HAL open science

Calcul d'interaction fluide-structure par méthode vortex et application à la synthèse d'images

Mathieu Coquerelle

▶ **To cite this version:**

Mathieu Coquerelle. Calcul d'interaction fluide-structure par méthode vortex et application à la synthèse d'images. Equations aux dérivées partielles [math.AP]. Institut National Polytechnique de Grenoble, 2008. Français. NNT : . tel-01096395

HAL Id: tel-01096395

https://theses.hal.science/tel-01096395v1

Submitted on 2 Aug 2018

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.

Copyright

Grenoble Institut National Polytechnique

N° attribué par la bibliothèque

THÈSE

pour obtenir le grade de

Docteur de Grenoble INP

Spécialité : **Mathématiques et Informatique**

préparée au laboratoire **Jean Kuntzmann**

dans le cadre de l'École Doctorale **Mathématiques, Sciences et Technologies de l'Information, Informatique**

présentée et soutenue publiquement
par

Mathieu Coquerelle

le 25 septembre 2008

Titre:
Calcul d'interaction fluide-structure par méthode de vortex et application en synthèse d'images

Directrice de thèse: **Marie-Paule Cani**
Directeur de thèse: **Georges-Henri Cottet**

Jury

Mme. Valérie Perrier,	Présidente du jury
M. Bertrand Maury,	Rapporteur
M. Jean-Michel Dischler,	Rapporteur
M. Antonio Susín Sánchez,	Examinateur
Mme. Marie-Paule Cani,	Directrice de thèse
M. Georges-Henri Cottet,	Directeur de thèse

Table des matières

Table des matières iii

I Introduction **1**

II De la simulation de fluides et de ses applications en synthèse d'images **9**

1 Modèles d'écoulements et résolution numérique **11**

1 Préambule 11
2 Modèle d'écoulement : formulation vitesse 12
2.1 Équation de Navier-Stokes compressible 12
2.2 Équation de Navier-Stokes incompressible 14
2.3 Vorticité 15
2.4 Précisions sur les différents termes de l'équation de Navier-Stokes 16
2.5 Conditions aux limites 19
3 Modèle d'écoulement incompressible : formulation vorticité 20
3.1 Préambule 20
3.2 Définition 21
3.3 Calcul de la vitesse 22
3.4 Cas particulier : 2D 23
4 Résolution des équations 24
4.1 Discrétisation 24
4.2 Eulerien vs. Lagrangien 25
4.3 Équation de Navier-Stokes : formulation vitesse par méthodes de type différences finies 31
4.4 Équation de Navier-Stokes en formulation vitesse par méthode particulaire 35
4.5 Équations de Navier-Stokes en formulation vorticité et particules Lagrangiennes 36
4.6 Équation de Navier-Stokes en formulation vorticité : méthode hybride particules/grille 43

5 Méthodes pour la simulation de fluides incompressibles en synthèse d'images . . . 49
5.1 Formulation vitesse . . . 49
5.2 Méthode de type ALE et maillages dynamiques . . . 51
5.3 Formulation vitesse par méthode particulaire . . . 52
5.4 Méthodes de vortex . . . 55
6 Bilan du chapitre . . . 57

2 Modèles de fluides multiphasiques incompressibles et transport d'interface 59
1 Fluides multiphasiques . . . 59
1.1 Préambule/Définition . . . 59
1.2 Équations de Navier-Stokes en densité et viscosité variable . . . 60
1.3 Approximation de Boussinesq . . . 62
1.4 Tension de surface . . . 63
2 Transport d'interface . . . 64
2.1 Nécessité de méthodes pour le transport d'interface . . . 64
2.2 Deux méthodes de transport . . . 65
3 Transport d'interface et fluides multiphasiques en synthèse d'images . 70
3.1 Fluides multiphasiques . . . 70
3.2 Méthodes pour le transport . . . 72
4 Bilan du chapitre . . . 79

3 Couplage fluide-structure 81
1 Introduction . . . 81
2 Structures immergées . . . 83
2.1 Solide rigide à mouvement imposé . . . 83
2.2 Membranes élastiques . . . 89
2.3 Solide rigide : couplage fluide-structure en formulation vitesse 91
3 Interactions fluide-structure en synthèse d'images . . . 93
3.1 Méthode ALE et maillages dynamiques . . . 94
3.2 Rigid Fluids . . . 95
3.3 Solides déformables . . . 96
3.4 Vortex : méthode des panneaux . . . 96
4 Bilan du chapitre . . . 97

III Méthode de vorticité pour le couplage fluide-structure 99

4 Méthode de vorticité pour la simulation de fluides 101
1 Fluide monophasique . . . 101
1.1 Préambule . . . 101
1.2 Rappel des équations et des méthodes de résolution . . . 101
1.3 Calcul de la vitesse et équation de transport . . . 103
1.4 Méthodes de remaillage et fonctions d'interpolation . . . 112

1.5 Diffusion 115
1.6 Application : simulation de fumée 116
2 Fluide à densité variable 125
2.1 Préambule 125
2.2 Reformulation de l'équation de vorticité 125
2.3 Transport de l'interface par level set 127
2.4 Application : simulation de fluide biphasique 131
3 Bilan du chapitre 134

5 Couplage fluide-structure pour les méthodes vortex 137
1 Introduction 137
2 Modèle d'interaction fluide-solide rigide en formulation vortex 138
2.1 Présentation du modèle 138
2.2 Résolution des équations 141
2.3 Validation numérique 150
2.4 Applications en synthèse d'images 151
3 Modèle de collision 159
3.1 Définition du modèle 160
3.2 Remarques sur la discrétisation 161
3.3 Applications numériques 162
4 Solides à membranes élastiques 168
4.1 Problématique 168
4.2 Approche adoptée 170
4.3 Discussion 173
5 Bilan du chapitre 174

IV Conclusion 175

A Compléments mathématiques 183
1 Lecture du document 183
1.1 Notations 183
2 Outils mathématiques 184
2.1 Numérique 184
2.2 Schémas d'intégration 185
2.3 Equation de la chaleur 186
3 Analyse 188
3.1 Développement limité 188

Bibliographie 191

Première partie

Introduction

Ce manuscrit porte sur mes travaux de recherche effectués au Laboratoire Jean Kuntzman à Grenoble sur le thème de la simulation de fluides. Plus particulièrement, je me suis intéressé à la représentation de l'écoulement de ces derniers en formulation vortex (ou « tourbillon »), point de vue alternatif aux méthodes plus traditionnelles orientées autour de calculs sur la vitesse ; je reviendrai sur ce choix dans la suite de ce chapitre. Je me suis spécialement intéressé à l'interaction de fluides et de structures telles que des solides rigides et des membranes élastiques, dans l'optique d'applications dans le domaine de la synthèse d'images.

De la nécessité de simuler des fluides

Les fluides — qu'ils soient liquides ou gazeux — ont toujours été d'un intérêt primordial pour l'Homme, car ils constituent une partie indispensable de son environnement : l'air et l'eau sont les éléments nécessaires à la survie. Dès le début de son histoire, l'Homme a cherché à mieux comprendre et à maîtriser ces éléments qui sont bien plus complexes et *imprévisibles* que les objets solides dont le comportement est plus intuitif. Pour la consommation, l'agriculture, l'hygiène et l'énergie, l'Homme a eu besoin de développer l'hydrodynamique et des technologies fiables afin d'acheminer l'eau vers les lieux de vie ou de culture : l'irrigation il y a plus de 8000 ans en Asie, l'aqueduc (dans lequel l'eau est acheminée sur plusieurs centaines de kilomètres) au début de l'empire Romain et le moulin (à eau ou à vent) ont été des avancements significatifs dans la maîtrise des liquides. L'essor des liens commerciaux a amené le développement de la navigation qui a nécessité la compréhension des phénomènes physiques liés à la poussée d'Archimède mais aussi des forces exercées sur une voile permettant de faire avancer le navire. Plus récemment, avec l'avènement de l'aéronautique s'est développée l'étude plus systématique des comportements physiques liés à l'interaction entre les fluides et les solides. Aujourd'hui, l'outil informatique est de plus en plus utilisé pour simuler les phénomènes physiques, car il permet de réduire les coûts imputés aux expérimentations, dans le but, par exemple, d'optimiser la forme d'une aile d'avion via la modélisation des phénomènes physiques. De même, la prévision des phénomènes météorologiques demande le calcul intensif de la dynamique de l'atmosphère et des océans afin de simuler le plus précisémment possible la physique. L'étude des fluides et la recherche de nouvelles techniques a donc été et est toujours d'une importance capitale — voire vitale — pour le développement de l'Homme.

L'étude mathématique du comportement des fluides a été initiée il y a 300 ans environ. Historiquement les premiers travaux sur les équations décrivant la dynamique des fluides ont été effectués par Leonhard Euler sur les fluides parfaits (incompressibles et non visqueux), basés sur les équations fondamentales de la dynamique d'Isaac Newton. Le principe est la conservation de la quantité de mouvement, qui fait intervenir la vitesse du fluide. Claude Navier et George Stokes se sont par la suite intéressés à la compressibilité des fluides ainsi qu'aux phénomènes de viscosité. De leurs travaux sont issues les équations de Navier-Stokes (*cf.* chapitre 1) aujourd'hui

utilisées pour la simulation de fluides. Ces équations sont non linéaires et donc complexes à résoudre. De surcroît, afin de capturer et calculer au mieux l'écoulement du fluide, une échelle assez fine doit être considérée, en fonction du problème étudié. La résolution de ces équations nécessite donc une forte puissance de calcul, c'est pourquoi aujourd'hui encore de nombreux travaux continuent à améliorer la précision ainsi que la rapidité des calculs. Le champ d'applications — dans la recherche aussi bien que dans l'industrie — où sont nécessaires une résolution précise des équations est très vaste : par exemple la météorologie, les chantiers navals, l'aérodynamisme (pour les avions et les voitures), l'énergie (pour les barages, éoliennes), l'astrophysique (pour le mouvement d'amas stélaires) et même en biologie (pour l'étude du comportement des cellules, de l'écoulement sanguin). Les techniques développées récemment pour le calcul d'interaction entre un fluide et des objets rigides permettent aujourd'hui de calculer efficacement et précisément l'écoulement autour d'obstacles de forme quelconque, tout en considérant les forces hydrodynamiques appliquées aux solides.

D'un point de vue plus esthétique, les fluides ont toujours été captivants du fait de leur complexité dynamique, comme par exemple la fumée s'échapant d'un feu ou les éclaboussures projetées par l'écoulement d'un liquide rencontrant rapidement un obstacle. C'est donc tout naturellement que la communauté de la synthèse d'images s'est intéressée à la modélisation de ces phénomènes. Au départ, les méthodes issues des mathématiques appliquées étaient trop prohibitives en temps de calcul. Elles étaient également considérées comme très peu intuitives pour les artistes souhaitant intégrer ces phénomènes dans leurs animations. En effet, le fait qu'un fluide soit difficilement contrôlable (à la différence du mouvement d'un personnage articulé par exemple) demande une certaine maîtrise de la physique sous-jacente pour bien choisir les paramètres en fonction de l'effet visuel désiré, et ainsi un nombre important d'aller-retours est nécessaire afin d'obtenir un résultat final satisfaisant. Cette dernière décennie, le développement de méthodes rapides et adaptées — s'éloignant parfois un peu de la résolution rigoureuse des équations d'écoulement du fluide — conjointement à celui de machines plus performantes, a permis d'intégrer ces phénomènes physiques complexes dans le domaine afin d'introduire des liquides (eau, huîle) et des gaz (pour la fumée) dans les jeux vidéos et les films d'animation.

Que cela soit dans le domaine des mathématiques appliquées ou celui de la synthèse d'images, la simulation de fluides, et plus particulièrement l'interaction de ce dernier avec des objets rigides ou déformables, est un problème très étudié du fait des nombreuses applications existantes dans l'industrie. La demande de méthodes précises et performantes permettant la simulation de phénomènes complexes fait qu'aujourd'hui de nombreux travaux de recherches continuent à faire progresser la réponse à un problème vieux de plusieurs siècles.

Des difficultés

Comme je l'ai évoqué précédemment, la simulation de fluide est une question qui a été et est encore aujourd'hui très étudiée. La complexité des phénomènes physiques apparaissant dans l'écoulement d'un fluide nécessite une discrétisation importante de l'espace afin de capturer au mieux la dynamique du fluide, notamment en présence d'obstacles et/ou de fortes turbulences. De plus, les équations à résoudre sont elles même difficiles car elles sont non linéaires. De ces deux difficultés émerge le besoin de développer des méthodes stables et efficaces afin de résoudre les équations le plus rapidement possible, tout en conservant la précision, primordiale dans la plupart des applications industrielles. Dans le domaine de la synthèse d'images, les techniques d'animation de fluides développées initialement ont souvent mis entre parenthèses la rigueur dans la résolution de la dynamique pour favoriser la simplicité d'intégration dans un programme et de manipulation du fluide, ainsi que d'augmenter la célérité du temps de calcul, critère important pour l'animateur. Dernièrement, avec l'avénement de machines plus puissantes, les travaux se sont refocalisés sur la résolution plus précise des équations, en mettant l'accent sur l'aspect localisé des phénomènes visibles en appliquant des techniques de type hiérarchiques et/ou adaptatifs fournissent un moyen de gagner du temps en se concentrant sur les régions turbulentes, où la précision est requise. Aussi, des méthodes de type hybride Eulerien-Lagrangien sont aujourd'hui souvent adoptées afin de palier certaines difficultés lors de la résolution des équations.

La manière dont sont résolues les équations est le point clef de la réussite d'une méthode. Plus précisément, dans le point de vue adopté pour la résolution de celles-ci, qu'il soit Lagrangien quand on suit une particule du fluide, Eulerien quand on s'intéresse à la vitesse en un point fixe, ou bien hybride, la discrétisation des équations va induire des erreurs et imposer des contraintes spécifiques. En Lagrangien, le maillage sous-jacent issu de la discrétisation des équations va devenir très irrégulier lors de fortes déformations dûes à l'aspect turbulent du fluide. Les calculs seront alors moins précis. Pour le point de vue Eulerien, la résolution des équations se fait sur un maillage simple et fixe, ce qui simplifie généralement la plupart des calculs, mais, par exemple, l'équation de transport de la vitesse (*cf.* premier chapitre) et des quantités convectées par le fluide sont soumises à une contrainte de stabilité importante sur le pas de temps et demande l'utilisation de méthodes adaptées pour traîter le terme non linéaire de l'équation de conservation de quantité de mouvement. Les méthodes de type hybride — comme les méthodes *Arbitrary Lagrangian-Eulerian* — permettent quant-à elles de palier en partie le problème de distortion de maillage des méthodes Lagrangiennes en adaptant la résolution des équations sur un maillage structuré se déformant avec le fluide. Ces techniques sont particulièrement adaptées au problème d'interaction fluide-structure. Seulement celles-ci restent, en pratique, difficiles à mettre en oeuvre, surtout en présence d'obstacles à forme complexe, du fait de la nécessité de générer et conserver un maillage couvrant correctement l'espace. Le choix du type de discrétisation influe donc de manière majeure sur la résolution des équations et pose en conséquence des problèmes particuliers à chaque

modèle.

La question de l'interaction de fluides entre eux ou avec des solides est complexe sur deux points : tout d'abord le modèle d'interaction qui doit permettre le couplage des forces dans deux milieux aux comportements différents puis le choix d'une représentation géométrique des domaines séparés qui doivent cohabiter. Que cela soit en synthèse d'images ou en mathématiques appliquées, les méthodes permettant d'intégrer entièrement l'interaction entre plusieurs fluides et d'autres objets (rigides ou déformables) restent souvent coûteuses et/ou difficiles à implémenter, ou restreintes à des simplifications du modèle plus général. Le problème spécifique, qui pourrait paraître simple de prime abord, de l'interaction de plusieurs fluides (l'eau et l'air, par exemple) demande une rigueur importante. Effectivement, les phénomènes physiques présents à la frontière et découlant de l'interaction des deux milieux, sont couplés au problème de transport de cette même interface entre les deux milieux. Il est alors nécessaire de faire face aux difficultés des deux sous-problèmes : celui du transport et celui de la simulation conjointe de deux milieux aux propriétés physiques différentes. L'interaction d'objets rigides ou déformables avec un ou plusieurs fluides est un problème d'autant plus complexe que pour la plupart des applications, et principalement en synthèse d'images où ces structures sont de forme arbitraire, la discrétisation des différentes domaines de calcul doit être effectuée précisément et de manière cohérente. Qui plus est, la différence des équations à résoudre dans les divers matériaux demande une méthode adaptée pour leur couplage. Actuellement, les techniques existantes ne permettent pas de répondre conjointement à tous les problèmes posés que sont l'efficacité, la précision, la simplicité d'implémentation et de manipulation, ainsi que la généricité pour l'intégration d'objets à géométrie quelconque en interaction.

Approche adoptée

J'ai effectué ma thèse en mathématiques et informatique, au sein de deux équipes : EVASION spécialisée dans le domaine de l'animation en synthèse d'images et EDP dont les membres étudient la résolution d'équations aux dérivées partielles dont les équations de Navier-Stokes font partie. J'ai travaillé sur l'élaboration de méthodes pour la simulation de fluides en ayant à l'esprit les contraintes liées aux deux domaines : rapiditié et simplicité pour le premier, et précision et efficacité pour le second.

Pour ce faire, j'ai axé mes recherches sur la formulation vortex, alternative à la formulation vitesse plus couramment considérée et qui est à la base de la théorie sur les tourbillons. Les vortex sont des élements purement rotationnels qui représentent les turbulences dans un écoulement : c'est au niveau des tourbillons que naissent les phénomènes physiques et visuels les plus intéressants. Un système d'équations simples lie la vorticité à la vitesse, cette dernière restant la base de tout mouvement du fluide. Cette formulation a depuis longtemps fait ses preuves en mathématiques appliquées car elle permet de focaliser l'effort de calcul aux endroits importants en

faisant fi des parties du domaine où le fluide n'est pas turbulent, tout en s'attachant à résoudre avec précision les équations de Navier-Stokes. La représentation de l'écoulement grâce à des particules de vorticité permet d'obtenir un fluide au comportement complexe avec un nombre d'éléments de calcul (les points de discrétisation) faibles et localisés aux endroits de forte turbulence, tout en permettant de calculer la vitesse du fluide en chaque point de l'espace. Les points de discrétisation sont donc présents au niveau des tourbillons, là où la dynamique est la plus importante et où visuellement les phénomènes les plus intéressants et complexes se déroulent. C'est donc tout naturellement que cette formulation des équations de Navier-Stokes répond aux contraintes présentes en synthèse d'images. Au commencement de ma thèse, dans ce domaine, très peu de travaux avaient été réalisés sur base de méthode vortex ; aujourd'hui plusieurs articles sont orientés autour de ce principe ; les auteurs y proposent des solutions alternatives, originales et efficaces pour l'animation de fluides. Plus précisément, j'ai concentré mes travaux sur l'utilisation d'une méthode hybride, les *Vortex In Cell*, avec remaillage, afin de palier aux contraintes de la formulation Lagrangienne et en profitant de la rapidité de la résolution d'une partie des équations, ici le calcul de la vitesse, sur une grille Eulerienne.

Concernant les interactions fluide-structure, j'ai choisi d'adopter les principes de frontières immergées et de domaine fictif. Ce dernier consiste à élargir le domaine du fluide à l'intérieur des domaines des autres milieux et de résoudre les équations du fluide dans l'intégralité du domaine, en considérant localement le comportement de l'objet *immergé*. Plus précisément, j'ai fondé mes travaux sur les méthodes de type *level set* — aujourd'hui couramment utilisées en simulation de fluide pour la synthèse d'images notamment — pour représenter implicitement l'interface entre chaque milieu, en utilisant l'information portée par la fonction afin de gérer les comportements physiques entre les différentes phases, que les interfaces soient entre fluide-fluide ou bien fluide-structure. A la différence des représentations explicites, la méthode des frontières immergées couplée à une représentation level set de l'interface permet d'intégrer très facilement un objet de forme quelconque dans un fluide. J'ai choisi de gérer le problème difficile d'interaction entre fluides et solides par une adaptation de la méthode de pénalisation, en formulation vortex.

Les avancées réalisées dans mon travail permettent ainsi de modéliser facilement les interactions entre fluides et solides tout en bénéficiant des avantages des méthodes vortex. Mon travail fournit donc un cadre innovant, efficace et précis pour la simulation de fluide en synthèse d'images.

Plan du mémoire

Dans le premier chapitre je présenterai les équations de Navier-Stokes décrivant de manière générale l'écoulement des fluides à une seule phase, ainsi que les modèles utilisés pour la résolution de celles-ci, qu'ils soient en formulation Lagrangienne ou Eulerienne. Je préciserai alors les méthodes numériques pour la résolution de ces équations, en formulation vitesse et en vorticité, celle qui nous intéresse plus particulièrement dans cette thèse. Ensuite, dans le chapitre deux, je m'intéresserai aux

équations régissant la dynamique de fluides à plusieurs phases et à leur résolution, ainsi qu'aux techniques de transport d'interface liées à ce problème, en me focalisant plus précisément sur les méthodes level set. Puis, dans le troisième chapitre, je proposerai un récapitulatif des méthodes existantes pour les interactions fluide-structure, notamment les méthodes basées sur le principe de frontières immergées sur lequel j'ai basé mes travaux. Nous verrons au chapitre quatre mes travaux sur la simulation de fluide en formulation vorticité par les méthodes particulaire et celle hybride des *Vortex In Cell* avec remaillage. Je proposerai une application pour la simulation de fumée dans un fluide à une seule phase. Subséquemment, je traiterai le problème de fluides à densité variable par une méthode originale en méthode vortex. Je proposerai alors un exemple d'application pour la simulation d'un fluide à deux phases avec transport d'interface, entre l'air et l'eau. J'exposerai ensuite, dans le chapitre cinq, mes travaux concernant le couplage entre fluides et structures, solides rigides et membranes élastiques, par une méthode de type frontière immergée, travaux basés sur un nouveau modèle de pénalisation de la vitesse en formulation vorticité. J'y présenterai aussi un nouveau modèle pour la collision inter-solides fondé sur la représentation par une fonction distance des solides immergés. Je proposerai pour ces techniques des validations numériques en comparaison des méthodes existantes, et fournirai des applications en synthèse d'images pour un traitement rapide et précis des interactions. Finalement, je concluerai ce manuscrit en proposant une discussion sur les travaux réalisés ainsi qu'en détaillant les perspectives que j'ai identifiées pour les recherches futures.

Deuxième partie

De la simulation de fluides et de ses applications en synthèse d'images

Chapitre 1

Modèles d'écoulements et résolution numérique

1 Préambule

Les équations de Navier-Stokes décrivent le comportement des fluides. Elles représentent une approximation du comportement réel des fluides, principalement des fluides Newtoniens qui nous entourent quotidiennement : les liquides comme l'eau, l'huile et les gaz comme l'air. Ces équations ont été introduites en 1822 par Claude Louis Marie Henri Navier, scientifique français né en 1785 et mort en 1835. En 1845, George Gabriel Stokes (1819-1903, Irlande) s'intéressa plus particulièrement aux fluides visqueux qu'il intégre aux équations précédemment décrites par Navier.

Les équations de Navier-Stokes permettent également de modéliser le mouvement de fluides non-Newtoniens (liquides « pâteux »), étudiés en rhéologie. Dans de tels fluides les contraintes visqueuses sont fonction du tenseur de déformation du fluide dont les lois peuvent évoluer avec le temps, en conséquence des équations plus complexes entrent en jeu.

L'équation de Navier-Stokes (cf. équation 1.3) modélise le mouvement des fluides **compressibles**, fluides qui peuvent se compresser sous l'effet de chocs. La compressibilité d'un fluide permet la propagation des ondes sonores ; c'est le changement de pression qui est responsable de la perception d'un son par l'oreille interne. La vitesse c de propagation du son dans un milieu est importante car elle permet de définir le nombre de Mach (Ma) comme le rapport entre la vitesse locale d'un fluide et c ; c'est le dépassement d'1 Mach qui provoque le *bang* d'un avion « dépassant le mur du son ». Lorsque le nombre de Mach est inférieur à 0.3 (soit tout de même $100\,m.s^{-1}$ dans l'air !), l'hypothèse de compressibilité est négligeable par rapport aux dimensions du mouvement étudié. Dans ces cas, on parle de fluides **incompressibles** : ce sont les fluides dont le volume ne change pas avec le temps. Cette approximation est valable pour la plupart des liquides et pour l'air lorsque les vitesses sont faibles, ce qui est le cas dans les phénomènes qu'on désire simuler en synthèse d'images. Pour modéliser le comportement de fluides incompressibles, on utilise l'équation de Navier-Stokes couplée à une équation comptant pour l'incompressibilité du fluide, cf. partie 2.2)

et imposant la conservation du volume. Dans la suite du document, j'utiliserai la tournure « les équations de Navier-Stokes »pour parler du couple formé par ces deux équations.

Dans cette thèse, nous nous intéresserons uniquement aux fluides que nous cotoyons quotidiennement et qui sont les bases en synthèse d'images pour, par exemple, simuler la fumée et les interactions entre l'eau et différents milieux. Ce sont donc les équations de Navier-Stokes pour les fluides Newtoniens incompressibles que je détaillerai et utiliserai tout au long de ce document.

Avant d'exposer les équations, il est nécessaire de connaître les deux points de vue utilisés pour représenter l'évolution du fluide à travers le temps : le point de vue Eulerien et le point de vue Lagrangien. Le point de vue Eulerien consiste à regarder les quantités physiques au fur et à mesure du temps en un certain point **fixe** de l'espace. Le point de vue Lagrangien quant à lui est l'observation de l'évolution des équations en une particule transportée par le fluide, et donc se déplaçant au cours du temps. La description Lagrangienne est une représentation intéressante car de nombreuses quantités sont simplement transportées par le fluide, ce qui facilite la résolution de certaines équations, ce que nous verrons par la suite.

Ce premier chapitre est dédié à l'étude et à la résolution des équations régissant le mouvement d'un fluide. En premier lieu je présente les équations régissant l'écoulement de fluides dans une formulation vitesse. J'y détaillerai plus précisément les équations de Navier-Stokes primordiales à la compréhension du problème. Dans une deuxième partie j'expose l'autre manière de considérer l'écoulement des fluides par la formulation vortex qui sera la base de mes travaux de thèse. Dans la troisème sous-partie je m'intéresserai aux méthodes existantes pour la résolution numérique des équations de Navier-Stokes, dans un formalisme Eulerien et Lagrangien. Enfin, je finirai ce chapitre par un tour d'horizon des méthodes développées à l'heure actuelle en synthèse d'images.

2 Modèle d'écoulement : formulation vitesse

2.1 Équation de Navier-Stokes compressible

Un peu toute la partie

Je présente tout d'abord l'équation de Navier-Stokes en compressible, puis en incompressible. J'explique ensuite la signification physique et l'effet de chacun des termes composant cette dernière. Finalement, je spécifierai les conditions aux bords nécessaires à la fermeture du problème.

Conservation de la masse

L'équation de conservation de la masse d'un fluide, définie par $M = \int_\Omega \rho \, d\mathbf{x}$ (où ρ est la densité du fluide en $kg.m^{-3}$) dans un domaine $\Omega(t)$, est obtenue à partir du

théorème de Reynolds. Elle s'écrit sous forme Eulerienne de la manière suivante :

$$\frac{\partial \rho}{\partial t}(\mathbf{x},t) + \nabla \cdot (\rho(\mathbf{x},t)\,\mathbf{u}(\mathbf{x},t)) = 0, \tag{1.1}$$

où $\mathbf{u}(\mathbf{x},t)$ est la vitesse du fluide en $m.s^{-1}$. Dans tout le document, afin d'en alléger la lecture, je ne spécifierai pas la position ($\mathbf{x}$) ni le temps (t) dans les équations, lorsque leur suppression n'apporte pas d'ambiguïté.

En l'absence de chocs, on peut écrire cette équation de transport comme :

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = \frac{\partial \rho}{\partial t} + (\mathbf{u} \cdot \nabla)\rho + \rho \nabla \cdot \mathbf{u} = 0. \tag{1.2}$$

Nous reconnaissons ici un terme d'advection $(\mathbf{u} \cdot \nabla)\rho$ qui représente le transport de la densité par le fluide, ainsi qu'un terme de variation de densité provenant de la compression du fluide : $\rho \nabla \cdot \mathbf{u}$. Ce dernier terme est nul dans le cas de fluides incompressibles, comme je le présente dans la partie 2.2.

Conservation du mouvement

L'équation de Navier-Stokes compressible représente la conservation de la quantité de mouvement du fluide au cours du temps. Elle nous permet, à partir de conditions initiales données, de connaître la vitesse du fluide en chaque point et en chaque instant. L'accélération — ou variation temporelle de la vitesse — est écrite comme la dérivée partielle $\partial \mathbf{u}(\mathbf{x},t)/\partial t$. L'équation aux dérivées partielles (EDP) de *conservation du mouvement* d'un fluide Newtonien sous forme conservative est la suivante :

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) - \nabla \cdot \sigma - \rho \mathbf{f} = 0, \tag{1.3}$$

où ici le signe $\otimes$ dénote le produit tensoriel et σ est le tenseur des contraintes visqueuses. L'étude de ce tenseur sort du cadre de cette thèse, il est largement étudié dans le domaine de la rhéologie et, dans le cas général, est soumis à une loi de comportement couplée à la vitesse. Une approximation de ce tenseur pour les fluides usuels peut être considéré :

$$\sigma = \mu \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T \right) + (\eta \nabla \cdot \mathbf{u} - p)\,\mathbf{I},$$

où $\mathbf{I}$ est la matrice identité, p la pression en bars [1], μ la viscosité dynamique exprimée en poiseuilles [2] et η la viscosité de volume en poiseuilles. Je présente dans les parties suivantes la loi de comportement pour la pression. Les coefficients de viscosités suivent une loi de comportement similaire à celle de la densité, et dans la suite, nous considérerons que celles-ci sont fonction de ρ car elles dépendent de la nature du fluide étudié.

[1] Le bar : $1\,bar = 10^5\,pascals = 100\,kN.m^{-2}$

[2] Le poiseuille : $1\ Pl = 1\ Pa.s = 1\ kg.m^{-1}.s^{-1}$

En développant l'équation 1.3 et en y injectant l'équation 1.2, nous trouvons l'équation de Navier-Stokes :

$$\begin{aligned} \rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) - \nabla\cdot\sigma - \rho\mathbf{f} &= 0, \\ \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} - \frac{1}{\rho}\nabla\cdot\sigma - \mathbf{f} &= 0, \end{aligned} \tag{1.4}$$

Les équations 1.2 et 1.4 doivent être couplées à une équation pour la conservation de l'énergie. Dans le cas d'un fluide Newtonien incompressible qui nous intéresse dans ce travail, ce couplage disparaît et la vitesse du fluide peut être calculée indépendamment de l'énergie.

Transport du volume

Le domaine $\Omega(t)$ est susceptible d'évoluer avec le temps par le mouvement du fluide qu'il contient. C'est en fait la surface $S(t)$ du domaine qui est déplacée, suivant la vitesse normale à cette dernière. La variation temporelle du volume peut s'écrire, à l'aide du théorème de Reynolds, de la manière suivante :

$$\frac{dV}{dt} = \int_{\Omega(t)} \nabla\cdot\mathbf{u}\, d\mathbf{x}$$

Conditions initiales

Le système d'équations 1.4 et 1.2 doit être complété de conditions initiales pour les variables $\mathbf{u}$ et ρ :

$$\mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}) \tag{1.5}$$

pour la vitesse et

$$\rho(\mathbf{x},0) = \rho_0(\mathbf{x}) \tag{1.6}$$

pour la densité du fluide. Le cas particulier de la pression sera expliqué en partie 2.4.

2.2 Équation de Navier-Stokes incompressible

L'incompressibilité d'un fluide est caractérisée par la conservation du volume en sus de la masse. Ceci peut se traduire par le fait qu'en chaque point de l'espace, il y a autant de fluide qui arrive qu'il n'en sort. Si on s'intéresse au transport du volume du fluide par ce dernier, l'équation de conservation de volume s'écrit :

$$\frac{dV}{dt} = \frac{d}{dt}\int_{\Omega} \nabla\cdot\mathbf{u}\, d\mathbf{x} = 0.$$

Par définition, le fluide conservant localement son volume, l'équation précédente se traduit par la divergence nulle locale de la vitesse :

$$\nabla \cdot \mathbf{u} = 0. \tag{1.7}$$

Grâce à cette propriété, l'équation de conservation de la masse 1.2 se reécrit comme :

$$\frac{\partial \rho}{\partial t} + (\mathbf{u} \cdot \nabla)\rho = 0, \tag{1.8}$$

qui se réduit donc à une équation de transport simple, ce qui signifie que la densité du fluide ne varie pas localement.

Nous pouvons maintenant aussi simplifier l'équation de conservation du mouvement 1.3, avec la contrainte d'incompressibilité, comme suit :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{1}{\rho}\nabla \cdot (\mu \nabla \mathbf{u}) + \frac{1}{\rho}\nabla p - \mathbf{f} = 0,$$

où le divergent du tenseur des contraintes a été développé. Sous l'hypothèse que la viscosité μ est constante, nous obtenons l'équation de Navier-Stokes incompressible

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nu \Delta \mathbf{u} + \frac{1}{\rho}\nabla p - \mathbf{f} = 0. \tag{1.9}$$

où la pression, dans ce cas particulier, permet d'assurer la contrainte d'incompressibilité ; je reviendrai sur ce point dans la partie suivante.

Les équations 1.7 comptant pour l'incompressibilité, 1.9 pour la conservation du mouvement, accompagnées de l'équation de transport 1.8 sont la base des méthodes de simulation de fluide traditionnellement utilisées en synthèse d'images.

Lorsqu'on considère une densité constante dans tout le domaine, il n'est pas nécessaire de se soucier de l'équation de transport 1.8, ce qui réduit la complexité du système d'équations à résoudre, en comparaison des fluides compressibles. Dans ce cas particulier, les équations à résoudre se réduisent aux équations 1.7 et 1.9.

2.3 Vorticité

La vorticité est définie comme le rotationnel de la vitesse :

$$\omega = \nabla \times \mathbf{u}. \tag{1.10}$$

Un fluide possède un écoulement dit *potentiel* si il est irrotationnel, *ie.* si $\omega = 0$, car la vitesse peut alors simplement s'exprimer comme le gradient d'un champ potentiel : $\mathbf{u} = \nabla \Psi$.

La vorticité tient un rôle primordial dans la dynamique des fluides que l'on souhaite étudier : c'est elle qui caractérise l'aspect turbulent d'un écoulement. En effet, le vortex est un tourbillon ; c'est la somme de nombreux tourbillons qui crée la complexité physique et visuelle d'un écoulement.

La vorticité étant le point clef de ma thèse, je la présente, ainsi que les méthodes s'y rattachant, plus en détails dans la partie 3.

2.4 Précisions sur les différents termes de l'équation de Navier-Stokes

Convection : $(\mathbf{u} \cdot \nabla)\mathbf{u}$

Le terme $(\mathbf{u} \cdot \nabla)\mathbf{u}$ est appelé convection (ou advection de la vitesse), il représente le transport de la vitesse par elle-même : c'est la variation de la vitesse en fonction du champ de vitesse environnant. La convection est un terme non linéaire qui rend en pratique difficile la résolution des équations de Navier-Stokes. Je présente en partie 4.3 une méthode largement utilisée en synthèse d'images pour approcher la convection.

De manière plus générale, on utilise le terme *advection* pour désigner le transport d'une quantité physique par le fluide. Je détaille cette notion dans la partie 4.2 dans laquelle est présentée la notion de dérivée particulaire.

Diffusion : $\nu \Delta u$

La viscosité d'un fluide est caractérisée par la quantité de vitesse diffusée : de la même manière que la chaleur se diffuse au cours du temps dans un objet, la vitesse se diffuse dans le milieu. Physiquement, le phénomène de diffusion représente un transfert d'énergie au niveau des petits échelles, où, la friction des particules (physiques) du fluide ralenti le mouvement du fluide en le dissipant sous forme de chaleur.

Intuitivement, plus la viscosité ν est importante, moins le fluide sera turbulent et plus le champ de vitesse sera « lisse »(par analogie, plus un objet est conducteur, plus la chaleur s'y répandra rapidement). Cette propriété physique tient un rôle très important dans la dynamique des fluides. En effet, le type d'écoulement peut être entièrement caractérisée par une grandeur sans dimension nommée **nombre de Reynolds**, du nom d'Osborne Reynolds (1842-1912) scientifique irlandais. Pour étudier l'écoulement, il convient d'adimensionner l'équation de Navier-Stokes incompressible 1.9 :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \mathrm{Re}^{-1}\Delta \mathbf{u} + \nabla p = 0$$

où le nombre Re issu de cette adimensionnement est défini en fonction des grandeurs caractéristiques suivantes : U la vitesse du fluide, ν sa viscosité et L la grandeur caractéristiques associée au phénomène à observer (grandeur physique décidée par convention) :

$$\mathrm{Re} = \frac{LU}{\nu}.$$

Si l'on s'intéresse au problème de l'étude de l'écoulement autour d'un cylindre, à Reynolds fixé — où L est pris comme le diamètre de l'objet— deux fluides aux propriétés différentes (viscosité et vitesse, par exemple) produiront un écoulement similaire.

Lorsque la viscosité tend vers 0 (ou de manière équivalente lorsque Re tend vers l'infini), on parle de *fluide parfait*. C'est un fluide où les forces de frottements

avec les autres milieux (une aile d'avion par exemple) sont négligeables et ainsi considérér comme nulles. Dans un fluide non parfait, il existe une *couche limite* autour d'un solide au contact du fluide dans laquelle la vitesse diminue en fonction de la distance au solide (l'illustration 1.1). L'épaisseur de cette zone de cisaillement dépend entièrement du nombre de Reynolds caractérisant l'écoulement.

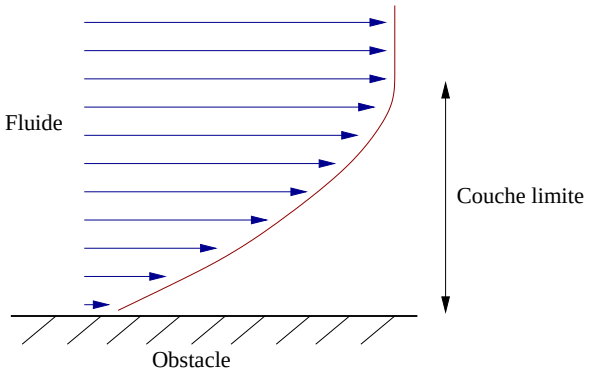


FIG. 1.1 – Effet de la viscosité sur l'écoulement d'un fluide à vitesse horizontale constante au centre du domaine en contact avec un solide. L'épaisseur et le profil de la couche limite sont fonction du nombre de Reynolds.

Les écoulements peuvent être regroupés en trois catégories :
- en dessous de Re = 2300, on parle de fluides *laminaires* : les phénomènes visqueux sont plus importants que les forces inertielles,
- entre 2300 et 4000, on parle de fluides *transitifs*,
- au dessus de 4000, de fluides *turbulents* où les forces inertielles prennent le dessus sur la viscosité.

Pour exemple, une personne nageant dans une piscine à une vitesse d'un mètre seconde créera un écoulement avec $\mathrm{Re} \simeq 10^6$. Une personne marchant à la même vitesse dans l'air le fera dans un écoulement à $\mathrm{Re} \simeq 10^5$, moins turbulent que dans l'eau. Ceci n'est pas intuitif car l'on pense que l'air est plus turbulent que l'eau, mais en fait c'est l'inverse : la viscosité cinématique de l'air est plus faible (de l'ordre de 10 fois plus petit) que celle de l'eau.

La Simulation Numérique Directe (SND, *Direct Numerical Simulation* en anglais) s'intéresse à la résolution des équations de l'écoulement du fluide au niveau de chaque point de discrétisation, où l'échelle des phénomènes étudiés est limitée à celle de l'échantillonage spatial. En comparaison, les méthodes de type *Large Eddy Simulation* (LES) introduisent dans les équations les effets du mouvement du fluide à des échelles plus petites que la discrétisation. Un facteur important est qu'en SND, telles qu'employées aujourd'hui dans notre domaine, la précision des calculs est limitée à des nombres de Reynolds de l'ordre de quelques milliers. La cause de cette limitation provient de la discrétisation des équations — limitation due à la puissance des ordinateurs — à des échelles qui ne peuvent capturer les phénomènes importants apparaissant dans les écoulements très turbulents. Lorsque la diffusion

est bien moins importante que les forces d'inertie, l'utilisation d'un maillage grossier ne permettra pas de résoudre la dissipation du mouvement à des échelles bien plus petites. Numériquement, la diffusion peut être considérée comme un lissage de la vitesse d'une particule de fluide sur les particules voisines, où son *rayon d'action* (l'épaisseur physique du noyau de lissage) est proportionnel à la viscosité. Si $\nu \ll h$ (où h est le pas d'espace du maillage), l'effet visqueux devient alors très faible, jusqu'à atteindre l'ordre des erreurs numériques dues à la discrétisation en espace. Afin de pouvoir calculer correctement un écoulement donné, la fréquence spatiale d'échantillonnage $1/N$ doit être choisie en fonction du régime du fluide, par :

$$N^3 \geq \mathrm{Re}^{2.25}$$

en 3D. Le nombre de points de discrétisation nécessaire croît donc très rapidement pour des fluides turbulents, la relation précédente implique qu'il faut une grille de taille 300^3 pour simuler convenablement un écoulement à $\mathrm{Re} = 2000$.

En pratique, un tel échantillonage est très coûteux (en coût mémoire et en temps de calcul), c'est pourquoi l'utilisation de maillages non uniformes peut s'avérer d'un intérêt majeur pour les simulations de fluides où localement Re est grand, comme par exemple lors de l'étude de la trainée derrière un cylindre. En contrepartie, ces méthodes sont souvent plus difficiles à mettre en oeuvre.

Pression : ∇p

La variation de pression d'un fluide est proportionnelle aux forces qui s'y appliquent. Plus elles sont élevées et plus le gradient de pression ∇p sera important. Nous connaissons tous la pression atmosphérique, celle-ci est généralement plus faible en altitude qu'au niveau du sol. En effet, si on suppose que la vitesse de l'air et sa variation sont nulles et que l'unique force extérieure est la gravité $\mathbf{f} = \mathbf{g} \simeq (0, 0, -1)$ (considérée comme constante), l'équation de Navier-Stokes se réduit alors à :

$$\nabla p = \rho \mathbf{g}.$$

On voit ainsi que le gradient de pression est vertical et la pression est donc plus forte à basse altitude car :

$$\frac{dp}{dz} = -\rho \text{ et ainsi } p = -\rho z + \text{cste},$$

c'est le poids de l'atmosphère qui est plus important au sol qu'en altitude. C'est grâce à ce fait qu'il est possible de mesurer (approximativement) l'altitude grâce aux altimètres qui mesurent la pression et la comparent à la pression considérée fixe au sol. De manière plus générale, nous pouvons comprendre le gradient de pression comme la direction dans laquelle se déplace le fluide : le fluide se déplace des hautes pressions vers les basses pressions. C'est entre autre ce phénomène qui est utilisé en météorologie pour prédire la direction du vent.

Dans un fluide compressible, l'équation de Navier-Stokes doit être fermée par une loi de comportement qui relie la pression à la densité : $F(\rho, T, p)$ où T est la température. Dans le cas où on considère des écoulement isothermes et isentropiques, on choisit :

$$p = \gamma \rho^{\alpha},$$

avec γ et α des constantes dépendant du fluide. En incompressible l'équation de conservation de la masse se transforme en un équation de transport sur la densité. La pression quant à elle ne joue plus du tout le même rôle. On peut montrer qu'elle intervient comme un multiplicateur de Lagrange associé à la condition d'incompressibilité. Dans ce cas, la pression se calcule donc a posteriori pour imposer la divergence nulle du fluide, comme nous le verrons en pratique dans la partie 4.3.

2.5 Conditions aux limites

Les conditions limites représentent physiquement le comportement des grandeurs étudiées. Par exemple, on peut considérer la température à la surface d'une plaque électrique comme constante et se diffusant dans l'air. Autrement, la vitesse normale d'un fluide au niveau d'un obstacle doit être nulle si l'objet n'est pas poreux. Numériquement, les conditions limites sont nécessaires pour fermer le système d'équations étudié, en plus des conditions initiales.

Considérons un domaine $\Omega(t)$ dans lequel évolue un fluide. Les conditions aux limites peuvent être soit « ouvertes »si le domaine est non borné, on considère alors les grandeurs à l'infini, soit « fermées», quand le fluide est contenu dans un espace borné, soit périodiques, où le signal se répète dans toutes les directions. Ce dernier type de condition, même s'il n'est pas physiquement réaliste pour les fluides, est souvent employé dans les simulations car c'est le plus simple à implémenter ; elle permet entre autre d'utiliser la transformée de fourier (notée TF) pour résoudre des équations complexes de manière rapide.

Il existe deux types principaux de conditions aux limites dans un domaine fini non périodique :

- les conditions de type Dirichlet : la valeur de la solution pour une grandeur Q est connue au bord du domaine :

$$Q(\mathbf{x})|_{\partial\Omega} = g(\mathbf{x}),$$

  .

- les conditions de type Neumann : la valeur de la dérivée normale de la solution est connue au bord du domaine :

$$\left.\frac{\partial Q(\mathbf{x})}{\partial \mathbf{n}}\right|_{\partial\Omega} = g(\mathbf{x}),$$

  où $\mathbf{n}$ est la normale à la surface $\partial\Omega$.

Dans le cas de l'équation de conservation de mouvement, une condition de Dirichlet homogène permet de définir une vitesse nulle sur les bords du domaine :

cette condition est souvent appelée *no-slip* (« pas de glissement »). C'est une bonne approximation de la vitesse d'un fluide de viscosité assez élevée au contact avec un objet solide ; elle est extraite du modèle de Stokes pour les fluides visqueux. Dans ce cas, l'équation au bord est :

$$\mathbf{u}(\mathbf{x})|_{\partial\Omega} = \mathbf{0}.$$

Un autre type d'écoulement consiste à considérer que le fluide peut glisser sur le bord du domaine, tout en assurant la non pénétration du fluide à l'extérieur du domaine (ou inversement, pas d'entrée de matière) en imposant une vitesse normale nulle aux bords :

$$\mathbf{u}(\mathbf{x}) \cdot \mathbf{n}|_{\partial\Omega} = 0.$$

Cette condition est celle employée en présence de fluides parfaits.

Dans le cadre de fluides incompressibles, il est nécessaire de préciser également les conditions aux bords pour la pression. Celles-ci sont plus dures à trouver car, dans ce cas particulier, la pression sert à forcer la condition d'incompressibilité. Dans la littérature, pour des conditions de type Dirichlet sur la vitesse, une condition de Neumann homogène est choisie pour la pression.

Les conditions aux limites sont primordiales pour la résolution des équations, il est nécessaire de bien y réfléchir afin de les résoudre correctement.

3 Modèle d'écoulement incompressible : formulation vorticité

3.1 Préambule

Depuis plusieurs années les méthodes basées sur la formulation en vorticité des équations de Navier-Stokes sont de plus en plus utilisées comme une alternative aux méthodes basées sur la formulation en vitesse de ces mêmes équations. En effet, dans beaucoup de cas pratiques la vorticité est localisée. Par exemple, l'étude des turbulences produites par l'écoulement d'un fluide autour d'un cylindre, montre que la partie intéressante du phénomène se situe dans le sillage de ce dernier, où se forment une série de vortex, pour un écoulement caractéristique avec un nombre Reynolds supérieur à 100 environ. Il apparait alors plus intéressant de s'intéresser à la vorticité présente dans le fluide et de focaliser les calculs aux endroits où elle se trouve. La vitesse du fluide dans tout le domaine est alors obtenue à partir du champ de vorticité.

En pratique, les méthodes basées sur les particules de vorticité (*cf.* 4.5) bénéficient de nombreux avantages en comparaison des méthodes Euleriennes traditionnelles en vitesse. Grâce aux particules de vorticité, il est possible d'obtenir la vitesse en chaque point de l'espace — qu'il soit borné ou non — de manière simple par la résolution d'une équation de Poisson. La vorticité est localisée : elle recouvre très

rarement la totalité du domaine de calcul ; ainsi on peut comprendre que cette représentation par particules permet de focaliser la précision des calculs aux endroits turbulents où la vorticité est présente.

C'est cette manière de résoudre les équations de Navier-Stokes que j'ai adoptée dans ma thèse. Mais avant d'entrer dans les détails techniques de résolution, je présente dans la partie ci-dessous les équations régissant le mouvement des fluides du point de vue de la vorticité.

3.2 Définition

Tout d'abord, rappelons ici que la vorticité est définie comme le rotationnel du champ de vitesse :

$$\omega = \nabla \times \mathbf{u} \tag{1.11}$$

Il est important de noter ici qu'en dimension 2, la vorticité est un scalaire. Cette propriété simplifie les calculs et permet ainsi une résolution plus efficace, en comparaison de la formulation vitesse, des équations du fluide, comme nous le verrons par la suite.

L'équation de vorticité s'obtient simplement en reprenant l'équation de conservation du mouvement pour un fluide incompressible 1.9 et en la reexprimant via l'opérateur rotationnel :

$$\nabla \times \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nu \Delta \mathbf{u} + \frac{1}{\rho} \nabla p - \mathbf{f} \right) = 0, \tag{1.12}$$

après développement et simplifications, nous obtenons l'équation de conservation de vorticité :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega - (\omega \cdot \nabla)\mathbf{u} - \nu \Delta \omega - \nabla \times \mathbf{f} = 0. \tag{1.13}$$

ou plus simplement en ôtant les forces extérieures :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = (\omega \cdot \nabla)\mathbf{u} + \nu \Delta \omega. \tag{1.14}$$

Il est important de noter que la divergence de la vorticité est nulle, trivialement par construction nous obtenons :

$$\nabla \cdot \omega = \nabla \cdot (\nabla \times \mathbf{u}) = 0.$$

Il est nécessaire d'expliquer comment on obtient l'équation 1.14 à partir de l'équation 1.12 ainsi que les sens de chaque terme. Nous allons développer chaque terme en partant de la gauche vers la droite, commençons par le plus simple, l'accélération :

$$\nabla \times \left(\frac{\partial \mathbf{u}}{\partial t} \right) = \frac{\partial \omega}{\partial t}$$

tout naturelement. Puis développons le terme de convection :

$$\nabla \times ((\mathbf{u} \cdot \nabla)\mathbf{u}) = (\mathbf{u} \cdot \nabla)\omega - (\omega \cdot \nabla)\mathbf{u}$$

Nous reconnaissons ici un terme d'advection de la vorticité $(\mathbf{u} \cdot \nabla)\omega$ qui sera utilisé lors de la formulation lagrangienne de l'équation de vorticité, ainsi qu'un terme d'*étirement* $(\omega \cdot \nabla)\mathbf{u}$. Il est facile de vérifier que ce terme d'étirement s'annule dans un écoulement bidimensionel. En 3D cet étirement peut être compris comme la déformation de la vorticité par le champ de vitesse, qui modifie l'orientation et l'intensité locale d'un vortex, bien que la circulation totale du fluide soit conservée. La diffusion de la vorticité vient quelque peu compenser cet étirement, et, lorsque la viscosité est faible, l'étirement crée des instabilités importantes dans l'écoulement. C'est pourquoi un écoulement 2D deviendra 3D pour des nombres de Reynolds importants.

Le terme de diffusion de la vitesse se retrouve transcrit en $\nu\Delta\omega$:

$$\begin{aligned} \nabla \times (\nu\Delta\mathbf{u}) &= \nu\Delta\omega + \nabla\nu \times \Delta\mathbf{u} \\ &= \nu\Delta\omega \end{aligned}$$

si on prend une viscosité constante dans le fluide. Un fait très important dans la formulation vorticité est la disparition du terme lié au gradient de pression :

$$\begin{aligned} \nabla \times \left(\frac{1}{\rho}\nabla p\right) &= \frac{1}{\rho}\nabla \times (\nabla p) + \nabla\left(\frac{1}{\rho}\right) \times \nabla p \\ &= 0 \end{aligned}$$

comme le rotationel d'un champ scalaire est nul et la densité constante.

Je détaillerai les conditions aux limites pour la vorticité dans la partie 4.5.

3.3 Calcul de la vitesse

L'équation d'incompressibilité du fluide $\nabla \cdot \mathbf{u} = 0$ induit que la vitesse peut s'écrire naturellement comme le rotationnel d'un champ ψ :

$$\mathbf{u} = \nabla \times \psi \tag{1.15}$$

La fonction ψ est appelée *fonction courant* (*stream function* en anglais) et est orthogonal à la vitesse. En 2D, cette fonction est un scalaire dans la direction orthogonale à l'écoulement. Dans ce cas particulier, une courbe isovaleur de la fonction courant représente une ligne de courant (*stream line*) du champ de vitesse sous-jacent.

Intéressons nous maintenant au lien entre vitesse, vorticité et fonction courant :

$$\begin{aligned} \omega &= \nabla \times \mathbf{u} = \nabla \times (\nabla \times \psi) \\ &= \nabla(\nabla \cdot \psi) - \Delta\psi \end{aligned}$$

En choisissant une fonction courant soléidale (à divergence nulle), nous obtenons l'équation de Poisson :

$$\omega = -\Delta\psi. \tag{1.16}$$

Ainsi, à partir de la vorticité nous pouvons extraire la vitesse du fluide par la simple résolution d'une équation de Poisson et de l'équation 1.15.

Il est aussi possible d'obtenir la vitesse du fluide en formulation vorticité, dans un domaine non borné, grâce à la loi de Biot-Savart. Provenant de l'électromagnétisme, celle-ci permet d'exprimer la vitesse du fluide comme le produit de convolution d'un noyau avec la vorticité :

$$\mathbf{u} = \mathbf{u}_\infty + \mathbf{K} \star \omega \tag{1.17}$$

où $\mathbf{K}$ est le rotationnel de la fonction de Green solution élémentaire de l'équation de la chaleur, et $\mathbf{u}_\infty$ est la vitesse du fluide à l'infini provenant du fait qu'un champ de vorticité définit un champ de vitesse à un champ de vitesse irrotationnel prêt. L'équation 1.17 peut aussi s'écrire :

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_\infty + \int_{\mathbb{R}^d} \mathbf{K}(\mathbf{x} - \mathbf{y}) \times \omega(\mathbf{y})\, dy \tag{1.18}$$

où d est la dimension de l'écoulement. Je reviendrai plus en détails sur la construction de ce noyau $\mathbf{K}$ pour la résolution numérique des équations, dans la partie 4.5.

Les conditions aux limites sont ici plus difficiles à identifier et à mettre en oeuvre qu'en formulation vitesse. En effet, pour l'étude d'un fluide, ce sont les conditions aux limites pour la vitesse qui nous intéressent. Or, ce n'est pas la grandeur manipulée directement en méthode vortex, et des conditions pour la vorticité et la fonction courant doivent donc être trouvées afin qu'elles correspondent à celles souhaitées sur la vitesse. Nous verrons par la suite qu'une condition de non pénétration peut être imposée sur les bords du domaine en choisissant des conditions de Dirichlet homogène pour ψ, et grâce un terme de vorticité supplémentaire dans le cas de l'équation utilisant la loi de Biot-Savart. La question du traitement de conditions de non glissement (dans le cas d'un fluide visqueux) induit, dans la formulation intégrale, un mécanisme de flux de vorticité partant des bords vers l'intérieur du domaine. Je reviendrai sur ce problème dans les chapitres 3, 4 et 5, où, dans ce dernier, je proposerai une nouvelle méthode simple et efficace pour intégrer implicitement des conditions aux bords de non glissement en formulation vorticité.

3.4 Cas particulier : 2D

Il est intéressant de noter que dans le cas d'un écoulement dans un plan, l'équation de vorticité 1.14 s'en trouve simplifiée car la vorticité devient un scalaire et le terme d'étirement s'annule. En effet, en 2D le champ de vitesse se situe dans un plan, $\{x, y\}$ par exemple, et donc $\mathbf{u} = (u_1\ u_2\ 0)^T$. Si l'on s'intéresse maintenant au rotationnel de $\mathbf{u}$, on obtient $\omega = (0\ 0\ w_3)^T$ car $u_3 = 0$ et $\partial Q/\partial z = 0$ quelque soit Q. Ainsi $(\mathbf{u} \cdot \nabla)\omega = 0$.
L'intérêt est donc double : tout d'abord l'équation à résoudre est plus simple et se réduit à une équation de transport-diffusion :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = \nu \Delta \omega \tag{1.19}$$

(si on ôte les forces extérieures) ; ensuite, la grandeur physique ω est un champ scalaire et donc son stockage en mémoire sur l'ordinateur est beaucoup plus faible

que dans le cas tridimensionnel. Nous verrons aussi par la suite que cette réduction à une inconnue permet de gagner en temps de calcul lors de la résolution numérique des équations.

Comme je l'ai noté plus haut, pour un écoulement 2D turbulent, avec un fort nombre de Reynolds, la dynamique du fluide ne se contente plus de rester dans le plan avec l'apparition d'instabilité du fait de l'étirement des vortex. L'écoulement devient alors tridimensionel.

4 Résolution des équations

Préambule

La résolution numérique des équations de Navier-Stokes consiste à calculer discrètement l'évolution de la vitesse d'un fluide au cours du temps. Je présenterai pour débuter la notion de discrétisation en partie 4.1 ainsi que les différents types de discrétisation couramment utilisés pour résoudre les équations, notamment celles de l'espace dans les parties traitant des méthodes Eulerienne, Lagrangienne et ALE en partie 4.2. Puis je rentrerai dans le détail des méthodes de résolution des équations de Navier-Stokes qu'elles soient en formulation vitesse ou vorticité et d'un point de vue Eulerien ou Lagrangien.

4.1 Discrétisation

Les équations de Navier-Stokes décrivent le mouvement de fluides de manière continue. Leur complexité ne permet évidemment pas de les résoudre de manière analytique. Nous allons donc approximer ces équations en les *discrétisant*, c'est-à-dire en nous intéressant aux phénomènes qui se passent en certains points (d'un nombre fini) discrets de l'espace et du temps. C'est en ces points qu'on va résoudre les équations et définir les approximations des dérivées spatiales et temporelles. L'objectif principal est de commettre le minimum d'erreurs par rapport à la solution de l'équation continue.

L'espace et le temps sont donc discrétisés afin de trouver une solution aux équations. La discrétisation doit être effectuée à une échelle au moins aussi fine que le phénomène qu'on souhaite simuler/observer : si on souhaite simuler l'écoulement autour d'un avion nous découperons l'espace de manière plus « grossière »que pour de l'eau dans un verre, tout en gardant le même nombre de points. Il est important de noter que le coût de résolution des équations, en terme de temps de calcul et d'occupation mémoire, dépend du nombre de points de discrétisation. Divers algorithmes possèdent différentes complexités — linéaires, quadratiques, etc. — ce qui est, en plus de l'erreur d'approximation, un critère primordial dans le choix de la méthode.

Erreur de discrétisation

On nomme erreur de discrétisation l'erreur entre la fonction continue f et son approximation discrète $\tilde{f}$:

$$\varepsilon = \frac{|\tilde{f}(x_i, t_n) - f(x,t)|}{|f(x,t)|}.$$

C'est cette erreur qu'on cherche à minimiser dans la résolution numérique des équations.

Différences finies

Afin de résoudre de manière discrète une EDP, il est nécessaire de discrétiser les différents opérateurs différentiels présents dans l'équation. Ainsi on utilise des schémas de *différences finies* qui permettent d'approximer ces opérateurs en fonction des valeurs $\tilde{f}$ connues sur un *maillage*, notion que je présente plus bas dans la partie 4.2. Les schémas de différences finies possèdent deux caractéristiques importantes : l'ordre d'approximation (*ie.* l'erreur commise par rapport à la solution exacte) et le support sur le maillage (*ie.* le nombre de points de discrétisation utilisés ainsi que leur localité). Le lecteur néophyte en méthodes numériques pourra se reporter à l'annexe 2.1 pour une explication plus détaillée sur ce sujet.

Schémas d'intégration

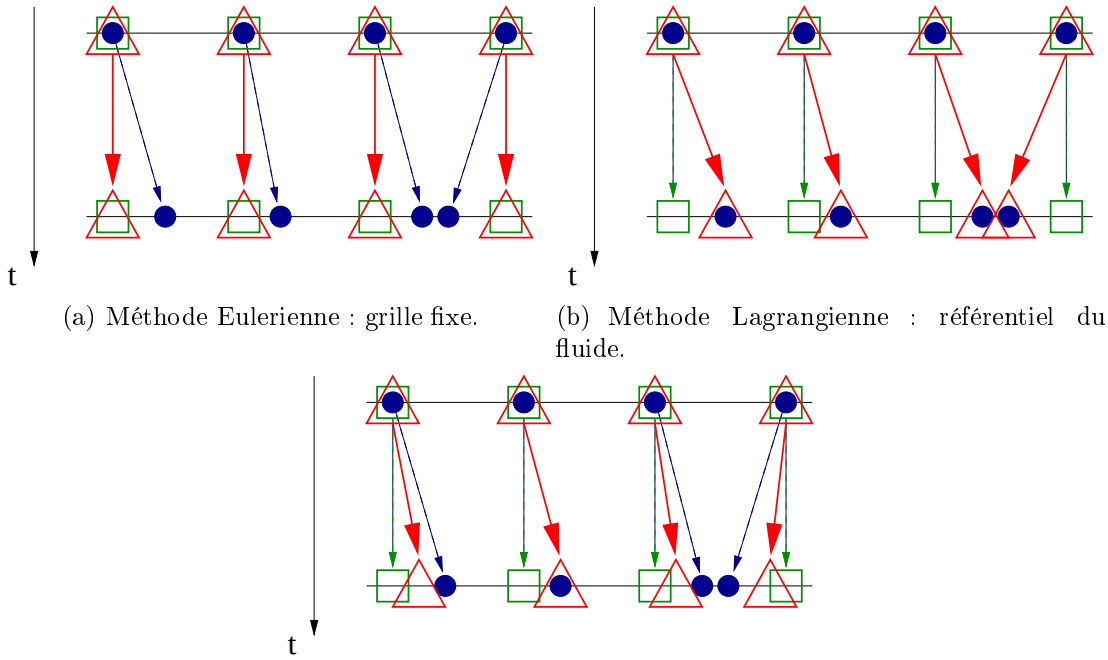
Il existe plusieurs manières de résoudre une EDP grâce aux différences finies. Suivant le problème auquel on s'intéresse, les schémas d'intégration ont différentes propriétés, principalement concernant la stabilité et l'ordre des erreurs commises. Il est hors de propos de présenter les nombreux schémas d'intégration existants ainsi que leur particularité face aux équations posées. Par contre, je spécifierai à chaque fois qu'il est nécessaire de le faire, la méthode utilisée. Je renvoie le lecteur à l'annexe 2.2 pour une étude plus détaillée de différents schémas d'intégration.

Pour résumer, le choix d'une méthode d'intégration se fait principalement en fonction des trois critères suivants :

- sa stabilité,
- l'erreur de discrétisation associée,
- le coût algorithmique pour résoudre l'équation.

4.2 Eulerien vs. Lagrangien

Je propose dans cette partie une présentation ainsi qu'une comparaison des différents points de vue adoptés pour résoudre numériquement les équations du fluide. Une comparaison graphique en figure 1.2 en résume les principes.



(a) Méthode Eulerienne : grille fixe.

(b) Méthode Lagrangienne : référentiel du fluide.

(c) Méthode *ALE* : maillage arbitraire.

FIG. 1.2 – Exemples de méthodes de discrétisation de l'espace et leur évolution au cours du temps. Les carrés verts représentent les points fixes d'une grille Eulerienne, les ronds bleus représentent le déplacement du milieu (de particules) et les triangles rouges sont les sommets du maillage.

Discrétisation Eulerienne

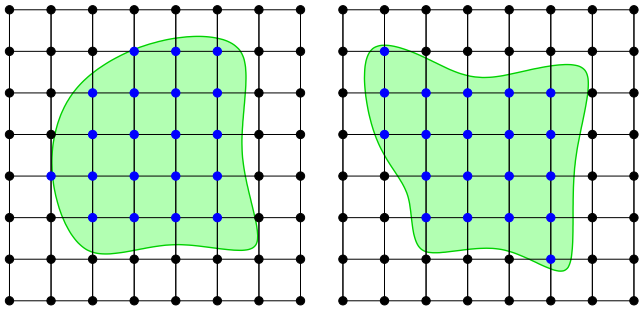
La discrétisation Eulerienne de l'espace consiste à subdiviser ce dernier en intervales fixes, qui ne bougent pas avec le temps. Ces intervales sont définis par des points de discrétisation qui forment un **maillage**. Pour notre problème, cela revient à observer le mouvement du fluide aux points de discrétisation et de le faire évoluer en fonction des points alentours. Pour ce faire, on découpe l'espace dans une grille régulière que l'on appelera grille Eulerienne. En une dimension, on subdivise l'intervale $[A, B]$ dans lequel on souhaite résoudre l'équation, en $N-1$ intervales et N points de grille. Chaque point x_i a pour coordonnées :

$$x_i = A + (i-1)h_i,\ x_1 = A,\ x_N = B,$$

avec $h_i = x_{i+1} - x_i$. Lorsque les intervales h_i sont identiques pour tout i, on parle de grille uniforme et $\forall i\ h_i = h = \frac{B-A}{N-1}$. Le même découpage est effectué pour la dimension « temps ».

La généralisation de cette discrétisation à un espace de plus haute dimension est similaire, la figure 1.3 montre un exemple de grille Eulerienne discrétisant le domaine d'un fluide en 2D.

Cette représentation est très intéressante car les points de discrétisation sont



(a) Temps t. (b) Temps $t+1$.

FIG. 1.3 – Exemples d'une grille Eulerienne uniforme discrétisant un fluide en 2D dont le domaine évolue avec le temps. Les points de discrétisation en bleu sont à l'intérieur du domaine, en noir à l'extérieur.

statiques et ne souffrent ainsi pas de problème de distortion qui peuvent apparaître avec une formulation Lagrangienne. Il est aussi important de noter que cette représentation Eulerienne est très pratique et simple à intégrer en informatique car elle consiste en un tableau à M dimensions (où M est le nombre de dimensions) dans lequel les accès aux valeurs autour de $f(x_i)$ sont très rapides. Malheureusement cette représentation n'est pas sans défaut. En effet, comme nous l'avons expliqué dans la partie 4.2, les équations de transport peuvent être simplifiées dans un référentiel Lagrangien : la résolution de ce type d'équations dans une formulation Eulerienne induit des erreurs. Par exemple, nous verrons dans la partie 2 que le transport de la surface d'un liquide sous forme Eulerienne ne permet pas de conserver parfaitement celle-ci. Toutefois, les méthodes Euleriennes sont largement utilisées du fait de leur simplicité.

Méthode Lagrangienne

Dans le cas particulier des équations décrivant l'écoulement des fluides, il est possible de se baser dans un autre référentiel que celui d'une grille Eulerienne pour décrire le mouvement : c'est le point de vue Lagrangien qui consiste à observer le fluide en des particules transportées par celui-ci. Les points de discrétisation formant le maillage sont alors ces particules. Une illustration d'un maillage Lagrangien est donné dans la figure 1.4.

