



HAL
open science

Développements et applications de la méthode SPH aux écoulements visqueux à surface libre.

Jean-Marc Cherfils

► **To cite this version:**

Jean-Marc Cherfils. Développements et applications de la méthode SPH aux écoulements visqueux à surface libre.. Mécanique [physics.med-ph]. Université du Havre, 2011. Français. NNT: . tel-00607041

HAL Id: tel-00607041

<https://theses.hal.science/tel-00607041>

Submitted on 7 Jul 2011

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

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

Thèse
présentée pour l'obtention du grade de
DOCTEUR DE L'UNIVERSITÉ DU HAVRE

Spécialité:
MÉCANIQUE DES FLUIDES

par

Jean-Marc CHERFILS

Laboratoire d'accueil : Laboratoire Ondes & Milieux Complexes - FRE CNRS 3102
Equipe : Hydrodynamique & Acoustique Marine
École Doctorale : Sciences Physiques, Mathématiques et de l'Information pour l'Ingénieur

DÉVELOPPEMENTS ET APPLICATIONS DE LA MÉTHODE SPH AUX ÉCOULEMENTS VISQUEUX À SURFACE LIBRE

Soutenue le 17 mars 2011 devant le jury composé de:

Rapporteurs

David	LE TOUZÉ	Maître de Conférences à l'Ecole Centrale de Nantes
Yves-Marie	SCOLAN	Professeur à l'Ecole Centrale de Marseille

Examineurs

Serge	HUBERSON	Professeur à l'Université de Poitiers
Mathieu	MORY	Professeur à l'Université de Pau et des Pays de l'Adour
Grégory	PINON	Maître de Conférences à l'Université du Havre
Elie	RIVOALEN	Professeur à l'INSA de Rouen (Directeur de thèse)



Je souhaite remercier les membres du Laboratoire Ondes & Milieux Complexes de l'Université du Havre. En particulier, ceux de l'équipe Hydrodynamique Marine. Mon directeur de thèse, Elie, pour sa passion communicative, ses conseils avisés et son aide. Grégory, pour son soutien et son accompagnement quotidien. Rémy et Paul, pour leur contribution au développement et le partage de connaissances au cours de leur stage. Gaelle, Jérôme, Julie et toutes les personnes qui ont été ou sont actuellement de passage au laboratoire.

Je remercie les membres du jury d'avoir accepté d'y participer et pour le grand intérêt qu'ils ont montré pour mon travail.

Un grand merci à ma famille : mes parents, mes frères et belles-sœurs et mes nièces, pour leur soutien, leur écoute et leur tolérance.

Je remercie mes amis musiciens : PPR, Guyom, Tonio, Francesco, Nico, Oliv' pour leur soutien et leurs conseils. Ils se sont montrés d'un grand secours quand des notes de musique ou toute autre activité plus ou moins culturelle me devenait indispensable. Des gros bisous à Anh, Arnaud, Raf, Sophie, Gab', Allane, Steph, Stany...

Je remercie également toutes les personnes, non citées ici, qui ont joué un rôle dans le déroulement de ces quatre années et ont compris l'implication morale que cette thèse pouvait demander.

Table des matières

Introduction	1
I La méthode Smoothed Particle Hydrodynamics	5
1 Contexte et modélisation en Mécanique des Fluides	9
1.1 Descriptions d'écoulements	9
1.1.1 Description eulérienne	10
1.1.2 Description lagrangienne	10
1.1.3 Méthodes arbitrairement Lagrange-Euler (ALE)	11
1.2 Discrétisation du domaine de calcul	11
1.2.1 Méthodes maillées	12
1.2.2 Méthodes sans-maillage	13
1.2.3 Méthodes adaptatives	14
1.2.4 Méthodes particulières	15
2 Méthodes particulières - Outils Mathématiques	19
2.1 Notations et définitions	19
2.1.1 Approximation et discrétisation	19
2.1.2 Opérateur différentiel	20
2.1.3 Opérateur d'approximation	21
2.1.4 Noyau d'interpolation	21
2.1.5 Régularité d'une fonction à interpoler	22
2.2 Interpolation	22
2.2.1 Opérateur intégral	22
2.2.2 Choix du noyau d'interpolation	24
2.3 Approximation de dérivées	29
2.3.1 Approche SPH	29
2.3.2 Approche PSE	31
2.4 Noyau employé	32

