order integration scheme (Gear scheme, Eq. (2.6)):

$$\left(\frac{\partial \mathbf{u}}{\partial t}\right)^{n+1} = \frac{\frac{3}{2}\mathbf{u}^{n+1} - 2\mathbf{u}^n + \frac{1}{2}\mathbf{u}^{n-1}}{\Delta t}, \quad (2.6)$$

and the convective term is linearized using Adams-Bashforth extrapolation (Eq. (2.7)):

$$\mathbf{u}^{n+1} \otimes \mathbf{u}^{n+1} \approx (2\mathbf{u}^n - \mathbf{u}^{n-1}) \otimes \mathbf{u}^{n+1}, \quad (2.7)$$

where n refers to the temporal iteration associated to physical time $n\Delta t$. The timestep Δt is assumed constant.

2.1 Physical boundary conditions

To complete the one-fluid formulation, boundary conditions for the velocity field \mathbf{u} on $\Gamma = \partial\Omega$ must be specified using penalisation [31]. This provides a unified treatment for imposing different types of boundary conditions, such as Dirichlet and Neumann conditions. Numerically, this method is based on a control term $\overline{\overline{B}}_{penu}(f(\mathbf{u}) - \mathbf{u}_\infty)$, which is added to the momentum equation:

$$NS(\mathbf{u}^{n+1}) + \overline{\overline{B}}_{penu}(f(\mathbf{u}^{n+1}) - \mathbf{u}_\infty) = 0, \quad (2.8)$$

where

$$\overline{\overline{B}}_{penu} = \begin{pmatrix} \alpha_u & 0 \\ 0 & \alpha_v \end{pmatrix} \quad (2.9)$$

is a tensor field whose diagonal components tend to infinity along the boundary Γ and are identically zero inside the fluid domain Ω . Here, $f(\mathbf{u}^{n+1})$ (Eq. (2.10) and Eq. (2.11)) is a discrete function of u^{n+1} and v^{n+1} , which is written as a linear combination of resolved velocities $u_{i,j}^{n+1}$, $v_{i,j}^{n+1}$ and their neighbors:

$$f(u) = a_0 u_{i,j}^{n+1} + a_1 u_{i-1,j}^{n+1} + a_2 u_{i+1,j}^{n+1} + a_3 u_{i,j-1}^{n+1} + a_4 u_{i,j+1}^{n+1}, \quad (2.10)$$

$$f(v) = a_0 v_{i,j}^{n+1} + a_1 v_{i-1,j}^{n+1} + a_2 v_{i+1,j}^{n+1} + a_3 v_{i,j-1}^{n+1} + a_4 v_{i,j+1}^{n+1}. \quad (2.11)$$

The treatment of the Dirichlet boundary conditions, applied to the left boundary for example (see Fig. 1) is controlled by the coefficients a_i which are then set to the following

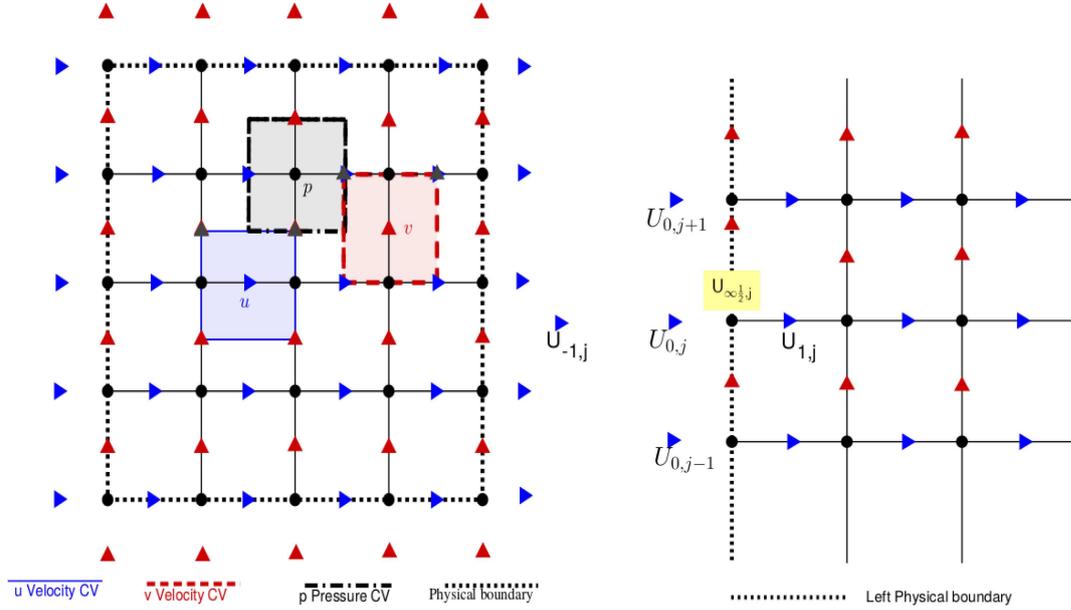


Figure 1: Left: staggered pressure and velocity unknowns in the global domain. Right: left boundary treatment in 2D.

values: $a_0 = a_2 = \frac{1}{2}$ and $a_1 = a_3 = a_4 = 0$. Thus, the discrete momentum equation on the left boundary becomes:

$$NS(u_{i,j}^{n+1}) + \alpha_u \left(\frac{1}{2} u_{0,j}^{n+1} + \frac{1}{2} u_{1,j}^{n+1} - u_{\infty \frac{1}{2}, j} \right) = 0. \quad (2.12)$$

To impose homogeneous Neumann boundary conditions, the coefficients are set to $a_0 = -1$, $a_2 = 1$ and $a_1 = a_3 = a_4 = 0$. The resulting equation in discrete form is:

$$NS(u_{i,j}^{n+1}) + \alpha_u \left(-u_{0,j}^{n+1} + u_{1,j}^{n+1} - u_{\infty \frac{1}{2}, j} \right) = 0, \quad (2.13)$$

with $u_{\infty \frac{1}{2}, j} = 0$.

The semi-discretized system, referred to as the penalized one-fluid model, finally reads:

$$\begin{aligned} & \frac{\rho(C^{n+1})}{\Delta t} \left(\frac{3}{2} \mathbf{u}^{n+1} - 2\mathbf{u}^n + \frac{1}{2} \mathbf{u}^{n-1} \right) + \nabla \cdot \left(\rho(C^{n+1}) \left(2\mathbf{u}^n - \mathbf{u}^{n-1} \right) \otimes \mathbf{u}^{n+1} \right) \\ & + \bar{\bar{B}}_{penu} \left(f(\mathbf{u}^{n+1}) - \mathbf{u}_\infty \right) \\ & = -\nabla p^{n+1} + \nabla \cdot \left[\mu(C^{n+1}) \left(\nabla \mathbf{u}^{n+1} + \left(\nabla \mathbf{u}^{n+1} \right)^T \right) \right] + \rho(C^{n+1}) \mathbf{g} + \mathbf{F}_s, \end{aligned} \quad (2.14)$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0, \quad (2.15)$$

$$\frac{C^{n+1} - C^n}{\Delta t} + \mathbf{u}^n \cdot \nabla C^n = 0. \quad (2.16)$$

3 Numerical methods

As mentioned in the introduction, multiple approaches to the numerical solution of the Navier-Stokes equations have been devised. In this section, such approaches for solving the motion equations in the context of two-phase flow modeling are presented.

3.1 Projection method

The literature on projection methods is vast and many variants and formulations exist. In the present work, the projection method proposed by Goda [16] is considered, where a pressure gradient is retained in the prediction step. As opposed to the original work of Goda, where the prediction step is performed explicitly, an implicit resolution is chosen here so as to overcome the time-step restrictions and guarantee numerical stability. The projection method essentially decouples velocity and pressure fields, and the resulting algorithm consists of two steps: first, a velocity prediction is performed, where \mathbf{u}^* is computed as the solution of a decoupled momentum equation.

$$\frac{\rho(C^{n+1}) \left(\frac{3}{2} \mathbf{u}^* - 2\mathbf{u}^n + \frac{1}{2} \mathbf{u}^{n-1} \right)}{\Delta t} + \nabla \cdot \left(\rho(C^{n+1}) \left(2\mathbf{u}^n - \mathbf{u}^{n-1} \right) \otimes \mathbf{u}^* \right) + \overline{\overline{B}}_{penu} (f(\mathbf{u}^*) - \mathbf{u}_\infty) = -\nabla p^n + \nabla \cdot [\mu(C^{n+1}) (\nabla \mathbf{u}^* + \nabla \mathbf{u}^{*T})] + \rho(C^{n+1}) \mathbf{g} + \mathbf{F}_s. \quad (3.1)$$

The predicted velocity however does not satisfy the divergence-free condition Eq. (2.15). In the second step, therefore, the intermediate velocity is corrected by projecting it on a divergence-free subspace. This step is performed by computing a pressure increment p^* as the solution of the following Helmholtz equation, with homogeneous Neumann boundary conditions:

$$\nabla \cdot \left(\frac{\Delta t}{\rho(C)} \nabla p^* \right) + \alpha_p (f(p^*) - p_\infty) = \nabla \cdot \mathbf{u}^*, \quad (3.2)$$

where the penalisation term α_p vanishes in Ω and tends to infinity as Γ is approached. The velocity and pressure updates are performed as follows:

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\Delta t}{\rho(C)} \nabla p^*, \quad (3.3)$$

$$p^{n+1} = p^n + p^*. \quad (3.4)$$

The appeal for the projection method stems from its simplicity and ease of implementation. However, this approach suffers from well-known limitations: the splitting operator introduces an error that reduces the temporal accuracy of the numerical solution, and introduces the need for appropriate boundary conditions for the pressure field in the projection step which do not exist in the original (coupled) system. Indeed, the inconsistency of the boundary conditions induces numerical errors in the pressure field, giving rise to

artificial boundary layers as described in [17]. Additionally, when multiphase flows are simulated with large density ratios, the challenging task is to solve the linear system arising from the discretization of the variable coefficient Helmholtz equation (3.2), which become ill-conditioned due to discontinuities in the Laplacian operator $\nabla \cdot \left(\frac{\Delta t}{\rho(C)} \nabla p^* \right)$.

3.2 Augmented Lagrangian method

The Augmented Lagrangian was proposed to alleviate the shortcomings of the projection method. Originally developed by Fortin and Glowinski [14] for single-phase Stokes flows, this method has been extended recently for two-phase flows [24, 38] to take advantage of its robustness and efficiency. The principle of this method is that the conservation equation of mass is not used in its original form, but rather transformed into an explicit pressure Eq. (3.6) that will be coupled to the velocity. The resolution consists in finding a saddle point on the couple $(\mathbf{u}^{n+1}, p^{n+1})$. Starting with $\mathbf{u}^{*,0} = \mathbf{u}^n$ and $p^{*,0} = p^n$, the solution is given by an Uzawa algorithm [24]: while $|\nabla \cdot \mathbf{u}^{*,m}| > \epsilon$, solve

$$\begin{aligned} & \frac{\rho(C) \left(\frac{3}{2} \mathbf{u}^{*,m} - 2\mathbf{u}^{*,0} + \frac{1}{2} \mathbf{u}^{*, -1} \right)}{\Delta t} + \nabla \cdot \left(\rho(C) \left(2\mathbf{u}^{*,0} - \mathbf{u}^{*, -1} \right) \otimes \mathbf{u}^{*,m} \right) + \overline{\overline{B}}_{penu} (f(\mathbf{u}^{*,m}) - \mathbf{u}_\infty) \\ & = -\nabla p^{*,0} + r \nabla (\nabla \cdot \mathbf{u}^{*,m}) + \nabla \cdot [\mu(C) (\nabla \mathbf{u}^{*,m} + \nabla \mathbf{u}^{(*,m)T})] + \rho \mathbf{g} + \mathbf{F}_s, \end{aligned} \quad (3.5)$$

$$p^{*,m} = p^{*,0} - r \nabla \cdot \mathbf{u}^{*,m}, \quad (3.6)$$

where ϵ is the divergence threshold and m is the iteration index. So for $m = 1, 2, \dots, n$, the iterative process simultaneously corrects the pressure and the velocity until convergence to the solution $(\mathbf{u}^{n+1}, p^{n+1})$. In practice, one iterative step of the Augmented Lagrangian with a suitable value of the parameter r suffices to obtain a good solution. A proper choice of parameter r guarantees that the incompressibility constraint Eq. (2.15) is satisfied while simultaneously ensuring that the magnitude of the Augmented Lagrangian term remains comparable to the largest of the other terms in the Navier-Stokes equations. For low r values, the importance is given to the Navier-Stokes equations whereas for large values, the velocity field will be divergence-free but will not satisfy the Navier-Stokes equations. In practice, values of r between 10 to 1000 times larger than the other dimensionless terms yield satisfactory results. The reader is referred to recent work on the automation of its computation for more information [37].

The Augmented Lagrangian method has many advantages. On the one hand, it avoids imposing boundary conditions on the pressure field. On the other hand, using the Augmented Lagrangian method, two-phase flows at high density and viscosity ratios ($\frac{\rho_2}{\rho_1} \ll 1$ or $\frac{\rho_2}{\rho_1} \gg 1$ as well as $\frac{\mu_2}{\mu_1} \ll 1$ or $\frac{\mu_2}{\mu_1} \gg 1$) can be simulated, as opposed to the ranges achievable with the Standard Projection method. However, the disadvantage of this method is its numerical cost, as a result of the effect of parameter r . Indeed, the simulation of two-phase flows with high density and viscosity ratios, together with large values of the parameter r , leads to ill-conditioned linear systems. In addition, it has to be highlighted that the numerical solution strongly depends on the choice of the parameter

r , itself largely sensitive to problem parameters such as the density and viscosity ratios, and grid sizes.

3.3 Fully Coupled method

To date, the Fully Coupled (FC) method, although widely used in the finite element community in the context of stationary and single-phase flow simulations, has not found much traction within the finite difference and finite volume communities. The objective of this study is to investigate Fully Coupled approaches in the context of unsteady two-phase flow simulated by finite volumes and penalty methods. They require the solution of systems of the following form:

$$\begin{pmatrix} \frac{1}{\Delta t} M_u^{(\rho)} + N_u^{(\rho)} + L_u^{(\mu)} & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ 0 \end{pmatrix}, \quad (3.7)$$

or $A\mathbf{x} = \mathbf{b}$. Eq. (3.7) represents a non-symmetric saddle-point problem, where the matrix B_u is the discrete negative divergence, B_p^T represents the discrete pressure gradient operator, $M_u^{(\rho)}$ is the velocity mass matrix, $L_u^{(\mu)}$ is the discrete velocity Laplacian, $N_u^{(\rho)}$ denotes the linearized convective operator, and \mathbf{f} represents the right-hand side vector.

To solve this sparse linear system, direct methods such as Gaussian elimination are very robust with respect to various problem but do not scale well with problem size and are impractical in three-dimensional configurations. In the context of large-scale systems, preferred resolution techniques include iterative solution techniques, such as the generalized minimal residual (GMRES [33]), algebraic or geometric multigrid methods [35], and the conjugate gradient and its variants [23]. In this study, the BiCGStab(2) algorithm [23] was found to perform slightly better than GMRES (10% on average) for both Fully Coupled and Augmented Lagrangian systems, and was therefore preferred in spite of the favorable monotonous convergence property of GMRES that BiCGStab(2) does not share.

Because the performances of the iterative methods depends on the conditioning of the system, the use of an efficient preconditioner proves necessary to accelerate the convergence of the iterative solver and to avoid possible numerical instabilities. The preconditioning can be considered as a transformation of the original system $A\mathbf{x} = \mathbf{b}$ into an equivalent one $P^{-1}A\mathbf{x} = P^{-1}\mathbf{b}$, where the preconditioning matrix P is an approximation of A ($P \approx A$) whose inverse can be efficiently applied.

In order to build a suitable preconditioner P , we consider the block LDU decomposition of the original matrix A , given by:

$$\begin{pmatrix} F_u & B_p^T \\ B_u & 0 \end{pmatrix} = \begin{pmatrix} I_u & 0 \\ B_u F_u^{-1} & I_p \end{pmatrix} \begin{pmatrix} F_u & 0 \\ 0 & S_u \end{pmatrix} \begin{pmatrix} I_u & F_u^{-1} B_p^T \\ 0 & I_p \end{pmatrix}. \quad (3.8)$$

Here $S_u = -B_u F_u^{-1} B_p^T$ is the Schur complement of the pressure block and $F_u = \frac{1}{\Delta t} M_u^{(\rho)} + N_u^{(\rho)} + L_u^{(\mu)}$ is the convection-diffusion-reaction operator. In this work, one of the most

Algorithm Triangular preconditioner

- 1: Solve $S_p z_p = r_p$.
 - 2: Update $\mathbf{r}_u \leftarrow \mathbf{r}_u - B_p^T z_p$.
 - 3: Solve $F_u \mathbf{z}_u = \mathbf{r}_u$.
-

popular block triangular preconditioner for saddle point problems is used, by taking the DU product of Eq. (3.8):

$$P = \begin{pmatrix} F_u & B_p^T \\ 0 & S_p \end{pmatrix}. \quad (3.9)$$

Thus, the application of the preconditioner of Eq. (3.9) entails the following backward substitution:

$$\begin{pmatrix} F_u & B_p^T \\ 0 & S_p \end{pmatrix} \begin{pmatrix} \mathbf{z}_u \\ z_p \end{pmatrix} = \begin{pmatrix} \mathbf{r}_u \\ r_p \end{pmatrix}, \quad (3.10)$$

where $\begin{pmatrix} \mathbf{r}_u \\ r_p \end{pmatrix} = \mathbf{b}$ and \mathbf{p} and $\begin{pmatrix} \mathbf{r}_u \\ r_p \end{pmatrix} = \mathbf{b}, \mathbf{v}^k, \mathbf{s}^k, \mathbf{t}^k$ and \mathbf{w}^k are a sequence of vectors generated at each BiCGSTAB(2) (for further details, the reader is referred to Appendix A).

When the triangular preconditioner P^{-1} is applied to A , the preconditioned system $P^{-1}A$ has a single eigenvalue $\lambda = 1$ of multiplicity $n_u + n_p$. The characteristic polynomial of $P^{-1}A$ has degree 2, which guarantees that a Krylov subspace solver such as GMRES converges in two iterations. Indeed, by considering the eigenvalue problem for the preconditioned system $P^{-1}A$, this values can easily be obtained from the following generalized eigenvalue problem:

$$\begin{pmatrix} F_u & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ p \end{pmatrix} = \lambda \begin{pmatrix} F_u & B_p^T \\ 0 & S_p \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ p \end{pmatrix}. \quad (3.11)$$

From the first line of equation of (3.11) we obtain $(1 - \lambda)(F_u \mathbf{u} + B_p^T p) = 0$, admitting two possible solutions. The first eigenvalue is $\lambda = 1$ of multiplicity n_u , and the second corresponds to $\mathbf{u} = -F_u^{-1} B_p^T p$. Then, the substitution of \mathbf{u} in the second row of (3.11) leads to $-B_u F_u^{-1} B_p^T p = \lambda S_p p$, which gives the second eigenvalue $\lambda = 1$ of multiplicity n_p .

Likewise, if instead of the upper triangular preconditioner $P = DU$, the diagonal block preconditioner D is used, then $\lambda = 1$ appears to be as an eigenvalue of multiplicity at least $n_u - n_p$ for any eigenvector $[\mathbf{u}^T, \mathbf{0}^T]^T$ with $B\mathbf{u} = 0$, while $\lambda = \frac{1}{2} \pm \frac{\sqrt{5}}{2}$ of multiplicity n_p as described in [11]. As the convergence depends on the eigenvalues λ of the preconditioned system (the smaller the eigenvalue the faster the convergence), the block triangular preconditioner is expected to perform better and is therefore selected for this work.

This approach has however one major drawback, which is that the inversion of S_p and F_u can not be performed in realistic computations. They are indeed more expensive than solving the saddle point (3.7) by direct methods. As a result, they must be approximated by matrices \hat{S}_p and \hat{F}_u .

Finding an effective approximation for the Schur complement S is not obvious. In recent years, a variety of preconditioners for incompressible flow problems has been introduced. These include the Cahouet-Chabard preconditioner [5] for homogeneous Stokes flows. This preconditioner has been extended [28] to two-phase flows in the case of steady Stokes flow. The Augmented Lagrangian formulation can also be used as a preconditioner, for example in the case of the Oseen problem [27] or the steady Stokes equations with variable viscosity [27]. Recently, Bootland and coworkers [2] have proposed two preconditioners coined the Pressure Convection Diffusion (PCD) preconditioner and the Least Squares Commutator (LSC) preconditioner. For these preconditioners, the results show that the PCD preconditioner is more efficient than the LSC preconditioner. In this work, the pressure convection diffusion (PCD) preconditioning with suitable scaling is therefore chosen, the aforementioned study showing it to be the most effective at solving unsteady two-phase flows [2]. This preconditioner is given by:

$$\hat{S}_{PCD}^{-1} = (M_p^{1/\mu})^{-1} + (A_p^{1/\rho})^{-1} \left(N_p^{(1)} + \frac{1}{\Delta t} M_p^{(1)} \right) (M_p^{(1)})^{-1}, \quad (3.12)$$

where $M_p^{1/\mu}$ is the diagonal pressure mass matrix scaled by the inverse of the viscosity $1/\mu$, $M_p^{(1)}$ is the diagonal standard pressure mass matrix, $N_p^{(1)}$ represents the convective term in the pressure space, and $A_p^{1/\rho}$ is the scaled Laplacian that corresponds to the discretisation of the term $\nabla \cdot (\frac{1}{\rho} \nabla P)$.

In order to approximate the inverse of the Schur complement \hat{S}_{PCD}^{-1} , the Semi Coarsening Multigrid (SMG) solver of the HYPRE library [13] is used, where the action of $A_p^{(1/\rho)}$ is required. This solver was indeed designed specifically for elliptic equations with variable coefficients. The two diagonal mass matrix are then inverted, namely $(M_p^{(1/\mu)})^{-1}$ and $(M_p^{(1)})^{-1}$, by applying a rescaling to suitable vectors followed by a matrix-vector product for the operator $F_p^{(1)} = N_p^{(1)} + \frac{1}{\Delta t} M_p^{(1)}$. As described in the work of Elman et al. [12], the use of this preconditioner requires appropriate boundary conditions, which must be compatible with the physical boundary conditions applied to the velocity \mathbf{u} . Indeed, the choice of Dirichlet boundary conditions for the velocity \mathbf{u} implies the application of Neumann boundary conditions for the discrete operators $A_p^{(1/\rho)}$ and $N_p^{(1)} + \frac{1}{\Delta t} M_p^{(1)}$. On the other hand, the use of inflow or outflow boundary conditions respectively for the velocity \mathbf{u} results in the use of Robin conditions in the case of inflow and Dirichlet or Neumann (depending on the Reynolds number) boundary conditions in the case of outflow for the discrete operators of the Schur approximation \hat{S}_{PCD} .

Thus, to build a robust and efficient preconditioning P , a good approximation \hat{F}_u is required. Many approximations of the velocity block F_u have been documented, for the most part based on the use of tailored multigrid implementations. These include product splitting, block Jacobi, block Gauss-Seidel, block diagonal, and block triangular preconditioners (see [1] for more details). In the current study, the block Gauss-Seidel precondi-

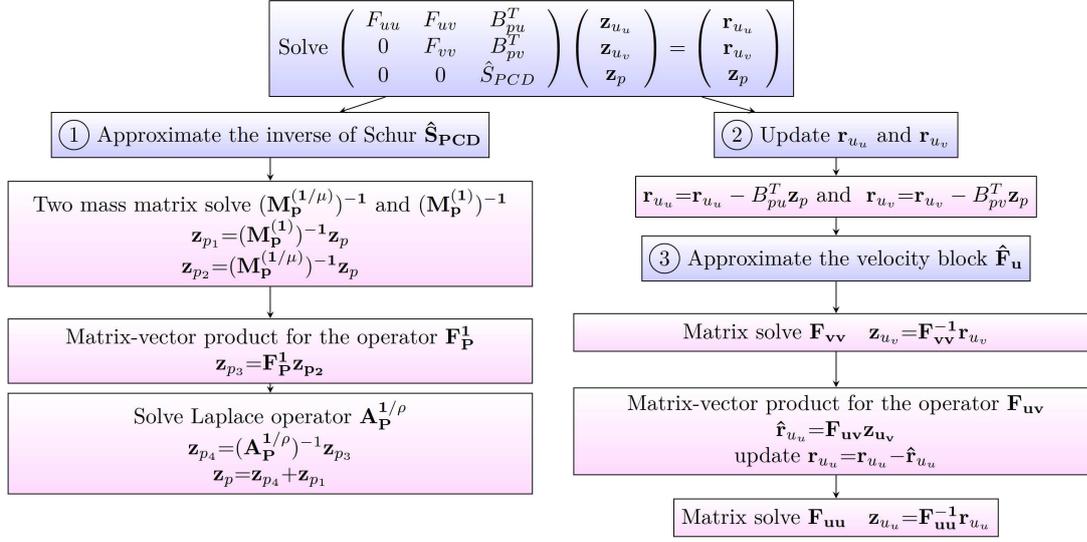


Figure 2: Graphical representation of the application of the block triangular preconditioner.

tioner

$$\hat{F}_u = \begin{pmatrix} F_{uu} & F_{uv} \\ 0 & F_{vv} \end{pmatrix}$$

was found to perform satisfactorily. This preconditioner requires the solution of two linear systems, F_{vv} and F_{uu} , which are both performed using the aforementioned multigrid solver (SMG). The algorithm applied to block triangular preconditioner is summarized in the graphical representation shown in Fig. 2.

4 Numerical results

4.1 Single phase flows: Driven cavity

The first benchmark problem considered is a lid driven cavity for viscous incompressible fluid flow in a two-dimensional square domain of side $L = 1m$. In this configuration, all boundaries are walls, the top side moves at velocity $u_0 = -1m \cdot s^{-1}$, while the other three sides are no slip walls. Three Reynolds numbers are solved ($Re = 100, 1000, \text{ and } 5000$) with the Fully Coupled solver, the lower two being compared with the results of Botella and Peyret [30]. The convergence rate is estimated by means of a Richardson extrapolation (for further details, the reader is referred to Appendix B). Several uniform Cartesian meshes are used, while the timestep is maintained constant and equal to $10^{-3}s$. As only the steady flow is considered here, the iterations are terminated when the stationary criterion

$$\max \left(\left\| \mathbf{u}^{n+1} - \mathbf{u}^n \right\|, \left\| \mathbf{v}^{n+1} - \mathbf{v}^n \right\| \right) \leq \epsilon \max \left(\left\| \mathbf{u}^{n+1} \right\|, \left\| \mathbf{v}^{n+1} \right\| \right) \quad (4.1)$$

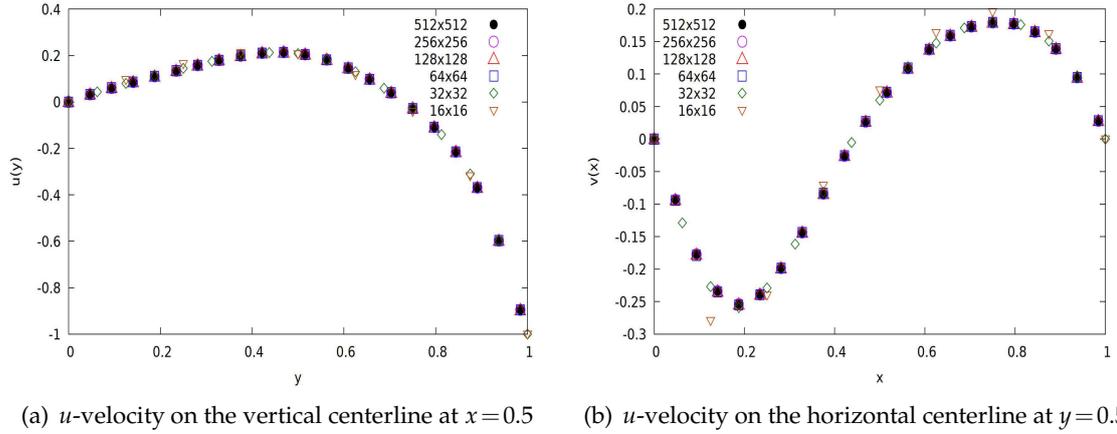


Figure 3: Lid driven cavity at $Re=100$ — comparison of the horizontal and vertical velocity along the centerline through the center of the cavity to results of Botella and Peyret [30].

Table 1: Lid driven cavity at $Re=100$ – comparison of some characteristic values along the middle-line to reference values of Botella and Peyret [30] - * is the Richardson interpolated value.