Une particule située en $\mathbf{X}$ se déplace à la vitesse $\mathbf{u}(\mathbf{X})$:

$$\frac{d\,\mathbf{X}}{dt} = \mathbf{u}(\mathbf{X}).$$

Celle-ci transporte une quantité Q qui évolue au cours du temps, comme par exemple la densité que nous avons présentée au début de ce chapitre. Comme la particule est transportée par le fluide, elle résout intrinséquement l'équation d'advection, et pour simplifier l'écriture, nous écrivons les équations en formulation Lagrangienne grâce

à la *dérivée particulaire* définie comme suit :

$$\frac{D\,Q}{Dt} \equiv \frac{\partial\,Q}{\partial t} + (\mathbf{u}\cdot\nabla)Q. \tag{1.20}$$

Ainsi, dans le référentiel de la particule, il est possible de reécrire la loi de conservation de la masse 1.8 :

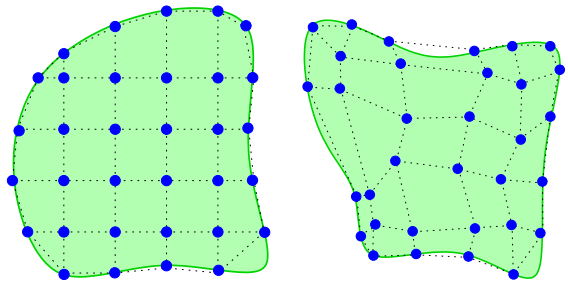
$$\frac{D\,\rho}{Dt} = 0.$$

Autrement dit : la densité portée par une particule dans un fluide incompressible ne varie pas au cours du temps. Toutes les quantités scalaires Q simplement transportées par le fluide telles que $\frac{D\,Q}{Dt} = 0$ sont appelées **scalaires passifs**. Etant donné que la partie droite de ce type d'équations est nulle, il n'y a pas d'erreur de discrétisation : l'erreur globale se réduit aux erreurs commises pour calculer la vitesse du fluide.

Par contre, cette représentation n'est pas sans désavantages. Comme les particules suivent le mouvement du fluide, leur répartition dans l'espace — supposée initialement *bonne* — peut se trouver distordue, disparate ou au contraire trop dense, ce qui aurait pour conséquence de moins bien représenter la fonction discrétisée et ainsi de poser des problèmes pour la résolution des équations. De plus, la définition des opérateurs de différenciation est quant à elle plus complexe du fait de l'aspect moins bien structuré des points de discrétisations. Des méthodes de « remaillage »sont souvent utilisées pour pallier ces problèmes. J'aborderai plus en détails ces points lors de la résolution des équations de Navier-Stokes en formulation vitesse et vorticité par particules Lagrangiennes, dans les parties 4.3 et 4.4.

Application à notre problématique La traduction des équations d'incompressibilité, de conservation du mouvement et de la masse en formulation Lagrangienne est la suivante :

$$\begin{aligned}
\frac{D\,\mathbf{u}}{Dt} &= \nu\Delta\mathbf{u} - \frac{1}{\rho}\nabla p + \mathbf{f} \\
\nabla\cdot\mathbf{u} &= 0, \\
\frac{D\,\rho}{Dt} &= 0.
\end{aligned}$$



(a) Temps t. (b) Temps $t+1$.

FIG. 1.4 – Exemples d'une grille Lagrangienne discrétisant un fluide en 2D dont le domaine évolue avec le temps. Les points de discrétisation en bleu suivent les particules du fluide. Des traits pointillés sont dessinés pour mieux visualiser le mouvement des particules.On peut visualiser ici un des défauts dont peuvent souffrir les méthodes Lagrangiennes : la mauvaise répartition des points de discrétisation par le transport.

Méthodes ALE

Les méthodes Euleriennes observent l'écoulement d'un point de vue fixe, les méthodes Lagrangiennes du point de vue des particules de fluide. Les méthodes ALE — pour *Arbitrary Lagrangian Eulerian* — quant à elles se basent en un point de vue arbitraire et permettent d'exprimer les équations dans ces nouveaux référentiels se déplaçant en fonction de critères choisis. Les méthodes Eulerienne et Lagrangienne en sont des cas particuliers. Par exemple, au bord d'un milieu la formulation Lagrangienne est plus adaptée car elle permet de suivre exactement cette frontière ; d'un autre côté, aux endroits très turbulents, une représentation Eulerienne semble plus adaptée car elle empêchera les problèmes de maillage dus à une déformation trop importante de celui-ci. Les méthodes ALE permettent de « mélanger »ces deux types de formulations en un même maillage évoluant au cours du temps.

Soient $\mathbf{x}$, $\mathbf{X}$ et χ les coordonnées des maillages Euleriens, Lagrangiens et ALE respectivement. Soient $f(\mathbf{x},t)$, $f^*(\chi,t)$ et $f^{**}(\mathbf{X},t)$ une quantité dans référentiels associés. Il est possible de définir la dérivée temporelle de f^* en fonction de f, f^{**} ainsi que d'une vitesse $\mathbf{c}$:

$$\frac{\partial f^*}{\partial t} = \frac{\partial f^{**}}{\partial t} - \frac{\partial f}{\partial \mathbf{x}} \cdot \mathbf{c} = \frac{\partial f^{**}}{\partial t} - (\mathbf{c} \cdot \nabla) f, \tag{1.21}$$

où $\mathbf{c} = \mathbf{u} - \mathbf{v}$ est la vitesse relative entre les maillages ALE et Lagrangien par rapport au référentiel Eulerien, avec $\mathbf{u}(\mathbf{X},t)$ la vitesse d'une particule du fluide et $\mathbf{v}(\chi,t)$ celle des points du maillage ALE. Il est aisé de voir qu'en prennant $\chi = \mathbf{x}$ (un maillage Eulerien) l'équation 1.21 est équivalente à la définition de la dérivée particulaire définie en 1.20, car $\mathbf{v} = 0$ et ainsi $\mathbf{c} = \mathbf{u}$. Si le maillage ALE est le même que le maillage Lagrangien : $\chi = \mathbf{X}$ alors $\mathbf{u} = \mathbf{v}$ et ainsi $\frac{\partial f^*}{\partial t} = \frac{\partial f^{**}}{\partial t}$.

Le détail des calculs permettant d'obtenir, entre autres, l'équation 1.21 est disponible dans le document [DHPRF04] expliquant très clairement les trois formulations,

leurs liens, et proposant des exemples d'application ainsi qu'une longue liste de références sur la gestion du maillage.

La formulation ALE est d'une très grande flexibilité car elle permet de déplacer les sommets d'un maillage de manière totalement libre, suivant des critères choisis pour le problème étudié. Elle est un bon compromis entre les méthodes Eulerienne et Lagrangienne. Malheureusement il ne supprime pas les difficultés qui peuvent se poser lorsque la déformation du maillage est importante et demande plus de temps de calcul pour la génération de celui-ci à chaque pas de temps.

Application à notre problématique La traduction des équations d'incompressibilité, de conservation du mouvement et de la masse en formulation ALE est la suivante :

$$\begin{aligned}
\left.\frac{\partial \mathbf{u}}{\partial t}\right|_{\chi} + (\mathbf{c}\cdot\nabla)\,\mathbf{u} &= \nu\Delta\mathbf{u} - \frac{1}{\rho}\nabla p + \mathbf{f}, \\
\nabla\cdot\mathbf{u} &= 0, \\
\left.\frac{\partial \rho}{\partial t}\right|_{\chi} + (\mathbf{c}\cdot\nabla)\,\rho &= 0.
\end{aligned}$$

Ici le signe $|_{\chi}$ désigne la valeur prise en les points du maillage ALE.

Bilan

La méthode ALE est une méthode couramment utilisée en mathématiques appliquées car son aspect générique permet une adaptation maillage au cours du temps, en fonction des contraintes numériques : bords du fluide se déplaçant et zones de fortes turbulences par exemple. Par contre, elle est beaucoup moins présente dans le domaine de la synthèse d'images. La préférence pour les méthodes Euleriennes peut être expliquée par leur simplicité d'implémentation et du fait que les temps de calcul avec ces dernières est plus court qu'avec les méthodes ALE. Les méthodes purement Lagrangiennes quant à elles souffrent de leur difficulté à résoudre précisément les équations, du fait des distortions du maillage au cours du temps. Ce problème est souvent un frein à l'utilisation des méthodes Lagrangiennes, nous verrons dans la partie 4.6 que des méthodes de remaillage peuvent être adoptées pour remédier à ce problème, dans le cadre des équations de Navier-Stokes en formulation vortex.

Je présente dans la partie 5 les méthodes principales utilisées en synthèse d'images pour la simulation de fluide, qu'elles soient Euleriennes, Lagrangiennes ou de type ALE.

4.3 Équation de Navier-Stokes : formulation vitesse par méthodes de type différences finies

Principe

Une des méthodes les plus communes pour la résolution des équations de Navier-Stokes, en mathématiques appliquées aussi bien qu'en synthèse d'images, est un algorithme de type fractionnaire où chaque terme de l'équation de conservation de mouvement est traité séparément. Cette technique a été initialement introduite et étudiée par Chorin [Cho68] où l'équation de Navier-Stokes est résolue sans la contrainte d'incompressibilité, puis la solution obtenue est projetée sur l'espace des champs de vitesse à divergence nulle. L'article phare de Jos Stam : « Stable Fluids »(dont je reparlerai dans la partie 5) est le premier à avoir présenté dans le domaine du graphique, une méthode simple et stable — donc adaptée à ce domaine — pour résoudre les équations. Le principe des algorithmes basés sur cette méthode est le suivant :

1. soit une vitesse $\mathbf{u}^n$ sur une grille Eulerienne, nous cherchons à calculer $\mathbf{u}^{n+1}$,
2. ajout des forces extérieures (la gravité par exemple) : $\mathbf{f}$,
3. traîtement du terme non-linéaire : $(\mathbf{u} \cdot \nabla)\mathbf{u}$,
4. calcul de la diffusion : $\nu \Delta \mathbf{u}$,
5. contrainte d'incompressibilité : $\nabla \cdot \mathbf{u}$.

Ces 5 étapes sont traitées de manière itérative : à chaque temps nous avançons du temps t (itération n) au temps $t + \delta t$ (itération $n + 1$).

Je présente dans les sous-parties suivantes les méthodes communément utilisées pour chacune des étapes présentées ci-dessus. Par contre je ne parlerai pas de l'étape d'ajout des forces extérieures car celle-ci ne présente aucune difficulté particulière : elle est ajoutée trivialement à l'accélération.

Traitement du terme non-linéaire

Une des difficultés dans la résolution de l'équation de conservation de mouvement provient de sa non-linéarité : du terme $(\mathbf{u} \cdot \nabla)\mathbf{u}$. Effectivement, dans une discrétisation du champ de vitesse sur une grille Eulerienne, ce terme impose une contrainte de stabilité sur le pas de temps δt utilisé si il est traité de manière explicite, contrainte nommée condition *CFL* pour Courant-Friedrichs-Lewy, scientifiques ayant étudié ce problème. Je reviendrai dans la partie 4.5 pour une discussion plus fournie concernant ce sujet.

Cette contrainte implique que $\delta t \leq \frac{\delta x}{|u|_{max}}$ où δx est le pas d'espace et $|u|_{max}$ la vitesse maximum du fluide. La condition CFL est très restrictive car le pas de temps doit être adapté de manière linéaire en fonction du pas d'espace utilisé (pour une vitesse donnée) et inversement linéaire en fonction de la vitesse du fluide. Elle sous tend qu'une particule de fluide ne peut pas parcourir plus d'une cellule (dans une grille Eulerienne). Pour pallier cette limitation, il peut être envisagé de résoudre

l'équation de manière semi implicite, qui est inconditionnellement stable. Par contre, cette méthode est très coûteuse car elle demande l'utilisation d'un algorithme itératif de type descente de Newton pour trouver la vitesse $\mathbf{u}$.

Plus communément sont utilisés des schémas de différences finies adaptés à l'équation de transport : les schémas *Weighted Essentially Non-Oscillatory* dits WENO, introduit par Liu, Osher et Chan [XDL94]. Ces schémas sont particulièrement adaptés aux fluides, dans lesquels un contraste de vitesse important provoque des comportements différents sur l'erreur introduite par la résolution de l'équation d'advection. L'ordre de l'erreur de discrétisation est grandement réduit grâce à ces schémas qui permettent aussi de limiter les oscillations apparaissant lors du transport de quantités discontinues comme pour les phases d'un fluide sur lesquelles je reviendrai en partie 2. Malheureusement l'utilisation de ces schémas ne permet pas de s'extraire de la contrainte sur le pas de temps, elle est néanmoins très courante dans le domaine des mathématiques appliquées où bien souvent des contraintes de pas de temps plus fortes entre en jeu, notamment à cause de la diffusion.

La méthode des caractéristiques, aussi nommée **Semi-Lagrangienne**, est une alternative intéressante et aujourd'hui largement utilisée en synthèse d'images pour contrecarrer les problèmes précédemment cités. Elle est inconditionnellement stable et à un coût de calcul linéaire en le nombre de points de discrétisation. Cette méthode basée sur la méthode des charactéristiques consiste à « tracer en arrière dans le temps »une particule du fluide pour résoudre l'équation de transport sur la vitesse :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = 0$$

ou encore

$$\hat{\mathbf{u}}^{n+1} = \mathbf{u}^n - \delta t \left(\mathbf{u}^n \cdot \nabla\right) \mathbf{u}^n.$$

J'ai noté par un « chapeau »sur $\mathbf{u}$ le fait que cette vitesse n'est pas la solution complète de l'équation de Navier-Stokes, mais une solution partielle comptant pour l'advection uniquement.

Pour chaque point de grille $\mathbf{x}$, il suffit de parcourir le champs de vitesse $\mathbf{u}$ à reculon et d'interpoler la vitesse située à un pas de temps précédent pour résoudre l'équation (cf. figure 1.5). Ainsi, si $\mathbf{s}(\mathbf{x}, t)$ est la trajectoire du point $\mathbf{x}$ au temps t en fonction du champs de vitesse $\mathbf{u}$, la solution à l'équation précédente est :

$$\hat{\mathbf{u}}^{n+1}(\mathbf{x}) = \mathbf{u}^n(\mathbf{p}(\mathbf{s}, t - \delta t))$$

où plus précisément la trajectoire est définie comme :

$$\mathbf{s}(\mathbf{x}, t - \delta t) = \mathbf{s}(\mathbf{x}, t) + \delta t \int_t^{t-\delta t} \mathbf{u}^n(\mathbf{s}(\mathbf{x}, \sigma))\, d\sigma.$$

Cette méthode inconditionnellement stable possède le désavatange d'être plus diffusive que les méthodes de type WENO, ce qui est une limitation si on veut faire

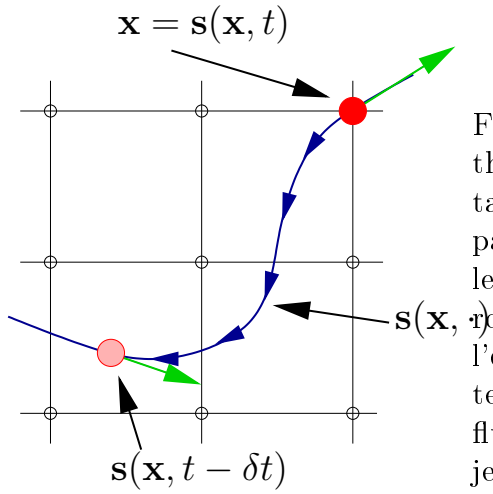


FIG. 1.5 – Illustration de la méthode semi-lagrangienne : en partant du point rouge foncé $\mathbf{x}$, nous parcourons en arrière dans le temps le champs de vitesse jusqu'au point rouge clair. La vitesse solution de l'équation $\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}$ est la vitesse en ce point. Les vitesses du fluide sont dessinées en vert, la trajectoire $\mathbf{s}$ en bleu.

des calculs précis, mais qui peut être acceptable dans le cadre de rendu de fluides en informatique graphique. Cependant, l'interpolation du champ de vitesse au pas de temps précédent implique une plus grande diffusion numérique que le traitement de la convection par des schémas WENO plus précis.

Une fois que cette partie transport de l'équation de Navier-Stokes est résolue, ils faut encore résoudre le terme de diffusion ainsi que la contrainte d'incompressibilité qui est primordiale.

Diffusion

Le terme de diffusion de la vitesse $\nu\Delta\mathbf{u}$ dans les équations de Navier-Stokes peut être traité de manière explicite ou bien implicite. L'équation de diffusion (ou équation de la chaleur dans le cas d'un algorithme à étape comme présenté précédemment) est :

$$\frac{\partial \mathbf{u}}{\partial t} = \nu\Delta\mathbf{u}. \tag{1.22}$$

Cette équation peut être résolue de différentes manières : implicitement, explicitement ou dans le cas particulier de conditions périodiques, dans l'espace des fréquences. Pour les deux premières méthodes, les schémas de différences finies permettent de résoudre numériquement l'équation, en résolvant :

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \delta t \nu \Delta \mathbf{u}^{n+1}$$

pour une résolution implicite inconditionnellement stable, ou en résolvant :

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \delta t \nu \Delta \mathbf{u}^n$$

pour un traitement explicite de la chaleur. Ce dernier est soumis à la condition de stabilité $\delta t < \frac{2D\,\delta x^2}{\nu}$, où D est la taille du vecteur $\mathbf{u}$. Je renvoie le lecteur dans la partie 1.5 pour un développement plus détaillé des méthodes de résolution pour la diffusion que j'ai utilisées au cours de ma thèse.

Méthode de projection

Une fois que l'équation de Navier-Stokes a été résolue sans le terme de pression et la contrainte d'incompressibilité, il est nécessaire de projeter la vitesse obtenue sur l'espace des vecteurs à divergence nulle. Je rappele l'équation de Navier-Stokes initiale :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nu \Delta \mathbf{u} = -\frac{1}{\rho}\nabla p + \mathbf{f}.$$

En enlevant le terme sur le gradient de pression, nous obtenons l'équation suivante sur une accélération auxiliaire $\frac{\partial \mathbf{u}^*}{\partial t}$ qui n'est pas à divergence nulle :

$$\frac{\partial \mathbf{u}^*}{\partial t} = -(\mathbf{u} \cdot \nabla)\mathbf{u} + \nu \Delta \mathbf{u} + \mathbf{f}.$$

A partir de cette accélération et de la vitesse au pas de temps précédent, nous obtenons la vitesse auxiliaire $\mathbf{u}^*$:

$$\mathbf{u}^* = \mathbf{u}(t) + \delta t \frac{\partial \tilde{\mathbf{u}}}{\partial t}(t).$$

Nous pouvons simplement trouver la solution finale pour la vitesse au pas de temps suivant en réinjectant l'équation précédente pour reformer l'équation de Navier-Stokes complète :

$$\mathbf{u}(t + \delta t) = \mathbf{u}^* - \delta t \, \frac{1}{\rho}\nabla p$$

ou en multipliant par la densité :

$$\rho \, \mathbf{u}(t + \delta t) = \rho \, \mathbf{u}^* - \delta t \, \nabla p.$$

En prenant le divergent de cette équation, nous obtenons une équation de Poisson pour la pression avec comme un terme de forçage fonction de la divergence de la vitesse :

$$\nabla \cdot \left(\rho \mathbf{u}^{t+\delta t}\right) = \nabla \cdot (\rho \mathbf{u}^*) - \delta t \, \nabla \cdot (\nabla p) = 0$$
$$\frac{1}{\delta t}\nabla \cdot (\rho \mathbf{u}^*) = \Delta p$$

Si on considère que ρ est constant — $\nabla \rho = \mathbf{0}$ — alors l'équation précédente se réduit à :

$$\frac{1}{\delta t}\rho \nabla \cdot \mathbf{u}^* = \Delta p \tag{1.23}$$

Les solveurs les plus courants (comme *FishPack* par exemple) permettent de résoudre les équations de Poisson de manière rapide en discrétisant l'opérateur Δ, je renvoie le lecteur à la partie 1.5 pour plus de détails quant à la résolution de l'équation de la chaleur, proche de ce problème. C'est souvent la partie la plus coûteuse dans la résolution des équations.

4.4 Équation de Navier-Stokes en formulation vitesse par méthode particulaire

Dans cette partie je présente succintement une approche utilisant des particules Lagrangiennes pour la résolution de l'équation de Navier-Stokes. En effet, une représentation naturelle du fluide consiste à considérer, non pas une grille fixe, mais des particules suivant le mouvement du fluide. Nous avons vu que cette définition Lagrangienne des points de discrétisation permet de simplifier les équations en incorporant l'équation de transport.

La méthode des *Smoothed Particle Hydrodynamics* (SPH) a été tout d'abord développée en astrophysique pour la simulation du mouvement de gaz compressibles [GM77]. Dans ce principe, un ensemble de particules, notées par l'indice p, un champ Q défini en ces points de discrétisation, tel que :

$$Q(\mathbf{x}, t) = \sum_p v_p Q_p \delta(\mathbf{x} - \mathbf{x}_p(t)) \tag{1.24}$$

où δ est la masse de Dirac, v_p est le volume de la particule et Q_p la quantité portée par celle-ci. Le nombre de points de discrétisation étant fini en pratique, ce champ est régularisé sur les particules. Q^ε représente ce dernier :

$$Q(\mathbf{x}, t) \simeq Q^\varepsilon(\mathbf{x}, t) = \sum_p v_p Q_p \zeta_\varepsilon(\mathbf{x} - \mathbf{x}_p(t)). \tag{1.25}$$

où ζ est une fonction *cutoff* à symétrie sphérique et ε en est le support. Généralement, le choix d'une Gausienne pour ζ est considéré ; je reviendrai sur cette question dans la partie suivante concernant la résolution de l'équation de vorticité par particules.

Les particules sont transportées par le fluide :

$$\frac{D\,\mathbf{x}_p}{Dt} = \mathbf{u}(\mathbf{x}_p, t)$$

de manière Lagrangienne. Dans le cadre d'un fluide compressible, la densité est elle aussi transportée sur la particule par :

$$\frac{D\,\rho_p}{Dt} = -\rho_p \nabla \cdot (\mathbf{u}(\mathbf{x}_p, t)),$$

de la même manière que le volume est modifié par

$$\frac{D\,v_p}{Dt} = -v_p \nabla \cdot (\mathbf{u}(\mathbf{x}_p, t)).$$

L'équation de Navier-Stokes compressible

$$\frac{\partial\,\mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nabla \cdot \sigma = 0$$

est elle aussi écrite au niveau des particules :

$$\frac{D\,\mathbf{u}_p}{Dt} = \nabla \cdot (\sigma(\mathbf{x}_p, t)).$$

Il convient alors de définir les opérateurs au niveau des particules, sous une forme conservative afin de réduire les erreurs dues à la discrétisation. Par exemple, le gradient de la vitesse est obtenu par :

$$\nabla \mathbf{u}(\mathbf{x}_p) = \sum_q v_q(\mathbf{u}_q - \mathbf{u}_p)\zeta'_\varepsilon(\mathbf{x}_q - \mathbf{x}_p).$$

ainsi, il est possible de définir l'ensemble des équations à résoudre pour la variation de la densité, du volume et de la vitesse d'une particule (en admettant connue la loi de comportement de σ, je ne m'attarderai pas sur ce point).

Ce formalisme est très intéressant pour les fluides compressibles mais il devient plus ardu à mettre en oeuvre dans le cas de fluides incompressibles. En effet, pour simuler ceux-ci, la contrainte d'incompressiblité doit être imposée, généralement par le biais de la pression, comme nous l'avons vu précédemment. Bien que des travaux [HA07, ESE07] proposent des méthodes pour tenter de palier cette difficulté, la résolution de l'équation de Poisson sur les particules est complexe et ainsi l'incompressibilité difficile à imposer. C'est pourquoi, comme nous le verrons dans la partie suivante, la formulation vorticité représente une alternative très intéressante aux méthodes SPH traditionnelles, pour la simulation d'un fluide incompressible par particules.

4.5 Équations de Navier-Stokes en formulation vorticité et particules Lagrangiennes

Intérêt

Comme nous l'avons précisé précédemment, un des intérêts de la formulation des équations de Navier-Stokes en vorticité est la localisation de celle-ci dans l'espace. La représentation Lagrangienne s'adapte tout particulièrement à cette disparité spatiale et temporelle. De plus, les équations 1.14 en 3D et 1.19 en 2D se prêtent naturellement à être formulées de manière Lagrangienne. C'est pourquoi la méthode de résolution des équations à l'aide de particules de vorticité est très intéressante en comparaison des méthodes Euleriennes traditionnelles.

Le livre [CK00] couvre la résolution numérique des méthodes vortex, le lecteur pourra s'y référer afin de trouver plus de détails ainsi que les preuves de convergence des méthodes résumées ici.

Discrétisation du champ de vorticité

Le champ de vorticité initial $\omega_0(\mathbf{x}) = \omega(\mathbf{x}, 0)$ est échantillonné par un ensemble de particules notées par l'indice p distribuées dans l'espace. La vorticité notée $\tilde{\omega}_0(\mathbf{x})$

en chaque point de l'espace est une approximation du champ de vorticité initial :

$$\omega_0(\mathbf{x}) \simeq \tilde{\omega}_0(\mathbf{x}) = \sum_p \omega_p \delta\left(\mathbf{x} - \mathbf{x}_p\right)$$

où α_p est la vorticité portée par la particule p et δ est la masse de Dirac. Numériquement, il est nécessaire d'utiliser une représentation lisse du champ de vorticité, c'est pourquoi la masse de Dirac est remplacée par une fonction *cutoff* ζ_ε régularisation de la masse de Dirac telle que $\int_{-\infty}^{\infty} \zeta_\varepsilon(x)\, dx = 1$. On peut par exemple utiliser une exponentielle :

$$\zeta_\varepsilon(x) = \frac{1}{\varepsilon\sqrt{\pi}} e^{-\left(\frac{x}{\varepsilon}\right)^2}.$$

Le champ de vorticité au fur et à mesure du temps est alors approximé par :

$$\omega(\mathbf{x}) \simeq \tilde{\omega}(\mathbf{x}) = \sum_p \omega_p\, \zeta_\varepsilon\left(\mathbf{x} - \mathbf{x}_p\right). \tag{1.26}$$

Le choix du paramètre ε doit être fait de telle manière que le support des particules se chevauche.

Les particules sont transportées par le fluides, ainsi leur vitesse est :

$$\frac{d\,\mathbf{x}_p}{dt} = \mathbf{u}_p = \mathbf{u}(\mathbf{x}_p). \tag{1.27}$$

L'équation de vorticité en 3D devient donc dans sa forme lagrangienne :

$$\frac{D\,\omega_p}{Dt} = (\omega_p \cdot \nabla)\mathbf{u}_p + \nu\Delta\omega_p$$

et en 2D simplement :

$$\frac{D\,\omega_p}{Dt} = \nu\Delta\omega_p.$$

Contrainte sur le pas de temps Les méthodes Lagrangiennes ne souffrent pas de la condition CFL, comme les méthodes sur grille. On peut comprendre la condition CFL sur l'équation de transport de manière intuitive : si, physiquement, une onde se déplace à une vitesse C, cette onde doit pouvoir se propager à la même vitesse sur la grille. Un schéma d'Euler explicite en temps sur une grille ne considérera que le voisinage proche des points de discrétisation, le domaine de dépendance. Dans le cas 1D, si au temps t l'onde se trouve au point $y = x - \delta t C$, celle-ci se trouvera physiquement au temps suivant $t + \delta t$ en x. Sur une grille, il faut un certains temps (nombre d'itérations) pour que l'onde soit transportée de $y = x_i$ jusqu'à $x = x_{i+k}$ (où x_i et x_{i+k} représentent les points du maillage, séparés par $k > 0$ points de grille). Numériquement, la discrétisation de l'espace doit donc être fonction de la vitesse du transport, ou réciproquement le pas de temps adapté. Cette contrainte n'apparait pas pour un traitement implicite de l'équation, où tous les points du domaine influent sur la solution en un point de grille.

Dans un point de vue Lagrangien, cette équation est résolue implicitement par le transport de la particule qui suit la caractéristique du champ de vitesse $\mathbf{u}$. Malgré cela, la déformation du « maillage »induit tout de même une contrainte sur le pas de temps. Effectivement, de manière intuitive, on peut s'apercevoir que deux particules ne peuvent se croiser, ce qui limite tout de même le pas de temps en fonction de la régularité du champ de vitesse.

En 1D, deux particules situées en y_1 et y_2 à l'instant t, telles que $y_1(t) < y_2(t)$ doivent conserver leur ordre après advection. Ainsi : $y_1(t+\delta t) < y_2(t+\delta t)$, avec

$$y_i(t+\delta t) = y_i(t) + \delta t\, u(y_i(t), t) \;\text{ pour } i = \{1, 2\}$$

pour le traitement explicite de l'équation de transport par un schéma d'Euler d'ordre 1. Après simplifications, nous obtenons la contrainte suivante : $\delta t < \left|\frac{\partial_1 u}{x}\right|^{-1}$ où $\frac{\partial_1 u}{x} = (u(y_2) - u(y_1))/(y_2 - y_1)$ est l'approximation de la dérivée de la vitesse à l'ordre 1.

La méthode particulaire est donc inconditionnellement stable pour une vitesse à coefficient constant. Dans le cas d'une variation linéaire en fonction de x : $u_{i+1} = \alpha u_i$ (avec $u_i = u(y_i, t)$), on a $u_i - u_{i+1} = (1-\alpha)u_i$. Posons $\alpha > 0$ et $u_i > 0$ (le cas opposé est symétrique), nous obtenons alors deux conditions sur le pas de temps :

$$\delta t \;>\; \frac{h}{(1-\alpha)u_i} \text{ si } \alpha > 1, \text{ ce qui est toujours vrai, et} \tag{1.28}$$

$$\delta t \;<\; \frac{h}{(1-\alpha)u_i} \text{ si } \alpha \leq 1. \tag{1.29}$$

Dans le cas $\alpha = 1$, le problème est équivalent à considérer une vitesse constante. Lorsque $\alpha \to 0$ (c'est-à-dire une décroissance exponentielle de la vitesse), l'équation 1.29 tend vers la condition CFL. Ceci signifie que, dans le cas d'une variation linéaire de la vitesse en espace, la méthode particulaire est toujours meilleure, sur le critère de la stabilité, que la résolution de l'équation de transport sur grille.

De manière plus générale, dans un espace de dimension N : $\delta t < min\left\{\left|\frac{\partial_1 u_j}{x_j}\right|^{-1}\right\} < |\nabla_1 \mathbf{u}|^{-1}$ (avec j de 1 à N, et ∇_1 représente l'approximation du gradient à l'ordre 1).

Algorithme

L'algorithme utilisé dans les méthodes vortex en formulation Lagrangienne peut se résumer de la manière suivante : étant donné un champ de vorticité approximé par un ensemble de particules portant une vorticité $\omega_p(t)$ sitée en $\mathbf{x}(t)$ à l'instant t, la vorticité à l'instant suivant est calculée ainsi :

1. calcul de la vitesse $\mathbf{u}_p$ d'une particule,
2. calcul du membre de droite de l'équation de vorticité (étirement et diffusion) : $\frac{D\omega}{Dt}$,
3. déplacement de la particule en $\mathbf{x}(t+\delta t)$ et changement de sa vorticité en $\omega(t+\delta t)$.

Calcul de la vitesse

Pour résoudre l'équation sur les particules, il est nécessaire de calculer la vitesse en chaque particule. Pour cela, on utilise la loi de Biot-Savart 1.18 qui permet d'obtenir la vitesse en chaque point de l'espace en fonction de la vorticité totale, et dans notre cas spécifiquement en la position des particules :

$$\mathbf{u}(\mathbf{x}_p) = \mathbf{u}_p = \int_\Omega \mathbf{K}(\mathbf{x}_p - \mathbf{y}) \times \omega(\mathbf{y})\, d\mathbf{y},$$

qui, grâce à l'équation 1.26 se réduit à une somme sur les particules :

$$\mathbf{u}_p = \sum_q \mathbf{K}(\mathbf{x}_p - \mathbf{x}_q) \times \omega_q.$$

Comme précisé dans la partie 3.3, le noyau $\mathbf{K}$ est singulier au voisinage de 0, numériquement il est donc nécessaire d'utiliser une version régularisée $\mathbf{K}_\varepsilon$ de ce noyau et ainsi l'équation précédente devient :

$$\mathbf{u}_p = \sum_q \mathbf{K}_\varepsilon(\mathbf{x}_p - \mathbf{x}_q) \times \omega_q, \tag{1.30}$$

où on construit $\mathbf{K}_\varepsilon$, par exemple, par convolution avec une fonction *cutoff* régularisée ζ_ε :

$$\mathbf{K}_\varepsilon = \mathbf{K} \star {\zeta_\varepsilon}^d,$$

en dimension d, avec $\zeta_\varepsilon(\mathbf{x}) = \varepsilon^{-1}\zeta\left(\mathbf{x}/\varepsilon\right)$ telle que $\int_{-\infty}^{+\infty} \zeta_\varepsilon(x)\, dx = 1$. En pratique, on prend le paramètre de régularisation ε de l'ordre de la distance séparant les particules. Je donnerai dans le chapitre 4 un exemple de noyau utilisé lors de ma thèse.

Il est important de noter qu'en présence d'obstacles, la condition au bord de non pénétration de la vitesse implique l'ajout d'un potentiel présent sur la surface de l'obstacle. Ce potentiel agit comme une couche de vorticité qui vient annuler la vitesse normale du fluide sur l'obstacle. Je reviendrai sur ce principe dans une des sous-parties ci-après.

Coût de l'algorithme L'algorithme pour le calcul de la vitesse possède un coût quadratique car pour chaque particule il faut effectuer une somme sur l'ensemble des particules existantes.

Étirement

Afin de calculer le terme d'étirement $(\omega \cdot \nabla)\mathbf{u}$ présent dans l'équation de vorticité en 3D, il est nécessaire de calculer le gradient de la vitesse. Pour ce faire, on peut choisir de dériver le noyau de Biot-Savart ou bien le noyau de régularisation des particules, d'après le produit de convolution de l'équation de Biot-Savart. L'utilisation d'un schéma qui permette de conserver la circulation totale est important, par

exemple, comme proposé dans [CK00]. Sous sa forme conservative, l'étirement est $\nabla\cdot(\omega\otimes\mathbf{u}) = (\omega\cdot\nabla)\mathbf{u}+\mathbf{u}(\nabla\cdot\omega)$. Sur les particules, le schéma suivant est conservatif :

$$\frac{D\,\omega_p}{Dt} = \sum_{j,q}(\mathbf{u}_q\omega_p^j + \mathbf{u}_p\omega_q^j)\frac{(\mathbf{x}_p^j - \mathbf{x}_q^j)}{|\mathbf{x}_p - \mathbf{x}_q|}\zeta_\varepsilon'(|\mathbf{x}_p - \mathbf{x}_q|),$$

où l'exposant j désigne la j-ième composante du vecteur, et sous l'hypothèse que ζ est symétrique.

Diffusion

En présence d'un fluide non parfait, il est nécessaire de résoudre l'équation de diffusion de la vorticité sur les particules :

$$\frac{D\,\omega}{Dt} = \nu\Delta\omega.$$

Il existe plusieurs méthodes pour ce faire, nous pouvons noter deux méthodes originales et basées sur différents points de vue : la méthode dite *random walk* et celle nommée *Particle Strength Exchange* (PSE) (*cf.* [DMG89]).

La première méthode consiste à considérer la diffusion comme résultante d'un mouvement Brownien au niveau des particules du fluide. Sur ce principe, un algorithme de type fractionnaire est utilisé pour tout d'abord advecter les particules, puis modifier la position de chaque particule selon une loi statistique adaptée, en fonction de la viscosité du fluide.

La deuxième méthode quant à elle consiste à considérer la diffusion comme un échange de vorticité entre les particules. Il est possible de réécrire l'opérateur de diffusion en fonction d'un noyau η adapté et régularisé :

$$\Delta_\varepsilon\,\omega(\mathbf{x}) = \varepsilon^{-2}\int[\omega(\mathbf{y}) - \omega(\mathbf{x})]\eta_\varepsilon(\mathbf{y}-\mathbf{x})d\mathbf{y},$$

qui peut se traduire au niveau d'une particule à :

$$\Delta_\varepsilon\omega_p = \varepsilon^{-2}\sum_q[\omega_q - \omega_p]\eta_\varepsilon(\mathbf{x}_q - \mathbf{x}_p). \tag{1.31}$$

Ainsi la diffusion peut être aisément évaluée en chaque particule dans une manière similaire au calcul de la vitesse et de l'étirement. Le lecteur trouvera dans [PWS+02, CK00, Pon06] (et dans leurs références) plus de détails sur la méthode PSE, y sont fournis des exemples de noyaux pour la diffusion, sur les particules ou dans le cas de méthodes hybrides, comme je l'expose dans la partie 4.6.

Conditions limites sur les tourbillons

Comme je l'ai évoqué précédemment, les conditions limites en formulation vortex sont moins directes qu'en formulation vitesse. Par exemple, il n'existe pas de condition directe sur la vorticité ou la fonction courant pour imposer une vitesse nulle

aux bords d'un obstacle. Très naturellement, les méthodes vortex par particules sont adaptées aux écoulements dans un milieu non borné (par exemple pour un gaz ou l'intérieur d'un liquide) où les conditions sur la vitesse sont connues à l'infini.

Pour des conditions périodiques, l'utilisation de cette méthode ne présente pas de difficulté particulière car il suffit simplement de bien considérer la position $\mathbf{x}_q$ des particules dans l'utilisation des noyaux présentés précédemment, en prenant bien garde que celles-ci se répètent dans toutes les directions d'une boîte périodique. Par contre le coût algorithmique en devient plus élevé car il faut considérer une infinité de particules dans toutes les directions. Néanmoins, il est possible de limiter le nombre de particules répétées en fonction de leur distance à la boîte initiale et d'une certaine erreur acceptable.

Traîter des conditions de type *no-slip* sur la vitesse aux bords d'un domaine est bien plus complexe en formulation vortex particules qu'en formulation vitesse. En effet, il n'existe pas de lien direct entre les conditions aux bords sur la vitesse et celles sur la vorticité. Par contre, l'annulation de la vitesse aux bords peut être obtenue par l'introduction de vorticité à l'intérieur du domaine — vorticité qui viendra annuler la vitesse de glissement — conjointement avec de bonnes conditions pour ω. Une méthode communément utilisée pour réaliser cette contrainte est la méthode dite des *panneaux*. Celle-ci consiste à discrétiser les bords du domaine en une juxtaposition de panneaux, lesquels vont générer une entrée de vorticité dans leur entourage (dans l'épaisseur de la couche limite, *cf.* partie 2.4) afin de réduire la vitesse aux bords dans le but de l'annuler. La figure 1.6 présente une représentation graphique de l'idée.

La vorticité générée est alors simplement ajoutée à la vorticité portée par les particules, la difficulté étant de calculer l'intensité et l'orientation de ce flux de vorticité. Cette solution est plus complexe en 3D qu'en 2D car la vorticité possède alors 3 composantes avec des conditions aux bords plus complexes sur ω elle même. Nous verrons dans la partie 5 une nouvelle méthode, que j'ai développé pendant ma thèse, permettant d'imposer la contrainte de non glissement sur la vitesse.

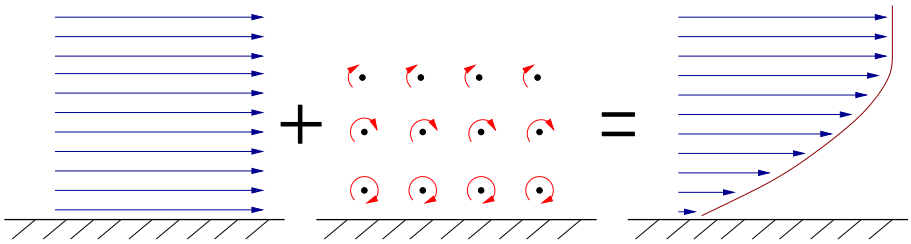


FIG. 1.6 – Illustration de la méthode de flux de vorticité qui permet d'annuler la vitesse aux bords. A gauche la vitesse avec frotements, au centre la vitesse générée par la vorticité ajoutée à l'intérieur du domaine et à droite le champ de vitesse résultant.

Difficultés et palliatifs

La contrainte majeure des méthodes vortex par particules est le coût du calcul de la vitesse en chaque particule. En effet, l'algorithme a une complexité quadratique (en $O(N^2)$) en le nombre de particules ; coût prohibitif lorsque ce dernier dépasse le millier, en 3D ce nombre est rapidement atteint en présence d'un écoulement moyennement turbulent. Cependant, des méthodes ont été développées afin de réduire les temps de calcul. La méthode de « Fast Multipoles »(FMM) appliquée aux méthodes vortex permet en effet d'effectuer le calcul des vitesses en $O(NlogN)$ ou même $O(N)$ par le regroupement hiérarchique des particules de vorticité en « paquêts »interagissant les uns avec les autres. À petite distance, les interactions sont calculées grâce au noyau de Biot-Savart ; à longue distance, une simplification de ce noyau permet d'obtenir une approximation de l'influence d'un paquêt de particules. L'erreur induite de cette approximation est connue et ainsi il est possible de décider à quelle distance cette dernière est utilisée. La FMM permet ainsi de réduire grandement le coût de calcul des vitesses sur les particules mais l'implémentation de cette méthode est assez complexe (ceci étant dû au découpage hiérarchique de l'espace) et son utilisation dans le cas général en présence d'interactions fluide/structure la complexifie encore plus.

Une autre difficulté rencontrée dans les méthodes vortex par particules est la dispersion de ces dernières au cours du temps. Les particules initialement (« bien ») distribuées sont advectées par le fluide, et, par la nature de ce dernier, ne sont pas garanties de rester convenablement réparties. Convenablement signifie ici que lorsque des particules s'éloignent d'une distance beaucoup plus grande que l'épaisseur ε du noyau, le fluide n'est plus bien représenté dans l'intervale entre celles-ci. Egalement, lorsque des particules se retrouve très rapprochées, l'épaisseur du noyau devient trop grande en comparaison de la distance pour fournir une bonne approximation de la vorticité dans cette région du fluide. Pour pallier ce problème, les méthodes dites de remaillage (*remeshing* en anglais) permettent de conserver une distribution spatiale des particules temporellement cohérente. Je détaillerai cette méthode de remaillage dans la partie 4.6.

La prise en compte de conditions aux bords de type no-slip pour la vitesse est ardue à mettre en oeuvre dans une formulation vorticité-particules. Cette difficulté fait encore partie aujourd'hui des problèmes étudiés dans les méthodes vortex, je proposerai dans mes travaux une solution élégante et intuitive permettant d'obtenir des conditions aux bords sur des surfaces immergées, méthode qui peut être appliquée naturellement pour les bords dans un domaine fermé.

Pour contrecarrer les limitations et difficultés des méthodes vortex par particules de vorticité, la méthode dite de **Vortex In Cell** (VIC) est fréquemment utilisée, c'est sur cette dernière que mes travaux de thèse sont basés. Je présente la méthode VIC dans la partie suivante.

4.6 Équation de Navier-Stokes en formulation vorticité : méthode hybride particules/grille

Principe

Dans l'objectif de pallier les difficultés induites par l'utilisation de particules dans la résolution des équations de Navier-Stokes en formulation vorticité, la méthode des Vortex In Cell (VIC) (pour « vortex en cellule ») propose une alternative très intéressante tout aussi bien en temps de calcul qu'en gain de précision dans la solution recherchée. Cette méthode consiste en l'utilisation conjointe de particules et d'une grille eulérienne en bénéficiant des avatanges des deux représentations. La grille sous-jacente aux particules permet de :

- calculer les vitesses des particules de manière rapide par l'utilisation de solveurs efficaces,
- conserver une répartition uniforme des particules de vorticité par une méthode de remaillage,
- et éventuellement de simplifier le calcul des termes de diffusion et d'étirement.

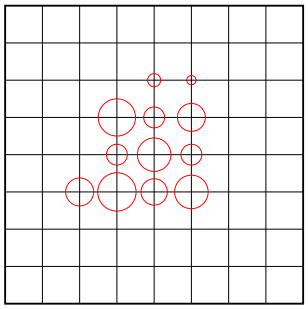
Sur ce principe, un algorithme simple est utilisé pour résoudre l'équation de conservation de vorticité à chaque pas de temps.

Algorithme

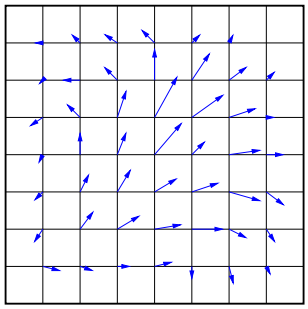
L'algorithme basé sur la méthode VIC permet de calculer l'évolution de la vorticité en chaque particule tout en conservant une distribution convenable des particules. Je propose au lecteur de suivre en parallèle de chaque étape le schéma 1.7 fournissant un support visuel à la compréhension de l'algorithme. Soit, à un instant t, un ensemble de particules situées aux positions $\mathbf{x}_p(t) = \mathbf{x}_p^n$, portant une vorticité $\omega(\mathbf{x}_p, t) = \omega_p^n$. Pour calculer la vorticité $\omega(\mathbf{x}_p^{n+1}, t + \delta t) = \omega_p^{n+1}$. au pas de temps suivant, une itération de l'algorithme est la suivante :

1. Calcul de la vitesse $\mathbf{u}^n$ sur la grille à partir de ω^n.
2. Calcul de l'étirement sur la grille (en fonction de ω^n et $\mathbf{u}^n$).
3. Interpolation de la vitesse et de l'étirement sur les particules.
4. Advection des particules et intégration du terme d'étirement $\rightarrow \mathbf{x}_p^{n+1}$ et $\omega_p^{n+1/2}$.
5. Remaillage : création de nouvelles particules basées sur la grille.
6. Diffusion sur la grille : $\omega^{n+1/2} \rightarrow \omega^{n+1}$ et donc ω_p^{n+1}.

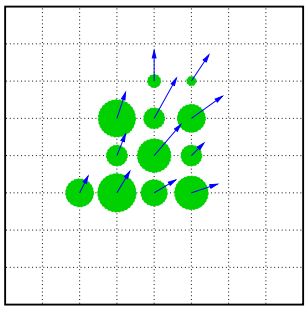
Il est important de noter qu'ici est utilisée une méthode dite de *splitting* (décomposition, ou fractionnaire en français) de l'équation de Navier-Stokes, similairement à la résolution des équations de Navier-Stokes en formulation vitesse-Eulerien par la méthode de Chorin, qui consiste ici à calculer les termes d'advection et d'étirement (étapes 2 et 4) séparément du terme de diffusion (étape 6). La notation $\omega^{n+1/2}$ correspond à la valeur intermédiaire de la vorticité à la « demi itération »de la méthode de splitting.. Le schéma 1.7 présente une illustration des différentes étapes de l'algorithme. Je détaille dans les parties suivantes les étapes principales de l'algorithme dans un ordre facilitant la compréhension.



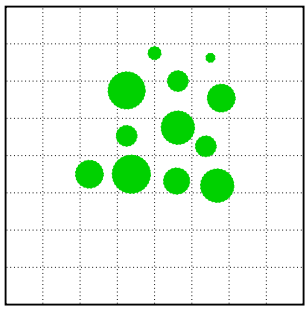
(a) Initilisation.



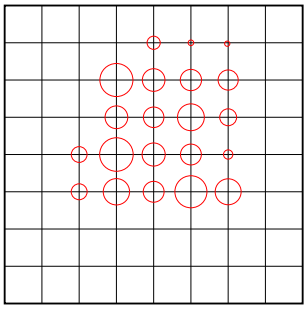
(b) Calcul de la vitesse et de l'étirement.



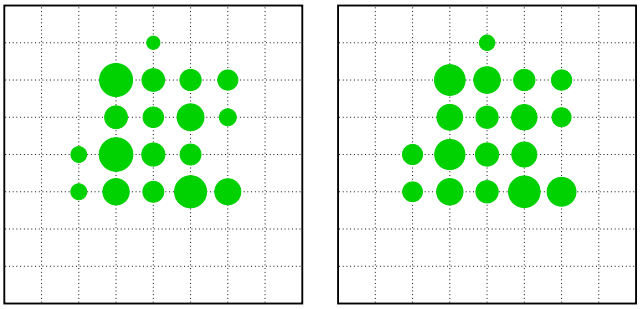
(c) Interpolation.



(d) Advection et intégration de l'étirement.



(e) Remaillage.



(f) Diffusion.

FIG. 1.7 – Algorithme utilisé pour la méthode des Vortex-In-Cells, coupe 2D. Les cercles rouges représentent la vorticité sur la grille, les ronds verts la vorticité portée par les particules. Dans la phase de remaillage, des particules ne sont créées que pour une vorticité assez grande. Il apparaît clairement l'intérêt de l'utilisation des particules avec la localisation des calculs pour l'advection.

Il est important de noter ces quelques points : il est possible d'utiliser une méthode purement particulaire de type PSE pour le calcul de la diffusion, notamment si le remaillage n'est pas effectué à chaque pas de temps. Les schémas PSE peuvent être adaptés sur une grille afin de bénéficier d'une plus grande rapidité de calcul, comme nous le verrons par la suite. Dans ce cas particulier, la méthode PSE peut être vue comme une méthode de différences finies. Un des avantages de traiter la

diffusion sur la grille est qu'il est possible de la résoudre de manière implicite, ce qui permet de ne pas contraindre le pas de temps pour des viscosités importantes.

Remaillage

Comme je l'ai expliqué dans la partie 4.2, résoudre les équations de Navier-Stokes dans une description Lagrangienne peut entrainer plusieurs désavantages, notamment la déformation du maillage transporté par le fluide peut entrainer de fortes distortions. Il en va de même pour les équations de Navier-Stokes en formulation vorticité car les particules sont transportées par le fluide. Les méthodes de remaillage permette de régulariser le maillage lorsque la déformation est trop importante. Remailler à chaque pas de temps permet de simplifier grandement les calculs en donnant la possibilité d'effectuer les calculs coûteux sur les particules en projetant la vorticité sur la grille. Ainsi, le calcul de la vitesse, de la diffusion et de l'étirement sont effectués sur la grille.

Etant donné un ensemble de N particules situées aux positions $\mathbf{x}_p$, nous souhaitons créer M nouvelles particules en de nouveaux points de l'espace $\mathbf{y}_p$ (basés sur la grille eulérienne) en conservant la vorticité globale. Pour cela, nous nous basons sur l'équation 1.26 définissant le champ de vorticité en chaque point de l'espace. Utiliser directement cette équation pour calculer la vorticité $\tilde{\omega}(\mathbf{y}_p)$ possède deux désavantages. Tout d'abord ce calcul a un coût algorithmique en $O(NM)$ car pour chaque nouvelle particule il faut effectuer N sommes. Deuxièmement, cette formulation ne permet pas de décider aisément de l'endroit où créer les nouvelles particules telles que la vorticité soit correctement conservée.

L'utilisation d'une grille sous-jacente aux particules possède deux intérêts : la répartition uniforme des nouvelles particules est assurée car elles seront basées sur les noeuds de la grille Eulerienne et il est facile de savoir à quel endroit celles-ci doivent être créées. En effet, nous avons défini les particules de vorticités comme possédant un support de taille ε (de l'ordre de h, la distance entre deux points de grille), il est ainsi possible de définir une approximation de la fonction cutoff pour localiser la vorticité autour de la position de la particule, la localisation étant de quelques points de grille autour de la position de l'ancienne particule.

Une fonction d'interpolation à support limité est donc utilisée pour « distribuer »la totalité de la vorticité d'une particule sur un ensemble de nouvelles particules. Le choix de la fonction d'interpolation doit être rigoureusement effectué dans le but de conserver les propriétés dynamiques du fluide. Il existe plusieurs fonctions d'interpolations utilisées dans les méthodes vortex, pour ne pas surcharger le manuscrit, je les présenterai dans le chapitre 4 dans lequel je détaille le schéma principal que j'ai utilisé au cours de ma thèse.

Diffusion

Après l'étape de remaillage, de nouvelles particules ont été créées basées sur les noeuds d'une grille eulérienne. L'étape suivante de diffusion peut maintenant

être simplifiée (en comparaison de la méthode sur les particules) via l'utilisation de méthodes existantes de résolution d'équations de diffusion ou bien par une approximation de l'équation 1.31.
L'opération de splitting de l'équation de vorticité consiste à résoudre séparément la diffusion du reste de la partie droite de l'équation de vorticité qui est présentée dans les prochaines parties. La partie diffusion de la méthode de splitting se résume donc à résoudre :

$$\frac{\partial \omega}{\partial t} = \nu \Delta \omega,$$

sur la grille ou sur les particules. Sur la grille, il est possible de résoudre cette équation de manière explicite :

$$\omega^{n+1} = \omega^{n+1/2} + \delta t \nu \Delta \omega^{n+1/2}$$

(où rappelons que $\omega^{n+1/2}$ est la vorticité issue de la première partie du splitting) ou bien implicite :

$$\omega^{n+1} = \omega^{n+1/2} + \delta t \nu \Delta \omega^{n+1}$$

de manière similaire à ce qui a été présenté dans la discussion sur la diffusion de la vitesse.

Il est aussi envisageable d'appliquer une méthode de type PSE en tirant avantage de la présence de la vorticité sur la grille. Comme les particules sont réparties uniformément sur celle-ci, il est possible d'accélérer le calcul des méthodes PSE (de coût algorithmique $O(N^2)$) en utilisant un noyau η_ε adapté. En effet, le support du noyau étant limité en fonction de l'épaisseur ε, une approximation permet de ne diffuser la vorticité que dans un proche voisinage, de l'ordre de ε, autour d'une particule. Par exemple, on peut considérer le voisinage direct — c'est-à-dire à une distance de h — d'une particule, soit 26 particules en 3D, et appliquer ainsi l'opérateur de diffusion en chaque particule pour un coût minimal en $O(N)$.
Il est important de noter que la première (équation de la chaleur en explicite) et troisième méthode (PSE) sont contraints par une condition de stabilité sur pas de temps : $\delta t \leq h^2/\nu$, alors que le traitement implicite de la diffusion ne possède pas cette contrainte. Malheureusement, le coût de résolution de l'équation implicite est beaucoup plus élevé qu'en explicite car la méthode revient à inverser une matrice (éparse) de taille M^2 où M est le nombre de points de grille au total. C'est pourquoi pour des valeur de viscosités ν faibles, l'utilisation de la méthode PSE est plus performante que les deux autres méthodes car elle est moins coûteuse que la méthode explicite sur la grille, particulièrement lorsque le nombre de particules est faible en comparaison du nombre de points de grille.

Calcul de la vitesse

Sur la grille, le calcul de la vitesse peut être effectué rapidement à l'aide de la fonction courant, en évitant d'utiliser un algorithme de coût quadratique sur les particules. Il suffit pour ce faire de résoudre l'équation de Poisson 1.16 sur la fonction courant ψ. Un solveur de type *FishPack* permet de résoudre rapidement

cette équation avec des conditions aux bords adaptées aux conditions limites désirées pour le fluide (je reviendrai sur ce problème dans une des sous parties suivantes). Une fois la fonction courant obtenue, il convient de calculer la vitesse $\mathbf{u}$ grâce à l'équation 1.15 et à l'aide de schémas de différences finies sur la grille. Ainsi la vitesse est connue en chaque point de discrétisation de l'espace occupé par le fluide, ce qui nous permet pour la suite de calculer l'étirement et d'attribuer à chaque particule de vorticité la vitesse du point de grille sur lequel elle « repose »à la suite du remaillage.

Le lecteur pourra soulever le fait que par cette méthode, la vitesse est calculée en tous les points du maillage, ce qui n'est pas nécessaire dans une formulation vorticité-particules. Deux éléments importants permettent de répondre à cette objection :

- le coût de la résolution numérique des équation 1.16 et 1.15 est beaucoup plus faible sur une grille ($O(NlogN)$) que la résolution de l'équation de Biot-Savart sur les particules ($O(N^2)$) lorsque le nombre de particules devient important dans le domaine de calcul, ce qui est fréquent lors de la simulation d'écoulements turbulents,
- nous verrons par la suite que la vitesse dans tout le domaine est très souvent nécessaire dans les applications qui nous intéressent (interactions fluides/gaz/solides) car des équations de transport — de densité et d'interface par exemple, cf chapitre 2 — complémentent l'équation sur la vorticité, la localisation des grandeurs à transporter n'étant pas nécessairement la même.

C'est pour ces raisons que la méthode VIC est aujourd'hui fréquemment adoptée dans la résolution des équations de Navier-Stokes en formulation vorticité.

Étirement

Nous pouvons aussi profiter de l'avantage de la grille Eulerienne sous-jacente aux particules pour calculer l'étirement de manière efficace. Pour ce faire, nous optons pour un schéma sur la grille dans lequel nous considérons l'étirement sous sa forme conservative :

$$\nabla \cdot (\omega \otimes \mathbf{u}) = (\omega \cdot \nabla)\mathbf{u} + \mathbf{u}\nabla \cdot \omega$$

Il est important d'utiliser un schéma conservatif car en pratique la solénoïdalité de la vorticité n'est pas assurée numériquement. Ainsi le terme d'étirement est calculé simplement par différences finies sur la grille, ce qui réduit d'autant plus le coût du calcul.

Conditions aux bords

Comme je l'ai présenté dans la partie concernant la méthode vortex par particule (cf. 4.5), les conditions aux bords de type *no-slip* pour la vorticité sont plus complexes que dans une formulation vitesse et nécessite d'être traitées avec grand soin. La méthode des Vortex-In-Cells permet de simplifier la prise en compte de ces conditions aux bords pour le calcul de la vitesse. En effet, il est possible de traiter très naturellement la périodicité dans une grille Eulerienne via un solveur adapté

(par transformée de Fourier par exemple) pour résoudre l'équation de Poisson sur ψ. Concernant les conditions aux bords de type no-slip, la tâche est plus ardue : il n'est pas possible d'imposer directement cette condition via la vorticité et la fonction courant ; l'utilisation d'algorithmes de type flux de vorticité est à envisager dans ce cas.

Cependant, il est aisé d'imposer des conditions de glissement et non pénétration de type *no-through* par simple dérivation de la fonction courant. Plaçons nous dans le cas d'une paroie statique en $z = 0$ et dans un domaine où le fluide évolue dans le demi espace $\mathbf{x} \in \{(x, y, z) \mid z > 0\}$. Effectivement, pour obtenir une vitesse normale nulle au bord

$$\mathbf{u} \cdot \mathbf{n}|_{\partial\Omega} = 0$$

c'est-à-dire $u_3 = 0$, il suffit de reprendre la définition de la vitesse grâce à la fonction courant : $\mathbf{u} = \nabla \times \psi$ et d'en extraire la troisième composante :

$$u_3 = \frac{\partial \psi_2}{\partial x} - \frac{\partial \psi_1}{\partial y} = 0.$$

Ainsi, une solution simple consiste à prendre une valeur constante pour les premières et deuxièmes composantes de la fonction courant sur le bord $z = 0$, l'équation précédente est satisfaite. Nous pouvons prendre les conditions de Dirichlet homogènes $\psi_1 = \psi_2 = 0$ car la fonction courant peut être définie à une constante près, pour la raison que nous en prenons le rotationnel afin d'obtenir $\mathbf{u}$. Il est nécessaire de fournir la condition au bord pour la troisième composante de la fonction courant, orthogonale à la paroie. Je rappelle ici que la fonction courant doit être solénoïdale : $\nabla \cdot \psi = \frac{\partial \psi_1}{\partial x} + \frac{\partial \psi_2}{\partial y} + \frac{\partial \psi_3}{\partial z} = 0$, nous pouvons donc en déduire qu'avec les conditions précédemment imposées, la variation de la composante tangeante de ψ sur la paroie doit être nulle, *ie.* $\frac{\partial \psi_3}{\partial z} = 0$ qui nous fournit une condition de Neumann homogène complétant ainsi le système.

Limitations

Nous venons de voir que la méthode hybride des VIC avec remaillage permet un gain sur le temps de calcul non négligeable pour le calcul de la vitesse, entre autre. Cependant, en comparaison des méthodes en formulation où la projection de la vitesse pour assurer l'incompressibilité demande la résolution d'une unique équation de Poisson, en 3D cette méthode vortex demande la résolution de trois équations de Poisson pour la fonction courant. D'un autre côté, elle permet de traiter de manière simple l'équation de transport de la vorticité par la méthode particulaire, et, du fait de la localisation des tourbillons, les calculs sont focalisés là où se trouve la vorticité. La résolution des équations en formulation vitesse nécessite quant à elle des calculs en chaque point de grille, pouvant ainsi accumuler des erreurs numériques aux endroits où la méthode vortex n'en commettra pas du fait de l'absence de particules.

Les conditions aux bords sont, en méthode vortex, un problème complexe encore largement étudié. Nous verrons dans le chapitre 5 la méthode que je propose dans le but de palier ces difficultés pour la prise en compte des interactions entre un fluide et des obstacles fixes ou soumis aux forces hydrodynamiques.

5 Méthodes pour la simulation de fluides incompressibles en synthèse d'images

Je présente dans cette partie quelques exemples choisis de méthodes pour la simulation de fluides dans le cadre de la synthèse d'images. Cet aperçu des méthodes existantes n'est pas destinée à être exhaustif, pour une bibliographie plus complète, le lecteur pourra consulter la thèse [Gén06] proposant un état de l'art plus complet des méthodes développées. J'ai choisi les articles suivants pour leur approche voisine de celle que j'ai adoptée dans ma thèse, une méthode hybride Eulerienne-Lagrangienne par particules de vorticité.

5.1 Formulation vitesse

Stable Fluids : un méthode rapide et stable

En 1999, Jos Stam [Sta99] est le premier à proposer une méthode rapide et stable pour le calcul 2D et 3D des équations de Navier-Stokes. La technique est basée sur une implémentation rapide de la projection de la vitesse du fluide, par l'implémentation d'un algorithme de transformée de Fourier rapide. La stabilité de la résolution des équations est assurée par la méthode des caractéristiques (ou semi-Lagrangienne) pour le calcul du terme non linéaire de l'équation de conservation de mouvement. Ainsi, les résultats obtenus dans cet article ont été les premiers à offrir un réalisme important pour un temps de calcul interactif, pour des discrétisations pouvant atteindre 30^3 points de grille. La figure 1.8 montre des images de simulations de fumée proposées par l'auteur. La méthode des caractéristiques est aujourd'hui acceptée, en synthèse d'images, comme une avancée majeure dans le domaine de l'animation de fluide ; elle est toujours utilisée dans les articles actuels. Par contre, l'interpolation du champ de vitesse lors de la phase semi-Lagrangienne induit un lissage du champ de vitesse. En comparaison, une méthode purement particulaire ne souffre pas de ces erreurs réduisant l'aspect turbulent d'un écoulement car l'équation de transport y est résolue directement sur les particules.

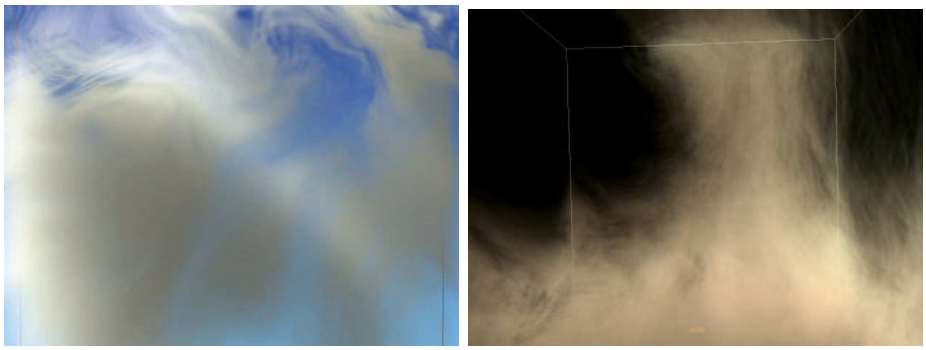


FIG. 1.8 – Simulation stable et efficace de fumée par la méthode *Stable Fluids* en 1999. Images extraites de [Sta99].

Particules de vorticité

Dans l'article [SRF05], Selle *et al.* proposent d'utiliser une méthode hybride où, en plus du calcul des vitesses sur une grille, sont ajoutées des particules de vorticité interagissant. Les auteurs argumentent que la méthode semi-Lagrangienne pour le calcul de la convection induit une perte de la turbulence aux petites échelles. Ils proposent ainsi de « récupérer »ces perturbations via l'introduction de particules de vorticité localisées, comme une force extérieure à l'équation de conservation du mouvement. La force ajoutée à la vitesse est $\mathbf{f} = \varepsilon h \mathbf{N} \times \omega$ où ε est un paramètre de contrôle de l'intensité du vortex, h le pas d'espace, $\mathbf{N} = \nabla|\omega|/|\nabla|\omega||$ et ω est la vorticité portée par la particule. L'équation de conservation du mouvement est alors résolue sur la grille pour la vitesse et en formulation vorticité sur les particules, puis, à chaque itération, la vorticité est reintroduite dans le fluide.

Cette solution *ad hoc* s'éloigne énormément du comportement physique réel du fluide et introduit un couplage incorrect entre vitesse et vorticité, qui n'a pas lieu d'être. De plus, la stabilité de la méthode n'est pas assurée et c'est à l'animateur de choisir des paramètres adaptés afin de minimiser les erreurs introduites par la méthode tout en lui permettant d'obtenir un résultat plus turbulent localement au niveau des particules. Les détails sur la répartition et l'initialisation de celles-ci ne sont pas donnés. Les résultats obtenus avec cette méthode montre néanmoins que les auteurs réussissent à « redynamiser »l'écoulement du fluide, ce qui est un critère important en synthèse d'images. Je montre dans la figure 1.9 des images de simulation de fumée issues de l'article [SRF05] ; leur méthode a été aussi appliquée sur la simulation de fluides à deux phases.

La méthode que j'ai adoptée dans ma thèse, basée sur une formulation purement vortex de l'équation de Navier-Stokes, est quant à elle physiquement correcte et donne des résultats aussi convaincants avec une résolution purement particulaire ou bien hybride, comme nous le verrons dans le chapitre 4.

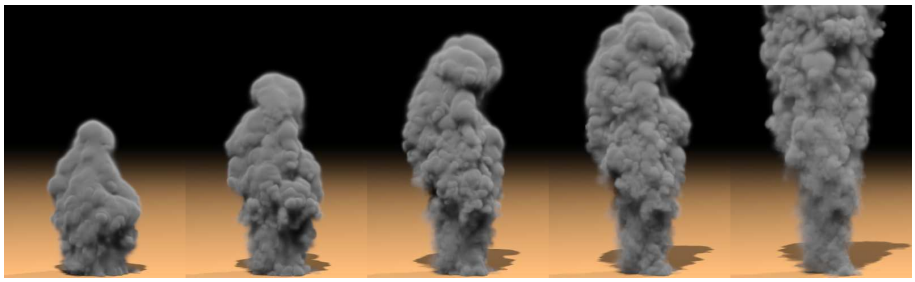


FIG. 1.9 – Évolution d'un nuage de fumée par la méthodes de Selle *et al.* . La simulation est calculée dans une boîte de taille $180 \times 260 \times 180$ avec 6000 particules de vorticité pour ajouter de la turbulence. Image tirée de [SRF05].