3	Application aux équations de la Mécanique des Fluides	37
3.1	Modélisation d'un fluide incompressible	37
3.1.1	Approche faiblement compressible (WCSPH)	38
3.1.2	Approche incompressible (ISPH)	40
3.2	Équations d'Euler	42
3.3	Équations de Navier-Stokes	45
II	Mise en œuvre, validations et applications	51
4	Étirement d'une goutte en deux dimensions	55
4.1	Description du problème	55
4.2	État de l'art sur la modélisation de cet écoulement	57
4.3	Modélisation par SPH : schéma simple	58
4.4	Modélisation par SPH : schéma de Colagrossi [35]	62
4.4.1	Correction de la vitesse « XSPH »	62
4.4.2	Viscosité artificielle	64
4.4.3	Réinitialisation de la masse volumique « MLS »	66
4.4.4	Correction de l'instabilité en tension	67
4.5	Modélisation par SPH : schéma de Ferrari <i>et al.</i> [75]	69
4.5.1	Flux Rusanov	69
4.5.2	Convergence du schéma	73
4.6	Amélioration de la précision	77
4.6.1	Normalisation du noyau et renormalisation des gradients	77
4.6.2	Influence de la disposition initiale des particules	86
5	Effondrement d'une colonne d'eau en deux dimensions	91
5.1	Description du problème	91
5.2	État de l'art sur la modélisation de cet écoulement	93
5.3	Modélisation par SPH	95
5.3.1	Conditions aux limites : parois solides	95
5.3.2	Conditions aux limites : surface libre	104
5.3.3	Initialisation	104
5.4	Application au cas $L/H = 2$	106
5.4.1	Résultats préliminaires	106
5.4.2	Corrections	108
5.4.3	Post-traitement et validation	115

6	Méthode de la frontière immergée et application aux écoulements visqueux	121
6.1	Méthode de la <i>frontière immergée</i>	122
6.1.1	SPH remaillée et <i>frontière immergée</i> , méthode de Hieber et Koumoutsakos [93] : RSPH-IB	123
6.1.2	SPH sans remaillage et <i>frontière immergée</i> corrigée par moindres carrés : SPH-IBMLS	127
6.2	Écoulement autour d'un carré forcé dans une boîte rectangulaire	129
6.2.1	État de l'art sur la modélisation de cet écoulement	131
6.2.2	Comparaison des modèles RSPH-IB et SPH-IBMLS à $Re = 50$	141
6.2.3	Validation du modèle SPH-IBMLS à $Re = 50$, $Re = 100$ et $Re = 150$. . .	145
6.2.4	Résultats à $Re = 300$ et $Re = 600$	149
7	Propagation d'ondes de gravité en canal à houle	157
7.1	Description du problème	157
7.1.1	Génération de la houle	157
7.1.2	Condition d'amortissement	158
7.2	Propagation de la houle par SPH	160
7.3	Plaque horizontale immergée en canal à houle	165
7.3.1	Viscosité négligée	168
7.3.2	Prise en compte de la viscosité	169
III	Aspects informatiques	177
8	Optimisation	181
8.1	Programmation objet ou tableaux ?	181
8.2	Recherche des interactions	185
8.2.1	Algorithme de recherche <i>Tree Search</i>	185
8.2.2	Algorithme de recherche <i>Linked List</i>	190
8.2.3	Accélération par la méthode de Verlet	192
8.3	Autres améliorations et perspectives	195
9	Parallélisation	199
9.1	Décomposition de domaine	200
9.2	Modifications du code séquentiel	202
9.2.1	Initialisation des simulations	206
9.2.2	Entrées et sorties fichier	207
9.2.3	Détection des zones d'interaction	209
9.2.4	Communications	209
9.2.5	Équilibrage de la charge	210