N	u_{center}	order	v_{center}	order	ω_{center}	order	u_{max}	order	v_{max}	order
64	0.20876	1.39	0.05804	3.03	1.17746	5.01	0.21375	-	0.17991	3.72
128	0.20904	1.41	0.05767	2.34	1.17461	5.97	0.21395	1.65	0.17956	2.09
256	0.20912	1.85	0.05757	1.89	1.17446	1.29	0.21402	1.56	0.17959	2.10
512	0.20914	1.98	0.05754	1.82	1.17440	1.47	0.21403	2.53	0.17957	1.91
1024	0.20914	2.00	0.05753	1.94	1.17439	2.05	0.21403	2.05	0.17957	2.04
*	-	-	-	-	1.17439	-	0.21403	-	0.179560	-
Ref. [30]	-	-	-	-	1.17441	-	0.21404	-	0.179572	-

is satisfied ($\epsilon=10^{-6}$). The Reynolds number is defined as:

$$Re = \frac{\rho u_0 L}{\mu}, \quad (4.2)$$

where ρ is the density, μ is the dynamic viscosity, L is the length of the cavity, and u_0 is the reference velocity of the flow.

Figs. 3(a) and 3(b) show the velocity plots along the middle-line of the cavity, obtained by the Fully Coupled method for $Re=100$ with various uniform Cartesian meshes. It is clear from these figures that the results are qualitatively similar starting from the mesh 32×32 for both velocities components u and v . Additionally, Table 1 includes the numerical values corresponding to the maximum of the velocity u on the vertical centerline (denoted u_{max}), the maximum of the velocity v on the horizontal centerline (v_{max}), the vorticity ω_{center} , the velocity u_{center} , and v_{center} at the center of the cavity. These values are compared to the reference solution. As can be seen in Table 1, all computational results

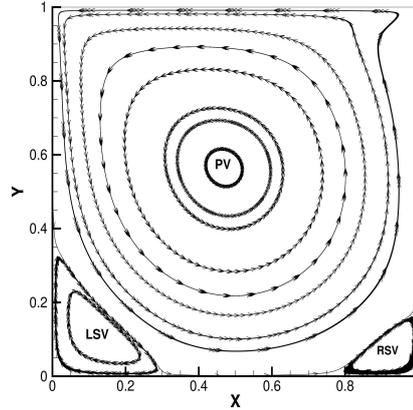


Figure 4: Stream function contours of primary and secondary vortex, obtained at $Re = 1000$.

are in good agreement with the reference [30] and the differences do not exceed 0.006%. The estimated accuracy for u_{center} , v_{center} , ω_{center} , u_{max} and v_{max} at the finest mesh are 2.00, 1.94, 2.05, 2.05 and 2.04, as expected given the second order accuracy of the spatially centered scheme.

A second test is considered for $Re = 1000$, for which the flow dynamics are largely dominated by inertial forces. The objective of this test is to investigate the ability of the solver to capture the different structures such as primary, right secondary and left secondary vortex, and to evaluate the vorticity at these locations. Again, the numerical results are compared to the reference solution [30], keeping the same numerical parameters as in the previous study.

At $Re = 1000$, the stream function of the flow is illustrated in Fig. 4. The existence of three distinct vortices is visible. The primary vortex denoted PV is large and centered in the cavity, the other secondary vortices represented by LSV and RSV are small and located near the left and right bottom corner of the cavity, respectively. In Fig. 5, the velocity u extracted from the vertical centerline and the velocity v along the horizontal centerline are presented. The numerical results show good agreement [30], which validates the accuracy of the solver. In addition, Tables 2, 3, and 4 summarize the values of the vorticity at the center of each vortex and their positions. All results exhibit second order accuracy in the asymptotic convergence zone. For the positions of the vortex, the difference is always smaller than 0.11%, and for the vorticity, the largest discrepancy is of the order of 0.06%.

To validate the ability of the Fully Coupled solver to simulate more complicated stationary problems, the case at $Re = 5000$ is also investigated. The profiles of the horizontal (u) and vertical (v) velocity components and vorticity (ω) along the centerlines $x=0.5$ and $y=0.5$ are shown in Fig. 6. Velocity extrema on the centerlines of the cavity, intensities of the vortex and their corresponding locations are given in Tables 5 and 6, respectively. As can be seen in these tables, the agreement between the current solution and the refer-

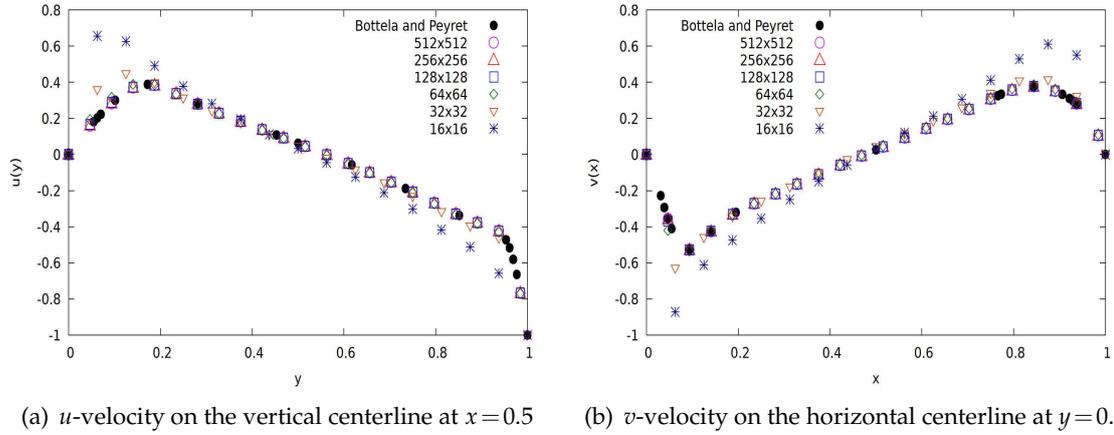


Figure 5: Lid driven cavity at $Re=1000$ — comparison of the horizontal and vertical velocity along the centerline through the center of the cavity to results of Botella and Peyret [30].

Table 2: Lid driven cavity at $Re=1000$ — Richardson extrapolation for the primary vortex: (x,y) the location of the center of the vortex, and ω is the value of the vorticity at this location.

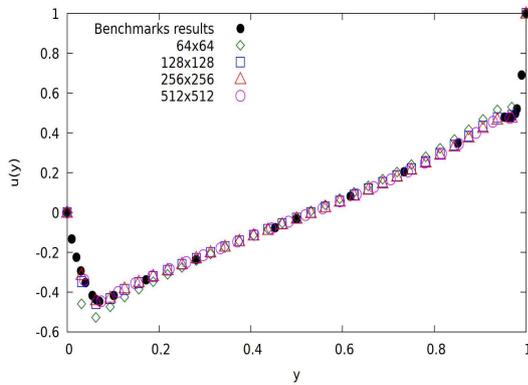
N	x	order	y	order	ω	order
32	0.460674		0.545429		2.16344	
64	0.467474		0.560757		2.07752	
128	0.468837	1.31	0.564257	2.13	2.06684	3.01
256	0.469113	2.30	0.565009	2.21	2.06693	-
512	0.469185	1.93	0.565183	2.11	2.06737	2.29
Extra	0.469179	-	0.56518	-	2.067369	-
Ref. [30]	0.4692	-	0.5652	-	2.067753	-

Table 3: Lid driven cavity at $Re=1000$ — Richardson extrapolation for the left secondary vortex: (x,y) the location of the center of the vortex, and ω is the value of the vorticity at this location.

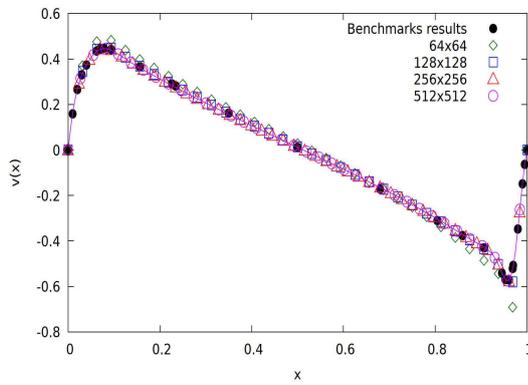
N	x	order	y	order	ω	order
32	0.08940		0.08617		-1.50012	
64	0.12906		0.10731		-1.23281	
128	0.13438	2.89	0.11081	2.59	-1.13287	1.41
256	0.13556	2.16	0.11157	2.20	-1.11431	2.42
512	0.13586	1.96	0.11174	2.12	-1.11053	2.30
Extra	0.13584	-	0.11173	-	-1.1105301	-
Ref. [30]	0.1360	-	0.1118	-	-1.109789	-

Table 4: Lid driven cavity at $Re=1000$ — Richardson extrapolation for the right secondary vortex: (x,y) the location of the center of the vortex, and ω is the value of the vorticity at this location.

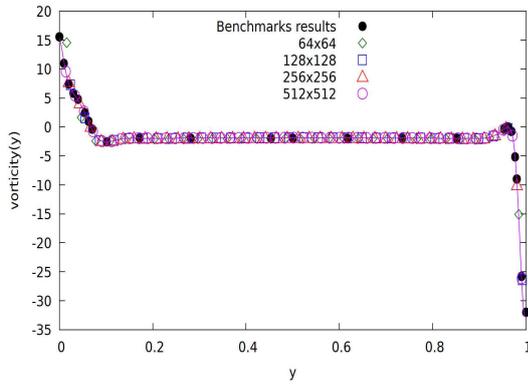
N	x	order	y	order	ω	order
32	0.9073		0.0985712		-0.736348	
64	0.9152		0.0808984		-0.401598	
128	0.916427	2.68	0.0786425	2.96	-0.361584	3.06
256	0.916657	2.41	0.07821	2.38	-0.354138	2.42
512	0.916709	2.14	0.0781233	2.31	-0.352588	2.26
Extra	0.916708	-	0.078116	-	-0.3525867	-
Ref. [30]	0.9167	-	0.0781	-	-0.3522861	-



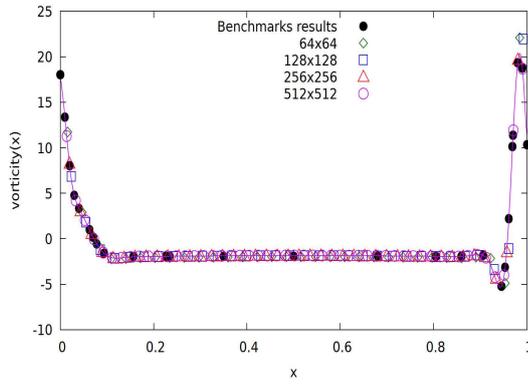
(a) u -velocity on the vertical centerline at $y=0.5$



(b) v -velocity on the horizontal centerline at $x=0.5$



(c) vorticity on the vertical centerline at $x=0.5$



(d) vorticity on the horizontal centerline at $y=0.5$

Figure 6: Lid driven cavity at $Re = 5000$ — Comparison of the horizontal, vertical velocity and vorticity along the centerline through the center of the cavity to reference results (given on the site <https://www.acenumerics.com/the-benchmarks.html>).

Table 5: Extrema of the velocity through the centerlines of the cavity, at $Re=5000$ obtained using a 512×512 Cartesian grid – Ref is the reference values given on the site <https://www.acenumerics.com/the-benchmarks.html>.

	y_{\min}	u_{\min}	x_{\max}	u_{\max}	x_{\min}	u_{\min}
Present work	0.0742196	-0.44616	0.079103	0.445349	0.958007	-0.574624
Ref	0.0742967	-0.4473136	0.0796881	0.4475084	0.9574605	-0.5761359

Table 6: Cavity vortex map, at $Re=5000$ obtained using a 512×512 Cartesian grid. Ref denotes the reference values given on the site <https://www.acenumerics.com/the-benchmarks.html>. PV, BR, BL, and TL denote primary, bottom right, bottom left, and top left vortices, respectively.

	x	x_{ref}	y	y_{ref}	ω	ω_{ref}
PV	0.515265	0.5150938	0.535095	0.5352620	-1.94649	-1.940636
TL1	0.0630822	0.06335432	0.909118	0.9092568	2.08812	2.091562
BR1	0.805512	0.8046258	0.072499	0.07274724	2.76827	2.752178
BR2	0.978491	0.9783737	0.018725	0.01877709	-0.0350574	-0.0349262
BL1	0.0727551	0.07285072	0.137474	0.1370630	1.52971	1.518861

ences values is small, albeit large than the smaller Reynolds number values (on the order of 1.64%).

4.2 Single phase flows: Green-Taylor vortex

The second test case is the unsteady Green-Taylor vortex, used as a benchmark to validate numerical implementations. This problem is defined in two dimensions on a square domain $\Omega = (0,1) \times (0,1)$, with periodic boundary conditions. In this case, the analytical solution has the following form with $\nu = \frac{\mu}{\rho}$:

$$u(x,y,t) = V_0 \cos(2\pi x) \sin(2\pi y) e^{-8\pi^2 \nu t}, \quad (4.3)$$

$$v(x,y,t) = -V_0 \sin(2\pi x) \cos(2\pi y) e^{-8\pi^2 \nu t}. \quad (4.4)$$

Using Eqs. (4.3)-(4.4), two interesting physical quantities can be evaluated: the kinetic energy E_k and the enstrophy ζ :

$$E_k = \int_0^1 \int_0^1 \rho \frac{(u^2 + v^2)}{2} dx dy = \frac{\rho V_0^2 F^2(t)}{4}, \quad (4.5)$$

$$\zeta = \int_0^1 \int_0^1 \rho \frac{\omega^2}{2} dx dy = 2\pi^2 \rho V_0^2 F^2(t), \quad (4.6)$$

where $\omega = \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right)$ is the vorticity in two dimensions and $F(t) = e^{-8\pi^2 \nu t}$.

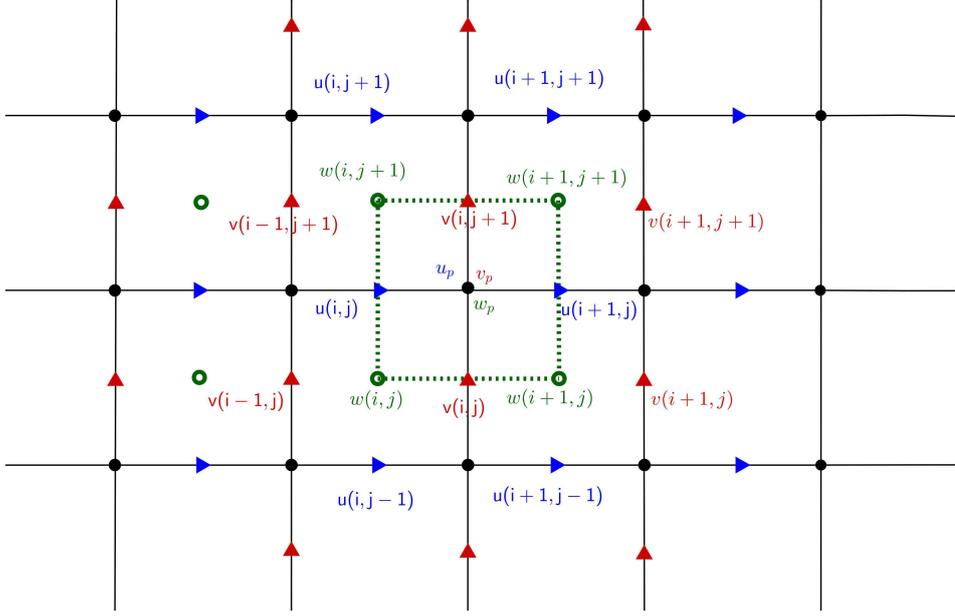


Figure 7: Calculation of the kinetic energy E_k and the enstrophy ζ .

The following discrete analogs of the kinetic energy E_k and enstrophy ζ are used:

$$E_k \simeq \sum_{\Omega_p} \frac{\rho}{2} (u_p^2 + v_p^2) \Omega_p, \quad (4.7)$$

$$\zeta \simeq \sum_{\Omega_p} \frac{\rho}{2} \omega_p^2 \Omega_p. \quad (4.8)$$

The squared velocity components u_p^2 et v_p^2 are obtained by the following interpolations:

$$u_p^2 = \frac{1}{2} (u_{i,j}^2 + u_{i+1,j}^2), \quad (4.9)$$

$$v_p^2 = \frac{1}{2} (v_{i,j}^2 + v_{i,j+1}^2). \quad (4.10)$$

The vorticity is defined as the rotational of the velocity ($\vec{\omega} = \nabla \times \vec{u}$). In 2D, the vorticity is reduced to a scalar that will be expressed at the viscosity node (see Fig. 7):

$$\omega(i,j) = \frac{v(i,j) - v(i-1,j)}{\Delta x} - \frac{u(i,j) - u(i,j-1)}{\Delta y}. \quad (4.11)$$

The square of vorticity ω_p^2 is obtained by the mean of the squares of the four vorticities that have been just expressed. The following equation is obtained:

$$\omega_p^2 = \frac{\omega^2(i,j) + \omega^2(i+1,j) + \omega^2(i,j+1) + \omega^2(i+1,j+1)}{4}. \quad (4.12)$$

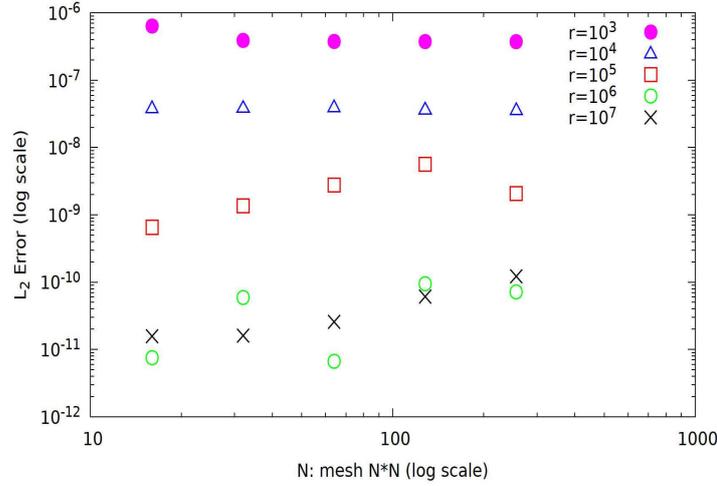


Figure 8: Relative L_2 error of the kinetic energy E_k for the inviscid Taylor-Green test case with different Lagrangian parameters r .

In order to quantify the efficiency and robustness of the methods that were presented previously, relative errors on the kinetic energy E_k and enstrophy ζ are calculated for both viscous and inviscid flows. The numerical results are confronted to the analytic values obtained from Eqs. (4.5) and (4.6). So, let q_t^{exact} be the exact quantity, q_t the numerical quantity and n_t the number of time steps. The error on q_t is estimated by the following expression:

$$Err_2^{rel}(q_t) = \left(\frac{\sum_{k=1}^{n_t} \|q_t^{exact} - q_t\|^2}{\sum_{k=1}^{n_t} \|q_t^{exact}\|^2} \right)^{1/2}. \quad (4.13)$$

For this case, three solvers are used: the Fully Coupled with PCD, the Standard Augmented Lagrangian (with $r = 10^6$ in the inviscid case and $r = 10^2$ in the viscous case) and the Standard Projection. Five meshes are simulated, ranging from 16×16 to 256×256 , with a residual of $\epsilon = 10^{-12}$ for the BiCGSTAB(2). A constant timestep of $\Delta t = 10^{-3}s$ is selected and 2000 time iterations are computed, corresponding to 0.3 seconds of flow motion. The fluid properties are $1kg \cdot m^{-3}$ for the density and $10^{-3}Pa \cdot s$ for the dynamic viscosity in viscous case respectively (Figs. 9 and 10). The characteristic velocity V_0 is chosen equal to $1m \cdot s^{-1}$.

In order to compare the three solvers more effectively, the effect of the Augmented Lagrangian parameter is considered in Fig. 8, for r values in the range 10^4 and 10^7 , and for each mesh size. It can be observed in this figure that a value $r = 10^6$ of the Augmented Lagrangian parameter appears to be a good choice in order to conserve the kinetic energy E_k to the order of machine precision. Note that beyond this value, the cost of calculating with this method becomes very high.

The L_2 error of the enstrophy ζ and kinetic energy E_k are shown in Figs. 9(a) and 9(b) for the inviscid flow. There is no dissipation phenomenon ($\mu = 0Pa \cdot s$). For the enstrophy

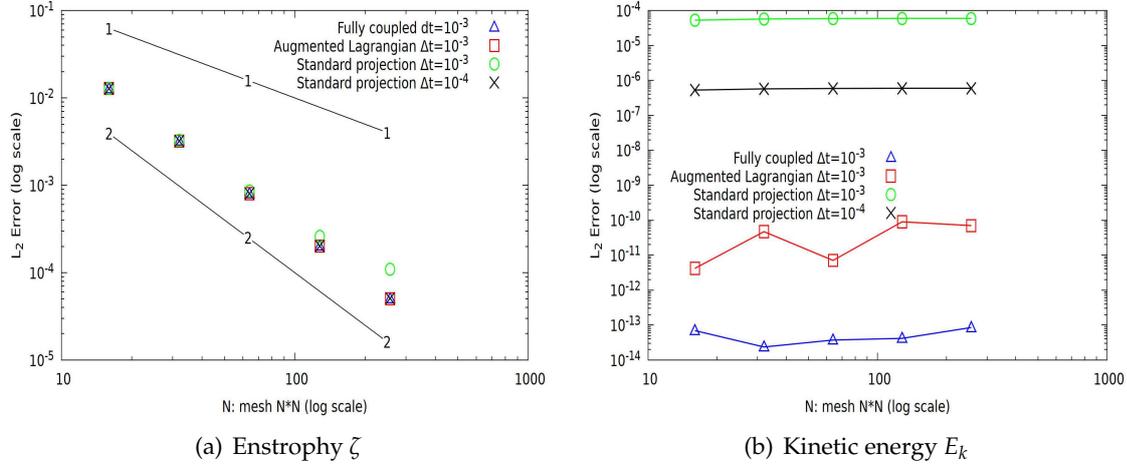


Figure 9: Kinetic energy E_k and enstrophy ζ conservations and accuracy of the different methods in the inviscid Taylor-Green test case.

ζ , as the approximations of the derivatives are second order, all solvers have second order accuracy, except the Standard Projection with $\Delta t = 10^{-3}s$ which is unable to follow the expected accuracy. Concerning the kinetic energy E_k , the results show that the Fully Coupled and the Augmented Lagrangian perform better than the Standard Projection, because they conserve the kinetic energy E_k to the order of machine accuracy. Therefore, the impact of the time step Δt on the Standard Projection is clearly observed due to operator splitting that introduces numerical dissipation. Indeed, when the time step is divided by 10, the error on the kinetic energy error is divided by 100. The same case was simulated on physical times up to 20 seconds, by keeping the same numerical parameters as in the first case. In this case, all solvers continue to converge to machine precision, except the Standard Projection that fails after 6 seconds due to the accumulation of errors, even for small time steps of the order $\Delta t = 10^{-6}s$. In addition, it has been shown that skew-symmetric schemes preserve energy to machine accuracy on fine meshes [15]. We also note that spectral methods are also able to preserve this invariant to machine precision [34]. The centered scheme combined with either Fully Coupled or Augmented Lagrangian techniques therefore also conserves kinetic energy discretely.

Result for the viscous case with $\mu = 10^{-3}Pa \cdot s$ are presented in Fig. 10. For the enstrophy ζ and the kinetic energy E_k , it is easily noticeable that the Fully Coupled and Augmented Lagrangian converge with second order accuracy. The effect of the time step is visible for the Standard Projection. Indeed, results with $\Delta t = 10^{-4}s$ are noticeably better than those obtained with $\Delta t = 10^{-3}s$. It can be concluded that the Standard Projection is not able to give good results with large time steps, because the splitting introduces spurious dissipation, and therefore leads to significant numerical errors. Moreover, the effect of numerical dissipation disappears for the Standard Projection, when small time steps are chosen, at the cost of increased computational time.

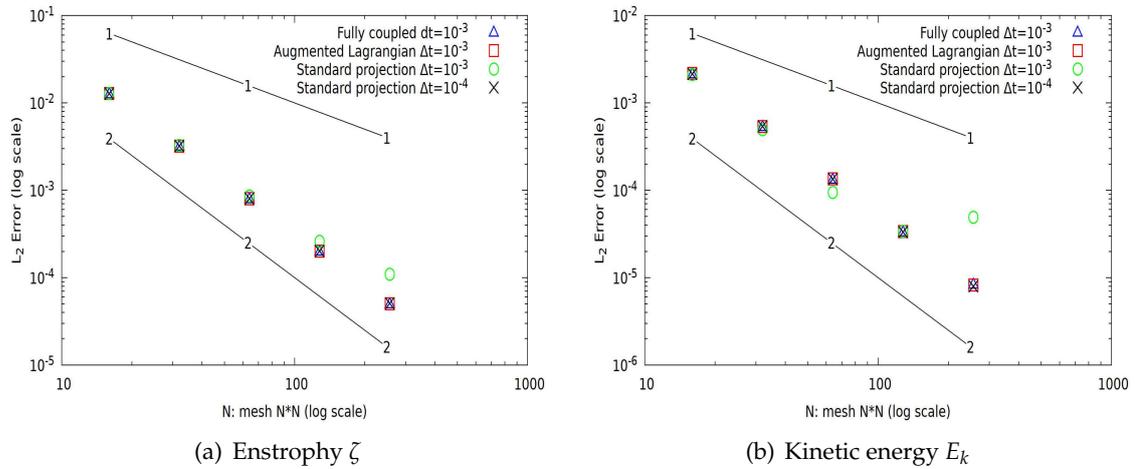


Figure 10: Kinetic energy E_k and Enstrophy ζ conservations and accuracy of the different methods in the viscous Taylor-Green test case.

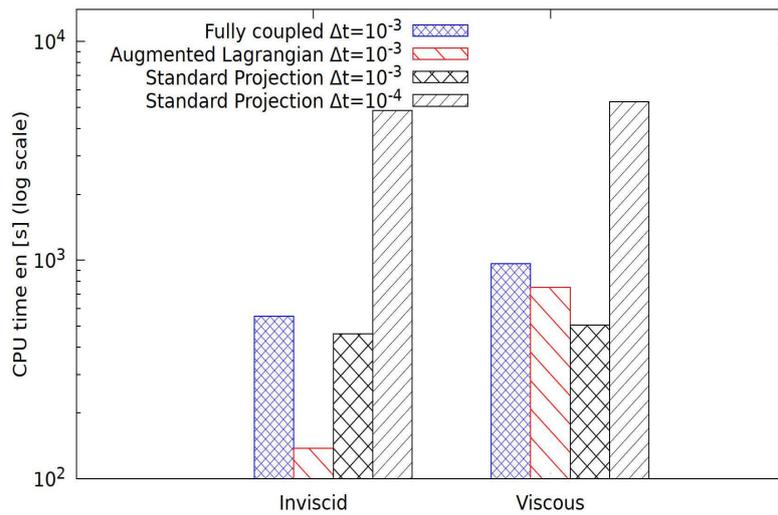


Figure 11: CPU time(s) of Navier-Stokes solvers for Taylor-Green test case (fine mesh 256×256).

Fig. 11 reports the CPU time (in second) for the three different solvers of Navier-Stokes in the viscous and inviscid cases, on the finest mesh 256×256 . Obtaining the same quality of numerical solutions whatever the studied case (viscous and inviscid), it is seen that the Standard Projection method is the most expensive in CPU time compared to the two other methods. We also observe for both cases that the Augmented Lagrangian method is faster than the Fully Coupled method. This can be explained by the fact that the two solvers need one iteration to converge to the exact solution, and that the size of

the system to be solved in the case of the Fully Coupled method is larger than that of the Augmented Lagrangian. In addition, the results show that the Fully Coupled method saves 84% (inviscid case) and 63% (viscous case) of the CPU time compared to the Standard Projection, while consuming more than 75% (inviscid case) and 22% (viscous case) of the CPU time compared to the Augmented Lagrangian.

4.3 Two-phase flows: Rising bubble

The rise of a two-dimensional bubble is studied numerically using the Fully Coupled approach with the PCD preconditioner, the Adaptive Augmented Lagrangian with $r = 10^3$ and the Standard Projection method. Two configurations are studied : the first corresponds to a numerical benchmark proposed by Hysing and coworkers [21]. Initially, a bubble of diameter $D = 0.5m$ is centered in a rectangular $2m$ high and $1m$ wide domain. Dirichlet and symmetric boundary conditions are used for the horizontal and the vertical boundaries respectively. The bubble is driven up by a constant gravity $g = 0.98m \cdot s^{-2}$ in the negative y direction, while the deformation of the bubble is counterbalanced by the surface tension, whose coefficient σ is set to $1.98N \cdot m^{-1}$. The first case ($\rho_1/\rho_2 = 10^2, \mu_1/\mu_2 = 10^1$) was simulated, but in this case all the methods give the same results as those of Hysing.