5.2 Méthode de type ALE et maillages dynamiques

Dans leurs articles [FOC+05, KFCO06], Klingner, Feldman *et al.* adoptent le choix d'une méthode de type ALE pour la simulation d'interactions entre fluides et solides rigides et déformables. L'espace occupé par le fluide est maillé en 3D par des tétraèdres, en profitant de mailleurs 3D rapides (*cf.* [ACSYD05]) qu'ils utilisent à chaque pas de temps afin de créer un nouveau maillage en évitant les problèmes pouvant apparaître en utilisant une structure statique — mais dont les points bougent — pouvant entrainer des incohérences et une mauvaise discrétisation lors de grandes déformations. Le maillage est adapté, ici raffiné, en fonction de certains critères : distance aux différentes surfaces (liquide-gaz, fluide-solide) où sont nécessaires une plus grande précision ou la turbulence par exemple.

Les opérateurs discrets sont définis à l'intérieur des cellules tétraédriques (se référer à l'article [KFCO06] pour plus de détails) et la convection est traîtée par une méthode de type semi-Lagrangienne (*cf.* 4.3), mais d'une manière particulière : les vitesses ne sont pas interpolées sur les noeuds du maillage entre un pas de temps et le suivant mais sont directement obtenues via la méthode semi-Lagrangienne. Soient M^n et M^{n+1} lest maillages au temps $n\delta t$ et $(n+1)\delta t$ respectivement, sur le maillage M^n sont stockées les vitesses $\mathbf{u}^n$. L'idée première consisterait à interpoler les valeurs $\mathbf{u}^n$ sur le maillage M^{n+1} et de résoudre l'équation de Navier-Stokes à partir de cette solution. Les auteurs utilisent plutôt la phase de traîtement du terme convectif afin d'éviter d'effectuer cette opération, ainsi l'étape semi-Lagrangienne est effectuée directement sur le maillage précédent à partir des positions $\mathbf{x}^{n+1}$ des noeuds du maillage au temps $(n+1)\delta t$; la figure 1.10 montre une représentation graphique de la méthode.

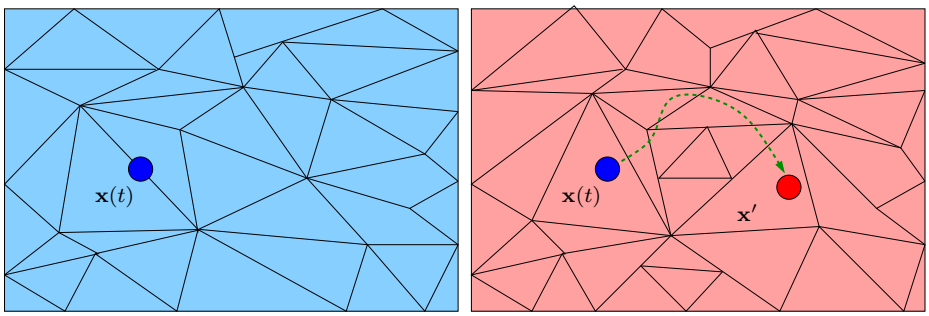


FIG. 1.10 – Maillage dynamique et traitement de la convection : la vitesse au niveau du point $\mathbf{x}$ au temps t dans le nouveau maillage (figure de gauche) est la vitesse située au point $\mathbf{x}'$ interpolée dans le maillage précédent (figure de droite). La méthode des caractéristiques est adoptée pour obtenir $\mathbf{x}' = \mathbf{s}(\mathbf{x}(t), t)$.

Les auteurs soutiennent que cette méthode n'introduit pas de lissage supplémentaire du fait qu'ils ne projètent pas la vitesse sur le maillage au pas temps suivant. Néanmoins, le fait d'interpoler la vitesse à l'intérieur du maillage précédent — du fait de la non concordance des sommets du maillage — introduit indubitablement un lissage supplémentaire à la phase semi-Lagrangienne.

La difficulté principale de cette formulation de type ALE basée sur des tétraèdres est d'obtenir et conserver une discrétisation correcte du domaine de simulation, problème complexe, notamment lors de la présence de paroies en interaction avec le fluide. La tâche est d'autant plus ardue que la forme des bords est arbitraire si on veut intégrer par exemple des solides de forme quelconque. Afin de coller au mieux à ceux-ci, les auteurs projètent les noeuds du maillage du fluide sur les bords, ce qui peut entrainer des problèmes de discrétisation : trous ou chevauchement des éléments ; ce souci peut alors rendre incorrecte la résolution des équations. De plus, le fait que le maillage soit généré de manière dynamique à chaque pas de temps rend difficile toute étude précise des schémas numériques, point non adressé par les auteurs. Néanmoins, les résultats obtenus par cette méthode sont convaincants, principalement du fait de l'adaptativité du maillage plus précis aux endroits de fortes turbulences, mais que le coût de calcul nécessaire est important, de l'ordre de la minute pour 100.000 éléments, soit l'équivalent d'une grille régulière de taille 47^3 ; nous verrons aux chapitres 4 et 5 que par une méthode hybride de *Vortex In Cell*, nous pouvons obtenir des temps de calcul beaucoup plus attrayants pour une qualité tout aussi intéressante.

Je reviendrai sur cette méthode dans le troisième chapitre car dans ces articles est aussi proposée une technique pour l'interaction du fluide avec des obstacles solides ou déformables.

5.3 Formulation vitesse par méthode particulaire

L'animation de fluides en synthèse d'images s'est toujours beaucoup intéressée à la représentation particulaire du fluide. Effectivement, cette représentation, où les particules peuvent être considérées comme des bouts de matière, apparaît très naturelle et simple à manipuler que lorsque les quantités sont discrétisées sur une grille. De plus, comme les particules peuvent être aisément concentrées spatialement aux endroits de grande importance : visuellement au niveau d'une interface entre liquide et gaz par exemple. Même si numériquement cette non uniformité de la discrétisation peut poser des problèmes, elle est un atout pour l'animation de par son intuitivité.

Mais, comme nous le verrons, cette représentation ne permet pas de résoudre exactement la contrainte d'incompressibilité qui est une des propriétés les plus importantes des fluides usuels. Dans ces formulations basées sur la vitesse, il est important de discrétiser la totalité du domaine occupé par le(s) fluides — même si en pratique, dans le cadre de fluides à deux phases, seulement les liquides sont discrétisés — et, malgré la possibilité d'adaptativité spatiale des particules (*cf.* [APKG07], par exemple), ceci peut rendre le coût des calculs important. En comparaison, les méthodes vortex par particules permettent d'implicitement résoudre la contrainte d'incompressibilité tout en focalisant la présence des points de discrétisation là où il est le plus important : à l'interface entre deux fluides et aux endroits de forte turbulence.

Smoothed Particle Hydrodynamics

Dans la famille des méthodes de résolution des équations de Navier-Stokes par particule, les *Smoothed Particle Hydrodynamics* (SPH) ont toujours été très attrayantes en synthèse d'images. En effet, elles permettent de simuler simplement des milieux très déformables, tels que les liquides, tout en proposant une représentation par particules faciles à manipuler, sans se contraindre à une discrétisation régulière de l'espace par une grille. Cette méthode a permis d'obtenir des animations réalistes de fluides en interaction avec divers objets, où, la plupart du temps, sont considérées des forces de répulsion entre les paroies des solides rigides ou déformables. Ces forces s'intègrent naturellement dans des modèles traditionnels pour l'animations de solides, ce qui en fait un avantage important.

Les premiers travaux notables en SPH dans le domaine de l'animation [DG96, SAC+99] ont permis l'animation rapide de milieux visqueux et de fluides, en profitant d'une représentation implicite de la surface d'un liquide portée par les particules — je reviendrai plus en détails sur la notion d'interface entre deux phases dans le chapitre suivant.

Reprise dans les travaux récents [MCG03, MSKG05, CBP05, APKG07], la méthode de base permet de résoudre l'équation de Navier-Stokes incompressible en évaluant le gradient de pression sur les particules afin de les déplacer pour approcher la contrainte d'incompressibilité. Ceci est effectué par le biais d'une force ajoutée aux particules, en fonction de la variation locale de densité obtenue par le champ lissé de la densité $\tilde{\rho}$ décrit par les particules :

$$\tilde{\rho}(\mathbf{x}) = \sum_j m_j W(\mathbf{x} - \mathbf{x}_j, h) \tag{1.32}$$

où m_j représente la masse de la particule j et $W(\mathbf{x} - \mathbf{x}_j, h)$ représente le noyau d'interpolation de support h. Dans le cas d'un fluide incompressible, cette densité doit rester constante en chaque point du fluide. D'après [DG96, MCG03, CBP05, APKG07], la masse d'une particule étant constante, la variation de pression due à l'incompressibilité peut se calculer par :

$$\mathbf{f}^{\text{pressure}} = -\nabla p(\mathbf{x}_i) = -\sum_j m_j \frac{p_j}{\rho_j} W(\mathbf{x}_i - \mathbf{x}_j, h) \tag{1.33}$$

où la pression p en chaque particule est données par

$$p(\mathbf{x}_i) = k\left(\rho(\mathbf{x}_i) - \rho_0\right) \simeq k\left(\tilde{\rho}(\mathbf{x}_i) - \rho_0\right)$$

où ρ_0 est la densité de la particule au temps $t = 0$ et k un coefficient dépendant de la nature du gaz. Une version symétrisée de cette équation est proposée dans [MCG03]. En pratique, intégrer l'équation 1.33 dans un schéma d'Euler traditionnel pour faire avancer les particules, ne permet pas d'assurer l'incompressibilité ; une méthode de relaxation est donc appliquée afin de repositionner les particules et minimiser la compression. Ces méthodes permettent d'approximer la contrainte d'incompressibilité, mais elles sont loin d'y parvenir de manière aussi précise qu'avec les méthodes

traditionnelles sur grille, et demandent un effort de calcul supplémentaire via des algorithmes coûteux d'échange particulaire en $O(N^2)$. De plus, l'argument statuant que la masse $m_j = \rho_j v_j$ d'une particule est constante est erroné puisque la densité est le volume sont soumis aux équations de transport en fluide compressible :

$$\frac{\partial \rho}{\partial t} + (\mathbf{u} \cdot \nabla)\rho = -\rho \nabla \cdot \mathbf{u} \text{ et } \frac{\partial v}{\partial t} + (\mathbf{u} \cdot \nabla)v = -v \nabla \cdot \mathbf{u},$$

car en pratique $\nabla \cdot \mathbf{u}$ n'est pas nul, ce qui peut poser des problèmes supplémentaires numériquement.

D'autre part, l'influence de particules éloignées est négligée dans la résolution des équations. Ceci permet, par un découpage efficace de l'espace, de réduire drastiquement le coût des calculs, tout en en réduisant la qualité.

Les méthodes SPH définies dans ces articles sont des alternatives aux méthodes plus traditionnelles pour la simulation de fluide, particulièrement adaptées pour des problèmes où la rapidité, l'aspect dynamique et visuel du fluide son prépondérants, face à la précision et à la correction des équations résolues.

Particle-Based Simulation of Fluids

Dans l'article [PTB+03], Premoze *et al.* ont adapté la méthode des *Moving Particles Semi-implicit* (MPS) [KTO96], similaire aux méthodes SPH et permettant d'imposer plus directement la contrainte d'incompressibilité, dans le même principe que méthode de projection. La vitesse au pas de temps suivant est calculée par un algorithme de type fractionnaire, où tout d'abord une vitesse violant la contrainte d'incompressibilité est calculée, sur les particules :

$$\mathbf{u}^* = \mathbf{u}^n + \frac{\delta t}{\rho}(\mu \Delta \mathbf{u}^n + \rho \mathbf{g}). \tag{1.34}$$

Afin de projeter ce champ de vitesse sur un champ à divergence nulle, les auteurs résolvent l'équation

$$\langle \Delta p \rangle_i = -\frac{\rho}{\delta t} \frac{\langle n^* \rangle_i - n_0}{n_0}, \tag{1.35}$$

où n_0 est la densité de particules initiale, et

$$\langle n^* \rangle_i = \sum_{j \neq i} W(\mathbf{x}_j - \mathbf{x}_i)$$

est la densité de particules autour de la particule i après advection par la vitesse $\mathbf{u}^*$. L'équation 1.35 est issue de la contrainte d'incompressibilité écrite sous la forme :

$$\langle \nabla \cdot \mathbf{u} \rangle = \frac{1}{\delta t} \frac{\langle n^* \rangle_i - n_0}{n_0}.$$

Une fois la pression obtenue, celle-ci vient corriger la vitesse :

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\delta t}{\rho} \langle \Delta p \rangle_i.$$

Le calcul de la pression en chaque particule nécessite l'inversion d'une matrice creuse et symétrique, ressemblant à celle utilisée en différences finies pour le même problème.

La méthode utilisée est une bonne alternative aux méthodes proposées précédemment pour les formulations SPH, car elle permet d'imposer plus proprement la contrainte d'incompressibilité. Néanmoins elle souffre de deux désavantages importants : l'absence de contrôle direct sur la position des particules peut amener à une mauvaise distribution de celle-ci au cours du temps, et nécessiterait une méthode de type remaillage afin de palier ce problème. Secondement, cette formulation vitesse nécessite tout de même une importante discrétisation de l'espace afin d'approcher au mieux le champ de vitesse — même en l'absence de turbulences — et, à la différence des méthodes vortex où la vorticité est localisée, la vitesse est présente partout.

5.4 Méthodes de vortex

En synthèse d'images, la vorticité comme brique de base pour la simulation de fluides a rarement été considérée. Souvent mise de côté du fait de l'apparente complexité de la résolution des équations, principalement pour le calcul de la vitesse, cette formulation a tout de même été introduite dans le domaine par Gamito *et al.* [GLG95] dès 1995. Leur article présente une méthode pour la simulation de fumée en 2D par la résolution de l'équation de conservation de la vorticité sur une grille Eulerienne. Je m'intéresse ici plus particulièrement aux deux articles récemment publiés pour la simulation de gaz par méthodes vortex.

Filaments de vorticité

Angelidis *et al.* [AN05] proposent dans leur article d'utiliser les propriétés des filaments de vorticité afin de simuler efficacement le mouvement d'un gaz et sa visualisation par particules de fumée. Ces travaux se concentrent donc particulièrement sur la simulation d'anneaux de vorticité 1D. Physiquement, ces filaments naissent de l'aspect turbulent du fluide et sont transportés par celui-ci. Sur ces filaments, la circulation est constante et, de plus, elle est conservée au cours du temps. Cette propriété est vraie dans le cas non visqueux où la circulation n'est pas dissipée dans son entourage ; le cas des gaz turbulents étudiés en synthèse d'images permet de négliger l'aspect visqueux afin d'obtenir les animations les plus dynamiques possibles. Ainsi, sous la supposition que les filaments sont conservés au cours du temps, les auteurs proposent une technique pour calculer efficacement la vitesse en chaque point de l'espace, allant même jusqu'à adopter une description multi-échelle adaptative des filaments.

Ces primitives sont alors la base d'une méthode pratique et facilement manipulable par un animateur pour simuler en temps réel les mouvements « simples »— dans le sens que le mouvement est défini par des primitives simples, cas particuliers du problème 3D plus général — d'un fluide, tout en fournissant un résultat visuel

intéressant et très dynamique. De surcroît, la méthode se prête très bien à des conditions aux bords dans un domaine non borné, ce qui peut être intéressant dans de nombreuses applications, comme dans les jeux vidéos par exemple.

FIG. 1.11 – Exemple de simulation de fumée par filaments de vorticité, en temps réel. Images extraites de [AN05].

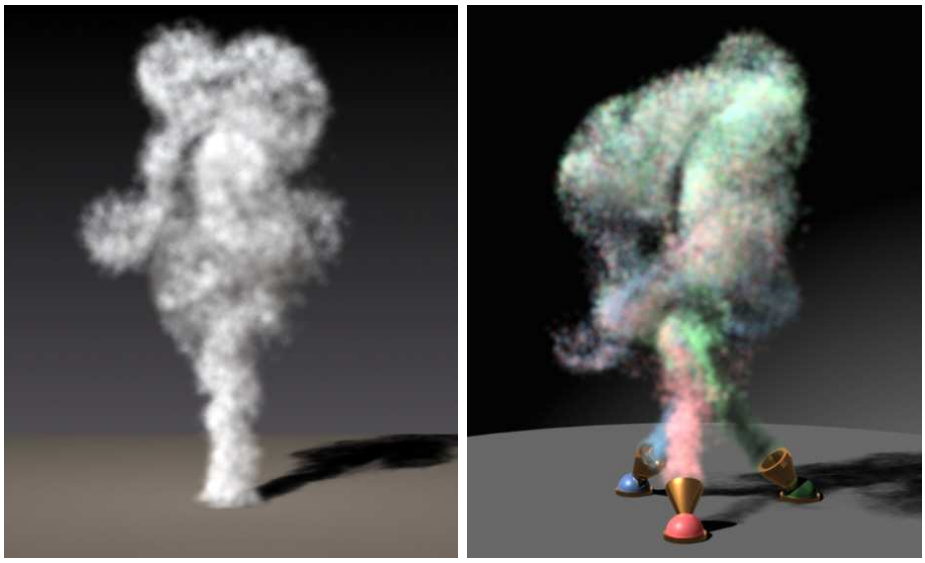


FIG. 1.12 – Exemple de simulation de fumée par particules de vorticité, en temps réel. Images extraites de [PK05].

Particules de vorticité

Dans l'article [PK05], Park *et al.* appliquent la méthode de vortex par particules aussi pour la simulation de gaz. La méthode est essentiellement telle que je l'ai expliqué en partie 4.5. Les auteurs ont utilisé une méthode de regroupement des particules afin d'accélerer les calculs, permettant ainsi d'avoir une simulation de fluide 3D efficace, même pour un nombre important de points de discrétisation.

En comparaison des méthodes SPH, cette représentation permet de résoudre implicitement la contrainte d'incompressibilité sur la vitesse tout en ne souffrant pas de la dissipation, due au traitement de la convection, présente dans les méthodes sur grille. Malheureusement, la répartition spatiale des particules reste un problème important qui peut être palié par une méthode de remaillage.

Mes travaux de thèse sont orientés autour de cette idée de résolution de l'équation de vorticité sur des particules, tout en bénéficiant des avantages fournis par l'utilisation d'une grille sous-jacente (méthode VIC) et d'une technique de remaillage afin de contrôler la bonne distribution des points de discrétisation.

6 Bilan du chapitre

Nous venons de voir dans ce chapitre les équations décrivant le mouvement d'un fluide incompressible, ainsi que les différents type de discrétisation pour la résolution numérique de celles-ci. Je me suis principalement intéressé aux méthodes basées sur des grilles Euleriennes et des particules Lagrangiennes. Ces dernières proposent une alternative attirante aux techniques sur grille car, entre autre, l'équation d'advection est résolue implicitement par le transport des particules, réduisant ainsi la complexité du problème en évitant de calculer explicitement le terme non linéaire de l'équation de Navier-Stokes. Les méthodes particulaires couplées aux techniques de type remaillage permettent de palier les problèmes usuellement rencontrés en simulation de fluide par particule : la mauvaise répartition des points de discrétisation lorsque le fluide est turbulent.

Les problèmes principaux rencontrés dans le domaine de la synthèse d'image pour la simulation de fluides sont la difficulté à maintenir un mouvement dynamique — le calcul de la convection par la méthode semi-Lagrangienne couramment usitée dissipe une partie de la vitesse importante pour les turbulences —, la complexité et le coût des équations à résoudre. Nous avons pu voir, dans les articles [AN05, PK05], que la formulation vortex offre aujourd'hui au domaine un point de vue différent et efficace pour la résolution des équations de Navier-Stokes, pour la simulation de fumée.

La formulation vortex de l'équation de Navier-Stokes possède plusieurs intérêts importants en comparaison de la formulation vitesse. Tout d'abord, la grandeur étudiée représente directement les zones turbulentes du fluide, ce qui est un critère important en synthèse d'images car c'est en ces endroits que les phénomènes visuels (et physiques) sont les plus intéressants. Puis, l'aspect localisé des tourbillons permet d'utiliser une discrétisation moins dense du domaine tout en permettant de calculer la vitesse dans tout l'espace. par définition, le champ de vitesse calculé à partir de la vorticité est à divergence nulle ; aucune projection n'est nécessaire pour assurer cette contrainte. Dans le cas 2D, pour un fluide non visqueux, l'équation de conservation du tourbillon se résume au transport de celui-ci, ce qui, dans une représentation particulaire, est traité sans erreur. En 3D, en présence de viscosité, l'équation comporte un terme d'étirement et de diffusion ; le nombre de points de

discrétisation augmente aussi de manière conséquente, et ainsi, les méthodes purement particulaires peuvent s'avérer très coûteuses. Cette difficulté est contournée par les méthodes hybrides où la vitesse (et possiblement l'étirement et la diffusion) est calculée sur une grille Eulerienne afin de bénéficier d'algorithmes rapides. Dans mes travaux, c'est la méthode des *Vortex In Cell*, couplée avec une technique de remaillage précis des particules, que j'ai choisi d'adapter.

Dans le chapitre suivant, je présenterai les techniques couramment employées pour la simulation de fluides à plusieurs phases, séparées par une frontière libre. Ce problème est plus complexe que dans le cas d'un fluide monophasique, car il demande l'adaptation de l'équation de Navier-Stokes en présence d'une variation de densité et de viscosité. Il est alors aussi nécessaire d'advecter ces quantités avec le fluide, des méthodes de transport d'interface sont alors couplées à l'équation de conservation de la quantité de mouvement. Je propose dans le chapitre 4 une nouvelle manière de résoudre l'équation de conservation du tourbillon pour un fluide multiphasique, où les interfaces sont transportées par une fonction *level set* par une méthode particulaire avec remaillage.

La difficulté principale rencontrée dans les méthodes vortex est la prise en compte de conditions aux bords d'obstacles. Il n'est pas possible d'imposer directement des conditions aux limites de type non glissement sur la vitesse, à partir du champ de vorticité. Je reviendrai sur ce problème dans le chapitre 2 où je détaille la méthode traditionnelement employée pour ce faire. Dans le chapitre 5, je présenterai une technique nouvelle en méthode vortex, efficace et précise, permettant d'imposer de manière avantageuse des conditions aux bords d'obstacles et de fournir un modèle simple pour les interactions fluide-structure.

Chapitre 2

Modèles de fluides multiphasiques incompressibles et transport d'interface

1 Fluides multiphasiques

1.1 Préambule/Définition

Nous avons étudié jusqu'à présent la dynamique et la modélisation de fluides monophasiques, c'est-à-dire à propriétés physiques — densité et viscosité — constantes. Cette étude est tout à fait valable dans le cas de simulation d'un courant dans un tuyau ou de particules de fumée dans l'air par exemple ; l'étude des fluides multiphasiques s'intéresse à l'interaction de fluides — les **phases** — aux propriétés différentes, comme l'air et l'eau par exemple. La modélisation de fluides multiphasiques (ou *biphasiques* dans le cas particulier de deux fluides) constitue un domaine largement étudié en synthèse d'images car la plupart des phénomènes que l'on souhaite simuler sont de ce type.

Ces fluides sont considérés comme immiscibles, c'est-à-dire qu'ils ne peuvent se mélanger : au niveau microscopique les molécules de l'un ne viennent pas s'intercaler entre celles de l'autre, au niveau macroscopique la séparation entre les phases est clairement délimitée par une *interface* souvent visible à l'oeil nu. La simulation de fluides multiphasiques et le mouvement des interfaces entre les fluides est un problème de type *surface libre*, où des conditions particulières sur le saut des quantités physiques entre les deux phases sont imposées.

Un phénomène physique très souvent observé sur les liquides est celui de la tension de surface (cf. figure 2.1). Cette dernière est due à des forces — prépondérantes à petites échelles — qui tendent à minimiser l'aire de la surface au contact d'autres milieux (l'air et l'objet solide dans la figure précédente) permettant ainsi la création de gouttes (plus ou moins) sphériques « défiant les lois de la nature » ; c'est le même phénomène que l'on peut observer sur les rebords d'un tube à essai : la capilarité. Plus généralement, le mouvement de l'interface et son intégration dans les modèles

FIG. 2.1 – Photographie[2] de gouttes d'eau sur une feuille. Leurs formes particulières sont dues à la tension de surface ; elles restent accrochées au support. On peut observer que les plus petites gouttes sont plus sphériques que les plus grosses, c'est parce qu'elles sont moins volumineuses, leur surface de contact plus petite et ainsi le rapport force de tension de surface/gravité plus faible que pour une goutte plus conséquente.

de simulation sont donc primordiaux dans le domaine de la synthèse d'images et demandent que l'on s'y intéresse en détail.

Tout d'abord je présenterai les équations de Navier-Stokes en présence de plusieurs phases, puis je présenterai différentes méthodes de transport d'interface ainsi que les travaux effectués récemment dans le domaine de la synthèse d'images.

1.2 Équations de Navier-Stokes en densité et viscosité variable

Soient deux fluides f_1 et f_2 évoluant dans deux domaines Ω_1 et Ω_2 tel que $\Omega_1 \cup \Omega_2 = \Omega$, et de densité ρ_1 et viscosité ν_1 pour f_1 (resp. ρ_2 et ν_2 pour f_2) :

$$\rho(\mathbf{x}) = \begin{cases} \rho_1 \text{ si } \mathbf{x} \in \Omega_1 \\ \rho_2 \text{ si } \mathbf{x} \in \Omega_2 \end{cases} \quad \text{et} \quad \nu(\mathbf{x}) = \begin{cases} \nu_1 \text{ si } \mathbf{x} \in \Omega_1 \\ \nu_2 \text{ si } \mathbf{x} \in \Omega_2 \end{cases} .$$

L'interface entre les deux fluides est caractérisée par la surface $\Gamma = \Omega_1 \cap \Omega_2$. Dans chaque fluide sont appliquées les équations de Navier-Stokes avec la contrainte de bord suivante : $\mathbf{u}_1 = \mathbf{u}_2$ sur Γ. Imposer cette contrainte au bord dans deux simulations parallèles complexifierait énormément la résolution des équations, difficulté à laquelle s'ajouterait la représentation de l'interface entre les deux fluides.
Il est possible de considérer la coexistence de deux fluides en interaction dans un

[2] © Remo Kottonau, http ://www.flickr.com/photos/rkottonau/

domaine unique Ω et de les différencier simplement par une densité et viscosité variables au sein du même domaine (nous aborderons la question de la tension de surface plus loin dans le chapitre). En effet, rappelons ici l'équation 1.9 de conservation du mouvement de Navier-Stokes pour les fluides incompressibles monophasiques :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nu \Delta \mathbf{u} + \frac{1}{\rho}\nabla p - \mathbf{f} = 0.$$

Dans le cadre des fluides multiphases, la densité ρ ainsi que viscosité (dynamique ou cinématique, μ ou $\nu = \mu/\rho$) ne sont pas constantes dans le domaine, elles dépendent donc de la position, suivant la dynamique du fluide présent à cet endroit. L'équation de conservation de mouvement modifiée pour une densité et viscosité variable est :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nabla \cdot (\nu \nabla \mathbf{u}) + \frac{1}{\rho}\nabla p - \mathbf{f} = 0 \tag{2.1}$$

où le terme de diffusion $\nabla \cdot (\nu \nabla \mathbf{u})$ est le terme original (en comparaison de l'équation 1.9), utilisé lorsque la viscosité n'est pas constante. En dévelopant ce terme nous obtenons :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nu \Delta \mathbf{u} - (\nabla \mathbf{u} \cdot \nabla)\, \nu + \frac{1}{\rho}\nabla p - \mathbf{f} = 0. \tag{2.2}$$

Il est facile de voir que nous retrouvons l'équation 1.9 lorsque $\nabla \nu = \mathbf{0}$. La différence principale avec les fluides à une seule phase est qu'ici il est nécessaire de transporter la densité ρ via l'équation de transport 1.8 :

$$\frac{\partial \rho}{\partial t} + (\mathbf{u} \cdot \nabla)\rho = 0. \tag{2.3}$$

De manière similaire on transporte ν (ou μ), bien que généralement la viscosité soit constante dans un type de fluide et ainsi facilement déductible de la densité.

Modification de la méthode de projection

Pour résoudre l'équation par la méthode de projection (cf. 4.3), il est nécessaire de résoudre l'équation suivante :

$$\frac{1}{\delta t}\nabla \cdot (\rho \mathbf{u}^*) = \Delta p \tag{2.4}$$

à la place de 1.23 parce que la densité varie. Une attention particulière doit être portée sur la régularité du champ de densité car de forts gradients peuvent poser des problèmes d'instabilité pour la résolution de l'équation de Poisson.

Modification de l'équation de vorticité

L'équation de vorticité quant à elle se complexifie nettement car, notamment, le terme de pression « réapparaît ». Rappelons ici l'équation 1.12 de conservation de la vorticité, à viscosité constante, sous sa forme initiale :

$$\nabla \times \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nu \Delta \mathbf{u} + \frac{1}{\rho}\nabla p - \mathbf{f} \right) = 0.$$

La simplification nous permettant de supprimer la pression n'est plus applicable car le gradient de densité n'est pas nul, ainsi :

$$\begin{aligned}\nabla \times \left(\frac{1}{\rho}\nabla p\right) &= \frac{1}{\rho}\nabla \times (\nabla p) + \nabla\left(\frac{1}{\rho}\right) \times \nabla p \\ &= \nabla\left(\frac{1}{\rho}\right) \times \nabla p.\end{aligned}$$

Il en va de même pour la viscosité :

$$\nabla \times (\nu \Delta \mathbf{u}) = \nu \Delta \omega + \nabla \nu \times \Delta \mathbf{u}.$$

L'équation de conservation de la vorticité s'écrit donc maintenant :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = (\omega \cdot \nabla)\mathbf{u} + \nu \Delta \omega + \nabla \nu \times \Delta \mathbf{u} - \nabla\left(\frac{1}{\rho}\right) \times \nabla p. \tag{2.5}$$

Nous observons que la différence entre cette dernière équation et l'équation de conservation de la vorticité à densité constante est qu'il y a création de vorticité supplémentaire uniquement à la discontinuité entre les phases, c'est-à-dire lorsque $\nabla \rho$ et ∇nu sont non nuls. Cette formulation est relativement complexe car elle demande le calcul de la pression p qui, dans les méthodes vortex, est moins évidente qu'en formulation vitesse. Pour ce faire, il est possible d'exprimer la pression sous la forme d'une équation de Poisson en prenant la divergence de l'équation 2.1. Le calcul de chacun des termes de cette équation est loin d'être évidente car elle demande de nombreux calculs concernant la vitesse, son gradient et son Laplacien. Je présente dans la partie suivante une approximation nous permettant de simplifier l'équation 2.5 dans le cas particulier de faibles variations de densité ; je propose en partie 2 une solution pour le cas général permettant de contourner le problème du calcul de la pression .

1.3 Approximation de Boussinesq

En présence de faible variation de densité entre les deux phases d'un fluide, l'approximation de Boussinesq permet de simplifier la difficulté liée au calcul de la pression. Cette approximation se base sur le fait que dans le cas de fluides à densité quasi identique, la force prépondérente — en comparaison des autres forces entre les deux phases — est celle due à la gravité : la poussée d'Archimède. Cette propriété est souvent utilisée en météorologie pour simplifier la résolution des équations ; voyons comment elle est utilisée dans les équations de Navier-Stokes en formulation vorticité.

Soit $\Delta\rho = \rho_1 - \rho_2$ la différence de densité (à ne pas confondre avec le Laplacien). L'approximation de Boussinesq peut s'appliquer lorsque $\rho' = \Delta\rho/\rho << 1$ (où ρ est soit ρ_1 ou ρ_2, comme ils sont proches). L'équation de conservation de mouvement devient alors :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = \mu \Delta \mathbf{u} - \nabla p + \rho' \mathbf{g} = 0 \tag{2.6}$$

(en ajoutant la gravité en force extérieure), qui nous ramène à l'équation suivante en formulation vorticité :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = (\omega \cdot \nabla)\mathbf{u} + \mu \Delta \omega + \nabla \rho' \times \mathbf{g}. \tag{2.7}$$

Cette équation est beaucoup plus simple à résoudre car il n'y a pas besoin de calculer la pression en formulation vorticité : cela nous ramène à résoudre l'équation de transport-diffusion (avec un terme d'étirement en 3D) augmentée d'une source comptant pour la gravité à l'interface : $\nabla \rho' \times \mathbf{g}$. Le lecteur peut noter que, comme dans la formulation générale, cette source est uniquement présente à l'endroit de la variation de densité, c'est-à-dire à l'interface entre les deux fluides, et lorsque celle-ci n'est pas parallèle à la gravité. Ainsi, pour des fluides stratifiés en position stable (avec une interface horizontale) il n'y a pas de création de vorticité émanant de la force gravitationnelle.

1.4 Tension de surface

La tension de surface (ou tension superficielle) est une force intervenant à l'interface entre deux fluides immiscibles. L'intensité de la force est fonction des deux matériaux concernés, elle est environ de $7{\cdot}10^{-2} N.m^{-1}$ entre l'eau et l'air par exemple. Elle est directement proportionnelle à la courbure de la frontière entre les deux milieux et est responsable de la forme des gouttes d'eau photographiées sur la figure 2.1 dans le cas particulier où trois phases sont en jeu : l'eau, l'air pour les deux fluides et la feuille qui est un solide.

Physiquement, cette tension de surface est liée à un saut de pression — qui est donc discontinue — entre les deux phases du fluides, à travers l'interface Γ :

$$[p] = \sigma \kappa$$

où $[p]$ dénote le saut de pression à l'interface, donc dans la direction de la normale à celle-ci, σ est un coefficient physique de tension de surface et κ la courbure de Γ. La courbure (ainsi que la normale que j'utilise ci-dessous) est entièrement définie par la géométrie de l'interface et nous verrons au sein de la partie suivante comment elle peut être calculée à partir de la représentation choisie pour la surface.

Deux approches courantes sont employées pour prendre en compte ce phénomène : soit par l'intégration du saut de pression comme condition aux bords de la frontière libre (*cf.* [KAC92]), soit par l'ajout d'un terme source localisé sur l'interface. Pour la première méthode, la condition aux bords doit être imposée à travers l'interface et elle est intégrée dans la résolution de l'équation de Poisson concernant la pression (en formulation vitesse). L'article sur la méthode *Ghost Fluid* [FAMO99] que je présente un peu plus loin peut être utilisé afin de résoudre ce problème, cette méthode utilise des *points fantômes* pour simuler le saut de pression dans les schémas de différences finies. Une autre approche (*cf.* [MS94], [Pes77]) consiste à considérer la tension superficielle comme une force extérieure venant agir sur le fluide, et est

directement intégrée dans l'équation de Navier-Stokes en second membre :

$$\mathbf{f} = \sigma\kappa\delta\,\mathbf{n}$$

où δ est la distribution de Dirac non nulle sur l'interface et $\mathbf{n}$ la normale à la surface. Il va de soit que δ est régularisée car l'épaisseur physique de Γ est infime en comparaison de l'échelle de discrétisation utilisée pour les équations ce qui, sans cette régularisation, introduirait une forte discontinuité.

2 Transport d'interface

2.1 Nécessité de méthodes pour le transport d'interface

La limite entre deux fluides est considérée d'un point de vue macroscopique comme une surface, bien que microscopiquement des échanges ont lieu à la frontière. Dans le cadre de la simulation numérique d'écoulements et dans le domaine de la synthèse d'images, ces phénomènes sont considérés comme négligeables. Grâce aux équations de transport de la densité et de la viscosité, il est possible de localiser la position de l'interface au cours du temps, celle-ci est la surface séparant les deux fluides que je noterai f_1 et f_2 (de densité ρ_1 et ρ_2 respectivement) dans la suite du document.

La densité est souvent représentée par une fonction continue — avec un gradient lisse — avant d'être discrétisée pour pouvoir résoudre l'équation de transport en évitant les problèmes d'imprécision émergeants dans le cas de fonctions à fort gradient. Cette représentation induit un phénomène de diffusion (d'un point de vue spatial) de l'interface. L'interface n'est alors plus représentée comme une surface « nette »mais possède une certaine épaisseur évoluant durant la résolution des équations.

Des méthodes appropriées de transport d'interface sont utilisées, tout autant en mathématiques appliqués qu'en synthèse d'images. Ces méthodes doivent être précises pour la conservation des propriétés physiques (le volume des fluides, la densité) autant que pour la localisation de l'interface où se déroule les phénomènes les plus intéressants, et qui est d'une importance capitale dans le domaine de la synthèse d'images puisque c'est la partie du fluide la plus visible dont le mouvement déterminera la plausibilité visuelle de l'animation. Les techniques de suivi d'interface doivent aussi permettre de modéliser le comportement des fluides que nous étudions. Un des principaux aspects est le changement de topologie, qui apparait par exemple lorsque deux gouttes d'eau se rencontrent ou se séparent. Le couplage du transport de l'interface avec la simulation de fluides sous-jacente doit être effectué de manière cohérente car ces deux problèmes sont indissociables et les erreurs commises sur l'un sont répercutées sur l'autre. Par exemple, un fluide dont l'incompressibilité est mal assurée perdra (ou gagnera) du volume et l'interface se réduira (resp. grossira) au cours du temps ; dualement, une interface transportée incorrectement pourra entrainer des courants non désirés dans le fluide.

Les méthodes de transport consistent à représenter, à chaque instant, la localisation spatiale de l'interface de manière *explicite* — où la position de l'interface est connue exactement — ou bien de manière *implicite* lorsque l'interface est une courbe de niveau extraite d'une fonction annexe transportée par le fluide. Ces méthodes sont ensuite utilisées pour déduire la densité et la viscosité du fluide à chaque instant en chaque point de l'espace afin de les intégrer dans les équations présentées dans la partie précédente. Le document d'Emmanuel Maitre [Mai06] présente un aperçu des différentes méthodes existantes largement utilisées dans différents domaines et tout particulièrement dans le cadre de la simulation de fluides ; je m'intéresserai ici plus particulièrement aux méthodes *level set* qui sont l'un des outils de base utilisés pour ma thèse.

2.2 Deux méthodes de transport

Je présente dans les sous-parties suivantes les méthodes phares utilisées en mathématiques appliquées et en synthèse d'images pour représenter les interfaces. Je m'attarderai particulièrement sur la méthode dite de *level set* ou *lignes de niveaux* qui fait partie des méthodes que j'ai utilisées dans ma thèse.

Surfaces paramétrées

En synthèse d'images, il est commun d'utiliser un maillage pour représenter un objet en 3D. Cette représentation est très pratique car elle permet de garder une représentation explicite de la surface au cours du temps. Cette dernière est représentée par un ensemble de points liés les uns aux autres et ce sont ces points qui sont déplacés suivant les forces appliquées à chacun d'eux.

Sur le même principe il est possible de représenter l'interface $\Gamma(t)$ entre deux fluides de manière explicite par une surface paramétrée par un ensemble de points $\mathbf{X}_i(t)$, structurés en maillage, suivant une formulation Lagrangienne. Cette méthode est appelée *front tracking* ou *suivi d'interface* en opposition aux méthodes dites de *capture d'interface* que je présenterai par la suite.

Les points du maillage sont advectés par le fluide et satisfont donc l'équation suivante :

$$\frac{d\,\mathbf{X}_i(t)}{dt} = \mathbf{u}(\mathbf{X}_i(t), t).$$

Un exemple de transport d'une interface grâce à une surface paramétrée est proposé en figure 2.2. Un des avantages est qu'il est possible de calculer facilement la courbure — intervenant dans les forces de tension de surface — en dérivant la paramétrisation de la surface.

La formulation Lagrangienne permet de résoudre exactement l'équation de transport, aux erreurs numériques près, erreurs provenant de l'interpolation de la vitesse $\mathbf{u}$ aux points $\mathbf{X}_i$. Malheureusement cette méthode souffre des inconvénients précédemment présentés concernant la résolution du transport en formulation Lagrangienne :

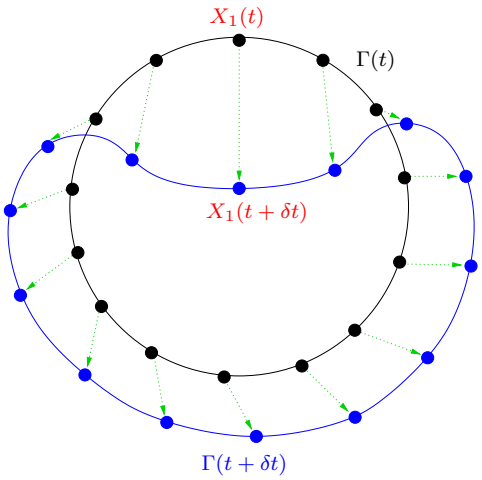


FIG. 2.2 – Illustration d'un transport d'une interface représentée par une surface paramétrée d'un instant t à un instant $t + \delta t$. Les flêches vertes sont les vitesses en chaque point $X_i(t)$.

le maillage peut s'avérer être très déformé par le champ de vitesse sous-jacent et ainsi moins bien représenter l'interface discrétisée entre deux points. Un deuxième désavantage important est que cette représentation ne permet pas naturellement de gérer (détecter et corriger) les changements de topologie apparaissant fréquement dans les écoulements de fluides multi-phasiques. Le problème du changement de topologie est encore plus difficile à résoudre en 3D qu'en 2D. De plus cette méthode ne permet pas de définir aisément un champ de densité lisse entre les deux milieux, par contre il est possible de déduire dans quel fluide un point d'une grille Eulerienne se trouve en fonction de l'intérieur et de l'extérieur de la surface définis par la normale. L'article [DFG+06] présente un algorithme complet pour le suivi d'interface.

Level sets

Les méthodes *level set* ou *ligne de niveaux* sont aujourd'hui largement utilisées dans la simulation de fluides en synthèse d'images comme en mathématiques appliquées. Elles consistent en la représentation implicite de l'interface Γ via une fonction auxiliaire ϕ transportée par le fluide. L'interface est définie par convention comme la ligne de niveaux zéro de ϕ :

$$\Gamma = \{\mathbf{x} \,|\, \phi(\mathbf{x}) = 0\}$$

où la normale $\mathbf{n}$ de l'interface est définie par :

$$\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}$$

et la courbure par :

$$\kappa = \nabla \cdot \mathbf{n} = \nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right).$$

Les méthodes level set font partie de la famille de méthodes dites de *capture d'interface* (*interface capturing* en anglais), elles sont présentées en détail dans le livre de Sethian [Set99]. Dans cette famille, nous pouvons aussi noter la méthode

des champs de phase qui consiste à représenter non pas l'interface mais les différents milieux grâce à une fonction caractéristique possédant une transition lisse d'une épaisseur fixe entre les deux milieux. Cette transition représente l'interface et l'équation de transport est ainsi focalisée à cet endroit. Néanmoins, le couplage des équations du fluide avec une méthode de champ de phase est plus complexe qu'avec des méthodes level set et certaines propriétés intéressantes (que je détaillerai plus loin) de ces dernières n'en sont pas déductibles.

La level set est un scalaire passif transporté par le fluide, elle vérifie donc l'équation de transport :

$$\frac{\partial \phi}{\partial t} + (\mathbf{u} \cdot \nabla)\phi = 0. \tag{2.8}$$

La formulation Lagrangienne est particulièrement adaptée à cette équation. Cette équation est un cas particulier de l'équation d'Hamilton-Jacobi pour le transport de level set :

$$\frac{\partial \phi(\mathbf{x}, t)}{\partial t} + F(\mathbf{x}, t)\,|\nabla\phi(\mathbf{x}, t)| = 0 \tag{2.9}$$

où F, la vitesse de variation de la level set est :

$$F = \mathbf{u} \cdot \mathbf{n} = \frac{1}{|\nabla\phi|}(\mathbf{u} \cdot \nabla)\phi,$$

en ôtant les paramètres $\mathbf{x}$ et t pour plus de clarté.

La level set permet aisément de séparer les deux milieux du fluide, obtenir une *fonction caractéristique*, à l'aide de la fonction Heaviside discontinue suivante :

$$H(\phi) = \begin{cases} 0 & \text{si } \phi < 0 \\ 1 & \text{sinon} \end{cases}$$

dont la dérivée est la distribution Dirac $\delta(\phi)$ telle que $\int_{-\infty}^{+\infty} \delta(y)\,dy = 1$ et $\delta(\phi)$ est nulle partout sauf en 0. Ainsi, si le fluide f_1 (resp. f_2) possède une quantité physique Q_1 (resp. Q_2) dans un domaine Ω_1 (resp. Ω_2), et qu'une grandeur P_Γ est portée sur l'interface, nous pouvons extraire de la level set les propriétés suivantes :

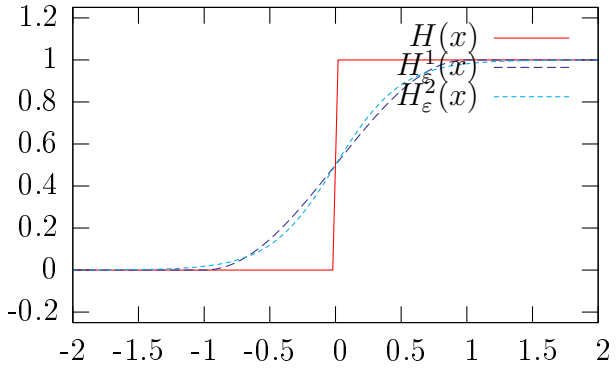
$$\begin{aligned} \Omega_1 &= \{\mathbf{x} \,|\, \phi(\mathbf{x}, t) > 0\}\,, \\ \Omega_2 &= \{\mathbf{x} \,|\, \phi(\mathbf{x}, t) < 0\}\,, \\ P(\mathbf{x}) &= P_\Gamma\,\delta(\phi) \text{ et} \\ Q(\mathbf{x}) &= H(\phi(\mathbf{x}))Q_1 + (1 - H(\phi(\mathbf{x})))Q_2, \end{aligned}$$

où cette dernière équation défini la quantité physique Q dans tout l'espace. Toute fonction $f(\phi)$ résout l'équation de transport :

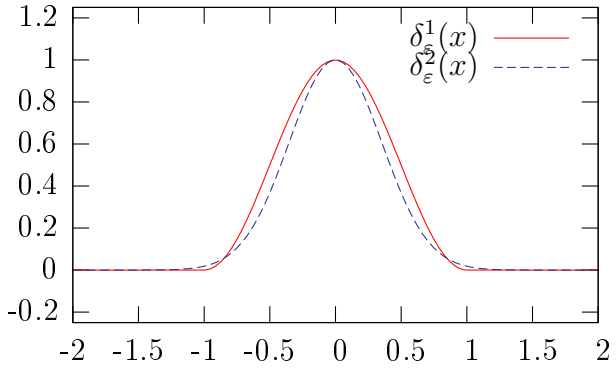
$$\frac{\partial f(\phi)}{\partial t} + (\mathbf{u} \cdot \nabla)f(\phi) = 0.$$

En prenant $Q = \rho = H(\phi)\rho_1 + (1 - H(\phi))\rho_2$, nous obtenons pour la densité du fluide biphasique (l'union de f_1 et f_2) l'équation de conservation de la masse 1.8.

Comme je l'ai présenté dans la partie précédente, une propriété importante pour la résolution des équations est la bonne continuité des quantités. Pour éviter toute discontinuité due à l'utilisation d'une fonction de type Heaviside, il est nécessaire d'utiliser une version régularisée de cette dernière, notée H_ε et telle que $H_\varepsilon(\phi) \to H(\phi)$ lorsque $\varepsilon \to 0$. Similairement, la dérivée de la fonction régularisée est une fonction ζ_ε régularisée telle que $\zeta_\varepsilon(\phi)|\nabla\phi| \to \delta(\phi)$ lorsque $\varepsilon \to 0$. Evidemment, on souhaite que ce paramètre soit le plus petit possible pour se rapprocher idéalement des équations de départ. Typiquement ε est de l'ordre de quelques points de grille : $\varepsilon \simeq h$. La figure 2.3 présente une illustration de telles fonctions.



(a) Fonctions Heaviside régularisées.



(b) Fonctions Dirac régularisées.

FIG. 2.3 – Exemples de fonctions Heaviside et Dirac régularisées pour $\varepsilon = 1$.

La plupart du temps, la level set est définie initialement comme une fonction distance, ou en d'autres termes $|\nabla\phi| = 1$. Cette propriété n'est pas conservée par le transport de la level set, pour le vérifier il suffit d'étudier la variation de la norme du gradient de ϕ. La fonction est étirée par le fluide et pour cette raison l'épaisseur ε initiale de l'interface n'est pas conservée malgré le fait que le « zéro »de la fonction ϕ soit bien transporté. Ce phénomène n'est pas désirable car il entraîne la perte de localité de forces qui peuvent se situer à l'interface comme la tension de surface par

exemple.

Pour pallier ce problème, de nombreuses méthodes dites de *reinitialisation* (ou *redistancing*) permettent de conserver une fonction distance pour ϕ. L'idée est de résoudre l'équation d'Hamilton-Jacobi suivante :

$$\frac{\partial \psi}{\partial \sigma} = Signe(\psi_0)(1 - |\nabla \psi|)$$

avec pour condition initiale $\psi_0 = \phi$ et la fonction $Signe(y)$ valant $-1, 0$ ou $+1$ suivant le signe de l'argument. Cette équation conserve la localisation de l'interface, c'est-à-dire $\psi_\infty = \phi$ pour les points $\{\mathbf{x} \,|\, \phi(\mathbf{x}) = 0\}$, et converge vers une fonction fonction distance pour un temps σ infini. L'article récent [GR00], par exemple, propose une méthode pour résoudre cette EDP introduite par Sussman et al. [MS94] dans le cadre de fluides biphasiques.

Une autre méthode largement utilisée en informatique graphique — et aujourd'hui dans divers domaines comme la vision par ordinateur — est la *Fast Marching Method*. L'algorithme consiste tout d'abord à déterminer les points de la grille Eulerienne les plus proches de l'interface, formant ainsi une bande. À partir de cette bande considérée comme fixe, une nouvelle valeur pour ψ est affectée aux points alentours de la bande, telle que celle-ci soit la distance à l'interface. En avançant de proche en proche, la solution obtenue par l'algorithme est la fonction distance désirée. La méthode originale a été proposée par J.A. Sethian dans [Set99].

Il est remarquable que le fait de modifier la level set par une méthode de reinitialisation induit une erreur non négligeable dans l'équation de transport de la level set *a posteriori*. En effet, même si la position de l'interface est conservée, les valeurs de ϕ environnantes sont altérées et transporter la level set reinitialisée ne permettra pas d'obtenir la même interface après transport que si l'opération n'avait pas eu lieu. Ceci est dû aux erreurs numériques introduites par la méthode de résolution de l'équation d'advection. De plus, la plupart du temps, pour la résolution des équations de Navier-Stokes une régularisation des quantités physiques est utilisée en fonction de ϕ et ainsi le transport de ces quantités est aussi biaisé par la reinitialisation.

Dans l'article [CM06a], Cottet et Maitre proposent une autre méthode pour obtenir une approximation locale d'une fonction distance : le rapport entre ϕ et la norme de son gradient : $\frac{\phi}{|\nabla \phi|}$ approche le comportement d'une fonction distance au voisinnage de l'interface. Pour interpréter cette propriété intuitivement, on peut imaginer qu'en écrasant une fonction distance 1D autour de l'interface, les valeurs de ϕ vont croître moins vite, la norme du gradient sera plus faible, et ainsi le rapport des deux compensera la perturbation. Cette méthode a montré des résultats comparables (*cf.* [CM06a]) aux méthodes de réinitialisation tout en possédant un coût algorithmique négligeable par rapport à ces dernières.

Nous verrons par la suite, dans la partie 2.2 concernant les membranes elastiques, ainsi que dans mes travaux en chapitre 5, que les informations fournies par la level set non reinitialisée permettent de conserver le taux de déformation induit sur l'interface et se trouvent très utiles pour l'intégration de solides en interaction avec le fluide ainsi que pour la gestion de collisions.

Je présenterai les méthodes principales de transport d'interfaces gravitant autour de méthodes *level sets* dans la partie 3.2 consacrée aux méthodes utilisées en synthèse d'images. Les méthodes level set faisant partie des briques de base utilisées lors de ma thèse, je reviendrai naturellement sur le sujet dans les chapitres 4 et 5.

3 Transport d'interface et fluides multiphasiques en synthèse d'images

Préambule

Dans cette partie je m'intéresse tout particulièrement aux méthodes utilisées en synthèses d'images pour le transport d'interface et la simulation de fluides à plusieurs phases. Ces méthodes sont pour la plupart issues de travaux effectués en mathématiques appliqués, le lecteur pourra se référer aux articles cités pour un point de vue théorique et numérique plus poussés.

3.1 Fluides multiphasiques

Historiquement, les méthodes de simulations de fluide en synthèse d'images se sont initialement limitées à des fluides biphasiques où uniquement la vitesse du liquide (l'eau par exemple) était simulée et où la vitesse dans le gaz (l'air) n'était pas traitée. L'eau étant beaucoup plus dense que le gaz, le mouvement de ce dernier était considéré comme négligeable. En conséquence les calculs étaient focalisés au niveau du liquide ainsi qu'à son interface car c'est le mouvement de cette dernière que l'on souhaite observer principalement. Dans ce cas la vitesse n'est pas définie partout dans le domaine, mais uniquement sur les cellules contenant du liquide et à l'interface. Une autre raison pour ce choix provient du fait que les équations sont plus difficiles à résoudre en présence de forts gradients de densité et du saut de pression au travers de l'interface. Les travaux des dernières années considèrent les fluides comme entièrement biphasiques ou plus récemment multiphasiques (avec deux phases ou plus).

Je présenterai tout d'abord la méthode courante nommée *Markers And Cells*, puis la méthode *Volume Of Fluid* et enfin la méthode *Ghost Fluid* permettant d'inclure des discontinuités dans les grandeurs physiques à travers l'interface. Il est important de noter que ces méthodes sont intrinsèquement liées à des techniques d'advection d'interface ; dans cette partie je m'intéresse uniquement à la partie simulation du fluide et non au transport de la frontière immergée, cette dernière étant traitée en partie 3.2.

Markers And Cells

La méthode *MAC* (pour *Markers And Cells*), tout d'abord introduite par Harlow et Welch [HW65] dans le cadre des méthodes numériques et ensuite reprise par Foster

et Metaxas [FM96] dans un des premiers articles phares de simulation de fluide en synthèse d'images, est basée sur une représentation Eulerienne du liquide couplée à une représentation Lagrangienne de l'interface entre celui-ci et l'air environnant. Afin de diminuer les oscillations numériques et d'augmenter la stabilité, les vitesses sont stockées non pas au centre des cellules, mais sur les faces de celles-ci. On parle alors de *grille décalée* où à chaque face est associée la vitesse normale en cette dernière, le flux entrant/sortant dans cette direction particulière. La vitesse en n'importe quel point est obtenue par interpolation.

D'une autre part, la surface libre du fluide est représentée par un ensemble de particules à une échelle plus fine que la grille. Ces particules situées autour de l'interface sont transportées par le fluide et permettent à chaque instant de connaître la position de la frontière au niveau de la grille Eulerienne et ainsi de fournir une localisation spatiale pour inclure les conditions aux bords adéquates.

Le lecteur pourra se référer à la thèse d'Olivier Génevaux [Gén06] dont les travaux sont basés sur cette méthode dans le cadre de la visualisation et de la simulation d'interactions entre un liquide et des objets déformables.

Méthode Volume-Of-Fluid (VOF)

La méthode VOF [CWH81] consiste à attribuer à chaque cellule d'une grille Eulerienne une valeur F comptant pour la fraction de fluide contenue à l'intérieur de celle-ci. Par convention est donnée la valeur 1 pour une cellule remplie entièrement de liquide, et 0 pour une cellule remplie de gaz. Pour les cellules du maillage contenant l'interface entre les deux est donnée une valeur représentant le pourcentage d'occupation d'un des fluides en fonction du volume de la cellule.

Afin de déplacer le liquide et surtout l'interface du fluide avec l'équation de transport :

$$\frac{\partial F}{\partial t} + (\mathbf{u} \cdot \nabla) F = 0,$$

il est nécessaire de connaître le champ de vitesse à l'intérieur du fluide ainsi que dans les cellules voisines de Γ. Pour ce faire, la vitesse est d'abord calculée au points intérieurs et dans les cellules contenant l'interface, en fixant de bonnes conditions aux bords pour les équations, notamment sur la pression (par exemple la pression atmosphérique est prise pour les cellules de l'air). La vitesse est ensuite *extrapolée* au delà de l'interface, vers l'autre phase, dans la direction de la normale.

Après avoir transporté la fraction de volume F, une représentation explicite de l'interface est construite à partir de celle-ci grâce à des méthodes de reconstruction (*contouring* en anglais) de surface dans les cellules et ensuite les fractions de volume sont de nouveaux calculées pour le pas de temps suivant.

Pour diminuer les instabilités pouvant apparaître dans la résolution des équations en présence d'une forte discontinuité dans la densité, le champ de densité est généralement lissé autour de l'interface.

Méthode Ghost Fluid

La méthode *Ghost Fluid* introduite par Fedkiw et al. [FAMO99] est une méthode permettant de traiter lors de la discrétisation des équations de Navier-Stokes un saut de pression à l'interface. Le choix a ici été fait de considérer une discontinuité entre les deux phases, à la différence des approches présentées précédemment. L'article *Discontinuous Fluids* [HK05b] de Hong et Kim en présente une adaptation en synthèse d'images dans le cas de fluides incompressibles. Dans cette méthode, les schémas de différences finies traditionnels sont utilisés pour dériver les grandeurs physiques, et notamment pour résoudre l'équation de Poisson sur la pression afin d'assurer l'incompressibilité du fluide. Les problèmes induits par la discontinuité sont palliés grâce à l'extrapolation de la pression au travers de l'interface — pour créer les *ghost nodes* (noeuds fantômes) — tout en conservant le saut de pression, ici proportionnel à la tension de surface (cf. figure 2.4).

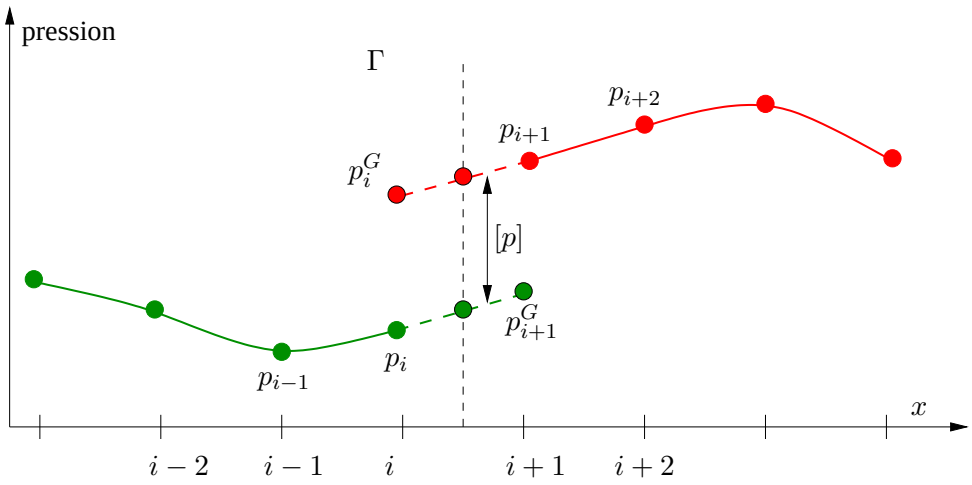


FIG. 2.4 – Saut de pression : la pression est extrapolée à travers l'interface sur les mailles suivantes afin de compter pour le saut de pression $[p]$. Les valeurs des *ghosts nodes* (notées par un G en exposant) sont utilisées dans les schémas de différences finies d'un côté comme de l'autre de l'interface.

3.2 Méthodes pour le transport

En filigrane derrière les méthodes de calcul de l'écoulement de fluides multiphasiques est le transport des phases elles-même et principalement de l'interface entre chacune. Comme je l'ai présenté en partie 2, il existe de nombreuses méthodes de transport d'interface. En synthèse d'images, les principales utilisées sont basées sur les méthodes de type *level set* pour deux raisons :

- leur capacité à capturer naturellement les changement de topologie : phénomène qui est recherché visuellement,
- leur simplicité de compréhension par le *non mathématicien* (en comparaison des méthodes de champ de phases par exemple) ainsi que d'implémentation

— tout du moins pour la méthode standard.

Je présente ici plus en détails les travaux principaux réalisés autour de cette méthode très populaire pour le transport d'interface. Je rappelle ici l'équation de transport d'une level set dans le cas particulier de la simulation de fluide :

$$\frac{\partial \phi}{\partial t} + (\mathbf{u} \cdot \nabla)\phi = 0.$$

C'est cette équation que l'on cherche à résoudre au mieux pour effectuer le moins d'erreurs possible. Un des problèmes fréquemment rencontré est la perte de volume, principalement lors de changements de topologie et lorsque la courbure de ϕ devient importante.

Particle level set

Afin d'améliorer la conservation du volume d'un fluide via le transport de l'interface, la méthode dite de *particle level set* mélange une résolution Eulerienne et Lagrangienne de l'équation. Des particules transportant *exactement* la level set sont utilisées pour corriger les erreurs induites par les méthodes Euleriennes de transport. Ici l'exactitude vient du fait que dans un formalisme Lagrangien, le transport de l'interface est effectué sans aucune erreur du fait même du choix de l'évolution du maillage.

L'algorithme (représenté graphiquement dans la figure 2.5) est le suivant :

1. Un ensemble de particules p sont créées dans une bande proche de l'interface et à chaque particule est associée la valeur initiale de la level set en ces points par interpolation : $\phi_p = \phi(\mathbf{x}_p, 0)$. Il y a donc des particules « chargées »positivement et d'autre négativement. En pratique, environ 32 particules sont utilisées par cellule de la grille.
2. A chaque pas de temps l'équation d'advection est résolue sur la grille pour faire évoluer ϕ. Les particules sont déplacées avec la vitesse du fluide. Idéalement, les particules négatives (resp. positives) devraient être situées à l'intérieur (resp. extérieur) de la level set correspondant à $\phi < 0$. Mais les erreurs numériques peuvent induire des incohérences entre les particules et la grille, principalement aux endroits de fortes déformations et fortes courbures.
3. Les particules sont utilisées pour corriger les valeurs de la level set sur la grille via l'utilisation d'une fonction d'interpolation.

Lors de fortes déformations dues à un fluide turbulent, les particules peuvent se trouver mal réparties et représentent dans ces cas moins bien la level set. Ainsi un mécanisme de remaillage (création de particules *fraîches*) est utilisé régulièrement afin de redistribuer correctement les particules.

Il est important de noter que dans l'article initial de Enright, Fedkiw et al. [EFFM02], les auteurs implémentent la correction de la level set par des particules représentant la distance à l'interface de manière sphérique. Une méthode plus adaptée, proposée par Pighin et al. dans [PPCC05], consiste à considérer les particules

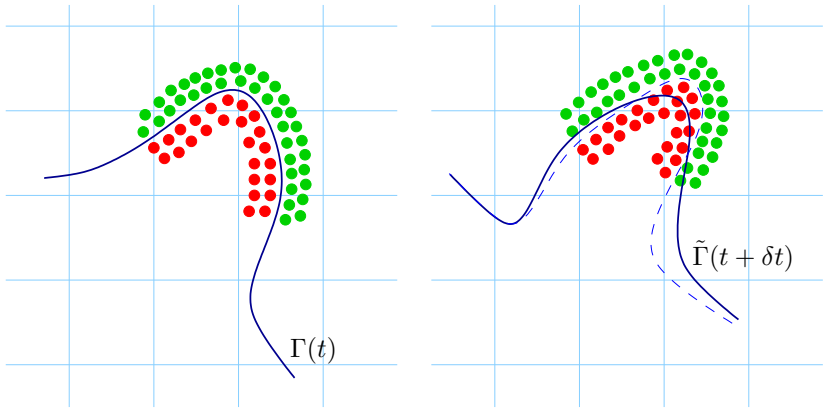
comme étant orientées vers la surface, en ajoutant une équation pour le transport du gradient de ϕ et en corrigeant la fonction dans cette direction. Cette dernière méthode permet une correction plus précise de la level set.

Il est peu discuté de l'utilisation conjointe des méthodes de reinitialisation des particules. En effet, après réinisialisation, les particules et la grille ne transportent pas une level set identique, et ainsi la correction apportée par les particules peut déformer de manière non cohérente ϕ sur la grille. Ce fait vient quelque peu contredire l'intérêt apporté par l'ajout de particules sous-maille.

La méthode des *particle level set* peut être considérée comme un raffinement local du maillage autour de l'interface. En effet, une discrétisation plus fine avec un maillage Eulerien permettrait d'obtenir une précision (pour la conservation de la masse) équivalente à l'utilisation de particules conjointes à la même échelle. De plus, le choix de la fonction d'interpolation pour la correction de la level set — *où, quand et comment* corriger — est primordiale et peut introduire au contraire un biais si elle est mal adaptée. Le choix du type d'interpolation, du remaillage ainsi que de la méthode de réinitialisation souvent adjointe sont les points sensibles de cette technique, et dans le sens de la méthode développée initialement, celle-ci s'éloigne quelque peu de la simple résolution de l'équation d'advection.

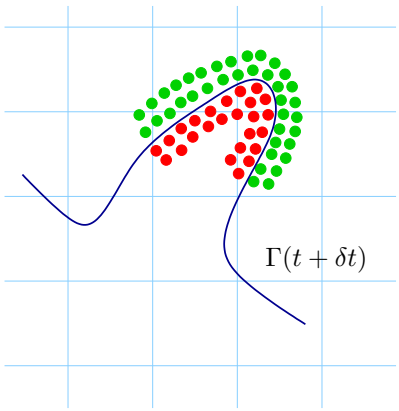
La raison principale — et souvent non discutée — du développement de cette méthode est que l'utilisation d'une représentation Lagrangienne à une échelle fine et Eulerienne à une échelle plus grossière permet de ne point trop souffrir de la contrainte due à la condition CFL sur le pas de temps — qui, je le rappelle, est proportionnelle à la taille d'une maille — condition inexistante en formulation Lagrangienne. Un autre point crucial est le fait qu'un algorithme de réinitialisation de la level set est fréquemment employé. Cet algorithme souvent coûteux — malgré son faible coût algorithmique, il l'est plus que le simple transport — rend prohibitif l'utilisation de grilles Euleriennes fines. Ces critères, rarement explicités, laissent la porte ouverte au développement d'autres méthodes. Nous verrons dans la partie concernant la méthode *Lagrangian Particle Level Set* qu'une méthode purement Lagrangienne permet d'obtenir des résultats équivalents voire meilleurs pour un nombre de particules beaucoup moins élevé.

Les méthodes gravitant autour des *Particle Level Set*, hybrides Euleriennes-Lagrangiennes, permettent aussi de mieux visualiser les détails de l'interface lorsque la grille Eulerienne devient trop grossière pour représenter les petites particules d'eau s'échappant de la surface principale du fluide. Dans ces cas particuliers, les particules de level set sont rendues directement sous la forme de sphères lors de la génération des images.



(a) Level set et particules à l'instant t.

(b) Level set et particules à l'instant $t+\delta t$, après advection : l'interface transportée par les particules ne correspond plus à celle capturée par la grille.



(c) Level set et particules à l'instant $t+\delta t$ après correction grâce aux particules.

FIG. 2.5 – Algorithme des Particles Level Set : correction des valeurs sur la grille par les particules plus précises après advection, zoom sur une partie de l'interface. Γ est l'interface dessinée en bleu, les points rouges et verts représentent les particules du côté positif et négatif (respectivement) portant la level set. Dans la figure du milieu, la courbe en traits représente la « vraie »interface alors que celle en trait gras représente l'interface $\tilde{\Gamma}$ après advection sur la grille et avant correction.