9.3	Analyse des performances	215
Conclusion		219
Annexes		223
A	Traitement des opérateurs différentiels en méthode PSE	225
A.1	Méthodologie	225
A.2	Application au gradient	227
A.3	Application à l'opérateur laplacien	228
A.4	Noyaux de dérivation	228
A.4.1	Noyaux gaussiens	229
A.4.2	Noyaux B-Splines	230
B	Correction de l'instabilité en tension	235
B.1	Présentation du cas test	235
B.2	Comparaison des modèles de Colagrossi [35] et de Ferrari <i>et al.</i> [75]	236
C	Intégration en temps	239
C.1	Variables d'intégration	239
C.2	Schémas	240
C.3	Condition CFL	245
D	Application et étude de l'influence des paramètres géométriques	247
D.1	Influence du rapport L/H sur la vitesse du front	249
D.2	Pression à l'impact	251
D.3	Débit du jet vertical post-impact	251
D.4	Cas $L/H = 1, 0$ et $(d - L)/H = 1, 0$ en deux dimensions d'espace	258
D.5	Cas $L/H = 1, 0$ et $(d - L)/H = 1, 0$ à trois dimensions d'espace	259
Bibliographie		289
Ressources électroniques		292

Introduction

La simulation numérique est une discipline récente, qui a connu un essor très rapide, permis par l'explosion des capacités informatiques. Certains considèrent que la première simulation numérique (et non pas le premier calcul numérique d'une *solution* connue mais complexe) fût réalisée dans le cadre du projet *Manhattan*, pour l'étude du processus de détonation nucléaire. L'expérience de Fermi-Pasta-Ulam [73], en 1953, est présentée comme la première simulation numérique à intérêt civil. D'autres considèrent que la simulation numérique a été imaginée bien plus tôt [71]. En effet, avant même l'apparition des premiers ordinateurs, des travaux de recherche portaient déjà sur la discrétisation d'équations et sur les algorithmes permettant de les résoudre [47]. Quand aucune solution analytique n'existe pour un problème donné, il n'y a pas d'autre choix que celui de tenter une reproduction de la réalité et d'observer son évolution. Les modèles réduits et autres dispositifs expérimentaux sont essentiels à l'observation de phénomènes et à leur compréhension. En Mécanique des Fluides, ils permettent notamment d'observer un écoulement que l'on veut étudier sans qu'il ne soit perturbé par un facteur incontrôlé. En simplifiant cet écoulement, on peut mettre en évidence l'influence de chaque paramètre du problème traité. Cependant, l'approche expérimentale, bien qu'indispensable, a ses limites. D'une part, certaines données ne sont pas mesurables, techniquement. D'autre part, la fabrication de modèles réduits limite souvent les dimensions et la diversité des paramètres du problème étudié. Ces modèles ont enfin un coût souvent non-négligeable, parfois même prohibitif. C'est pour ces raisons que la simulation numérique est devenue indispensable, à la fois pour la recherche, mais aussi dans l'industrie. Modéliser un écoulement et le faire évoluer par résolution des équations adéquates autorise l'étude de grandeurs inaccessibles expérimentalement. Les coûts matériels sont moindres, permettant la réalisation d'un grand nombre d'essais. C'est d'ailleurs dans les domaines où l'expérience est difficilement envisageable (Astrophysique, Physique relativiste, Biologie du corps humain. . .) que la simulation numérique a connu ses plus grandes avancées.

Dans d'autres disciplines, telles que la Mécanique des Fluides, l'expérience et la simulation sont actuellement des outils complémentaires. Les données recueillies au cours de campagnes d'essais sont nécessaires à la validation des nouveaux codes de calcul. A l'inverse, la modélisation assistée par ordinateurs et la simulation sont une étape cruciale pour la conception de prototypes, leur optimisation et la réduction de leur coût. En Aérodynamique et Hydrodynamique par exemple, la simulation joue un rôle primordial dans l'étude et la compréhension de la turbulence. Dans le



domaine du génie côtier et des constructions marines, les applications de la simulation numérique sont multiples. Elle permet par exemple de reproduire des conditions de mer extrêmes et de tester la résistance et la fiabilité d'ouvrages marins existants. Elle peut être un outil d'aide à la conception de nouvelles digues : la possibilité de modifier à volonté le modèle testé peut aboutir à de meilleurs résultats en termes de réflexion de la houle ou de franchissement lors de tempêtes. La simulation offre la possibilité d'imaginer de nouveaux ouvrages de protection ou de conversion de l'énergie marine, plus efficaces, moins coûteux ou plus respectueux de l'environnement.

En conception navale, la tenue en mer de bateaux et leur manœuvrabilité sont régulièrement testés sur des modèles réduits en bassin (figure 1). Les moyens techniques et humains déployés lors de telles campagnes d'essais sont très importants. L'informatique est alors une étape essentielle pour la conception des maquettes et leur amélioration.