The second configuration is characterized by high density and viscosity ratios, respectively ($\rho_1/\rho_2 = 10^3$ and $\mu_1/\mu_2 = 10^2$). The main interest of this case is to compare the time-to-solution of the three methods. To provide detailed comparisons, three time steps are chosen : $\Delta t = 10^{-3}s, 5 \cdot 10^{-3}s$ and $10^{-2}s$ while target residual is kept constant ($\epsilon = 10^{-7}$ for the BiCGSTAB(2) solver).

In the following, the temporal evolution of two quantities will be considered in order to compare qualitatively the different methods.

- Center of mass Y_c

$$Y_c = \frac{\int_{\Omega_2} y \, dx}{\int_{\Omega_2} 1 \, dx}, \quad (4.14)$$

- Vertical velocity U_c

$$U_c = \frac{\int_{\Omega_2} \mathbf{U} \, dx}{\int_{\Omega_2} 1 \, dx}. \quad (4.15)$$

Here Y_c denotes the center of mass of the bubble, U_c denotes the velocity at the center of mass of the bubble, and Ω_2 represents the subdomain occupied by the bubble.

Before comparing the three solvers, the solutions computed on the finest grid ($h = 1/320$) with the Fully Coupled method are compared to the corresponding results of the codes TP2D (group 1), FreeLIFE (group 2) and MoonNMD (group 3) given by Hysing et al. [21]. As can be seen in Fig. 12(b), the estimate of the center of mass by the Fully Coupled method is in good agreement with the references values. Regarding the rise

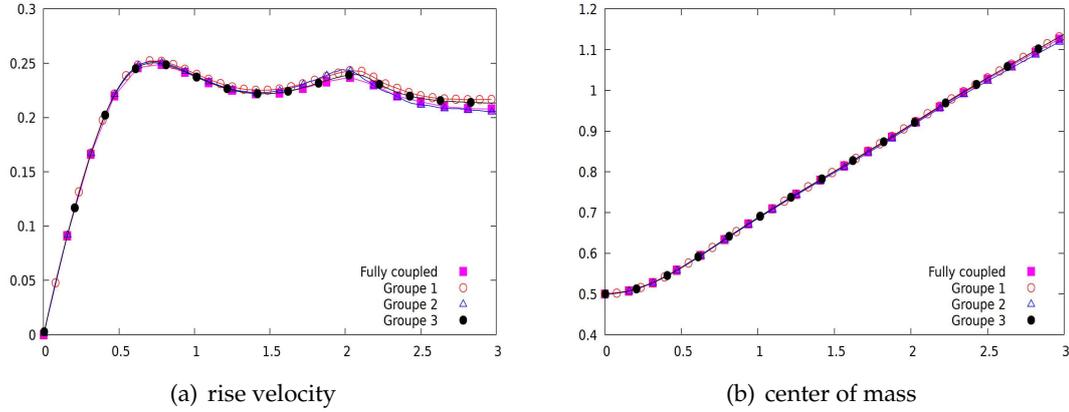


Figure 12: Vertical velocity and center of mass for test case 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$). Comparisons between the Fully Coupled method and all groups of Hysing et al. [21].

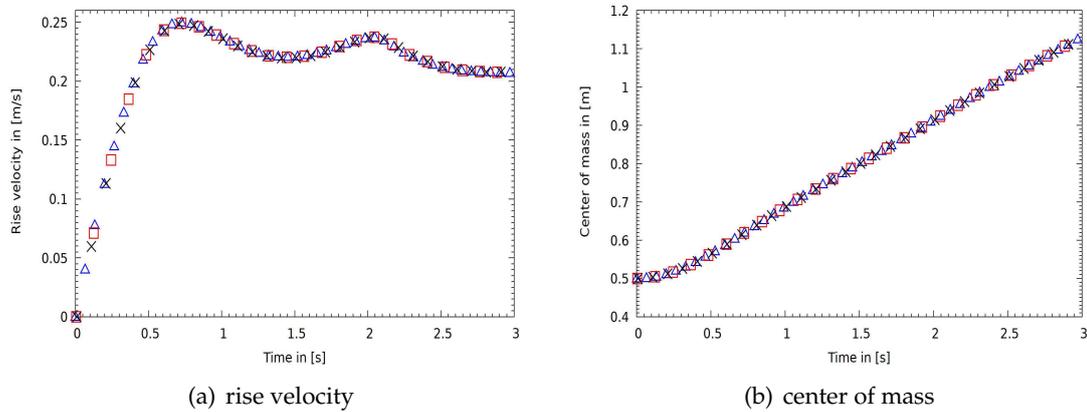


Figure 13: Vertical velocity and center of mass for test case 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$) obtained by the Fully Coupled method with several time steps. $\Delta t = 10^{-3}$ s (represented by triangles), $\Delta t = 5 \cdot 10^{-3}$ s (represented by squares) and $\Delta t = 10^{-2}$ s (represented by crosses).

velocity of the bubble, the solutions seem to deviate from each other starting at $t = 1 \cdot s$. At long times ($t = 2 \cdot 5$ s), the solution of the Fully Coupled method and the reference of group 2 seem to converge although it is difficult to discriminate between methods in this case.

Figs. 13, 14 and 15 show the vertical velocity and the center of mass of the bubble, obtained on the fine mesh 320×640 with FC, ALL and PR methods. The solutions provide results in good agreement with those of Hysing et al. [21], although they are not presented here. Also, the simulations show that the results obtained by the Fully Coupled and the Augmented Lagrangian methods are insensitive to the time step. Indeed, for these two methods the rise velocity and the center of mass profiles are qualitatively similar

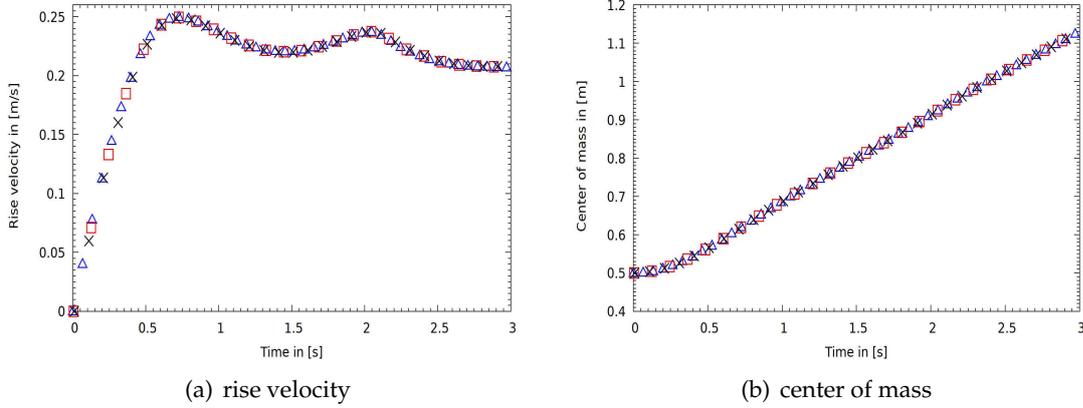


Figure 14: Vertical velocity and center of mass for test case 2 ($\rho_1/\rho_2=10^3, \mu_1/\mu_2=10^2$) obtained by the Adaptive Augmented Lagrangian ($r=10^3$) with several time step. $\Delta t=10^{-3}s$ (represented by triangles), $\Delta t=5\cdot 10^{-3}s$ (represented by squares) and $\Delta t=10^{-2}s$ (represented by crosses).

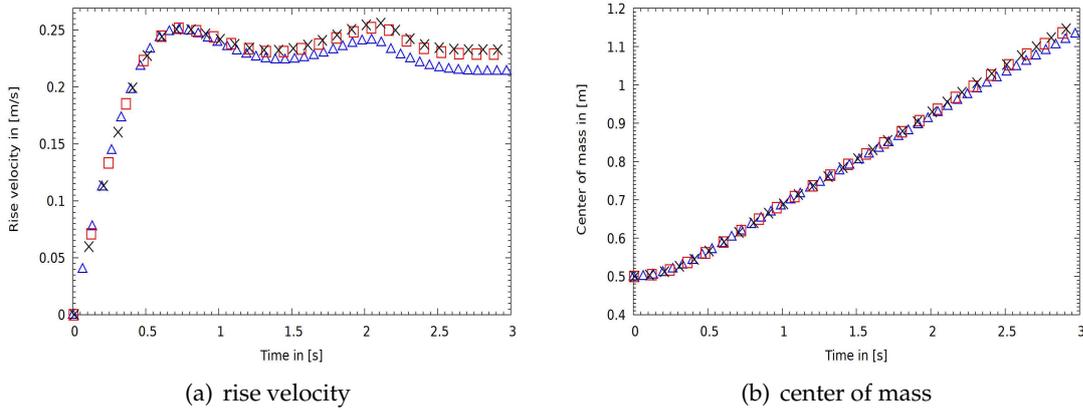


Figure 15: Vertical velocity and center of mass for test case 2 ($\rho_1/\rho_2=10^3, \mu_1/\mu_2=10^2$) obtained by the Standard Projection with several time step. $\Delta t=10^{-3}s$ (represented by triangles), $\Delta t=5\cdot 10^{-3}s$ (represented by squares) and $\Delta t=10^{-2}s$ (represented by crosses).

regardless of the time step. This is in contrast to the projection method, for which large time steps clearly induce significant errors on the rise velocity. Fig. 16 compares the shape of the bubble at $t=3s$ for all three methods using different time steps. It can be observed from these figures that the Fully Coupled and the Augmented Lagrangian methods are more accurate than the Standard Projection method. Indeed, in the resolved regions of the bubble, the overall shape of the bubble is converged, and the filamentary regions resemble the reference results [21]. The solutions were not precisely compared with those of Hysing, because even in the case of Hysing there is a significant dispersion between the methods, none of which can serve as a reference solution. The overall shape of the

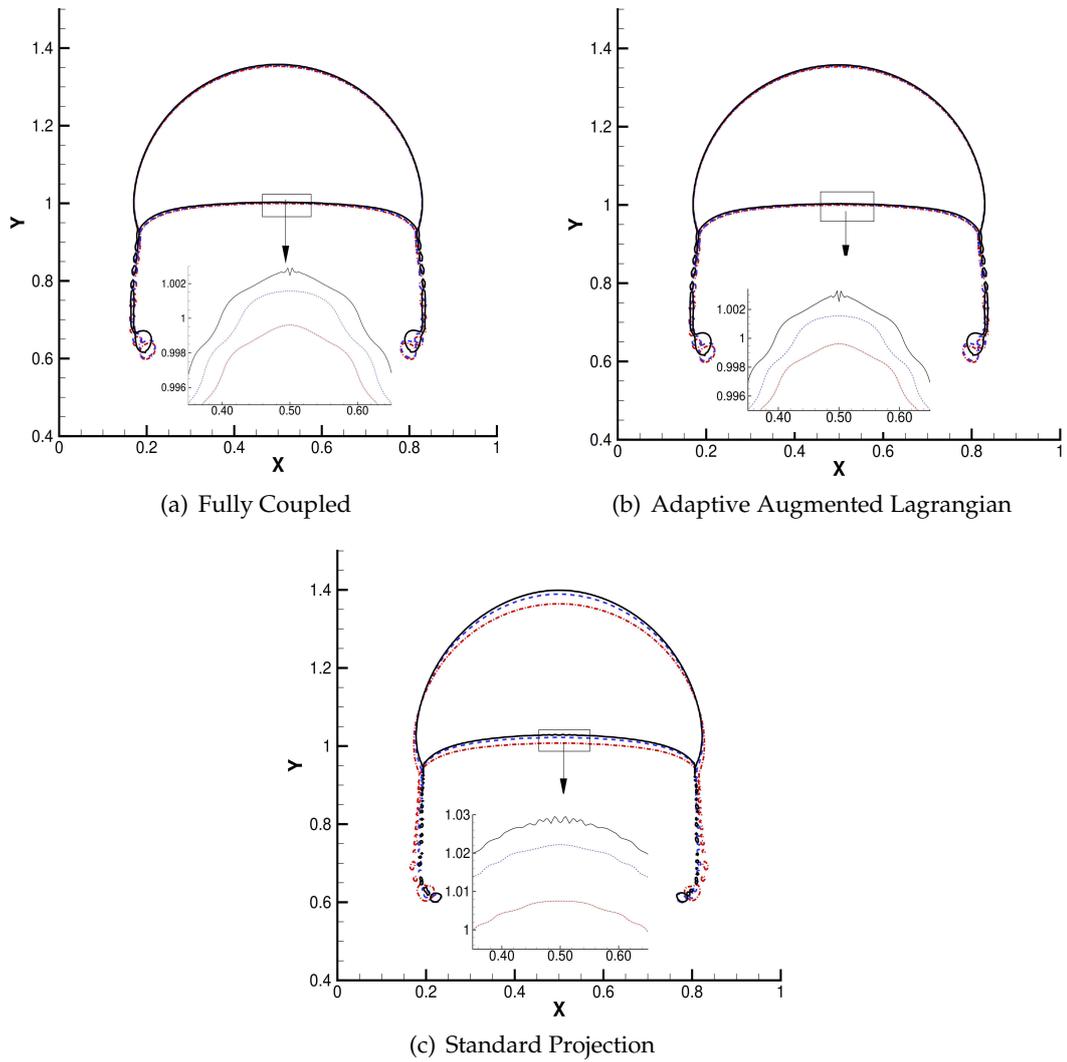


Figure 16: Bubble shapes at time $t = 3s$ for test case 2 ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$) with several time steps. $\Delta t = 10^{-3}s$ (represented by dash-dotted line), $\Delta t = 5 \cdot 10^{-3}s$ (represented by dashed lines) and $\Delta t = 10^{-2}s$ (represented by solid lines).

bubble however is similar to that of Hysing, whereas for the projection method, a large difference is noticed when the time step is changed, in the resolved regions but also on the point of break up. For the second case ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$), we conclude that the Fully Coupled and the Augmented Lagrangian methods are able to give the best results with large timesteps, which is not possible for the projection method.

In the following, the CPU times necessary for each method to simulate the rising bubble are considered. These CPU times are shown in Fig. 17. For the three time steps, the Augmented Lagrangian method is the slowest to converge. On the other hand, the dif-

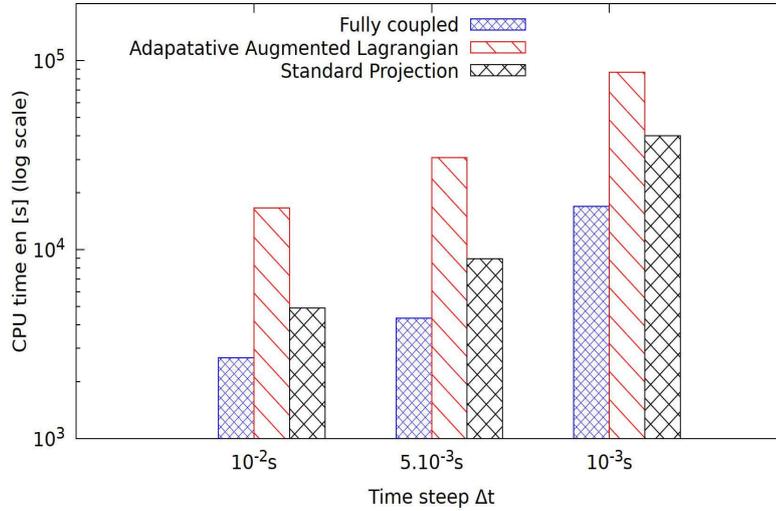


Figure 17: CPU time(s) of Navier-Stokes solvers for rising bubble test case 2 (fine mesh 320×640).

ference between the Fully Coupled and the Standard Projection method is significant, as well for the small time step $\Delta t = 10^{-3}s$ the Fully Coupled method saves 80% of the CPU time compared to the Augmented Lagrangian, while it saves 57% more time than the Standard Projection. These large differences can be explained by the fact that the Augmented Lagrangian and Standard Projection take a lot of iterations to converge. As larger density and viscosity ratios are considered, the use of a very large value of the parameter r is required to satisfy the incompressibility constraint. This leads to ill-conditioned linear system, hence the high cost of the Augmented Lagrangian method. For the Standard Projection, the cost of the computation is related to the resolution of the Helmholtz's equation with variable coefficients.

4.4 Two-phase flow: Free fall of a dense square

The present test case aims at checking the ability of the Fully Coupled method to simulate multiphase flows with high density and viscosity ratios, compared to other methods like the Augmented Lagrangian and the Standard Projection. The two-dimensional free fall of a dense square is tested. Initially, a rigid square of side $L = 0.00125m$, density $\rho_2 = 10^7 kg \cdot m^{-3}$ and dynamic viscosity $\mu_2 = 10^5 Pa \cdot s$ is released without velocity in a quiescent gas. Gravity is set at $g = 9.81 m \cdot s^{-2}$ while the surface tension coefficient σ is set to zero. The square is centered at $(0.05m, 0.175m)$ in a rectangular cavity full of gas whose density and viscosity are $\rho_1 = 1 kg \cdot m^{-3}$ and $\mu_1 = 1 Pa \cdot s$ respectively. The dimensions of the cavity are $0.2m$ high and $0.1m$ wide. The interest of this case lies in the possibility of comparing numerical solutions (Eq. (4.15)) to an exact reference value for the falling velocity $V_c = -gt$, obtained analytically by neglecting the effect of the gas on the solid. Considering

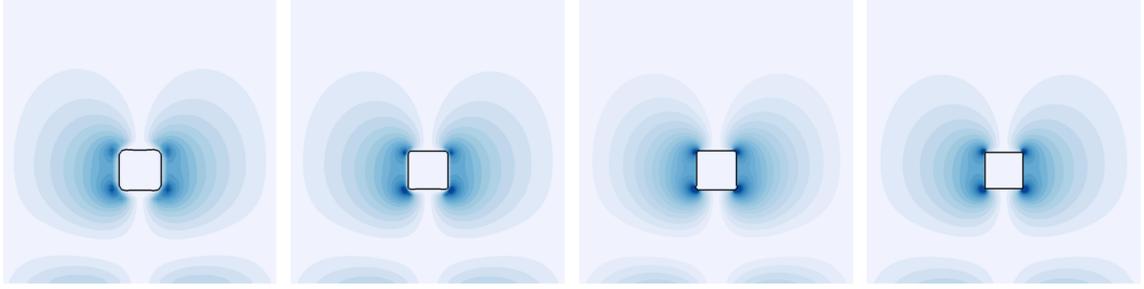


Figure 18: Adaptive Augmented Lagrangian method ($r=10^3$): vorticity magnitude and isoline ($C=0.5$) for the free fall of a dense cylinder with 64×128 , 128×256 , 256×512 and 512×1024 meshes.

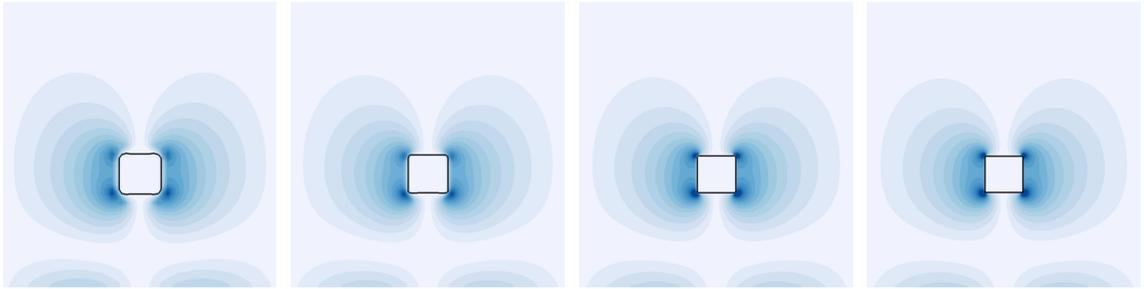


Figure 19: Fully Coupled method: vorticity magnitude and isoline ($C=0.5$) for the free fall of a dense cylinder with 64×128 , 128×256 , 256×512 and 512×1024 meshes.

analytical solutions as the appropriate reference solution, the errors committed in this test are evaluated using the definition (4.13), where q_t^{exact} is the falling velocity.

The case is simulated on four Cartesian grids (64×128 , 128×256 , 256×512 and 512×1024), with a constant time step $\Delta t = 5 \cdot 10^{-5} s$ and a residual of $\epsilon = 10^{-7}$ for the BiCGSTAB(2). Figs. 18 and 19 illustrate the simulation results for the four grids at time $t = 0.15 s$ after 3000 temporal iterations with the AAL and FC methods. The corresponding vorticity magnitude and isoline $C = 0.5$ of the color function at the interface are presented. It can be seen that the Fully Coupled and the Augmented Lagrangian approaches give qualitatively similar results for the same time step ($\Delta t = 5 \cdot 10^{-5} s$). On the contrary, the Standard Projection method requires a smaller time step ($\Delta t = 5 \cdot 10^{-6} s$) to obtain a comparable result. Despite this small time step, the simulation with the Standard Projection on a 128×256 grid fails to converge after a few thousand iterations because the correction step is not able to reduce the divergence of the predicted velocity.

Fig. 20 presents a comparison of the error levels reached by the FC and AAL solvers on the calculation of the vertical velocity at $t = 0.15 s$, as well as the corresponding CPU time. As shown in Fig. 20, the Fully Coupled method gives low relative errors, it also saves 25% of the CPU time compared to the Adaptive Augmented Lagrangian on the

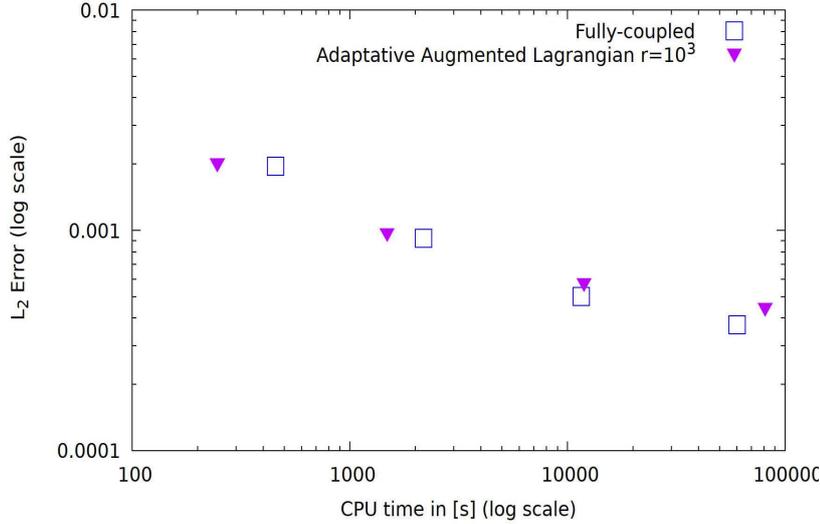


Figure 20: CPU time(s) and error (%) of the vertical velocity for free fall of a dense cylinder.

fine mesh 512×1024 . Nevertheless, the solutions obtained are always better with the Fully Coupled method than with the Augmented Lagrangian, even though the BiCG-Stab(2) error levels saturate as the number of iterations increases. However, the Standard Projection method requires a time step 10 times smaller than the other methods to achieve the same error levels. Finally, this method cannot converge on the mesh 128×256 .

5 Conclusion

Three methods were investigated to solve the Navier-stokes equations for incompressible single- and two-phase flows at large density and viscosity ratios. In the single-phase case, the results obtained for the lid-driven cavity case by the Fully Coupled method agree well with those of the literature, and exhibit second order convergence. The unsteady Green-Taylor vortex case has proven the ability of the Fully Coupled and Augmented Lagrangian methods to preserve energy within machine accuracy, whereas the Standard Projection methods introduces numerical dissipation. In two-phase cases, all methods were validated on two benchmarks : the rise of a two-dimensional bubble and the free fall of a dense object. These benchmarks highlighted the fact that the Fully Coupled and Adaptive Augmented Lagrangian methods are more efficient to solve two-phase flows (with larger time step Δt). On the contrary, the Standard Projection method has difficulties for large time steps, due to the accumulation of errors introduced by the pressure-velocity splitting. Large viscosity ratios however introduce a strong coupling between the velocity components, and future work will therefore focus on the preconditioning of the velocity block.

Acknowledgments

The authors gratefully acknowledge support from CINES (National Computing Center for Higher Education) and CCRT (National Computing Center of CEA) under project number A0032B06115. We are also grateful for I-SITE FUTURE, UPE and PIA under funding ANR-16-IDEX-0003.

A The BiCGSTAB(2) method

The biconjugate gradient stabilized method (BiCGSTAB(2)) is an iterative method introduced by Van Der Vorst [23] to solve efficiently large, sparse and nonsymmetric linear systems. Contrary to other BiCG variants such as the conjugate gradient squared algorithm (CGS) and the biconjugate gradient (BiCG), this specific method combines two BiCG (called even and odd) with a Generalized Conjugate Residual (GCR) algorithm. This solver is deemed robust and very stable, while reaching convergence quickly in spite of the higher computational cost of each iteration.

In practice, for more difficult problems, it is essential to combine the BiCGSTAB(2) algorithm with a suitable preconditioner P of Eq. (3.9) to accelerate the convergence of the iterative solver and to avoid possible numerical instabilities. Instead of solving $Ax=b$, $P^{-1}Ax=P^{-1}b$ is preferred, a system for which $\text{cond}(P^{-1}A) \ll \text{cond}(A)$. Therefore, in the BiCGSTAB(2) algorithm, every matrix-vector product $Ax=p$ will be followed by the calculation $P^{-1}y=p$, the same transformation must be applied to the second member $P^{-1}b=b$. Then, to solve the $Ax=b$ system by giving an initial guess $x^{(0)}$, BiCGSTAB(2) proceeds as below:

Algorithm: Preconditioned BiCGSTAB(2) (solve $Ax=b$ with the tolerance ϵ)

1. Initializations

$$\begin{aligned} x^{(0)} &\in \mathbb{R}^N \\ b &= P^{-1}b \\ p &= Ax^{(0)} \\ q &= P^{-1}p \\ r^{(0)} &= b - q \\ \tilde{r}^{(0)} &= p_0 = r^{(0)} \\ \rho_0^{(0)} &= \omega_2^{(0)} = 1, \alpha^{(0)} = 0 \end{aligned}$$

2. For $k=1,2,3,\dots$ until convergence do

$$\rho_0^{k-1} = -\omega_2^{k-1} \rho_0^{k-1}$$

Even step

$$\rho_1^k = (\tilde{r}^{(0)}, r^{k-1})$$

$$\begin{aligned}
\beta^k &= \alpha^{k-1} \rho_1^k / \rho_0^{k-1}, \quad \rho_0^k = \rho_1^k \\
p &= Au^{k-1} \\
v^k &= P^{-1} p \\
\gamma^k &= (\tilde{r}^{(0)}, v^k), \quad \alpha^k = \rho_0^k / \gamma^k \\
r^{k+1} &= r^k - \alpha^k v^k \\
p &= Ar^k \\
s^k &= P^{-1} p \\
x^{k+1} &= x^k + \alpha^k u^k
\end{aligned}$$

Odd step

$$\begin{aligned}
\rho_1^k &= (\tilde{r}^{(0)}, s^k), \quad \beta^k = \alpha^k \rho_1^k / \rho_0^k \\
v^k &= s^k - \beta^k v^k \\
p &= Av^k \\
w^k &= P^{-1} p \\
\gamma^k &= (\tilde{r}^{(0)}, w^k), \quad \alpha^k = \rho_0^k / \gamma^k \\
u^k &= r^k - \beta^k u^k \\
r^k &= r^k - \alpha^k v^k \\
s^k &= s^k + \alpha^k w^k \\
p &= As^k \\
t^k &= P^{-1} p
\end{aligned}$$

GCR part

$$\begin{aligned}
\omega_1^k &= (r^k, s^k), \quad \mu^k = (s^k, s^k), \quad v^k = (s^k, t^k) \\
\tau^k &= (t^k, t^k), \quad \omega_2^k = (r^k, t^k), \quad \tau^k = \tau^k - \frac{v^k v^k}{\mu^k} \\
\omega_2^k &= (\omega_2^k - v^k \omega_1^k / \mu^k) / \tau^k, \quad \omega_1^k = (\omega_1^k - v^k \omega_2^k / v^k) / \mu^k \\
x^k &= x^k + \omega_1^k r^k + \omega_2^k s^k + \alpha^k u^k \\
r^k &= r^k - \omega_1^k s^k - \omega_2^k t^k \\
u^k &= u^k - \omega_1^k v^k - \omega_2^k w^k
\end{aligned}$$

3. **test convergence**
If residual $\leq \epsilon$ exit
end if
end for

B The Richardson extrapolation

In order to compute the convergence rate of the numerical solutions, Richardson extrapolation [29] is used. It allows evaluating the convergence order α of the numerical method, but also obtaining an approximation f^{ex} of the exact solution f , without the need for an analytical solution to the problem. For this purpose, we choose generally (at least) three

meshes (with size grids noted h_1 , h_2 and h_3 such as $h_1/h_2 = h_2/h_3$) in the convergence region.