Semi Lagrangian Contouring Method

Bargteil et al. [BGOS06] ont introduit dans le domaine de la synthèse d'images une méthode hybride de suivi d'interface et de résolution semi-Lagrangienne de l'équation de transport. Leurs travaux[3] sont basés sur ceux de Strain [Str01] qui s'est tout d'abord intéressé à ce couplage. La méthode consiste à utiliser un maillage

[3]dont une implémentation est disponible à l'adresse

explicite de l'interface — maillage à partir duquel une fonction distance est aisément (mais chèrement) calculable — et à utiliser une méthode semi-Lagrangienne pour obtenir la level set transportée au pas de temps suivant, en se basant sur la distance au maillage explicite existant au pas de temps précédent. La raison de l'utilisation d'une méthode semi-Lagrangienne pour calculer l'advection plutôt que de transporter simplement le maillage, est que, comme présenté en partie 2, cette dernière opération peut créer de fortes distorsions de maillage et rend difficile le traitement des changements de topologie.

Rappelons ici le principe de la méthode semi-Lagrangienne dans le cadre du transport de level set : elle consiste à calculer les valeurs de celle-ci aux points de discrétisation en suivant la trajectoire du fluide en arrière dans le temps pour obtenir la valeur de ϕ à l'endroit atteint. Ceci se traduit par l'équation :

$$\phi(\mathbf{x}, t + \delta t) = \phi(\mathbf{x}', t)$$

où $\mathbf{x}' = \mathbf{s}(\mathbf{x}, t)$ et $\mathbf{s}$ représente la trajectoire d'une particule du fluide passant par $\mathbf{x}$ au temps $t + \delta t$ (*cf.* partie 4.3 pour plus de détails). La valeur de ϕ au pas de temps précédent est calculée de manière directe comme la distance à un maillage explicite représentant la surface. L'algorithme (dont une représentation graphique est donnée en figure 2.6) peut donc être résumé ainsi : étant donné un maillage $\Gamma_M(t)$ représentant explicitement l'interface Γ au temps t, calculer la level set $\phi(\mathbf{x}, t + \delta t)$ représentant implicitement l'interface Γ au temps $t + \delta t$. Pour tout point $\mathbf{x}$ de la grille :

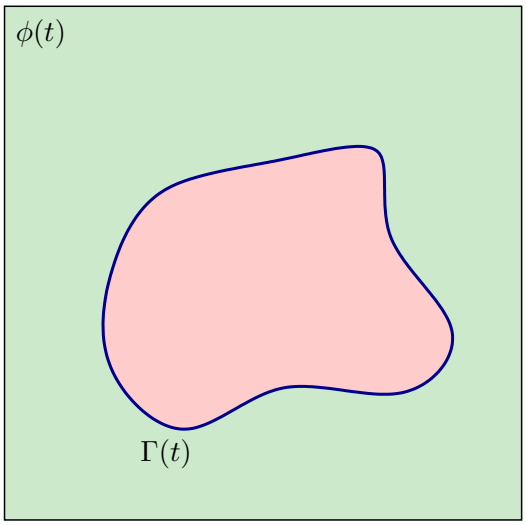
1. tracer la trajectoire $\mathbf{s}(\mathbf{x}, t+\delta t)$ en arrière dans le temps de façon semi-Lagrangienne : $\mathbf{x}' = \mathbf{s}(\mathbf{x}, t)$,
2. obtenir la distance d du point $\mathbf{x}'$ à l'interface grâce au maillage $\Gamma_M(t)$ en conservant la distance point-triangles la plus proche,
3. affecter cette distance comme nouvelle valeur pour la level set : $\phi(\mathbf{x}, t + \delta t) = d \simeq \phi(\mathbf{x}', t)$.

Grâce à la grille Eulerienne du pas de temps précédent, il est possible d'accélérer le calcul de la distance entre le point $\mathbf{x}'$ et les triangles, en ne considérant que les triangles proches de $\mathbf{x}'$.

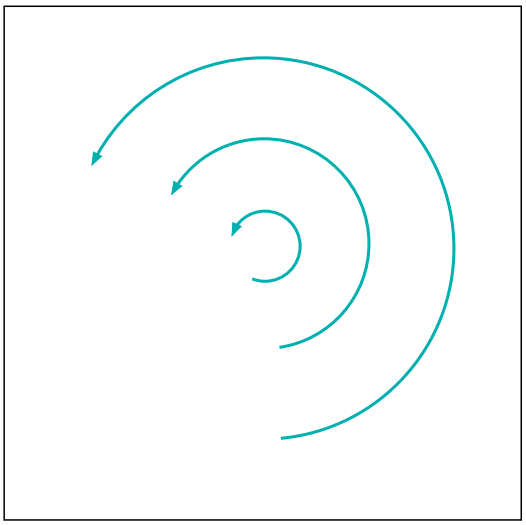
Après cette étape, ϕ est connue en chaque point de la grille et un nouveau maillage $\Gamma_M(t + \delta t)$ est créé, qui servira de base pour l'itération suivante. Des méthodes de *contouring* (de type *Marching Cubes* [LC87], par exemple) sont employées pour créer un maillage fermé représentant au mieux l'interface représentée implicitement par la fonction level set ϕ. Il est important de remarquer que, comme la level set sur la grille n'est pas une fonction distance (elle est une fonction distance transportée par le fluide), celle-ci est réinitialisée en calculant simplement la distance de chaque point $\mathbf{x}$ au triangle de $\Gamma_M(t + \delta t)$ le plus proche.

Cette méthode a l'avantage de permettre de transporter d'autres informations que la level set avec le fluide : par exemple Bargteil et al. l'utilisent pour transporter une texture associée aux points du maillage de l'interface en gardant la cohérence

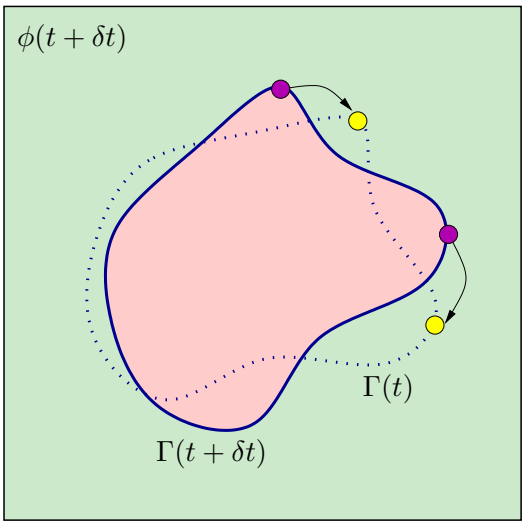
temporelle, ce qui est d'un intérêt majeur pour la visualisation. Cette opération est effectuée en mettant en correspondance chaque point du nouveau maillage $\Gamma_M(t+\delta t)$ avec un point (ou plus généralement une interpolation des points d'un triangle) du maillage précédent. Je renvoie le lecteur à la thèse d'Adam Bargteil [Bar06] pour plus de détails sur ce sujet.



(a) Level set ϕ et interface Γ à l'instant t.



(b) Champ de vitesse : rotation autour du centre.



(c) Level set ϕ et interface Γ à l'instant $t+\delta t$ après advection. Les points mauves ont pour valeur la distance à $\Gamma(t)$ aux points jaunes associés.

FIG. 2.6 – Calcul de $\phi(\mathbf{x}, t+\delta t)$ sur une grille Eulerienne, grâce au maillage $\Gamma_M(t)$ (en pointillés sur la figure de droite) du pas de temps précédent.

Bien que des exemples qualitatifs soient présentés dans le papier, aucun test quantitatif ne vient valider la conservation du volume transporté. La création d'un

nouveau maillage à partir de la level set sur la grille introduit un biais dans l'équation de transport en comparaison avec une représentation implicite de la fonction. En effet, un choix doit être effectué pour approximer la forme de l'interface au sein d'une cellule (je rappelle que la technique des *Marching Cubes* permet de construire une face par cellule de la grille contenant l'interface), faisant fi de la consistance globale avec la fonction level set transportée. La bonne conservation du volume apparente peut s'expliquer par une définition très fine de la grille employée, basée sur une structure hiérarchique de type *octree*.

Lagrangian Particle Level Set

Cette méthode proposée par Hieber et Koumoutsakos [HK05a] se base sur une résolution 100% Lagrangienne de l'équation de transport. En effet, à la différence de la méthode des *Particles Level Set* (notée *Hybrid Particle Level Set Method*, HPLSM, en opposition à la purement *Lagrangienne Particle Level Set Method*, LPLSM, dans l'article sus-cité) introduite par Fedkiw et largement utilisée en synthèse d'images, cette méthode ne se base pas sur une advection de la level set sur la grille (même couplée avec des particules de correction), mais utilise uniquement des particules afin de résoudre très simplement l'équation de transport dans un formalisme purement Lagrangien, où les particules sont advectées par le fluide comme des scalaires passifs. Chaque particule située en $\mathbf{x}_p$ porte une quantité ϕ_p :

$$\frac{D\,\phi(\mathbf{x}_p)}{Dt}\frac{D\,\phi_p}{Dt} = 0$$

et, similairement aux particules de vorticité (*cf.* 4.5), la level set est connue partout dans l'espace grâce à un noyau de mollification ζ adaptés :

$$\phi(\mathbf{x}) = \sum_{p=1}^{N} \phi_p \zeta(\mathbf{x} - \mathbf{x}_p).$$

Par dérivation de ce noyau (suffisamment régulier), il est aisé de calculer la normale et la courbure de la level set en chaque point.

Pour éviter tout problème de distorsion (apparaissant lorsque les particules sont trop proches ou trop éloignées) émanent de l'aspect turbulent du champ de vitesse transportant les particules, un procédé de remaillage est employé, à la manière des particules de vorticité. Il est même envisageable d'utiliser des particules à support variable et d'utiliser une répartition adaptative des particules pour focaliser la précision au niveau de l'interface.

Les résultats obtenus grâce à cette méthode simple mais plus proche mathématiquement des équations de transport sont voisins de ceux obtenus grâce à la HPLSM, concernant la conservation du volume sur des cas tests de déformation d'un volume par un champ de vitesse fixe. De manière surprenante, au vu des discussions soulevées sur les méthodes de reinitialisation dans les articles sur les level set, les résultats de [HK05a] avec reinitialisation après remaillage des particules sont moins bons que

ceux sans cette opération. Ceci tend à faire penser que les méthodes de reinitialisation, si non nécessaires pour le problème sous-jacent, peuvent être contreproductives en plus d'être coûteuses.

4 Bilan du chapitre

Nous avons pu voir dans ce chapitre les principales méthodes en simulation de fluides multiphasiques, sujet très en vogue en ce moment en synthèse d'images car les demandes en animation et rendu sont presques exclusivement focalisées autour des interactions eau-air. Pour l'étude de phénomènes plus complexes et afin d'obtenir des animations encore plus intéressantes, les interactions fluides-solides sont aujourd'hui un sujet en plein essor et je m'y intéresserai dans le chapitre suivant.

Le lecteur aura noté qu'un des points primordiaux dans le calcul d'écoulements de fluides multiphasiques est le problème de transport de l'interface vers lequel de nombreux travaux sont aujourd'hui encore orientés. Les méthodes existantes permettent d'effectuer un suivi précis de l'interface — dans le sens de la conservation du volume — et fournissent aux calculs d'écoulements une brique de base capitale.

Je montrerai au chapitre 4 et plus particulièrement dans la partie 2 comment il est possible de simuler des fluides multiphases dans une formulation vorticité avec l'utilisation combinée des méthodes level set.

Chapitre 3

Couplage fluide-structure

1 Introduction

Nous venons de voir le problème de la simulation de fluides à plusieurs phases ainsi que les méthodes existantes permettant de simuler de tels milieux. Ces types de fluides nous intéressent particulièrement dans le cadre de la synthèse d'images de par leur présence quotidienne dans notre environnement. La simulation simultanée de fluides en interaction avec des milieux possédant un comportement physique différent, tels que les solides rigides ou déformables, est un problème complexe car elle requiert le couplage des forces agissant sur chacun des milieux, couplage qui doit être effectué de manière rigoureuse dans chacun des modèles employés — que cela soit pour la mécanique du fluide et du solide par exemple. Cette question nous concerne particulièrement car ce sont ces interactions qui sont les plus intéressantes — où l'intérêt est ici relatif aux phénomènes les plus complexes et où les applications sont nombreuses — dans l'étude des phénomènes physiques, comme par exemple l'écoulement autour d'une aile d'avion ou d'un bateau, et tout particulièrement en synthèse d'images où la multiplicité des objets présents dans une scène implique leur interaction qui doit être au final plausible (dans le sens physique et visuel) pour l'observateur.

Les difficultés émanent, plus précisément, de la différence de modélisation de chacun des milieux — par exemple une représentation Eulerienne pour un fluide, et Lagrangienne en considérant uniquement le mouvement du centre de gravité pour un solide rigide. C'est à l'interface entre ces domaines que vont se dérouler les échanges physiques qui sont à incorporer dans les équations de mouvement de chaque milieu. L'ensemble des forces s'appliquant au solide (par exemple) sera constitué des forces extérieures (la gravité par exemple) et des forces hydrodynamiques agissant sur lui. Il en va de même pour le fluide : le couplage de toutes ces forces compose la dynamique du système global. Physiquement, à l'interface entre les deux milieux la vitesse doit être identique. En effet, comme dans un fluide visqueux en interaction avec une paroi il n'y a pas de glissement, la vitesse du fluide y est égale à celle du solide. Par exemple, autour d'un solide statique comme le pilier d'un pont, cette propriété va induire la création d'une trainée de vortex après celui-ci, dans le sens

du courant, du fait du différentiel de vitesse avec la rivière. Inversement, le fluide va modifier le mouvement de l'objet : les forces hydrodynamiques présentes à l'interface (la pression du fluide sur la paroi dans le cas d'un fluide Newtonien incompressible) entrent dans la dynamique de l'objet ; dans le même exemple, le pilier qu'on imagine statique subit tout de même une forte pression qui le pousse (principalement) vers l'aval si bien qu'un fort débit pourra faire rompre la structure.

Au delà du problème de modélisation intervient la difficulté de la discrétisation de l'espace ainsi que de la représentation géométrique des domaines. Que l'on représente l'interface de manière explicite ou implicite, pour le solide comme pour le fluide, il est important de contrôler la discrétisation spatiale du fluide (puisque c'est le point de vue qui nous intéresse dans cette thèse) afin d'intégrer au mieux la structure avec laquelle il interagit. C'est pourquoi deux grandes familles de méthodes existent pour gérer ces interactions : les méthodes où le maillage du fluide est adapté pour « coller »aux bords de l'objet considéré et les autres où ce dernier est intégré de manière implicite en superposant un maillage « traditionnel »pour la simulation de fluides au maillage de l'objet.

La première famille de méthodes est basée sur une formulation ALE (*Arbitrary Lagrangian Eulerian*) des équations de Navier-Stokes où, je le rappelle, le maillage est déplacé et déformé au besoin entre une grille cartésienne et le point de vue d'une particule de fluide ; l'article de Hu *et al.* [HPZ01] (par exemple) présente cette technique en détail dans le cadre d'interaction avec un solide. Cette discrétisation ALE est particulièrement adaptée au problème car elle permet de lier physiquement le maillage du fluide aux frontières de l'objet discrétisées spatialement. C'est en ces points du maillage que vont se dérouler les échanges entre les deux milieux : la vitesse du fluide en ces derniers doit être identique à celle du solide et les forces agissant sur la paroi peuvent y être calculée, elles sont ensuite « transmises »au modèle physique pour la dynamique de la structure considérée.

Ces modèles sont très précis car ils permettent d'adapter le maillage du fluide autour de l'objet et de focaliser les calculs à cet endroit. De plus, la vitesse est explicitement la même (aux erreurs numériques près) à l'interface entre les deux milieux du fait de la représentation explicite de celle-ci. Malheureusement, comme pour toutes les méthodes ALE, ces modèles peuvent être complexe à mettre en oeuvre du fait justement de la précision du maillage qui doit être créé et transporté en respectant les frontières de la structure, ce qui peut s'avérer difficile pour des géométries complexes en 3D. Comme pour les méthodes Lagrangiennes, ces méthodes ALE nécessitent un remaillage lors de grandes déformations du maillage, ce qui peut être fréquent proche d'objets solides pour des fluides turbulents par exemple, et ne bénéficient pas de l'avantage en terme de temps de calcul des méthodes basées sur une grille Eulerienne.

Les méthodes basées sur l'idée de *frontières immergées*, introduite par Peskin [Pes77] pour les membranes élastiques, proposent une alternative permettant de palier ces deux principaux problèmes. Ces méthodes sont basées sur le calcul de la dynamique du fluide dans un domaine plus large que néssaire, au delà de la fron-

tière fictive du fluide, jusqu'à l'intérieur des objets en interaction. La discrétisation de ce domaine est effectué sur une grille Eulerienne qui ne bouge pas et à l'intérieur de laquelle les objets sont plongés. Une représentation implicite de ces derniers est adoptée (par exemple à l'aide de *volume of fluids* ou *level sets*) et à la dynamique de ceux-ci est assurée à l'intérieur de ce sous-domaine du fluide via l'imposition des contraintes physiques inhérentes aux modèles choisis ; le couplage est donc directement effectué dans le fluide et retranscrit sur les objets. Bien que la non conformité du maillage avec la paroi d'un objet ne permet pas d'imposer exactement le mouvement de celui-ci à la surface, ces méthodes réussissent à capturer correctement les interactions dans la limite où le pas de grille diminue (où l'erreur de discrétisation spatiale autour de l'objet réduit). Ces méthodes permettent, plus ou moins aisément, d'introduire des objets de forme quelconque tout en bénéficiant de la rapidité de calcul des méthodes sur grille. De plus, elles sont assez naturellement adaptées pour l'introduction de membranes élastiques entourées, des deux côtés de la surface, de fluide. Les articles [Pes77, GPH+01, SP05, BIM04] sont basés sur ce principe afin de simuler l'interaction d'un fluide avec un solide rigide.

Dans ce chapitre, je présente tout d'abord plus en détails les techniques pour l'interaction fluide-structure en m'attardant plus particulièrement sur le principe de frontière immergée, car c'est sur ce dernier que j'ai axé mes travaux de recherche. Ensuite, j'exposerai les principales méthodes aujourd'hui utilisées en synthèse d'images pour les interactions fluides-structures.

2 Structures immergées

Cette partie est consacrée aux travaux existants en mathématiques appliquées concernant les interactions fluides-structures pour le cas de solides rigides et de membranes élastiques. Tout d'abord j'exposerai les méthodes permettant d'intégrer dans un fluide une structure à mouvement imposé, prédéterminé. Puis je présenterai les méthodes permettant le couplage d'un fluide avec une membrane élastique. Je terminerai en détaillant le principe de *domaine fictif* principalement utilisé pour l'interaction complète de solides rigides plongés dans un fluide.

2.1 Solide rigide à mouvement imposé

Je présente ici trois méthodes permettant l'intégration d'un solide rigide, à l'intérieur d'un fluide, dont la position et la vitesse aux bords sont connus à chaque instant. Le problème se résume donc à imposer au fluide la vitesse du solide au niveau de l'interface afin d'imposer une condition de non glissement. Je m'intéresse particulièrement à trois méthodes : la première est la méthode des *panneaux* utilisée en formulation vortex, la deuxième nommée *frontière élargie*, en formulation vitesse, est basée sur un agrandissement du domaine du solide chevauchant celui du fluide, et finalement la troisième est la méthode de *pénalisation* où le fluide incorpore le

domaine du solide dans lequel la vitesse est pénalisée.

Méthodes des panneaux et flux de vorticité

Comme nous l'avons vu au chapitre premier, imposer des conditions sur la vitesse formulation vortex est un problème difficile. Il est d'autant plus difficile en présence d'un solide rigide soumis aux forces hydrodynamiques. La complexité du problème provient du fait que la condition imposée sur la vitesse ne permet pas d'imposer directement une condition sur le champ de vorticité, aussi, il faut ajouter une condition aux bords pour la vorticité (question adressée dans le premier chapitre).

D'une manière générale, la vitesse du fluide peut s'écrire, par la décomposition d'Helmholtz comme la somme d'un champ à divergence nulle ($\mathbf{u}_\omega$) et d'un champ irrotationnel ($\mathbf{u}_\Phi$) :

$$\mathbf{u} = \mathbf{u}_\omega + \mathbf{u}_\Phi. \tag{3.1}$$

La vorticité est définie comme

$$\omega = \nabla \times \mathbf{u} = \nabla \times \mathbf{u}_\omega \tag{3.2}$$

et la divergence de $\mathbf{u}_\omega$ est nulle :

$$\nabla \cdot \mathbf{u}_\omega = 0. \tag{3.3}$$

Des équations 3.2 et 3.3, nous pouvons déduire une équation de Poisson sur la vorticité :

$$\Delta \mathbf{u}_\omega = -\nabla \times \omega. \tag{3.4}$$

Comme nous l'avons vu dans le premier chapitre, une solution pour ce problème peut être obtenue par la loi de Biot-Savart (*cf.* équation 1.18). La vitesse provenant de l'aspect rotationnel du fluide est donc obtenue, reste à définir la deuxième composante irrotationnelle, qui peut s'écrire comme :

$$\mathbf{u}_\Phi = \nabla \Phi. \tag{3.5}$$

En combinant cette dernière équation à la contrainte d'incompressibilité, nous obtenons une équation de Laplace pour Φ :

$$\Delta \Phi = 0. \tag{3.6}$$

Clairement, la décomposition 3.1 n'est pas unique. Une solution particulière peut être obtenue en imposant des conditions aux bords sur les vitesses. Par exemple, afin de prendre en compte une condition de non pénétration sur la surface d'un solide, il convient de poser :

$$\mathbf{u} \cdot \mathbf{n} = (\mathbf{u}_\omega + \nabla \Phi) \cdot \mathbf{n} = \mathbf{u}_b \cdot \mathbf{n}, \tag{3.7}$$

où $\mathbf{u}_b$ est la vitesse connue du solide. L'équation 3.7 peut se reécrire :

$$\nabla \Phi \cdot \mathbf{n} = (\mathbf{u}_b - \mathbf{u}_\omega) \cdot \mathbf{n}, \tag{3.8}$$

qui fournit la condition aux bords nécessaire pour résoudre 3.6. Ainsi, la vitesse du fluide définie par le champ de vorticité et dont la composante normale est nulle aux bords du domaine s'écrit :

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_\infty + \int_{\mathbb{R}^d} \mathbf{K}(\mathbf{x}-\mathbf{y}) \times \omega(\mathbf{y})\, dy + \nabla\Phi. \tag{3.9}$$

La résolution de l'équation de Laplace 3.6 peut être effectuée sur une grille de manière simple lorsque le maillage est adapté aux frontières du fluide. Néanmoins, cette discrétisation de l'espace devient très rapidement complexe à mettre en oeuvre en 3D et en présence de parois à géométrie quelconque.

La méthode des panneaux [Hes90] (*cf.* [CK00] pour une description détaillée des techniques existantes) permet d'imposer une condition de non pénétration à l'interface entre une paroi fixe et le fluide, dans une approche différente. Dans les articles [LK93, PWS+02], les auteurs ont appliqué la méthode avec succès. Le principe de la technique est le suivant : considérer une vitesse de pénétration nulle aux bords peut se traduire par l'apparition d'une fine couche de vorticité (notée γ) sur la surface, en fonction de la vitesse tangentielle relative à l'intérieur et à l'extérieur de l'obstacle, soit :

$$\gamma(\mathbf{x}) = \mathbf{n}(\mathbf{x}) \times [\mathbf{u}(\mathbf{x}_i) - \mathbf{u}(\mathbf{x}_e)],$$

où $\mathbf{x}_i$ (resp. $\mathbf{x}_e$) représente le point $\mathbf{x}$ à l'intérieur (resp. extérieur) du solide. La technique consiste à créer un ensemble de panneaux, représentant de manière discrète la géométrie de la frontière, portant une vorticité γ (constante ou linéaire sur la longueur du panneau, suivant la précision requise) dont l'intensité va imposer l'annulation de la vitesse normale au niveau du panneau, et par extension à l'ensemble de la surface. Cette feuille de vorticité est ajoutée sous une forme intégrale dans l'équation de Biot-Savart permettant le calcul de la vitesse :

$$\mathbf{u}(\mathbf{x},t) = \int_\Omega \mathbf{K}(\mathbf{x}-\mathbf{y}) \times \omega(\mathbf{y},t)\, d\mathbf{y} + \int_\Gamma \mathbf{K}(\mathbf{x}-\mathbf{y}) \times \gamma(\mathbf{y},t)\, d\mathbf{y}$$

où Ω est le domaine contenant le fluide et Γ la surface du solide représentée par les panneaux. L'intensité de chaque panneau doit être évaluée pour chaque temps t de manière à annuler la vitesse normale aux bords. Ce calcul nécessite l'inversion d'une matrice A suivant l'équation

$$A\gamma = \mathbf{b}$$

où $\mathbf{b}$ est la vitesse normale en chacun des panneaux à l'instant t. Si les panneaux ne bougent pas, la matrice A est constante en temps et il n'est donc pas nécessaire de l'inverser qu'une seule fois initialement.

Cette méthode, particulièrement adaptée à la formulation Lagrangienne de l'équation de vorticité, peut être remplacée par une technique plus simple dans le cas d'une méthode hybride où la vitesse est calculée via la fonction courant $\mathbf{\Psi}$. En effet, une vitesse nulle aux bords d'un obstacle se traduit par une condition sur $\mathbf{\Psi}$:

$$\mathbf{u} \cdot \mathbf{n} = 0 = (\nabla \times \mathbf{\Psi}) \cdot \mathbf{n} \ \text{ sur } \Gamma$$

où $\mathbf{n}$ est la normale à la surface. Comme la fonction courant peut être choisie à une constante près (du fait que l'on en prend le rotationnel), une condition de Dirichlet $\mathbf{\Psi} = cste$ sur le bord permet d'annuler simplement la vitesse normale. Cette condition doit être complétée d'une condition de Neumann $\frac{d\mathbf{\Psi}}{d\mathbf{n}} = 0$ afin de conserver la propriété $\nabla \cdot \mathbf{\Psi} = 0$. Dans le cas où on utilise une grille régulière pour discrétiser l'espace dans lequel est plongé un solide non aligné sur les points de grille, imposer cette condition s'avère non trivial. Cette méthode permet néanmoins de ne pas calculer le potentiel Φ afin d'annuler la vitesse de pénétration aux bords ; cette condition est assurée lors de la résolution de $\mathbf{\Psi}$.

En présence d'un fluide visqueux, la vitesse du fluide au niveau de la surface d'un solide doit être identique à celle de la paroi. Le problème est donc d'annuler (si on se place dans le cas d'une paroi fixe) la vitesse tangentielle du fluide en sus de la vitesse normale. Physiquement, ceci peut être interprété comme l'apparition d'une couche de vorticité (dont l'épaisseur est fonction de la viscosité) dans le fluide proche du bord, permettant de contrebalancer le glissement du fluide. La méthode dite de *flux de vorticité* permet, par la diffusion de vorticité depuis la paroi vers l'intérieur du fluide, d'annuler la vitesse tangentielle résiduelle après les étapes de transport et diffusion de l'équation de Navier-Stokes (*cf.* schéma 1.6 au premier chapitre). Pour résumer, l'algorithme est le suivant :

1. Résoudre l'équation de Navier-Stokes en formulation vorticité,
2. Calculer la vitesse de glissement aux bords,
3. Introduction d'une couche de vorticité près du bord visant à annuler le glissement.

L'intensité et la localisation de ce flux de vorticité sont calculées à partir d'une méthode similaire à celle des panneaux, en évaluant discrétement la vitesse tangentielle résiduelle en plus de la vitesse normale. Bien sûr, des conditions aux bords adéquates doivent être adoptées pour la vorticité, comme précisé dans le premier chapitre.

Cette méthode, bien qu'élégante et adaptable aux méthodes purement particulaires est coûteuse — elle requiert l'évaluation discrète du flux de vorticité au niveau de chaque panneau et ajoute un nombre important de particules, augmentant ainsi le coût du calcul de la vitesse — et demande la discrétisation explicite de la surface, ce qui peut être un problème difficile en 3D en présence d'objets à géométrie complexe. Nous verrons dans le chapitre 5 une nouvelle méthode beaucoup plus simple à implémenter, moins coûteuse et s'adaptant très naturellement à tout type de géométrie. Les bases de cette technique sont présentées dans les parties suivantes dans le cadre d'une formulation vitesse des équations de Navier-Stokes.

Frontière élargie

La méthode de frontière élargie (*fat boundary method* en anglais) introduite par Maury [Mau01] puis adaptée par Bertoluzza *et al.* [BIM04] pour la simulation de

fluide s'intéresse à l'intégration d'un solide rigide immergé dans un fluide. Via une zone tampon autour du « trou »formé par l'objet dans le maillage du fluide, sont d'abord résolues les équations sur le domaine complet, puis est traitée la correction locale nécessaire pour l'interaction des deux milieux. L'idée intuitive est d'utiliser un maillage efficace pour les calculs sur le fluide et un maillage adapté autour du solide, et ensuite de faire correspondre les deux solutions. Les deux domaines, Ω celui du fluide et B celui du solide, se superposent comme présenté sur la figure 3.1. Par exemple, le problème de la résolution de l'équation de Poisson :

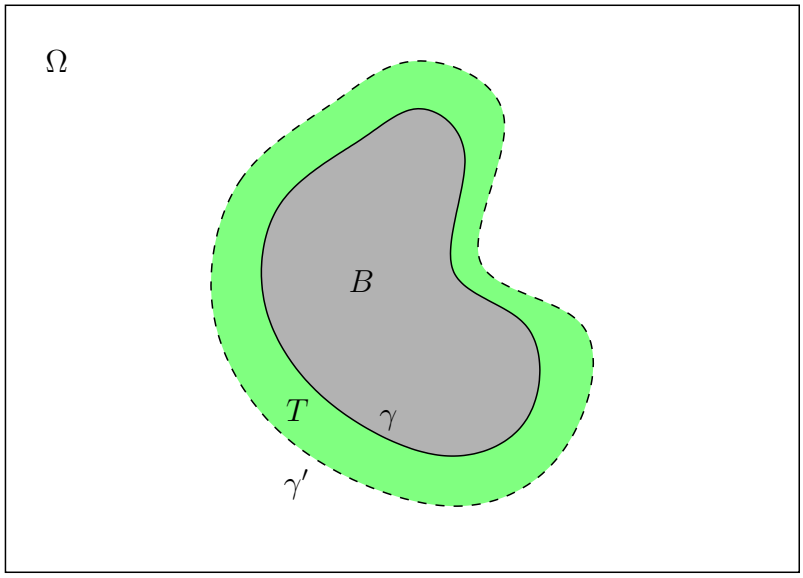


FIG. 3.1 – Le trou dans le domaine Ω représenté par B, sa surface γ, la zone « tampon »T close par la frontière γ' à l'intérieur du domaine Ω.

$$\left\{ \begin{array}{rcll} -\Delta u & = & f & \text{dans } \Omega \setminus \bar{B}, \\ u & = & 0 & \text{sur } \gamma, \end{array} \right. \tag{3.10}$$

(où $\gamma = \partial B$) avec $u = g$ sur $\partial\Omega$, se ramène à résoudre les deux problèmes suivants :

$$\left\{ \begin{array}{ll} a) & \left\{ \begin{array}{rcll} -\Delta v & = & f & \text{dans } T \\ v & = & \hat{u} & \text{sur } \gamma' \end{array} \right. \\ b) & -\Delta \hat{u} = \bar{f} + \frac{\partial v}{\partial n}\delta_\gamma \ \text{ dans } \Omega \end{array} \right. \tag{3.11}$$

où T est le sous-domaine clos par γ et γ' formant la zone tampon (*cf.* figure 3.2), où $\bar{f}$ est l'extension de f par 0 dans B et le terme $\frac{\partial v}{\partial n}\delta_\gamma$ peut être compris comme la diffusion de la condition au bord réelle sur γ vers l'intérieur du domaine $\Omega \setminus \bar{B}$. La seconde équation est complétée d'une condition au bord de Ω identique au problème précédent. Pour plus de détails sur cette méthode, la preuve de sa correction et ses applications dans le cadre d'une formulation par éléments finis, le lecteur peut se référer à la thèse d'Ismail [Ism04].

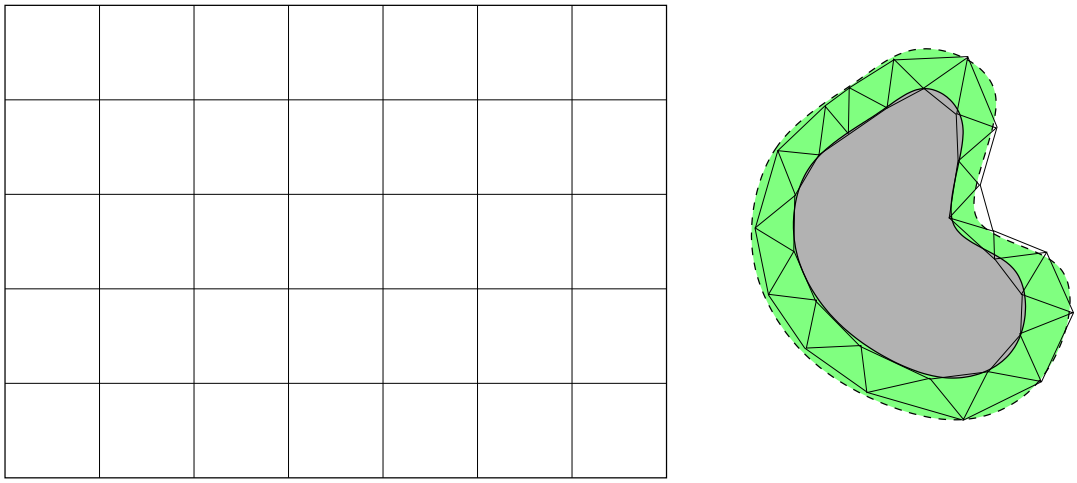


FIG. 3.2 – Exemple de discrétisation (grossière) pour les domaines Ω et T, la zone prolongée au delà de l'interface de B où sont résolus les problèmes 3.11.b et 3.11.a respectivement.

Bien qu'il y ait maintenant deux problèmes à résoudre au lieu d'un seul, l'intérêt de cette méthode réside dans le fait de pouvoir utiliser un solveur rapide sur tout le domaine pour résoudre l'équation 3.11.b) et un solveur adapté à la forme du maillage entourant B, qui dans le cas d'objets rigides ne bouge pas à la différence des méthodes ALE. Il est intéressant de voir que cette méthode pourrait servir à l'application d'une condition de non pénétration sur la vitesse en méthode vortex, via le calcul de la fonction courant, solution de l'équation de Poisson $\Delta\mathbf{\Psi} = -\omega$. Mais cette méthode semble d'ores-et-déjà plus ardue pour appliquer une condition de non glissement aux bords d'un objet, cette contrainte ne pouvant être remplie seule et directement par la résolution de l'équation sur $\mathbf{\Psi}$. Néanmoins, la technique peut faire penser à celle de flux de vorticité permettant de diffuser la vorticité dans la couche limite autour de la paroi afin d'annuler la vitesse tangentielle. Pour ma part, j'ai utilisé la méthode de pénalisation — présentée ci-dessous pour la formulation vitesse — s'intégrant plus naturellement dans les méthodes vortex de part son découplage de l'équation sur la fonction courant.

Méthode de pénalisation

La méthode de pénalisation consiste à pénaliser la vitesse du fluide à l'intérieur d'un solide afin d'imposer la vitesse rigide à l'intérieur et aux bords du sous-domaine B occupé par l'obstacle. Ce sous-domaine de Ω peut être alors perçu comme un milieux poreux dans lequel on va chercher à annuler la vitesse du fluide (relativement à la vitesse du solide), c'est-à-dire dont la perméabilité tend vers l'infini. Dans [ABF99], Angot *et al.* introduisent la méthode de pénalisation dans un fluide où les obstacles sont fixes. Deux types de pénalisation sont proposés : la pénalisation L^2 consiste à amortir la vitesse du fluide dans B ; l'équation de Navier-Stokes

pénalisée devient donc :

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \mu \Delta \mathbf{u} + \nabla p + \eta^{-1} \chi \mathbf{u} = \mathbf{f} \tag{3.12}$$

où χ est la fonction caractéristique du solide :

$$\begin{aligned} \chi(\mathbf{x}) &= 1 \quad \text{dans } B, \\ \chi(\mathbf{x}) &= 0 \quad \text{dans } \Omega. \end{aligned}$$

La méthode H^1 quant à elle revient à pénaliser tous les termes linéaires dans l'équation :

$$(1 + \eta^{-1}\chi)\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - (1 + \eta^{-1}\chi)\, \nabla \cdot \sigma + \eta^{-1}\chi \mathbf{u} = \mathbf{f} \tag{3.13}$$

avec $\sigma = \mu \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T \right) - p\mathbf{I}$ le tenseur des contraintes ($\mathbf{I}$ étant la matrice identité). Le paramètre de pénalisation η contrôle la perméabilité de l'obstacle, et lorsque $\eta \to 0$, la vitesse du fluide dans B coincide avec celle (ici nulle) du solide. Une analyse numérique de cette méthode est fournie dans l'article [SVAC08] pour des solides fixes. Dans le même principe, l'article [JLM05] propose une méthode, basée sur les élements finis, permettant d'imposer la contrainte de rigidité via le tenseur de déformation $\mathbf{D}(\mathbf{u}) = \frac{\nabla \mathbf{u} + \nabla \mathbf{u}^T}{2}$. La solution recherchée est celle minimisant la déformation totale $\eta^{-1} \int_B \mathbf{D}(\mathbf{u})$, avec $\eta << 1$.

Je présenterai plus précisément cette méthode au chapitre 5 car c'est elle sur laquelle je me suis basé pour mes travaux sur l'interaction de solides rigides avec des fluides.

2.2 Membranes élastiques

Peskin est le précurseur des méthodes de frontières immergées, avec initialement des applications aux membranes élastiques dans le cadre de simulations de voies sanguines et du coeur [Pes77, PP93] (et les références dans celles-ci), et repris par Goldstein *et al.* [GHS93] (et les travaux postérieurs [GHS95, SB96]) pour des parois rigides — que je n'ai pas détaillé précédemment, mais qui reste dans la même formulation qui suit. Ces membranes souples possèdent leur propre dynamique, modélisée par des lois d'élasticité sur leur surface, en réponse à l'étirement, déformation tangentielle, et à la déformation normale, en rapport avec la courbure de la surface. De plus, on s'intéresse au cas d'une membrane imperméable, ce qui se traduit, dans le cas d'un fluide visqueux, par :

$$[\mathbf{u}] = 0 \text{ sur } \Gamma$$

où Γ est la surface et $[\mathbf{u}]$ note le saut de vitesse au travers de celle-ci. Ainsi, à la manière des fluides multiphases, la membranes est simplement transportée par le fluide, et pour les points $\mathbf{X}$ sur celle-ci, l'équation de transport est appliquée :

$$\frac{\partial \mathbf{X}}{\partial t} = \mathbf{u}(\mathbf{X}).$$

Dans les travaux de Peskin, la surface est paramétrée, représentée de manière Lagrangienne (avec tous les avantages et inconvénients discutés dans les chapitres précédents). Pour simplifier la notation, intéressons nous à l'écoulement d'un fluide 2D dans lequel est plongée une courbe fermée paramétrée par $s \in [0, L]$ au cours du temps : $\mathbf{X}(s,t)$.

Contrairement aux solides rigides étudiés ci-dessus, la membrane n'est pas simplement soumise aux forces hydrodynamiques, elle possède aussi son propre comportement élastique. Peskin propose de considérer les interactions entre ces deux milieux en intégrant les forces élastiques présentes sur la surface à l'intérieur du fluide. Est alors ajoutée à l'équation de conservation du mouvement du fluide un terme de forçage localisé sur l'interface. L'ensemble des équations à résoudre est donc :

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) + \nabla p = \mu\Delta\mathbf{u} + \mathbf{f}, \tag{3.14}$$

$$\nabla\cdot\mathbf{u} = 0, \tag{3.15}$$

$$\mathbf{f}(\mathbf{x},t) = \int_0^L \mathbf{F}(s,t)\delta(\mathbf{x}-\mathbf{X}(s,t))\,ds, \tag{3.16}$$

$$\frac{\partial \mathbf{X}(s,t)}{\partial t} = \mathbf{u}(\mathbf{X}(s,t),t) = \int_\Omega \mathbf{u}(\mathbf{x},t)\delta(\mathbf{x}-\mathbf{X}(s,t))\,d\mathbf{x}, \tag{3.17}$$

$$\mathbf{F}(s,t) = \mathbf{S}(\mathbf{X}(\cdot,t),t). \tag{3.18}$$

La force $\mathbf{F}$ est définie par la fonction $\mathbf{S}$ déterminant le comportement élastique de la membrane (le signe $\cdot$ est présent pour prendre en compte de manière générale toute loi de comportement, locale ou non). La masse de Dirac δ localise la force mais aussi définie la vitesse de l'interface en fonction de la vitesse du fluide, car numériquement la masse de Dirac est régularisée par un paramètre ε. Asymptotiquement — lorsque $\varepsilon \to 0$ — au niveau de l'interface, le fluide se comportera alors comme la membrane élastique. Le couplage de toutes ces équations permet effectivement de prendre en compte toutes les interactions entre les différents milieux.

A partir de cette même formulation, Lai et Peskin [LP00] proposent une intégration numérique d'ordre 2 — formellement, car en fait la discrétisation de la fonction Dirac lisse irrémédiablement le champ de vitesse autour de l'interface (*cf.* [BL92] pour plus de détails) — pour résoudre les équations, afin de réduire la viscosité numérique introduite par la fonction Dirac régularisée. Les résultats obtenus avec cette technique ont même permis d'obtenir des résultats intéressants pour l'étude de l'écoulement autour d'un cylindre fixe, via une force élastique très raide permettant d'éviter la déformation du cylindre et ainsi de simuler le comportement d'un solide rigide contenu par la membrane.

Dans une approche similaire, les travaux de Cottet *et al.* [CM06b, CMM08] sont axés sur la représentation de la membrane par une surface implicite, définie par une level set ϕ (*cf.* partie 2) transportée par le fluide. La surface Γ est alors définie comme :

$$\Gamma = \{\mathbf{x} \mid \phi(\mathbf{x}) = 0\}.$$

La force élastique est alors extraite de cette formulation implicite, via le gradient $\nabla\phi$ et l'étirement de la mebrane $|\nabla\phi|$. Si initialement la level set est une fonction distance, à l'interface, $|\nabla\phi| - 1$ dénote le changement d'air de la surface [CM04]. Ainsi, une énergie issue de la membrane élastique peut être dérivée :

$$\xi = \int_\Gamma E\left(|\nabla\phi|\right)\, d\mathbf{x} = \int_\Omega E\left(|\nabla\phi|\right)\delta(\mathbf{x})\, d\mathbf{x}$$

où E est une loi élastique fonction de l'étirement de la membrane $|\nabla\phi|$; et de manière discrète avec un paramètre de régularisation ε pour une fonction *cut-off* ζ :

$$\xi = \int_\Omega E\left(|\nabla\phi|\right)\frac{1}{\varepsilon}\zeta\left(\frac{\phi}{\varepsilon}\right)\, d\mathbf{x}.$$

De cette énergie est extraite la force qui est introduite dans l'équation de Navier-Stokes comme terme de forçage, similairement à [Pes77] :

$$\mathbf{f} = \left\{\mathbb{P}_{\nabla\phi^\perp}\left(\nabla E'(|\nabla\phi|)\right) - E'(|\nabla\phi|)\kappa(\phi)\mathbf{n}(\phi)\right\}|\nabla\phi|\frac{1}{\varepsilon}\zeta\left(\frac{\phi}{\varepsilon}\right) \tag{3.19}$$

où $\mathbb{P}_{\nabla\phi^\perp}$ est la projection sur le plan orthogonal à $\nabla\phi$, et $\kappa(\phi)$ et $\mathbf{n}(\phi)$ la courbure et la normale (respectivement) de la surface définies en formulation level set, dans le voisinage de $\phi = 0$. Dans cette équation, on peut remarquer que la force est séparée en une composante tangentielle (à gauche) et une normale (à droite) portées sur l'interface. On peut noter que dans le cas où E est une loi linéaire (E' est alors constant et $\nabla E'$ est nul), l'équation 3.19 est similaire à la tension de surface. Pour une force élastique tendant à diminuer l'étirement, les auteurs choisissent $E'(|\nabla\phi|) = \lambda(|\nabla\phi| - 1)$.

Cette formulation Eulerienne de la membrane permet d'éviter les difficultés inhérentes à la représentation Lagrangienne, plus précisément la nécessité de remaillage de la surface paramétrée dans le cas de grandes déformations et la généralisation simple en dimension supérieure à 2. Par contre, même si d'un point de vue continu la force élastique empêche tout changement de topologie, en pratique la discrétisation des équations ne permet pas d'assurer cette propriété, ce qui peut être indésirable dans le cas d'une membrane élastique. Les résultats obtenus avec cette méthode sont concluants sur des cas tests et les auteurs fournissent des applications pour la simulation de globules rouges, considérés comme des membranes élastiques contenant un fluide.

2.3 Solide rigide : couplage fluide-structure en formulation vitesse

Intéressons nous maintenant à l'interaction totale d'un solide rigide soumis aux forces hydrodynamiques d'un fluide. Le principe des méthodes basées sur la technique de *frontière immergée* est de plonger la structure interagissant avec le fluide dans ce dernier sans en modifier le maillage. Ainsi, le domaine du fluide est élargi

pour couvrir entièrement l'objet, permettant l'utilisation des méthodes traditionnelles rapides et adaptées (par exemple pour les solveurs d'équation de Poisson sur grille régulière) comme en présence d'une fluide unique. Le principe initialement proposé par Peskin [Pes77] concernait l'interaction d'un fluide avec une membrane élastique.

Basée sur la même idée, la méthode dite de *domaine fictif* introduite par Glowinski *et al.* (dans la suite d'articles [GPH+01, GPH+99, PJB+02]) puis améliorée par Sharma et Patankar [SP05, Pat01] permet l'interaction du fluide avec un solide rigide soumis aux forces hydrodynamiques. L'utilisation de multiplicateurs de Lagrange forçant la contrainte de rigidité à l'intérieur du sous-domaine du fluide occupé par le solide permet d'imposer la condition de Dirichlet sur la vitesse aux bords du solide (et par extension à l'intérieur). L'article [SP05] propose une amélioration de la méthode, au niveau du temps de calcul et de la simplicité, dans le sens où imposer la contrainte de rigidité dans le solide peut alors être perçu comme une projection du champ de vitesse à la méthode de Chorin pour la contrainte d'incompressibilité. Le problème est réduit au calcul de trois multiplicateurs de Lagrange (notés par le vecteur λ) issus de la contrainte de rigidité à l'intérieur du domaine $P(t)$ du solide :

$$\mathbf{D}[\mathbf{u}] = \frac{1}{2}(\nabla\mathbf{u} + \nabla\mathbf{u}^T) = 0 \tag{3.20}$$

à laquelle s'ajoute la condition d'incompressibilité et de condition aux bords $\partial P(t)$ sur la vitesse : $\mathbf{u} = \mathbf{u}_r$ où l'indice r dénote la grandeur physique dans le solide. En réécrivant la contrainte de rigidité 3.20, nous pouvons obtenir les trois multiplicateurs de Lagrange soumis aux trois équations :

$$\nabla \cdot (\mathbf{D}[\mathbf{u}]) = \nabla \cdot (\mathbf{D}[\lambda]) = 0 \quad \text{dans } P(t), \tag{3.21}$$

$$\mathbf{D}[\mathbf{u}] \cdot \mathbf{n} = \mathbf{D}[\lambda] \cdot \mathbf{n} = 0 \quad \text{sur } \partial P(t). \tag{3.22}$$

Afin d'intégrer la contrainte dans le fluide, on réécrit l'équation de Navier-Stokes sous la forme :

$$\rho\left(\frac{\partial\,\mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) = -\nabla p + \nu\Delta\mathbf{u} + \rho\mathbf{g} + \mathbf{f}$$

complétée de l'équation d'incompressibilité et où $\rho(\mathbf{x})$ varie en fonction de la densité du fluide et du solide suivant $\mathbf{x}$. $\mathbf{f}$ est définie comme $\nabla \cdot (\mathbf{D}[\lambda])$ pour être consistant avec l'équation de mouvement dans le fluide et dans le solide (*cf.* [SP05] pour plus de détails). Tout d'abord l'équation est résolue pour l'itération n, sans le terme $\mathbf{f}$, avec une méthode traditionnelle pour obtenir $\hat{\mathbf{u}}$ à divergence nulle. Puis, similairement à la méthode de projection, on définit la vitesse au pas de temps suivant comme :

$$\rho\left(\frac{\mathbf{u}^{n+1} - \hat{\mathbf{u}}}{\delta t}\right) = \mathbf{f} \;\text{ dans } P(t). \tag{3.23}$$

Les équations 3.21 et 3.22 sont alors écrites pour $\mathbf{u}^{n+1}$ en injectant l'équation précédente, ce qui fournit les équations recherchées pour $\mathbf{f}$. Une fois $\mathbf{f}$ calculé, $\mathbf{u}^{n+1}$ est

recouvré grâce à l'équation 3.23, où les contraintes d'incompressibilité et de rigidité (à l'intérieur du solide) sont assurées. La vitesse à l'intérieur du solide satisfait ces propriétés mais ne représente pas particulièrement la vitesse du solide soumis aux forces hydrodynamiques. C'est pourquoi il faut introduire les lois de comportement du solide pour la résolution sur $\mathbf{f}$ en définissant $\hat{\mathbf{u}} = \mathbf{u}_r + \mathbf{u}'$ où $\mathbf{u}_r$ est la vitesse du solide :

$$\begin{aligned} \mathbf{u}_r &= \mathbf{U} + \mathbf{\Omega} \times \mathbf{r}, \\ M\mathbf{U} &= \int_{P(t)} \rho \hat{\mathbf{u}} \, d\mathbf{x} \\ \mathbf{I}\mathbf{\Omega} &= \int_{P(t)} \rho \mathbf{r} \times \hat{\mathbf{u}} \, d\mathbf{x} \end{aligned}$$

où M est la masse du solide, $\mathbf{U}$ sa vitesse de translation, $\mathbf{\Omega}$ sa vitesse angulaire, $\mathbf{I}$ la matrice d'inertie et $\mathbf{r}$ le vecteur depuis le centre de gravité du solide. Ainsi, la vitesse $\mathbf{u}^{n+1} = \mathbf{u}_r$ est bien la vitesse du solide soumis aux forces exercées par le fluide et les forces extérieures, dans le sous domaine $P(t)$. Les vitesses de translation et de rotation du solide ainsi calculées permettent de faire avancer ce dernier vers le pas de temps suivant.

Cette méthode est particulièrement adaptée à la formulation vitesse car elle consiste à projeter celle-ci sur l'espace des vitesses satisfaisant les contraintes du solide. Malheureusement ce principe se prête beaucoup plus difficilement aux méthodes vortex où la quantité manipulée est la vorticité et non directement la vitesse.

3 Interactions fluide-structure en synthèse d'images

Face à la difficulté et au coût des méthodes d'interaction entre fluides et solides rigides ou déformables, la communauté de la synthèse d'images s'est longtemps limitée à la simulation de fluides contraints par des solides rigides fixes, globalement facilement intégrables dans la résolution des équations en imposant le non-glissement sur les faces d'un maillage Eulerien. Dans ce contexte, de manière plus générale, les parois d'un solide fixe peuvent être discrétisées pour se conformer à la grille, posant néanmoins le problème « d'escalier »(ou alliassage) induit par cette approximation. Plus récemment, les méthodes de type ALE ont été adaptées à l'animation [FOC+05,KFCO06] pour l'interaction d'objets rigides ou déformables avec un fluide. La méthode de domaine fictif quant à elle a été reprise dans l'article *Rigid Fluids* [CMT04] pour l'intégration de solides rigides dans une formulation Eulerienne standard. Dans un modèle plus microscopique pour l'interaction, Genevaux [Gén06] utilise des ressorts connectés entre la paroi et le fluide pour empêcher ce dernier de pénétrer le solide tout en appliquant les forces hydrodynamiques. Dans le cadre des méthodes vortex, [PK05] proposent d'utiliser la méthode des panneaux afin d'imposer des conditions de non glissement dans une formulation purement Lagrangienne.

D'autres techniques moins basées sur la physique ont été introduites afin de contrer les difficultés et de satisfaire les contraintes apparaissant en synthèse d'images

— le temps de calcul, la « manipulation »de la simulation par un artiste, l'interaction avec des objets aux comportements très divers — comme par exemple les *thin deformable and rigid shells* [GSLF05]. L'étude détaillée de ces méthodes sortant du cadre de cette thèse, elles ne seront pas présentées dans ce document. C'est pourquoi je me tiendrai aux méthodes annoncées plus haut, dans l'ordre de leur énonciation.

3.1 Méthode ALE et maillages dynamiques

J'ai présenté dans le premier chapitre la méthode de type ALE proposée dans les articles [FOC+05, KFCO06]. Dans cette formulation, la gestion de l'interaction entre le fluide et un solide (la généralisation à plusieurs solides est simple) est gérée via la résolution de l'équation de Poisson pour la projection de la vitesse partielle sur l'espace des vitesses à divergence nulle (*cf.* 4.3). Un ensemble de n faces du maillage pour le fluide forment l'interface avec le solide. La matrice $\mathbf{R}$ de taille $n \times n$ (détaillée ci-après) multipliée par le vecteur des pressions aux points associés fournit l'accélération totale du solide.

$$\mathbf{R} = \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_n \end{bmatrix} \begin{bmatrix} \mathbf{M}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}^{-1} \end{bmatrix} \begin{bmatrix} A_1 {b_1}^T & \cdots & A_n {b_n}^T \end{bmatrix}$$

avec $\mathbf{b}_i = \left[{\mathbf{n}_i}^T \mid (\mathbf{r}_i \times \mathbf{n}_i)^T \right]$ le vecteur contenant la normale et le vecteur orthogonal à $\mathbf{n}_i$ et au vecteur $\mathbf{r}_i$ en direction du centre de gravité du solide au niveau de la face f_i, la matrice $\mathbf{M}$ est la matrice identité multipliée par la masse du solide, I_d, $\mathbf{J}$ la matrice d'inertie et A_i l'air de la face f_i. Afin de trouver la pression, l'opérateur discret modifié $\mathbf{D}$ correspondant à la divergence est appliqué à l'équation de Poisson pour trouver la pression :

$$\delta t \mathbf{DAp} = -\mathbf{Du}^*$$

où $\mathbf{u}^*$ est la vitesse partielle provenant de l'ajout des forces extérieures à la vitesse obtenue par la méthode semi-Lagrangienne, et $\mathbf{A}$ la matrice formée par l'opérateur discret gradient multiplié par $-\rho^{-1}$ pour les faces à l'intérieur du fluide et formée de la ligne correpondante de la matrice $\mathbf{R}$ pour les faces aux bords du solide. A la différence des méthodes de domaine fictif, celle-ci possède l'intérêt d'obtenir l'interaction entre le fluide et les structures en une seule passe lors de la projection, elle a été utilisée pour le couplage de parois déformables agissant sur un fluide. La difficulté majeure de cette méthode, comme je l'ai précisé dans le premier chapitre, est la génération d'un maillage cohérent épousant les bords du domaine du mieux possible, problème inexistant dans les techniques basées sur des frontières immergées.

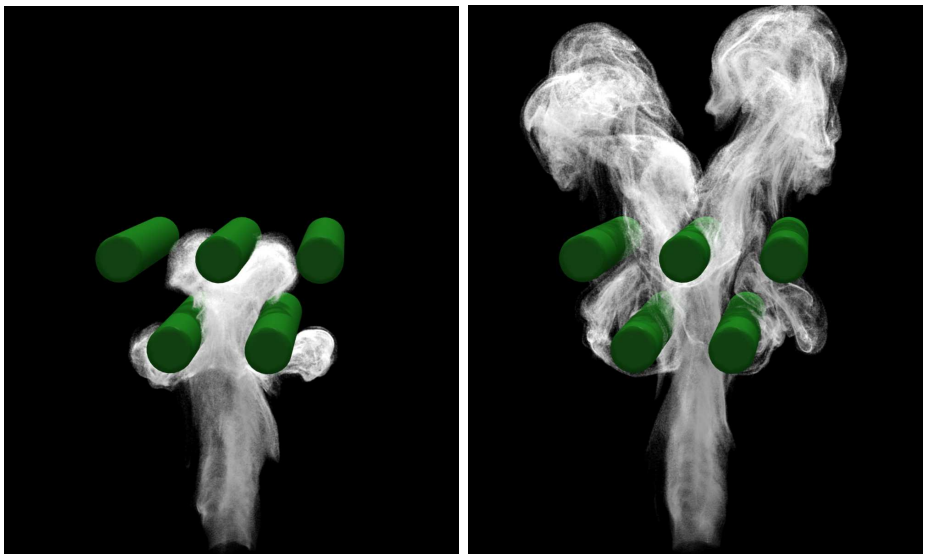


FIG. 3.3 – Interactions fluide-structure par la méthode hybride proposée dans [KFCO06]. Calculs haute résolution. Images extraites de [KFCO06].

3.2 Rigid Fluids

L'article *Rigid Fluids* [CMT04] reprend la méthode de domaine fictif introduite par Glowinski *et al.* et améliorée par la suite par Patankar *et al.* dans le cadre d'une simulation par différences finies sur une grille Eulerienne. La méthode est utilisée telle quelle, dans le cadre de simulations d'interactions entre liquides, gaz et solides, avec une technique de type *volume of fluids* à une seule phase (avec extrapolation de la vitesse au delà de l'interface liquide-gaz), comme présenté précédemment. Cet article est parmi les premiers à proposer des interactions efficaces entre fluides et solides rigides. La méthode demande néanmoins la résolution de système d'équations sur les quatre multiplicateurs de Lagrange (un pour la pression et trois pour la contrainte de rigidité), alors que les méthodes pénalisant la vitesse à l'intérieur du solide (comme vu précédemment) nécessitent un calcul de coût linéaire à chaque pas de temps (voir chapitre 5).

FIG. 3.4 – Interactions liquide-solide rigide par la méthode de domaine fictif, en présence de plusieurs solides et d'une surface libre. Images extraites de [CMT04].

3.3 Solides déformables

Dans l'article [GHD03] (et pour plus de détails, consulter la thèse [Gén06]), Genevaux *et al.* proposent — sur une formulation vitesse résolue par la méthode des *Markers And Cells* — d'intégrer l'interaction entre un liquide et des solides déformables par l'utilisation de forces de répulsions (similaires à un réseau masses-ressorts) entre des marqueurs situés dans le fluide et les points de discrétisation situés à la surface du solide. Ces forces tendent à repousser le fluide de la paroi via le transfert de celles-ci comme force extérieure en partie droite de l'équation de Navier-Stokes. Symétriquement, les forces exercées à l'interface entre les deux milieux sont répercutées à l'intérieur du solide via sa modélisation par un réseau de masses-ressorts, entraînant le couplage complet des deux milieux. Bien qu'adoptant un point de vue simpliste pour la dynamique du solide et son interaction avec le fluide, ces travaux précurseurs ont permi d'obtenir des résultats intéressants tout en permettant une intégration aisée et rapide de solides déformables dans un modèle de simulation de fluides existant. Néanmoins, cette solution ne permet pas de résoudre rigoureusement le problème d'interaction entre deux milieux à la dynamique différente. Les interactions considérées sont calculées entre chaque paire de points (entre les marqueurs et les sommets du solide), ce qui demande un temps de calcul important, bien qu'il puisse être réduit en considérant la localité des interactions.

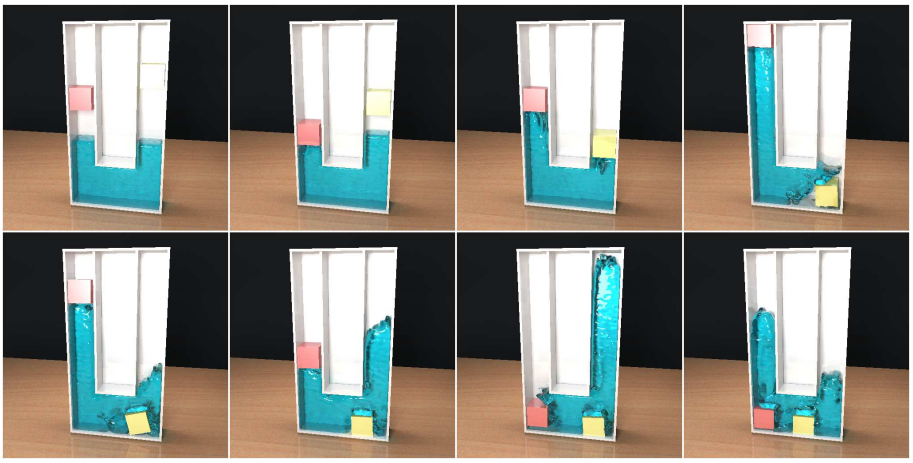


FIG. 3.5 – Interactions fluide-structure par une méthode de type masse-ressort. Images extraites de [GHD03].

3.4 Vortex : méthode des panneaux

Récemment, en 2005, Park et Kim [PK05] ont adaptés à la synthèse d'images la méthode des panneaux pour l'interaction de solides rigides avec un gaz, en adoptant une formulation vorticité des équations de Navier-Stokes, avec l'aide de particules de vorticité. Afin de pouvoir contrôler l'épaisseur de la couche limite, dans l'optique de la manipulation du logiciel de simulation par un artiste désireux de modifier le

comportement du fluide, les auteurs introduisent un paramètre $\sigma \in [0, 1]$ modulant l'intensité du flux de vorticité depuis la paroi en direction du fluide. Cet outil permet alors de maîtriser localement le taux de frottements entre le solide et le fluide, ceci peut-être compris comme une modification locale de la viscosité du fluide. Ainsi, pour des valeurs de σ proches de 0 pour des surfaces « lisses », le fluide va « glisser »contre la paroi, alors qu'avec des valeurs de σ proches de 1, la couche limite dynamique et visuelle paraîtra plus épaisse. Les auteurs effectuent au préalable le calcul de l'inverse de la matrice $\mathbf{A}$ provenant de l'équation $\mathbf{A}\gamma = \mathbf{b}$ permettant d'obtenir l'intensité de la vorticité γ au niveau d'un panneau pour la condition de non pénétration. En effet, $\mathbf{A}$ ne varie pas, même après transformation rigide (translation et/ou rotation) de la surface de l'objet, ce qui permet de considérer un solide suivant un mouvement prédéterminé.

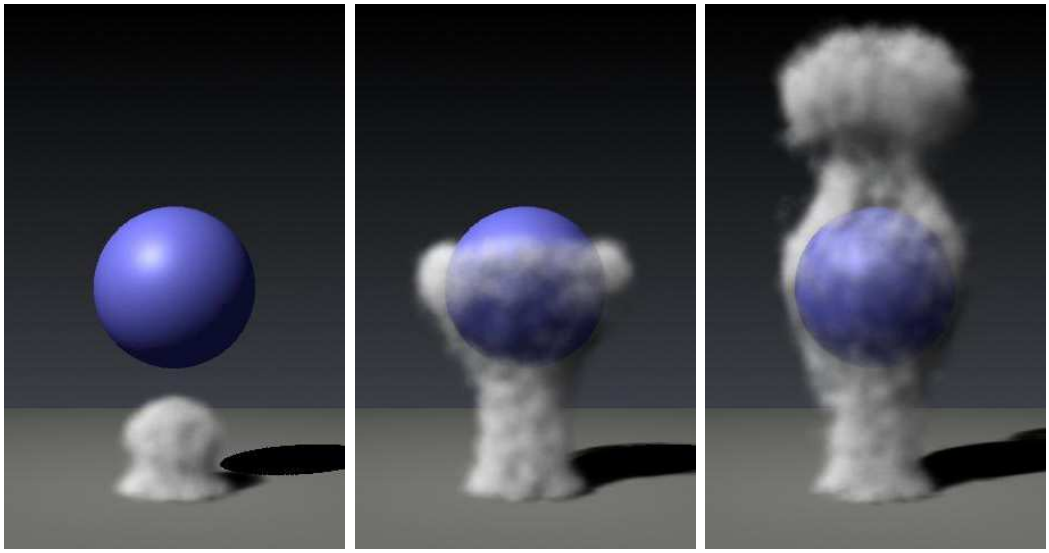


FIG. 3.6 – Interaction fluide-structure par la méthode des panneaux, par une résolution de l'équation de Navier-Stokes par particules de vorticité. Images extraites de [PK05].

4 Bilan du chapitre

Nous venons de voir les principales méthodes existantes pour l'interaction de structures, solides rigides ou déformables et membranes élastiques, en mathématiques appliquées et leur pendant en synthèse d'images. Je me suis particulièrement intéressé aux méthode de type *domaine fictif* qui présentent pour moi un avantage significatif : celui de permettre d'intégrer très simplement tout type de maillage à l'intérieur d'un fluide dans lequel les calculs peuvent être effectués rapidement, par des méthodes sur grille traditionnelles par exemple. Ceci est un avantage d'autant plus grand qu'en méthode vortex, imposer une contrainte aux bords d'un solide peut

s'avérer délicat, et encore plus lorsque sa forme est quelconque — ce qui est bien souvent le cas en synthèse d'images — où la méthode des panneaux serait difficile à mettre en oeuvre.

Nous verrons dans le chapitre 5 une technique permettant l'interaction entre fluides et solides rigides grâce à une méthode de pénalisation s'intégrant aisément dans une formulation vorticité. Mais avant cela, dans le chapitre suivant, je présente les fondements nécessaires pour la simulation de fluides par méthode vortex, sur lequels seront ajoutées les interactions fluide-structure, dont des membranes élastiques, avec des applications en synthèse d'images.

Troisième partie

Méthode de vorticité pour le couplage fluide-structure

Chapitre 4

Méthode de vorticité pour la simulation de fluides

1 Fluide monophasique

1.1 Préambule

Je présente en premier lieu mes travaux effectués sur les fluides à une unique phase, le cas le plus simple. Le passage à plusieurs phases est présenté dans la section suivante 2. Tout d'abord j'effectue un rappel des équations de Navier-Stokes en formulation vorticité, ainsi que sur les méthodes de résolution pour ces équations. Puis je reviens en détails sur les points de calcul traités succintement dans la première partie du manuscrit, concernant le calcul de la vitesse, de la diffusion, sur les méthodes de remaillage et diverses méthodes numériques nécessaires à la résolution des équations. Je terminerai cette section en présentant un exemple de simulation de fluide monophasique avec deux méthodes différentes : la méthode purement particulaire et la méthode *Vortex In Cell* avec remaillage (notée par le sigle VIC dans la suite) qui sera la base de la suite de mes travaux.

1.2 Rappel des équations et des méthodes de résolution

Je rappelle ici les équations déjà présentées dans le chapitre premier concernant la formulation vorticité, en y spécifiant les particularités associées à mon travail de recherche. Préalablement, il est important de noter que dans le cas particulier d'un fluide monophasique, aucune équation sur la densité n'est résolue. Ceci est la conséquence du fait que dans l'équation de conservation de la vorticité 1.14 :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = (\omega \cdot \nabla)\mathbf{u} + \nu \Delta \omega,$$

seule la viscosité ν entre dans l'équation, et cette viscosité étant constante au sein du fluide, aucune équation de transport pour celle-ci, comme pour la densité, n'est nécessaire. Le problème en est d'autant plus simplifié.