Cette thèse s'est déroulée au Laboratoire Ondes & Milieux Complexes de l'Université du Havre, dans l'équipe Hydrodynamique Marine. Les activités de recherche portent sur la conception et la fiabilité d'ouvrages de protection des côtes et sur les énergies marines. Ainsi, une des thématiques est notamment l'étude de l'action de la houle sur les ouvrages marins, comme une fine plaque horizontale immergée. Dans ce cas précis, les techniques expérimentales (LDV, PIV...) permettent, en canal à houle, de mesurer le champ de vitesse à partir de particules en suspension. On peut également mesurer avec précision l'amortissement et la déformation de la surface libre autour d'obstacles [20]. En revanche, l'analyse complète de données telles que la pression reste difficile, actuellement, sans intrusion dans le milieu fluide étudié. C'est à ce niveau que la simulation numérique peut apporter une réponse complémentaire, par exemple. La simulation peut également être un outil précieux pour l'élaboration de nouvelles formes de structures marines, avant toute tentative de mesure expérimentale.

Le but de ces travaux de thèse était donc la conception et validation d'un code de simulation en méthode Smoothed Particle Hydrodynamics (SPH), à surface libre, pour des applications dans le domaine de l'hydrodynamique marine. Plusieurs méthodes, autres que SPH, permettent la simulation d'écoulements fluides et la prise en compte d'une surface libre : Boundary Element Method (BEM), Volume Of Fluid (VOF)... Cependant, aucune d'entre elles ne permet, à la fois, d'accéder aux propriétés en tout point du domaine fluide et de suivre naturellement, sans la mise en œuvre d'algorithmes complexes, l'évolution de la surface libre. Comme cela est décrit plus loin, la méthode SPH apporte déjà, par nature, une solution à ce problème de prise en compte de la surface libre (voir chapitre 5). C'est d'ailleurs cette capacité qui est à l'origine de son succès. Succès tempéré toutefois par son manque de *maturité*. En effet, aucun code SPH n'est actuellement l'objet d'une commercialisation à grande échelle, pour la simple raison que la méthode requiert un savoir-faire et une maîtrise des nombreux paramètres disponibles. Sans prise de précautions, une telle méthode peut proposer des solutions très variées du même cas simulé. Cependant, cela est vrai aussi pour de nombreux codes commerciaux actuels. Mais pour certains, la principale originalité de la méthode SPH, l'absence de maillage, est encore considérée comme une faille de la méthode

et la raison évidente d'un manque de précision. De plus, les temps de simulation nécessaires étant relativement importants, la méthode ne suscite pas encore l'intérêt qu'elle mérite.

Pourtant, la recherche est active autour de la méthode SPH, tout particulièrement au cours des quinze dernières années, en raison de son application à l'hydrodynamique marine [155]. Nous nous sommes donc intéressés à la mise en œuvre d'un modèle robuste et performant, dont les paramètres seraient contrôlés et indépendants, pour la modélisation d'écoulements à surface libre et leur interaction avec différents types de structures. Nous avons porté un soin particulier à la stabilité du modèle, à la réduction du nombre de paramètres spécifiques à la méthode ou aux écoulements simulés.

La prise en compte de conditions limites, toujours problématique en méthode particulière, et en particulier en méthode SPH, a d'ailleurs été étudiée. Nous avons mis en place une extension de la méthode dite *de frontière immergée* [183], pour la représentation d'objets, fixes ou mobiles, à la géométrie complexe, dans le fluide. Cette nouvelle approche, robuste et algorithmiquement simple, peut être une solution adaptée au problème de la prise en compte de conditions d'adhérence, d'une part, sur des géométries arbitraires et variées, d'autre part.

Enfin, une partie du travail a porté sur les aspects informatiques de la conception d'un tel code de calcul. L'optimisation et la parallélisation du logiciel ont été effectués en vue de rendre envisageables de grandes simulations, en deux ou trois dimensions, moyennant des temps de calcul raisonnables.