The Richardson extrapolation can be a useful method for extracting the convergence order and values extrapolated to the higher order. So, let f^{ex} be the extrapolated quantity, f_{h_i} the numerical solution on the grid of size h_i , α the convergence order and C_α a coefficient depending on α . In practice, using three grids with $h_1/h_2 = h_2/h_3$, the value of the convergence order is given as follows:

$$\alpha = \frac{\ln[(f_{h_1} - f_{h_2}) / (f_{h_2} - f_{h_3})]}{\ln(h_1/h_2)}. \quad (\text{B.1})$$

By using Eq. (B.1), the extrapolated quantity f^{ex} is given by the following:

$$f^{ex} = f_{h_3} - C_\alpha h_3^\alpha, \quad (\text{B.2})$$

where

$$C_\alpha = \frac{f_{h_2} - f_{h_3}}{f_{h_2}^\alpha - f_{h_3}^\alpha}. \quad (\text{B.3})$$

References

- [1] M. Benzi and B. Uçar. Block triangular preconditioners for M-matrices and Markov chains. *Electronic Transactions on Numerical Analysis*, 26:209–227, 2007.
- [2] N. Bootland, A. Bentley, C. Kees, and A. Wathen. Preconditioners for Two-Phase Incompressible Navier-Stokes Flow. *SIAM Journal on Scientific Computing*, 41(4):B843–B869, 2019.
- [3] J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *Journal of Computational Physics*, 100(2):335–354, 1992.
- [4] J. W. Cahn and J. E. Hilliard. Free energy of a nonuniform system. I. Interfacial free energy. *The Journal of Chemical Physics*, 28(2):258–267, 1958.
- [5] J. Cahouet and J.-P. Chabard. Some fast 3D finite element solvers for the generalized Stokes problem. *International Journal for Numerical Methods in Fluids*, 8(8):869–895, 1988.
- [6] J.-P. Caltagirone and J. Breil. Sur une méthode de projection vectorielle pour la résolution des équations de Navier-Stokes. *Comptes Rendus de l'Académie des Sciences – Series IIB – Mechanics-Physics-Astronomy*, 327(11):1179–1184, 1999.
- [7] C.-Y. Ku. A novel method for solving ill-conditioned systems of linear equations with extreme physical property contrasts. *Computer Modeling in Engineering and Sciences*, 96:409–434, 2013.
- [8] A.-J. Chorin. Numerical solution of the Navier-Stokes equations. *Mathematics of Computation*, 22(104):745–762, 1968.
- [9] C. C. Douglas, L. Lee, and M.-C. Yeung. On solving ill conditioned linear systems. *Procedia Computer Science*, 80:941–950, 2016. International Conference on Computational Science 2016, ICCS 2016, 6-8 June 2016, San Diego, California, USA.
- [10] M.-S. Dodd and A. Ferrante. Fast pressure-correction method for incompressible two-fluid flows. *Journal of Computational Physics*, 273:416–434, 2014.
- [11] S. David, E. Howard and J.-W. Andrew. Finite elements and fast iterative solvers: With applications in incompressible fluid dynamics. 01 2006.

- [12] H. Elman and S.-R. Tuminar. Boundary conditions in approximate commutator preconditioners for the Navier-Stokes equations. *Electronic Transactions on Numerical Analysis*, 35:257–280, 2009.
- [13] D.-R. Falgout and U.-M. Yang. hypre: A library of high performance preconditioners. In Peter M. A. Sloot, Alfons G. Hoekstra, C. J. Kenneth Tan, and Jack J. Dongarra, Eds., *Computational Science – ICCS 2002*, pages 632–641, Springer Berlin Heidelberg, 2002.
- [14] M. Fortin and R. Glowinski. Augmented Lagrangian methods: Applications to the numerical solution of boundary-value problems. *ZAMM – Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 65(12):622–622, 1985.
- [15] D. Fuster. An energy preserving formulation for the simulation of multiphase turbulent flows. *Journal of Computational Physics*, 235:114–128, 2013.
- [16] K. Goda. A multistep physics technique with implicit difference schemes for calculating two- or three-dimensional cavity flows. *Journal of Computational Physics*, 30:76–95, 1979.
- [17] J.-L. Guermond. Remarques sur les méthodes de projection pour l’approximation des équations de Navier-Stokes. *Numer. Math.*, 67(4):465–473, 1994.
- [18] J.-L. Guermond and A. Salgado. A splitting method for incompressible flows with variable density based on a pressure poisson equation. *Journal of Computational Physics*, 228:2834–2846, 2009.
- [19] R. Guo and Y. Xu. Efficient, accurate and energy stable discontinuous Galerkin methods for phase field models of two-phase incompressible flows. *Communications in Computational Physics*, 26:1224–1248, 2019.
- [20] F.-H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The Physics of Fluids*, 8(12):2182–2189, 1965.
- [21] S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, S. Hysing, and L. Tobiska. Quantitative benchmark computations of two-dimensional bubble dynamics. *International Journal for Numerical Methods in Fluids*, 60:1259–1288, 2009.
- [22] C. Kahle, J. Bosch, and M. Stoll. Preconditioning of a coupled Cahn-Hilliard Navier-Stokes system. *Communications in Computational Physics*, 23(2):603–628, 2018.
- [23] D.-C. Sorensen, J.-J. Dongarra, L.-S. Duff, and H. A. Vander Vorst. *Numerical Linear Algebra for High Performance Computers*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998.
- [24] P. Lubin, T. N. Randrianarivelo, J.-P. Caltagirone, and S. Vincent. An adaptive augmented Lagrangian method for three-dimensional multi-material flows. *Computers and Fluids*, 33:1273–1289, 2004.
- [25] I. Kataoka. Local instant formulation of two-phase flow. *International Journal of Multiphase Flow*, 12(5):745–758, 1986.
- [26] G.-H. Golub, M. Benzi, and J. Liesen. Numerical solution of saddle point problems. *Acta Numerica*, 14:1–137, 2005.
- [27] M. Benzi and M. Olshanskii. An augmented Lagrangian-based approach to the oseen problem. *SIAM Journal on Scientific Computing*, 28:2095–2113, 2006.
- [28] M. Cai, A. Nonaka, J. B. Bell, B. E. Griffith, and A. Donev. Efficient variable-coefficient finite-volume Stokes solvers. *Communications in Computational Physics*, 16(5):1263–1297, 2014.
- [29] M. Marc, N. Xavier, G. Stéphane, and G. Stéphane. Benchmark solution for a three-dimensional mixed-convection flow, Part 2: Analysis of Richardson extrapolation in the presence of a singularity. *Numerical Heat Transfer*, 60:346–369, 2011.
- [30] B. Olivier and P. Roger. Benchmark spectral results on the lid-driven cavity flow. *Computers and Fluids*, 27:421–433, 1998.

- [31] C.-H. Bruneau, P. Angot, and P. Fabrie. A penalization method to take into account obstacles in viscous flows. *Numerische Mathematik*, 81:497–520, 1999.
- [32] Y. Saad. ILUT: A dual threshold incomplete LU factorization. *Numerical Linear Algebra with Applications*, 1(4):387–402, 1994.
- [33] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.
- [34] C.-W. Shu, W.-S. Don, D. Gottlieb, O. Schilling, and L. Jameson. Numerical convergence study of nearly incompressible, inviscid Taylor-Green vortex flow. *Journal of Scientific Computing*, 24(1):1–27, 2005.
- [35] K. Stüben. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics*, 128(1):281–309, 2001.
- [36] R. Témam. Sur l’approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (i). *Archive for Rational Mechanics and Analysis*, 32(2):135–153, 1969.
- [37] S. Vincent, A. Sarthou, J. P. Caltagirone, F. Sonilhac, P. Février, C. Mignot, and G. Pianet. Adaptive augmented Lagrangian techniques for simulating unsteady multiphase flows. 07 2007.
- [38] S. Vincent, A. Sarthou, J. P. Caltagirone, F. Sonilhac, P. Février, C. Mignot, and G. Pianet. Augmented Lagrangian and penalty methods for the simulation of two-phase flows interacting with moving solids. Application to hydroplaning flows interacting with real tire tread patterns. *Journal of Computational Physics*, 230(4):956–983, 2011.
- [39] D. L. Youngs. Time-dependent multi-material flow with large fluid distortion. In K. W. Morton and M. J. Baines, Eds., *Numerical Methods for Fluid Dynamics*, pages 273–285. Academic Press, 1982.

Annexe G

Article publié dans Fluids MDPI

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/348233181>

Monolithic Solvers for Incompressible Two-Phase Flows at Large Density and Viscosity Ratios

Article in *Fluids* · January 2021

DOI: 10.3390/fluids6010023

CITATION

1

READS

175

3 authors:



Elouafa Mohamed

Université Gustave Eiffel

5 PUBLICATIONS 3 CITATIONS

SEE PROFILE



Stéphane Vincent

Université Gustave Eiffel

258 PUBLICATIONS 2,949 CITATIONS

SEE PROFILE



Vincent Le Chenadec

CentraleSupélec

27 PUBLICATIONS 332 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



multiscale interface tracking [View project](#)



Experimental & Numerical studies on liquid-liquid mass transfer of species with very high Sc ($0 \sim 10^3$). [View project](#)

Article

Monolithic Solvers for Incompressible Two-Phase Flows at Large Density and Viscosity Ratios

Mohamed El Ouafa * , Stephane Vincent and Vincent Le Chenadec 

MSME, University Gustave Eiffel, UMR CNRS 8208, University Paris-Est Créteil, 5 Boulevard Descartes, Champs Sur Marne, 77454 Paris, France; stephane.vincent@u-pem.fr (S.V.); vincent.lechenadec@u-pem.fr (V.L.C.)

* Correspondence: mohamed.elouafa@u-pem.fr

Abstract: In this paper, we investigate the accuracy and robustness of three classes of methods for solving two-phase incompressible flows on a staggered grid. Here, the unsteady two-phase flow equations are simulated by finite volumes and penalty methods using implicit and monolithic approaches (such as the augmented Lagrangian and the fully coupled methods), where all velocity components and pressure variables are solved simultaneously (as opposed to segregated methods). The interface tracking is performed with a Volume-of-Fluid (VOF) method, using the Piecewise Linear Interface Construction (PLIC) technique. The home code Fugu is used for implementing the various methods. Our target application is the simulation of two-phase flows at high density and viscosity ratios, which are known to be challenging to simulate. The resulting strategies of monolithic approaches will be proven to be considerably better suited for these two-phase cases, they also allow to use larger time step than segregated methods.

Keywords: velocity-pressure coupling; fully coupled solvers; augmented lagrangian; two-phase flows; saddle point; projection method; preconditioning; smooth VOF



Citation: El Ouafa, M.; Stephane, V.; Le Chenadec, V. Monolithic Solvers for Incompressible Two-Phase Flows at Large Density and Viscosity Ratios. *Fluids* **2021**, *6*, 23. <https://doi.org/10.3390/fluids6010023>

Received: 2 December 2020

Accepted: 18 December 2020

Published: 5 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Understanding the physics of unsteady incompressible two-phase flows at large contrast ratios [1] and their numerical modeling via Navier-Stokes equations has become an important tool that allows understanding further physics of multiphase flows and therefore controls the behavior of these flows in several industrial and environmental applications. However, they remain among the most difficult to simulate numerically for two correlated reasons: the difficulties in the mathematical formalism of these equations resulting from the velocity-pressure coupling [2] and the numerical processing of the interfaces [3] which can be decoupled from solving the Navier-Stokes system. On the one hand, the numerical processing of the interfaces, in particular the interface tracking, which is indispensable for each time iteration, is required for the modeling of discontinuities in physical properties across the interface, in particular in the presence of large density and viscosity ratios, together with the conservation of mass. On the other hand, the main difficulty lies in the linear system that arises from these equations. This is a challenging problem to invert because of (1) the incompressibility constraint that results in a saddle-point structure (zero diagonal pressure block), and (2) large viscosity ratios that strongly couple the velocity components in the interface vicinity.

In the past two decades, much progress has been achieved to treat the different aspects related to the presence of complex shape interfaces and capillary phenomena. On the one hand, Sharp-Interface methods (SIM) considering the interface as a discontinuity, is still the most currently used by the scientific community to deal with two phase-flows. This class of methods fall along to one of the two following approaches: Front Tracking and Front Capturing methods. Front Tracking methods of Tryggvason et al. [4] rely on the Lagrangian advection of a chain of markers connected by segments, where the position of the interface is thus represented by these markers. This method has the advantage of

providing an explicit description of the position of the interface, and provides accurate results since it is described at a smaller scale than the mesh associated with the Eulerian velocities (sub-grid resolution). However, problems of connectivity and topology changes make this method difficult to use, especially in 3D. The Interface capturing methods on the other hand consist of capturing the presence of the front directly by using a characteristic phase function. In these approaches, we can cite the level-set method developed by the mathematicians Stanley Osher and James Sethian [5,6], for which a regular description of the interface is used, represented by the zero of a regular function, the signed distance to the interface. Although this method is accurate for the computation of normals and curvature, it does not ensure the mass conservation due to the numerical diffusion of the advection schemes such as the WENO scheme [7], which can cause a modification of the position of the interface. In addition to this, these schemes require the use of a large stencil, which is computationally expensive. Conservative level-set method introduced by Ollson and Kreiss [8,9] is often used, requires the construction of an additional color function, by introducing a diffuse profile via a hyperbolic tangent function. This method conserves the mass of each phase, but require a reinitialization after each advection step. The Volume Of Fluid (VOF) method introduced by Hirt and Nichols [10], use a stiff description of the interface, for which the interface presence is explicitly taken into account by the volume fraction so called color function. The interface location is reconstructed by means of linear approximations in the cells [11]. This method remains very interesting since it allows ensuring a very volume conservation of each phase. However, it suffers from many limitations such as difficult access to the geometric characteristics such as curvatures and normals, and the generation of artificial deformation near the interface when its thickness is of the order of a cell. Finally, the CLSVOF [12] method which involves a coupling between the Level-Set and VOF methods, benefits from the advantages of both these methods. Indeed, the mass is well conserved, and the geometrical properties can be easily calculated. Although these important advantages are important, unfortunately, the implementation of this method remains difficult. A last class of front-capturing method is the Moment Of Fluid method [13] that can be viewed as an extension of the VOF with the addition of the centroid of the phase together with the volume fraction.

Another class of two-flow modeling approach, called the Diffuse Interface Method (DIM), relies on the definition of a physical interfacial thickness whereby the physical properties vary rapidly yet continuously. In the literature, a wide variety of this class of methods has been proposed. These include phase-field methods [14], based on the convection-diffusion equation of the phase, by introducing physical effects that govern interfaces. This method does not require any reinitialization to ensure mass conservation, and has proved also to be very accurate for computing curvature and surface tension forces. Nevertheless, their main drawback it is computationally more expensive, and the mesh size must be small enough to capture the smaller resolved interfacial structures. Another DIM approach is to use the Second Gradient (SG), a technique developed initially by van der Waals [15], and extended for the simulation of liquid-vapor flows with phase change by Didier Jamet [16]. The idea is to propose a smooth variation of the physical properties along the diffuse interface by holding into account the influence of the gradient of the density field and the double gradient of the velocity field. Even if it is challenging to model, this method can yield simulate complex coupled problems. However, both of these interface diffuse methods preserve the interface dynamics through the conservation and give a satisfactory description for the normal and curvatures. It has been demonstrated that at high resolutions, the VOF method yielded higher accuracy at a lower cost compared to diffuse interface methods (see [17]). In this work, our choice was to go with the VOF methods. The remedies for these drawbacks can be circumvented using another class of methods allowing to regularize the VOF. Constrained interpolation profile (CIP) methods of [18], consist of solving hyperbolic equations for the color function. Based on the CIP, the color function and its gradient are calculated to form a cubic interpolation function. This method is known for low numerical diffusivity and good stability. However, its

main drawback lies in the interface thickness that is not controlled. A recent regularized VOF method consists in introducing an original smooth volume of fluid function (SVOF) obtained by solving a low cost diffusion equation, as described in [19,20]. This numerical procedure has a priori two advantages. On the one hand, the spread of interfaces through which the physical properties vary strongly but continuously. On the other hand, as the physical properties are continuous, the convergence properties for the fully coupled solver become better, even in the presence of large contrast ratios. In addition, the application of the continuum surface tension force model [21] to SVOF significantly improves the representation of the approximation of surface tension forces.

Furthermore, the treatment of velocity-pressure coupling has been the subject of much work and is a relatively well-controlled problem concerning single-phase flows. On the contrary, in the context of two-phase flow modeling at large density and viscosity ratios, numerical difficulties appear, related to the presence of large interface distortions involving ill-conditioned linear systems. To deal with the velocity-pressure coupling challenge, two strategies can be used: segregated methods introduced independently by Chorin [22] and Temam [23], approximate the original system using time splitting, thereby resulting in two decoupled equations: one to update the velocity field and the other the pressure field although these methods stem from their simplicity and ease of implementation. They still suffer from serious limitations: the splitting operator introduces an error that reduces the temporal accuracy of the numerical solution [24], and the need for pressure boundary conditions that do not exist in the original problem and who are more sensitive to the ratio of densities through the interfaces. The implicit and monolithic approaches such as the augmented Lagrangian (AL) and the fully coupled (FC) methods, solve simultaneously all velocity components and pressure variables, hence preserving the consistency of the discretized system with the continuous equations (while the segregated methods have a time splitting error). The augmented Lagrangian has been developed by Fortin and Glowinski [25] for single-phase Stokes flows and extended by Vincent et al. [26,27] for two-phase flows. The advantages of the augmented Lagrangian method are numerous. The resolution is very robust, and it allows us to simulate a large number of complex unsteady two-phases problems, in which density and viscosity ratio may exceed 10^6 . Besides, this approach avoids imposing boundary conditions on the pressure field. The main drawback of the augmented Lagrangian is its numerical expense to achieve low residuals for the inversion of a linear system and divergence levels. Fully coupled methods [28–31], on the other hand, are purely algebraic, using specific various block preconditioning and Schur complements approximation techniques to solve a saddle point on both fields (velocity and pressure). The resulting strategies, although more difficult to implement and more expensive on a time step basis, prove to be more robust, and very effective for simulating two-phase flows with very large ratios of density and viscosity. They lead to having better convergence properties, with little CFL restrictions.

While the vast majority of related documented studies rely on explicit and segregated methods (such as projection methods), we describe here a fully coupled method for dealing with velocity-pressure coupling, to take advantage of its robustness, principally for the simulation of two-phase flows at large density and viscosity ratios, which has been little studied in the literature. Furthermore, it has been demonstrated that the fully-coupled solver hence preserving the consistency of the discretized system with the continuous equations at large contrast ratio in most cases, which allows to simplify the implementation and to potentially overcome the momentum conserving method contributions (Sussman [32], Raessi [33], Le chenadec [34], Desjardins [35], Ménard [36], ...). To characterize the interface location, a PLIC VOF approach is used in the present work. The conclusions arising on solvers would be clearly applicable to level set, CLSVOF, phase field or MOF methods also.

This paper is organized as follows: In Section 2 the Naviers-Stokes equations are formulated for two-phase incompressible flows, and the temporal and spatial discretizations are presented. Section 3 is dedicated to numerical results for two-phase flows to show the advantages of the Fully coupled methods. In Section 4 concluding remarks are drawn.

2. Model and Numerical Methods

2.1. Penalized One-Fluid Model

In the framework of numerical simulation of two-phase incompressible flows involving non-miscible fluids, the one-fluid formulation, developed by Kataoka [37], is considered here. In each of the two fluid phases, the equations to be solved are the conservation of linear momentum, the conservation of volume (the phases are immiscible and incompressible), augmented with the capillary effect. Heat transfer phenomena will be neglected, and therefore we do not have to solve the energy equation. An additional equation will also be solved: the transport equation of a scalar field called the color function.

We consider an open bounded domain $\Omega \subset \mathbb{R}^3$, with boundary Γ . Let be \mathbf{u} the fluid velocity, p the pressure field, t the time, ρ the mixture density, \bar{T} viscous stress tensor, \mathbf{f} the momentum body force, \mathbf{F}_s the capillary term acting on the interface, modeled in this study by the continuum surface tension force model (CSF) [21] and \bar{B}_{penu} a tensor field for specifying boundary conditions, whose diagonal components tend to infinity along the boundary Γ and are identically zero inside the fluid domain Ω . The coupling between the one fluid model developed by Kataoka and a penalty method [38] (to treat different types of boundary conditions) give the penalized one-fluid model. In its conservative form, the equations governing the multiphase flow motion then read:

$$\frac{\partial \rho(C)\mathbf{u}}{\partial t} + \nabla \cdot (\rho(C)\mathbf{u} \otimes \mathbf{u}) + \bar{B}_{penu}(f(\mathbf{u}) - \mathbf{u}_\infty) = -\nabla p + \nabla \cdot \bar{T} + \mathbf{f} + \mathbf{F}_s, \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (2)$$

To characterize the topology of the interface and the fluid properties (mass density and viscosity), the color function C , which vary between 0 and 1, indicates the presence of the interface. It is just advected by the incompressible fluid velocity that is zero divergence under the assumption that no phase change occurs. As previously mentioned, only isothermal incompressible two-phase are considered here. The tracking of the spatio-temporal evolution of the colour function C , requires the resolution of the following advection equation:

$$\frac{\partial C}{\partial t} + \mathbf{u} \cdot \nabla C = 0. \quad (3)$$

To complete the one-fluid formulation, the mixture density $\rho(C^s)$ and the expression of the viscous stress tensor \bar{T} must be specified. As mass is an extensive quantity, the mixture density is defined according to the following constitutive law (arithmetic average, which allows satisfying the conservation of mass intrinsically):

$$\rho(C^s) = \rho_1 C^s + (1 - C^s)\rho_2, \quad (4)$$

where ρ_1 and ρ_2 are the densities of fluid 1 and 2. C^s is the smoothed color function ranging from 0 and 1 as C , whose spread around $C = 0.5$ is controlled. This function is obtained by performing a smoothing operation on the color function C by solving the following Helmholtz equation (the reader is referred to [19,20] for more information),

$$\nabla \cdot D \nabla C^{s,n+1} + C^{s,n+1} = C^{s,n} \quad (5)$$

where $D = L_i \Delta h^2$ is the diffusion coefficient, which is then defined as the product of the control volume Δh^2 and a numerical parameter L_i . It can be noticed that this parameter allows to ensure that the smoothed color function C^s spans a distance $L_i = k \Delta h$ on either side of the interface, where k is taken between 0.2 and 2 (see [19,20]). Note that the smoothing operation described above in Equation (5), is equivalent to replacing the interface discontinuity with a zone of adjustable thickness, through which the transition between the two phases has taken place. As a consequence, this method has some advantages. On the one hand, it avoids the generation of artificial deformation near the interface and provides

large enough discrete stencils to estimate the normal and the curvature of the interface. On the other hand, using this technique leads to better convergence properties for the implicit solvers, which proves to be more robust, when coping with large viscosity ratios that strongly couple the velocity components in the interface vicinity.

Concerning the viscous stress tensor $\overline{\overline{T}}$, Caltagirone and Vincent [39] proposed a new technique called “Viscous Penalty Method” of second order. The major interest of this formulation is that it allows distinguishing all the physical contributions reflecting the effects of compressibility, elongation, shearing, and rotation. The splitting of these contributions then makes it possible to act differently on each term to strongly impose the associated stress. After suitable decomposition, we assume the viscous stress tensor is given by:

$$\overline{\overline{T}} = \kappa \begin{pmatrix} \frac{\partial u}{\partial x} & 0 \\ 0 & \frac{\partial v}{\partial y} \end{pmatrix} + \zeta \begin{pmatrix} 0 & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & 0 \end{pmatrix} - \eta \begin{pmatrix} 0 & \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} & 0 \end{pmatrix} \quad (6)$$

where $\kappa = 2\mu$ for the elongation viscosity, $\zeta = 2\mu$ for the shearing viscosity, and $\eta = \mu$ for the rotation viscosity, the evaluation of the viscous operator $\nabla \cdot \overline{\overline{T}}$ involves in particular the estimates of the elongation viscosity $\kappa = 2\mu$ at the pressures nodes, and the pure shearing $\zeta = 2\mu$ and rotation $\eta = \mu$ viscosities at the center of the Cartesian grid cells. By using a mixed average for the viscosity, which was initially proposed by Benkenida and Magnaudet [40], and then Vincent et al. [41], the general jump conditions of the momentum equations across the interface are satisfied. In this situation, the diagonal viscous stress is proportional to an elongation viscosity which is determined as a linear function of the smoothed color function called C^s ,

$$\kappa_{i,j} = 2\mu_1 C_{i,j}^s + 2\mu_2 (1 - C_{i,j}^s) \quad (7)$$

As regards to the off diagonal components of the viscous stress tensor, the pure shearing ζ and rotation η viscosities will be computed by a harmonic average combined with a linear interpolation (Equation (9)) of the adjacent color functions located at the vertices of the cell (see Figure 1), as follows:

$$\zeta_{i,j} = 2\eta_{i,j} = 2 \times \left(\frac{\mu_1 \mu_2}{\mu_2 \tilde{C}_{i,j}^s + \mu_1 (1 - \tilde{C}_{i,j}^s)} \right) \quad (8)$$

With,

$$\tilde{C}_{i,j}^s = \frac{C_{i,j}^s + C_{i,j-1}^s + C_{i-1,j}^s + C_{i-1,j-1}^s}{4} \quad (9)$$

2.2. Discretization Schemes and Solvers

In the present work, the unsteady two-phase flow equations are approximated by finite volumes and penalty methods (to treat different types of boundary conditions), on a staggered mesh according to the MAC scheme of Harlow and Welch [42]. The discrete velocity and pressure variables are located in a staggered way, each having its own control volume, as shown in the Figure 1. This choice guarantees the consistency of the differential operators such as the divergence and the gradient, and it also avoids oscillations on the pressure field. Knowing that the solution of the advection Equation (3) is performed with the VOF method, using the Piecewise Linear Interface Construction (PLIC) technique. The resulting semi-discrete system of mass and momentum conservation equations is advanced in time, implicitly using a second-order integration scheme. In contrast, a centered scheme for the pressure gradient, divergence velocity, convective, and viscous terms is used. To avoid applying the Newton linearization and invert the full Jacobian, which

proves more efficient but more expensive for unsteady flow, the convective fluxes are linearized using Adams-Bashforth extrapolation as follows:

$$\mathbf{u}^{n+1} \otimes \mathbf{u}^{n+1} \approx \left(2\mathbf{u}^n - \mathbf{u}^{n-1} \right) \otimes \mathbf{u}^{n+1}. \tag{10}$$

Here, n refers to a time index associated with physical time $n\Delta t$ and Δt is assumed to be the constant time step of each solved iteration. So, after suitable linearization and discretization, and, as an implicit Fully Coupled (FC) resolution is targeted, the generic system is a nonsymmetric linear system of saddle point type which takes the following form:

$$\begin{pmatrix} \frac{1}{\Delta t} M_u^{(\rho)} + N_u^{(\rho)} + L_u^{(\mu)} & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ 0 \end{pmatrix} \tag{11}$$

or $Ax = \mathbf{b}$. Here, the matrix B_u is the discrete negative divergence and B_p^T represents the discrete pressure gradient operator, $M_u^{(\rho)}$ is the velocity mass matrix, $L_u^{(\mu)}$ is the discrete viscous velocity Laplacian, $N_u^{(\rho)}$ denotes the convective matrix and \mathbf{f} represents the right-hand side vector. Our strategy to solve this sparse linear system is to employ the BiCGStab(2) [43] solver on the entire system, by choosing compressed storage raw (CSR) structure in order to store only the non-null coefficients of the matrix, and then built an efficient preconditioner to accelerate the convergence of the iterative solver. This strategy can be established on the use of pressure convection-diffusion (PCD) preconditioning for the pressure block, and the block Gauss-Seidel preconditioner for the velocity block (See [44] for further details).