Comme nous l'avons vu dans le premier chapitre, l'équation précédente peut être écrite dans un point de vue Lagrangien grâce à la dérivée particulaire :

$$\frac{D\,\omega}{Dt} = (\omega \cdot \nabla)\mathbf{u} + \nu\Delta\omega \tag{4.1}$$

ce qui revient à résoudre sur un groupe de particules une équation de diffusion augmentée d'un terme d'étirement en 3D. La partie advection de l'équation est quant à elle résoulue implicitement par le transport des particules.

Dans ce contexte, le champ de vorticité $\omega(\mathbf{x}, t)$ est représenté par un ensemble de particules (dont les quantité transportées sont notées par un p ou un q en indice) plutôt que par des valeurs discrètes aux sommets d'un maillage. La vorticité est définie dans l'espace dans l'équation 1.26 comme :

$$\omega(\mathbf{x}, t) \simeq \tilde{\omega}(\mathbf{x}, t) = \sum_p \omega_p\, \zeta_\varepsilon\left(\mathbf{x} - \mathbf{x}_p(t)\right)$$

où ζ_ε est un noyau de mollification sur lequel je reviendrai dans la partie suivante. Ces particules sont simplement transportées par le fluide :

$$\frac{d\,\mathbf{x}_p(t)}{dt} = \mathbf{u}_p = \mathbf{u}(\mathbf{x}_p(t)).$$

L'un des problèmes principaux des méthodes vortex est justement le calcul de la vitesse des particules. Plusieurs méthodes existent, et dans le cadre particulaire, le noyau de Biot-Savart 1.30 est tout particulièrement adapté. Il permet de calculer la vitesse en fonction de la vorticité portée par chacune et de leur position respective, permettant de définir la vitesse en chaque point de l'espace :

$$\mathbf{u}(\mathbf{x}, t) = \sum_p \omega_p\, \mathbf{K}_\varepsilon(\mathbf{x} - \mathbf{x}_p(t)) \tag{4.2}$$

où $\mathbf{K}_\varepsilon$ est un noyau de Biot-Savart régularisé sur lequel je reviendrai dans la partie suivante. Ainsi, l'équation 4.2 est utilisée pour obtenir la vitesse en chaque particule et pour résoudre les termes de transport et d'étirement de 4.1.

Malheureusement, une approche entièrement particulaire de l'équation de vorticité possède plusieurs désavantages dont les principaux sont le coût de calcul lié aux algorithmes existants ainsi que le problème de mauvaise répartition des particules au fur et à mesure du temps, problème inhérent à toute méthode Lagrangienne. C'est pour palier ces difficultés que la méthode des VIC avec remaillage a été développée : elle permet de prendre avantage de la simplicité des méthodes Eulerienne tout en conservant les bienfaits de la représentation Lagrangienne, tout particulièrement en ce qui conserne l'équation de transport. En effet, cette méthode hybride consiste à utiliser une grille sous-jacente à l'ensemble de particules afin de faciliter les calculs (pour la vitesse notamment) et de maîtriser la répartition spatiale des particules par le biais de techniques de remaillage.

La clef des VIC remaillés réside donc dans cette partie très sensible qu'est le remaillage — ou *remeshing* en anglais — où la vorticité est transférée entre les particules et la grille, et vice-versa. En effet, l'opération de remaillage doit être effectuée régulièrement pour éviter que la répartition des particules devienne trop disparate dans le cas de grandes déformations. Ceci pourrait donner lieu à de fortes pertes de précisions dans la résolution des équations, ou à une modification complexe — et donc non souhaitable — des noyaux d'interpolation. Je parle plus en détail de ce problème dans la partie 1.4 où je donne des exemples de fonctions d'interpolations utilisées dans notre contexte.

Bien que je présente en partie 1.6 un exemple d'utilisation d'une méthode purement Lagrangienne, la méthode VIC avec remaillage est la base de mes travaux ; pour plus de lisibilité, j'abuserai du terme « VIC »pour faire allusion à ces deux méthodes couplées (calcul de la vitesse sur la grille plus remaillage). Je rappelle ici l'algorithme général de résolution des équations de Navier-Stokes en formulation vorticité par cette méthode, comme présenté en partie 4.6. Le reste de cette partie détaillera successivement les étapes essentielles de cet algorithme, en insistant sur les choix effectués pour obtenir un calcul efficace. L'algorithme est le suivant :

1. Calcul de la vitesse sur la grille à partir de ω^n.
2. Calcul de l'étirement sur la grille.
3. Interpolation de la vitesse et de l'étirement sur les particules.
4. Advection des particules et intégration du terme d'étirement.
5. Remaillage : création de nouvelles particules basées sur la grille.
6. Diffusion sur la grille : obtention de ω^{n+1}.

1.3 Calcul de la vitesse et équation de transport

Via le noyaux de Biot-Savart

Nous avons vu dans la partie 4.5 qu'il est possible de calculer la vitesse du fluide décrit par un ensemble de particules de vorticité grâce à la loi de Biot-Savart (équation 1.30) discrétisée sur ces dernières. Dans la plupart des cas, il n'est nécessaire de calculer cette vitesse qu'à la position des particules de vorticité, car, dans le cas d'un fluide monophasique, il n'existe qu'une unique équation de transport sur la vorticité qu'il faut résoudre. Le noyau de Biot-Savart étant singulier en zéro, il convient d'utiliser un noyau régularisé supprimant cette singularité. La construction de tels noyaux est détaillée dans le livre [CK00] ; deux critères principaux participent au choix du noyau : l'ordre de la fonction *cutoff* influant sur la précision de $\mathbf{K}_\varepsilon$ et le coût de l'évaluation numérique du noyau. Ce dernier point est primordial car le coût de l'algorithme de calcul des vitesses sur les N particules est en $O(N^2)$; ainsi une réduction du coût de l'évaluation de $\mathbf{K}_\varepsilon$ entraine un gain de temps substantiel. Par exemple, en 2D, le noyau suivant est d'ordre 4 (où la fonction cutoff utilisée est

d'ordre 4) :

$$\mathbf{K}_\varepsilon(\mathbf{x}) = -\frac{\mathbf{x}}{2\pi r^2}\left(1 + (r^2/\varepsilon^2 - 1)e^{-r^2/\varepsilon^2}\right)$$

où $r = |\mathbf{x}|$. Son équivalent en 3D :

$$\mathbf{K}_\varepsilon(\mathbf{x}) = -\frac{\mathbf{x}}{4\pi r^3}\left(1 + e^{-r^3/\varepsilon^3} - 4e^{-2r^3/\varepsilon^3}\right)$$

donné par Beale et Majda [BM85]. C'est ce dernier noyau que j'ai utilisé dans mes travaux.

Le calcul de l'exponentielle sur une machine n'étant pas très rapide, il peut-être envisagé d'utiliser des valeurs tabulées — c'est-à-dire précalculées sur des intervales définis — afin d'en accélérer l'évaluation. Cette discrétisation du noyau induit des erreurs dans les calculs, mais dans le cadre de la synthèse d'images, et particulièrement pour des applications temps-réel, ce gain en temps de calcul peut être substantiel (*cf.* 1.6) et la perte de précision négligeable visuellement.

Reste à définir la valeur de ε. En pratique ce paramètre est choisi de l'ordre de l'espacement (initial) h des particules ; attention car ce choix n'est plus judicieux lorsque les particules s'éloignent ou se rapprochent au cours du temps. Une méthode particulaire basée sur des noyaux à support ε variables doit être considérée dans des méthodes purement Lagrangiennes. Le choix d'une méthode hybride pour mes travaux m'a amené à ne pas traiter cette question. Le lecteur pourra se documenter dans l'article [CKOS00] pour une discussion sur ce sujet.

L'algorithme de sommation permettant de calculer les vitesses sur chaque particule demande exactement N^2 évaluations du noyau $\mathbf{K}_\varepsilon$:

$$\mathbf{u}_p = \sum_q \mathbf{K}_\varepsilon(\mathbf{x}_p - \mathbf{x}_q) \times \omega_q$$

est donné par l'algorithme suivant :

Algorithme 1: Calcul de la vitesse par le noyau de Biot-Savart

```
pour tous les particules p faire  u_p ← 0
pour tous les particules p faire
    pour tous les particules q faire
        u_p ← u_p + K_ε(x_p − x_q) × ω_q
    fin
fin
```

Le noyau de Biot-Savart étant impair, $\mathbf{K}_\varepsilon(\mathbf{x}) = -\mathbf{K}_\varepsilon(-\mathbf{x})$, il est possible de calculer de manière plus rapide la vitesse sur toutes les particules. En effet, la contribution (sans la multiplication par la vorticité) d'une particule p en une particule q étant l'opposée de celle de q sur p, une seule évaluation du noyau est nécessaire par paire de particules. L'algorithme 2 présente de manière synthétique la méthode que j'ai adoptée dans ma thèse pour les simulations purement Lagrangiennes.

Algorithme 2: Calcul rapide de la vitesse par le noyau de Biot-Savart

```
S ← ∅
pour tous les particules p faire  u_p ← 0
pour tous les particules p faire
    pour tous les particules q ∉ S faire
        K ← K_ε(x_p − x_q)
        u_p ← u_p + K × ω_q
        u_q ← u_q − K × ω_p
    fin
    S = S ∪ p
fin
```

Cet algorithme a une complexité équivalente au précédent, c'est-à-dire en $O(N^2)$, mais le nombre d'opérations effectivement réalisées est plus faible : il est très exactement de $\sum_{i=1}^{N} i = N(N+1)/2$ évaluations du noyau, ce qui est environ deux fois plus rapide que l'algorithme « naïf ».

Comme nous l'avons vu au premier chapitre, cette méthode est adaptée naturellement à des conditions aux bords ouvertes, c'est-à-dire quand le domaine n'a pas de limite explicite. Pour des conditions aux bords de type périodique, le calcul de la vitesse est plus coûteux car il faut alors prendre en compte un ensemble de particules virtuelles recopiées périodiquement dans toutes les directions. L'équation permettant de calculer la vitesse $\mathbf{u}_p$ est alors :

$$\mathbf{u}_p = \sum_{\mathbf{k}\in\mathbb{Z}^d} \sum_{q} \mathbf{K}_\varepsilon(\mathbf{x}_p - (\mathbf{x}_q + \mathbf{k}\mathrm{L})) \times \omega_q$$

en dimension d, où L est la taille de la boîte supposée identique dans les d directions. Ici $\mathbf{x}_q + \mathbf{k}L$ dénote les positions d'une particule q d'un point de vue périodique. En pratique, seul un nombre restreint de particules sont « ajoutées », c'est-à-dire que $\mathbf{k}$ ne prend qu'un nombre limité de valeurs. Par exemple, si on considère uniquement les particules dans le voisinage proche ($\mathbf{k} = \{-1, 0, 1\}^d$), ceci revient à considérer les *boîtes* adjacentes et multiplie le coût du calcul par 3^d.

Il est plus difficile d'imposer des conditions de type *no-through* ou *no-slip*. Pour ce faire, la méthode des panneaux [PWS+02] (par exemple) peut-être employée, une application de cette technique en synthèse d'images est présentée dans l'article [PK05]. Dans le chapitre suivant, je propose une nouvelle méthode pour imposer des conditions de non glissement sur les parois d'un obstacle qui peut s'intégrer naturellement aux méthodes purement Lagrangiennes.

Via la fonction courant sur la grille

Dans le cadre de la méthode VIC, le calcul de la vitesse est effectué sur la grille. En effet, l'intérêt de la grille sous-jacente est de pouvoir bénéficier de la rapidité des

techniques de résolution de l'équation de Poisson nécessaire au calcul de la fonction courant :

$$\Delta\psi = -\omega. \tag{4.3}$$

Je rappelle qu'en 2D il suffit de résoudre une seule équation pour la troisième composante non nulle de ψ et ω, alors qu'en 3D il faut en résoudre 3, une pour chaque composante. Il est important de noter que le coût algorithmique de ces méthodes est fonction du nombre M de points de grille qui est différent du nombre de particules de vorticité présentes dans le domaine. Ainsi, il n'est envisageable de comparer la méthode précédente à celles qui suivent qu'en prennant en compte le taux de remplissage de la grille par les particules.

Pour résoudre l'équation 4.3, j'ai utilisé deux méthodes différentes en fonction des conditions aux bords : la résolution dans le domaine fréquentiel grâce à un algorithme rapide de transformée de Fourier pour des conditions périodiques, et par différences finies pour des conditions de type *no-through* sur la vitesse — je rappelle qu'il n'est pas possible de résoudre directement une condition de type *no-slip* en formulation vorticité.

La première méthode consiste à utiliser une représentation fréquentielle des grandeurs étudiées, représentation particulièrement adaptée pour les signaux périodiques. Par exemple, grâce à la transformée de Fourier (notée TF) et son inverse (notée TFI), il est possible de résoudre facilement l'équation de Poisson dans l'espace de Fourier. Soient $\hat{\psi}(\xi) = \mathcal{F}(\psi(\mathbf{x}))$ la TF de ψ, et $\hat{\omega}(\xi) = \mathcal{F}(\omega(\mathbf{x}))$ celle du champ de vorticité. La solution de l'équation de Poisson s'obtient simplement :

$$\begin{aligned}
\mathcal{F}(\Delta\psi) &= -\mathcal{F}(\omega) \\
4\pi^2\xi^2 i^2 \hat{\psi} &= -\hat{\omega} \\
\hat{\psi} &= \frac{\hat{\omega}}{4\pi^2\xi^2}
\end{aligned}$$

avec $i^2 = -1$. Ainsi on trouve la solution dans le domaine spatial par la TFI de la solution, c'est-à-dire :

$$\psi = \mathcal{F}^{-1}\left(\frac{\hat{\omega}}{4\pi^2\xi^2}\right).$$

Le coût de la TF et de la TFI étant de $O(MlogM)$, cette méthode est donc beaucoup plus intéressante en temps de calcul en comparaison de la méthode purement particulaire, en considérant un remplissage important du domaine de calcul, $N \simeq M$. Evidemment, elle demande que la répartition des points de discrétisation soit uniforme. Toutefois, il est envisageable de considérer une répartition non uniforme de ces derniers en utilisant une transformée de Fourier non uniforme.

L'implémentation de cette technique se fait de manière assez simple. Tout d'abord, j'ai choisi d'utiliser une bibliothèque permettant d'effectuer des transformées de Fourier de manière rapide. Les algorithmes de type *Fast Fourier Transform* (transormée de Fourier rapide) procurent un calcul efficace pour ces opérations. Ce dernier est d'autant plus efficace lorsque le nombre d'éléments — les sommets de la grille —

formant la donnée en entrée est une puissance de 2. C'est pourquoi dans les simulations des grilles de taille L^d (où d est la dimension) sont généralement utilisées. La bibliothèque *FFTW* [FJ] fournit une implémentation de cette méthode, c'est elle que j'ai intégrée dans mon logiciel.

En pratique, à partir d'un tableau de taille $M = \mathrm{L}^d$ stockant les valeurs de ω spatialement sur la grille, le résultat obtenu est un tableau de taille $\mathrm{L}^{d-1} \times \mathrm{L}/2$ où sont stockées les valeurs complexes associées aux fréquences du signal représentant le champ étudié. La raison pour laquelle la taille de ces données est deux fois plus petite qu'en représentation spatiale est que la transformée de Fourier d'un signal à valeurs réelles est symétrique : l'amplitude d'une fréquence f est la même qu'à la fréquence $-f$.

L'algorithme permettant de calculer ψ à partir de la vorticité, en omettant la propriété énoncée précédemment pour plus de clarté, est donc le suivant :

Algorithme 3: Calcul de la vitesse via la fonction courant grâce à la FFT en 3D

// L'indice f représente le tableau stockant les valeurs complexes du champ associé.
// Les crochets $[i]$ représentent le $i^{\text{ème}}$ élément du tableau.
// α est une constante comptant pour le facteur d'échelle du problème et le facteur lié aux transformées de Fourier.
// $\alpha = h^3/\mathrm{L}^3$, où h est l'espacement spatial des points de grille, supposé constant
// et L le nombre de points de grille.
// F est le sous-domaine des fréquences étudiées : $\mathrm{F} \subset \mathbb{Z}$ et $\mathrm{F} = [-\mathrm{L}/2, ..., \mathrm{L}/2]$

$\omega_f \leftarrow \mathcal{F}(\omega)$
pour tous les *fréquences* $\xi \in F^3$ **faire**
  $\psi_f[\xi_1][\xi_2][\xi_3] \leftarrow \omega_f[\xi_1][\xi_2][\xi_3]/(4\pi^2\xi^2)$
fin
$\psi \leftarrow \mathcal{F}^{-1}(\psi_f)$
pour tous les *les points* $\mathbf{x}$ *du tableau* **faire**
  $\psi[x_1][x_2][x_3] \leftarrow \alpha * \psi[x_1][x_2][x_3]$
fin

Intéressons nous maintenant à la condition aux bords de type *no-through*. Cette approche est différente des méthodes spectrales dans le sens où des conditions aux bords doivent être explicitement données pour la grandeur qui nous intéresse ici : la fonction courant. Comme je l'ai expliqué dans le premier chapitre en partie 4.6, il est possible d'imposer une condition de non pénétration aux bords d'une boîte de calcul en imposant des conditions de type Dirichlet homogène pour les composantes de ψ tangentes au bord de la boîte et de type Neumann homogène pour la

composante normale. Ceci est aisément obtenu par les algorithmes de résolution de l'équation de Poisson basés sur des schémas de différences finies en jouant sur les valeurs contenues dans la matrice à inverser pour résoudre ce problème. J'ai intégré la bibliothèque *FishPack* dans mon logiciel, cette dernière permet de résoudre de manière efficace l'équation de Poisson avec les conditions souhaitées grâce à des schémas de différences finies d'ordre 2 ou 3 influençant l'erreur numérique commise lors du calcul. Le coût de tels algorithmes est en $O(MlogM)$.

Une fois les valeurs de ψ connues en chaque point de grille, il faut maintenant calculer la vitesse. Dans la méthode VIC, ceci est simplement effectué en calculant numériquement le rotationel de la fonction courant : $\mathbf{u} = \nabla \times \psi$ sur la grille à l'aide de schémas de différences finies. Il est possible d'utiliser des schémas d'ordre élevé pour calculer la vitesse ; un soin particulier doit être apporté dans une boîte fermée où des schémas d'ordre moins élevé (ayant donc un support plus petit) ou décentrés (*cf.* annexe 2.1) doivent être utilisés.

Étirement

Dans mon implémentation de la méthode VIC, j'ai utilisé le fait que la vorticité et la vitesse se trouvent, en début d'itération, toutes les deux sur la grille pour calculer de manière efficace l'étirement du tourbillon. J'ai donc utilisé la définition de l'étirement sous forme conservative :

$$\frac{D\,\omega}{Dt} = \nabla \cdot (\omega \otimes \mathbf{u}) \tag{4.4}$$

puis interpolé ce terme sur les particules avant le transport. Numériquement, lorsque la vitesse est calculée sur la grille, j'ai utilisé le même schéma de différences finies que pour calculer $\nabla \times \psi$, soit un schéma d'ordre 4.

Transport de la vorticité

L'équation de transport sur la vorticité est résolue dans un repère Lagrangien comme expliqué dans le premier chapitre. Ceci permet de résoudre exactement l'équation de transport, sans erreur numérique, ainsi que de s'abstraire de la condition CFL liée à la résolution de l'équation en Eulerien. Une fois la vitesse $\mathbf{u}_p$ calculée en chacune des particules, que ce soit par le noyau de Biot-Savart ou par le calcul des vitesses sur la grille, l'équation de transport :

$$\frac{\partial\,\omega_p}{\partial t} + (\mathbf{u}_p \cdot \nabla)\omega_p = \frac{D\,\omega_p}{Dt} = 0$$

est résolue implicitement par le transport de la particule par le fluide :

$$\frac{d\,\mathbf{x}_p}{dt} = \mathbf{u}(\mathbf{x}_p) = \mathbf{u}_p.$$

À chaque pas de temps, après le calcul de la vitesse sur les particules, leur position est donc mise à jour :

$$\mathbf{x}_p(t+\delta t) = \mathbf{x}_p(t) + \delta t\, \mathbf{u}_p(t)$$

de manière explicite au premier ordre en temps. Afin d'approcher au mieux la vitesse réelle du fluide et de suivre au mieux la caractéristique de celle-ci, il est envisageable d'utiliser d'autres schémas d'intégration pour le déplacement des particules. J'ai utilisé un schéma de type Runge-Kutta d'ordre 2 (*cf.* figure 4.1) — équivalent de la méthode du point milieu (*mid-point method* en anglais) — modifié pour mes simulations. Initialement, la méthode Runge-Kutta 2 consiste à évaluer la vitesse au demi pas de temps suivant et d'utiliser cette vitesse pour advecter la particule vers le pas de temps suivant :

$$\mathbf{x}_p(t+\delta t) = \mathbf{x}_p(t) + \delta t\, \mathbf{u}(\mathbf{x}_p(t) + \frac{\delta t}{2}\mathbf{u}(\mathbf{x}_p(t), t), t + \frac{\delta t}{2}).$$

Or le calcul de la vitesse $\mathbf{u}(\mathbf{x}_p(t) + \frac{\delta t}{2}\mathbf{u}(\mathbf{x}_p(t), t), t + \frac{\delta t}{2})$ au demi pas de temps

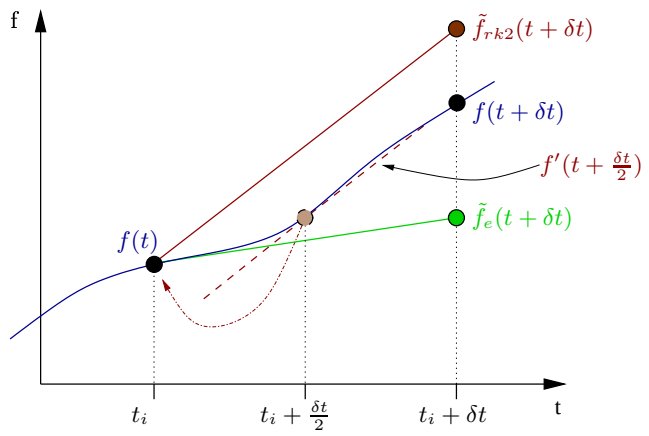


FIG. 4.1 – Comparaison graphique de la méthode d'intégration d'Euler standard d'ordre 1 (en vert) et d'un schéma de type Runge-Kutta 2 (en marron) ; la courbe bleue est la fonction de référence à approcher. La distance entre la valeur $\tilde{f}$ calculée pour le temps $t+\delta t$ est plus petite avec la seconde méthode car ce schéma est d'ordre 2.

suivant nécessite la vorticité $\omega(\mathbf{x}, t + \frac{\delta t}{2})$ également au demi pas de temps suivant, ce qui nécessite la résolution du reste de l'équation de vorticité (pour la diffusion et l'étirement en 3D). Ces calculs sont coûteux et ceci doublerait le temps de calcul global nécessaire à la résolution de l'équation de conservation (ainsi que, comme on le verra plus tard, sur le transport de la densité en présence d'un fluide multiphasique). Dans l'intention de réduire ce temps de calcul tout en conservant une bonne précision dans l'équation de transport, j'ai utilisé une approximation du champ de vitesse au demi pas de temps suivant, de manière similaire à [Wey06]. Ces travaux montrent que cette approximation permet bien de résoudre le déplacement à l'ordre 2, dans le

cadre de méthodes couplées au remaillage dans une formulation vitesse. J'ai utilisé cette approche dans la méthode VIC implémentée (voir ci-dessous).

Dans une méthode purement Lagrangienne, le champ de vorticité transporté par une particule peut être pris comme une approximation de la vorticité au temps suivant, comptant uniquement la résolution de l'équation de transport :

$$\omega_p(t + \frac{\delta t}{2}) \simeq \omega_p(t)$$

qui transporte aussi le champ de vitesse, en omettant la diffusion et l'étirement des vortex. Ainsi, l'équation de Runge-Kutta 2 modifiée consiste à calculer une vitesse approchée $\tilde{\mathbf{u}}_p$ telle que :

$$\tilde{\mathbf{u}}_p = \sum_q \mathbf{K}_\varepsilon(\mathbf{x}_p(t + \frac{\delta t}{2}) - \mathbf{x}_q(t + \frac{\delta t}{2})) \times \omega_q(t)$$

où $\mathbf{x}_p(t + \frac{\delta t}{2}) = \mathbf{x}_p(t) + \frac{\delta t}{2}\mathbf{u}_p(t)$ (et de manière similaire pour $\mathbf{x}_q$), et de l'intégrer dans l'équation de déplacement de la particule :

$$\mathbf{x}_p(t + \delta t) = \mathbf{x}_p(t) + \delta t\, \tilde{\mathbf{u}}_p(t).$$

Comme il a été proposé dans [Wey06], il est aussi possible d'utiliser un champ de vitesse $\mathbf{u}(\mathbf{x})$ constant sur le demi pas de temps. Dans une formulation vorticité, l'équivalent correspond à approcher la vitesse de la particule au demi pas de temps suivant $\tilde{\mathbf{u}}_p$ en fonction du champ de vorticité $\omega(\mathbf{x}, t) = \omega(\mathbf{x}, t + \frac{\delta t}{2})$ constant.

Dans mon implémentation pour la simulation purement Lagrangienne, j'ai donc résolu l'équation sur la vitesse grâce au noyau de Biot-Savart aux positions des particules à l'instant $t + \frac{\delta t}{2}$ après une intégration d'Euler explicite à l'ordre 1, c'est-à-dire :

$$\tilde{\mathbf{u}}_p = \sum_q \mathbf{K}_\varepsilon(\mathbf{x}_p(t + \frac{\delta t}{2}) - \mathbf{x}_q(t)) \times \omega_q(t)$$

avec

$$\mathbf{x}_p(t + \frac{\delta t}{2}) = \mathbf{x}_p(t) + \frac{\delta t}{2}\mathbf{u}_p.$$

Dans le cadre d'une méthode VIC où la vitesse est calculée sur la grille en résolvant l'équation de Poisson sur la fonction courant, la méthode utilisée est similaire à cette dernière. La seule différence provient du fait que les particules de vorticité sont situées n'importe où dans le domaine, lorsqu'on met de côté le remaillage pour l'instant. Il faut alors interpoler les valeurs de la vitesse $\mathbf{u}(\mathbf{x})$ stockées aux sommets de la grille au niveau des positions des particules $\mathbf{x}_p$. Ces interpolations viennent ajouter des erreurs en fonction du pas de discrétisation spatiale δx aux erreurs sur le schéma de déplacement. Par chance, si un remaillage est effectué à chaque pas de temps, les particules se trouvent alors situées à l'instant t exactement aux sommets du maillage et aucune interpolation ne doit être effectuée si on adopte un schéma de remaillage

adapté (*cf.* partie 1.4). Mais l'utilisation d'une méthode de type Runge-Kutta avec un champ de vitesse fixe dans l'intervalle $[t, t+\frac{\delta t}{2}]$ nécessite tout de même d'effectuer une interpolation à la position $\mathbf{x}_p(t+\frac{\delta t}{2})$ de la particule au demi pas de temps. Cependant, bien qu'elle rajoute de l'erreur en fonction de δx et augmente le temps de calcul en $O(N)$ (une interpolation par particule pour les N particules), l'utilisation de cette méthode reste intéressante car elle permet de réduire l'erreur temporelle d'un ordre de grandeur alors que les fonctions d'interpolation généralement utilisées ont déjà un ordre d'erreur quadratique en espace (*cf.* partie 1.4).

Étudions l'intérêt et la limite de cette méthode : en utilisant un schéma d'interpolation d'ordre 2, l'erreur commise en comparaison d'un schéma de transport traditionnel en $O(\delta t)$ est en $O(\delta x^2) + O(\delta t^2)$. Le schéma d'Euler d'ordre 1 est donc préférable si et seulement si

$$O(\delta t) < O(\delta t^2) + O(\delta x^2).$$

C'est-à-dire si

$$O(\delta t) < O(\delta x^2). \tag{4.5}$$

Or, si on adopte une contrainte sur le pas de temps à cause de la résolution de l'équation de diffusion sur la vorticité, nous avons en 1D l'inégalité suivante :

$$\delta t \leq \frac{\delta x^2}{2\nu} < \frac{\delta x^2}{\nu}.$$

Si on prend un pas de temps égal à $\frac{\delta x^2}{2\nu}$, l'inégalité 4.5 devient :

$$O(\delta t) < \nu O(\delta t).$$

Autrement dit, la méthode basée sur celle de Runge-Kutta 2 est toujours plus intéressante si la viscosité est faible ($\nu \ll 1$), ce qui est le cas pour les simulations qui nous intéressent dans le cadre de cette thèse. De manière plus générale, si δt est choisi de l'ordre de δx, les deux méthodes sont équivalentes en terme d'erreurs numériques et le schéma d'Euler d'ordre 1 reste moins coûteux en temps de calcul.

Conditions aux bords pour le transport En présence de conditions aux bords de type périodiques, il est nécessaire de repositionner les particules déplacées à l'intérieur de la boîte, que cela soit dans un cadre purement particulaire ou dans une méthode VIC. En 1D, soit l'intervalle périodique $[A, B[$ dans lequel évoluent les particules, la particule p est située en $x_p(t)$ tel que $x_p(t) \in [A, B[$, lorsque celle-ci sort de la boîte, c'est-à-dire $x_p(t+\delta t) \notin [A, B[$, il convient de déplacer cette particule à l'intérieur du domaine en la remplaçant par son équivalente en périodique :

$$\begin{aligned} x_p(t+\delta t) &\leftarrow x_p(t+\delta t) - (B-A) \qquad \text{si } x_p(t+\delta t) >= B \quad \text{et} \\ x_p(t+\delta t) &\leftarrow x_p(t+\delta t) + (B-A) \qquad \text{si } x_p(t+\delta t) < A. \end{aligned}$$

Dans le cas d'une boîte fermée, les particules ne peuvent physiquement pas sortir de la boîte. Néanmoins, la vitesse des particules près des bords du domaine peuvent

les amener à traverser celui-ci si le pas de temps n'est pas modifié en conséquence. Ce dernier doit ainsi être adapté afin que les particules proches des bords ne s'échappent pas. Rappelons l'équation de transport pour une vitesse $\tilde{\mathbf{u}}_p$ donnée :

$$\mathbf{x}_p(t+\delta t) = \mathbf{x}_p(t) + \delta t\, \tilde{\mathbf{u}}_p,$$

en 1D la position $x_p(t+\delta t)$ doit appartenir à l'intervalle $]A, B[$ (en omettant la présence de particules aux bords), en d'autres termes :

$$\begin{aligned} \mathbf{x}_p(t) + \delta t\, \tilde{\mathbf{u}}_p > A \Rightarrow \delta t < \frac{A - x_p(t)}{\tilde{\mathbf{u}}_p} \quad \text{et} \\ \mathbf{x}_p(t) + \delta t\, \tilde{\mathbf{u}}_p < B \Rightarrow \delta t < \frac{B - x_p(t)}{\tilde{\mathbf{u}}_p} \end{aligned}$$

comme uniquement une vitesse négative (resp. positive) peut faire sortir x_p du domaine au delà de A (resp. de B). En pratique, comme la condition de non pénétration est assurée, cette contrainte n'entre pas en jeu car la vitesse orthogonale aux bords aux points proches de ceux-ci est très faible.

Par ailleurs, un soin particulier doit être apporté aux fonctions d'interpolation utilisées avec la méthode VIC dans un domaine fermé. Effectivement, il est nécessaire d'utiliser des fonctions d'interpolation adaptées à un maillage fermé, le support de ces dernières pouvant prendre plus de deux points de grille. La section suivante revient sur ces cas particuliers dans le cadre d'une technique de remaillage, étape lors de laquelle j'ai utilisé les mêmes fonctions d'interpolation.

1.4 Méthodes de remaillage et fonctions d'interpolation

La méthode des VIC s'appuie sur le remaillage fréquent des particules de vorticité afin de bénéficier au mieux des avantages des méthodes sur grille. Le point clef de cette méthode est l'utilisation de fonctions d'interpolation adaptées afin de préserver au mieux les propriétés dynamiques du fluide. C'est pourquoi les fonctions d'interpolation choisies pour distribuer les particules sur la grille doivent être le moins diffusives possible et conserver le maximum de moments sur la vorticité. De manière générale, la qualité d'une fonction d'interpolation est liée aux nombres de points utilisés pour celle-ci, c'est-à-dire de son support. Le livre [CK00] présente en profondeur différentes fonctions d'interpolations ainsi que les propriétés associées.

Décrivons maintenant plus précisément le fonctionnement de la méthode de remaillage dans le cadre du présent travail. Je me place dans le cas 1D pour plus de clarté, la méthode en dimension supérieure s'en déduit naturellement. Un ensemble de particules situées aux sommets x_p d'une grille uniforme d'espace $h = 1$ sont advectées par le fluide. Leur nouvelle position associée est notée par $\tilde{x}_q$. Soit un champ $\tilde{\Gamma}$ transporté par cet ensemble de particules, où chacune d'entre elle porte $\tilde{\Gamma}_q = \tilde{\Gamma}(\tilde{x}_q)$. Nous souhaitons interpoler cette quantité aux points de grille x_p où seront créées les particules qui serviront à la prochaine itération de l'algorithme des

VIC que j'ai implémenté. La valeur portée par une particule fraîchement créée au point x_p (sur la grille) est calculée de la manière suivante :

$$\Gamma(x_p) = \sum_q \tilde{\Gamma}_q W(x_p - \tilde{x}_q) \tag{4.6}$$

où $W(x)$ est une fonction d'interpolation. Pour une raison de conservation, il va de soi que

$$\int \Gamma(x) = \int \tilde{\Gamma}(x)$$

ce qui correspond de manière discrète à :

$$\sum_q W(x - x_q) = 1.$$

L'étude du développement limité de $\Gamma(x_p) - \sum_q \tilde{\Gamma}_q W(x_p - \tilde{x}_q)$ renseigne sur l'ordre de la fonction d'interpolation. Il est primordial que la fonction d'interpolation conserve aux mieux les moments du champ étudié — principalement la vorticité —, ce qui est traduit par :

$$\int x\Gamma(x) = \int x\tilde{\Gamma}(x)$$

pour le premier moment et

$$\int |x|^2\Gamma(x) = \int |x|^2\tilde{\Gamma}(x)$$

pour le second.

La fonction « triangle »(*cf.* figure 1.4) est la plus simple et la plus naturelle, elle est définie par :

$$\Lambda(x) = \begin{cases} 0 & \text{si } |x| > 1 \\ 1 - |x| & \text{sinon,} \end{cases}$$

qui est d'ordre 1, qui possède un support de 2 points au maximum.

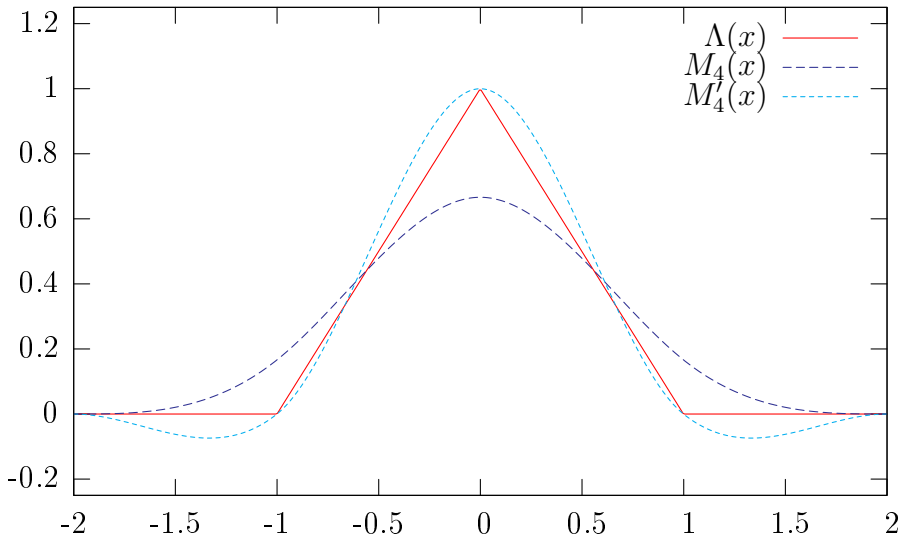


FIG. 4.2 – Différentes fonctions d'interpolation utilisées dans les méthodes vortex. Le schéma M_4' est celui que j'ai adopté dans mes travaux.

Pour ma part, j'ai utilisé un schéma d'interpolation de type M_4' (*cf.* figure 1.4) — utilisé initialement dans les méthodes de type SPH [GM77] pour leurs bonnes propriétés de conservation — qui d'ordre 3 et donne des résultats convaincants pour les méthodes vortex. La fonction est la suivante :

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

Cette dernière possède un support de 4 points, ce qui est convenable du point de vue temps de calcul. Je renvoie le lecteur à [CK00] pour des comparaisons entre différentes fonctions d'interpolation. La construction d'une fonction d'interpolation pour des particules en dimension plus élevée s'effectue simplement par le produit tensoriel de fonctions 1D sur chacune des composantes de la position :

$$W(\mathbf{x}) = \Pi_{i=1}^{d} W(x_d).$$

L'évaluation numérique de l'équation 4.6 demande un algorithme de coût $O(MN)$ où M est le nombre de points de grille et N le nombre de particules. Or, la fonction d'interpolation W possédant un support limité, il est aisé de construire un algorithme de coût linéaire en le nombre de particules en évaluant W uniquement aux points du maillage faisant partie du support de cette fonction. Par exemple l'interpolation d'une particule sur la grille, à l'aide du schéma M_4' en 3D, demande 64 évaluations de cette fonction. Dans le cas plus général d'une grille régulière d'espacement h, il convient de redimensioner les grandeurs x en évaluant $M_4'(x/h)$.

Dans le cadre de conditions périodiques, le remaillage est effectué de manière triviale en considérant les points de grilles de manière périodique lors de l'interpolation des particules sur ceux-ci. Avec des boîtes fermées, l'utilisation du schéma M_4' n'est pas possible pour les particules proches des bords. En effet, des points de grille à l'extérieur du domaine entrent dans la distribution des quantités à transférer. Comme je l'ai précisé précédemment, les vitesses aux bords du domaine dans le cadre de conditions de non-pénétration restent faibles et il est possible de négliger la contribution sur ces points extérieurs au prix de la perte (faible) de conservation des quantités transportées. Il est aussi envisageable d'utiliser des fonctions d'interpolation à support plus compact afin de distribuer entièrement ces quantités à l'intérieur du domaine ou bien de considérer des fonctions décentrées adaptées. Par contre, comme pointé par [Wey06], adopter un mélange de différents types de fonctions pour l'intérieur et le bord du domaine ne permet pas de conserver de manière consistante les quantités interpolées. Néanmoins, dans le cadre de la synthèse d'images, l'erreur introduite peut être considérée comme négligeable, et c'est cette approche que j'ai adoptée dans mes travaux en utilisant un schéma adapté :

$$\Lambda_d(x) = \begin{cases} 0 & \text{si } |x| > 3 \\ \frac{x(x-1)}{2} & \text{si } 2 < |x| \leq 3 \\ x(2-x) & \text{si } 1 < |x| \leq 2 \\ 1 - \frac{3x}{2} + \frac{x^2}{2} & \text{sinon,} \end{cases}$$

pour les particules se trouvant à moins d'une case du bord 0 (et de manière similaire pour les autres bords).

1.5 Diffusion

Dans le cadre de ma thèse, j'ai utilisé plusieurs méthodes pour calculer le terme de diffusion de la vorticité $\frac{d\omega}{dt} = \nu\Delta\omega$ (où ici la dérivée peut-être prise du point de vue soit Lagrangien soit Eulerien). Sur les particules, la résolution de cette équation peut être effectuée, entre autres, à l'aide de la méthode de *Particle Strength Exchange* (PSE), en utilisant un noyau de diffusion η régularisé comme présenté en équation 1.31, méthode basée sur la formulation intégrale l'opérateur de diffusion. L'implémentation du PSE sur les particules s'effectue d'une manière similaire au calcul de vitesses par la loi de Biot-Savart. En intégrant ce terme avec un schéma d'Euler explicit à l'ordre un, nous obtenons :

$$\omega_p^{n+1} = \omega_p^n + \frac{\nu\delta t}{\varepsilon^2} \sum_q (\omega_q^n - \omega_p^n)\, \eta_\varepsilon(\mathbf{x}_q^n - \mathbf{x}_p^n).$$

Ce calcul peut être effectué dans la même boucle de calcul que pour celui de la vitesse sur la particule p tout en profitant des mêmes possiblités d'accélération dues à la symétrie du noyau η_ε. Il est aussi possible de précalculer des valeurs de η_ε pour éviter l'évaluation directe et coûteuse de ce noyau en acceptant les erreurs induites par cette simplification. Dans mes travaux, j'ai pris pour η un noyau Gaussien $\mathcal{G}$:

$$\eta_\varepsilon(\mathbf{x}) = \varepsilon^{-2}\eta(\mathbf{x}/\varepsilon) = \varepsilon^{-2}\mathcal{G}(\mathbf{x}/\varepsilon) = \varepsilon^{-2}(4\pi)^{-1}e^{-|\mathbf{x}|^2/(4\varepsilon^2)}.$$

Les méthodes VIC bénéficient d'un des intérêts supplémentaires par rapport aux méthodes particulaires. Du fait du splitting des opérations, le traitement du terme de diffusion $\nu\Delta\omega$ se fait indépendemment du terme de transport. Le fait d'effectuer cette opération après le remaillage permet de profiter de la localisation de la vorticité sur une grille régulière et ainsi d'avoir la possibilité d'appliquer un choix large de méthodes de résolution de l'équation de la chaleur. Comme l'espacement entre les particules est fixé, il est possible de précalculer les valeurs du noyau de la méthode PSE afin de réduire grandement le coût de calcul. De plus, comme ce noyau est à décroissance rapide, il est possible de le réduire encore en ne considérant qu'un voisinage proche (une ou deux cellules) des points de grille diminuant ainsi le coût algorithmique en $O(N)$. Cette méthode peut être ainsi vue comme une méthode classique de différences finies.

Autrement, il est possible d'utiliser les méthodes habituelles de résolution de l'équation de la chaleur, je renvoie le lecteur à l'annexe 2.2 pour une présentation plus complète de celles-ci. Si la viscosité est faible, une méthode explicite — la plus rapide et facile à implémenter — peut être appliquée tant qu'elle ne contraint pas trop le pas de temps, soumis à la condition $\delta t < h^2/(\nu 6)$ en 3D. Sinon une méthode de type implicite, spatialement ou dans le domaine fréquentiel est à envisager, pour cela j'ai utilisé les solveurs *FishPack* et *fftw3*, respectivement.

Il est aussi possible de résoudre exactement l'équation de la chaleur dans le domaine fréquentiel en utilisant le noyau de la chaleur, solution élémentaire de l'équation. Le champ solution $\hat{\omega}(\xi, t)$ peut être calculé très simplement en fonction de la solution initiale $\hat{\omega}(\xi, 0)$:

$$\hat{\omega}(\xi, t) = e^{4\pi^2\xi^2\nu t}\hat{\omega}(\xi, 0), \tag{4.7}$$

équation qui, sous forme itérative, s'écrit :

$$\begin{aligned}\hat{\omega}(\xi, t+\delta t) &= e^{4\pi^2\xi^2\nu\delta t}\hat{\omega}(\xi, t),\\ \hat{\omega}^{n+1}(\xi) &= e^{4\pi^2\xi^2\nu\delta t}\hat{\omega}^{n}(\xi).\end{aligned}$$

C'est cette approche que j'ai adoptée pour les écoulements périodiques.

Il est important de noter que si les calculs de la vitesse et de la diffusion sont effectués dans le domaine fréquentiel, et d'après l'algorithme des VIC, nous pouvons gagner en temps de calcul en évitant de calculer une fois superflue la transformée de Fourier inverse $\hat{\omega}^{n+1}$ et la transformée de Fourier $\hat{\omega}^{n+2}$ entre la fin d'un pas de temps et le début du suivant.

1.6 Application : simulation de fumée

La première application des méthode vortex que je présente est la simulation d'un fluide à une seule phase. Afin de pouvoir visualiser le mouvement du fluide, j'utilise un ensemble de particules de fumée simplement transportées par celui-ci. Le rendu est effectué grâce à la bibliothèque 3D OpenGL aujourd'hui largement répandue. La fumée est représentée par un champ scalaire α indiquant la densité de

fumée (sans masse) et sur lequelle est résolue l'équation de transport :

$$\frac{D\,\alpha}{Dt} = 0.$$

La visualisation de ce champ de densité se fait à l'aide de particules visuelles orientées vers la caméra, représentées par un disque semi transparent — dont l'opacité est fonction de la densité — de taille fixe, assez grand pour que les particules se chevauchent dans le plan image et donnent une impression de fumée plus ou moins dense mais compacte. Le choix d'une primitive différente pour la fumée pourrait être considérée, principalement par l'utilisation de *Shader*, afin d'augmenter la qualité du rendu. Mais, dans le domaine de la simulation de fluides, le fait que le calcul des vitesses soit très long mène à accepter de passer plus de temps dans le rendu, par une méthode de type lancé de rayon ou *photon mapping* par exemple, afin d'obtenir des résultats encore plus convaincants. J'ai trouvé que le rendu par disques semi-transparents fournissaient d'assez bons résultats visuels pour un temps de calcul négligeable.

Pour une visualisation plus nette du mouvement du fluide, j'ai aussi attribué une couleur variable à chaque point de l'espace en transportant une donnée *couleur* en même temps que la densité. Le transport de la quantité α est effectuée soit de manière Lagrangienne (exemples 1 et 2) où 10000 particules sont réparties aléatoirement là où la densité n'est pas nulle.

Les exemples que je présente ci-dessous sont en 3D, le cas 2D étant plus simple et visuellement moins parlant, je ne m'y intéresserai pas ici. Le cas étudié ici est celui d'une turbine propulsant un jet d'air instantanné vers les z positifs, à l'intérieur d'un domaine à vitesse initiale nulle. La turbine est modélisée par un solide rigide fixe dont la vitesse n'est pas nulle ; l'intégration des solides dans le fluide sera expliqué dans le chapitre suivant. La vitesse à l'intérieur de cette turbine cylindrique, de diamètre $0.2\,m$ et de hauteur $0.5\,m$, est de $2\,m.s^{-1}$ pour une durée de 200 millisecondes après laquelle le fluide évolue sans force extérieure. Sauf note contraire, la viscosité ν est de $10^{-4}\,m.s^{-1}$. Le pas de temps $\delta t = 10^{-2}\,s$ est constant pour toutes les simulations, il est notable que ce pas de temps est plutôt faible en comparaison de la vitesse du fluide (de l'ordre de $2\,m.s^{-1}$) et permet d'obtenir des résultats très dynamiques physiquement et visuellement.

Tout d'abord je montre un calcul basé sur la méthode vortex purement Lagrangienne et y explique les résultats attendus pour ce cas test et les résultats obtenus. Ensuite je présente les calculs effectués par la méthode des *Vortex In Cells* pour les mêmes conditions.

Sauf indication contraire, tous les calculs numériques présentés dans ce mémoire ont été effectués un Dell Precision M70, équipé d'un processeur Intel Centrino 2GHz et de 1Go de mémoire RAM.

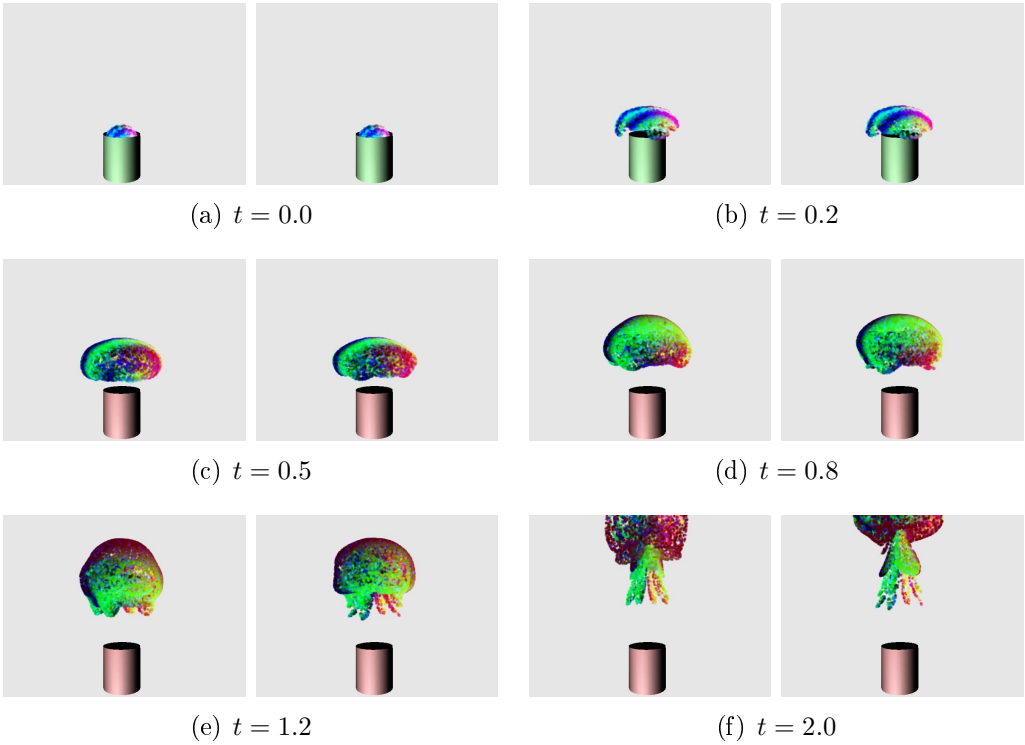
Exemple 1 : Particules et conditions libres

Ce premier exemple prend place dans un domaine aux conditions libres : la vitesse et la vorticité à l'infini sont considérées comme nulles : $\mathbf{u}_\infty = \omega_\infty = 0$. J'ai donc naturellement choisi pour cet exemple de résoudre l'équation de vorticité sur des particules Lagrangiennes, avec la technique de calcul de la vitesse et de son gradient par le noyau de Biot-Savart ainsi qu'un schéma de type PSE pour la diffusion.

Initialement, un ensemble de 648 particules de vorticité sont placées autour de la surface du cylindre car c'est à cet endroit que la vorticité est créée pour simuler le jet. Par souci de simplicité, aucune nouvelle particule n'est créée au cours du temps, alors qu'il serait nécessaire d'en augmenter le nombre pour simuler au mieux la turbine ; les résultats obtenus ainsi sont assez convaincants pour ne pas complexifier le problème.

L'espacement initial entre les particules est de $h = 0.0625\,m$ et le paramètre de régularisation est fixé initialement à $\varepsilon = 1.5h$ afin que les particules se chevauchent. Augmenter ce paramètre lisse plus le champ de vorticité et ainsi la vitesse, nous souhaitons évidemment limiter cette diffusion du champ de vorticité régularisé afin d'avoir une simulation et un résultat visuel les plus dynamiques possible. Il serait envisageable d'adapter ε en fonction de l'espacement des particules, lorsqu'aucun remaillage n'est effectué. Malheureusement — comme avec le temps des paquets de particules vont se former — réduire le paramètre, même si cela permettra localement d'avoir plus de précision, diminuera la qualité de la représentation du champ de vorticité pour les particules s'éloignant. En conséquence, j'ai choisi de conserver ε constant au cours de la simulation, le problème de support étant palié dans la suite de mes travaux par l'utilisation d'une méthode VIC.

Je présente en figure 4.3 les résultats obtenus par la méthode particulaire avec un schéma Eulerien simple pour le calcul de la vitesse et de la variation de vorticité en parallèle de ceux obtenus avec un schéma de type Runge-Kutta 2 pour un pas de temps deux fois plus élevé ($dt_{rk2} = 0.02$). La couleur du cylindre est verte lorsque la turbine est en route et rouge lorsqu'elle est éteinte, c'est-à-dire lorsque plus aucune force extérieure n'est ajoutée à la simulation. La boule de particules de fumée est éjectée vers le haut par l'anneau de vorticité initialement situé autour du cylindre, tangeant à celui-ci dans la direction normale à la vitesse. Puis cet anneau est transporté par le fluide et est déformé par lui même (le terme d'étirement) tout en se diffusant spatialement.



(a) $t = 0.0$ (b) $t = 0.2$

(c) $t = 0.5$ (d) $t = 0.8$

(e) $t = 1.2$ (f) $t = 2.0$

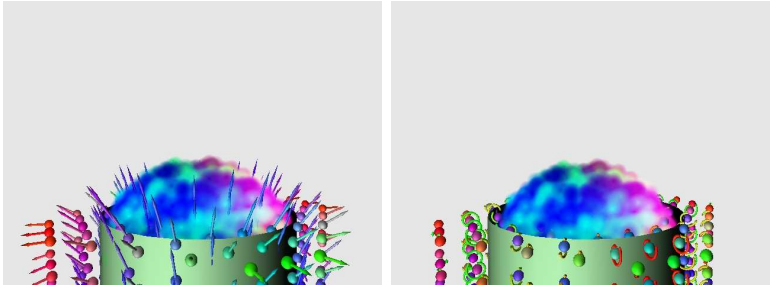
FIG. 4.3 – AJet de fumée simulé par une méthode particulaire. Comparaison pour certains pas de temps des résultats obtenus par un schéma d'Euler explicite d'ordre 1 (images de gauche) et un schéma de type Runge-Kutta 2 (images de droite). Les résultats obtenus avec ce dernier sont plus réalistes et dynamiques.

A l'instant $t = 0.2$, les particules du bas ont été poussées rapidement vers celles du haut (dans l'orientation de l'image) et celles du dessus aux bords sont poussées sur les côtés et redescendent en comparaison du centre de gravité de l'ensemble des particules, faisant de celui-ci une sorte de dôme. Entre le temps $t = 0.5$ et $t = 0.8$, les particules de vorticité évoluent de manière autonome et continuent de transporter la fumée de manière presque linéaire vers le haut. Au temps $t = 1.2$ on peut observer des déformations assez importantes sur la « boule »de fumée et notamment l'apparition de quatre queues formées de particules qui sont toujours aspirées par le vortex mais moins rapidement que le reste de la forme. Ces structures se trouvent localisées autour de directions particulières, ici sur les droites $x = y$ et $x = -y$. D'un pourrait croire que cette asymétrie est due à des erreurs dans les calculs, notamment celui du gradient de $\mathbf{u}$, mais comme les calculs sont effectuées de manière Lagrangienne, c'est en fait la répartition initiale des particules — sur les points d'une grille Eulerienne dans les directionx (x, y, z) — qui crée cette distorsion. Une rotation d'un angle θ de cette grille initiale autour de l'axe des z modifie de ce même angle l'orientation de ces trainées. A l'instant $t = 2.0$, nous voyons clairement la formation d'un « champignon »similairement aux résultats attendus, forme typique, et impressionnante, du mouvement des particules observé à la suite de tests d'explosions atomiques. On peut voir sur les images correspondant à la simulation utilisant

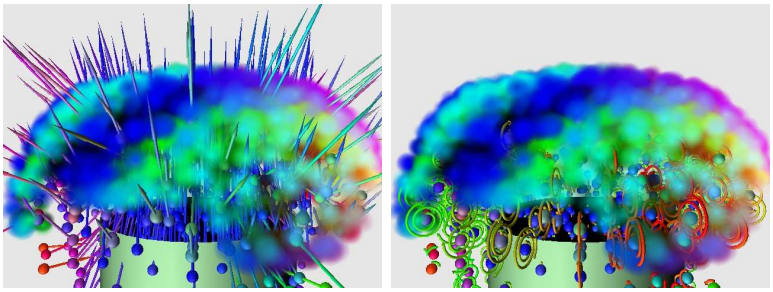
une méthode de Runge-Kutta à l'ordre 2 une meilleure conservation de l'anneau de vorticité, ceci est particulièrement visible sur l'image au temps 2 où avec le schéma d'intégration simple la « boule »de particules tout en haut a éclaté : les particules ont un mouvement beaucoup moins réaliste. La légère disymétrie apparaissant au temps $t = 1.2$ est produite par les erreurs numériques venant perturber sensiblement les calculs, une petite erreur sur le calcul de la vitesse et de la variation de vorticité sera propagée à l'ensemble des particules à l'instant d'après. Le terme de diffusion de la vorticité a pour effet de diminuer ces erreurs en les lissant sur l'ensemble des particules. La moins bonne répartition des particules avec le temps entre aussi en jeu dans la faible qualité du champ de vitesse par endroit, notamment lorsque les particules sont très proches : le calcul de la vitesse y est alors trop lissé car la régularisation du noyau de Biot-Savart vient couper l'influence des particules juxtaposées.

La figure 4.4 présente les particules de vorticité utilisées pour cette dernière simulation. Dans les couples d'images, celle de gauche représente la vitesse au niveau des particules où la couleur est modulée en fonction de la direction (la composante rouge pour x, vert pour y et bleu pour z) et la longueur de la flêche est proportionnelle à la norme de $\mathbf{u}_p$. Les images à droite représentent quant à elles la vorticité au niveau de chaque particule : une spirale est dessinée autour de celles-ci, dans le plan normal au vecteur vorticité et où le même code couleur est utilisé. La longueur de la spirale est proportionnelle à la norme de ω_p. La vorticité transportée augmente dans les premiers instants du fait de la force extérieure introduite par la turbine en marche, et par la suite elle est modifiée par l'étirement et la diffusion du champ de vorticité. On peut observer qu'au niveau du pied du champignon, peu de particules sont présentes et leur espacement plus élevé qu'initialement ce qui induit une moins bonne représentation du champ de vitesse localement — et ainsi un transport moins précis de la fumée — en comparaison du chapeau.

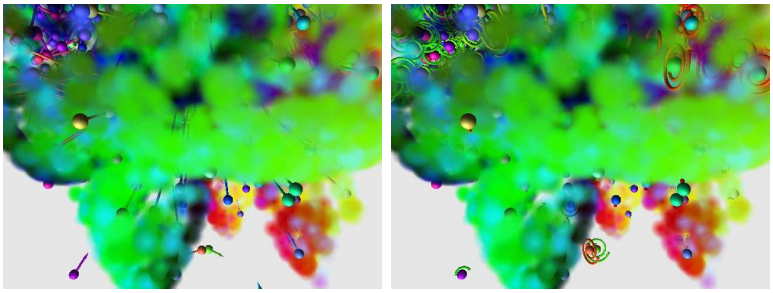
Le temps de calcul pour la résolution d'une itération avec l'algorithme rapide, en ne considérant que les particules de vorticité, est d'environ 140 millisecondes pour l'intégration d'ordre 1 et de 280 millisecondes pour le schéma RK2. Le pas de temps pour le schéma RK2 est deux fois plus grand que pour le premier exemple, le temps de calcul global est donc ainsi le même pour des résultats plus précis et réalistes avec le schéma d'ordre 2. En comparaison, en quadruplant le nombre de particules de vorticité ($N = 2600$), la simulation prend 3 secondes et 6 secondes (pour le schéma d'ordre 1 et RK2, respectivement) par itération, ce qui représente une augmentation de 2000% du temps de calcul.



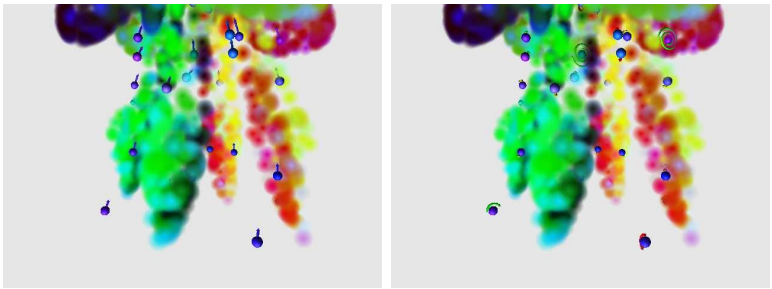
(a) $t = 0.0$



(b) $t = 0.2$



(c) $t = 1.2$



(d) $t = 2.0$

FIG. 4.4 – Jet de fumée simulé par une méthode particulaire. Zoom sur les particules de vorticité utilisées pour la simulation, à différents instants. À gauche : les vecteurs vitesse sont dessinés, à droite la vorticité est représentée par une spirale dans le plan tangeant au vecteur vorticité, et de longueur proportionnelle à la norme de celle-ci.

Pour transporter les particules de fumée, il est nécessaire de calculer la vitesse en chacune d'elles. Afin d'obtenir une visualisation réaliste de la fumée, un grand

nombre de particules est nécessaire (ici 10000) ce qui rendrait prohibitif l'évaluation directe de la vitesse par le noyau de Biot-Savart. Cet algorithme a un coût en $O(MN)$ où M est le nombre de particules de fumée et N celui de particules de vorticité, avec $M \gg N$. Au lieu de cela, j'ai utilisé une grille sous-jacente englobant les particules de fumée et calculé la vitesse du fluide en chacun de ces points. Puis, la vitesse attribuée à chaque particule est une interpolation de la vitesse portée par la grille, réduisant ainsi le coût algorithmique en $O(LN)+O(M)$ où L est le nombre de point de grille, avec $L \ll N$ et le coût $O(M)$ représente le coût des interpolations. Pour ces exemples, une grille de taille $8 \times 8 \times 8$ a été suffisante pour obtenir un mouvement réaliste des particules avec le schéma d'interpolation M_4', assez proche de celui du fluide réel. J'ai utilisé le schéma M_4' comme fonction d'interpolation.

Nous avons pu voir dans cet exemple les difficultés inhérentes à la méthode purement Lagrangienne, principalement le coût en temps de calcul avec un nombre de particules important — soucis qui sera une limitation discriminante pour les simulations multiphasiques que nous verrons par la suite — et les problèmes résultants de la mauvaise répartition des particules au cours du temps. Néanmoins, cette formulation permet d'obtenir des résultats très satisfaisants pour des simulations de phénomènes nécessitant un nombre faible de particules qui permettent tout de même d'obtenir une animation très dynamique et réaliste du mouvement d'un fluide, laissant libre le champ ouvert à des simulations vortex temps-réel du fait de la localité de la vorticité en comparaison des méthodes basées sur la vitesse. Les articles [AN05] et [PK05] fournissent des exemples similaires, le premier permettant des calculs très rapides et réalistes sur une base de méthode d'anneaux de vorticité.

Exemple 2 : VIC et boîte périodique

Dans cet exemple, j'ai utilisé la méthode des *Vortex In Cells* afin de simuler la même expérience que précédemment. Les conditions sont identiques, à part que le nombre de particules évolue avec le temps : les particules d'importance mineure — face à un critère donné : ici le rapport entre l'intensité de la vorticité en la particule et la vorticité maximale — sont supprimées après le remaillage et des particules sont créées aux points de grille proches et à l'intérieur de la turbine si de la vorticité doit être ajoutée à ces endroits (je reviendrai sur les explications en rapport avec la création de vorticité dans le chapitre concernant l'interaction de solides rigides avec le fluide).

Il est important de noter que les résultats obtenus ne peuvent être identiques à la méthode particulaire car dans cette dernière le paramètre de régularisation des particules reste constant à chaque instant — ce qui permet pas de résoudre précisément les équations lorsque celles-ci s'espacent et se rapprochent — alors que pour la méthode VIC, les particules sont espacées uniformément et la vitesse ainsi calculée correspondant exactement (aux erreurs d'approximation près des schémas numériques) au champ de vorticité sur la grille. Néanmoins, les résultats obtenus pour les mêmes pas d'espace et de temps avec la méthode VIC sont moins bons

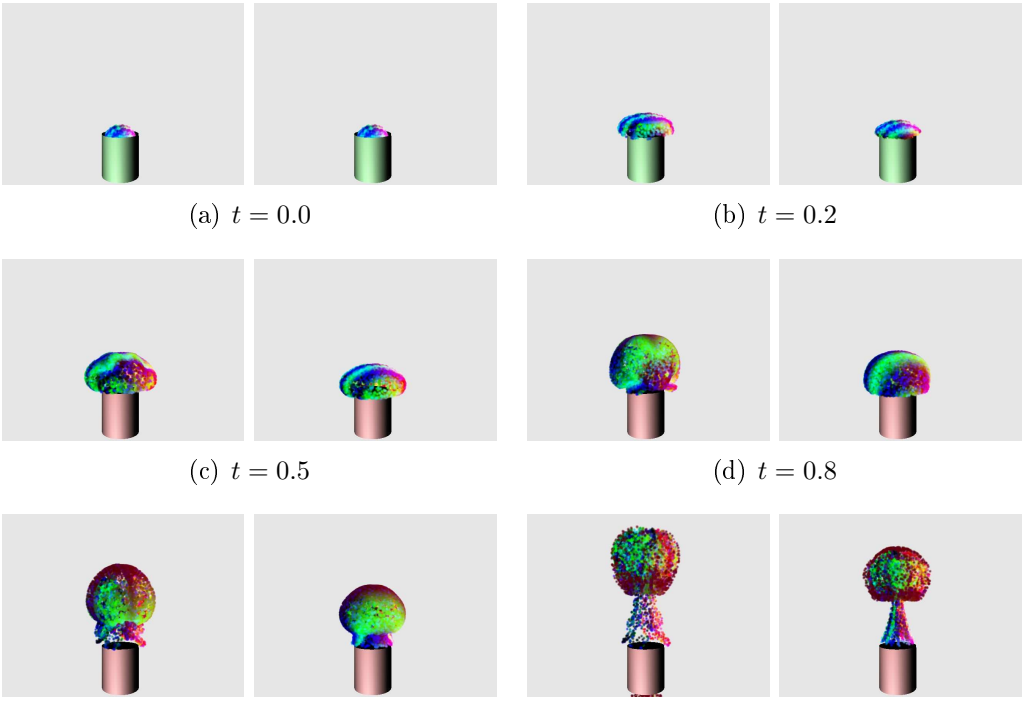
qu'avec la méthode particulaire, ceci est dû à la méthode de remaillage qui induit une diffusion supplémentaire de la vorticité et au fait que la grille Eulerienne soit uniforme, ce qui implique que deux particules proches vont se « mélanger »là où la formulation Lagrangienne les aurait gardées séparées.

C'est pour ces raisons que j'ai augmenté légèrement la résolution spatiale du maillage Eulerien en fixant le pas d'espace à $h = 1/24$ pour la première simulation et $h = 1/32$ dans la deuxième que je présente ci-dessous. Le pas de temps est quant à lui toujours constant à 0.01. Le fluide évolue dans une boîte périodique de taille $(1, 1, 2)$ (en x, y et z respectivement), assez grande pour que la fumée ait le temps de former le « champignon ». J'ai choisi ces conditions aux bords car imposer des conditions libres est plus complexe ; j'ai donc pris une boîte assez grande pour émuler au mieux ces conditions. Cependant, il est important de noter que la condition de périodicité influe de manière importante sur le comportement du fluide, même s'il reste similaire à la simulation purement Lagrangienne avec des conditions libres aux bords, tout du moins sur le temps de simulation qui nous intéresse.

L'évaluation numérique des opérateurs gradient, rotationnel et Laplacien est effectuée à l'aide de schémas centrés d'ordre 4. La viscosité étant assez faible pour les échelles, la contrainte de pas de temps sur la résolution de l'équation de la chaleur est assurée, j'ai donc utilisé un schéma Eulerien explicite très peu cher. La fonction d'interpolation utilisée pour les transferts particules-grille (pour le remaillage) et grille-particules (pour le schéma RK2 sur la vitesse) est la fonctione M'_4, pour les raisons énoncées plus haut dans ce chapitre. Il est important de noter que similairement à la méthode particulaire, les résultats obtenus avec un schéma de type RK2 sont bien meilleurs qu'avec un schéma d'ordre 1, bien que le splitting sur l'équation de conservation de la vorticité induise une erreur d'ordre 1 en temps.