Ce manuscrit s'articule autour des principales tâches qui nous ont occupées au cours de cette thèse. Dans la première partie, la méthode SPH est présentée. Les équations de la Mécanique sont mises en forme selon le formalisme de la méthode et les différentes variantes présentes dans la bibliographie sont décrites. Ce travail, que l'on pourrait croire préliminaire à l'écriture d'un code de calcul, a cependant nécessité beaucoup de simulations différentes afin de comparer les approches. La seconde partie présente l'application de ce modèle SPH à trois configurations différentes, en guise de validation. Bien d'autres cas ont été étudiés au cours de cette thèse mais les trois choisis sont essentiels car ils ont permis l'analyse progressive de chaque élément du modèle SPH. D'abord, le schéma non-visqueux à surface libre, seul, est étudié puis amélioré sur le cas d'une goutte de fluide étirée. Ensuite, un second cas représentant l'effondrement d'une colonne d'eau soumise à l'action de la gravité permet la validation du code sur un problème combinant conditions aux bords solides et surface libre. Enfin, le modèle visqueux est analysé et une nouvelle méthode de conditions aux limites pour les corps immergés est présentée. Elle est appliquée à l'écoulement généré par le forçage d'un cylindre de section carrée dans une boîte.

Le modèle final est employé pour la simulation d'ondes de gravité se propageant en canal. Cette application montre l'intérêt de la méthode SPH pour l'étude de la houle et de son action sur un obstacle immergé. Une plaque horizontale est ainsi placée dans le canal et utilisée comme amortisseur de houle (figure 2). Les différents apports de cette thèse sont exploités sur la simulation de cet écoulement, les atouts et limites du modèle sont alors discutés.





FIGURE 1 – Bassin de traction B600 du Bassin d’Essais des Carènes de Val de Reuil (gauche) et essai de résistance sur un modèle réduit dans le même bassin. (Photos : Bassin d’Essais des Carènes)

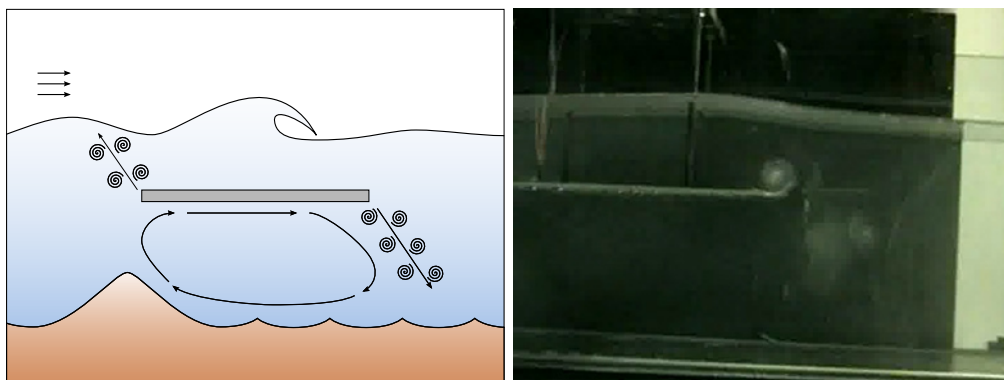


FIGURE 2 – Écoulement oscillant autour d’une plaque immergée (gauche). Tourbillon généré au bord de la plaque en canal à houle (droite).

Les aspects plus techniques de l’optimisation et la programmation sont exposés dans la troisième partie. Leur étude était nécessaire pour envisager des simulations à grandes échelles de temps et d’espace. Des solutions et perspectives pour un développement futur seront finalement proposées.

Première partie

**La méthode Smoothed Particle
Hydrodynamics**



Dans le domaine de la simulation numérique en Mécanique des Fluides, *Smoothed Particle Hydrodynamics* (SPH) est une méthode récente, en constante évolution. Apparue initialement en 1977 dans deux articles de Gingold et Monaghan [78] et de Lucy [138], elle se présente alors comme un modèle pour la simulation de corps gravitaires en Astrophysique. La méthode SPH y est définie comme une nouvelle méthode particulière "pure" où la particule, porteuse d'informations, permet la représentation du milieu fluide modélisé. La méthode SPH peut effectivement être vue comme un formalisme mathématique permettant la résolution d'une grande variété de systèmes d'équations aux dérivées partielles à la dynamique complexe. En revanche, les outils mis en œuvre dans ce formalisme sont déjà bien connus ou se rapprochent grandement de ceux employés dans d'autres méthodes numériques plus anciennes. Dans la partie suivante, nous allons tenter de situer SPH dans l'ensemble des méthodes connues de Mécanique des Fluides Numériques. Nous allons ensuite définir l'ensemble des outils mathématiques employés. Enfin, les équations de Navier-Stokes seront mises en forme selon le formalisme SPH, les différentes stratégies d'approximation de chaque terme des équations seront décrites.