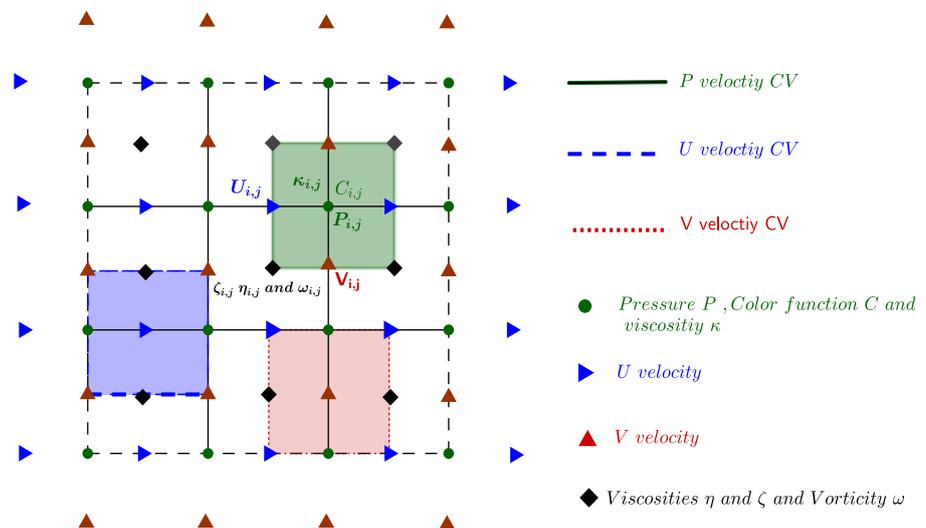


Figure 1. Staggered pressure and velocity unknowns in the global domain. the pressure $p_{i,j}$, the elongation viscosity $\kappa_{i,j}$ and the colour function $C_{i,j}$ are located at the nodes of the Eulerian grid, the horizontal velocity $u_{i,j}$ is defined on the left and right sides of the control volume (in green) associated with $p_{i,j}$ and vertical velocity $v_{i,j}$ is defined on the bottom and top sides of the control volume associated with $p_{i,j}$, the vorticity $\omega_{i,j}$, and the rotational $\eta_{i,j}$ and $\zeta_{i,j}$ shear viscosities are located at the vertices of the control volume associated with $p_{i,j}$.

3. Numerical Results

As mentioned in the introduction, different approaches have been used to solve the Navier-stokes equations. In the following, we have conducted different simulations of two-phase flow modeling, to demonstrate the accuracy and performance of our new fully coupled solver compared to existing resolution algorithms, and implemented in a home-made code (FUGU) developed at MSME Research Laboratory. The common point of these

simulations is that the high density and viscosity ratios seriously complicates the resolution, which constitutes a challenging task.

3.1. Free Fall of Dense Cylindre

The present test case aims at checking the ability of the fully coupled method to simulate multiphase flows with high density and viscosity ratios compared to the standard projection (SP) and adaptative augmented Lagrangian methods. The two-dimensional free fall of a dense cylinder is tested, which has a great interest in having an exact solution but also more severe parameters with high density and viscosity ratios. The initial configuration, consists of a rigid cylinder of radius $r = 0.0125$ m, density $\rho = \rho_2$ and dynamic viscosity $\mu = \mu_2$ which is released without velocity in air. Gravity is set at $g = -9.81 \text{ m} \cdot \text{s}^{-2}$ in the y direction, while the surface tension coefficient σ is set to zero. The cylinder is centered at (0.05 m, 0.15 m) in a rectangular cavity full of air whose density and viscosity are $\rho = \rho_1$ and $\mu = \mu_1$ respectively. The cavity is 0.2 m high and 0.1 m wide. The boundary conditions are slip lateral walls ($\mathbf{u} \cdot \mathbf{n} = 0$) and no-slip on the top and bottom walls ($\mathbf{t} \cdot (\nabla \mathbf{u} + \nabla^T \mathbf{u}) \cdot \mathbf{n} = 0$).

Concerning the numerical parameters, the simulations are carried out on five Cartesian grids (50×100 , 100×200 , 200×400 , and 400×800), with a residual of $\epsilon = 10^{-6}$ for the BiCGSTAB(2). As far as the time derivatives, a constant time step $\Delta t = 6.25 \times 10^{-5}$ s is chosen, and 2300 time steps are computed, corresponding to 0.144 s of the flow motion. The physical parameters for the test-cases are given in Table 1.

Table 1. Physical parameters describing the test cases. Here, ρ_2 and μ_2 refers to the fluid density and viscosity respectively inside the droplet, ρ_1 and μ_1 , refers to the density and viscosity of the surrounding fluid respectively and g is the gravity.

Test Case	ρ_1 [$\text{kg} \cdot \text{m}^{-3}$]	ρ_2 [$\text{kg} \cdot \text{m}^{-3}$]	μ_1 [$\text{Pa} \cdot \text{s}$]	μ_2 [$\text{Pa} \cdot \text{s}$]	g [$\text{m} \cdot \text{s}^{-2}$]	ρ_2/ρ_1	μ_2/μ_1
1	1.1768	10^5	1.85×10^{-5}	10^5	-9.81	10^5	10^{10}

The results obtained by the fully coupled and adaptative augmented Lagrangian solvers are presented in Figure 2 at $t = 0.144$ s. It can be seen that these method stays more robust when coping with large viscosity ratios. Hence, a good agreement is observed between both methods, with slight differences in the vorticity fields. Unfortunately, the standard projection method fails to converge towards the end of the simulation because the solver was unable to reach the desired residual to solve the Poisson equation with variable coefficients, and therefore the correction step was not able to reduce the divergence of the predicted velocity. The time step must, therefore, be decreased to ensure a correct quality of numerical solution, performed by the standard projection method. Nevertheless, significant differences can be observed from the field of vorticity magnitude.

3.1.1. Comments on the Falling Velocity and Center of Mass

To evaluate the accuracy of the Fully-coupled method, comparison of results with quantitative and qualitative results are now proposed. To do so, two distinct quantities will be used to describe their temporal evolution. These quantities are defined as follows:

- Center of mass Y_c

$$Y_c = \frac{\int_{\Omega_2} y \, dx}{\int_{\Omega_2} 1 \, dx} \tag{12}$$

- Vertical velocity U_c

$$U_c = \frac{\int_{\Omega_2} \mathbf{U} \, dx}{\int_{\Omega_2} 1 \, dx} \tag{13}$$

Here Y_c denotes the center of mass of the cylinder, U_c denotes the velocity at the center of mass of the cylinder, and Ω_2 represents the subdomain occupied by the cylinder. Since the

exact solution of the falling velocity ($U_c = -gt$) and center of mass ($Y_c = -gt^2/2 + 0.15$) are known, relative errors obtained during this test case can be evaluated using Equation (14).

$$Err_2^{rel}(q_t) = \left(\frac{\sum_{k=1}^{nt} \|q_t^{ref} - q_t\|^2}{\sum_{k=1}^{nt} \|q_t^{ref}\|^2} \right)^{1/2} \tag{14}$$

where q_t^{ref} is the reference quantity, q_t is the temporal evolution of quantity q , nt is the number of time steps, n is the mesh refinement level and h the cell size.

Time evolution of the falling velocity and center of mass are shown in Figure 3 for different grid refinements. This figure demonstrates a clear convergence of the numerical simulations to the exact solution even for coarse grids. The corresponding error between the numerical results and the exact solution is shown in Table 2. It is clear from this table that the error level is always low for the fully coupled numerical solutions. Even though errors are saturated (perhaps due to the fact that the simulation are given with a dense and viscous fluid around the cylinder whereas the theory is obtained in vacuum), the results are at least accurate to around 0.01% compared to the exact solution.

In order to be aware of the influence of the numerical methods and their discretization, the results given by the fully coupled method on the finer grid are compared to those of standard projection and adaptative augmented Lagrangian solvers, the performances of each solver are also evaluated. As shown in Table 3, the error levels are comparable for the Fully-coupled and augmented Lagrangian solver for the same time step $\Delta t = 6.25 \times 10^{-5}$ s. However, the fully coupled solver is faster because it saves 56% of the CPU time compared to the adaptative augmented Lagrangian. These large differences can be explained by the fact that the augmented Lagrangian takes a lot of iterations to converge, with a saturation of the BiCG-Stab(2) solver. Furthermore, the use of a very large value of the parameter r is required to satisfy the incompressibility constraint. This leads to an ill-conditioned linear system, hence inducing the high cost of the augmented Lagrangian method. This is in contrast with the projection method, which requires a time step ten times smaller than other methods to obtain the same quality of solution. This choice seems reasonable in order to reduce the numerical dissipation due to splitting operators. However, this brings the standard projection method to be the most expensive in CPU time.

Table 2. Free fall of a dense cylindre—Values of $Err_2^{rel}(U_c)$ and $Err_2^{rel}(Y_c)$ according to grid, obtained with Fully-coupled (FC) method, throughout the simulation, after 0.144 s.

Grid	Δt	$Err_2^{rel}(U_c)$	$Err_2^{rel}(Y_c)$
50 × 100	6.25×10^{-5} s	5.05×10^{-5}	8.41×10^{-4}
100 × 200	6.25×10^{-5} s	2.37×10^{-5}	5.74×10^{-4}
200 × 400	6.25×10^{-5} s	2.62×10^{-5}	2.38×10^{-4}
400 × 800	6.25×10^{-5} s	3.07×10^{-5}	2.21×10^{-4}

Table 3. Free fall of a dense cylindre—Values of $Err_2^{rel}(U_c)$ and $Err_2^{rel}(Y_c)$ obtained with FC, AAL, and SP at the end of the simulation for $t = 0.144$ s, on a grid of size 400 × 800. NC indicates the simulation stopped due to divergence of the solver.

Method	$Err_2^{rel}(U_c)$	$Err_2^{rel}(Y_c)$	CPU Time in [s]
FC with $\Delta t = 6.25 \times 10^{-5}$ s	5.05×10^{-5}	2.21×10^{-4}	44,986
AAL with $\Delta t = 6.25 \times 10^{-5}$ s	6.63×10^{-5}	1.25×10^{-4}	101,933
SP with $\Delta t = 6.25 \times 10^{-5}$ s	NC	NC	-
SP with $\Delta t = 6.25 \times 10^{-6}$ s	3.36×10^{-5}	7.05×10^{-5}	486,941

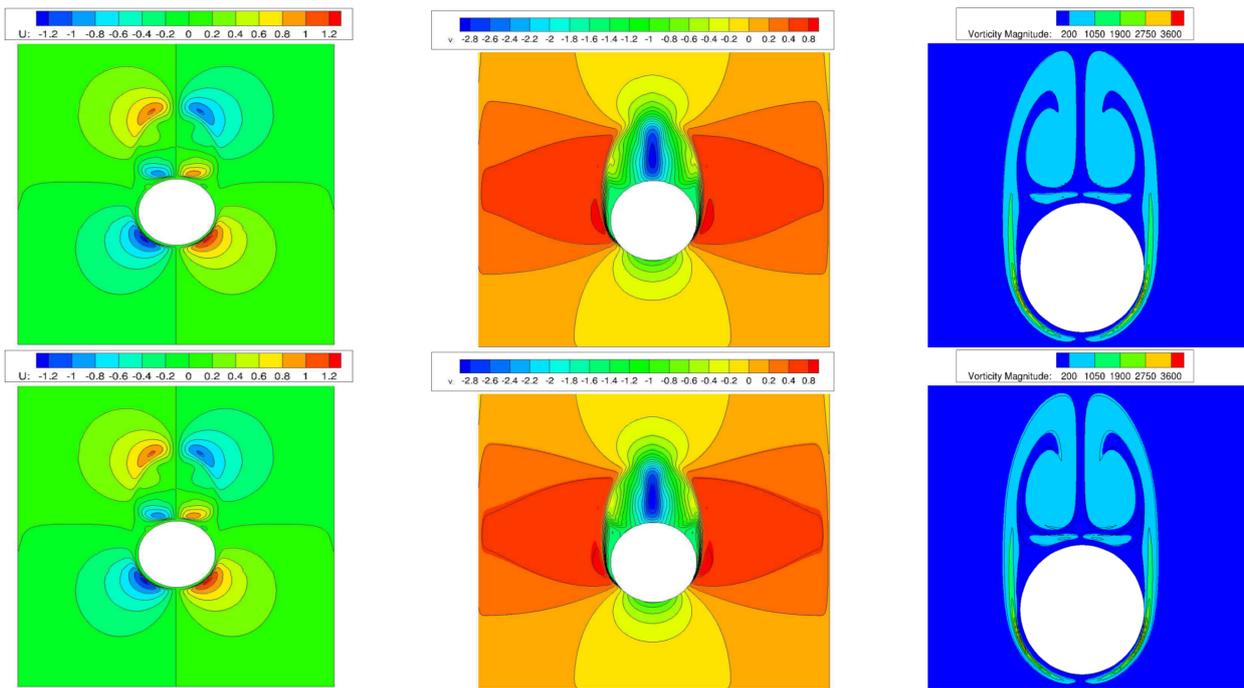


Figure 2. Free fall of dense cylinder at $t = 0.144$ s, obtained on a fine mesh 400×800 by the three methods. In the first row, FC (shown by fields) and AAL (by isolines). In the second row, FC (shown by fields) and SP (by isolines with $\Delta t = 6.25 \times 10^{-6}$ s). The different fields presented in each row are: left horizontal velocity $[0 \text{ m}, 0.1 \text{ m}] \times [0 \text{ m}, 0.11 \text{ m}]$, middle: vertical velocity $[0 \text{ m}, 0.1 \text{ m}] \times [0 \text{ m}, 0.10 \text{ m}]$, and right: vorticity magnitude $[0.02 \text{ m}, 0.08 \text{ m}] \times [0.034 \text{ m}, 0.089 \text{ m}]$.

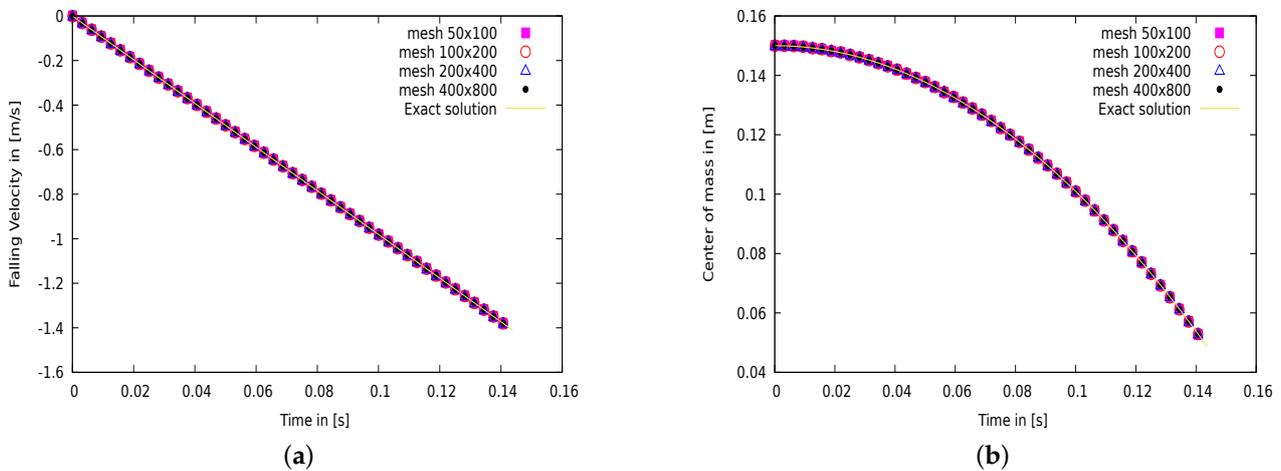


Figure 3. Vertical velocity and center of mass for test case 1 ($\rho_1/\rho_2 = 10^5, \mu_1/\mu_2 = 10^{10}$) obtained by the fully coupled method with several meshes. (a) Vertical velocity and (b) center of mass.

3.1.2. Comments on the Enstrophy

The enstrophy is a crucial integral quantity that is associated to dissipation and the effects of small-scale turbulence. It has to be noted that a two-dimensional work is performed here and that in two-dimensions, the energy cascade that could be related to turbulence and enstrophy is different from what occurs in three dimensions. Mathematically, the enstrophy is defined as the integral of vorticity square over the domain, and it will be written as Ens_1 for fluid 1 :

$$Ens_1 = \int_V \frac{\omega^2}{2} dV \tag{15}$$

where $\omega = \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right)$ is the vorticity in two dimensions. The following discrete analog of the enstrophy Ens_1 is used, in which $\tilde{C}_{i,j}$ is the color function located at the viscosities nodes ($\tilde{C}_{i,j} = 0$ in fluid 1 and $\tilde{C}_{i,j} = 1$ in fluid 2).

$$Ens_1 \simeq \sum_{i,j} (1 - \tilde{C}_{i,j}) \frac{\omega_{i,j}^2}{2} \Delta V \tag{16}$$

In this part, the evolution of the enstrophy in fluid 1 is investigated, for different grid sizes and test cases as in the previous study. The results are illustrated in Figure 4 on a logarithmic scale, for different grid sizes. As can be seen in this figure, the enstrophy increases over time in a monotonous way. As far as the convergence of the enstrophy, the 200×400 grid simulation is enough to converge the enstrophy for the fully coupled and the adaptative augmented Lagrangian solvers. In contrast, the enstrophy diverges on the finer grid for the standard projection, for the same reasons as the one mentioned before. And a time step ten times smaller is then needed to obtain the same solutions as the exact solvers (fully coupled and adaptative augmented Lagrangian solvers).

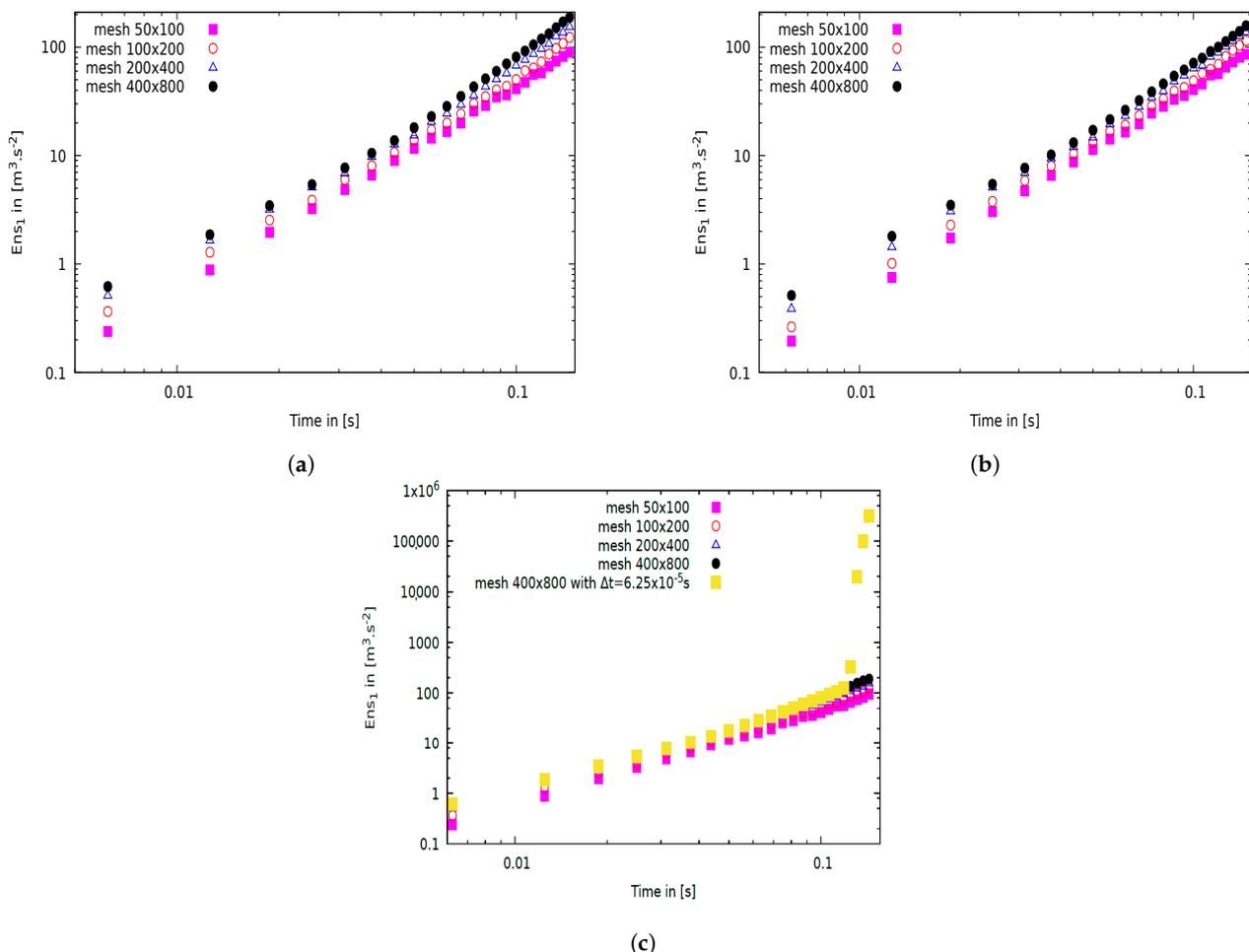


Figure 4. Time evolution of enstrophy in fluid 1 for the free fall of dense cylinder case. Obtained by several numerical methods with several Cartesian mesh. (a) FC method, (b) AAL method and (c) SP method.

3.2. Liquid Sheet Atomisation

To demonstrate the efficiency and robustness of the monolithic solvers for preserving kinetic energy, a flow interacting with a liquid sheet of thickness $\delta = 3 \times 10^{-4}$ m is

investigated [45]. This problem is defined in two dimensions on a square domain of side $L = 3 \times 10^{-3}$ m with periodic and symmetry boundary conditions for the horizontal and vertical sides of the domain, respectively. Simulations are carried out on a 256×256 Cartesian grids with a density ratio of 100 and 1000, while viscosity and capillary forces are set to zero, in order to preserve the initial kinetic energy during the simulation. The initial velocity field is defined as:

$$U = \begin{cases} u = A - 0.04 \cos\left(\frac{2\pi x}{L}\right) \frac{L}{2\pi} \frac{-2}{\delta} e^{-\frac{2y}{\delta}} \\ v = 0.04 \sin\left(\frac{2\pi x}{L}\right) e^{-\frac{2y}{\delta}} \end{cases} \quad (17)$$

$$A = \begin{cases} 30 \text{ m} \cdot \text{s}^{-1} \text{ in the gas} \\ 2 \text{ m} \cdot \text{s}^{-1} \text{ in the liquid} \end{cases} \quad (18)$$

The results of the time evolutions of the kinetic energy are reported in Figure 5 for all density ratios. It can be observed that whatever the density ratio, the total kinetic energy obtained by the fully coupled solver is conserved over time to almost computer error, which is in good agreement with the theory. As regard to the adaptative augmented Lagrangian solver, a slight increase in the kinetic energy is noticed here but the results are encouraging. On the contrary, the standard projection crashes after a few time iterations even for very small time steps. Consequently, the kinetic energy diverges due to the numerical dissipation generated by the splitting of operators. Besides, it has been shown that the conserving momentum method improves the conservation of the kinetic energy until the interface begins to deform. After this instant, kinetic energy decreases with projection methods, as reported for example in the work of Mukundan et al. [45]. On the other hand, these interfacial instabilities appear more quickly for the classic WENO and conservative Weno schemes. In this work, a centered scheme with either Fully Coupled or Augmented Lagrangian techniques conserves the kinetic energy for longer times compared to existing methods in the literature without the appearance of interfacial instabilities, which is an interesting result allowing to encourage more researchers to work with the monolithic solvers.

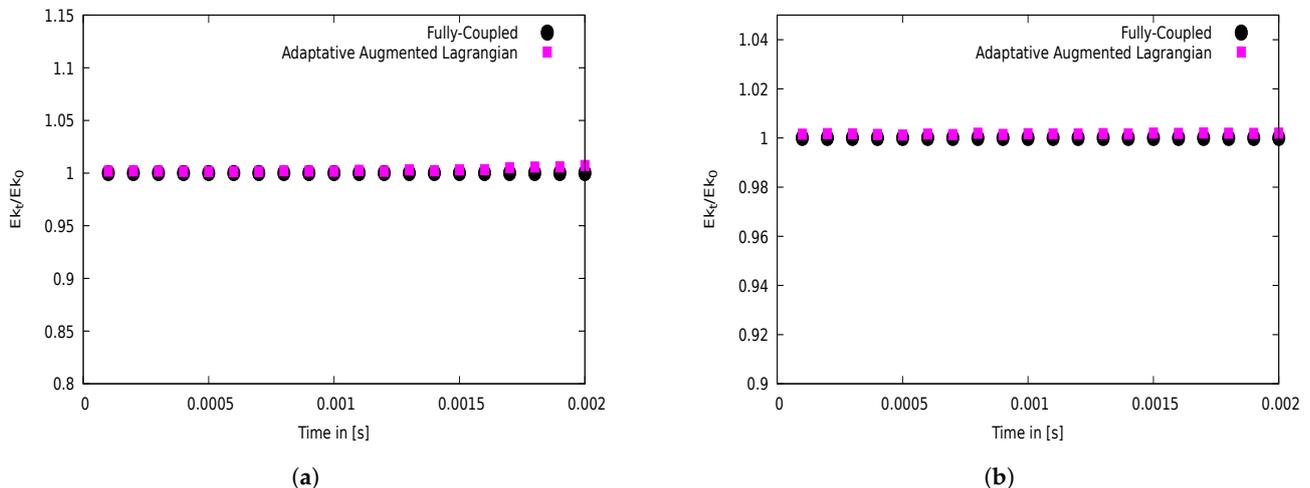


Figure 5. Time evolution of normalized kinetic energy for the high density ratio periodic Liquid sheet. Results obtained by different numerical methods, where (a) density ratio is 1000 and (b) density ratio is 100.

3.3. Simulation of 2D Viscous Jet Buckling

The jet buckling problem is an essential phenomenon since it is involved in many industrial applications such as the food industry or the aeronautics and aerospace applications with the filling of rocket boosters with propergol. The corresponding flow conditions are at low Reynolds number. This problem has been studied initially by Tomé et al. [46], who have provided a physical criterion which establishes the experiments with buckling jets. Then

many numerical investigations [47–49] have been conducted to simulate this problem in 2d and 3d configurations, for which the computational results are compared with experiments. The interest of this case lies in two reasons: on the one hand, the possibility of comparing the numerical solutions with experimental ones. On the other hand, the strong viscosity and density gradients near the interface yield ill-conditioned linear systems, which is a task challenge at the algebraic level. For numerical simulations, a square cavity of side $L = 1$ m full of gas is considered. In its upper part, a highly viscous fluid is injected at inlet velocity and inlet width equal to $-1 \text{ m} \cdot \text{s}^{-1}$ and $D = 0.08$ m, respectively, while the gravitational force acts with a constant intensity $g = 9.81 \text{ m} \cdot \text{s}^{-2}$ in the negative y-direction. For boundaries, a slip wall is employed on the solid walls and outflow boundary conditions at the outlet in the upper part of the cavity, outside the injector. The characteristics of the simulations are given in Table 4:

Table 4. Physical parameters describing the test cases. Here, ρ_2 and μ_2 refer to the propergol density and viscosity respectively, ρ_1 and μ_1 , refer to the density and viscosity of the gas respectively, $Re = \frac{\rho_2 D V_0}{\mu_2}$ is the Reynolds number associated to the propergol, $\frac{H}{D}$ is the aspect ratios of the jet and V_0 is the inlet velocity. In the present work, propergol (fluid 2) is assumed to behave like a Newtonian fluid, for the sake of simplicity.

Test Case	ρ_1 [$\text{kg} \cdot \text{m}^{-3}$]	ρ_2 [$\text{kg} \cdot \text{m}^{-3}$]	μ_1 [$\text{Pa} \cdot \text{s}$]	μ_2 [$\text{Pa} \cdot \text{s}$]	Re	$\frac{H}{D}$	V_0 [$\text{m} \cdot \text{s}^{-1}$]	ρ_2/ρ_1	μ_2/μ_1
1	1.1768	1800	1.85×10^{-3}	500	0.1728	12.5	0.6	1529	1.6×10^5
2	1.1768	1800	1.85×10^{-3}	300	0.6	10	1	1529	2.7×10^5
3	1.1768	1800	1.85×10^{-3}	300	3.6	10	6	1529	2.7×10^5

In their study, Tomé et al. give two important parameters to ensure the occurrence of buckling: the aspect ratios of the jet $H/D > 3\pi$ and $Re < 0.52$, where Re is the Reynolds number based on the inflow diameter D and the velocity inlet V_0 . Thus, the objective of this study is to compare the performance of three methods (the fully-coupled, adaptative augmented Lagrangian, and the standard projection) to deal with this kind of problem and to provide a good prediction of their associated physical behavior whereby the matrices resulting from discretization will be solved with the same solver, the BiCGSTAB(2) with a residual of $\epsilon = 10^{-6}$.