Nous pouvons observer sur les images de la figure 4.5 les résultats obtenus pour $h = 1/24$ et $h = 1/32$. La forme de champignon apparaît dans les deux cas à partir du temps $t = 2$ avec une précision moins élevée pour la simulation à plus faible résolution, où le chapeau de la forme se détériore davantage à cause des erreurs numériques plus grandes. De plus, l'espacement entre les points de grille étant plus grand dans le premier cas, la diffusion de la vorticité introduite par le schéma de remaillage après transport étale plus la vorticité ce qui semble faire avancer les particules plus vite vers le haut de la boîte. Les résultats obtenus avec un pas d'espace deux fois plus petit qu'avec la simulation particulaire s'accorde plutôt bien avec le comportement observé ci-dessus. Le temps de calcul est de 110 (resp. 300) millisecondes par itération en début de simulation en présence de $N = 3060$ (resp. $N = 4880$) particules et croît linéairement en fonction de N jusqu'à atteindre 200 (resp. 360) millisecondes au temps $t = 2$ avec $N = 15000$ (resp. $N = 25600$). Plus en détail, environ 31% du temps de calcul est dédié au remaillage, 21.5% au schéma RK2 en comptant les interpolations de la vitesse, 24% pour la résolution de l'équation de Poisson sur la fonction courant (pour les trois composantes) et le reste pour les calculs annexes par différences finies. C'est donc la résolution de l'équation de transport (et le remaillage associé) qui est la plus onéreuse.

En comparaison avec la méthode purement particulaire, le temps de calcul est du même ordre pour un nombre de particules bien plus élevé et des résultats qualitativement similaires. Ce temps de calcul grandit en défaveur de la méthode purement Lagrangienne, lorsque le nombre de particules augmente, alors qu'avec la méthode VIC, le temps de calcul augmente raisonnablement. La qualité du résultat semble plus grande avec la méthode particulaire pour un nombre de particules moins élevé. Ceci peut s'expliquer par deux faits : tout d'abord les conditions aux bords considérées dans ce cas sont libres, ce qui est plus proche du résultat attendu visuellement. Ensuite, l'équation de vorticité est résolue sur une particule par la résolution d'intégrales sur tout le domaine, plus particulièrement pour l'étirement qui, en méthode VIC, est calculé sur la grille par différences finies. De plus, l'équation de transport est résolue exactement dans une formulation Lagrangienne, alors qu'avec la méthode VIC avec remaillage, cette dernière opération introduit de la diffusion et des erreurs numériques.



(a) $t = 0.0$ (b) $t = 0.2$

(c) $t = 0.5$ (d) $t = 0.8$

(e) $t = 1.2$ (f) $t = 2.0$

FIG. 4.5 – Jet de fumée simulé par une méthode VIC avec remaillage. Captures du mouvement du fluide à différents instants, pour une grille de dimension $h = 1/24$ (à gauche) et $h = /32$ (à droite).

2 Fluide à densité variable

2.1 Préambule

La présence de plusieurs phases dans une même simulation compléxifie la résolution du problème en deux points :

- l'équation de Navier-Stokes est plus difficile à résoudre, du fait d'une densité et/ou viscosité non constante,
- le problème conjoint du transport de la densité et/ou viscosité est lui aussi délicat.

J'ai présenté au chapitre 2 les méthodes existantes concernant ces deux problèmes. J'ai expliqué notamment de quelle manière il est possible d'intégrer la densité variable dans les équations en formulation vorticité, en précisant l'approximation de Boussinesq valable pour les fluides à faible variation de densité.

Dans cette partie, je me focalise tout d'abord sur la description d'une méthode de résolution de fluide multiphasiques en vorticité, autour d'une reformulation de l'équation de conservation. Puis je reviendrai sur le sujet du transport d'interface, en présentant les résultats obtenus avec différentes méthodes level set. Finalement, je présenterai des exemples d'application sur des fluides biphasiques qui nous intéressent tout particulièrement en synthèse d'images.

2.2 Reformulation de l'équation de vorticité

Pour plus de lisibilité, je rappelle ici l'équation 2.5 de conservation de la vorticité en présence d'un fluide à densité et viscosité variable :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = (\omega \cdot \nabla)\mathbf{u} + \nu \Delta \omega + \nabla \nu \times \Delta \mathbf{u} - \nabla \left(\frac{1}{\rho}\right) \times \nabla p.$$

Cette équation est difficile à résoudre car elle demande le calcul supplémentaire de la pression p, ce qui fait perdre un peu de l'intérêt de la formulation vortex. Il est toutefois possible de considérer une simplification de cette équation, en présence de faible variation de densité : l'approximation de Boussinesq :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = (\omega \cdot \nabla)\mathbf{u} + \mu \Delta \omega + \nabla \rho' \times \mathbf{g}$$

où rappelons ici que $\nu = \mu/\rho$. Nous verrons dans la sous-partie dédiée aux exemples, que cette approximation permet d'obtenir des résultats plutôt convaincants en synthèse d'image, même lorsque la différence de densité devient importante.

Toutefois, il est possible de reécrire l'équation de vorticité à densité variable de manière à supprimer le terme de pression et ne conserver que les grandeurs connues explicitement à l'instant t : la vitesse $\mathbf{u}$ et la vorticité ω. En effet, reprenons l'équation de Navier-Stokes à densité et viscosité variables sous la forme suivante :

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) - \nabla \cdot (\mu \nabla \mathbf{u}) + \nabla p - \rho \mathbf{f} \quad = \quad 0,$$

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}\right)-\nabla\cdot(\mu\nabla\mathbf{u})+\nabla p-\rho\mathbf{f} \quad = \quad 0,$$

où $\otimes$ est le produit tensoriel, et comme $\nabla\cdot\mathbf{u}=0$ et $\frac{\partial\rho}{\partial t}+(\mathbf{u}\cdot\nabla)\rho=0$. Prenons maintenant le rotationnel de cette équation afin d'en obtenir l'équivalent en formulation vorticité :

$$\begin{aligned}\nabla\times\left(\rho\left(\frac{\partial \mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}\right)-\nabla\cdot(\mu\nabla\mathbf{u})+\nabla p-\rho\mathbf{f}\right)= \\ \rho\left(\frac{\partial \omega}{\partial t}+(\mathbf{u}\cdot\nabla)\omega-(\omega\cdot\nabla)\mathbf{u}\right) \\ -\mu\Delta\omega-\nabla\mu\times\Delta\mathbf{u}-\nabla\times(([\nabla\mathbf{u}]\cdot\nabla)\,\mu) \\ -\rho\nabla\times\mathbf{f}+\nabla\rho\times\left(\frac{\partial \mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}-\mathbf{f}\right)\end{aligned}$$

qui, en divisant maintenant par ρ nous donne :

$$\frac{\partial \omega}{\partial t}+(\mathbf{u}\cdot\nabla)\omega \quad = \quad (\omega\cdot\nabla)\mathbf{u}+\nu\Delta\omega+\nabla\times\mathbf{f} \tag{4.8}$$

$$-\frac{\nabla\rho}{\rho}\times\left(\frac{\partial \mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}-\mathbf{f}\right) \tag{4.9}$$

$$+\frac{1}{\rho}\left(\nabla\mu\times\Delta\mathbf{u}+\nabla\times(([\nabla\mathbf{u}]\cdot\nabla)\,\mu)\right). \tag{4.10}$$

Nous reconnaissons dans la première partie 4.8 de cette l'équation l'équation de conservation de vorticité à densité et viscosité constantes, dans le terme 4.9 une partie de l'équation de Navier-Stokes en formulation vitesse similaire au dernier terme de l'équation 2.5 et qui s'annule lorsque la densité est constante, et dans 4.10 un terme provenant de la variation de viscosité μ du fluide qui disparaît lorsque μ est constant.

On peut remarquer que, comme la variation de densité et/ou viscosité est non nulle uniquement sur l'interface, de même qu'avec l'approximation de Boussinesq, il y a création de vorticité seulement localement, au niveau de la transition entre les phases.

Cette nouvelle équation fait intervenir le terme de transport de la vitesse si problématique en formulation vitesse. Mais étant donné que nous ne manipulons pas directement la vitesse, comme cette dernière est calculée en fonction de ω (via la fonction courant), et que ce terme est localisé à l'interface, les erreurs commises sur l'évaluation numérique de ce terme auront moins d'impact sur la qualité de l'écoulement qu'en formulation vitesse-pression. Afin de calculer numériquement ce terme, j'ai utilisé des schémas de différences finies sur la grille. L'évaluation numérique de la convection ne permet pas d'assurer que $\frac{\partial \mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}+\mathbf{f}$ soit à divergence nulle, néanmoins, comme le terme $\frac{\nabla\rho}{\rho}\times\left(\frac{\partial \mathbf{u}}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}-\mathbf{f}\right)$ conserve analytiquement la propriété de solénoïdalité du champ de vorticité, ceci revient à considérer

uniquement la partie à divergence nulle de ce terme, lorsqu'il est introduit sur les tourbillons. Cependant, dans 4.9, la partie $\nabla\rho/\rho$ peut devenir numériquement peu régulière dans le cas de fortes variations de densité. L'épaisseur de la régularisation du champ de densité doit donc être adaptée en fonction du rapport de densité afin de capter au mieux le terme $\nabla\rho/\rho$.

La dernière ligne 4.10 de l'équation est à évaluer uniquement si la viscosité dynamique μ n'est pas constante, où dans ce cas il est nécessaire de respecter la condition de stabilité si on traite le terme de diffusion de manière explicite (ce qui sera généralement le cas pour des viscosité faibles lorsque la discrétisation n'est pas trop fine). Le dernier terme de cette ligne peut être développé afin d'effectuer moins d'erreurs numériques introduites par l'évaluation de dérivées d'ordre 2. Afin de réduire au maximum l'impact des erreurs numériques introduites par l'évaluation de ces termes, il faut, autant que faire se peut, utiliser des grandeurs adaptées pour ρ et μ, soit $\nabla\rho/\rho \simeq \nabla\rho$ et/ou $\nabla\mu$ assez faible, voire nul. Les résultats obtenus avec cette méthode sont tout à fait satisfaisants, néanmoins une étude plus approfondie sur le sujet serait à envisager.

2.3 Transport de l'interface par level set

Comme nous l'avons vu au chapitre 2, le transport de la densité peut être effectué à l'aide d'une fonction level set annexe, elle même transportée par le fluide. C'est cette approche que j'ai choisie dans mes travaux de thèse. C'est aujourd'hui une méthode largement étudiée et utilisée en synthèse d'images pour la simulation de fluide ; les méthodes level set s'intègrent naturellement dans les équations du fluide et fournissent des schémas de transport efficaces et qualitativement intéressants, tout en fournissant une représentation intéressante pour le calcul de la normale et la courbure par exemple. En comparaison, une méthode de type champ de phases pour le transport de ρ ou d'un scalaire passif est moins directe et demande la résolution d'équations couplées plus ardues dans le cadre de la simulation de fluide.

Les méthodes de capture d'interface de type level set sont elles plus libres car elles résolvent une équation de transport simple :

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

où ϕ est une fonction de telle sorte que

$$\Gamma = \{\mathbf{x} \mid \phi(\mathbf{x}) = 0\}$$

représente l'interface entre deux milieux. Le choix du prolongement de ϕ en dehors de l'interface importe peu dans les équations non régularisées, où, par exemple, la variation de densité entre deux phases s'écrit :

$$\rho(\mathbf{x}) = \rho_1 \left(1 - H(\phi(\mathbf{x}))\right) + \rho_2 H(\phi(\mathbf{x})),$$

avec ρ_1 et ρ_2 les densités associées à chaque phase. Mais comme nous l'avons vu au chapitre 2, la singularité de ρ rend difficile en pratique la résolution des équations,

et c'est pour cette raison qu'il est nécessaire de régulariser cette grandeur :

$$\rho(\mathbf{x}) = \rho_1 \left(1 - H_\varepsilon(\phi(\mathbf{x}))\right) + \rho_2 H_\varepsilon(\phi(\mathbf{x})).$$

Dans ce cas, le choix du prolongement de ϕ a son importance si on veut contrôler rigoureusement l'épaisseur de cette régularisation. L'utilisation d'une fonction distance signée pour ϕ est un choix intéressant car elle permet de contrôler la localisation de la variation de ρ (et donc de $\nabla\rho$) dans une fine bande autour de l'interface, de l'ordre du pas d'espace h en choisissant $\varepsilon = O(h)$.

Nous verrons aussi dans le chapitre suivant comment il est possible de profiter d'une représentation d'un objet rigide et d'une membrane déformable par une fonction distance initiale, dans le cadre d'interactions fluide/structure, augmentant ainsi l'intérêt d'utiliser ce type de modélisation.

Pour le transport des fonctions level set, j'ai choisi dans mes travaux la méthode nommée *Lagrangian Particle Level Set* (*cf.* 3.2). En effet, celle-ci est basée sur le même principe Lagrangien que j'ai utilisé pour le transport de la vorticité sur le principe des méthode de type *Vortex In Cell*. Dans cette formulation, la level set ϕ est discrétisée par un ensemble de N particules portant chacune une quantité ϕ_p. Le champ ϕ est alors défini en n'importe quel point par :

$$\phi(\mathbf{x}) = \sum_{p=1}^{N} \phi_p \zeta_\varepsilon(\mathbf{x} - \mathbf{x}_p)$$

où ζ_ε est une fonction *cut off* régularisée de support de l'ordre du pas de discrétisation spatial des particules. Les particules sont transportées par le fluide

$$\frac{D\,\mathbf{x}_p}{Dt} = \mathbf{u}(\mathbf{x}_p)$$

et l'équation de transport sur l'une d'elle est

$$\frac{D\,\phi(\mathbf{x}_p)}{Dt} = \frac{D\,\phi_p}{Dt} = 0.$$

Afin de palier les problèmes apparaissant lors de fortes déformations du maillage, un algorithme de remaillage (similaire à celui utilisé pour les méthodes vortex) est nécessaire et permet une bonne résolution de l'équation de transport dans une approche hybride Lagrangienne-Eulerienne. Cette méthode possède l'avantage de ne pas être soumise à une condition de type CFL qui serait handicapante lors du transport de l'interface dans un fluide turbulent et rapide. En effet, dans le but de conserver précisément la localisation des deux phases — ce qui est une question importante numériquement pour la précision des calculs et visuellement car la disparition d'une partie du volume aux endroits de forte courbure serait gênante pour l'observateur — il est nécessaire de discrétiser finement le maillage et donc de réduire le pas d'espace réduisant ainsi le pas de temps s'il est soumis à la contrainte CFL.

En pratique, il est nécessaire de remailler les particules uniquement lorsque le champ représenté par les particules devient trop déformé. Numériquement, la distorsion au temps t peut être mesurée par :

$$H_j(t) = \sum_p v_p \zeta_\varepsilon(\mathbf{x}_p(t) - \mathbf{x}_j)$$

où les $\mathbf{x}_j$ représentent les points de discrétisation de la grille Eulerienne sous-jacente et $v_j = h^d$ est le volume d'une cellule en dimension d. Initialement, $H_j(0) = 1$ et cette grandeur doit être conservée avec le temps, ou en d'autres termes, les particules sont considérées comme ne représentant plus correctement le champ transporté losrque $\Delta H = N^{-1} \sum_j \frac{|H_j(t) - H_j(0)|}{H_j(0)}| > \epsilon$ où ϵ est une valeur assez petite qui nous permet de contrôler la fréquence du remaillage en fonction des erreurs de discrétisations induites par le transport des particules. Comme le champ de vitesse utilisé pour l'advection des particules est fourni par la résolution des équations de Navier-Stokes, et que celle-ci peut demander des pas de temps assez petit, il n'est pas nécessaire de remailler à chaque pas de temps. Néanmoins, sur la base de notre méthode VIC avec remaillage, le pas de temps est assez large et contrôlé principalement par la turbulence locale du fluide $|\omega|_\infty^{-1}$, qui est une mesure de la distorsion du champ de vitesse et donc des particules. C'est pourquoi j'ai choisi de remailler à chaque pas de temps les particules de level set sur la grille. J'ai effectué le remaillage avec la même fonction M_4' que pour les particules de vorticité, soit :

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

Afin de contrecarrer en partie les erreurs numériques — tout de même présentes lorsque $\nabla \cdot \mathbf{u} = 0$ — déformant le volume, j'ai utilisé la distortion, telle une renormalisation, comme moyen de correction du remaillage :

$$\phi(x) = \frac{\sum_p M_4'(x - x_p)\phi_p}{\sum_p M_4'(x - x_p)}, \tag{4.11}$$

en 1D, le cas 3D utilisant simplement le produit tensoriel des fonctions M_4'.

Du prolongement de la fonction level set

Dans le cadre d'un fluide multiphase, le transport de l'interface est un point primordial de la solution. Effectivement, c'est la variation de densité à l'interface qui produit la dynamique du fluide et un soin particulier doit donc être porté sur la précision de la représentation de la densité. Si initialement la level set est choisie comme une fonction distance signée, cette propriété n'est pas conservée par le transport de celle-ci par une fonction vitesse quelconque, même dans un fluide incompressible. Le champ de vitesse déforme le prolongement de ϕ par :

$$\frac{\partial \nabla \phi}{\partial t} + (\mathbf{u} \cdot \nabla)\nabla\phi = \left([\nabla \mathbf{u}]^T \cdot \nabla\right)\phi.$$

Le gradient de ϕ est donc étiré par le fluide et sa norme n'est donc pas conservée par l'équation d'advection, comme $|\left([\nabla \mathbf{u}]^T \cdot \nabla\right)\phi| \neq 0$ dans le cas général. L'épaisseur de l'interface régularisée se trouve donc modifiée, ce qui n'est pas souhaitable en pratique. En effet, la régularisation de ρ approche l'équation de Navier-Stokes lorsque le paramètre de régularisation tend vers 0 : nous voulons donc pouvoir contrôler l'épaisseur de l'interface ainsi que limiter les distorsions sur celle-ci qui pourraient entrainer des erreurs numériques importantes. Dans mes travaux, j'ai remarqué que l'absence de maîtrise sur le prolongement de ϕ rend les simulations instables et très perturbées lorsque l'interface entre deux phases est en interaction avec un fluide peu visqueux. Les applications en synthèses d'images demandent une dynamique turbulente pour un fluide afin d'obtenir des animations réalistes, permettant l'apparition de phénomènes intéressants comme ceux observés dans la réalité entre l'air et l'eau par exemple. Bien que la méthode de renomarlisation consistant, intuitivement, à diviser ϕ par la norme de $\nabla\phi$ afin que la level set se comporte localement comme une fonction distance, soit une alternative très peu coûteuse aux méthodes de reinitialisation, j'ai trouvé que dans le cadre de la simulation de plusieurs phases, elle permettait de stabiliser un peu les résultats sans toutefois résoudre complétement le problème.

C'est pour ces raisons que j'ai choisi — comme il est souvent considéré dans les méthodes pour la simulation de fluides multiphases — d'appliquer une technique de réinitiliasation de la level set afin qu'elle reste, autant que faire se peut, proche d'une fonction distance signée. Pour ce faire, j'ai opté pour une méthode de type Hamilton-Jacobi qui consiste à résoudre une EDP sur ϕ afin de l'approcher d'une fonction distance signée. Le principe consiste à propager le front, représenté par la courbe (en 2D) de niveau 0 de la level set, au-delà de celui-ci dans la direction normale avec une vitesse unitaire. Je rappelle ici l'équation à résoudre sur ψ :

$$\frac{\partial \psi}{\partial \sigma} = Signe(\psi_0)(1 - |\nabla\psi|). \tag{4.12}$$

Avec $\psi_0 = \phi$ et pour un temps fictif σ infini, la propriété $|\nabla\psi| = 1$ est assurée tout en conservant la localisation de l'interface : $\Gamma = \{\mathbf{x} \mid \phi(\mathbf{x}) = 0\} = \{\mathbf{x} \mid \psi(\mathbf{x}) = 0\}$. L'équation 4.12 peut être reécrite sous la forme d'une équation de transport avec un terme de forçage, en posant $|\nabla\phi| = \frac{\nabla\phi\cdot\nabla\phi}{|\nabla\phi|}$:

$$\frac{\partial \psi}{\partial \sigma} + (u_{\nabla\phi} \cdot \nabla)\phi = Signe(\psi_0), \tag{4.13}$$

avec $u_{\nabla\phi} = Signe(\psi_0)\nabla\phi/|\nabla\phi|$.

Numériquement, j'ai résolu cette équation à l'aide de schémas WENO adaptés à l'équation de transport ; je renvoie le lecteur aux documents référencés dans le chapitre 2 dans lesquels il pourra trouver plus de détails sur la résolution de ce type d'équations. J'ai préféré cette méthode de reinitialisation aux algorithmes de type *Fast Marching* pour la raison que celle-ci est plus contrôlable en terme d'erreur de discrétisation et de convergence (bien qu'un grand nombre de pas de temps soit nécessaire dans le cas général pour une fonction quelconque). Entre deux pas de temps

de simulation, la level set est étirée, mais pas nécessairement de façon importante, ce qui me permet d'appliquer simplement quelques pas d'itération afin de résoudre l'équation d'Hamilton-Jacobi pour obtenir un résultat convenable. Par *convenable* je signifie que la solution soit assez proche d'une fonction distance. En effet, j'ai vu qu'en pratique il n'était pas nécessaire que cette propriété soit assurée (aux erreurs numériques près) pour stabiliser les simulations ; c'est pourquoi l'utilisation d'une méthode de type *Fast Marching* ne m'a pas semblé nécessaire. Il est toutefois important de noter que cette dernière permet de mieux conserver la localisation de l'interface car elle fixe les valeurs de ϕ aux points de grille englobant la courbe de niveau 0, à la différence de la résolution de l'EDP qui elle peut modifier légèrement la position de Γ à cause des erreurs numériques. En pratique, je n'ai pas observé de problème de déplacement de l'interface après reinitialisation de ϕ.

Une remarque concernant le choix de la méthode de reinitialisation : notre transport de l'interface étant effectué par des particules Lagrangiennes, il serait envisageable d'appliquer une technique adaptée à cette représentation, comme le pointe les auteurs de [HK05a]. Cependant, comme j'ai appliqué un algorithme de remaillage à chaque pas de temps, les valeurs de ϕ sont alors connues sur la grille et ainsi l'utilisation d'une méthode traditionnelle me semblait la plus adaptée.

2.4 Application : simulation de fluide biphasique

Je présente dans cette partie un exemple de simulation d'un fluide possédant deux phases, dans le but de valider visuellement les résultats obtenus avec notre méthode. Cette application concerne la chute de gouttes initialement situées dans l'air et tombant sur une surface séparant l'air du liquide. La simulation est calculée dans une boîte de taille $[0, 0.1]^3$ avec des conditions périodiques. Ce choix est motivé par le fait que les calculs sont plus simples et rapides en présence de ce type de conditions aux bords, en comparaison d'une boîte fermée. Afin d'obtenir un résultat réaliste, j'ai imposé implicitement des conditions de type non glissement sur les bords de la boîte de calcul en considérant un cube fixe immergé permettant d'annuler la vitesse du fluide aux bords du domaine. Pour ce faire, j'ai utilisé la méthode que je décrirai dans le chapitre suivant pour l'interaction fluide-structure, le lecteur peut considérer ici une manière d'émuler efficacement les conditions aux bords de non glissement dans une résolution périodique des équations.

Six gouttes de densité $\rho_1 = 1000$ sont soumises à une gravité $\mathbf{g} = (0, 0, -9.8)$ et sont en interaction avec une deuxième phase de densité $\rho_2 = 1$. La viscosité du fluide est fixée à $\nu = 10^{-5}$ dans les deux phases. L'interface entre les fluides est représentée par une fonction level set reinitialisée tous les pas de temps par la résolution de l'équation de type Hamilton-Jacobi, avec un pas de temps fictif de $\delta\sigma = h/2$ où h est le pas d'espace. J'ai trouvé que deux itérations suffisaient largement à stabiliser la simulation, ce qui permet de limiter le coût de ce calcul. La vitesse est calculée par la transformée de Fourier (comme dans l'exemple à une phase) et la diffusion traitée de manière exacte dans le domaine fréquentiel. Afin de capter précisément le mouvement des gouttes, j'ai fixé le pas de temps à $\delta t = 0.001$.

Je présente en figure 2.4 les résultats obtenus pour une grille de taille 64^3. Nous pouvons voir au temps $t = 0.04$ la forme particulière que prend une goutte en l'absence de tension de surface, l'ajout de cette dernière rendrait plus difficile la déformation des gouttes et tendrait à conserver une forme sphérique. Il est nécessaire d'utiliser un coefficient de tension important pour obtenir un résultat « naturel », ceci demandant l'utilisation de petits pas de temps. Les résultats obtenus sans tension de surface étant suffisamment réalistes, je n'ai pas considéré ce cas. A l'instant $t = 0.06$, les gouttes sont presques toutes entrées en contact avec la surface. Sur l'image correspondante, nous pouvons voir que la goutte la plus proche de la caméra a perdu une partie importante de son volume : pour des déformations aussi importantes, l'utilisation d'une grille plus fine serait à envisager. Au temps $t = 0.09$, nous pouvons voir l'apparition d'une fine éclaboussure issue du mouvement turbulent de la surface. La surface de l'eau se stabilise dans les temps suivants.

Je présente dans le tableau 4.1 la répartition du coût algorithmique dans le cas d'une grille de taille 64^3. Le coût du calcul pour une itération est d'environ 12 secondes lorsque le nombre de particules atteint 200.000. Ce taux de remplissage important est dû aux fortes turbulences présentes, surtout dans l'air d'où partent initialement les gouttes. Leur chute provoque des tourbillons importants, même après stabilisation, loin de la surface. Nous pouvons voir que le calcul du terme $\frac{1}{\rho} \times \nabla p$ (écrit sous cette forme pour alléger la lecture) est négligeable. Le transport de la surface par particules représente 23% du coût total dédié à la level set et la reinitialisation 77%, bien que, pour cette dernière, seulement 2 itérations soient effectuées à chaque pas de temps. J'ai comparé le temps de calcul nécessaire pour la méthode des particules Lagrangienne avec celle proposée dans [BGOS06] (la *semi-Lagrangian contouring method*) : le temps de calcul pour une boîte de même taille est similaire. Il est important de noter que le code que j'ai utilisé pour l'algorithme de reinitialisation n'a pas été optimisé. Il serait envisageable, par exemple, de résoudre l'équation uniquement dans l'entourage de l'interface.

Vitesse + diffusion	Advection	Remaillage
4.5%	14%	11%

Terme multiphase	Transport de la surface	Reinitialisation	Autre
4%	12.5%	41.5%	4.5%

TAB. 4.1 – Pourcentage du temps de calcul attribué à chaque partie de la résolution des équations, pour l'exemple de l'hélice à mouvement imposé. La taille de la grille est de 64^3, le nombre maximum de particules est 200.000 et 9 secondes sont nécessaires pour chaque itération. Le terme « advection »désigne ici le transport des particules de vorticité par le schéma RK2. Le « terme multiphase »représente le calcul de $\frac{1}{\rho} \times \nabla p$.

Nous pouvons voir dans cet exemple que la nouvelle méthode de simulation de fluides multiphases par méthode vortex, couplée avec un transport de level set par particules, permet d'obtenir des résultats visuellement réalistes, même pour des

variations de densité importantes. Il est important de noter que le pas de temps doit être choisi afin de capturer au mieux les phénomènes étudiés.

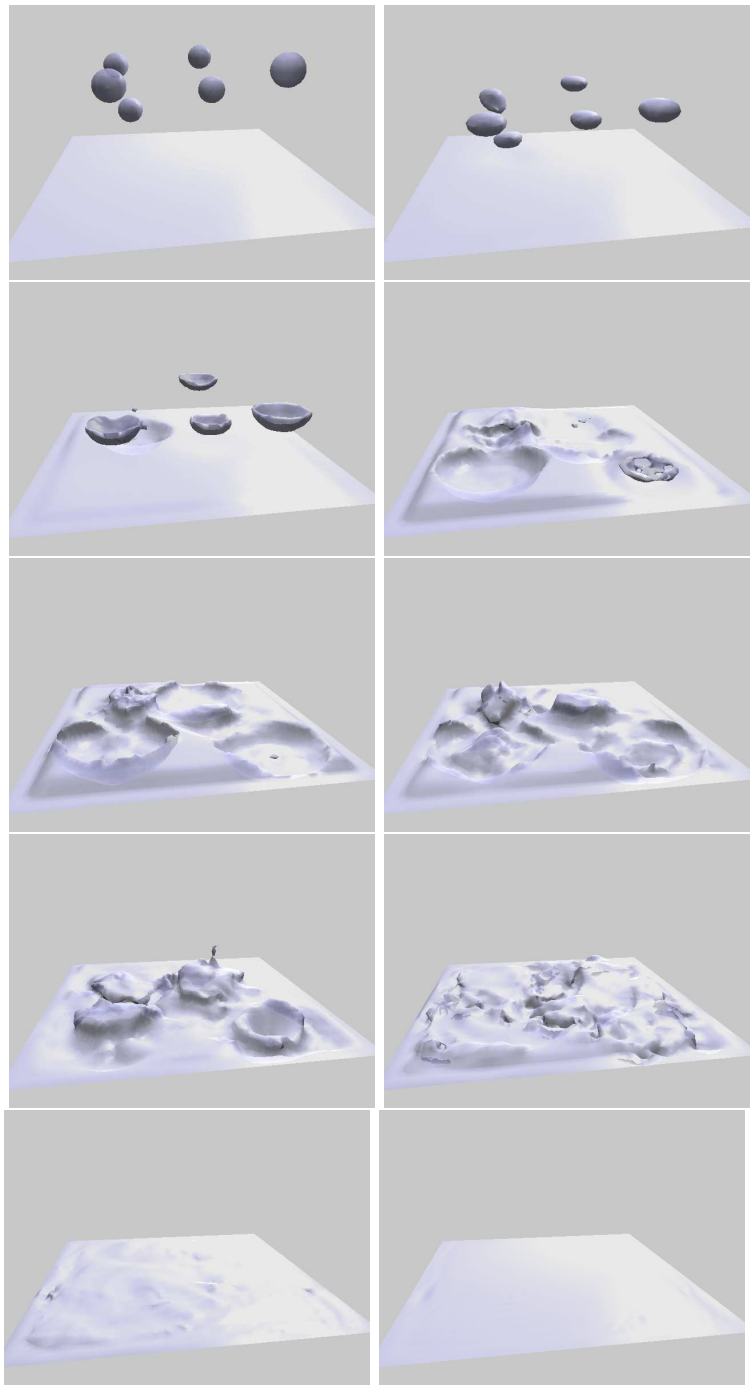


FIG. 4.6 – Images extraites de la simulation d'un fluide à deux phases où des gouttes d'eau tombent sur une surface plane initialement. Captures aux temps $t = 0, 0.025, 0.04, 0.07, 0.08, 0.09, 0.135, 0.3, 0.6$.

3 Bilan du chapitre

Nous venons de voir dans ce chapitre une adaptation d'une méthode vortex pour la simulation de fluides monophasiques et multiphasiques. Mes travaux, basés sur une implémentation efficace de la technique des particules de vorticité et principalement des *Vortex In Cell* couplés au remaillage, permet d'obtenir des résultats très dynamiques, même pour un nombre de points de discrétisation faible, en comparaison des méthodes utilisées communément en synthèse d'images. J'ai implémenté des méthodes efficaces et précises pour la résolution des différentes parties de l'équation de conservation du tourbillon, notamment grâce à l'utilisation de la tranformée de Fourier rapide dans le cas de conditions périodiques. Cette méthode permet de résoudre dans une même passe la diffusion de la vorticité dans le domaine fréquentiel de manière exacte, sans contrainte sur le pas de temps, ainsi que l'équation de Poisson pour le calcul de la vitesse par la fonction courant. L'utilisation de particules de vorticité pour la résolution de l'équation de transport est un avantage certain, contrecarrant le problème majeur provenant de l'utilisation de la méthode des caractéristiques pour le calcul de la convection en formulation vitesse. Ensuite, la technique du remaillage appliquée fréquemment, conjointement à l'emploi de fonctions d'interpolations adaptées assurant une faible dissipation numérique des vortex et ainsi de la dynamique générale du fluide, permet de palier les problèmes de mauvaise distribution des points de discrétisation dans une formulation purement particulaire. De plus, la résolution implicite de l'équation de transport sur les particules permet de réduire grandement la contrainte sur le pas de temps.

J'ai aussi présenté dans ce chapitre une méthode nouvelle pour l'intégration d'un fluide à plusieurs phases dans une formulation vortex de l'équation de Navier-Stokes. La prise en compte d'une variation de densité ajoute un terme à cette dernière, terme qui peut être calculé efficacement sur la grille à chaque pas de temps, puis interpolé sur les particules advectées. Pour le mouvement de l'interface, j'ai choisi d'utiliser une méthode de level set via une technique particulaire, avec remaillage, dans le même principe que pour les particules de vorticité. Le remaillage des particules de level set doit être effectué lorsque la distribution de celles-ci s'éloigne trop de l'uniformité. La déformation de la répartition des points de discrétisation suivant celle des particules de vorticité, l'étape de remaillage est effectué à chaque pas de temps, celui-ci étant souvent plus grand que la contrainte CFL. Toutefois, afin de calculer plus précisément le mouvement du fluide et de l'interface, il est nécessaire de considérer des pas de temps adaptés au problème étudié, dans le but de capter précisément les phénomènes issus de l'interaction des phases, phénomènes visuellement importants en synthèse d'images. Dans ce cas, l'utilisation d'un critère sur le taux de déformation du maillage pourrait être utilisé afin de réduire la fréquence du remaillage si nécessaire, et ainsi réduire les erreurs numériques introduites par cette technique.

Comme nous l'avons vu dans les chapitres 1 et 3, l'introduction d'objets en interaction avec le fluide dans une formulation vortex, est une question ardue. Je propose dans le chapitre suivant une technique innovante en méthode vortex pour le

calcul des interactions entre un fluide et des solides rigides, ainsi que des membranes déformables.

Chapitre 5

Couplage fluide-structure pour les méthodes vortex

1 Introduction

Nous venons de voir dans le chapitre précédent une méthode vortex pour la simulation de fluides à une ou plusieurs phases, nous permettant d'obtenir des résultats convaincants dans une formulation innovante en synthèse d'images. Dans ce domaine, afin d'obtenir des images encore plus attrayantes et réalistes, l'animation de fluides a rapidement nécessité l'intégration d'autres types d'objets : tout d'abord des solides rigides puis aujourd'hui des structures au comportement plus complexe comme des objets élastiques, ceux-ci interagissant avec un gaz, un liquide ou les deux à la fois. En mathématiques appliqués, le problème de l'interaction est depuis longtemps étudié et reste encore actuellement un sujet de recherche important du fait de sa difficulté. En effet, la différence de la dynamique des divers milieux en interaction demande une étude précise des comportements à l'interface entre ces derniers et des modèles adéquats doivent ainsi être développés afin de résoudre au mieux les équations concernées. De plus, les natures variées de la géométrie des surfaces des différents milieux — par exemple fixe pour un solide rigide et admettant des changements de topologie pour la surface d'un liquide — requiert d'adapter le modèle (et parfois la discrétisation de l'espace) afin de correspondre au mieux aux bords, et ainsi résoudre correctement les équations.

Dans le cadre des méthodes vortex, nous avons vu que l'incomporation de parois fixes au contact d'un fluide est un problème qu'il n'est pas aisé de résoudre du fait de la résolution des équations sur la vorticité et indirectement sur la vitesse. Les méthodes des panneaux et de flux de vorticité fournissent une solution à ce problème, mais celles-ci sont difficiles à mettre en oeuvre en 3D — pour la discrétisation de la surface ainsi que pour la résolution numérique — et nécessitent des calculs supplémentaires qui peuvent limiter leur utilisation dans des applications simples et rapides. Ceci s'avère être un critère important dans le domaine de la synthèse d'images.

J'introduis dans ce chapitre un nouveau modèle, par méthode vortex, pour l'in-

teraction de fluides avec des solides rigides soumis aux forces hydrodynamiques ainsi qu'avec des membranes élastiques plongées dans un fluide. Ce modèle est basé sur le principe de frontière immergée, dans lequel le domaine du fluide est étendu à l'intérieur des structures introduites afin de bénéficier de méthodes de résolution rapides pour le calcul du mouvement du fluide. De plus, de la même manière que pour l'incorporation d'une interface entre deux fluides, j'ai basé la représentation de ces structures sur une formulation *level set* fournissant l'avantage d'être beaucoup plus simple à manipuler et à intégrer dans un fluide qu'un maillage explicite (notamment en dimension 3), ainsi que de fournir une information utile pour le calcul des forces de collision entre solides et de la déformation d'une membrane, comme nous le verrons par la suite. Nous verrons que dans le cas particulier d'un schéma d'intégration d'Euler d'ordre un, la méthode de pénalisation est similaire à la méthode de domaine fictif selon le modèle présenté dans [GPH+01]. Le modèle de collision que j'ai développé pour les interactions entre solides s'intègre naturellement dans la résolution des équations générales pour un fluide avec structures immergées ; l'avantage majeur est qu'il est adapté à tout type de géométrie pour les solides. Sur le même principe, j'ai intégré des membranes élastiques, représentées par une fonction level set.

Je valide ces techniques sur des cas test étudiés dans le domaine des mathématiques appliquées et les applications en synthèse d'images nous montrent que, dans sa globalité, la méthode proposée est une alternative séduisante aux méthodes d'interaction traditionnelles en formulation vitesse et tourbillon.

En première partie de ce chapitre, je présenterai le modèle de pénalisation pour l'interaction fluide-structure et son adaptation à une résolution des équations de Navier-Stokes en formulation vortex. J'en démontrerai la validité par l'étude des résultats numériques obtenus sur ces cas tests, en les comparant aux résultats obtenus par les méthodes de type domaine fictif. Puis, en deuxième partie, j'exposerai le modèle collision entre solides développé ainsi que son intégration dans la méthode décrite précédemment. Finalement, je montrerai en troisième partie comment j'ai incorporé une membrane élastique en interaction dans un fluide. J'agrémenterai chaque section d'applications en synthèse d'images, dans lesquelles différents milieux intéragissent de manière réaliste avec des fluides à une ou plusieurs phases.

2 Modèle d'interaction fluide-solide rigide en formulation vortex

2.1 Présentation du modèle

Dans cette partie, j'explique l'adaptation du modèle de pénalisation pour le calcul d'interaction entre un fluide et un solide rigide en formulation vorticité. La généralisation à plusieurs solides en découle directement. La manière de résoudre ces équations, leur discrétisation et les particularités d'implémentation seront déve-

loppées par la suite dans la section 2.2.

Modèle physique

Nous étudions l'interaction d'un solide rigide $S(t)$ évoluant de manière libre, soumis aux forces hydrodynamiques, au contact d'un fluide incompressible contenu dans un domaine $F(t)$. Intéressons nous tout d'abord à la résolution du problème en formulation vitesse. L'interaction entre ces deux milieux peut être modélisée par les équations de Navier-Stokes dans le domaine du fluide, par un déplacement rigide dans $S(t)$ et par la continuité de la vitesse et de la déformation à l'interface :

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) = \nabla\cdot\sigma + \rho\mathbf{g} \quad \text{pour } \mathbf{x}\in F(t) \text{ et } t>0, \tag{5.1}$$

$$\mathbf{u} = \mathbf{V} + \mathbf{\Omega}\times(\mathbf{x}-\mathbf{x}_G) \quad \text{pour } \mathbf{x}\in S(t) \text{ et } t>0, \tag{5.2}$$

$$\mathrm{M}\dot{\mathbf{V}} = -\int_\Sigma \sigma\,\mathbf{n}\,d\mathbf{x} + \mathrm{M}\mathbf{g}, \tag{5.3}$$

$$\mathrm{J}\dot{\mathbf{\Omega}} = -\int_\Sigma \sigma\times(\mathbf{x}-\mathbf{x}_G)\,d\mathbf{x}, \tag{5.4}$$

avec $\nabla\cdot\mathbf{u}=0$ et où $\rho=\rho_F$ (resp. $\rho=\rho_S$) est la densité du fluide (resp. solide) dans $F(t)$ (resp. $S(t)$), $\mathbf{g}$ la gravité, $\mathbf{V}$ est la vitesse de translation du solide, $\mathbf{\Omega}$ sa vitesse de rotation, $\mathbf{x}_G$ son centre de gravité, M sa masse et J sa matrice d'inertie. Ici $\mathbf{n}$ est la normale du solide pointant vers l'intérieur et σ est le tenseur de déformation $\sigma = \mu\left(\nabla\mathbf{u}+\nabla\mathbf{u}^{\mathrm{T}}\right) - p\mathbf{I}$ (avec $\mathbf{I}$ la matrice identité de la même dimension que $\mathbf{u}$). $\Sigma(t)$ dénote ici l'interface entre le solide et le fluide : $\Sigma(t)=\partial S(t)$. Le système d'équations 5.1-5.4 doit être complété de conditions aux bords pour $F(t)$ (comme vu dans le chapitre précédent) et de conditions initiales pour les grandeurs étudiées. L'équation 5.2 donne la vitesse rigide du solide et le couple d'équations 5.3 et 5.4 fournissent l'accélération du solide soumis à la gravité et aux forces hydrodynamiques.

Modèle de pénalisation

Nous considérons alors, dans le principe des structures immergées, un domaine $K=S(t)+F(t)$ pour le fluide étendu à l'intérieur du solide, et pénalisons la vitesse du fluide dans $S(t)$ afin d'imposer un mouvement rigide. Notons χ_S la fonction caractéristique du solide et λ le paramètre d'utilisation tel que $\lambda\gg 1$, la pénalisation de l'équation de Navier-Stokes s'écrit alors :

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) = \nabla\cdot\sigma + \rho\mathbf{g} + \lambda\rho\left(\bar{\mathbf{u}}-\mathbf{u}\right) \quad \text{pour } \mathbf{x}\in K. \tag{5.5}$$

La vitesse $\bar{\mathbf{u}}$ du solide est alors déduite de celle du fluide :

$$\begin{aligned}\bar{\mathbf{u}}(\mathbf{x}) &= \mathrm{M}^{-1}\int_S \rho_S\,\mathbf{u}\,d\mathbf{x} + \left(\mathrm{J}^{-1}\int_S \rho_S\,\mathbf{u}\times(\mathbf{x}-\mathbf{x}_G)\,d\mathbf{x}\right)\times(\mathbf{x}-\mathbf{x}_G)\\ &= \mathrm{M}^{-1}\int_K \chi_S\rho\,\mathbf{u}\,d\mathbf{x} + \left(\mathrm{J}^{-1}\int_K \chi_S\rho\,\mathbf{u}\times(\mathbf{x}-\mathbf{x}_G)\,d\mathbf{x}\right)\times(\mathbf{x}-\mathbf{x}_G),\end{aligned} \tag{5.6}$$

sur l'intégralité du domaine, où la densité est définie comme

$$\rho = \rho_F(1 - \chi_S) + \rho_S \chi_S.$$

La fonction caractéristique du solide est alors transportée par la vitesse du solide via l'équation d'advection :

$$\frac{\partial \chi_S}{\partial t} + (\bar{\mathbf{u}} \cdot \nabla)\chi_S = 0. \tag{5.7}$$

Notons quelques remarques sur ce système d'équations. Dans le cas où $\bar{\mathbf{u}}$ est donné, ce schéma de pénalisation est identique au schéma L^2 présenté par Angot *et al.* dans [ABF99]. Il est aussi important de noter que, l'utilisation de la vitesse $\bar{\mathbf{u}}$ plutôt que $\mathbf{u}$ (les deux étant équivalents lorsque $\lambda \to \infty$) pour le transport de χ_S permet d'imposer un mouvement exactement rigide pour la fonction caractéristique. En pratique, ceci est important car elle évite toute déformation du solide à cause des erreurs numériques. Nous verrons par la suite que ceci permet d'appliquer un schéma rapide et exact (à une erreur d'interpolation prêt) pour le transport de χ_S.

Il convient maintenant de trouver une technique permettant de résoudre l'équation 5.5. Pour ce faire, j'utilise un algorithme de type fractionnaire, à la manière de la méthode de projection, où tout d'abord la vitesse incompressible est calculée puis pénalisée. Soient $\mathbf{u}^n = \mathbf{u}(t_n)$ et $\bar{\mathbf{u}}^n = \bar{\mathbf{u}}(t_n)$ les vitesses au temps t_n du fluide et du solide respectivement. L'algorithme avançant la solution du temps t_n au temps $t_{n+1} = t_n + \delta t$ est le suivant :

1. Résoudre l'équation de Navier-Stokes 5.1 avec la contrainte d'incompressibilité sur un pas de temps δt pour obtenir $\tilde{\mathbf{u}}^n$,
2. Calculer $\bar{\mathbf{u}}^n$ par l'équation 5.6 en utilisant $\tilde{\mathbf{u}}^n$ pour la vitesse du fluide,
3. Résoudre l'équation
$$\frac{\partial \mathbf{u}}{\partial t} = \lambda \chi_S(\bar{\mathbf{u}}^n - \mathbf{u}). \tag{5.8}$$

Dans le cas particulier où $\lambda = \delta t^{-1}$ et en adoptant un schéma d'Euler explicit pour discrétiser 5.8, le résultat obtenu est :

$$\mathbf{u}^{n+1} = \begin{cases} \tilde{\mathbf{u}}^n & \text{dans } F(t) \text{ et} \\ \bar{\mathbf{u}}^n & \text{dans } S(t). \end{cases}$$

Nous pouvons alors reconnaître la méthode de projection proposée par Sharma et Patankar [SP05].

Avantages de la méthode

Tout d'abord, l'équation linéaire 5.8 est beaucoup plus simple à résoudre que le calcul des multiplicateurs de Lagrange de la méthode décrite dans [GPH+01, SP05].

Elle est aussi plus intuitive et très aisée à intégrer dans un code existant pour la résolution des équations de Navier-Stokes. De plus, cette formulation permet de séparer aisément les erreurs de discrétisation (que nous verrons par la suite) des erreurs de modélisation, comme le montrent les études numériques proposées dans [Bos08]. Il est aussi envisageable d'utiliser une pénalisation H^1 (où tous les termes linéaires de l'équation de Navier-Stokes sont pénalisés) afin de mieux imposer la contrainte de rigidité à l'intérieur du solide ; néanmoins, cette solution n'a pas été adoptée, les résultats obtenus avec la pénalisation L^2, plus simple, étant satisfaisants. Enfin, il est possible de résoudre exactement l'équation 5.8 qui est une Équation Différentielle Ordinaire (EDO) linéaire d'ordre un, si on considère χ_S fixe sur un pas de temps. Je reviendrai sur ce point dans la partie 2.2.

A la différence des méthodes de projection qui s'avéreraient difficiles à dériver, la méthode de pénalisation se prête naturellement à la formulation vortex où elle apparaît comme un terme générateur de vorticité.

Modèle de pénalisation en formulation vortex

En prenant le rotationnel de l'équation 5.5 de pénalisation en formulation vitesse, nous obtenons :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = (\omega \cdot \nabla)\mathbf{u} + \nu \Delta \omega + \nabla \left(\frac{1}{\rho}\right) \times \nabla p + \lambda \nabla \times (\chi_S(\bar{\mathbf{u}} - \mathbf{u})) \,, \tag{5.9}$$

où pour une meilleure lisibilité, la viscosité μ est considérée comme constante et le terme ∇p doit être remplacé par $\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \mathbf{g}\right)$ comme expliqué dans le chapitre précédent à propos de la simulation de fluides à plusieurs phases. La vitesse du solide $\bar{\mathbf{u}}$ est calculée grâce à l'équation 5.6.

En dévelopant le terme $\nabla \times (\chi_S(\bar{\mathbf{u}} - \mathbf{u}))$, nous pouvons reécrire l'équation précédente comme :

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = (\omega \cdot \nabla)\mathbf{u} + \nu \Delta \omega + \nabla \left(\frac{1}{\rho}\right) \times \nabla p + \lambda \chi_S(\bar{\omega} - \omega) + \lambda \delta_\Sigma \, \mathbf{n} \times (\bar{\mathbf{u}} - \mathbf{u}), \tag{5.10}$$

où $\mathbf{n}$ est la normale de l'interface Σ et δ_Σ la masse de Dirac localisée sur celle-ci. Nous pouvons alors que le dernier terme est équivalent à celui présent dans les méthodes de flux de vorticité, permettant d'annuler la vitesse de glissement aux bords ; l'avant dernier terme quant à lui pénalise la vorticité à l'intérieur du solide.

Comme je le montre dans la partie suivante, cette méthode s'intègre aisément dans un code de simulation de fluide par méthode de vorticité, comme pour la méthode des Vortex In Cell que j'ai utilisé pour mes travaux.

2.2 Résolution des équations

Je présente dans cette partie la manière dont est calculée l'interaction entre un fluide et un solide rigide, par la résolution de l'équation de Navier-Stokes pénalisée. J'y explique ensuite comment le solide, représenté par une fonction level set est transporté.

Pénalisation en formulation vitesse

Dans le cadre de l'algorithme de type fractionnaire présenté plus haut, il est nécessaire de résoudre l'équation 5.8 (dans une formulation plus large) : $\frac{\partial \mathbf{u}}{\partial t} = \lambda\chi_S(\bar{\mathbf{u}} - \mathbf{u})$, avec $\lambda \gg 1$. Le terme $\bar{\mathbf{u}}$ défini par 5.6 est constitué de deux intégrales comprenant le produit de $\mathbf{u}$ avec χ_S et ρ. Ces deux dernière grandeurs évoluant en fonction du fluide par une équation de transport, 5.6 est donc non linéaire et ainsi il est difficile d'intégrer l'équation 5.8 avec une vitesse rigide autre que celle calculée au pas de temps courant. C'est pourquoi, par la suite, je considère $\bar{\mathbf{u}} = \bar{\mathbf{u}}^n$ constant sur la durée d'un pas de temps, ce qui est une approximation des intégrales raisonnable dans le cas où la vitesse du fluide dans S varie peu d'un pas de temps à l'autre. Dans le cas où le solide est fixe — c'est-à-dire non soumis à la gravité et aux forces hydrodynamiques — $\bar{\mathbf{u}} = 0$ et le problème en est donc simplifié. Je présente ci-dessous les méthodes de résolutions développées dans le cas général, le graphique en figure 5.1 fourni une représentation graphique des différents schémas d'intégration utilisés.

La première approche que j'ai adoptée a consisté à intégrer $\frac{\partial \mathbf{u}}{\partial t}$ de manière explicite à l'ordre un en posant

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \delta t\lambda\chi_S(\bar{\mathbf{u}}^n - \mathbf{u}^n). \tag{5.11}$$

Cette formulation est soumise à une contrainte de stabilité $\lambda < 2\delta t^{-1}$, ceci se vérifie simplement : en posant $\bar{\mathbf{u}} = 0$ par exemple, nous avons $\mathbf{u}^{n+1} = (1 - \delta t\lambda\chi_S)\mathbf{u}^n$, et ainsi pour satisfaire $|\mathbf{u}^{n+1}| < |\mathbf{u}^n|$, nous devons avoir $(1-\delta t\lambda\chi_S) < 1$ si et seulement si $1-\delta t\lambda\chi_S < 1$ (toujours vrai) et $-(1-\delta t\lambda\chi_S) < 1$ qui, dans le pire cas où $\chi_S = 1$, nous donne la condition de stabilité énoncée. Afin de palier ce problème restrictif sur le choix de λ qui nous demanderai de choisir de petits pas de temps afin de pénaliser correctement la vitesse fluide, j'ai implémenté un schéma itératif consistant à diviser δt en pas de temps fictifs et recalculer une vitesse intermédiaire à chaque itération. Ceci n'est pas trop contraignant en formulation vitesse mais l'est beaucoup plus dans une formulation vortex (dont la dérivation est fournie dans la sous-section suivante) où la vitesse doit être recalculée par la résolution de l'équation de Poisson ou via le noyau de Biot-Savart pour chaque sous-itération.

La contrainte de stabilité sur λ peut être supprimée en adoptant un schéma implicite. En effet, à l'ordre 1, en posant

$$\begin{aligned} \mathbf{u}^{n+1} &= \mathbf{u}^n + \delta t\lambda\chi_S(\bar{\mathbf{u}}^n - \mathbf{u}^{n+1}) \\ &= \frac{\mathbf{u}^n + \delta t\lambda\chi_S\bar{\mathbf{u}}^n}{1 + \delta t\lambda\chi_S}, \end{aligned} \tag{5.12}$$

la méthode devient inconditionnellement stable. La preuve est triviale lorsque $\bar{\mathbf{u}} = 0$. Dans le cas général, étudions la convergence de la vitesse du fluide vers la vitesse imposée $|\mathbf{u} - \bar{\mathbf{u}}|$, et posons $\delta t\lambda\chi_S = \alpha$:

$$|\mathbf{u}^n - \bar{\mathbf{u}}| \quad > \quad |\mathbf{u}^{n+1} - \bar{\mathbf{u}}|$$

$$\begin{aligned} &> \left|\frac{\mathbf{u}^n - \alpha\bar{\mathbf{u}}}{1+\alpha} - \bar{\mathbf{u}}\right| \\ |\mathbf{u}^n - \bar{\mathbf{u}}| &> \left|\frac{\mathbf{u}^n - \bar{\mathbf{u}}}{1+\alpha}\right| \end{aligned}$$

qui est toujours vrai quelquesoit $\alpha > 0$.

Afin d'augmenter l'ordre du schéma, il serait envisageable d'adopter un schéma de type Crank-Nicholson :

$$\begin{aligned} \mathbf{u}^{n+1} &= \mathbf{u}^n + \frac{1}{2}\delta t\lambda\chi_S\left((\bar{\mathbf{u}}^n - \mathbf{u}^{n+1}) + (\bar{\mathbf{u}}^{n+1} - \mathbf{u}^{n+1})\right) \\ &= \frac{(1+\delta t\lambda\chi_S(\bar{\mathbf{u}}^n - 1/2\mathbf{u}^n}{1+1/2\delta t\lambda\chi_S} \end{aligned}$$

qui est inconditionnellement stable lui aussi (prouvable par la même méthode que précédemment). Par contre, ce schéma converge très mal pour $\lambda \gg 1$ et oscille même pour $\delta t\lambda\chi_S > 2$. Ce que l'on prouve en posant $\bar{\mathbf{u}} = 0$, le facteur pénalisant la vitesse $\mathbf{u}^n$ tend donc vers :

$$\lim_{\lambda\to\infty} \frac{1-1/2\delta t\lambda\chi_S}{1+1/2\delta t\lambda\chi_S} = -1,$$

pour $\chi_S \neq 0$. Malgré tout, il reste meilleur que le schéma implicite pour $\delta t\lambda\chi_S \leq 2$.

Cependant, il n'est pas nécessaire d'utiliser un quelconque schéma de discrétisation puisque l'équation étudiée est une équation différentielle ordinaire linéaire et qu'il est alors possible de résoudre analytiquement. En effet, l'équation

$$\frac{\partial\,\mathbf{u}}{\partial t} = \lambda\chi_S(\bar{\mathbf{u}} - \mathbf{u})$$

peut s'écrire sous la forme

$$\frac{d\,\mathbf{u}}{dt} + a\mathbf{u} = \mathbf{b},$$

avec $a = \lambda\chi_S$ et $\mathbf{b} = \lambda\chi_S\bar{\mathbf{u}}$. Cette EDO admet une solution générale lorsque a ne dépend pas de $\mathbf{u}$:

$$\mathbf{u}(t) = \left(\mathbf{u}_0 + \int_0^t \mathbf{b}(\tau)e^{a\tau}\,d\tau\right)e^{-at} \tag{5.13}$$

où $\mathbf{u}_0$ est la solution initiale $\mathbf{u}_0 = \mathbf{u}(0)$, équation qui, en admettant que $\mathbf{b}$ est constant dans l'intervale $[0,t]$ conduit à :

$$\mathbf{u}(t) = \left(\mathbf{u}_0 + \mathbf{b}\frac{1}{a}(e^{at}-1)\right)e^{-at} \tag{5.14}$$

si $a \neq 0$ (c'est-à-dire à l'intérieur du solide, avec $\delta t > 0$ et $\lambda > 0$) et à

$$\mathbf{u}(t) = (\mathbf{u}_0 + \mathbf{b}t)\,e^{-at}$$

autrement. Dans ce dernier cas, comme $\chi_S = 0$, la solution est simplement :

$$\mathbf{u}(t) = \mathbf{u}_0. \tag{5.15}$$

Comme nous l'avons vu, la solution 5.14 est correcte si et seulement si a ne dépend pas de $\mathbf{u}$. a dépend de $\mathbf{u}$ puisque χ_S est transporté par le fluide, mais dans le cas considéré précédemment où la fonction caractéristique est fixe sur un pas de temps, cette solution exacte est toujours meilleure que celle donnée par les précédents schémas, qui sont en fait des approximations de cette dernière. Comme nous connaissons au pas de temps courant la vitesse $\bar{\mathbf{u}}$ du solide, il serait envisageable de mieux approcher l'intégrale avec une formule de quadrature plus précise, en fixant cette vitesse pour le pas de temps et en considérant l'évolution de χ_S par l'équation de transport. Néanmoins, les résultats obtenus en considérant χ_S fixe sur le pas de temps étant tout à fait satisfaisants (voir partie 2.3), je n'ai pas jugé nécessaire d'adopter une autre approximation.

La solution de l'étape de pénalisation en formulation vitesse, pour avancer une vitesse $\mathbf{u}^n$ au pas de temps suivant est donc :

$$\begin{aligned}
\mathbf{u}^{n+1} &= \left(\mathbf{u}^n + \lambda\chi_S\bar{\mathbf{u}}\frac{1}{\lambda\chi_S}(e^{\lambda\chi_S\delta t}-1)\right)e^{-\lambda\chi_S\delta t} \\
&= \left(\mathbf{u}^n + \bar{\mathbf{u}}(e^{\lambda\chi_S\delta t}-1)\right)e^{-\lambda\chi_S\delta t} \\
&= (\mathbf{u}^n - \bar{\mathbf{u}})\,e^{-\lambda\chi_S\delta t} + \bar{\mathbf{u}} \quad \text{pour } \chi_S > 0 \text{ et} && (5.16)\\
\mathbf{u}^{n+1} &= \mathbf{u}^n \quad \text{sinon.} && (5.17)
\end{aligned}$$

Numériquement, il est préférable de développer l'équation 5.16 afin d'éviter que le produit $e^{\lambda\chi_S\delta t}e^{-\lambda\chi_S\delta t}$ ne soit pas égal à un du fait des erreurs numériques dans le calcul de l'exponentielle, principalement lorsque $\lambda\chi_S\delta t \to \infty$. De plus, on peut ainsi aisément se rendre compte qu'à l'intérieur du solide, la vitesse du fluide tend de manière exponentielle vers $\bar{\mathbf{u}}$ lorsque $\lambda \to \infty$.

En présence de plusieurs solides de vitesse respective $\bar{\mathbf{u}}_i$ et fonction caractéristique χ_{S_i}, l'équation de pénalisation dans l'algorithme fractionnaire s'écrit :

$$\frac{\partial\,\mathbf{u}}{\partial t} = \sum_i \lambda\chi_{S_i}(\bar{\mathbf{u}}_i - \mathbf{u}). \tag{5.18}$$

Et, en remplaçant de manière adéquate a et $\mathbf{b}$ utilisé précédemment, la vitesse au pas de temps suivant est obtenue par :

$$\begin{aligned}
\mathbf{u}^{n+1} &= \left(\mathbf{u}^n - \frac{\sum_i \chi_{S_i}\bar{\mathbf{u}}_i}{\sum_i \chi_{S_i}}\right) exp^{\left(-\lambda\delta t\sum_i \chi_{S_i}\right)} + \frac{\sum_i \chi_{S_i}\bar{\mathbf{u}}_i}{\sum_i \chi_{S_i}} \quad \text{pour } \sum_i \chi_{S_i} > 0 \text{ et} && (5.19)\\
\mathbf{u}^{n+1} &= \mathbf{u}^n \quad \text{sinon.} && (5.20)
\end{aligned}$$

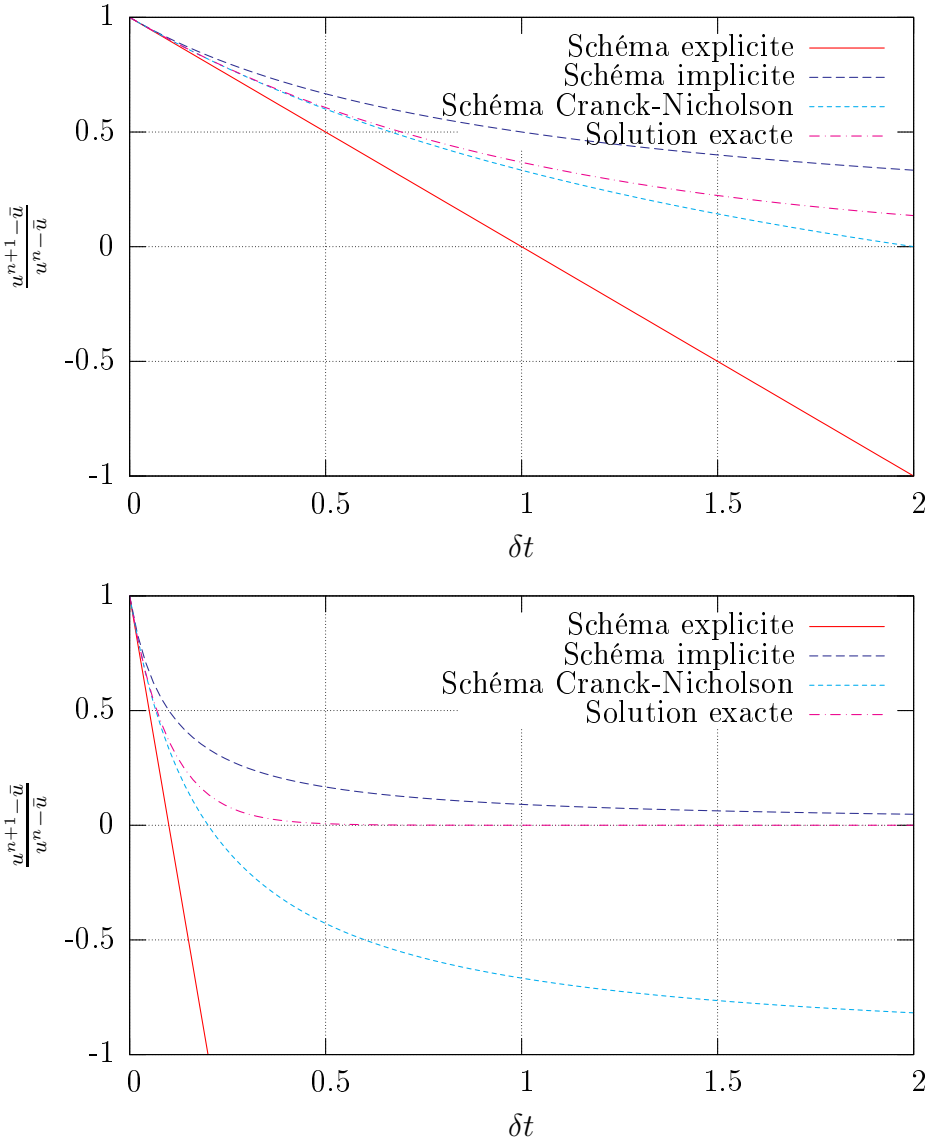


FIG. 5.1 – Convergence de différents schémas d'intégration en comparaison de la solution exacte pour l'équation de pénalisation dans un algorithme fractionnaire. Ici $\bar{u} = 0$, $\chi_S = 1$ et $\lambda = 1$ pour la figure en haut et $\lambda = 10$ pour la figure en bas.

Pénalisation en formulation vorticité

Revenons maintenant à la formulation vorticité des équations de Navier-Stokes. En pratique, j'ai choisi de résoudre l'équation 5.9 présentée précédemment, d'une manière similaire à l'algorithme proposé pour la formulation vitesse, en calculant la vorticité au pas de temps suivant en fonction de la vitesse pénalisée au pas de temps suivant, soit :

$$\begin{aligned} \omega^{n+1} &= \nabla \times \mathbf{u}^{n+1} \\ &= \nabla \times \left[\left(\mathbf{u}^n - \bar{\mathbf{u}} \right) e^{-\lambda \chi_S \delta t} + \bar{\mathbf{u}} \right] \end{aligned} \tag{5.21}$$

dans le cas d'un seul solide pour plus de lisibilité. Le calcul de cette vorticité pénalisant la vitesse est introduit dans l'algorithme de type *VIC* avec remaillage, à la suite du calcul de la vitesse $\mathbf{u}^n$. L'équation 5.21 est ainsi calculée numériquement sur la grille par différences finies. La vitesse du solide est calculée au préalable à partir de la vitesse $\mathbf{u}^n$ comme présenté en équation 5.6.

L'algorithme que j'ai utilisé est donc le suivant :

1. Calcul de la vitesse sur la grille à partir de ω^n.
2. Calcul de la vitesse $\bar{\mathbf{u}}^n$ du solide par l'équation 5.6.
3. Calcul d'une vorticité intermédiaire $\tilde{\omega}^{n+1}$ par l'équation 5.21, sur la grille à partir de ω^n. $\tilde{\omega}^{n+1}$ remplace ω^n dans les étapes suivantes.
4. Calcul de l'étirement sur la grille par l'équation 4.4.
5. Interpolation de la vitesse $\mathbf{u}^n$ et de l'étirement sur les particules.
6. Advection des particules et intégration du terme d'étirement.
7. Remaillage : création de nouvelles particules basées sur la grille.
8. Diffusion sur la grille : obtention de ω^{n+1}.

Quelques précisions sont à apporter quant à la résolution de l'équation 5.21. Il serait envisageable de développer cette dernière afin d'éviter de calculer explicitement le rotationnel de $\mathbf{u}$. Par exemple, dans le cas d'un solide à vitesse nulle imposée, l'équation de pénalisation en formulation vorticité est :

$$\nabla \times \left(\mathbf{u}^n e^{-\lambda \chi_S \delta t}\right) = \left(\omega^n - \lambda \delta t \nabla \chi_S \times \mathbf{u}^n\right) e^{-\lambda \chi_S \delta t}.$$

Ainsi, aucune différence finie n'est nécessaire sur la vitesse $\mathbf{u}$, ce qui évite l'introduction d'erreurs numériques supplémentaires qui pourraient naître même au delà du solide, si ω n'est pas strictement égal à $\nabla \times \mathbf{u}$. Il est toutefois nécessaire de calculer le gradient de la fonction caractéristique $\nabla \chi_S = \delta_\Sigma \mathbf{n}$ qui est singulière. Même si cette formulation est plus élégante, j'ai trouvé qu'en pratique elle restreint la plage du paramètre λ à des valeurs de l'ordre de δt^{-1} au dela duquel de très fortes valeurs pour la vorticité apparaissent à l'interface, déstabilisant la simulation. Cette limitation est à mon sens due à deux facteurs : tout d'abord l'évaluation numérique de $\nabla \chi_S$ en conjonction de l'utilisation de χ_S peut, si elle est soumise à des erreurs importantes, introduire des erreurs élevées sur le champ de vorticité. Ensuite, comme numériquement $\omega \neq \mathbf{u}$, l'utilisation couplée de ces deux grandeurs dans la même équation peut induire des erreurs supplémentaires.