Chapitre 1

Contexte et modélisation en Mécanique des Fluides

Afin de cerner l'originalité des méthodes développées au cours de ces travaux de thèse, il est nécessaire de situer brièvement la modélisation employée dans son contexte. Il existe en effet un grande variété d'approches à la modélisation en Mécanique des Fluides. Il s'agit ici de les classer de façon conceptuelle sans prétendre en établir une liste exhaustive. Ainsi, nous allons d'abord les différencier selon la description de l'écoulement employée, qui intervient directement sur l'écriture des équations étudiées. Ensuite, la discrétisation va être présentée comme un critère supplémentaire, indépendant du précédent, bien que certaines méthodes de discrétisation soient plus adaptées à une description donnée : une méthode maillée pour un écoulement décrit de façon eulérienne ou une méthode particulière pour du suivi lagrangien. Enfin, nous allons placer la méthode SPH dans ce contexte et présenter quelques codes de calcul actuels.

1.1 Descriptions d'écoulements

Les méthodes numériques sont couramment divisées en deux catégories, selon la description choisie pour l'écoulement étudié : eulérienne ou lagrangienne [11, pages 71-73][115, §3-§7 et §13-§16]. Le choix entre ces deux façons de procéder est crucial car il détermine l'écriture du système d'équations aux dérivées partielles à résoudre et chaque description pose ses propres limites à la variété des écoulements pouvant être modélisés. Plus récemment, les méthodes ALE (Arbitrairement Lagrange-Euler) sont apparues, proposant non pas une alternative aux deux descriptions précédentes, mais plutôt la possibilité de passer de l'une à l'autre selon certains critères définis par l'utilisateur. Ces représentations diffèrent par la position de l'*observateur* (qui représente un point de calcul : maille, particule ou volume de contrôle...) par rapport à l'écoulement.



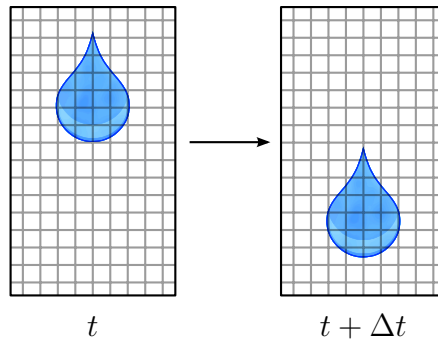


FIGURE 1.1 – Description eulérienne d'un écoulement.

1.1.1 Description eulérienne

En description eulérienne, l'*observateur* est fixe par rapport à l'écoulement, il voit le fluide "défiler" devant lui (figure 1.1). Les méthodes eulériennes sont, depuis longtemps, les plus répandues en Mécanique des Fluides Numérique. Cela en raison de l'aisance procurée par le fait de ne pas déplacer les éléments de la discrétisation (mailles, particules ...). On trouve notamment les méthodes de Différences Finies (DF) [99], Éléments Finis (EF) [241] et Volumes Finis (VF) [224], les méthodes de Volumes de Fluides (VOF) [95]... Ces méthodes proposent généralement une précision nettement supérieure que les alternatives lagrangiennes, sauf quand il s'agit de modéliser une surface libre ou des zones de forte vorticit . Dans le cas des différences finies, une montée en ordre des schémas est permise par la forme cartésienne du maillage. Les outils mathématiques actuels sont d'ailleurs conçus pour ce genre de discrétisation. A l'inverse, ces méthodes souffrent généralement d'un défaut de représentation des interfaces, qu'il s'agit de capturer au sein du domaine de calcul. Des algorithmes existent à cet effet (fractions de volume, fonctions de couleur) mais la complexité de la méthode s'en trouve accrue.