First the evolution of the liquid over time is considered for three Reynolds numbers $Re = 0.1728$, $Re = 0.6$ and $Re = 3.6$ (see Figure 6) using a 320×320 Cartesian grid and a time step $\Delta t = 5 \times 10^{-4}$ s. From these figures, It is noticeable that the numerical calculations with the fully-coupled method provide excellent results for the physics of the jet, which are in good agreement with the theoretical criterion given in [46]. Indeed, the first case $Re = 0.1728$ corresponds to a buckling jet, whereas second and third cases (also in first case for short times), $Re = 0.6$ and $Re = 3.6$ the jet oscillation does not appear. In the first case, it is observed that the diameter of the jet decreases in the middle part of the cavity, but the jet remains symmetric, this is because no buckling occurs. On the contrary, in the first case, as soon as it hits the lower wall of the cavity, the jet destabilizes, and helicoidal instability develops and traps air bubbles in the coiling motion of the viscous fluid. Finally, the streamlines are shown in Figure 7 for the first case, which gives the possibility to evaluate the interaction of both flows. From these figures, recirculations are observed, the size of which increases and then decreases alternately with the oscillations of the jet.

The results obtained with the adaptative augmented Lagrangian method are presented in Figure 8 for $Re = 0.1728$, using also a 320×320 Cartesian grid and a time step $\Delta t = 5 \times 10^{-4}$ s. The computation shows coherence between the fully coupled and the adaptative augmented Lagrangian methods at times $t = 1$ s and $t = 2$ s. Then, significant differences are noticed because the instability did not occur at the same time as with the fully coupled method, due to the nature of the augmented Lagrangian parameter, which behaves like a viscosity. These differences are also due to the unstable nature of the flow that amplifies numerical errors differently in accordance with the difference of the solvers

that are not sensitive similarly to truncation errors. The good point is that they trigger an instability for the same Re and in the relevant injection situation. However, it has been seen that the standard projection method for $Re = 0.1728$ on 320×320 mesh, fails to converge after 2800 iterations because the correction step was not able to reduce the divergence of the predicted velocity. The cases of $Re = 0.6$ and $Re = 3.6$ were simulated with the adaptive augmented Lagrangian and the standard projections methods. However, in these cases, all methods give the same results as those of the fully coupled method.

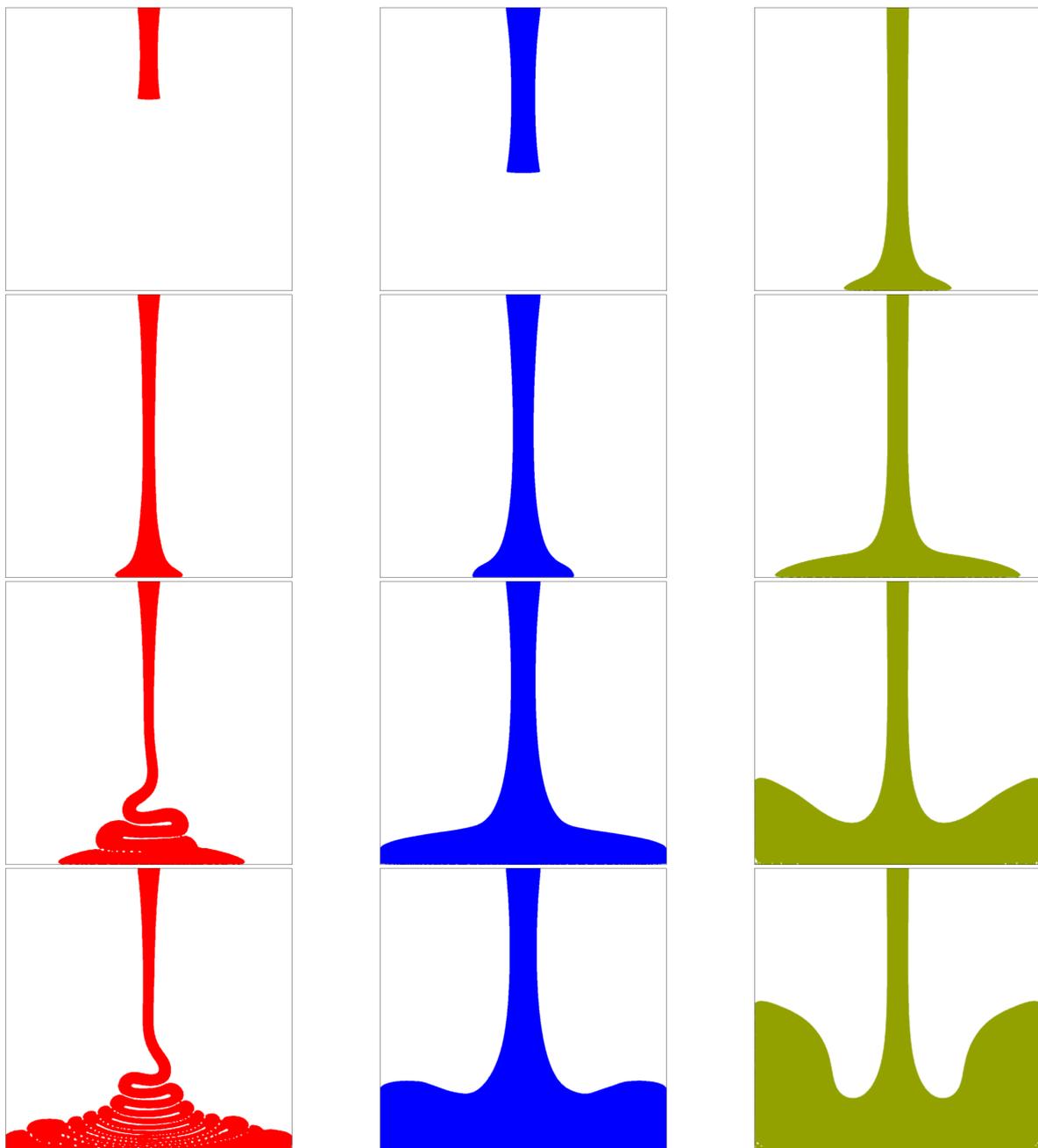


Figure 6. Numerical simulation of a viscous liquid (propergol) jet injected in a square cavity ($1 \text{ m} \times 1 \text{ m}$) on a 320×320 grid: the interface is plotted at time $t = 0.25 \text{ s}$, 0.25 s and 0.2 s (first row), $t = 0.75 \text{ s}$, 0.75 s and 0.3 s (second row), $t = 1.25 \text{ s}$, 1.25 s and 0.6 s (third row) and $t = 2 \text{ s}$, 2 s and 0.9 s (fourth row). First column $Re = 0.1728$, second column $Re = 0.6$ and third column $Re = 3.6$. Simulations are carried out with the FC method.

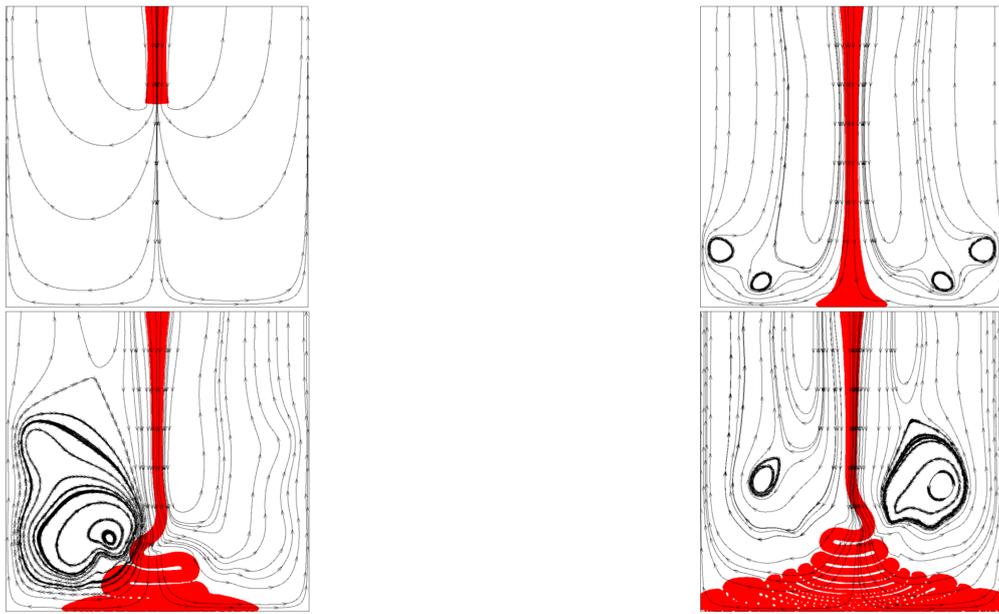


Figure 7. Numerical simulation of a viscous liquid (propergol) jet injected in a square cavity ($1\text{ m} \times 1\text{ m}$) on a 320×320 grid: the interface is plotted at time $t = 0.25\text{ s}$ and 0.75 s (first row), $t = 1.25\text{ s}$ and 2 s (second row). Test case 1 corresponds to $Re = 0.1728$.

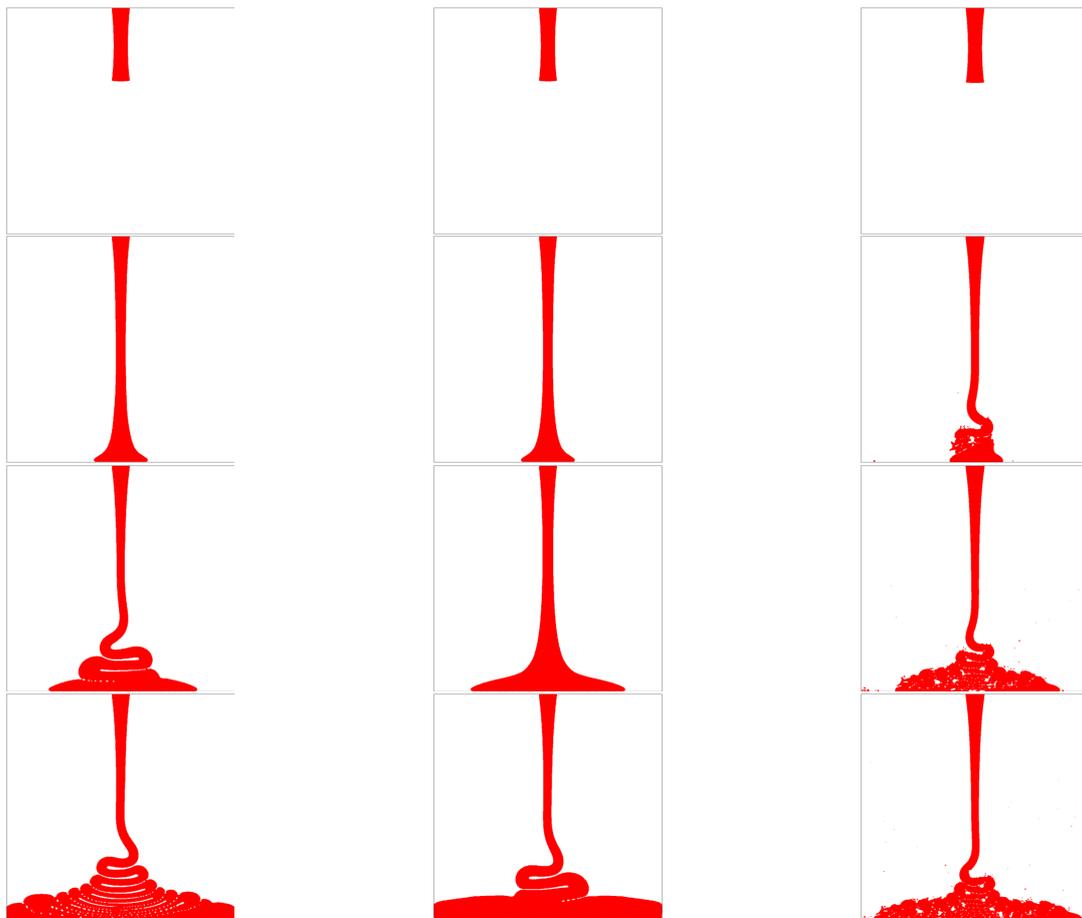


Figure 8. Numerical simulation of a viscous liquid (propergol) jet injected in a square cavity ($1\text{ m} \times 1\text{ m}$) on a 320×320 grid: the interface is plotted at time $t = 0.25\text{ s}$ (first row), $t = 0.75\text{ s}$ (second row), $t = 1.25\text{ s}$ (third row) and $t = 2\text{ s}$ (fourth row). First column (FC method), second column (AAL method) and third column (SP method).

4. Conclusions

A monolithic solver for simulating incompressible two-phase flows, at large density and viscosity ratios, has been developed and coupled with a VOF method such as PLIC technique. The coupled strategies belong to the two distinct classes of methods: the adaptive augmented Lagrangian method and the fully-coupled solver. These were chosen in order to overcome the difficulties associated with the segregated methods (such as projection methods). This allowed us to simulate complex cases characterized by high density and viscosity ratios, such as the free fall of dense cylinder and the viscous jet buckling, but also to potentially overcome the momentum conserving method in the case of Liquid sheet atomisation. We have also shown that this monolithic solvers allow to use larger time steps than segregated methods, which fails to converge in some cases. In future work, it will be intended to investigate fully coupled solvers on complex 3D and on a massively parallel computers.

Author Contributions: Conceptualization M.E.O. and S.V.; theoretical developments, analysis and implementation M.E.O.; simulations M.E.O. and S.V.; base CFD software S.V.; writing—original draft preparation, M.E.O.; writing—review and editing, M.E.O., S.V. and V.L.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors are grateful for access to the computational facilities of the French CINES (National computing center for higher education) and CCRT (National computing center of CEA) under project number A0032B06115. We are also grateful for I-SITE FUTURE, UPE and PIA under funding ANR-16-IDEX-0003.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Nangia, N.; Griffith, B.E.; Patankar, N.A.; Bhalla, A.P.S. A robust incompressible Navier-Stokes solver for high density ratio multiphase flows. *J. Comput. Phys.* **2019**, *390*, 548–594. [\[CrossRef\]](#)
2. Guermont, J.; Mineev, P.; Shen, J. An overview of projection methods for incompressible flows. *Comput. Methods Appl. Mech. Eng.* **2006**, *195*, 6011–6045. [\[CrossRef\]](#)
3. Mirjalili, S.; Jain, S.; Dodd, M. *Interface-Capturing Methods for Two-Phase Flows: An Overview and Recent Developments*; Annual Research Brief; Center for Turbulence Research: Stanford, CA, USA, 2017; pp. 117–135.
4. Tryggvason, G.; Bunner, B.; Esmaeeli, A.; Juric, D.; Al-Rawahi, N.; Tauber, W.; Han, J.; Nas, S.; Jan, Y.J. A Front-Tracking Method for the Computations of Multiphase Flow. *J. Comput. Phys.* **2001**, *169*, 708–759. [\[CrossRef\]](#)
5. Osher, S.; Sethian, J. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Comput. Phys.* **1988**, *79*, 12–49. [\[CrossRef\]](#)
6. Bonito, A.; Nochetto, R.H. (Eds.) Chapter 6—A review of level set methods to model interfaces moving under complex physics: Recent challenges and advances. In *Handbook of Numerical Analysis*; Elsevier: Amsterdam, The Netherlands, 2020; Volume 21, pp. 509–554.
7. Jiang, G.S.; Shu, C.W. Efficient Implementation of Weighted ENO Schemes. *J. Comput. Phys.* **1996**, *126*, 202–228. [\[CrossRef\]](#)
8. Olsson, E.; Kreiss, G. A conservative level set method for two phase flow. *J. Comput. Phys.* **2005**, *210*, 225–246. [\[CrossRef\]](#)
9. Olsson, E.; Kreiss, G.; Zahedi, S. A conservative level set method for two phase flow II. *J. Comput. Phys.* **2007**, *225*, 785–807. [\[CrossRef\]](#)
10. Hirt, C.; Nichols, B. Volume of fluid (VOF) method for the dynamics of free boundaries. *J. Comput. Phys.* **1981**, *39*, 201–225. [\[CrossRef\]](#)
11. Youngs, D. Time-Dependent Multi-material Flow with Large Fluid Distortion. *Numer. Methods Fluid Dyn.* **1982**, *24*, 273–285.
12. Sussman, M.; Puckett, E.G. A Coupled Level Set and Volume-of-Fluid Method for Computing 3D and Axisymmetric Incompressible Two-Phase Flows. *J. Comput. Phys.* **2000**, *162*, 301–337. [\[CrossRef\]](#)
13. Ahn, H.; Shashkov, M.; Christon, M. The Moment-of-Fluid Method in Action. *APS* **2007**, *60*, KB-003. [\[CrossRef\]](#)
14. Ding, H.; Speltz, P.; Shu, C. Diffuse interface model for incompressible two-phase flows with large density ratios. *J. Comput. Phys.* **2007**, *226*, 2078–2095. [\[CrossRef\]](#)
15. van der Waals, J.D. Thermodynamische Theorie der Kapillarität unter Voraussetzung stetiger Dichteänderung. *Z. Phys. Chem.* **1894**, *13U*, 657–725.
16. Jamet, D.; Lebaigue, O.; Coutris, N.; Delhay, J. The Second Gradient Method for the Direct Numerical Simulation of Liquid—Vapor Flows with Phase Change. *J. Comput. Phys.* **2001**, *169*, 624–651. [\[CrossRef\]](#)

17. Mirjalili, S.; Ivey, C.B.; Mani, A. Comparison between the diffuse interface and volume of fluid methods for simulating two-phase flows. *Int. J. Multiph. Flow* **2019**, *116*, 221–238. [[CrossRef](#)]
18. Sato, Y.; Ničeno, B. A conservative local interface sharpening scheme for the constrained interpolation profile method. *Int. J. Numer. Methods Fluids* **2012**, *70*, 441–467. [[CrossRef](#)]
19. Pianet, G.; Vincent, S.; Leboi, J.; Caltagirone, J.P.; Anderhuber, M. Simulating compressible gas bubbles with a smooth volume tracking 1-Fluid method. *Int. J. Multiph. Flow* **2010**, *36*, 273–283. [[CrossRef](#)]
20. Guillaument, R.; Vincent, S.; Caltagirone, J.P. An original algorithm for VOF based method to handle wetting effect in multiphase flow simulation. *Mech. Res. Commun.* **2015**, *63*, 26–32. [[CrossRef](#)]
21. Brackbill, J.; Kothe, D.; Zemach, C. A continuum method for modeling surface tension. *J. Comput. Phys.* **1992**, *100*, 335–354. [[CrossRef](#)]
22. Chorin, A.J. Numerical Solution of the Navier-Stokes Equations. *Math. Comput.* **1968**, *22*, 745–762. [[CrossRef](#)]
23. Témam, R. Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (I). *Arch. Ration. Mech. Anal.* **1969**, *32*, 135–153. [[CrossRef](#)]
24. Guermond, J.L. Remarques sur les méthodes de projection pour l'approximation des équations de Navier-Stokes. *Numer. Math.* **1994**, *67*, 465–473. [[CrossRef](#)]
25. Fortin, M.; Glowinski, R. Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems. *ZAMM J. Appl. Math. Mech. Z. Angew. Math. Mech.* **1985**, *65*, 622.
26. Vincent, S.; Caltagirone, J.P.; Lubin, P.; Randrianarivelo, T.N. An adaptative augmented Lagrangian method for three-dimensional multi-material flows. *Comput. Fluids* **2004**, *33*, 1273–1289. [[CrossRef](#)]
27. Vincent, S.; Sarthou, A.; Caltagirone, J.; Sonilhac, F.; Février, P.; Mignot, C.; Pianet, G. Augmented Lagrangian and penalty methods for the simulation of two-phase flows interacting with moving solids. Application to hydroplaning flows interacting with real tire tread patterns. *J. Comput. Phys.* **2011**, *230*, 956–983. [[CrossRef](#)]
28. Benzi, M.; Golub, G.H.; Liesen, J. Numerical solution of saddle point problems. *Acta Numer.* **2005**, *14*, 1–137. [[CrossRef](#)]
29. Elman, H.C.; Silvester, D.J.; Wathen, A.J. *Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics*; Oxford University Press: Oxford, UK, 2005.
30. Bootland, N.; Bentley, A.; Kees, C.; Wathen, A. Preconditioners for Two-Phase Incompressible Navier–Stokes Flow. *SIAM J. Sci. Comput.* **2019**, *41*, B843–B869. [[CrossRef](#)]
31. Cai, M.; Nonaka, A.J.; Bell, J.B.; Griffith, B.E.; Donev, A. Efficient Variable-Coefficient Finite-Volume Stokes Solvers. *Commun. Comput. Phys.* **2014**, *16*, 1263–1297. [[CrossRef](#)]
32. Bussmann, M.; Kothe, D.B.; Sicilian, J.M. Modeling High Density Ratio Incompressible Interfacial Flows; Volume 1: Fora, Parts A and B; 2002; pp. 707–713. Available online: http://xxx.lanl.gov/abs/https://asmedigitalcollection.asme.org/FEDSM/proceedings-pdf/FEDSM2002/36150/707/4546310/707_1.pdf (accessed on 27 December 2020)
33. Raessi, M.; Pitsch, H. Consistent mass and momentum transport for simulating incompressible interfacial flows with large density ratios using the level set method. *Comput. Fluids* **2012**, *63*, 70–81. [[CrossRef](#)]
34. Le Chenadec, V.; Pitsch, H. A monotonicity preserving conservative sharp interface flow solver for high density ratio two-phase flows. *J. Comput. Phys.* **2013**, *249*, 185–203. [[CrossRef](#)]
35. Desjardins, O.; Moureau, V.; Pitsch, H. An accurate conservative level set/ghost fluid method for simulating turbulent atomization. *J. Comput. Phys.* **2008**, *227*, 8395–8416. [[CrossRef](#)]
36. Asuri Mukundan, A.; Ménard, T.; Herrmann, M.; Motta, J.; Berlemont, A. Analysis of dynamics of liquid jet injected into gaseous crossflow using proper orthogonal decomposition. In Proceedings of the AIAA SciTech 2020 Forum, Orlando, FL, USA, 6–10 January 2020; doi:10.2514/6.2020-0357. [[CrossRef](#)]
37. Kataoka, I. Local instant formulation of two-phase flow. *Int. J. Multiph. Flow* **1986**, *12*, 745–758. [[CrossRef](#)]
38. Angot, P.; Bruneau, C.H.; Fabrie, P. A penalization method to take into account obstacles in viscous flows. *Numer. Math.* **1999**, *81*, 497–520. [[CrossRef](#)]
39. Caltagirone, J.P.; Vincent, V. Sur une méthode de pénalisation tensorielle pour la résolution des équations de Navier-Stokes. *Comptes Rendus De L'Académie Des Sci. Ser. IIB Mech.* **2001**, *329*, 607–613. [[CrossRef](#)]
40. Benkenida, A.; Magnaudet, J. Une méthode de simulation d'écoulements diphasiques sans reconstruction d'interfaces. *Comptes Rendus De L'Académie Des Sci. Ser. IIB Mech. Phys.* **2000**, *328*, 25–32. [[CrossRef](#)]
41. Vincent, S.; de Motta, J.C.B.; Sarthou, A.; Estivalezes, J.L.; Simonin, O.; Climent, E. A Lagrangian VOF tensorial penalty method for the DNS of resolved particle-laden flows. *J. Comput. Phys.* **2014**, *256*, 582–614. [[CrossRef](#)]
42. Harlow, F.H.; Welch, J.E. Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface. *Phys. Fluids* **1965**, *8*, 2182–2189. [[CrossRef](#)]
43. Dongarra, J.J.; Duff, I.S.; Sorensen, D.C.; Van der Vorst, H.A. *Numerical Linear Algebra for High Performance Computers*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 1998.
44. Elouafa, M.; Vincent, S.; Le Chenadec, V. Navier-Stokes Solvers for Incompressible Single-and Two-Phase Flows. Working Paper Accepted in CiCP Journal. Available online: <https://hal-upec-upem.archives-ouvertes.fr/hal-02892344/document> (accessed on 27 December 2020)
45. Mukundan, A.A.; Ménard, T.; de Motta, J.C.B.; Berlemont, A. A 3D Moment of Fluid method for simulating complex turbulent multiphase flows. *Comput. Fluids* **2020**, *198*, 104364. [[CrossRef](#)]

46. Tomé, M.F.; Duffy, B.; McKee, S. A numerical technique for solving unsteady non-Newtonian free surface flows. *J. Non-Newton. Fluid Mech.* **1996**, *62*, 9–34.
47. Vincent, S.; Caltagirone, J.P. Efficient solving method for unsteady incompressible interfacial flow problems. *Int. J. Numer. Methods Fluids* **1999**, *30*, 795–811. [[CrossRef](#)]
48. Xu, X.; Ouyang, J.; Jiang, T.; Li, Q. Numerical simulation of 3D-unsteady viscoelastic free surface flows by improved smoothed particle hydrodynamics method. *J. Non-Newton. Fluid Mech.* **2012**, *177–178*, 109–120. [[CrossRef](#)]
49. Oishi, C.; Martins, F.; Tomé, M.; Alves, M. Numerical simulation of drop impact and jet buckling problems using the eXtended Pom—Pom model. *J. Non-Newton. Fluid Mech.* **2012**, *169–170*, 91–103. [[CrossRef](#)]

Annexe H

**Communication issue du Congrès
Français de Mécanique (CFM)**

Développement d'un solveur tout-couplé parallèle 3D pour la simulation des écoulements diphasiques incompressibles

M. El Ouafa, S. Vincent, V. Le Chenadec, B. Trouette

Univ Gustave Eiffel, Univ Paris Est Creteil, CNRS, UMR 8208, MSME, F-77454 Marne-la-Vallée,
France

email : mohamed.elouafa@univ-eiffel.fr

Résumé

Les équations de Navier-Stokes diphasiques jouent un rôle crucial dans plusieurs applications industrielles et environnementales. Cependant, leur résolution pose de nombreux problèmes et devient un enjeu important pour les mathématiciens ainsi que les ingénieurs. Dans le contexte des écoulements diphasiques incompressibles, les difficultés sont multiples : d'une part, le couplage vitesse-pression induit par la contrainte d'incompressibilité, donne lieu à un problème de point selle pour lequel des solveurs dédiés doivent être développés. Et, d'autre part, les forts rapports de densité et de viscosité en présence de grandes déformations de l'interface conduisent à des systèmes linéaires très mal conditionnés. Nous décrivons ici un solveur tout-couplé parallèle 3D, dédié à la simulation des écoulements diphasiques incompressibles. Les équations linéarisées de quantité de mouvement et de continuité découlant de la solution implicite des vitesses et de la pression dans les fluides sont résolues simultanément. Nous avons simulé un cas de l'instabilité d'un jet visqueux pour vérifier les modèles, notamment en présence de forts rapports de densité et de viscosité. Nous avons également réalisé un cas d'inversion de phase 3D, pour démontrer la capacité de notre nouveau solveur tout-couplé à résoudre des problèmes diphasiques sur un milliards de cellules.

Abstract :

The Navier-Stokes equations play a crucial role in several industrial and environmental applications. However, their resolution causes many problems and remains a challenging task for mathematicians as well as engineers. In the context of incompressible two-phase flows, the difficulties are multiple : on the one hand, the velocity-pressure coupling induced by the incompressibility constraint, gives rise to a saddle point problem for which suitable solvers must be developed. And, on the other hand, the high density and viscosity ratios in the presence of large interface deformations lead to very ill-conditioned linear systems. We describe here a fully-coupled 3D parallel solver for the simulation of two-phase incompressible flows. The linearized momentum and continuity equations arising from the implicit solving of the fluid velocities and pressure are solved simultaneously. The instability of a 3D viscous jet is simulated, in order to validate the models, especially in the presence of strong density and viscosity ratios. We also performed a 3D phase inversion case, in order to exhibit the capacity of our new fully-coupled solver to deal with two-phase problems on a billion cells.