En pratique, j'ai donc calculé explicitement le rotationnel de l'équation de pénalisation sur la grille par un schéma aux différences finies. Les erreurs supplémentaires introduites par cette technique ne se sont pas avérées problématiques et la méthode permet d'obtenir de bons résultats, comme nous le verrons dans la partie 2.3.

Une remarque importante concernant le choix d'un algorithme fractionnaire en conjonction de la pénalisation. Si on considère une vitesse rigide $\bar{\mathbf{u}} = 0$, la vitesse du fluide ne sera complètement annulée (dans la limite où $\lambda \to 0$) qu'à l'itération

suivante, lors du calcul de $\mathbf{u}^{n+1}$. Cependant, la résolution du reste de l'équation de conservation du tourbillon , notamment en ce qui concerne l'étirement et la diffusion, modifie la vorticité. Ainsi, la vitesse à l'intérieur du solide ne sera pas nulle, même si on soustrait les erreurs numériques dues à la résolution de l'équation de Poisson sur ψ et au calcul de $\nabla \times \psi$ par différences finies. S'il s'avère nécessaire d'avoir une vitesse du fluide plus proche de celle du rigide à l'intérieur de ce dernier, il est possible de recalculer une vitesse intermédiaire $\tilde{\mathbf{u}}^{n+1}$, à partir de $\tilde{\omega}^{n+1}$, juste après la phase de pénalisation de la vorticité. Ceci augmente néanmoins le temps de calcul, et, dans mes expérimentations, la différence sur les résultats obtenus (comme par exemple dans le cas de la sédimentation d'une sphère que je présenterai par la suite) est minime.

Dans une formulation vitesse, il existe un problème similaire. En effet, dans un algorithme de type fractionnaire où est calculée la pression afin d'assurer l'incompressibilité du fluide, deux choix se présentent : pénaliser la vitesse après ou avant la projection. Dans le premier cas, le terme de pénalisation $\frac{d\mathbf{u}}{dt} = -\lambda\chi_S\mathbf{u}$ (je me suis placé dans le cas d'un obstacle fixe pour plus de simplicité) conduit à l'ajout d'une vitesse dont la divergence est non nulle sur le bord : $\nabla \cdot \frac{d\mathbf{u}}{dt} = -\lambda(\mathbf{u}\cdot\nabla)\chi_S$ (où la vitesse $\mathbf{u}$ est considérée comme incompressible à cette étape de l'algorithme). La vitesse alors obtenue après intégration du terme de pénalisation satisfait alors $\mathbf{u} = 0$ à l'intérieur du solide (dans la limite où $\lambda \to 0$), mais viole la contrainte d'incompressibilité, de manière proportionnelle à λ. Ceci peut poser des problèmes numériques, notamment pour l'équation de transport. Dans le second cas où la pénalisation est traitée avant, la vitesse obtenue à l'intérieur du solide, après l'étape de projection, ne sera pas annulée du fait des erreurs numériques commises lors de la résolution de l'équation de Poisson sur p.

Calcul de la vitesse du solide

Au début de chaque pas de temps, il est nécessaire de calculer la vitesse du solide $\bar{\mathbf{u}}$ afin de l'injecter dans l'étape suivante de pénalisation. Reprenons l'équation permettant de calculer cette vitesse à partir de celle du fluide :

$$\begin{aligned}\bar{\mathbf{u}} &= \mathbf{V} + \mathbf{\Omega} \times (\mathbf{x} - \mathbf{x}_G) \\ &= \mathrm{M}^{-1}\int_K \chi_S \rho\, \mathbf{u}\, d\mathbf{x} + \left(\mathrm{J}^{-1}\int_K \chi_S \rho\, \mathbf{u} \times (\mathbf{x}-\mathbf{x}_G)\, d\mathbf{x}\right) \times (\mathbf{x}-\mathbf{x}_G) \quad (5.22)\end{aligned}$$

avec $\mathbf{x}_G$ le centre de gravité du solide. La masse du solide est calculée simplement par

$$\mathrm{M} = \int_K \chi_S \rho_S = \rho_S \int_K \chi_S\, d\mathbf{x}$$

où pour rappel K est le domaine total contenant le fluide et encapsulant le solide et ρ_S est la densité du solide supposée constante. La matrice d'inertie est calculée par la relation :

$$\mathrm{J} = \int_K \chi_S \rho_S \left((\mathbf{r}\cdot\mathbf{r})\mathbf{I}_d - \mathbf{r}\otimes\mathbf{r}\right)\, d\mathbf{x}$$

où $\mathbf{r} = \mathbf{x} - \mathbf{x}_G$ et où $\otimes$ dénote le produit tensoriel, et $\mathbf{I}d$ est la matrice identité. L'inverse de cette matrice de taille 3×3 (en 3D) est aisé à calculer et ceci est effectué une seule fois pour chaque solide.

En pratique, afin d'éviter les problèmes d'imprécisions dus à la discrétisation et stabiliser la résolution des équations avec des grandeurs lisses, la fonction caractéristique χ_S est régularisée par un paramètre ε, avec $\varepsilon \to 0$. C'est pourquoi numériquement, dans l'évaluation de l'équation 5.22, c'est la densité $\rho_\varepsilon = \rho_F(1-\chi_{\varepsilon,S}) + \rho_S \chi_{\varepsilon,S}$ (où $\chi_{\varepsilon,S}$ dénote la fonction caractéristique régularisée) que j'ai utilisée.

Déplacement du solide par méthode level set

Le solide est localisé dans l'espace à chaque instant par sa fonction caractéristique χ_S. Au lieu de transporter directement celle-ci, j'ai choisi d'utiliser une fonction level set ϕ de laquelle sera déduite χ_S. J'ai choisi d'utiliser une fonction distance signée représentant la distance à l'interface, négative à l'intérieur du solide et positive à l'extérieur. Ce choix a été motivé par deux raisons principales : dans un souci de simplicité, toutes les interfaces interagissant avec le fluide (comme l'interface entre deux fluides, entre fluide et solide et entre fluide et membrane déformable comme nous le verrons par la suite) sont basées sur la même représentation implicite. Ensuite, point plus important, cette fonction nous donne une information capitale pour le développement de la méthode de collision présentée en partie 3 : elle nous permettra de calculer rapidement la distance séparant deux solides.

La fonction caractéristique est définie en fonction de ϕ par la relation :

$$\chi_S = H(\phi)$$

où H est la fonction Heaviside. La level set est transportée par le fluide et il est alors nécessaire de résoudre l'équation d'advection :

$$\frac{\partial \phi}{\partial t} + (\bar{\mathbf{u}} \cdot \nabla)\phi = 0, \phi(\cdot, 0) = \phi_0$$

Le transport de ϕ est effectué avec une vitesse rigide dans tout le domaine, cette propriété permet de conserver une fonction distance à chaque instant. Malheureusement la résolution de cette équation sur une grille, bien que le mouvement soit rigide, induirait indéniablement des erreurs dues à la discrétisation de l'équation. Pour palier ce problème, une méthode de type Lagrangienne similaire à celle présentée dans le chapitre 4 pour le transport de l'interface du fluide, mais sans remaillage — il serait inutile puisque le mouvement est rigide et ainsi le maillage ne se déforme pas dans le repère du solide — serait adaptée car elle permet de conserver exactement la fonction distance dans ce cas particulier.

Mais afin de réduire le temps de calcul nécessaire au transport de ϕ, j'ai décidé d'utiliser la propriété de rigidité du déplacement, nous ramenant ainsi à une simple interpolation du champ ϕ_0 initial en suivant en arrière la caractéristique $X(0; \mathbf{x}, t)$ du champ de vitesse :

$$\phi(\mathbf{x}, t) = \phi_0(\mathbf{X}(0; \mathbf{x}, t)). \tag{5.23}$$

Sous l'hypothèse de la rigidité du mouvement, les vitesses de translation et angulaire du solide permettent aisément de définir sa position et son orientation à chaque pas de temps en fonction de sa position initiale. En effet, si on considère la vitesse $\bar{\mathbf{u}}$ comme constante sur le pas de temps $[t^n, t^{n+1}]$, on peut alors déduire la position $\mathbf{X}^{n+1} = \mathbf{X}(t^{n+1}; \mathbf{x}, 0)$ (défini par la fonction caractéristique traçant la trajectoire de $\mathbf{x}$ en avant dans le temps) en fonction de $\mathbf{X}^n$, par une translation et une rotation. Soient

$$\Theta^n = |\mathbf{\Omega}^n|\delta t, \ \frac{\mathbf{\Omega}^n}{|\mathbf{\Omega}^n|} = (\alpha, \beta, \gamma)$$

qui représentent respectivement le pas de rotation et le vecteur normalisé de rotation autour duquel le solide tourne. La matrice de rotation $\mathcal{R}^n$ est définie par

$$\mathcal{R}^n = \begin{bmatrix} 1-2b^2-2z^2 & 2ab-2cd & 2ac+2bd \\ 2ab+2cd & 1-2a^2-2c^2 & 2bc-2ad \\ 2ac-2bd & 2bc+2ad & 1-2a^2-2b^2 \end{bmatrix}$$

avec

$$a = \alpha sin\frac{\Theta^n}{2}, b = \beta sin\frac{\Theta^n}{2}, c = \gamma sin\frac{\Theta^n}{2}, d = cos\frac{\Theta^n}{2}.$$

La position au temps t^{n+1} de n'importe quel point $\mathbf{X}$ de l'espace se déplaçant avec le solide peut donc s'écrire en fonction de sa position précédente :

$$\mathbf{X}^{n+1} = \mathbf{X}^n + \delta t V^n + \mathcal{R}^n(\mathbf{X}^n - \mathbf{x}_G^n),$$

et par accumulation des transformations :

$$\mathbf{X}^{n+1} = \mathcal{M}^{n+1}(\mathbf{x} - \mathbf{x}_G^0) + \mathbf{x}_G^{n+1}$$

où $\mathcal{M}$ est la matrice des rotations accumulées telle que $\mathcal{M}^{n+1} = \Pi_{i=0}^n \mathcal{R}^i$. La position du centre de gravité est obtenue par simple translation, comme la rotation est nulle en ce point.

Ainsi, il est rapide de calculer en chaque point $\mathbf{X}^{n+1}$ de la grille, la valeur de la level set en fonction des transformations accumulées :

$$\phi(\mathbf{X}^{n+1}, t^{n+1}) = \phi_0(\mathbf{x}) = \phi_0(\frac{1}{\mathcal{M}^{n+1}}(\mathbf{X}^{n+1} - \mathbf{x}_G^{n+1}) = \mathbf{x}_G^0. \tag{5.24}$$

Il n'est pas nécessaire de calculer explicitement l'inverse de la matrice $\mathcal{M}$. J'ai multiplié à chaque pas de temps la matrice stockée au pas de temps précédant par l'inverse de la rotation $\mathcal{R}^n$ facilement calculable. En pratique, j'ai utilisé une fonction bilinéaire pour l'interpolation du champ initial ϕ_0.

Cette technique possède deux atouts majeurs. En premier, l'erreur commise sur le transport à chaque pas de temps est constante et dépend uniquement du choix de la fonction d'interpolation. Secondement, la résolution de l'équation de transport est très rapide et s'étend à plusieurs solides rigides de manière triviale, pour un

coup unitaire très faible, réduit à une multiplication matrice vecteur. La partie la plus coûteuse de la méthode réside dans les interpolations, celles-ci pouvant être optimisées : si on considère uniquement une bande autour de l'interface en plus de l'intérieur du solide, il n'est pas nécessaire d'interpoler ϕ_0 pour les points trop éloignés. Le choix d'une fonction d'interpolation linéaire pourrait suffire si la précision de la localisation de l'interface n'est pas considéré comme un paramètre important pour la simulation.

2.3 Validation numérique

Des tests de validation de la méthode de pénalisation de la vitesse en formulation vorticité ont été effectués par Magni [Mag07] sur le problème 2D de l'écoulement autour d'un cylindre fixe. Dans cette partie, j'étudie le problème de la sédimentation d'une sphère plongée dans un fluide et soumise à la force gravitationnelle ainsi qu'aux forces hydrodynamiques. Relachée initialement avec une vitesse nulle, la sphère va accélérer jusqu'à atteindre une vitesse terminale fixe ; ce phénomène est dû à la présence de frottements entre le fluide et les parois du solide. Théoriquement, il existe une solution analytique dans le cas d'un fluide dans un domaine de taille infini. Ce problème a été étudié expérimentalement (voir [MP00,BL03] et les références de ces articles), dans le cas de sphères tombant dans un canal cylindrique. Les parois du cylindre freinant la chute de la sphère, Brown et Lawler [BL03] fournissent la correction à apporter à la vitesse terminale en fonction du rapport $\alpha = d/L$ entre le diamètre de la sphère et la largeur L du canal. Plus ce rapport est faible et plus la vitesse terminale est réduite, ce phénomène est à imputer au ralentissement du fluide à l'approche des bords du cylindre.

Glowinski *et al.* [GPH+01] ont calculé numériquement — en se basant sur leur méthode de domaine fictif, avec une résolution par éléments finis — les vitesses terminales de sphères plongées dans des fluides en faisant varier la densité et la viscosité de ces derniers et le diamètre du solide. Les résultats que je fournis ici sont basés sur les mêmes paramètres que ces cas tests, en utilisant la technique de pénalisation basée sur la méthode VIC avec remaillage décrite précédemment.

La boîte de calcul est de taille $[0,1] \times [0,1] \times [0,4]$ sur les bords de laquelle une condition de non pénétration est imposée sur la vitesse. Un cylindre fixe de diamètre 1 est plongé dans le fluide, et la condition de non glissement sur ses bords est imposée par une méthode de pénalisation. La densité du fluide est de 1, la gravité $\mathbf{g} = (0, 0, -980)^T$ et le paramètre de pénalisation $\lambda = 10^6$. Tout d'abord, j'ai effectué des tests de convergence en pas d'espace sur une sphère de densité 1.02. La figure 5.2 présente les vitesses terminales obtenues au temps $t = 1.2$ en fonction de $\delta x = h$, pour une sphère de diamètre $d = 0.3$ dans un fluide de viscosité $\nu = 0.02$. Ce graphique montre que notre méthode converge à l'ordre 1 en espace vers une vitesse terminale limite lorsque $h \to 0$.

Le tableau 5.1 montre une comparaison entre nos résultats (dans la limite où $h \to 0$), les résultats expérimentaux avec correction de la vitesse terminale du fait des parois et les résultats obtenus par la méthode de domaine fictif dans [GPH+01].

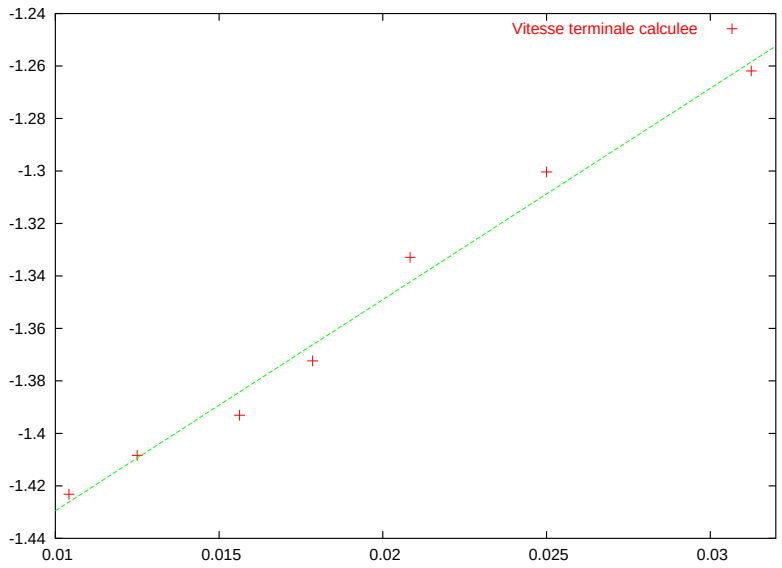


FIG. 5.2 – Vitesses terminales en fonction du pas d'espace h pour le problème de la sédimentation d'une sphère de diamètre 0.3 et de densité 1.02 tombant dans un fluide de densité 1 et de viscosité 0.02.

Les diamètres de la sphère considérés sont $d = 0.2, 0.3$, plongée dans des fluides dont la viscosité croissante est $\nu = 0.02, 0.05, 0.1$. Les résultats que nous avons obtenus sont proches de ceux mesurés expérimentalement, avec un taux d'erreur de l'ordre de 5-11%. Une des raisons possibles pouvant influencer sur les résultats sont les conditions aux bords empêchant toute fuite du fluide en haut et en bas du domaine, ce qui peut modifier la dynamique de l'écoulement. Pour les mêmes cas test, j'ai précisé les vitesses terminales données dans [GPH+01] pour des simulations avec $h = 1/64$. Afin de pouvoir comparer notre méthode avec cette dernière, j'ai aussi précisé dans le tableau 5.1 nos vitesses terminales pour $h = 1/64$; les résultats obtenus sont très proches.

L'analyse de ces résultats valide donc l'approche proposée qui, pour la première fois en formulation vortex, permet de simuler simplement (par la méthode de type frontière immergée) et précisément l'interaction entre une sphère et un fluide par une méthode de type domaine fictif. Les résultats obtenus sont qualitativement similaires à ceux de la méthode traditionnelle de type domaine fictif en formulation vitesse, nous amenant à considérer la méthode de pénalisation comme une alternative intéressante en formulation vortex.

2.4 Applications en synthèse d'images

Les exemples ci-dessous montrent des applications de la méthode de pénalisation en formulation vortex en 3D sur des problèmes intéressants en synthèse d'images. Le

diamètre	viscosité	U_E	$U_{h\to 0}$	Erreur relative	$U_{h=1/64}$	U_G
0.2	0.10	0.2571	0.275	8%	0.256	0.2567
0.2	0.05	0.4603	0.513	11%	0.475	0.4844
0.2	0.02	0.9129	1.016	11%	0.937	0.9480
0.3	0.10	0.4047	0.435	7%	0.401	0.4072
0.3	0.05	0.7493	0.795	6%	0.748	0.7599
0.3	0.02	1.4359	1.51	5%	1.39	1.392

TAB. 5.1 – Comparaison entre les vitesses terminales U de sphères tombant dans un fluide, où la viscosité et le diamètre de la sphère sont variés. Les résultats expérimentaux sont notés U_E, U_G ceux obtenus dans [GPH+01] et $U_{h=1/64}$ nos résultats avec $h = 1/64$. L'erreur relative est calculée entre les données expérimentales et nos résultats asymptotiques.

premier exemple concerne le mouvement imposé d'une hélice (ou d'un ventilateur) générant un champ de vitesse dans une boîte fermée. Le deuxième exemple implique le même objet géométrique à déplacement libre devant lequel est imposé un champ de vitesse amenant la rotation de l'hélice autour de son centre de gravité. Le dernier exemple proposé montre les possibilités de la méthode dans le cadre d'interactions entre un solide rigide et un fluide à plusieurs phases.

Hélice à mouvement imposé

Nous nous intéressons ici au cas d'une hélice tournant à une vitesse imposée autour de son centre de gravité. Ce mouvement créé une aspiration de l'air situé sous le solide, transporant des particules de fumée à l'arrière de ce dernier. La simulation prend place dans un domaine de taille $[0, 1]^3$ où des conditions aux bords périodiques sont imposées pour les calculs. Un solide rigide fixe de forme cubique, centré dans le domaine, permet d'assurer de manière implicite des conditions de non glissement aux bords. Le fluide a une densité de 1 et une viscosité de 10^{-5} afin d'assurer une dynamique turbulente, ce qui est un critère visuel important. L'hélice est considérée comme fixe, mis-à-part une rotation d'environ $2,5$ tours par seconde, imposée sur l'axe des z, permettant ainsi de simuler le mouvement d'un ventilateur par exemple.

La figure 5.3 montre des images tirées de la simulation dans laquelle un ensemble de particules de fumée est généré au niveau du sol (à $z = 0.1$) puis transporté passivement par le fluide. Dès les premiers instants, la méthode de pénalisation vient créer un champ de vitesse suivant le mouvement des palles, localement autour de l'hélice. Puis ce mouvement induit un phénomène d'aspiration — du fait de la forme et de l'orientation des palles — à l'avant de l'hélice dans la direction des z positifs ($t = 1$ et $t = 1.15$). Les particules de fumée sont alors attirées vers le nez et ensuite projetées vers les bords par l'effet de rotation des palles ($t = 1.8$). La figure 5.4 montre les particules de vorticité les plus véloces autour d'une des palles dans les premiers instants de la simulation. On y voit clairement la création de deux vortex de signe opposé derrière la palle (dans le sens du mouvement), ce qui correspond au

phénomène observé dans l'écoulement autour d'un cylindre.

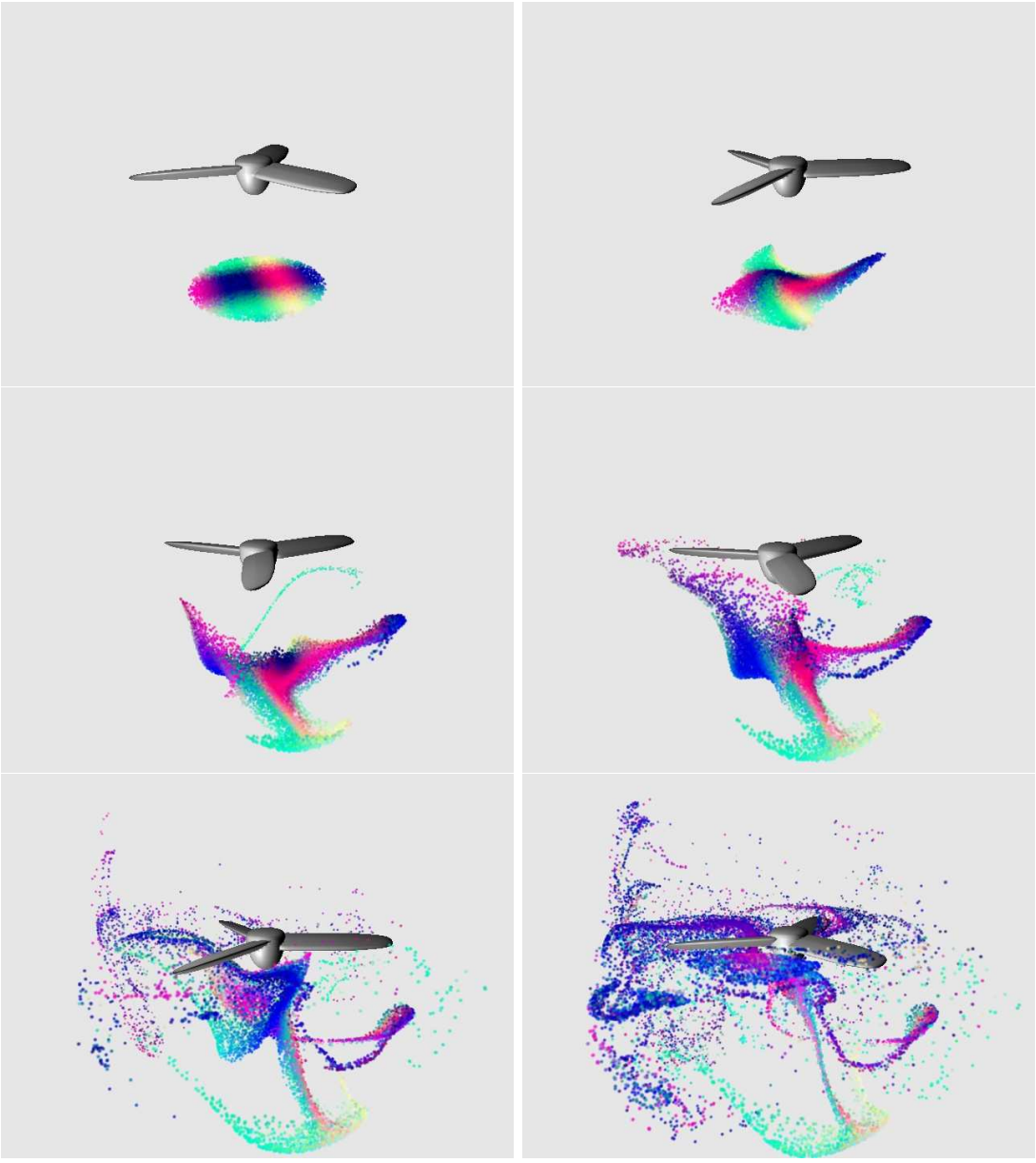


FIG. 5.3 – Images tirées de la simulation d'une hélice à rotation imposée. Captures aux instants $t = 0, 1.6, 2.6, 3.0, 3.5, 4.0$. La rotation de l'hélice créé une aspiration de l'air situé sous elle. Les particules de fumée sont ensuites projetées vers les bords du domaine.

Dans la simulation présentée en figure 5.3, le pas de temps est de $\delta t = 0.01$. Le domaine est discrétisé en 64^3 points de grille, et le temps de calcul en présence d'environ 150.0000 particules est de $2,5$ secondes par itération. La répartition du temps de calcul est donné dans le tableau 5.2. En comparaison des exemples proposés dans

le chapitre précédent, le coût algorithmique comptant pour l'ajout des deux solides, l'hélice et la boîte englobante, en interaction avec le fluide représente environ 18% du temps de calcul. Le transport des level set des solides se résume au déplacement de l'hélice (le cube restant fixe), et cette étape représente $4,5\%$ du temps de calcul.

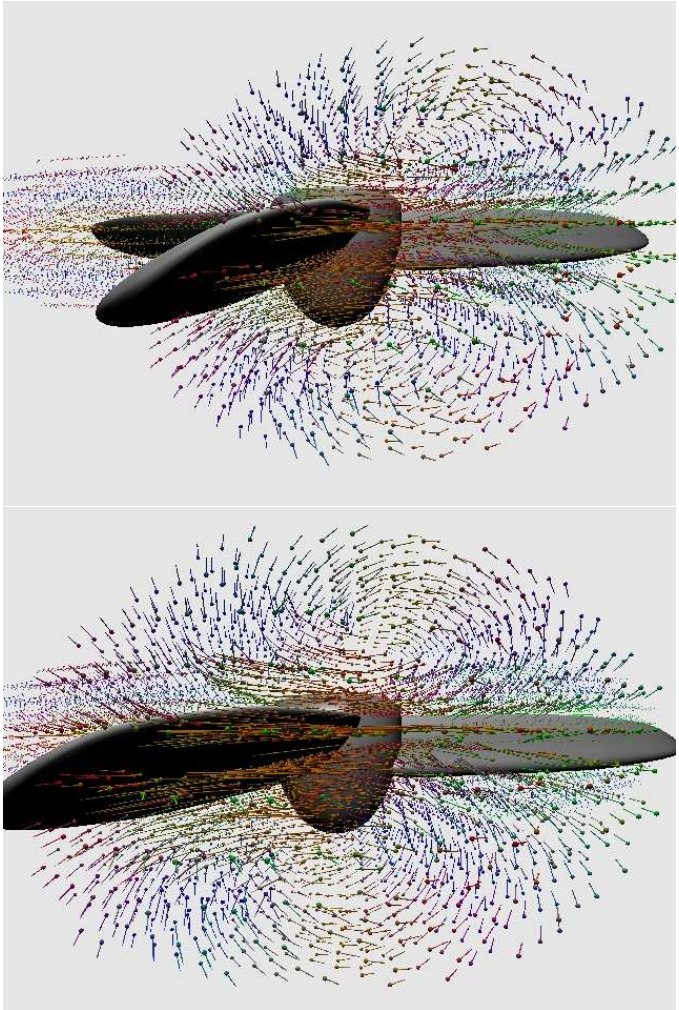


FIG. 5.4 – Visualisation de la vitesse à l'endroit des particules de vorticité, dans les premiers instants du calcul. On peut y voir la création de deux vortex de signe opposé le long de la palle nous faisant face, à l'arrière de celle-ci.

Le coût du calcul global pourrait être grandement réduit par la suppresion de particules de vorticité de faible intensité, mais, dans le souci de conserver une dynamique importante pour le mouvement de la fumée, j'ai conservé une grande majorité de celles-ci.

Vit. + diff.	Advection	Remaillage
14%	23%	22%

Pénalisation	Calcul $\bar{\mathbf{u}}_i$	Transport de S_i	Autre
13,5%	4,5%	4,5%	18,5%

TAB. 5.2 – Pourcentage du temps de calcul attribué à chaque partie de la résolution des équations, pour l'exemple de l'hélice à mouvement imposé. La taille de la grille est de 64^3, le nombre maximum de particules est 150.000 et $2,5$ secondes sont nécessaires pour chaque itération. L'ajout des deux solides rigides dans le fluide n'occupe qu'un total d'environ 18% du temps de calcul. Le terme « advection »désigne ici le transport des particules de vorticité par le schéma RK2.

Hélice à rotation libre

Dans ce deuxième exemple, je m'intéresse au problème, dual du précédent, de la rotation d'une hélice induite par un flux faisant face à celle-ci. Le centre de gravité de l'hélice est fixé, et le calcul de la vitesse de cette dernière ne prend en compte que la composante z du mouvement, comme le serait une éolienne solidement fixée au sol. Une vitesse de fluide $\mathbf{u} = (0, 0, 1)^T$ est imposée devant l'hélice par la méthode de pénalisation sur un cylindre faisant office de générateur de vent. Afin de pouvoir mieux capter le phénomène, j'ai considéré une boîte — plus grande que précédemment — de taille $[0, 4]^3$ avec des conditions périodiques, et une discrétisation par une grille de taille 64^3. Aucun solide ne vient imposer des conditions de non glissement aux bords de ce domaine, afin que le mouvement de l'hélice ne soit pas trop perturbé par le ralentissement du fluide aux bords et dans l'optique d'obtenir une simulation visuellement plus dynamique. Même si la périodicité de la vitesse vient indubitablement perturber ce mouvement, comme la boîte est grande par rapport à l'hélice, ceci n'est pas observable dans les premiers instants.

La figure 5.5 présente quelques résultats visuels obtenus sur ce cas test, où des particules de fumée sont introduites au cours du temps à l'avant de l'hélice afin de visualiser le mouvement du fluide. La figure 5.6 fournit la vitesse de rotation de l'hélice autour de son axe z en fonction du temps. Nous pouvons voir qu'elle atteint une vitesse de rotation maximum et stable à partir de $t = 0.2$. Le coût algorithmique pour cet exemple est similaire au précédent.

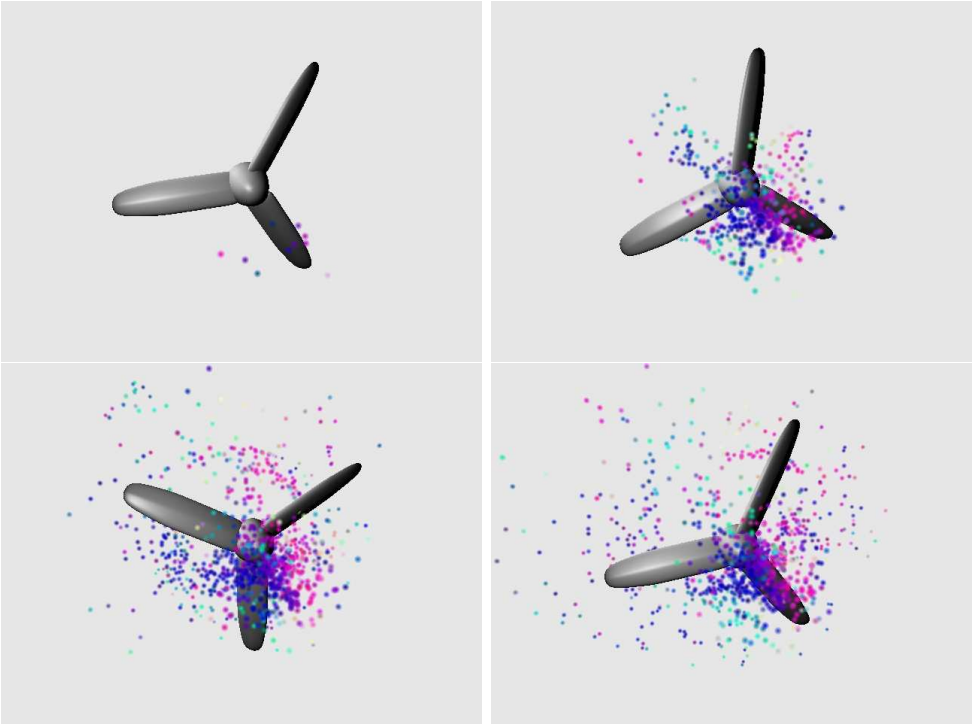


FIG. 5.5 – Images tirées de la simulation d'une hélice tournant sous l'effet du vent. Captures aux instants $t = 0, 0.1, 0.24, 0.34$. Les particules de fumée captent le mouvement du fluide.

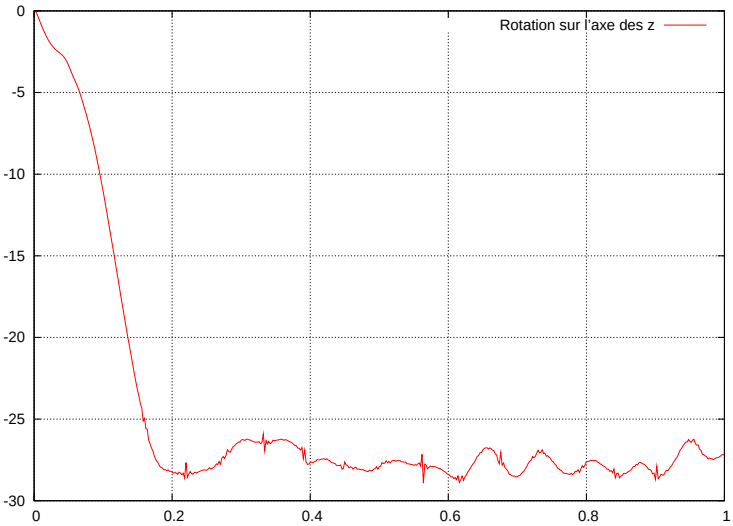


FIG. 5.6 – Vitesse de rotation de l'hélice autour de son axe principal, en fonction du temps, dans le cas du mouvement d'une hélice faisant face au vent.

Interaction d'un solide rigide avec l'air et l'eau

Ce troisième exemple présente les résultats obtenus avec la méthode de pénalisation en présence d'un fluide à deux phases. Un solide rigide tombe sous l'effet de la gravité du haut de la boîte de calcul en direction de la surface de l'eau, initialement horizontale. Le domaine est un cube de 20 centimères de côté, pour une discrétisation de 128^3 avec des conditions aux bords périodiques. La viscosité des deux phases est fixée à 10^{-5}, et la densité à 1 et $1000kgm^{-3}$ pour l'air et l'eau, respectivement. La densité du solide est de $800kgm^{-3}$ et le pas de temps $\delta t = 10^{-3}$. Un cube rigide et statique entoure la boîte de calcul afin d'imposer la condition de non glissement aux bords.

La figure 5.7 présente un rendu réaliste des résultats obtenus, grâce au logiciel gratuit Blender[1]. Au temps $t = 0.1$, la tasse a entre profondément dans l'eau sous l'effet de la vitesse acquise dans l'air. Puis, comme elle est moins dense que l'eau, remonte pour flotter à la surface. Ensuite l'eau pénètre à l'intérieur de la tasse. Les calculs ont été effectués sur un Itanium Alpha 64 1.4 GHz avec 32 Go de mémoire RAM. Environ 60 secondes de calcul sont nécessaires par itération, la répartition du temps pour chaque étape du calcul est fourni dans le tableau 5.3. J'ai mesuré que les calculs sur cette machine sont environ 2 à 3 fois supérieurs à ceux obtenus sur le Pentium 2GHz. Dans l'article *Rigid Fluids* [CMT04], l'auteur reporte un temps de calcul de 27.5 secondes par itérations — sur un Pentium 4 2Ghz similaire à la machine citée précédemment — pour la simulation de 9 solides en interaction avec une surface libre, dans une boîte de calcul de taille $64 \times 68 \times 64$, où 4% du temps de calcul est utilisé pour les interactions. Ceci représenterait environ 200 secondes par itération pour une discrétisation de 128^3, en estimant que le temps de calcul croît linéairement. En comparaison, la simulation présentée ici est donc approximativement 7 fois plus rapides.

Je propose en figure 5.8 un second exemple de simulation où un canard rigide tombe dans l'eau, avec les mêmes paramètres que précédemment. Au temps $t = 0.3$, le canard heurte le bord et se couche entre les temps temps $t = 0.5$ et $t = 0.7$. La collision est traitée par la méthode que je détaille dans la partie suivante. Dans cet exemple, les temps de calculs sont similaires à ceux donnés dans le tableau 5.3.

Vitesse + diffusion	Advection	Remaillage	Pénalisation
5.5%	21.5%	14%	19%

Terme multiphase	Transport de la surface	Reinitialisation	Autre
4%	7.5%	15%	13.5%

TAB. 5.3 – Pourcentage du temps de calcul attribué à chaque partie de la résolution des équations, pour l'exemple de la tasse tombant dans sur une surface libre. La taille de la grille est de 128^3, le nombre maximum de particules est 10^6 et 60 secondes sont nécessaires pour chaque itération.

[1] http ://www.blender.org

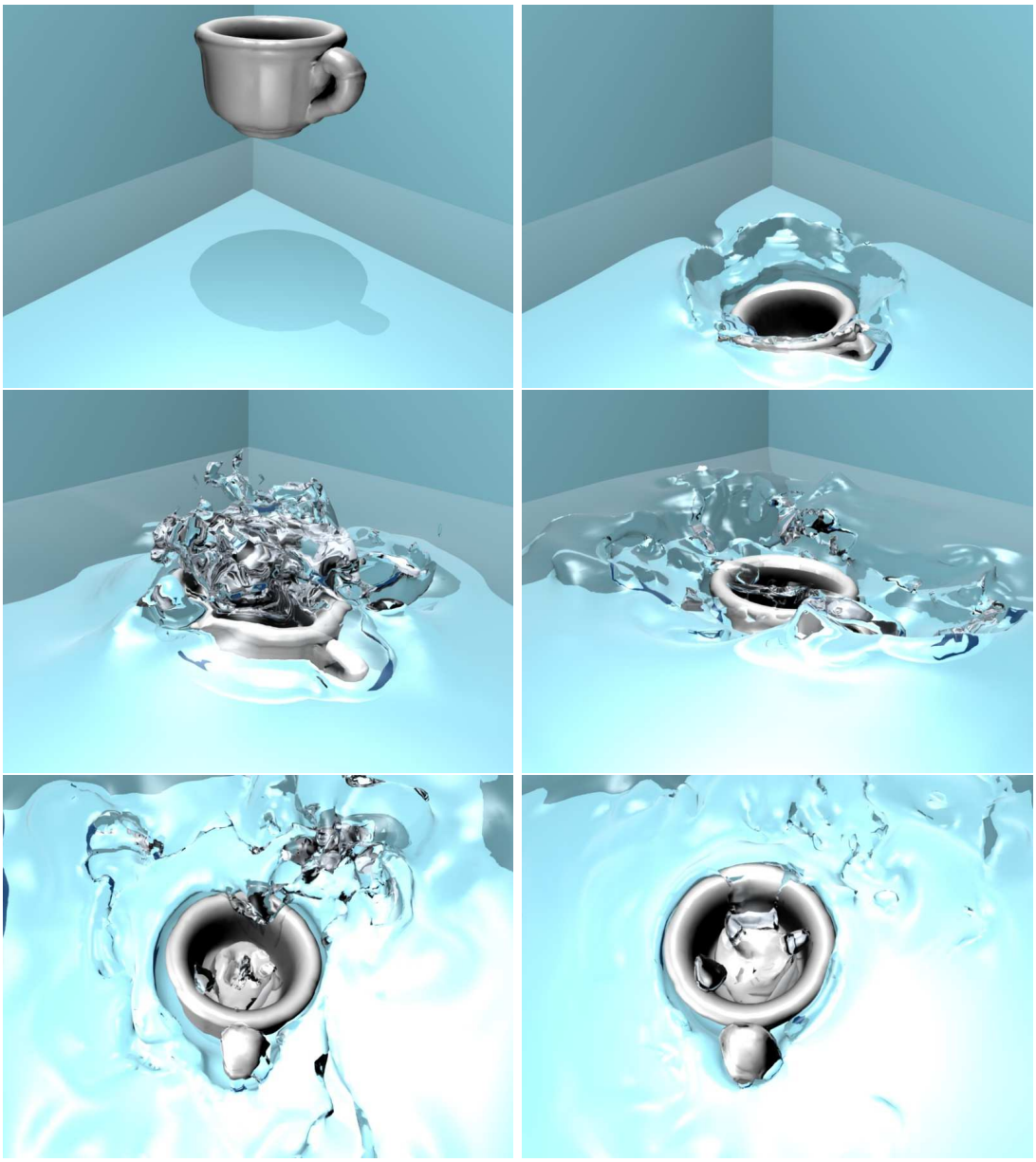


FIG. 5.7 – Images tirées de la simulation d'une tasse tombant sous l'effet de la gravité dans un bac d'eau. Captures aux instants $t = 0, 0.1, 0.15, 0.2, 0.25, 0.3$.

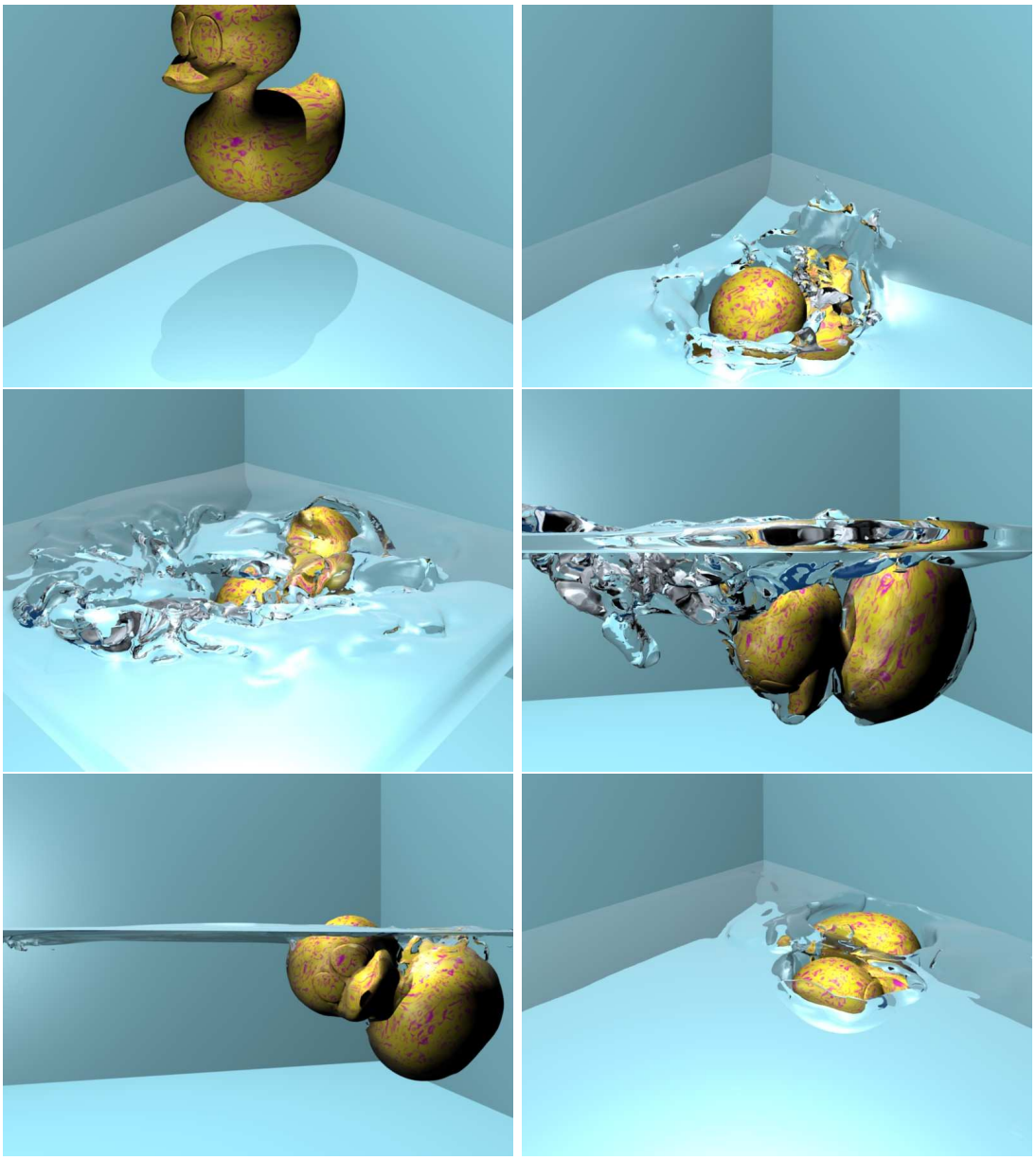


FIG. 5.8 – Images tirées de la simulation d'un modèle de canard rigide tombant sous l'effet de la gravité dans un bac d'eau. Captures aux instants $t = 0, 0.1$ pour la première ligne, $t = 0.2$ pour la deuxième (avec deux prises de vue différentes) et $t = 0.5, 0.7$ pour la dernière ligne.

3 Modèle de collision

Je présente dans cette partie le modèle de collision entre solides développé lors de mes travaux. Dans le cas où plusieurs solides sont en interaction dans un fluide, un modèle de collision entre ces solides est nécessaire. En effet, des expérimentations

montrent qu'il y a effectivement collision quand deux solides plongés dans un fluide entrent en contact, provocant un rebond pour les deux objets. Mais, comme il est noté dans [GPH+01] et étudié dans de récents travaux théoriques, les équations de Navier-Stokes couplées avec des conditions de non glissement aux bords n'est pas compatible avec le contact des solides. Ceci peut se comprendre intuitivement par le fait que, dans le cas d'une discrétisation de l'espace infiniment fine, il existera toujours des particules de fluide présentent entre les objets, les empêchant ainsi de se toucher physiquement. Numériquement, un modèle de collision est important afin d'éviter toute inter-pénétration qui pourrait être entrainée par les erreurs de discrétisation. En synthèse d'images, il est évident que la collision entre plusieurs solides de forme quelconque est nécessaire au réalisme d'une animation. Le modèle que je propose permet aisément de traîter ce problème.

Tout d'abord j'expliquerai les fondements d'un nouveau modèle de collision basé sur une représentation des solides par une fonction level set. Puis je proposerai une manière de l'intégrer dans les équations de Navier-Stokes en formulation vorticité. Finalement je présenterai des résultats numériques comparant notre méthode à celle proposée dans [GPH+01] ainsi que des exemples pour la synthèse d'images.

3.1 Définition du modèle

Le modèle de collision proposé permet de conserver l'énergie, ce qui est une propriété physique primordiale en mécanique du solide. Dans le cadre de solides en interaction avec un fluide visqueux, l'énergie est en partie diffusée dans le fluide au moment du contact, au niveau de la couche limite. Ce nouveau modèle est basé sur la représentation Eulerienne de l'interface, fournie par la fonction level set associée à chaque solide. Le fait que celle-ci soit une fonction distance (comme nous l'avons vu précédemment, cette propriété est conservée par le transport) nous permet de connaître exactement, en chaque point de discrétisation, la distance séparant les deux solides, et c'est sur cette base qu'est dérivée la force de collision. Dans le cadre de surfaces implicites [CD97], les auteurs proposent une méthodologie similaire dans laquelle la force de collision entre deux objets est évaluée en fonction du recouvrement de l'un par l'autre. C'est dans cette zone de chevauchement qu'est alors calculée la force de répulsion — grâce à la fonction distance, la profondeur de pénétration est connue. Une nouvelle surface de contact supprimant ce chevauchement en est alors déduite et propagée à l'intérieur des objets incriminés.

Notre approche est différente : l'intégration de la force de collision dans les équations se fait de manière simple et ne nécessite pas un traitement particulier lors du chevauchement entre deux objets. De plus, notre modèle ne nécessite pas de paramètre et s'intègre aisément dans un code de simulation de fluide. La représentation level set adoptée, en plus de servir à calculer la force, permet de la distribuer sur la surface du solide plongé dans le fluide.

Modèle 1D

Dans cette section, je présente le modèle de collision dans le cas particulier 1D, le cas général basé sur une formulation level set est décrit dans la prochaine section. Soit un solide représenté par un point, à la position $x(t) > 0$. L'autre surface considérée est un mur localisé en $x = 0$. Afin de modéliser la collision entre ces deux éléments, nous choisissons un système dynamique qui est localisé sur une courte distance et singulier à l'endroit du contact (pour empêcher toute pénétration) :

$$\ddot{x} = \frac{\kappa}{x} e^{-x/\varepsilon} \tag{5.25}$$

où κ est un coefficient de la même dimension que le carré de la vitesse $\dot{x}$. Ce système est un modèle Hamiltonien agissant dans une couche de l'ordre de ε à partir de $x = 0$.

Modèle level set

Le modèle level set proposé sert à la discrétisation du système dynamique présenté ci-dessus, ainsi qu'à appliquer la force de collision sur la surface des objets. Grâce à cette représentation, il est possible de calculer la distance, la normale à la surface permettant d'orienter la force ainsi que l'intégrale discrète sur la surface. Soient deux solides S_i et S_j dont les grandeurs associées sont notées respectivement par l'indice i et j. Soient ϕ_i et ϕ_j les level set représentant la distance signée à chacune des surface. La généralisation du modèle de collision 5.25 dans une formulation level set est traduite comme une force dans le fluide :

$$\mathbf{f}_{col}(\mathbf{x}) = \rho \sum_{ij} \zeta\left(\frac{\phi_i(\mathbf{x})}{\varepsilon}\right) \frac{\kappa_{ij}}{\varepsilon} \frac{\nabla\phi_j(\mathbf{x})}{\phi_j(\mathbf{x})} e^{\frac{-\phi_j(\mathbf{x})}{\varepsilon}}. \tag{5.26}$$

Dans cette équation, ζ est une fonction *cut off* dont la régularisation localise la force dans une épaisseur de l'ordre de ε autour de l'interface, κ_{ij} peut être interprété comme l'intensité de cette force, qui doit être de l'ordre du carré de la vitesse relative des deux objets au moment du contact (c'est-à-dire quand la distance est de l'ordre de ε). $\nabla\phi_j$ est la normale à la surface de S_j orientée vers l'extérieur. Cette force est donc active à une distance de ε et présente dans le fluide au niveau de la surface régularisée sur une épaisseur du même ordre.

L'équation de Navier-Stokes en formulation vortex incorporant la pénalisation et la force de contact en présence de plusieurs solides est donc :

$$\frac{\partial\omega}{\partial t} + (\mathbf{u}\cdot\nabla)\omega = (\omega\cdot\nabla)\mathbf{u} + \nu\Delta\omega + \nabla\left(\frac{1}{\rho}\right)\times\nabla p + \lambda\nabla\times\left(\sum_i \chi_{S_i}(\bar{\mathbf{u}}_i - \mathbf{u})\right) + \frac{\nabla\times\mathbf{f}_{col}}{\rho}. \tag{5.27}$$

3.2 Remarques sur la discrétisation

Le modèle de collision proposé inclu une force singulière au niveau du contact physique entre les objets. En pratique, ce contact doit à tout prix être évité, ce

qui n'est pas assuré lors de la discrétisation temporelle des équations de Navier-Stokes où, si le pas de temps est choisi trop grand, les solides peuvent se chevaucher, ou tout du moins être à une distance $d \ll \varepsilon$ l'un de l'autre. C'est pourquoi il est important d'utiliser un pas de temps δt et une intensité de rebond κ_{ij} (s'il n'est pas calculé automatiquement) adéquats afin d'éviter l'introduction de forces démesurément grandes dans le fluide. Cette contrainte sur δt intervient uniquement lorsque les solides sont assez proches l'un de l'autre, à une distance de l'ordre de ε.

Un autre point important est que la différenciation numérique de cette force de collision singulière, dans l'équation de vorticité, peut entrainer des erreurs importantes dans la résolution des équations du fluide. Afin de palier ce problème, dans le cas de solides rigides j'ai choisi de moyenner cette force (dans sa formulation vitesse) à l'intérieur du solide et de l'inclure directement dans la vitesse $\bar{\mathbf{u}}$ de ce dernier lors de l'étape de pénalisation. Cette technique permet de régulariser la force de collision et de l'appliquer directement à la totalité du solide, comme il est souvent considéré en mécanique du solide. En contrepartie cette méthode a pour conséquence de ne pas considérer les forces situées au delà du solide — à l'intérieur du fluide — comme la fonction cut off est régularisée en pratique. Ainsi, les simulations calculées sont rendues plus stables, permettant d'utiliser de plus grands pas de temps.

En pratique, j'ai opté pour $\varepsilon = h$ où h est le pas d'espace afin de limiter l'influence de la force de collision à une distance de la taille d'une cellule de la grille. Ce choix permet de laisser les solides se déplacer librement, uniquement soumis aux forces hydrodynamiques, autant que faire se peut et d'ainsi conserver au maximum l'action du fluide situé entre les deux objets. De plus, la force de collision peut être considérée négligeable lorsque les solides sont à une distance supérieure à kh (avec $k > 1$), nous permettant de réduire le coût de calcul lorsque les solides sont assez éloignés.

3.3 Applications numériques

Validation 2D sans fluide

Cet exemple démontre la très bonne conservation de l'énergie dans le cas d'un disque entrant en collision avec un mur, en l'absence de fluide, dans le cas 2D. Un disque de rayon 0.1 initialement situé en $y = 0.3$ tombe sous l'effet de la gravité $g = -1$ (dans la direction y) sur un mur situé en $y = 0$. Les fonctions level set associées à chaque solide rigide sont triviales à définir et sont discrétisées dans une grille de taille $h = 1/128$. La zone de collision est définie par le paramètre $\varepsilon = h$, le pas de temps par $\delta t = h/10$ et κ_{ij} fixé à 0.36. La figure 5.9 montre que le disque rebondit sans dissipation. Il est important de noter que le premier rebond (à $t \simeq 1$) est légèrement plus haut que la position de départ. Ceci est dû au schéma d'Euler explicite utilisé pour la discrétisation, qui ne permet pas de conserver l'énergie exactement. Mais cette légère erreur ne pose pas de problème important en pratique, c'est pourquoi je n'ai pas considéré un schéma implicite pour l'intégration, schéma qui serait plus difficile à mettre en oeuvre. En conclusion, la méthode de collision présentée plus haut sous une formulation level set permet de traîter les rebonds en conservant

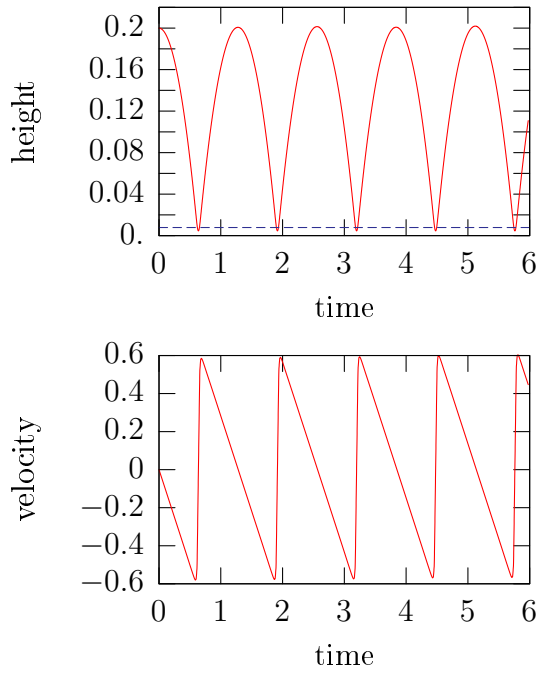


FIG. 5.9 – Rebond d'un disque sur un plan situé en $y = 0$ en l'absence de fluide. La hauteur y du disque et sa vitesse sont données en fonction du temps. La ligne en pointillés dans le premier graphique représente la hauteur $y = h$ où h est le pas d'espace. Le pas de temps utilisé est $\delta t = h/10$ et le paramètre de collision $\varepsilon = h = 1/128$.

correctement l'énergie du système.

Validation 2D avec fluide

Intéressons nous maintenant au couplage fluide-solides rigides relevé de la prise en compte des forces de collision. Ce cas test est similaire au précédent, auquel a été ajouté la présence d'un fluide. Cet exemple a été étudié dans [GPH+01] et, afin de nous comparer avec leurs résultats, les mêmes paramètres ont été adoptés. Dans [GPH+01], la force de collision est calculée à partir de la connaissance explicite de la distance entre le centre de gravité du disque et la position du mur. La boîte de calcul est de taille $[0, 2] \times [0, 6]$, la densité du fluide est de 1 et sa viscosité de 0.01. Le disque initialement situé en $(1, 4)$ a un rayon de 0.125 et une densité de 1.5 ; la gravité est fixée à -980 sur l'axe des y. Comme nous l'avons vu sur l'exemple de sédimentation dans la partie précédente, le disque atteint une vitesse terminale et, dans le cas qui nous intéresse ici, rebondit sur le fond de la boîte. La figure 5.10 montre une comparaison des résultats obtenus dans [GPH+01] avec les notres. Les courbes de position et de vitesse correspondent très bien, avec une légère variation de la vitesse après le contact. Cette dissemblance est due en grande partie à la

FIG. 5.10 – Sédimentation d'un disque plongé dans un fluide, soumis à la gravité, et rebond sur le mur situé en $y = 0$. Les résultats obtenus par la méthode présentée ici (images de droite) sont très proches de ceux présentés dans [GPH+01] (images de gauche). Les images du haut représentent la hauteur du disque, celles du bas sa vitesse. Les courbes solides sont les résultats calculés pour $h = 1/256$, $\delta t = 7.5\,10^{-4}$, et les courbes en pointillés pour $h = 1/384$, $\delta t = 5\,10^{-4}$.

différence de modèle de collision où le notre considère de façon implicite la distance entre les deux solides ainsi que la force de collision. Nous verrons dans l'exemple suivant que la technique proposée dans [GPH+01] intégrée dans notre modèle de fluide donne des résultats similaires avec toutefois une réponse à la collision un peu plus raide.

Cet exemple nous permet de conclure que cette nouvelle méthode de collision, intégrée dans les équations de couplage fluide-structure par pénalisation, permet de calculer précisément la force de collision au contact, de manière qualitativement proche des méthodes plus traditionnelles. Un point important est que ce nouveau modèle ne se limite pas à des solides rigides de forme sphérique. Il permet, comme nous le verrons par la suite, de faire interagir de manière simple deux objets de forme quelconque, ce qui est d'un intérêt majeur, particulièrement en synthèse d'images.

Collisions de deux sphères en sédimentation

Le problème de la collision de deux sphères soumises aux forces hydrodynamiques et à la gravité a été étudié expérimentalement [FJL87] et numériquement dans [GPH+01,SP05]. Initialement, deux sphères de même rayon sont alignées selon l'axe z défini par la gravité, à une distance d'environ un diamètre. En tombant, la sphère sur le dessus se trouve dans le sillage de celle en dessous et subit donc moins de frictions. Elle descend alors plus vite que la première et finit par la toucher. L'alignement vertical des deux sphères est instable et cette instabilité les amène par la suite à se séparer. Ce phénomène est connu en Anglais sous le nom de « *drafting, kissing and tumbling* », et il a été étudié expérimentalement dans [FJL87].

Dans l'étude de ce cas, la boîte est de taille $[0,1] \times [0,1] \times [0,4]$ et la gravité est fixée à $\mathbf{g} = (0, 0, -980)$. La densité du fluide est de 1 et sa viscosité de 0.01, les sphères ont un rayon $r = 0.083$ et une densité de 1.14. Elles sont initialement positionnées en $(0.5, 0.5, 3.5)$ et $(0.5, 0.5, 3.16)$, et, afin d'enclencher l'instabilité, j'ai décalé d'un pas de grille dans la direction (x, y) la sphère du dessous. Le pas d'espace choisi pour cette simulation est le même que pour [GPH+01] : $h = 1/60$ et le pas de temps est fixé initialement à $\delta t = 0.007$. Il sera légèrement modifié par la suite pour répondre à la contrainte $\delta t < k|\omega|_\infty^{-1}$, j'ai pris $k = 0.8$ et le pas de temps est réduit jusqu'à atteindre 0.054 au moment de la collision. Le paramètre de collision ε est choisi égal à h et le coefficient pour l'amplitude de la réponse à la collision est fixé à $\kappa_{12} = \kappa_{21} = 25$.

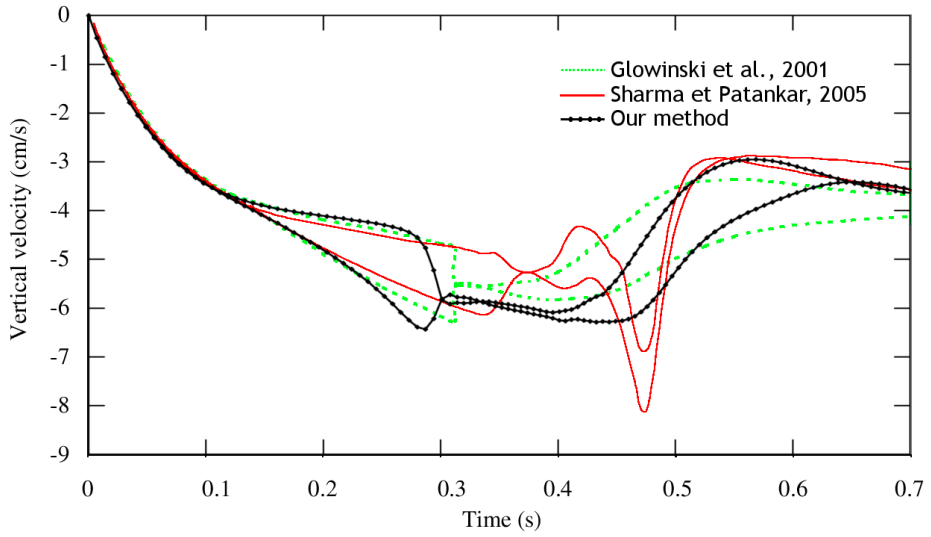


FIG. 5.11 – *Drafting, kissing and tumbling* de deux sphères tombant dans un fluide. Comparaison des résultats obtenus avec les méthodes de [GPH+01, SP05] et celle proposée ici. L'instabilité, amplifiée après la collision, provoquant la séparation des solides peut entrainer divers comportements suivant la méthode de calcul utilisée : les résultats obtenus après le temps $t = 0.3$ ne sont pas qualitativement comparables.

Les vitesses verticales des sphères calculées par notre méthode sont comparées à celles de [GPH+01] et [SP05] dans la figure 5.11. Nos résultats concordent bien avec ceux obtenus par Glowinski *et al.* [GPH+01], mis-à-part quelques instants avant la collision où la sphère du haut accélère plus. Cette différence peut être due au fait que dans [GPH+01] l'instabilité déviant le vecteur liant les centres de gravité de l'axe vertical, est enclenchée plus vite dans leur simulation ; dans nos résultats, la sphère du dessus reste donc plus longtemps dans le sillage de la seconde et accélère d'autant plus, rapprochant l'instant de la collision. Il est important de noter que la différence est plus grande avec les résultats obtenus dans [SP05], dont la modélisation est pourtant plus proche de notre approche. Comme noté par Sharma et Patankar [SP05], la manière dont sont résolues numériquement les équations influe grandement sur la réponse des deux sphères à l'instabilité. Les résultats fournis par ces auteurs montrent que les deux sphères se séparent dans deux direction opposées alors que pour [GPH+01] et les tests effectués avec notre méthode, les deux sphères sombrent du même côté comme le montre la figure 5.12. La figure 5.13 nous renseigne sur la distance $|c_1 - c_2| - 2r$ (où c_i représente le centre de gravité du solide i) séparant les deux sphères. Nous pouvons voir que la distance minimum entre les deux sphères avant séparation est de l'ordre de h.

En figure 5.14, je présente la vitesse verticale des sphères en fonction du temps, obtenues dans notre formulation vortex avec deux calculs de la collision différents : avec notre modèle level set et avec le modèle de [GPH+01]. Pour ce dernier, d'après

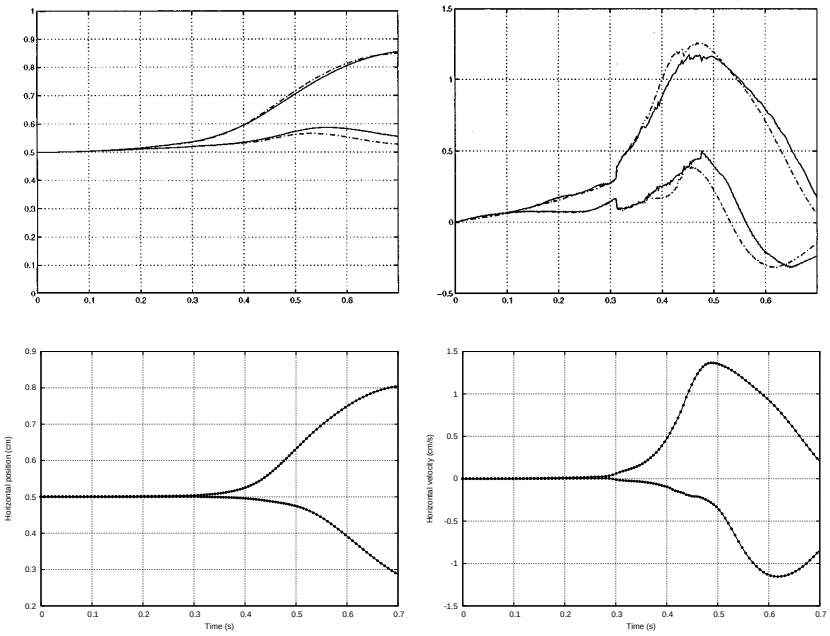


FIG. 5.12 – *Drafting, kissing and tumbling* de deux sphères tombant dans un fluide : composantes horizontales de la position (à gauche) et de la vitesse (à droite) du centre de gravité des deux sphères. En haut : les résultats extraits de [GPH+01], en bas : ceux obtenus avec la méthode proposée ici.

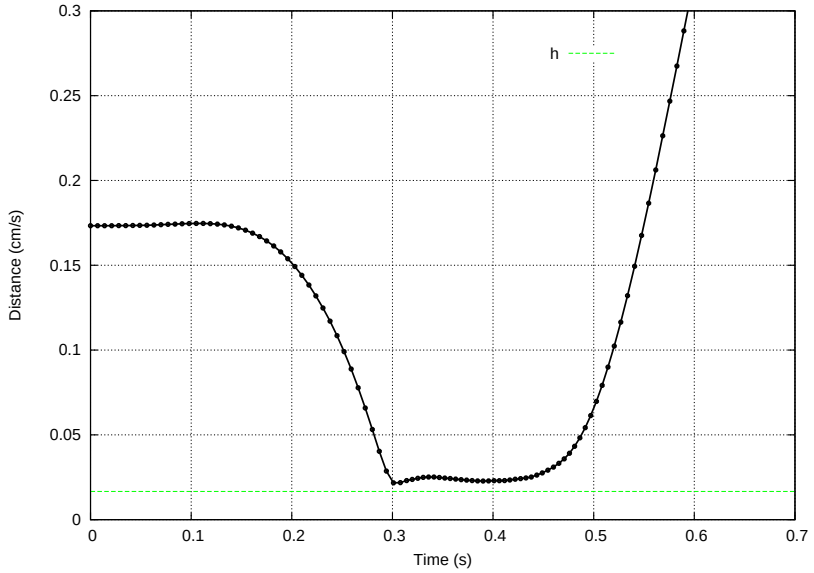


FIG. 5.13 – *Drafting, kissing and tumbling* de deux sphères tombant dans un fluide : distance entre les deux sphères, calculée au niveau des l'interfaces.

leur notation, j'ai utilisé les paramètres $\rho = 2/60$, $\varepsilon = 10^{-4}$ et $c_{ij} = 3.33 \times 10^{-6}$. La réponse obtenue au moment du contact entre les deux sphères est similaire, bien qu'elle soit un peu plus raide pour le second modèle. Ce résultat semble physiquement plus correct — bien qu'il soit difficile d'en juger à cette échelle — car il utilise la distance exacte, calculée explicitement, entre les deux sphères avec un modèle adapté aux sphères. Le modèle que j'ai proposé converge avec $h \to 0$ vers une formulation similaire, mais il permet quant à lui de considérer tout type de géométrie pour les solides en considérant les forces apparaissant à l'interface.