1.1.2 Description lagrangienne

En description lagrangienne, l'*observateur* fixe un volume infinitésimal de fluide et suit son évolution pendant toute la durée de l'écoulement. Ainsi, depuis la position initiale de cette particule élémentaire, l'utilisateur a connaissance de toute sa trajectoire dans le domaine de calcul (figure 1.2). A l'inverse des méthodes eulériennes, les méthodes lagrangiennes permettent le suivi d'interfaces sans traitement algorithmique supplémentaire, a priori. Elles s'appliquent donc plus particulièrement à la modélisation d'écoulements à surface libre, ou diphasiques. La méthode Smoothed Particle Hydrodynamics, les méthodes Vortex [45] ou les méthodes Lattice Boltzmann [31] font partie de cette famille de méthodes lagrangiennes.

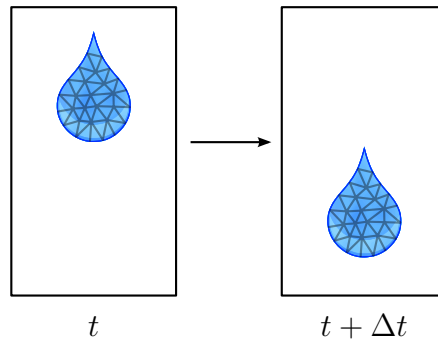


FIGURE 1.2 – Description lagrangienne d'un écoulement.

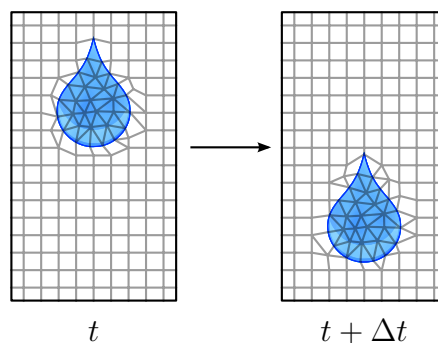


FIGURE 1.3 – Description ALE d'un écoulement.

1.1.3 Méthodes arbitrairement Lagrange-Euler (ALE)

En description Arbitrairement Lagrange-Euler (ALE), l'*observateur* se déplace à une vitesse propre dans le référentiel de l'écoulement. Cette vitesse est arbitraire et peut varier selon les éléments de contrôle. Cela dans le but d'améliorer la précision du modèle en évitant les distorsions de maillages ou pour permettre la prise en compte de géométries complexes sans réinitialisation globale du maillage (figure 1.3). Plusieurs méthodes proposent ce type d'approche, c'est le cas de la méthode Volumes Finis, ou bien encore en méthode particulière avec ALE-SPH [166] et la méthode des Volumes Finis Particulaires (FVPM) [171].

1.2 Discrétisation du domaine de calcul

Au delà de la forme choisie pour les lois de conservation modélisées, l'implémentation de ces équations sur la plate-forme de calcul influence directement les résultats obtenus. D'une part en termes de performances, la façon dont un algorithme est traduit en langage de programmation a un impact direct sur sa rapidité d'exécution. D'autre part, en termes de convergence et stabilité numérique, le mode de discrétisation du fluide est déterminant sur la précision elle-même et sur



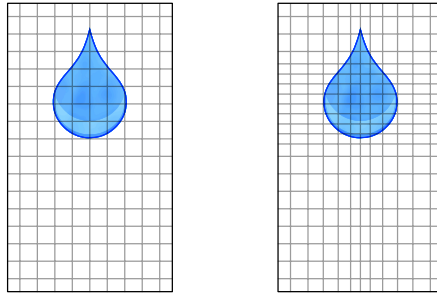


FIGURE 1.4 – Maillage cartésien, uniforme (à gauche) ou non (à droite).

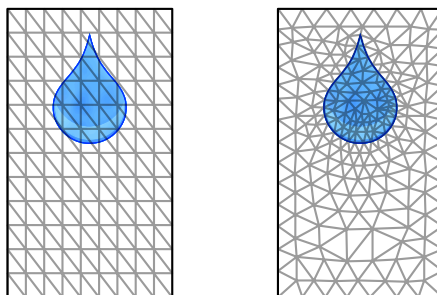


FIGURE 1.5 – Maillage structuré (à gauche) et non-structuré (à droite).

l'efficacité du code de calcul. En effet, selon l'existence d'un maillage cartésien ou non, structuré ou non-structuré, la modélisation ne va pas être la même et toutes les discrétisations n'entraînent pas la même complexité numérique. Bien sûr, les techniques de programmation ont évolué en fonction des besoins de la simulation numérique. Ainsi, le choix de modélisation va directement influencer le choix du langage et pour un mode de discrétisation, il y aura différentes façons de gérer la mémoire du calculateur selon ce que le développeur privilégiera : aisance à la manipulation ou optimisation de l'espace mémoire.