Mots clefs : Solveur tout-couplé, point-selle, 3D parallèle, écoulements diphasiques, fort ratios

1 Introduction

Les écoulements diphasiques incompressibles mettant en œuvre un liquide et un gaz sont omniprésents dans de nombreuses applications industrielles et académiques et, par conséquent, ils font l'objet de nombreuses études numériques. Les écoulements d'impact de gouttes sur des surfaces, la fabrication de dépôts par jet de plasma thermique, le déferlement d'une vague sur une plage, l'atomisation des jets ou les sprays rencontrés lors du refroidissement de parois de bâtiments ou dans les moteurs sont, par exemple, des problèmes diphasiques et turbulents.

La modélisation et la simulation numérique de ce type d'écoulement reste un grand défi en raison, entre autres des aspects multi-échelles (allant de la phase séparée à l'échelle des petites interfaces) et multi-physiques (phases séparées ou dispersées, frottements, effets capillaires, changement de phase, mouillabilité...). En effet, les forts rapports de densité et de viscosité dans les écoulements diphasiques engendrent souvent un cisaillement important et des discontinuités à l'interface fluide-fluide. Ces discontinuités compliquent particulièrement la discrétisation des équations de conservation, et conduisent à des instabilités numériques. En outre, le traitement du couplage vitesse-pression dans les écoulements monophasiques incompressibles a fait l'objet de nombreux travaux et est relativement bien maîtrisé. Au contraire, dans le cadre d'écoulements diphasiques avec de larges rapports de densité et de viscosité, des difficultés numériques apparaissent, liées à la présence de fortes déformations de l'interface et à des sauts de propriétés des fluides matériels qui y sont associés. Elles conduisent à des systèmes linéaires mal conditionnés, ce qui rend le problème beaucoup plus difficile à résoudre. Au niveau discret, le problème persiste sous la forme de recherche d'un point selle dont la résolution est coûteuse. Pour faire face à ce problème, de gros progrès ont cependant été réalisés au cours des dernières décennies, et de nombreuses méthodes ont été proposées, elles peuvent être classés ainsi en deux catégories. D'une part, les méthodes approchées, introduites par Chorin et Temam [6, 21] et leurs variantes [10, 4, 3, 11], approximent le système original en utilisant le splitting en temps, ce qui aboutit à deux équations découplées : l'une pour mettre à jour le champ de vitesse et l'autre le champ de pression. Cela soulève un certain nombre de problèmes, tels que l'apparition de couches numériques en raison de splitting en temps, ou la nécessité de conditions limites de pression qui n'existent pas dans le problème original. D'autre part, les méthodes exactes résolvent les deux champs (vitesse et pression) simultanément, préservant ainsi la consistance du système discrétisé avec les équations continues. Dans ces méthodes exactes, le système à point selle peut être résolu soit directement par un solveur tout-couplé [16, 1, 18, 17], soit par une méthode de Lagrangian augmenté (AL) [9, 23]. L'intérêt de ces méthodes réside dans le fait qu'elles permettent une résolution très précise et robuste, sans erreur de splitting (sauf celle liée à la linéarisation du terme d'inertie) des écoulements diphasiques incompressibles.

Nous développons ici une méthode exacte 3D parallèle, tout-couplée, pour simuler des écoulements supposés incompressibles et diphasiques. Nous décrivons d'abord le modèle, les schémas de discrétisations et les solveurs seront présentés dans un second temps. Nous validons ensuite notre approche numérique de VOF lorsqu'elle est appliquée à la résolution de l'équation d'advection de la fraction volumique. Ensuite, nous considérerons un cas 3D d'inversion de phase, pour montrer la capacité de la méthode à traiter des cas diphasiques sur plus d'un milliard de cellules. Enfin, un cas 3D d'un jet visqueux sera présenté à fin de mettre en valeur l'intérêt du solveur tout-couplé à simuler des problèmes raides à fort rapport de masse volumique et de viscosité.

2 Méthodes numériques

2.1 Approche diphasique de la résolution des équation de Navier-Stokes

Dans ce travail, l'écoulement diphasique est décrit par le modèle à un fluide de Katoaka [14], basé sur une approche eulérienne où les phases sont vues via des fractions de volume. Dans chacune des deux phases fluides, les équations à résoudre sont : l'équation de conservation de la masse et l'équation de conservation de la quantité de mouvement, avec un terme source additionnel dans l'équation de la quantité de mouvement (la force capillaire) et une équation d'advection supplémentaire pour la fonction de phase C . La fonction C , qui peut prendre des valeurs comprises entre 0 et 1 représente le taux de présence d'un des deux fluides dans un volume de contrôle. Les équations de modèle à un fluide dans le cas diphasique s'écrivent ainsi :

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \overline{\overline{\mathbf{B}}}_{penu} (f(\mathbf{u}) - \mathbf{u}_\infty) = -\nabla p + \nabla \cdot \overline{\overline{\mathbf{T}}} + \rho \mathbf{g} + \mathbf{F}_s, \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (2)$$

$$\frac{\partial C}{\partial t} + \nabla \cdot (\mathbf{u}C) = C \nabla \cdot \mathbf{u}, \quad (3)$$

$$\rho(C) = \rho_1 C + (1 - C)\rho_2, \quad (4)$$

$$\mu(C) = \mu_1 C + (1 - C)\mu_2 \quad (5)$$

Avec \mathbf{u} le champ de vitesse, p le champ de pression, t le temps, ρ et μ la masse volumique et la viscosité du fluide de mélange, données par une moyenne arithmétique, avec ρ_1 , ρ_2 , μ_1 et μ_2 les masses volumiques et les viscosités des fluides 1 et 2 respectivement. En outre, $\overline{\overline{\mathbf{T}}}$ représente le tenseur des taux de déformations pour un fluide Newtonien [5], tandis que $\overline{\overline{\mathbf{B}}}_{penu}$ est un tenseur d'ordre 2 pour spécifier les conditions aux limites [19], dont les composantes diagonales tendent vers l'infini le long de la frontière et sont identiquement nulles à l'intérieur du domaine de fluide. En outre, \mathbf{F}_s est le terme capillaire agissant sur l'interface, modélisé dans cette étude par le modèle CSF de Brackbill [2].

2.2 Schémas de discrétisation et solveurs

Dans le présent article, les équations du modèle diphasique incompressible et instationnaire sont approximées via des volumes finis pénalisés (pour traiter différents types de conditions aux limites), sur un maillage décalé. Les variables discrètes de vitesse et de pression sont disposées de manière décalée, chacune ayant son propre volume de contrôle, comme le montre la Figure 1. Ce choix garantit la concordance des opérateurs différentiels tels que la divergence et le gradient, et il évite également les oscillations sur le champ de pression. Sachant que la solution de l'équation d'advection (3) est effectuée avec la méthode VOF conservative de Weymouth [12], qui conserve le volume à la précision machine. Ainsi, Le système semi-discret résultant de la discretisations des équations de conservation de la masse (Eq. 2) et de la quantité de mouvement est avancé dans le temps, en utilisant un schéma implicite De Gear du second ordre. En revanche, un schéma centré pour le gradient de pression, la divergence de vitesse, les termes convectifs et visqueux est utilisé. Pour éviter de mettre en œuvre la linéarisation de Newton et d'inverser le Jacobien complet, qui s'avère plus efficace mais plus coûteux pour les écoulements instationnaires, les termes convectifs sont linéarisés en utilisant l'extrapolation d'Adams-Bashforth comme

suit :

$$\mathbf{u}^{n+1} \otimes \mathbf{u}^{n+1} \approx (2\mathbf{u}^n - \mathbf{u}^{n-1}) \otimes \mathbf{u}^{n+1}. \quad (6)$$

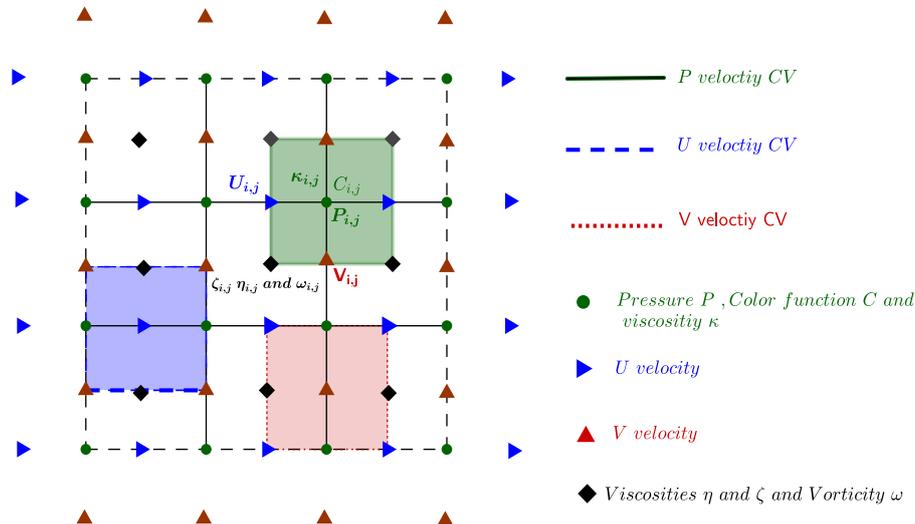


FIGURE 1 – Grille décalée de vitesse et pression : représentation des nœuds supplémentaires de vitesse en dehors de la frontière physique.

Ici, n désigne un indice de temps associé au temps physique $n\Delta t$ et Δt est supposé être le pas de temps constant à chaque itération résolue. Ainsi, après une linéarisation et une discrétisation appropriées, et, comme une résolution implicite tout-couplé (Fully Coupled) est visée, le système générique est un système linéaire non symétrique de type point selle. Les deux inconnues du problème, *i.e.* la vitesse \mathbf{u} et la pression p , sont couplées par la contrainte d'incompressibilité $\nabla \cdot \mathbf{u} = 0$. Ce système de points selle peut être reformulé comme suit :

$$\begin{pmatrix} \frac{1}{\Delta t} M_u^{(\rho)} + N_u^{(\rho)} + L_u^{(\mu)} & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ 0 \end{pmatrix} \quad (7)$$

Où la matrice B_u est la partie discrète de la divergence et B_p^T représente l'opérateur discret du gradient de pression. En outre, $M_u^{(\rho)}$ est la matrice de masse de la vitesse, $L_u^{(\mu)}$ est le Laplacien discret de la vitesse, $N_u^{(\rho)}$ désigne la matrice associée au terme convectif et \mathbf{f} représente le terme source. Le système réduit peut être réécrit sous la forme $A\mathbf{x} = \mathbf{b}$ avec A la matrice associée aux coefficients de discrétisation spatio-temporelle et \mathbf{b} le second membre associé à chaque composante du vecteur solution. La solution du problème diphasique est alors :

$$\mathbf{x} = A^{-1} \times \mathbf{b} \quad (8)$$

Notre stratégie pour résoudre ce système linéaire creux est de faire appel au solveur BiCGStab(2) [13] sur la totalité du système, en choisissant la structure CSR (compressed storage raw) afin de ne stocker que les coefficients non nuls de la matrice, et ensuite de construire un préconditionneur efficace pour accélérer la convergence du solveur itératif. Cette stratégie peut être mise en place en utilisant le préconditionneur pression-convection-diffusion (PCD) [1] pour le bloc pression, et le préconditionneur triangulaire par

bloc pour le bloc de vitesse (nous invitons le lecteur à consulter [17, 18] pour plus de détails).

3 Résultats

3.1 Cas test pour l'équation d'advection

Afin de montrer les capacités de notre méthode VOF à décrire correctement l'interface, même lorsque celle-ci est fortement étirée et déformée, un cas test standard de transport d'interface est étudié. Ce cas test a été proposé par Enright et al. [7] à partir des travaux de Leveque [15]. Initialement, une goutte sphérique de rayon $r = 0.15$ est placée aux coordonnées $(0.35, 0.35, 0.35)$ dans une boîte périodique de côté $L = 1$. Dans cette configuration, l'interface est transportée avec le champ de vitesse suivant :

$$\begin{aligned} u(x, y, z) &= 2 \sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) \\ v(x, y, z) &= -\sin(2\pi x) \sin^2(\pi y) \sin(2\pi z) \\ w(x, y, z) &= -\sin(2\pi x) \sin(2\pi y) \sin^2(\pi z) \end{aligned} \quad (9)$$

Ce champ de vitesse est modulé par une fonction de temps $f(t)$ définie par $f(t) = \cos(\pi t/T)$, avec $T = 3s$ la période. Ainsi, le champ de vitesse est réversible, en d'autres termes, la goutte revient à sa position initiale en $t = T$.

La figure 2 présente les résultats que nous avons obtenus pour une interface sphérique avec un maillage de 512^3 . Nous pouvons voir que nous arrivons à des interfaces qui sont similaires à celles proposées dans les travaux de Douglas Enright. Nous avons également réalisé une étude de convergence en calculant l'ordre de précision spatial. Pour ce faire, nous avons choisi deux types d'erreur : d'une part, l'erreur de conservation de masse intégrée sur le domaine de calcul. Elle est définie comme suit,

$$\epsilon_{mass} = \frac{\int_V c(\mathbf{x}, t) dV - \int_V c(\mathbf{x}, 0) dV}{\int_V c(\mathbf{x}, 0) dV} \quad (10)$$

Et d'autre part, l'erreur géométrique qui mesure la différence entre l'interface calculée et l'interface exacte :

$$\epsilon_{geo} = \int_V [c(\mathbf{x}, T) - c_{exacte}(\mathbf{x}, T)] dV \quad (11)$$

Nous dressons les résultats de l'étude de convergence dans le tableau 1. Nous constatons une conservation de la masse à la précision machine quelque soit le maillage. En outre, un ordre 2 est obtenu sur l'erreur géométrique ce qui valide notre méthode de transport.

Maillage	ϵ_{mass}	ordre	ϵ_{geo}	ordre
32^3	2.05×10^{-12}	-	7.82×10^{-3}	-
64^3	4.69×10^{-13}	-	2.72×10^{-3}	1.52
128^3	5.72×10^{-14}	-	6.50×10^{-4}	2.07
256^3	2.18×10^{-14}	-	1.51×10^{-4}	2.11

TABLE 1 – Etude de convergence schéma VOF de Weymouth : Goutte sphérique dans un champs de vitesse réversible.

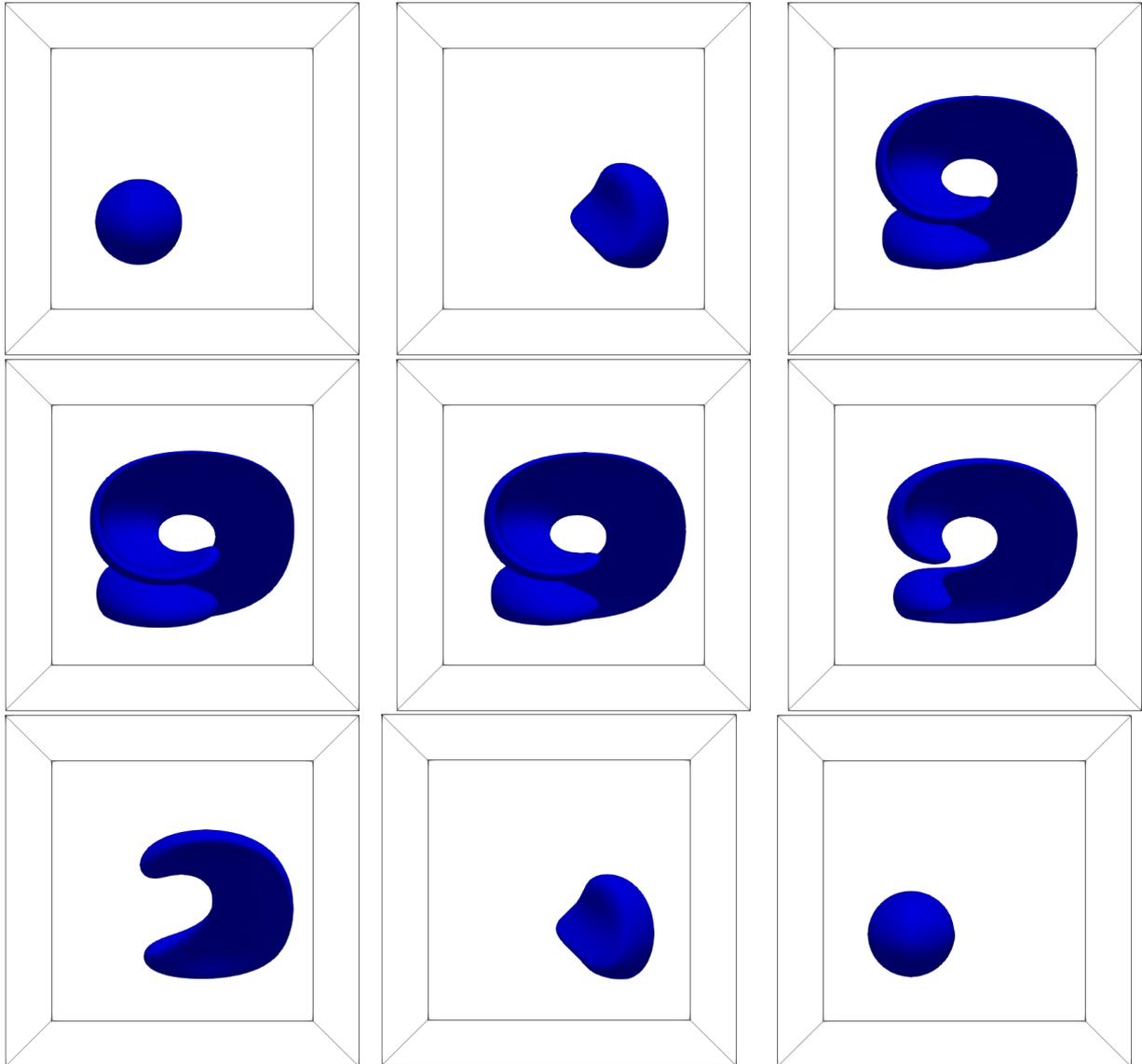


FIGURE 2 – Maillage 512^3 : Évolution de l'interface sphérique dans un champ de vitesse réversible à divergence nulle

3.2 Cas d'application : L'inversion de phase eau-huile

Un écoulement diphasique, eau-huile par exemple, dans lequel une des phases est dispersée dans une autre, peut correspondre à une inversion de phase, c'est-à-dire, la phase dispersée devenant la phase continue. Ce phénomène instationnaire est ensuite accompagné de plusieurs processus interfaciaux à petite et grande échelle, tels que la déformation, la formation de ligaments, la rupture de l'interface et la coalescence (voir figure 3b).

Ici, nous menons une simulation numérique directe sur un cas d'inversion de phase étudié récemment par Saeedipour et al [20], afin de montrer la capacité de notre solveur tout-couplé à faire des simulations massivement parallèle. Les quantités d'intérêt seront donc les grandeurs macroscopiques, telles que l'énergie potentielle, l'énergie cinétique, l'ensrophie, la dissipation visqueuse, l'aire interfaciale et le volume du fluide léger dans la partie supérieure du domaine. Dans ce qui suit, nous présentons la géométrie et les paramètres physiques du problème. Ensuite, nous discutons les résultats des simulations

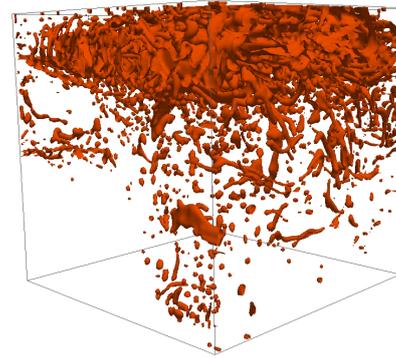
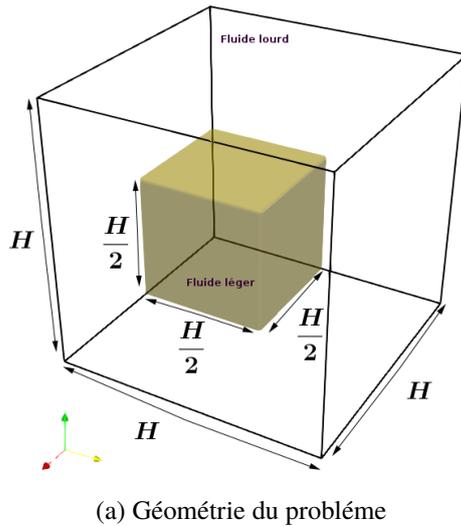


FIGURE 3 – Problème de l'inversion de phase liquide/liquide dans une boîte fermée [8]

numériques ainsi que leurs comparaisons avec les références.

3.2.1 Géométrie et paramètres physiques

Comme décrit sur la figure 3a, un volume cubique d'un fluide léger (identifié comme le fluide 2) est placé dans la partie inférieure d'une boîte remplie d'un fluide lourd (le fluide 1). Nous pourrions assimiler les fluides 1 et 2 comme étant respectivement de l'eau et de l'huile. Nous imposerons une viscosité égale pour les 2 fluides ($\mu_1 = \mu_2 = \mu$). La boîte est de dimension (H, H, H) et la goutte de fluide 1 $(H/2, H/2, H/2)$. Nous considérerons un glissement libre et un mouillage partiel sur les murs extérieurs. Nous considérerons également que les angles de contact de 90° sont équivalents à la symétrie d'un miroir. L'accélération gravitationnelle est choisie comme étant $g = (0, 0, -9.81)$. Les différents paramètres du cas d'étude sont donnés dans le tableau 2. Les nombres adimensionnels qui gouvernent la dynamique de ce problème sont donc le nombre de Reynolds (Re) et le nombre de Weber (We) définis de la manière suivante :

$$Re = \frac{\rho_1 H U_g}{\mu_1} \quad (12)$$

$$We = \frac{\rho_1 H U_g^2}{\sigma} \quad (13)$$

Dans cette étude, nous utiliserons la définition suivante pour la vitesse caractéristique :

$$U_g = \frac{\rho_2 - \rho_1}{\rho_1} \sqrt{\frac{gH}{2}} \quad (14)$$

Ainsi, nous définissons l'énergie cinétique et l'énergie potentielle pour la phase n de la manière suivante :

$$Ek_n = \frac{1}{2} \int_{\Omega} C_n \rho_n \mathbf{u}^2 dV \quad (15)$$

$$Ep_n = \int_{\Omega} C_n \rho_n g z dV \quad (16)$$

avec C_n la fonction couleur définie telle que $C_n = 1$ dans le fluide n et $C_n = 0$ ailleurs. Nous définissons également C_n la fonction couleur de la phase n : $C_1 = C$ et $C_2 = 1 - C$. L'évolution en temps de l'intégrale de l'ensrophie dans les deux fluides sera également déterminée. Elle sera définie par :

$$Er_n = \int_{\Omega} C_n (\nabla \times \mathbf{u})^2 dV \quad (17)$$

En ce qui concerne la dissipation visqueuse de la phase n , elle est définie comme suit :

$$Diss_n = \int_{\Omega} C_n 2\mu_n \overline{\mathbf{S}} : \overline{\mathbf{S}} dV \quad (18)$$

Avec $\overline{\mathbf{S}} = 1/2 (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)$. Quant à l'aire interfaciale, et contrairement aux études précédentes [20, 8], où elle est approximée comme étant $\sum_{int} S_{cell}$ où $S_{cell} = \Delta x \Delta y = \Delta y \Delta z = \Delta x \Delta z$, dans cette étude, elle est évaluée à partir de la normale à l'interface \mathbf{n} , en calculant dans un cube unitaire, la surface coupée par le plan $\mathbf{n} \cdot \mathbf{x} = \alpha$. Enfin, la grandeur Vol représente le volume de fluide léger, elle permet de vérifier que le volume de fluide léger reste constant au cours du temps.

Concernant les paramètres numériques, les simulations ont été réalisées sur une grille cartésienne allant de 128^3 à 1024^3 cellules, avec un résidu de $\epsilon = 10^{-5}$ du solveur BiCGStab(2). En ce qui concerne les dérivées temporelles, un pas de temps constant $\Delta t = 10^{-3}s$ a été choisi, et 10000 incréments temporels seront ainsi calculés, correspondant à 10 secondes de simulation de l'écoulement. Pour le cas 1024^3 , il faut noter que les simulations ont été réalisées sur 6696 processeurs du cluster HPC Occigen du CINES et ont pris environ 5 jours de temps de calcul.

ρ_1	ρ_2	μ_1	μ_2	g	σ	H	Re	We
$(kg.m^{-3})$	$(kg.m^{-3})$	$(Pa.s)$	$(Pa.s)$	$(m.s^{-2})$	$(kg.s^{-2})$	(m)		
1000	900	1	1	9.81	0.45	1	221	121

TABLE 2 – Propriétés physiques des fluides et nombres adimensionnels

3.2.2 Résultats

Dans nos simulations numériques présentées sur la figure 4, nous observons les mêmes phénomènes que ceux présentés dans les travaux de Saedipour et al. [20]. Tout d'abord la montée du fluide léger vers le haut de la boîte. Ensuite, le fluide léger recircule avec plus au moins de fragmentation et dispersion. Pour l'ensemble des grandeurs que nous avons calculées, nous remarquons que les convergences en maillage par rapport aux références sont atteintes à partir d'un nombre de cellules égal à 256^3 (Figures 5, 6, 7, 8, 9, 10, 11, 12, 13, 14), à l'exception de l'aire interfaciale. Nous avons également remarqué que les courbes présentant des pics, comme l'énergie cinétique, l'énergie potentielle, l'ensrophie et la dissipation, retrouvent ces pics aux mêmes instants que la référence. Enfin, nous constatons que le volume de fluide léger Vol reste constant tout au long de la simulation, quelque soit le maillage.

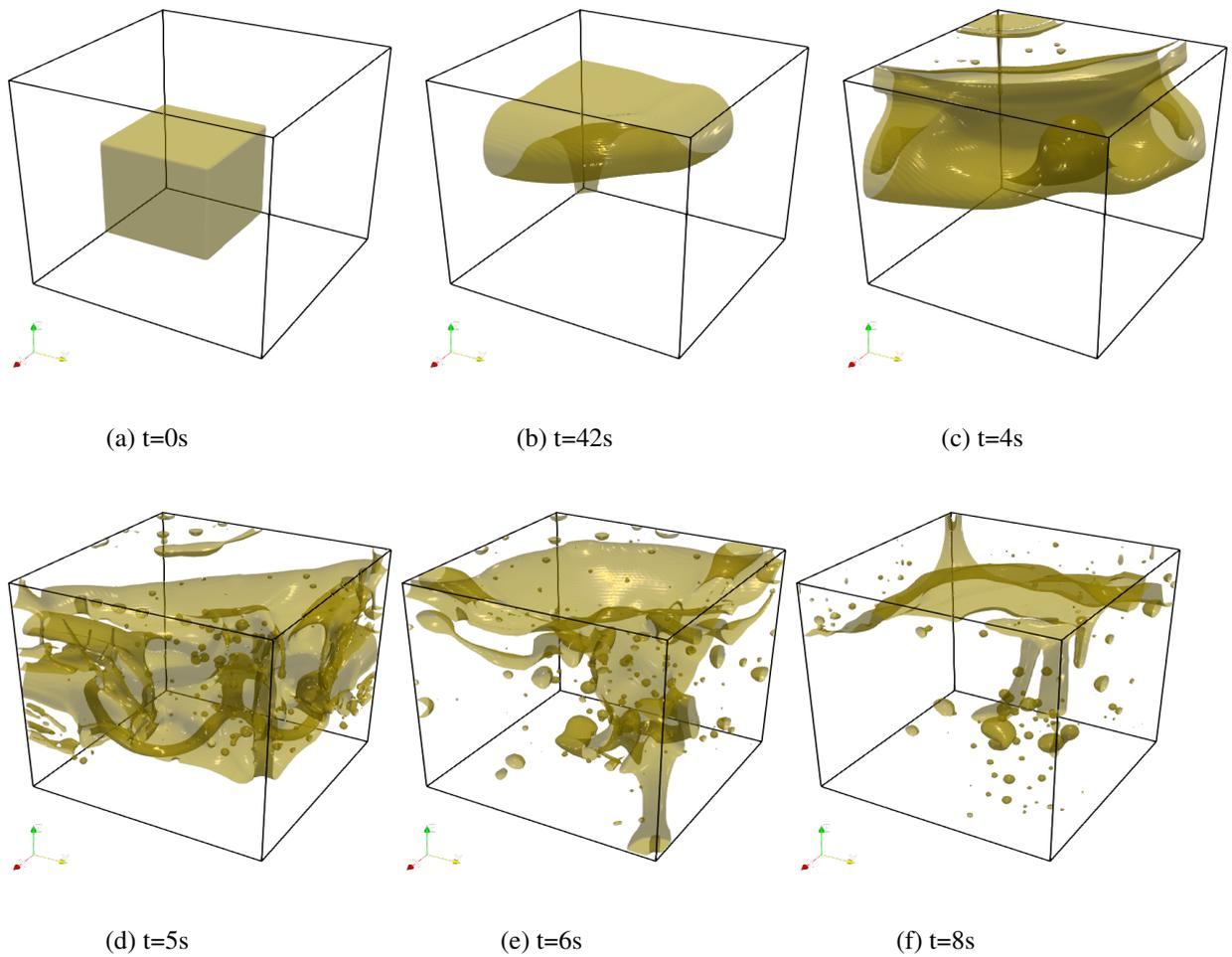


FIGURE 4 – Maillage 256³ : l'inversion de phase liquide/liquide au cours du temps. L'interface des deux fluides est représentée en jaune.