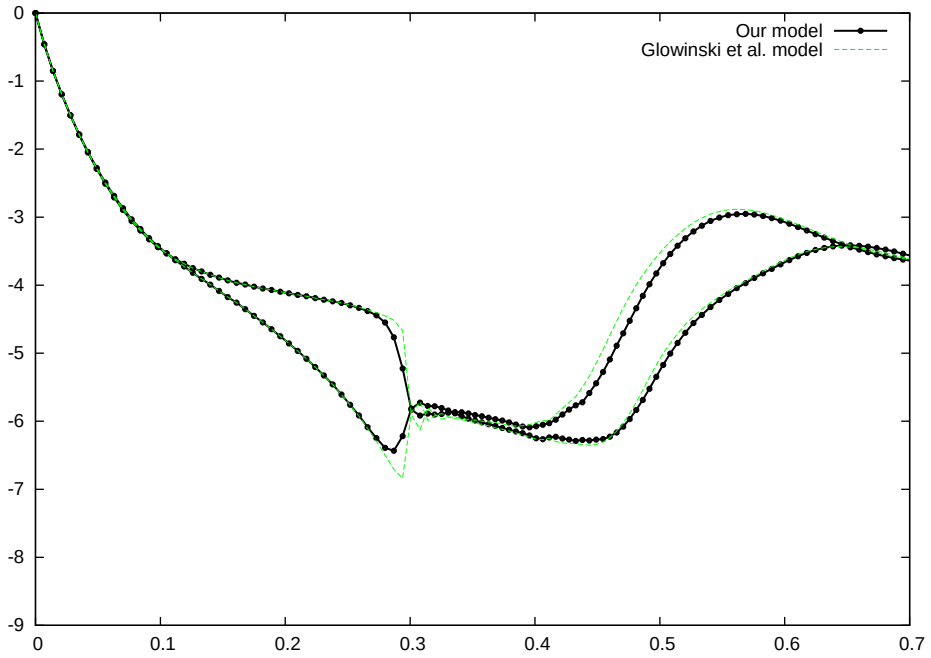


FIG. 5.14 – *Drafting, kissing and tumbling* de deux sphères tombant dans un fluide : comparaison des vitesses des sphères calculées par méthode vortex, en utilisant le modèle de collision de [GPH+01] et notre approche level set.

Intéressons nous maintenant au coût des calculs nécessaires pour l'obtention de ces résultats. Pour le même pas d'espace, la résolution du problème dans [GPH+01] a nécessité 120 secondes par itération sur un DEC 500MHz ; aucun temps de calcul n'est fourni dans l'article [SP05]. Utilisant notre approche sur la même machine précisée en partie 1.6, le coût du calcul varie de $9s$ à $18s$ en fonction du nombre de particules : 5000 au début de la simulation et 400.000 à la fin. En considérant les performances relatives des deux machines, les deux méthodes sont comparables ; néanmoins, notre méthode vortex est plus robuste et notre simulation a été obtenue avec des pas de temps de 5 à 7 fois plus grands que ceux utilisés dans [GPH+01]. Ce gain d'efficacité est un point important, autant en mathématiques appliqués où des calculs intensifs peuvent être nécessaires, qu'en synthèse d'images où des résultats rapides sont escomptés. Dans l'article *Rigid Fluids* [CMT04] proposant une

adaptation de la méthode de domaine fictif de Sharma et Patankar, le coût du calcul d'une itération pour une simulation comprenant plusieurs solides et une interface pour un fluide à plusieurs phases, est de $27,5$ secondes sur un domaine de taille $64 \times 68 \times 64$ (soit environ trois fois moins d'éléments que pour notre simulation). En comparaison, notre méthode — à laquelle il faudrait ajouter le transport de l'interface du liquide pour une comparaison plus précise, qui, comme nous l'avons vu au chapitre précédent, peut occuper jusqu'à 60% du temps dans mon implémentation — apparaît donc plus rapide.

Collision entre objets de forme quelconque

Afin de démontrer l'efficacité de notre méthode pour la prise en compte de collisions entre des solides rigides à géométrie quelconque, j'ai considéré la collision d'une ellipsoide contre une sphère statique. Dans le but de séparer le problème d'interaction de celui de la collision, j'ai supprimé la présence du fluide dans cette simulation. Les résultats obtenus — dont la visualisation 3D du mouvement est donné en figure 5.15 — montrent que notre technique capture correctement les forces de collisions entrainant une rotation des objets en contact. Pour mieux se rendre compte du phénomène, j'ai fixé la position de la sphère et fait tomber l'ellipsoide sous l'effet de la gravité. Les paramètres sont les suivants : $\varepsilon = h$, $h = 1/64$, $\delta t = 0.005$ et $\kappa_{ij} = 5$. L'ellipsoide est initialement décalée par rapport à la sphère afin d'enclencher la rotation comme nous pouvons le voir en figure 5.15. Cette simulation montre clairement la capacité de la nouvelle méthode de collision à capturer une force de collision induisant de la rotation.

4 Solides à membranes élastiques

4.1 Problématique

La représentation des solides rigides adoptée précédemment, par une fonction level set sur une grille Eulerienne, est similaire à celle utilisée dans [CM06a] pour la simulation de membranes élastiques. Au début de mes recherches pour ma thèse, j'ai adopté cette méthode comme piste pour la simulation d'objets rigides. En effet, en l'absence de mouvement du fluide contenu à l'intérieur d'une membrane élastique, si la raideur de cette dernière est considérée comme infinimment grande, la dynamique du milieu considéré peut alors être perçu comme celle d'un solide rigide, comme il a été simulé dans [GHS93] sur une base de méthode de *frontière immergée*. Un des problèmes majeurs de cette technique est la nécessité d'utiliser des pas de temps petits dans le cas de raideurs importantes, car les forces en jeu en présence de fortes déformations deviennent alors très grandes, ce qui déstabilise les calculs pour des pas de temps importants.

Le temps de calcul est un critère important et, donc, cette approche est trop prohibitive pour les applications visées dans mes travaux. Ainsi, la méthode de pénalisation que j'ai développé pour le calcul des interactions entre fluides et solides

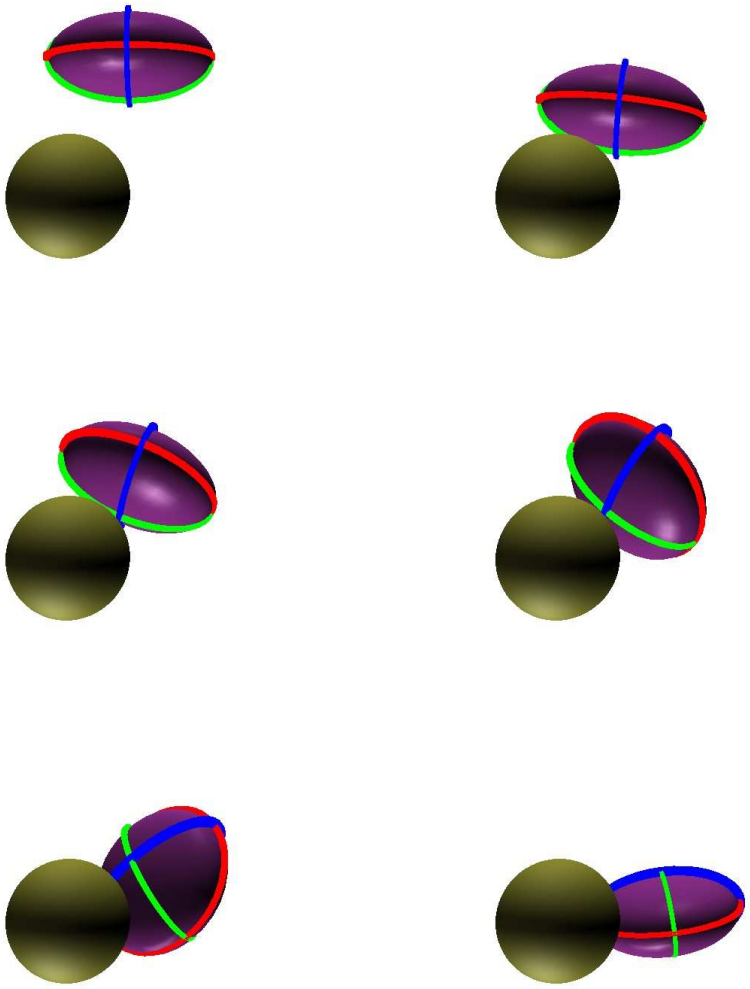


FIG. 5.15 – Images de la collision d'un ellipsoïde et d'une sphère fixe, en l'absence de fluide. Notre méthode de collision basée sur une formulation level set permet de capturer la rotation induite après le choc. Images prises aux temps $t = 0, 0.15, 0.175, 0.2, 0.25, 0.3$, de gauche à droite et haut en bas. Les axes x, y et z de l'ellipsoïde sont dessinés en rouge, vert et bleu, respectivement.

rigides en méthode vortex, est apparue comme une méthode efficace pour résoudre ce problème. À la différence des méthodes de membranes élastiques, la méthode qui a été proposée au début de ce chapitre permet de calculer de manière précise le mouvement d'un solide rigide, tout en permettant l'utilisation de pas de temps plus élevés, pour un coût de calcul faible. La déformation du domaine du solide étant nulle lorsque le paramètre de pénalisation tend vers l'infini, la level set associée peut donc être transportée de manière très simple en réduisant les erreurs numériques à celles d'interpolations de la fonction initiale.

En synthèse d'images, la simulation d'objets autres que rigides est d'une importance capitale. En effet, de nombreux objets présents dans notre environnement possèdent un comportement déformable, voire même plastique — lorsque la déformation est importante et que l'objet ne recouvre pas sa forme initiale. La simulation conjointe de solides déformables en interaction avec un fluide est un problème compliqué. L'objet déformable ayant son comportement propre — la simulation d'objets déformables est, lui aussi, un vaste champ de recherche — l'interaction avec le fluide doit alors être calculé à l'interface entre les deux milieux. Il pourrait être envisagé de simuler le solide mou en même temps que le fluide, dans une approche similaire à celle proposée pour les solides rigides. Néanmoins, la variété et la complexité des comportements des solides déformables rendraient très difficile la formulation de telles interactions. Il est aussi important de noter que, dans le cadre de fluides incompressibles, la déformation de l'objet en interaction doit toujours satisfaire cette contrainte, ce qui limite la palette de comportements des objets déformables. Dans le cas où le volume de l'objet ne reste pas constant, l'incompressibilité du fluide n'est plus assurée ; ce problème sort du cadre de mes travaux actuels.

4.2 Approche adoptée

Pour la simulation d'objets déformables, j'ai donc choisi d'utiliser une méthode basée sur celle des membranes élastiques décrites dans [CM06a]. La force élastique est calculée à partir de la level set ϕ utilisée pour représenter l'interface, en fonction de l'étirement implicitement représenté par $|\nabla\phi|$:

$$\mathbf{f} = \left\{\mathbb{P}_{\nabla\phi^\perp}\left(\nabla E'(|\nabla\phi|)\right) - E'(|\nabla\phi|)\kappa(\phi)\mathbf{n}(\phi)\right\}|\nabla\phi|\frac{1}{\varepsilon}\zeta\left(\frac{\phi}{\varepsilon}\right), \tag{5.28}$$

où je rappelle que ε est le paramètre de régularisation, et $\mathbb{P}_{\nabla\phi^\perp}(\mathbf{v}) = \mathbf{v} - (\mathbf{v}\cdot\mathbf{n})\,\mathbf{n}$ est la projection du vecteur $\mathbf{v}$ sur le plan tangent, orthogonal à la normale $\mathbf{n}(\phi) = \mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}$. Ici E' est la dérivée de la l'énergie provenant de l'élasticité du matériaux. En prennant une énergie quadratique en fonction de l'étirement de la level set par rapport à l'étirement initial supposé unitaire ($|\nabla\phi_0| = 1$) :

$$E(|\nabla\phi|) = \frac{1}{2}(|\nabla\phi| - 1)^2$$

comme énergie, nous obtenons

$$E'(|\nabla\phi|) = |\nabla\phi| - 1. \tag{5.29}$$

En injectant 5.29 dans 5.28, et en développant, nous trouvons :

$$\mathbf{f} = \{\nabla(|\nabla\phi|) - (\nabla(|\nabla\phi|)\cdot\mathbf{n})\,\mathbf{n} - (|\nabla\phi|-1)\kappa(\phi)\mathbf{n}\}\,|\nabla\phi|\frac{1}{\varepsilon}\zeta\left(\frac{\phi}{\varepsilon}\right).$$

En posant $\nabla(|\nabla\phi|) = \mathbf{Hn}$ — avec $\mathbf{H}$ ma Hessienne de ϕ : $\mathbf{H}_{ij} = \partial^2\phi/\partial x_i\partial x_j$ — et en développant l'expression de la courbure en formulation level set :

$$\begin{aligned}
\kappa &= \nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\\
&= \frac{1}{|\nabla\phi|}\Delta\phi + \left(\nabla\left(\frac{1}{|\nabla\phi|}\right)\cdot\nabla\right)\phi\\
&= \frac{1}{|\nabla\phi|}\Delta\phi - \left(\left(\frac{1}{|\nabla\phi|^2}\mathbf{Hn}\right)\cdot\nabla\right)\phi\\
&= \frac{1}{|\nabla\phi|}\left(\Delta\phi - (\mathbf{Hn})\cdot\mathbf{n}\right),
\end{aligned}$$

après simplifications, nous obtenons une formule plus simple pour la force élastique :

$$\mathbf{f} = (\mathbf{Hn} - \Delta\phi\mathbf{n} + \kappa\mathbf{n})\,|\nabla\phi|\frac{1}{\varepsilon}\zeta\left(\frac{\phi}{\varepsilon}\right). \tag{5.30}$$

Comme décrit dans [CM06a], j'utilise le principe de *renormalisation* qui permet de considérer localement la fonction level set déformée comme une fonction distance, ainsi la force élastique ajoutée à l'accélération du fluide est :

$$\mathbf{f} = \sigma\,(\mathbf{Hn} - \Delta\phi\mathbf{n} + \kappa\mathbf{n})\,\frac{1}{\varepsilon}\zeta\left(\frac{\phi}{|\nabla\phi|\varepsilon}\right) \tag{5.31}$$

où σ est le coefficient comptant pour la raideur de la membrane. En pratique, il est important d'utiliser à la place de 5.30 (en prennant $E'(|\nabla\phi|) = |\nabla\phi|-1$), l'équation 5.31 développée car celle-ci évite de calculer numériquement $\nabla E'$ et $\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right)$, ce qui réduit les erreurs numériques pouvant être importantes pour des fortes variations de $|\nabla\phi|$.

Cependant, la méthode n'est pas utilisable telle quelle si on souhaite faire interagir des fluides (avec plusieurs phases par exemple), avec des solides rigides et des membranes élastiques. Effectivement, nous souhaitons simuler des objets déformables qui conservent tout de même un aspect proche de leur forme initiale et les larges déformations sont trop contraignantes en terme de discrétisation temporelle pour être intégrées dans une simulation de fluide. Pour ce faire, nous devons donc considérer des objets élastiques comportant une raideur assez importante, ce qui implique une contrainte forte sur le pas de temps, augmentant ainsi grandement le temps de calcul.

En considérant le problème d'un autre point de vue, si l'objet est très peu déformé — *ie.* $|\nabla\phi|-1$ est proche de 0 — les forces élastiques resteront faibles et ainsi

la simulation sera stable, même pour des pas de temps de grandeur « acceptable ». Le taux de déformation $|\nabla\phi| - 1$ varie en fonction de l'écoulement autour de la membrane. Si ce dernier est proche de celui d'un mouvement rigide, alors la déformation sera faible et ainsi les forces élastiques assez faibles pour ne pas nécessiter l'utilisation de petits pas de temps

En pratique, même en absence de forces extérieures au fluide, à cause des erreurs de discrétisation, une level set initialisée comme une fonction distance, ne verra pas la force associée nulle. Ceci peut provoquer des oscillations lorsque σ est élevé et ainsi tout de même rendre la simulation instable. De plus, en formulation vortex, il est nécessaire de calculer numériquement le rotationnel de 5.31, ce qui introduit indubitablement un degré de plus d'erreurs numériques pouvant déstabiliser d'autant plus la simulation.

Couplage avec la méthode de pénalisation

Pour pallier les problèmes énoncés précédemment, j'ai choisi de coupler le modèle de membranes élastiques avec le modèle de pénalisation utilisé pour les solides rigides, afin de pouvoir simuler des solides *quasi-rigides* à membranes élastique, et ainsi obtenir un comportement intéressant d'objets déformables. La discrétisation du terme de pénalisation n'assure pas exactement un mouvement rigide à l'intérieur et notamment de l'interface de l'objet. Ceci est dû à la discrétisation temporelle — où, par exemple, la force induite par la gravité vient déformer le domaine du solide à chaque pas de temps — et à la discrétisation spatiale, du fait de la réprésentation implicite de l'interface par une fonction level set et du paramètre de régularisation. Bien que dans la limite où $h \to 0$, $\varepsilon \to 0$, $\delta t \to 0$ et $\lambda \to 0$, le mouvement rigide est assuré, en pratique, l'interface sera tout de même déformée. Cette déformation sera ensuite réduite par la force élastique située au niveau de la membrane.

Dans une approche similaire à celle adoptée pour l'introduction de la force de traitement de la collision entre deux solides, la force élastique est introduite dans le fluide via l'étape de pénalisation, en modifiant la vitesse du solide élastique. La vitesse utilisée pour pénaliser celle du fluide est donc :

$$\bar{\mathbf{u}}' = \bar{\mathbf{u}} + \mathbf{f} \tag{5.32}$$

où $\bar{\mathbf{u}}$ est la vitesse du solide calculée par 5.22 et $\mathbf{f}$ la force provenant de la membrane élastique, obtenue par 5.31. De la même manière que pour la collision — où la moyenne de la force à l'intérieur du solide est calculée afin d'éviter l'introduction de vitesses trop importantes déstabilisant les calculs — j'ai choisi de régulariser la force élastique à l'interface en lissant la force sur un support de quelques points de grille, grâce à une Gaussienne :

$$\mathbf{f}_\varepsilon = \int_\Omega \zeta(\mathbf{x}, \varepsilon)\mathbf{f}(\mathbf{x})\,\mathrm{d}\mathbf{x}, \tag{5.33}$$

où ici $\zeta(\mathbf{x}, \varepsilon)$ est une Gaussienne 1D centrée en $\mathbf{x}$ et de support ε, par exemple :

$$\zeta(\mathbf{x}, \varepsilon) = \frac{1}{\sqrt{\pi}} e^{|\mathbf{x}|^2}.$$

Cette régularisation permet de lisser localement la force $\mathbf{f}$ qui, pour des coefficients de raideur importants, nécessiterait l'emploi de petit pas de temps afin de conserver une simulation stable. À la différence de la méthode utilisée pour la collision, ce lissage est effectué localement (et non sur tout l'intérieur du solide), afin de conserver l'aspect localisé de la force élastique sur la membrane. Ce lissage est d'autant plus important qu'en formulation vortex, c'est le rotationnel de cette force qui est injecté dans le champ de vorticité. Sans cette étape, et en présence de fortes irrégularités dans la force élastique, utiliser directement $\mathbf{f}$ introduirait de plus grandes erreurs numériques qu'avec $\mathbf{f}_\varepsilon$. En contrepartie, le comportement élastique de la membrane est lissé. Mais ceci n'est pas un problème important dans le cas de solides à membrane élastique où l'aspect « mou »de l'objet prime, tout particulièrement pour les applications en synthèse d'images.

4.3 Discussion

Les résultats obtenus avec cette méthode ne sont pas assez satisfaisants pour l'instant, c'est pourquoi je ne présente pas ici d'applications en synthèse d'images. Effectivement, la dynamique importante du fluide nécessaire au réalisme des animations implique des fortes déformations de la surface qui ne peuvent être compensées par l'élasticité de la membrane que grâce à des coefficients de raideur importants, nécessitant de très petits pas de temps pour assurer la stabilité de la simulation. Néanmoins, le principe de la méthode est une piste pour de futurs travaux sur les modèles d'objets déformables.

L'utilisation sous-jacente du modèle pour les solides rigides, couplé à celui de la membrane élastique, permet de limiter les déformations à l'intérieur et aux bords de l'objet, en réduisant la vitesse du fluide environnant, tout en permettant à l'objet de se déformer et à la force élastique d'agir. Ainsi, nous pouvons nous attendre à obtenir des déformations plus lisses de la membrane et, donc, stabiliser les calculs.

Néanmoins, le modèle de membrane élastique proposé ici ne permet pas de simuler rapidement des objets avec une élasticité forte, la plage des paramètres utilisables (pour des pas de temps non prohibitifs) reste tout de même assez faible. Il ne serait pas envisageable de simuler le mouvement d'un canard de bain en plastique, par exemple. De plus, ce modèle capte uniquement les déformations surfaciques. Ainsi, la membrane peut posséder plusieurs formes d'équilibre différentes, ce qui pourrait mener, dans le cas de larges déformations, à un comportement non désiré par l'animateur. Une formulation volumique de la force élastique semblerait alors plus adaptée et intuitive pour les applications en synthèse d'images.

La représentation de l'objet déformable par une level set est un atout car elle permet de mesurer le taux de déformation de ce dernier en plus d'éviter les difficultés pouvant apparaître lors de l'utilisation d'un maillage explicite pour la surface. Un autre avantange est que cette formulation capte, de manière intrinsèque, les changements de topologie. Ceci pourra permettre, à l'avenir, d'émuler le comportement de fluides « patteux », sans avoir à utiliser des modèles plus complexes de fluides visco-elasto-plastiques.

5 Bilan du chapitre

J'ai présenté dans ce chapitre une méthode novatrice en formulation vortex pour le calcul d'interactions fluide-solide rigide ainsi qu'un nouveau modèle de collision. La technique permettant d'imposer des conditions de non glissement aux bords d'un solide est basée sur une le principe de pénalisation de la vitesse, l'équation associée est adaptée en formulation tourbillon. J'ai proposé une manière de résoudre celle-ci exactement, menant à une méthode inconditionnellement stable. Cette approche est différente des techniques traditionnellement utilisées car elle est basée sur le principe de domaine fictif dans lequel le domaine du solide est immergé à l'intérieur de celui du fluide, ce qui permet l'utilisation de méthodes rapides pour la résolution des équations sur une grille uniforme. Le solide est capturé à l'aide d'une fonction level set qui est transportée avec la vitesse rigide. J'ai ainsi exposé une méthode efficace et possédant une erreur constante en temps, pour déplacer la fonction implicite associée au solide. Nous obtenons par cette technique des résultats qualitativement similaires à ceux de la littérature, sur le cas test d'une sphère tombant dans un fluide sous l'effet de la gravité, et pour des temps de calcul comparables. Les exemples proposés comme application en synthèse d'images montrent l'efficacité de la méthode dans sa globalité, dans le cadre d'interaction entre un solide rigide et un fluide à plusieurs phases, dont la surface libre est transportée par une méthode level set.

J'ai aussi proposé dans ce chapitre un nouveau modèle de collision, basé sur la représentation des solides par une fonction distance signée. Celle-ci permette de connaître la distance locale entre deux solides et cette information est ensuite injectée dans l'équation de la force de collision. Ainsi, la formulation level set de cette dernière permet d'éviter la pénétration de deux solides à géométrie quelconque tout en captant correctement les rotations induites par le choc. Le modèle est validé sur des cas tests avec ou sans fluide, notamment sur le problème du *drafting, tumbling and kissing* étudié numériquement dans [GPH+01,Pat01] dans le cadre de la méthode de domaine fictif traditionnelle en formulation vitesse.

Le modèle pour la simulation de membranes élastiques proposé ici est basé dans le même cadre que celui des surfaces libres et des solides rigides, par l'utilisation d'une fonction level set. Celle-ci permet de mesurer l'étirement de la membrane et ainsi est extraite une force élastique. Bien que ces travaux préliminaires ne sont pas encore concluants, le couplage avec la méthode de pénalisation utilisée afin de réduire les déformations de la surface et ainsi stabiliser les calculs, est une piste pour des travaux futurs dans le but de simuler efficacement des solides déformables en interaction avec un fluide.

Quatrième partie

Conclusion

J'ai proposé dans ce manuscrit relatant mes travaux effectués lors de ma thèse, une méthode complète pour le calcul d'interactions entre fluides multiphasiques et des solides, dans une formulation torubillon et avec une discrétisation par méthode particulaire. Les résultats obtenus montrent que ces méthodes sont une alternative intéressante aux méthodes plus traditionnelles de simulation de fluides axées sur la vitesse. En synthèse d'images, mes travaux sont les premiers à utiliser une méthode hybride particule-grille pour la simulation de fluide à plusieurs phases au contact de solides. Ce travail met en évidence un gain d'efficacité en comparaison des méthodes existantes, pour des résultats qualitativement similaires.

Récapitulatif et discussion

Le choix de calculer l'écoulement d'un fluide à partir de sa composante rotationnelle est judicieux car la complexité du mouvement — qui est un critère important, particulièrement en synthèse d'images — est totalement liée à la quantité de vorticité. Celle-ci caractérise les zones où apparaissent les phénomènes dynamiquement significatifs. D'un point de vue visuel, la vorticité est associée à la perception des comportements turbulents des écoulements. De plus, la localisation des tourbillons — en comparaison du champ de vitesse qui est, la plupart du temps, non nul dans tout le domaine — permet de focaliser les calculs sur une petite partie du domaine de calcul. L'utilisation de particules de vorticité est un atout majeur de la méthode grâce au traitement naturel de la convection. Cette étape du calcul est un frein important en formulation vitesse, du fait de la forte diffusion numérique induite par l'interpolation du champ de vitesse par la méthode semi-Lagrangienne, ou de la contrainte CFL imposée lors de l'utilisation de schémas de différences finies. Nous avons vu que la seule contrainte sur le pas de temps dans nos calculs, pour éviter le croisement des particules, est en pratique moins forte que les conditions CFL traditionnelles. Les résultats obtenus sur la simulation de fumée montrent que des pas de temps importants peuvent être choisis, tout en conservant un aspect dynamique significatif.

La formulation vortex ne nécessite pas la projection du champ de vitesse pour assurer l'incompressibilité, le champ de vitesse calculée est implicitement à divergence nulle. Ceci permet de s'abstraire du calcul du terme de pression utilisé en formulation vitesse pour assurer cette propriété. En contrepartie, le calcul direct de la vitesse sur les particules est coûteux et augmente quadratiquement avec le nombre de particules. La méthode hybride particule-grille que j'ai intégrée dans mes travaux permet de pallier ce problème en utilisant une grille sous-jacente, sur laquelle est calculée efficacement le mouvement du fluide. Afin d'éviter une déformation importante au cours du temps de la répartition des particules, j'ai appliqué une technique de remaillage. Cette méthode permet de conserver une répartition correcte des points de discrétisation, ce qui est un critère important pour la stabilité et la précision des calculs ; le choix de fonctions d'interpolation adaptées est un des points clef de la méthode. L'utilisation d'une grille permet en outre le calcul des termes non directement impliqués dans l'advection. J'ai proposé dans mes travaux de résoudre l'équation de

diffusion d'une manière exacte, pour des conditions aux bords périodiques, en même temps que le calcul de la fonction courant, ce qui permet un gain en précision et en temps de calcul.

Les travaux sur les méthodes vortex existants en synthèse d'images, bien que répondant très correctement à la question de la simulation de fumée, ne permettent pas de calculer la vaste panoplie d'écoulements et d'interactions possibles. La méthode hybride que j'ai adoptée dans ma thèse permet quant à elle la simulation de phénomènes complexes, notamment de fluides à plusieurs phases en interaction avec des solides.

Pour la simulation de fluides réalistes — par exemple l'air et l'eau en interaction — j'ai proposé dans mes travaux une nouvelle méthode efficace pour le calcul d'un fluide à plusieurs phases, en formulation vortex. Cette méthode est simple et permet d'éviter de calculer explicitement la pression qui réapparaît dans l'équation de vorticité en présence d'une variation de la densité. Afin de capter le mouvement de la surface séparant deux phases, j'ai utilisé une méthode particulaire pour le transport de la fonction level set captant l'interface, approche qui est similaire à celle des particules de vorticité. En pratique, j'ai remarqué qu'il est important que cette fonction soit proche d'une fonction distance signée afin d'éviter les instabilités naissant lors de la déformation de la level set ; pour ce faire j'ai intégré un algorithme de reinitialisation par la résolution d'une équation d'Hamilton-Jacobi.

Les résultats obtenus avec cette approche en formulation vortex sont les premiers en synthèse d'images, et démontrent la possibilité d'utiliser cette formulation pour des calculs complexes faisant intervenir plusieurs fluides en interaction. La localisation de la vorticité autour de l'interface est un autre point important qui ajoute à l'intérêt de l'utilisation d'une telle approche.

L'application de conditions de non glissement aux bords de solides est un problème difficile en méthode vortex. Dans mon manuscrit, j'ai proposé une méthode novatrice en formulation tourbillon pour répondre à cette question. Cette méthode est basée sur le principe de domaine fictif où le maillage utilisé pour les calculs sous-jacents reste simple — sur une grille uniforme dans le cas présent — et les solides sont immergés dans le fluide. Ce principe est très intéressant car il permet de s'abstraire de la discrétisation de l'espace autour de l'objet — ce qui est un problème difficile en 3D et dans le cas de géométries complexes — et de bénéficier de la rapidité des calculs sur une grille régulière. La méthode de pénalisation que j'ai mise en oeuvre dans une formulation vortex permet de calculer très naturellement et efficacement le couplage entre fluides et solides rigides. Cette méthode est plus intuitive et simple à mttre en oeuvre que les techniques plus traditionnelles, comme la méthode des panneaux couplée à un flux de vorticité. Il est possible de simuler des conditions de type non glissement aux bords de la boîte de calcul par l'insertion d'un cube englobant. Ceci permet de se ramener à un domaine périodique et facilite grandement le traitement des conditions aux bords.

J'ai montré qu'il était possible de résoudre exactement et de manière incondition-

nellement stable l'équation de pénalisation. Ceci permet d'utiliser des paramètres de pénalisation très grands et ainsi de diminuer les erreurs numériques. J'ai présenté une validation numérique montrant que cette méthode fournit des résultats qualitativement similaires à ceux obtenus par les méthodes couramment utilisées dans la littérature en formulation vitesse.

Dans le domaine de la synthèse d'images, ces travaux sont les premiers à proposer une méthode complète pour l'interaction fluide-solide en formulation vortex. Les résultats présentés ont été obtenus avec un temps de calcul par itération plus faible que celui exposé dans l'article *Rigid Fluids*. Il est important de noter qu'à la différence des méthodes de type ALE, où le maillage est adapté à la forme des objets en interaction, la méthode proposée ici ne permet pas d'assurer exactement la condition de non glissement sur le bord des solides, à cause des erreurs de discrétisation. Néanmoins, en pratique, le gain de temps et de simplicité en comparaison des méthodes ALE permet l'utilisation de maillages relativement fins et donc compense largement cette perte de précision.

À partir d'une représentation des solides par une fonction distance, j'ai développé une nouvelle méthode pour le traitement des collisions. Cette technique profite de l'information contenue dans la fonction level set pour calculer une force de répulsion. La formulation présentée permet de prendre en compte naturellement la composante rotationnelle de la collision, pour des solides à géométrie quelconque. Appliquée dans le calcul des interactions fluide-structure, la force de collision assure la non pénétration des objets — pénétration qui en pratique est possible à cause des erreurs de discrétisation — en les maintenant, à l'équilibre, à une distance de l'ordre du pas de grille. Cette nouvelle méthode pour le traitement des collisions ne vise pas à remplacer les méthodes traditionnelles pour la simulation de solides rigides où des calculs plus précis peuvent être mis en oeuvre. Par contre, elle s'accorde tout à fait dans le cas général où un modèle de collision simple ne pourrait être adopté. L'information pertinente dans le modèle présenté ici est la distance entre les objets, qui est fournie par la fonction level set. Aussi, la force est localisée sur l'interface, à l'endroit du contact, et elle peut ainsi servir au traitement de la collision entre des objets à géométrie quelconque, et éventuellement déformables.

En conclusion, la méthode que j'ai développée au cours de ma thèse permet de calculer les interactions entre fluides à plusieurs phases et solides rigides, dans un même cadre de calcul, en étant basée sur une représentation des surfaces concernées par des fonctions level set. Le principe du traitement du terme d'advection par des particules est aussi bien utilisé pour le transport de la vorticité que pour les surfaces libres. Ainsi, la méthode proposée permet de gérer d'une manière simple et cohérente les interactions entre fluides et solides.

Perspectives

Une des directions futures principale, dans la lignée de mes travaux, consiste à adopter un point de vue multi résolution et/ou adaptatif pour la résolution des équations. En effet, l'aspect localisé des tourbillons se prette particulièrement à ce type de discrétisation car, en dehors des zones de turbulence, la vorticité est nulle et ainsi un gain de temps et de précision importants est possible en localisant les points de discrétisation aux endroits nécessaires. Cette piste demande l'utilisation de techniques de remaillage adaptées afin d'interpoler correctement la vorticité portées par les particules sur une grille non uniforme. Comme les vortex sont souvent localisés au niveau des interfaces, la discrétisation adaptative pourrait être effectuée de manière concordante pour les grandeurs physiques et la level set utilisée pour l'interface entre deux phases.

Je pense que les méthodes particulaire et hybride pour la résolution des équations en formulation vortex pourraient être simplifiées et accélérées, afin de pouvoir calculer en temps interactif l'écoulement d'un fluide. Les méthodes de type *fast multipole* permettent déjà d'accélérer le calcul de la vitesse en regroupant spatialement les particules proches, mais ces techniques restent néanmoins difficiles à mettre en oeuvre. En sacrifiant quelque peu la précision des calculs, il serait possible de calculer en temps réel un écoulement complexe, pour la simulation de fumée par exemple, dans le cadre d'applications en synthèse d'images ; un petit nombre de particules permet déjà d'obtenir des comportements très dynamiques et réalistes. Afin de calculer efficacement le mouvement de l'interface séparant les deux phases, j'utiliserai une méthode du type *Twinned Meshes* [BN07] qui permet de déformer en temps interactif des surfaces implicites avec la possibilité de changements topologiques.

Lors de ma thèse, je me suis beaucoup intéressé aux méthodes de transport d'interface. La nécessité d'utiliser des techniques de reinitialisation pour les level set afin de stabiliser les calculs, ralentit grandement les temps de calcul. De nombreuses recherches sont aujourd'hui encore, axées sur le problème du transport d'interface. L'analyse de la déformation de la level set fournirait des pistes pour la dérivation de nouveaux schémas de transport incluant un terme de forçage, permettant de conserver au mieux la fonction distance. Une autre approche consisterait à remplacer la résolution coûteuse de l'équation d'Hamilton-Jacobi lors de la reinitialisation — qui consiste à transporter le niveau zéro de la level set avec une vitesse unitaire suivant la normale — par une technique de type champ de phase, basée sur la diffusion de l'interface à volume constant, puis par la récupération d'une fonction distance à partir de la fonction caractéristique lissée.

La représentation des interfaces interagissant avec le fluide par des fonctions level set est intéressante car elle permet de s'abstraire des difficultés liées au maillage des surface, et cette représentation permet d'extraire les déformations de celles-ci. Afin de pallier les difficultés rencontrées pour la simulation de membranes élastiques dans le but d'animer des objets déformables en interaction avec le fluide, l'utilisation d'un modèle de déformation du volume de l'objet me semble plus appropriée. De la même manière que pour les membranes élastiques, la déformation du volume peut

être étudiée à partir de la variation de la fonction level set associée au solide. Ainsi, une force élastique de volume peut être utilisée afin de simuler plus exactement le mouvement d'un solide déformable.

Annexe A

Compléments mathématiques

1 Lecture du document

1.1 Notations

Typographie

Dans tout le document, les vecteurs sont écrits en gras et les scalaires en police normale. Ainsi, la vorticité lorsqu'elle est vectorielle est écrite $\boldsymbol{\omega}$ et lorsqu'elle est scalaire ω. Les matrices sont dénotées par des lettres majuscules, et les vecteurs par des lettres minuscules.

Table des sigles utilisés

Sigle	Signification
$\mathbf{u}$	la vitesse
ρ	la densité
σ	le tenseur des contraintes visqueuses
p	la pression
μ	la viscosité dynamique
ν	la viscosité cinématique
ω	la vorticité
Ψ	la fonction courant
Φ	la fonction potentielle
Re	le nombre de Reynolds
$\mathbf{g}$	la gravité
$\mathbf{K}$	le noyau de Biot-Savart

Sigle	Signification
δt	le pas de temps
δx	le pas d'espace
h	le pas d'espace
δ	la masse de Dirac
ζ	une fonction cutoff
ε	un paramètre de régularisation
κ	la courbure d'une surface
ϕ	une fonction level set
χ	la fonction caractéristique associée à un solide
$\mathcal{F}$	la transformée de Fourier
$\bar{\mathbf{u}}$	la vitesse d'un solide
λ	le paramètre de pénalisation

2 Outils mathématiques

2.1 Numérique

Différences finies

Soit f une fonction discrétisée en espace et en temps, la résolution d'une EDP sur f consiste à calculer, au fur et à mesure du temps et en chaque point d'un **maillage**, la valeur de $\tilde{f}$ recherchée : $\tilde{f}(x_i, t_n)$. Le terme maillage représente le découpage de l'espace en points discrets (les sommets) liés par des connexions logiques permettant de parcourir le maillage de manière simple et lui fournissant une structure.

Pour résoudre une EDP, il faut définir les opérateurs de différenciation (les dérivées partielles) de manière discrète. On utilise alors des **schémas de différences finies** obtenus à partir du développement limité - ou développement en série de Taylor - des fonctions étudiées, que je détaille dans la partie 3.1. Ces schémas permettent d'approximer les dérivées d'une fonction discrète $\tilde{f}$ en fonction des valeurs de celle-ci. Par exemple, la dérivée première $\tilde{f}' = \frac{\partial \tilde{f}}{\partial x}$ de $\tilde{f}$ en un point x_i peut s'écrire :

$$\tilde{f}'(x_i) = \frac{\tilde{f}(x_i) - \tilde{f}(x_{i-1})}{h} + O(h) \text{ ou aussi } \tilde{f}'(x_i) = \frac{\tilde{f}(x_{i+1}) - \tilde{f}(x_i)}{h} + O(h)$$

Le terme $O(h)$ indique que les termes omis sont de l'ordre de h. De ces deux formules nous pouvons extraire des approximations à l' « ordre 1»où les termes en $O(h)$ ont été supprimés :

$$\tilde{f}'(x_i) \simeq \frac{\tilde{f}(x_i) - \tilde{f}(x_{i-1})}{h} \text{ et } \tilde{f}'(x_i) \simeq \frac{\tilde{f}(x_{i+1}) - \tilde{f}(x_i)}{h}$$

sont nommées respectivement différences finies avant et arrière.

Il est aisé de comprendre que plus h diminue (donc plus le nombre N de points augmente pour un intervalle donné), plus l'erreur sera faible : elle diminuera de

manière linéaire. Il est possible d'utiliser un schéma d'ordre plus élevé :

$$\tilde{f}'(x_i) \simeq \frac{\tilde{f}(x_{i+1}) - \tilde{f}(x_{i-1})}{2h}$$

ou bien

$$\tilde{f}'(x_i) \simeq \frac{-3\tilde{f}(x_i) + 4\tilde{f}(x_{i+1}) - \tilde{f}(x_{i+2})}{2h}$$

qu'on nomme respectivement différence centrée et décentrée d'ordre 2. Nous pouvons observer qu'un schéma plus précis (avec une erreur plus faible) possède généralement un support plus large, il nécessite plus de points, ce qui est plus coûteux en temps de calcul.

Attention, l'ordre de l'erreur du schéma n'est pas à confondre avec l'ordre de la dérivée de la fonction étudiée. En effet, il existe des schémas de différences finies pour les dérivées d'ordre supérieur (f', f'', etc.), par exemple la dérivée seconde centrée de $\tilde{f}$ en x_i approximée à l'ordre 2 est :

$$\tilde{f}''(x_i) \simeq \frac{\tilde{f}(x_{i+1}) - 2\tilde{f}(x_i) + \tilde{f}(x_{i-1})}{h^2}.$$

2.2 Schémas d'intégration

Il existe plusieurs manières de résoudre une EDP grâce aux schémas de différences finies vus précédemment. Suivant le problème auquel on s'intéresse, les schémas ont différentes propriétés principalement concernant la stabilité et l'ordre des erreurs commises. Les deux méthodes les plus courantes sont la méthode explicite et la méthode implicite dont je présente le principe ci-dessous.

Par exemple considérons l'**équation de la chaleur** en 1D :

$$\frac{\partial \phi}{\partial t} = \nu \Delta \phi = \nu \frac{\partial^2 \phi}{\partial x^2} \tag{A.1}$$

où ν est le coefficient de diffusivité. Nous cherchons la solution $\phi(x,t)$ avec $x \in \Omega$ et $t \in]0,T]$ dans une domaine spatial Ω, à partir de la solution initiale $\phi_0 = \phi(x,0)$ connue. Autrement dit, nous cherchons la valeur de la chaleur à chaque instant en chaque point du domaine. Comme nous l'avons vu dans la section précédente, il est possible d'approximer la dérivée $\frac{\partial \phi}{\partial t}$ à l'ordre 1 (nous utiliserons dorénavant le signe $=$ plutôt que $\simeq$ pour faciliter la lecture) :

$$\frac{\partial \phi}{\partial t} = \frac{\phi(x, t+\delta t) - \phi(x,t)}{\delta t}$$

où δt est appelé le **pas de temps**. En arrangeant différemment cette équation, nous obtenons :

$$\phi(x, t+\delta t) = \phi(x,t) + \delta t \frac{\partial \phi}{\partial t}.$$

Ainsi, il est possible de calculer toutes les valeurs de $\phi(x,t)$ pour tous les temps t si on en connait la dérivée $\frac{\partial \phi}{\partial t}$.

Le schéma implicite en temps est défini par un schéma de différences finies de type arrière :

$$\phi(x,t+\delta t) = \phi(x,t) + \delta t \frac{\partial \phi}{\partial t}(x,t+\delta t) \tag{A.2}$$

alors qu'un schéma explicite en temps est défini par un schéma de type avant :

$$\phi(x,t+\delta t) = \phi(x,t) + \delta t \frac{\partial \phi}{\partial t}(x,t). \tag{A.3}$$

La différence fondamentale entre ces deux formulations est le fait que la première comprend les valeurs inconnues de la dérivée de ϕ au pas de temps $t+\delta t$, alors qu'à partir de la deuxième il est beaucoup plus simple de trouver la solution $\phi(x,t+\delta t)$ car la partie droite de l'équation est entièrement connue à l'instant t courant.

Pour plus de clareté dans l'écriture des équations, notons $\phi_{i+m}^{n+l} = \phi(x+m\delta x, t+j\delta t)$; ainsi $\phi_i^{n+1} = \phi(x,t+\delta t)$. En développant la partie droite de l'équation A.1 avec un schéma de différences finies d'ordre 2 centré, et en réarrangeant les équations A.2 et A.3, nous obtenons respectivement :

$$\phi_i^{n+1} - \delta t \nu \frac{\phi_{i+1}^{n+1} - 2\phi_i^{n+1} + \phi_{i-1}^{n+1}}{\delta x^2} = \phi_i^n$$

et

$$\phi_i^{n+1} = \phi_i^n + \delta t \nu \frac{\phi_{i+1}^{n} - 2\phi_i^{n} + \phi_{i-1}^{n}}{\delta x^2}.$$

Les schémas implicites sont inconditionnellement stables ; en contrepartie, ils sont souvent plus difficiles et coûteux à résoudre numériquement.

2.3 Equation de la chaleur

Traitement implicite

Avec un schéma de différences finies implicite d'ordre 2, l'erreur est en $O(h^2)$. La méthode est inconditionnellement stable mais nécessite l'inversion d'une matrice de taille $N \times N$ où N est le nombre de points de grille. Reprenons l'exemple du traitement de l'équation de la chaleur de manière implicite en 1D :

$$u_i^{n+1} - \delta t \nu \frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{\delta x^2} = u_i^n.$$

Ce système d'équation peut s'écrire sous forme matricielle :

$$\begin{pmatrix} \dots & & & \\ & D & (1-2D) & D \\ \dots & & & \end{pmatrix} \begin{pmatrix} \dots \\ u_{i-1}^{n+1} \\ u_i^{n+1} \\ u_{i+1}^{n+1} \\ \dots \end{pmatrix} = \begin{pmatrix} \dots \\ u_{i-1}^{n} \\ u_i^{n} \\ u_{i+1}^{n} \\ \dots \end{pmatrix},$$

$$\begin{aligned} \mathbf{A}\mathbf{U}^{n+1} &= \mathbf{U}^n, \\ \mathbf{U}^{n+1} &= \mathbf{A}^{-1}\mathbf{U}^n \end{aligned}$$

où $D = -\delta\nu$. On peut repérer la forme particulière de la matrice $\mathbf{A}$: elle est tridiagonale (diagonale sur une bande), ce qui permet d'utiliser des méthodes rapides pour l'inverser (par décomposition LU ou de Choleski par exemple). Le lecteur pourra trouver la description et l'implémentation de telles méthodes dans le fameux *Numerical Recipes in C* [riCte07]. Par contre en dimensions supérieures la matrice à inverser a une forme « moins régulière »ce qui rend son inversion plus délicate et coûteuse.

Traitement explicite

Avec un schéma de différences finies explicite d'ordre 2, l'erreur est en $O(h^2)$. Le calcul est linéaire (donc très rapide) en le nombre N de points. Par contre la méthode souffre d'une contrainte de pas de temps : $\delta t \leq \frac{h^2}{2^d\nu}$ où d est la dimension du problème (généralement 2 ou 3). Je renvoie le lecteur à la nombreuse littérature concernant la résolution de l'équation de la chaleur pour obtenir la preuve de cette contrainte. Néanmoins, afin d'en comprendre la raison, le lecteur peut noter qu'il faut un certain temps pour que la « chaleur »se propage dans le milieu : ce que les schémas explicites ne permettent pas d'effectuer du fait de leur support compact en espace. Cette contrainte quadratique en le pas d'espace est très restrictive lorsque la viscosité ν est importante et il vaut parfois mieux (en terme de temps de calcul global de l'algorithme) résoudre l'équation de diffusion de manière implicite pour se dégager de cette contrainte.

Traitement dans le domaine fréquentiel

Dans le cas particulier d'un écoulement périodique, il est possible d'utiliser la Transformée de Fourier (TF) de la vitesse pour simplifier le traitement de l'équation de la chaleur, ici en 1D. Soit $\mathcal{F}$ l'opérateur de TF :

$$\hat{u}(\xi) = \mathcal{F}\left(u(x)\right) = \int_{-\infty}^{+\infty} f(x) e^{-i\xi x}\, dx$$

et son inverse :

$$u(x) = \mathcal{F}^{-1}\left(\hat{u}(\xi)\right) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} \hat{f}(\xi) e^{i\xi x}\, d\xi,$$

avec $i = \sqrt{-1}$. Ainsi, l'équation de la chaleur 1.22 en 1D devient par transformation :

$$\begin{aligned}
\mathcal{F}\left(\frac{\partial u}{\partial t}\right) &= \mathcal{F}\left(\nu \Delta u\right) \\
\mathcal{F}\left(\frac{\partial u(x,t)}{\partial t}\right) = \frac{\partial \mathcal{F}\left(u(x,t)\right)}{\partial t} = \frac{\partial \hat{u}(\xi,t)}{\partial t} &= \nu \frac{\partial^2}{\partial \xi^2}\int_{-\infty}^{+\infty} u(x,t) e^{-i\xi x}\, dx \\
&= \nu i^2 \xi^2 \int_{-\infty}^{+\infty} u(x,t) e^{-i\xi x}\, dx \\
&= \nu i^2 \xi^2 \hat{u} \\
&= -\nu \xi^2 \hat{u}
\end{aligned}$$

Ainsi pour résoudre l'équation, après un peu d'algèbre, il suffit de calculer :

$$u^{n+1} = \mathcal{F}^{-1}\left(\hat{u}^n - \delta t \nu \xi^2 \hat{u}^n\right) \tag{A.4}$$

de manière explicite — où la contrainte de stabilité est à satisfaire — ou bien

$$u^{n+1} = \mathcal{F}^{-1}\left(\frac{\hat{u}^n}{1+\delta t \nu \xi^2}\right) \tag{A.5}$$

de manière implicite qui, pour approximativement le même coût de calcul (à une division prêt), est à préférer car elle n'est pas soumise à la contrainte de stabilité. La résolution de l'équation de la chaleur dans le domaine fréquentiel a un coût linéaire, que ce soit de manière explicite ou implicite. Par contre, il faut bien sûr calculer la transformée de Fourier et son inverse de manière discrète : la méthode de Fast Fourier Transform (FFT) permet d'effectuer ce calcul en $O(NlogN)$ lorsque N est une puissance de 2. Le coût global de la résolution de l'équation de la chaleur est dans ce cas $O(N) + O(NlogN) = O(NlogN)$.

En pratique, il est rare de résoudre les équations de Navier-Stokes périodiques entièrement dans l'espace de Fourier car la résolution du terme non-linéaire ainsi que la contrainte d'incompressibilité (cf. partie 4.3) y sont plus complexes. C'est pourquoi la plupart du temps, les méthodes sont hybrides : on utilise la formulation fréquentielle pour la résolution de l'équation de diffusion et la formulation spatiale pour les autres parties des équations.

3 Analyse

3.1 Développement limité

Soit f une fonction m fois dérivable, le développement limité de f à l'ordre m est :

$$\begin{aligned} f(x) &= f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + ... + \frac{f^{(m)}(a)}{m!}(x-a)^m + o((x-a)^m) \\ &= f(a) + \sum_i^m \frac{f^{(i)}(a)}{i!}(x-a)^i + o((x-a)^m) \end{aligned}$$

où $o((x-a)^m) = R(x)$ est un reste représentant l'erreur d'approximation. Grâce à un peu d'algèbre, on peut exprimer la dérivée $f'(a)$ en un point d'une grille eulerienne en fonction des valeurs de f aux points de grille alentour. L'erreur d'approximation sera alors fonction du dernier terme retenu pour le développement limité. Par exemple, la dérivée première de f en un point x_i est approximée à l'ordre 1 par :

$$f'(x_i) = \frac{f(x_i) - f(x_{i-1})}{h} \text{ ou bien } f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{h}$$

ou à l'ordre 2 par :

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{2h}.$$

On peut observer qu'un schéma plus précis (avec une erreur plus faible) nécessite plus de points.

Soit f une fonction m fois dérivable, le développement limité de f à l'ordre m est :

$$\begin{aligned} f(x) &= f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + ... + \frac{f^{(m)}(a)}{m!}(x-a)^m + o((x-a)^m) \\ &= f(a) + \sum_{i}^{m} \frac{f^{(i)}(a)}{i!}(x-a)^i + o((x-a)^m) \end{aligned}$$

où $o((x-a)^m) = R(x)$ est un reste représentant l'erreur d'approximation. Si la dérivée $m^{\text{ième}}$ de f est assez régulière alors :

$$\lim_{x \to a} \frac{R(x)}{(x-a)^m} = 0.$$

Bibliographie

[ABF99] P. Angot, C.-H. Bruneau, and P. Fabrie. A penalization method to take into account obstacles in incompressible viscous flows. *Numer. Math.*, 81 :497–520, 1999.

[ACSYD05] P. Alliez, D. Cohen-Steiner, M. Yvinec, and M. Desbrun. Variational tetrahedral meshing. In *Proceedings of ACM SIGGRAPH 2005*, pages 617–625, 2005.

[AN05] Alexis Angelidis and Fabrice Neyret. Simulation of smoke based on vortex filament primitives. In *SIGGRAPH/Eurographics Symposium on Computer Animation*, July 2005.

[APKG07] Bart Adams, Mark Pauly, Richard Keiser, and Leonidas J. Guibas. Adaptively sampled particle fluids. *ACM Trans. Graph.*, 26(3) :48, 2007.

[Bar06] Adam W. Bargteil. *Tracking and Texturing Liquid Surfaces*. PhD thesis, University of California, Berkeley, 2006.

[BGOS06] Adam W. Bargteil, Tolga G. Goktekin, James F. O'Brien, and John A. Strain. A semi-lagrangian contouring method for fluid simulation. *ACM Transactions on Graphics*, 25(1), 2006.

[BIM04] Silvia Bertoluzza, Mourad Ismail, and Bertrand Maury. The fat boundary method : Semi-discrete scheme and some numerical experiments. *Lecture Notes in Comp. Sc. and Eng.*, 2004.

[BL92] R. P. Beyer and R. J. Leveque. Analysis of a one-dimensional model for the immersed boundary method. *SIAM J. Numerical Analysis*, 29 :332, 1992.

[BL03] P.P. Brown and D.F. Lawler. Sphere drag and settling velocity revisited. *J. Environ. Eng.*, 129 :222–231, 2003.

[BM85] J. T. Beale and A. Majda. High order accurate vortex methods with explicit velocity kernels. *Journal of Computational Physics*, 58, 1985.

[BN07] Antoine Bouthors and Matthieu Nesme. Twinned meshes for dynamic triangulation of implicit surfaces. In *Graphics Interface*, Montréal, may 2007.

[Bos08] Claire Bost. *Méthodes Level-Set et pénalisation pour le calcul d'interactions fluide-structure*. PhD thesis, Université Joseph Fourier - Grenoble Université, 2008.

[CBP05] Simon Clavet, Philippe Beaudoin, and Pierre Poulin. Particle-based viscoelastic fluid simulation. In *SCA '05 : Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 219–228, New York, NY, USA, 2005. ACM.

[CD97] M.-P. Cani and M. Desbrun. Animation of deformable models using implicit surfaces. In *IEEE Trans. Visual. Comput. Graph.*, volume 3, 1997.

[Cho68] Alexandre Joel Chorin. Numerical solution of the navier-stokes equations. *Mathematics of Computation*, 22(104) :745–762, 1968.

[CK00] Georges-Henri Cottet and Petros Koumoutsakos. *Vortex Methods : Theory and practice*. Cambridge University Press, 2000.

[CKOS00] G.-H. Cottet, P. Koumoutsakos, and M. Ould-Salihi. Vortex methods with spatially varying cores. *Journal of Computational Physics*, 162, 2000.

[CM04] Georges-Henri Cottet and Emmanuel Maitre. A level-set formulation of immersed boundary methods for fluid-structure interaction problems. *C. R. Acad. Sci. Paris*, 338 :581–586, 2004.

[CM06a] Georges-Henri Cottet and Emmanuel Maitre. A level set method for fluid-structure interactions with immersed surfaces. *Math. Models Meth. Appl. Sci.*, 16(3) :415–438, March 2006.

[CM06b] Georges-Henri Cottet and Emmanuel Maitre. A level set method for fluid-structure interactions with immersed surfaces. *Mathematical Models and Methods in the Applied Sciences*, 16 :415–438, 2006.

[CMM08] Georges-Henri Cottet, Emmanuel Maitre, and Thomas Milcent. Eulerian formulation and level set models for incompressible fluid-structure interaction. *M2AN*, 42 :471–492, 2008.

[CMT04] Mark Carlson, Peter J. Mucha, and Greg Turk. Rigid fluid : animating the interplay between rigid bodies and fluid. In *ACM Trans. Graph.*, volume 23, pages 377–384, 2004.

[CWH81] B.D. Nichols C. W. Hirt. Volume of fluid (vof) method for the dynamics of free boundaries. *JCP*, 39, Issue 1, 1981.

[DFG+06] J. Du, B. Fix, J. Glimm, X. Jia, X. Li, Y. Li, and L. Wu. A simple package for front tracking. *Journal of Computational Physics*, 213, 2006.

[DG96] Mathieu Desbrun and Marie-Paule Gascuel. Smoothed particles : a new paradigm for animating highly deformable bodies. In *Proceedings of the Eurographics workshop on Computer animation and simulation '96*, pages 61–76, New York, NY, USA, 1996. Springer-Verlag New York, Inc.

[DHPRF04] J. Donea, Antonio Huerta, H.-P. Ponthot, and A. Rodriguez-Ferran. *Encyclopedia of Computational Mechanics, chapitre 14*, volume 1. John Wiley & Sons, 2004.

[DMG89] P. Degond and S. Mas-Gallic. The weighted particle method for convection-diffusion equations, part 2 : the anisotropic case. *Math. Compute.*, 53, 1989.

[EFFM02] D. Enright, R. Fedkiw, J. Ferziger, and I. Mitchell. A hybrid particle level set method for improved interface capturing. *JCP*, 183, 2002.

[ESE07] M. Ellero, M. Serrano, and P. Espanol. Incompressible smoothed particle hydrodynamics. In *JCP*, pages 1732–1752, 2007.

[FAMO99] Ronald P. Fedkiw, Tariq Aslam, Barry Merriman, and Stanley Osher. A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method. *JCP*, 152, 1999.

[FJ] Matteo Frigo and Steven G. Johnson. Fftw : the fastest fourier transform in the west.

[FJL87] A. F. Fortes, D. D. Joseph, and L. S. Lundgren. Nonlinear mechanis of fluidization of beds of spherical particles. *J. Fluid Mechanics*, 177 :467–483, 1987.

[FM96] Nick Foster and Dimitri Metaxas. Realistic animation of liquids. *Graphical models and image processing : GMIP*, 58(5) :471–483, 1996.

[FOC+05] Bryan E. Feldman, James F. O'Brien, Mark Carlson, Bryan M. Klingner, and Tolga G. Goktekin. Fluids in deforming meshes. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation 2005*, July 2005.

[GHD03] O. Genevaux, A. Habibi, and J.-M. Dischler. Simulating fluid-solid interaction. In *Graphics Interface*, June 2003.

[GHS93] D. Golstein, R. Handler, and L. Sirovich. Modeling a no-slip flow with an external force field. *JCP*, 105 :354, 1993.

[GHS95] D. Golstein, R. Handler, and L. Sirovich. Direct numerical simulation of turbulent flow over a modelled riblet covered surface. *Journal of Fluid Mechanics*, 302 :333, 1995.

[GLG95] M. N. Gamito, P. F. Lopes, and M. R. Gomes. Two-dimension simulation of gaseous phenomena using vortex particles. In *In the 6th Eurographics Workshop on Computer Animation and Simulation*, volume 28, pages 3–15, 1995.

[GM77] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics : theory and application to non-spherical stars. In *Month notices Roy. Astrom. Soc.*, pages 375–389, 1977.

[GPH+99] R. Glowinski, T. W. Pan, T. I. Hesla, D. D. Joseph, and J. Périaux. A distributed lagrange multiplier/fictitious domain method for particulate flows. *Int. J. Multiphase Flow*, 25 :755–794, 1999.

[GPH+01] R. Glowinski, T. W. Pan, T. I. Hesla, D. D. Joseph, and J. Périaux. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies : Application to particulate flow. *J. Comput. Phys.*, 169(2) :363–426, 2001.

[GR00] Peter Smereka Giovanni Russo. A remark on computing distance functions. *JCP*, 163, Issue 1, September 2000.

[GSLF05] E. Guendelman, A. Selle, F. Losasso, and R. Fedkiw. Coupling water and smoke to thin deformable and rigid shells. In *SIGGRAPH/ACM TOG*, volume 24, pages 973–981, 2005.

[Gén06] Olivier Génevaux. *Simulation de Liquides à l'Aide des Équations de Navier-Stokes, et Visualisation, à Destination de l'Infographie.* PhD thesis, Université Louis Pasteur de Strasbourg, 2006.

[HA07] X. Y. Hu and N. A. Adams. An incompressible multi-phase sph method. In *JCP*, pages 264–278, 2007.

[Hes90] J. L. Hess. Panel methods in computational fluid dynamics. *Ann. Rev. Fluid Mech.*, 22 :255–274, 1990.

[HK05a] Simone E. Hieber and Petros Koumoutsakos. A lagrangian particle level set method. *J. Comput. Phys.*, 210(1) :342–367, 2005.

[HK05b] Jeong-Mo Hong and Chang-Hun Kim. Discontinuous fluidss. In *SIGGRAPH/ACM TOG*, volume 24, 2005.

[HPZ01] Howard H. Hu, N. A. Patankar, and M. Y. Zhu. Direct numerical simulations of fluid-solid systems using the arbitrary lagrangian-eulerian technique. *J. Comput. Phys.*, 169(2) :427–462, 2001.

[HW65] Francis H. Harlow and J. Eddie Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids*, 8(12) :2182–2189, 1965.

[Ism04] Mourad Ismail. *Méthode de la Frontière Élargie pour la Résolution de Problèmes Elliptiques dans des Domaines Perforés. Applicationaux Écoulements Fluides Tridimensionnels.* PhD thesis, Université Pierre et Marie Curie - Paris VI, 2004.

[JLM05] João Janela, Aline Lefebvre, and Bertrand Maury. A penalty method for the simulation of fluid - rigid body interaction. *ESAIM Proceedings*, 14 :115–123, September 2005.

[KAC92] A. A. Wheeler K. A. Cliffe, S. J. Tavener. An orthogonal mapping technique for the computation of a viscous free-surface flow. *Int. J. Numer. Methods Fluids*, 15, 1992.

[KFCO06] Bryan M. Klingner, Bryan E. Feldman, Nuttapong Chentanez, and James F. O'Brien. Fluid animation with dynamic meshes. In *Proceedings of ACM SIGGRAPH 2006*, pages 820–825, August 2006.

[KTO96] S. Koshizuka, H. Tamako, and Y. Oka. A particle method for incompressible viscous flow with fluid fragmentation. *JCFD*, 29, 1996.

[LC87] William E. Lorensen and Harvey E. Cline. Marching cubes : A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21(4) :163–169, 1987.

[LK93] A. Leonard and P. Koumoutsakos. High resolution vortex simulation of bluff body flows. *Journal of wind engineering and industrial aerodynamics*, 46-47 :315–325, 1993.

[LP00] Ming-Chih Lai and Charles S. Peskin. An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *JCP*, 160 :705–719, 2000.

[Mag07] Adrien Magni. Méthodes particulaires avec pénalisation pour la simulation d'écoulements autour d'obstacles. Master's thesis, Université Bordeaux 1, June 2007.

[Mai06] Emmanuel Maitre. Review of numerical methods for free interfaces. In *Ecole Thématique "Modèles de champ de phase pour l'évolution de structures complexes"*, 2006.

[Mau01] Bertrand Maury. A fat boundary method for the poisson equation in a domain with holes. *J. of Scientific Computing*, 16(3) :319–339, 2001.

[MCG03] Matthias Müller, David Charypar, and Markus Gross. Particle-based fluid simulation for interactive applications. In *SCA '03 : Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 154–159, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[MP00] N. Mordant and J.-F. Pinton. Velocity measurement of a settling sphere. *The European Physical Journal B*, 18 :343–352, 2000.

[MS94] Stanley Osher Mark Sussman, Peter Smereka. A level set approach for computing solutions to incompressible two-phase flow. *JCP*, 114, Issue 1, September 1994.

[MSKG05] Matthias Müller, Barbara Solenthaler, Richard Keiser, and Markus Gross. Particle-based fluid-fluid interaction. In *SCA '05 : Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 237–244, New York, NY, USA, 2005. ACM.

[Pat01] Neelesh A. Patankar. A formulation for fast computations of rigid particulate flows. *Center Turbul. Res., Ann. Res. Briefs*, pages 185–196, 2001.

[Pes77] C. S. Peskin. Numerical analysis of blood flow in the heart. *J. Comput. Phys.*, 25 :220–252, 1977.

[PJB+02] T. W. Pan, D. D. Joseph, R. Bai, R. Glowinski, and V. Sarin. Fluidization of 1204 spheres : Simulation and experiment. *J. Fluid Mechanics*, 451 :169–191, 2002.

[PK05] Sang Il Park and Myoung Jun Kim. Vortex fluid for gaseous phenomena. In *SIGGRAPH/Eurographics Symposium on Computer Animation*, July 2005.

[Pon06] Philippe Poncet. Finite difference stencils based on particle strength exchange schemes for improvement of vortex methods. *Journal of Turbulence*, 7(23), 2006.

[PP93] C. S. Peskin and B. F. Printz. Improved volume conservation in the computation of flows with immersed elastic boundaries. *JCP*, 105, 1993.

[PPCC05] Frédéric Pighin, Sanjit Patel, Anson Chu, and Jonathan M. Cohen. Oriented particle level set for fluid simulation. *Symposium on Computer Animation, poster*, 2005.

[PTB+03] S. Premoze, T. Tasdizen, J. Bigler, A. Lefohn, and R. Whitaker. Particle based simulation of fluids. In *Proceedings of Eurographics*, pages 401–410, 2003.

[PWS+02] P. Ploumhans, G. S. Winckelmans, J. K. Salmon, A. Leonard, and M.S. Warren. Vortex methods for direct numerical simulation of three-dimensional bluff body flows : application to the sphere at re=300,500 and 1000. *JCP*, 178 :427–463, 2002.

[riCte07] Numerical recipes in C third edition. . online, 2007.

[SAC+99] Dan Stora, Pierre-Olivier Agliati, Marie-Paule Cani, Fabrice Neyret, and Jean-Dominique Gascuel. Animating lava flows. In *Proceedings of the 1999 conference on Graphics interface '99*, pages 203–210, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[SB96] E. M. Saiki and S. Biringen. Numerical simulation of a cylinder in uniform flow : Application of a virtual boundary method. *JCP*, 123 :450, 1996.

[Set99] J. A. Sethian. *Level Set Methods and Fast Marching Methods.* Cambridge University Press, 1999.

[SP05] Nitin Sharma and Neelesh A. Patankar. A fast computation technique for the direct numerical simulation of rigid particulate flows. *J. Comput. Phys.*, 205(2) :439–457, 2005.

[SRF05] A. Selle, N. Rasmussen, and R. Fedkiw. A vortex particle method for smoke, water and explosions. In *SIGGRAPH/ACM TOG*, volume 24, pages 910–914, 2005.

[Sta99] Jos Stam. Stable fluids. In Alyn Rockwood, editor, *Siggraph 1999, Computer Graphics Proceedings*, pages 121–128, Los Angeles, 1999. Addison Wesley Longman.

[Str01] J. A. Strain. A fast semi-lagrangian contouring method for moving interfaces. *JCP*, 169, 2001.

[SVAC08] A. Sarthou, S. Vincent, P. Angot, and J.-P. Caltagirone. The sub mesh penalty method. *5th International Symposium on Finite Volumes for Complex Applications*, 2008.

[Wey06] Lisl Weynans. *Méthode particulaire multi-niveaux pour la dynamique des gaz, application au calcul d'écoulements multifluides.* PhD thesis, Université Joseph Fourier, December 2006.

[XDL94] T. Chan X-D. Liu, S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *JCP*, 115, 1994.