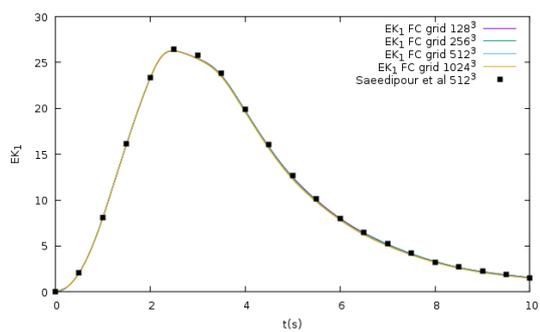


FIGURE 5 – EK_1 : l'énergie cinétique de l'eau

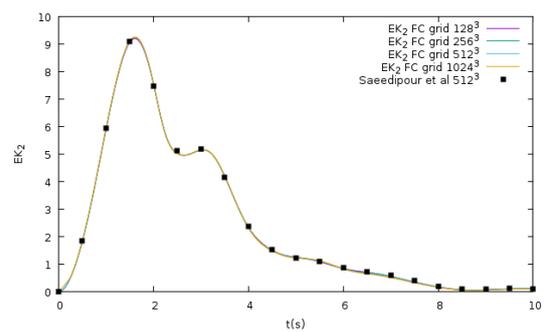


FIGURE 6 – EK_2 : l'énergie cinétique de l'huile

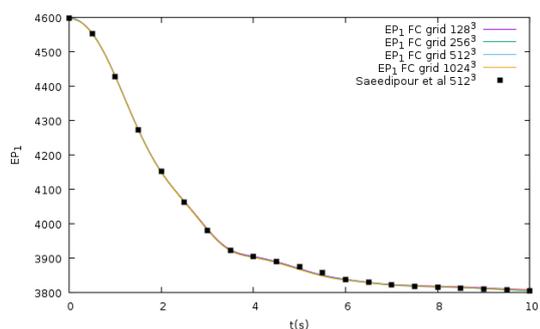


FIGURE 7 – EP_1 : l'énergie potentielle de l'eau

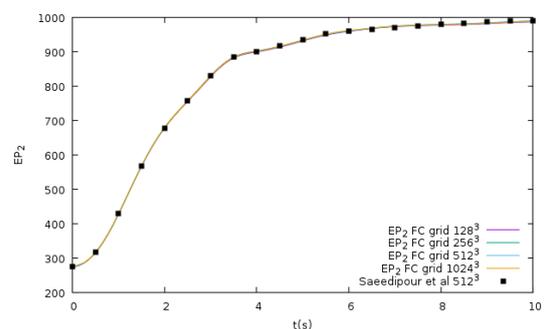


FIGURE 8 – EP_2 : l'énergie potentielle de l'huile

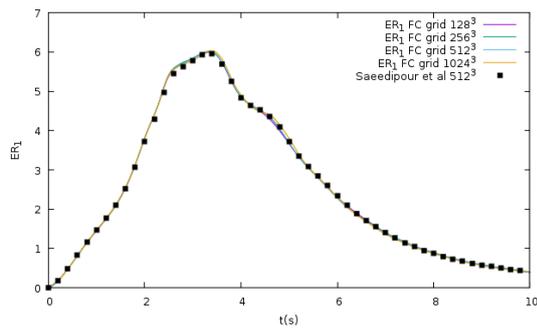
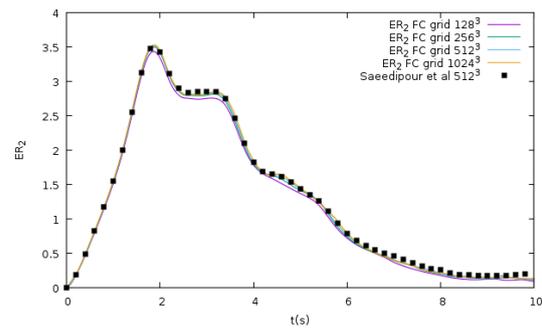
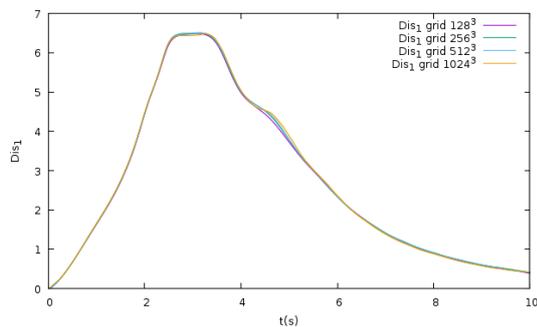
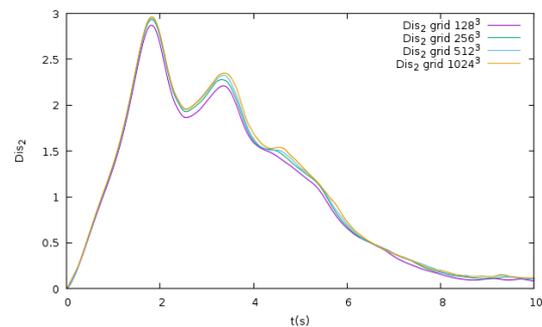
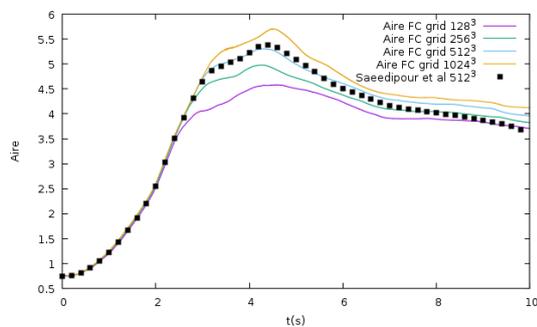
FIGURE 9 – ER_1 : l'enstrophie de l'eauFIGURE 10 – ER_2 : l'enstrophie de l'huileFIGURE 11 – Dis_1 : la dissipation de l'eauFIGURE 12 – Dis_2 : la dissipation de l'huile

FIGURE 13 – L'aire interfaciale

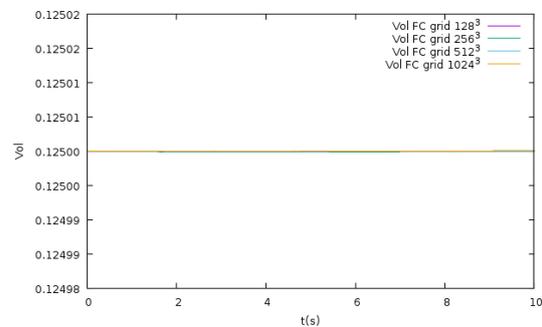


FIGURE 14 – Vol : le volume de l'huile

3.3 Caractérisation de l'instabilité 3D d'un jet visqueux à faible nombre de Reynolds

Le problème de l'instabilité d'un jet est un phénomène crucial, dans la mesure où il est impliqué dans de nombreuses applications industrielles, telles que l'industrie alimentaire ou les applications aéronautiques et aérospatiales, avec le remplissage des propulseurs de fusée avec du propergol. Les conditions d'écoulement qui y correspondent sont à faible nombre de Reynolds. Initialement étudié par Tomé et al. [22], qui ont permis entre autre de caractériser le développement des instabilités du jet en fonction du nombre de Reynolds Re et du rapport H/D entre la hauteur de la cavité et le diamètre d'injection, Peu d'études numériques sont consacrées à l'étude de l'injection d'un fluide visqueux en régime de Stokes dans une cavité. Dans ce contexte, l'intérêt de ce cas est double : d'une part, la possibilité de confronter

les solutions numériques aux solutions expérimentales. D'autre part, les forts gradients de viscosité et de densités près de l'interface engendrent des systèmes linéaires très mal conditionnés, ce qui représente un véritable défi au niveau algébrique.

3.3.1 Géométrie et paramètres physiques

Le problème étudié est le suivant : une cavité cubique de côté H , remplie d'air est étudiée. Dans sa partie supérieure, on injecte un fluide très visqueux à un débit constant, tandis que la force gravitationnelle agit avec une intensité constante g dans le sens négatif de la direction z . Pour les bords du domaine, nous avons utilisé des conditions limites de glissement sur les parois solides, et des conditions limites de sortie, dans la partie supérieure de la cavité. Les caractéristiques de la simulation numérique sont données dans le tableau 3. En outre, Tomé et al [22] donnent deux paramètres déterminants pour l'ap-

Test case	ρ_1	ρ_2	μ_1	μ_2	Re	$\frac{H}{D}$	V_0	ρ_2/ρ_1	μ_2/μ_1
1	1.1768	1800	1.85×10^{-3}	500	0.1728	12.5	0.6	1529	1.6×10^5
2	1.1768	1800	1.85×10^{-3}	300	0.6	10	1	1529	2.7×10^5

TABLE 3 – Les paramètres physiques décrivant le cas du jet visqueux. Ici, ρ_2 et μ_2 représentent respectivement la masse volumique et la viscosité du propergol, ρ_1 et μ_1 représentent respectivement la masse volumique et la viscosité du gaz, $Re = \frac{\rho_2 D V_0}{\mu_2}$ est le nombre de Reynolds associé au propergol, $\frac{H}{D}$ est le rapport d'aspect du jet et V_0 est la vitesse d'entrée. - pour des raisons de simplicité, le propergol est supposé se comporter comme un fluide newtonien.

partition de l'instabilité : le rapport d'aspect du jet $H/D > 3\pi$ et $Re < 0,52$, où Re est le nombre de Reynolds basé sur le diamètre d'entrée D et la vitesse d'entrée V_0 . Ainsi, le but de cette étude est de vérifier la capacité de la méthode entièrement couplée à traiter ce type de problème et de fournir une bonne prédiction de leur comportement physique associé. En ce qui concerne les simulations numériques, les matrices résultant de la discrétisation seront résolues avec le même solveur, le BiCGStab(2) avec un résidu de $\epsilon = 10^{-5}$.

3.3.2 Résultats

Les figures 15-16 présentent l'évolution du liquide au cours du temps pour deux nombres de Reynolds : $Re = 0.1728$ et $Re = 0.6$, en utilisant un maillage de 100^3 raffiné au niveau de l'injecteur. De ces figures, nous pouvons constater que nos simulations numériques avec le solveur tout-couplé procurent de très bons résultats quant à la physique du jet, qui sont en bon accord avec le critère théorique donné par Tomé et al [22]. En effet, le premier cas $Re = 0.1728$ correspond à une instabilité du jet, alors que dans le deuxième, $Re = 0.6$ l'oscillation du jet n'apparaît pas.

4 Conclusion

Ce travail nous a permis de mettre en place un solveur tout-couplé parallèle 3D pour la simulation des écoulements diphasiques incompressibles. Nous avons également développé une méthode permettant de suivre la dynamique de l'interface au cours du temps, tout en conservant le volume à la précision machine. À travers les deux cas test que nous avons passé, notre solveur tout-couplé est très robuste lorsqu'il fait face à des problèmes raides de forts rapports de masse volumique et de viscosité (comme

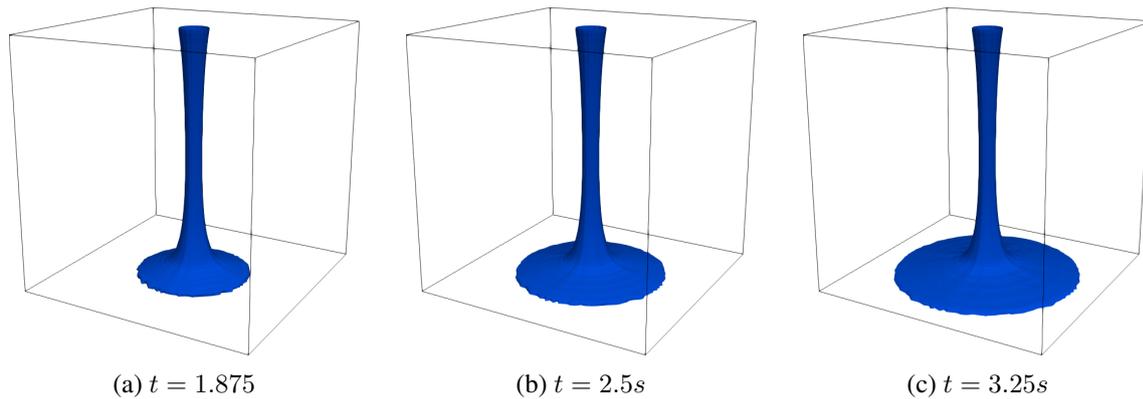


FIGURE 15 – Maillage 100^3 , $Re = 0.6$: Le jet 3D visqueux au cours du temps. L'interface des deux fluides est représentée en rose.

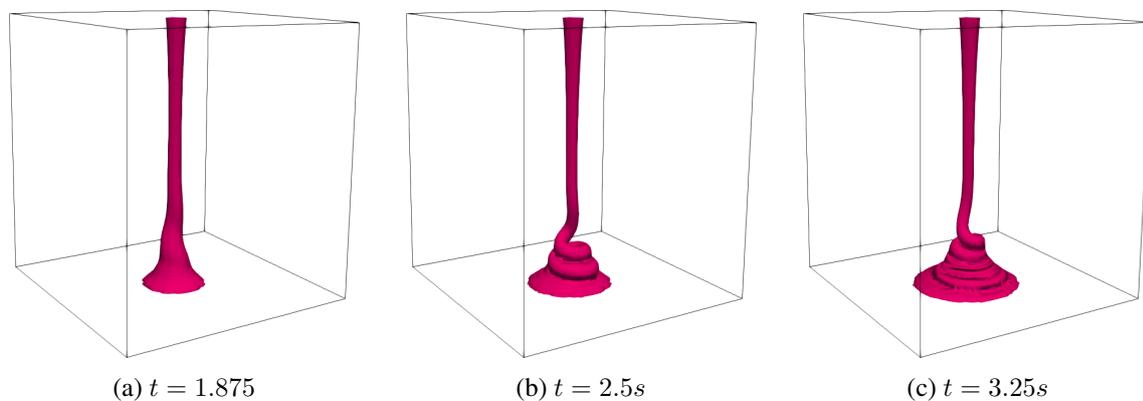


FIGURE 16 – Maillage 100^3 , $Re = 0.1728$: Le jet 3D visqueux au cours du temps. L'interface des deux fluides est représentée en vert.

dans le cas du jet visqueux). Ainsi, sur un problème de l'inversion de phase 3D, le solveur tout-couplé s'est montré très efficace pour simuler des problèmes diphasique sur un milliards de cellules. Les futures étapes à envisager sont l'extention du solveur tout-couplé à la simulation des écoulements diphasiques compressibles avec ou sans changement de phase.

5 Remerciements

Nous remercions le CINES (Centre Informatique National de l'Enseignement Supérieur) et le TGCC accordé par GENCI sous les numéros de projet A0092B06115 pour l'accès aux moyens de calcul. Nous remercions également le personnel technique et administratif de ces centres et agences de supercalculateurs pour leur aide.

Références

- [1] N. Bootland, A. Bentley, C. Kees, and A. Wathen. Preconditioners for Two-Phase Incompressible Navier–Stokes Flow. *SIAM Journal on Scientific Computing*, 41(4) :B843–B869, jan 2019.
- [2] J.U Brackbill, D.B Kothe, and C. Zemach. A continuum method for modeling surface tension. *Journal of Computational Physics*, 100(2) :335 – 354, 1992.

- [3] J-P. Caltagirone and J. Breil. Sur une méthode de projection vectorielle pour la résolution des équations de navier-stokes. *Comptes Rendus de l'Académie des Sciences - Series IIB - Mechanics-Physics-Astronomy*, 327(11) :1179 – 1184, 1999.
- [4] J-P. Caltagirone and S.Vincent. A kinematics scalar projection method (ksp) for incompressible flows with variable density. *Open Journal of Fluid Dynamics*, 05, 03 2015.
- [5] J-P. Caltagirone and S. Vincent. Sur une méthode de pénalisation tensorielle pour la résolution des équations de navier–stokes. *Comptes Rendus de l'Académie des Sciences - Series IIB - Mechanics*, 329(8) :607 – 613, 2001.
- [6] A-J. Chorin. Numerical solution of the navier-stokes equations. *Mathematics of Computation*, 22(104) :745–762, 1968.
- [7] D. Enright, F. Losasso, and R. Fedkiw. A fast and accurate semi-lagrangian particle level set method. *Computers Structures*, 83(6) :479–490, 2005. Frontier of Multi-Phase Flow Analysis and Fluid-Structure.
- [8] J.-L. Estivalezes, W. Aniszewski, F. Auguste, Y. Ling, L. Osmar, J.-P. Caltagirone, L. Chirco, A. Pedrono, S. Popinet, A. Berlemont, J. Magnaudet, T. Ménard, S. Vincent, and S. Zaleski. A phase inversion benchmark for multiscale multiphase flows. *Journal of Computational Physics*, 450 :110810, 2022.
- [9] M. Fortin and R. Glowinski. Augmented lagrangian methods : Applications to the numerical solution of boundary-value problems. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 65(12) :622–622, 1985.
- [10] K. Goda. A multistep physics technique with implicit difference schemes for calculating two- or three-dimensional cavity flows. *Journal of computational physics* 30, 1979.
- [11] J.L. Guermond and Jie Shen P. Mineev. 2006-guermond-p6011-an overview of projection methods for incompressible flows. *Comput. Methods Appl. Mech. Engrg.*, 195 :6011–6045, 2006.
- [12] G.Weymouth and D.Yue. Conservative volume-of-fluid method for free-surface simulations on cartesian-grids. *J. Comput. Physics*, 229 :2853–2865, 04 2010.
- [13] D-C. Sorensen J-J. Dongarra, L-S. Duff and Henk A. Vander Vorst. *Numerical Linear Algebra for High Performance Computers*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998.
- [14] I. Kataoka. Local instant formulation of two-phase flow. *International Journal of Multiphase Flow*, 12(5) :745 – 758, 1986.
- [15] R. J. LeVeque. High-resolution conservative algorithms for advection in incompressible flow. *SIAM J. Numer. Anal.*, 33 :627–665, 1996.
- [16] Neelesh A. Patankar N. Nangia, Boyce E. Griffith and A-P-S Bhalla. A robust incompressible navier-stokes solver for high density ratio multiphase flows. *Journal of Computational Physics*, 390 :548 – 594, 2019.
- [17] M. El Ouafa, S. Vincent, and V. Le Chenadec. Monolithic solvers for incompressible two-phase flows at large density and viscosity ratios. *Fluids*, 6(1), 2021.
- [18] M. El Ouafa, S. Vincent, and V. Le Chenadec. Navier-stokes solvers for incompressible single- and two-phase flows. *Communications in Computational Physics*, 29(4) :1213–1245, 2021.
- [19] C-H. Bruneau P. Angot and P. Fabrie. A penalization method to take into account obstacles in viscous flows. *Numerische Mathematik*, 81 :497–520, 02 1999.

- [20] M. Saeedipour, S. Vincent, and J.-L. Estivalezes. Toward a fully resolved volume of fluid simulation of the phase inversion problem. *Acta Mechanica*, 232(7) :2695–2714, July 2021.
- [21] R. Témam. Sur l'approximation de la solution des équations de navier-stokes par la méthode des pas fractionnaires (i). *Archive for Rational Mechanics and Analysis*, 32(2) :135–153, Jan 1969.
- [22] M-F. Tomé, B. Duffy, and S. McKee. A numerical technique for solving unsteady non-newtonian free surface flows. *Journal of Non-Newtonian Fluid Mechanics*, 62(1) :9 – 34, 1996.
- [23] S. Vincent, A. Sarthou, J.P. Caltagirone, F. Sonilhac, P. Février, C. Mignot, and G. Pianet. Augmented lagrangian and penalty methods for the simulation of two-phase flows interacting with moving solids. application to hydroplaning flows interacting with real tire tread patterns. *Journal of Computational Physics*, 230(4) :956–983, 2011.

Annexe I

MUF compressible de fluides non miscibles

Formulation du MUF compressible

On s'intéresse ici à l'écriture d'équations de conservation de la masse et le bilan de la quantité de mouvement en formulation conservative pour des écoulements diphasiques et compressibles. Selon les développements menés dans la subsection 3.2.5, on peut montrer à partir du développement de l'équation de conservation de la masse formulée que cette équation se réduit à une équation en pression comme suit :

$$\chi_T \frac{dp}{dt} + \beta \frac{dT}{dt} + \nabla \cdot \bar{u} = 0 \quad (\text{I.1})$$

Où : $\chi_T = \frac{1}{\rho} \left(\frac{\partial \rho}{\partial p} \right)_T$ et $\beta = \frac{1}{\rho} \left(\frac{\partial \rho}{\partial T} \right)_p$ sont respectivement, les coefficient de compressibilité isotherme, et de dilatation thermique isobare.

Pour un écoulement isotherme, on écrit la variation thermodynamique de la pression (I.1) entre deux instants comme suit :

$$\nabla \cdot \bar{u}^{n+1} + \frac{\chi_T}{\Delta t} p^{n+1} = \frac{\chi_T}{\Delta t} p^n \quad (\text{I.2})$$

Concernant l'équation du bilan de quantité de mouvement, sa formulation compressible est écrite à l'aide d'un schéma d'euler d'ordre 1 comme suit :

$$\frac{\rho}{\Delta t} \left(\bar{u}^{n+1} - \bar{u}^n \right) + \nabla \cdot \left(\rho \bar{u}^n \otimes \bar{u}^{n+1} \right) + \overline{\overline{B_{penu}}} \left(f(\bar{u}^{n+1}) - \bar{u}_\infty \right) = -\nabla p^{n+1} + \nabla \cdot \overline{\overline{T}^{n+1}} + \rho \bar{g} + \bar{F}_s, \quad (\text{I.3})$$

Si on considère un problème eau/air, les propriétés du fluide équivalent sont définies à partir de

la fonction couleur C , mais aussi de la pression p . Elles s'écrivent :

$$\rho = C\rho_{eau} + (1 - C)\frac{P + P_{ref}}{rT_{ref}} \quad (\text{I.4})$$

$$\chi = C\chi_{eau} + (1 - C)\frac{1}{P + P_{ref}} \quad (\text{I.5})$$

$$\mu = C\mu_{eau} + (1 - C)\mu_{air} \quad (\text{I.6})$$

Intégration du MUF compressible

Après une discrétisation appropriée via une approche de type volumes finis pénalisés, le système générique qui en résulte est un système linéaire non symétrique qui prend la forme suivante :

$$\begin{pmatrix} F_u & B_p^T \\ B_u & C_p \end{pmatrix} \begin{pmatrix} \bar{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \bar{f} \\ \bar{s} \end{pmatrix} \quad (\text{I.7})$$

où $F_u = M_u^{(\rho)} + N_u^{(\rho)} + L_u^{(\mu)}$, et enfin :

- $B_u \bar{u}^{n+1} \approx \int_{\Omega_{i,j,k}} \nabla \cdot \bar{u}^{n+1} dV$ est la partie discrète de la divergence,
- $C_p p^{n+1} \approx \int_{\Omega} \frac{\chi T}{\Delta t} p^{n+1} dV$ est la matrice de compressibilité,
- $\bar{s} \approx \int_{\Omega} \frac{\chi T}{\Delta t} p^n dV$ est le second membre associé à la pression,
- $B_p^T p^{n+1} \approx \int_{\Omega_{i,j,k}} \nabla p^{n+1} dV$ est l'opérateur de gradient de pression discret,
- $M_u^{(\rho)} \bar{u}^{n+1} \approx \int_{\Omega_{i,j,k}} \left(\frac{\rho^{n+1}}{\Delta t} \bar{u}^{n+1} + \overline{\overline{B_{penu} f}}(\bar{u}^{n+1}) \right) dV$ est la matrice de masse,
- $L_u^{(\mu)} \bar{u}^{n+1} \approx - \int_{\Sigma_{i,j,k}} \overline{\overline{\mathbf{T}^{n+1}}} \cdot \bar{n} dS$ est l'opérateur de contraintes visqueuses,
- $N_u^{(\rho)} \bar{u}^{n+1} \approx \int_{\Sigma_{i,j,k}} \rho \bar{u}^{n+1} \bar{u}^n \cdot \bar{n} dS$ est la partie discrète du terme convectif,
- $\bar{f} \approx \int_{\Omega_{i,j,k}} \left(\frac{\rho^n}{\Delta t} \bar{u}^n + \overline{\overline{B_{penu} \bar{u}_\infty}} + \bar{F}_s \right) dV$ est le second membre associé à la vitesse.

Les techniques de préconditionnement

Afin de construire un préconditionneur \mathcal{P} approprié, nous considérons la décomposition \mathcal{LU} par blocs du système I.7, donnée par :

$$\begin{pmatrix} F_u & B_p^T \\ B_u & C_p \end{pmatrix} = \begin{pmatrix} I_u & 0 \\ B_u F_u^{-1} & I_p \end{pmatrix} \begin{pmatrix} F_u & B_p^T \\ 0 & C_p - B_u F_u^{-1} B_p^T \end{pmatrix} \quad (\text{I.8})$$

En prenant le bloc \mathcal{U} comme préconditionneur, \mathcal{P} prendra la forme suivante :

$$P = \begin{pmatrix} F_u & B_p^T \\ 0 & C_p - B_u F_u^{-1} B_p^T \end{pmatrix} \quad (\text{I.9})$$

avec $C_p - B_u F_u^{-1} B_p^T$ est le complément de Schur du bloc de pression, qui peut être réécrit comme ci-dessous en définissant rigoureusement une nouvelle matrice G :

$$C_p - B_u F_u^{-1} B_p^T \approx \hat{S}_p + C_p = \hat{S}_p (I + \hat{S}_p^{-1} C_p) = \hat{S}_p (I + G) \quad (\text{I.10})$$

Par conséquent, l'action directe de l'inverse de \mathcal{P} sur un vecteur standard \bar{b}^* doit être effectuée par une substitution élémentaire. Ce qui peut être fait en résolvant deux systèmes linéaires et un produit matrice-vecteur. L'algorithme correspondant est le suivant :

Algorithm 1 Triangular preconditionner

- 1: Solve $\hat{S}_p (I + G) \bar{b}_p = \bar{b}_p^*$.
 - 2: Update $\bar{b}_u^* \leftarrow \bar{b}_u^* - B_p \bar{b}_p$.
 - 3: Solve $F_u \bar{b}_u = \bar{b}_u^*$.
-

Concernant l'application de l'inverse du bloc $\hat{S}_p (I + G)$ sur un vecteur b_p^* , on suppose que les matrices $(I + G)$ et \hat{S}_p sont inversibles, et que l'inverse $(I - G)^{-1}$ peut être développé comme une série infinie des différents termes qui sont présents dans l'opérateur $(I + G)$. En utilisant le développement en série de Neumann suivant [123] :

$$(I - G)^{-1} = I + G + G^2 + \dots + G^n = \sum_{k=0}^{\infty} G^k \quad (\text{I.11})$$

$$(I + G)^{-1} = (I - (-G))^{-1} = I - G + G^2 - G^3 + \dots + G^n = \sum_{k=0}^{\infty} (-1)^k G^k \quad (\text{I.12})$$

Ainsi, l'inverse du bloc $\hat{S}_p (I + G)$ peut s'écrire comme :

$$\left[\hat{S}_p (I + G) \right]^{-1} = (I + G)^{-1} \hat{S}_p^{-1} = \left[\sum_{k=0}^{\infty} (-1)^k G^k \right] \hat{S}_p^{-1} \quad (\text{I.13})$$

à l'ordre 0 \Rightarrow

$$\left[\hat{S}_p (I + G) \right]^{-1} \approx \hat{S}_p^{-1} \quad (\text{I.14})$$

à l'ordre 1 \Rightarrow

$$\left[\hat{S}_p(I + G)\right]^{-1} \approx \hat{S}_p^{-1} - \hat{S}_p^{-1}C_p\hat{S}_p^{-1} \quad (\text{I.15})$$