Les méthodes numériques présentent donc toutes des difficultés d'implémentation liées à la discrétisation. Une classification des discrétisations spatiales est la suivante (les méthodes spectrales et bien d'autres ont été omises).

1.2.1 Méthodes maillées

Les méthodes maillées sont basées sur une représentation du domaine de calcul par des points ou nœuds, reliés par des arrêtes. Selon le nombre de dimensions physiques, ces points vont délimiter des éléments de calculs aux formes initialement simples (carré, triangle) et qui tendent à se complexifier (quadrilatère, hexagone) en deux dimensions, et tétraèdres ou hexaèdres en trois di-

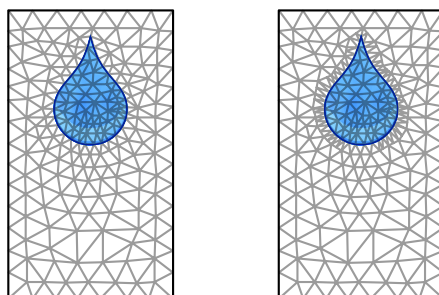


FIGURE 1.6 – Maillage conforme (à gauche) et non-conforme (à droite, certaines mailles triangulaires ont un nombre de mailles voisines supérieur à 3).

mensions. Le maillage peut être cartésien uniforme ou non (figure 1.4) (différences finies, volumes de fluides), non-structuré (figure 1.5) (éléments finis, volumes finis), ou encore non-conforme (figure 1.6)) (Galerkin discontinu).

Ainsi un maillage de tétraèdres ne va pas présenter les mêmes facilités de manipulation qu'un maillage cartésien uniforme. A l'inverse, un maillage cartésien ne peut pas être adapté à une forme de domaine de calcul complexe et limite par conséquent sévèrement les extensions possibles de la méthode. C'est pourquoi les méthodes basées sur des maillages non-structurés, non-uniformes ont remporté un vif succès dans le domaine de la modélisation numérique, employés conjointement avec des méthodes de type éléments finis ou volumes finis (code commerciaux tels que Fluent ou libres tels que OpenFOAM).

Outre la possibilité de prendre en compte des géométries complexes, il est souvent utile de pouvoir modéliser des frontières mobiles, ces méthodes nécessitent alors l'emploi d'un remaillage du domaine de calcul afin de tenir compte de sa déformation. Cette procédure peut être réalisée en employant des logiciels prévus à cet effet (Gambit, ICEM) mais la procédure est souvent lourde et déterminante dans le choix de la méthode de calcul. Il est également possible, sur des configurations simples, de conserver le même maillage et de seulement modifier les coordonnées des points pour tenir compte de la déformation du domaine, c'est souvent le cas en éléments finis appliqués à des problèmes de Mécanique du Solide (déformation de structure) ou des Fluides (écoulements confinés) (figure 1.7). Dans les cas extrêmes, le maillage peut finalement présenter de telles distorsions, dues à la déformation du domaine de calcul, que celui-ci perd en précision ou même en stabilité numérique.

1.2.2 Méthodes sans-maillage

C'est dans les configurations où le domaine change de forme, où une interface doit être suivie, que les méthodes sans maillage présentent tous leurs avantages. En effet, des méthodes comme Element Free Galerkin (EFG) [14], Diffuse Element Method (DEM) [168] ou les méthodes parti-



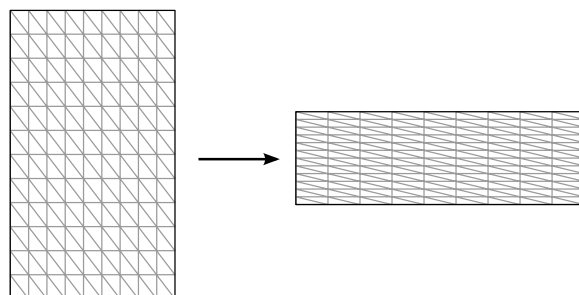


FIGURE 1.7 – Maillage déformé par la prise en compte de frontières mobiles

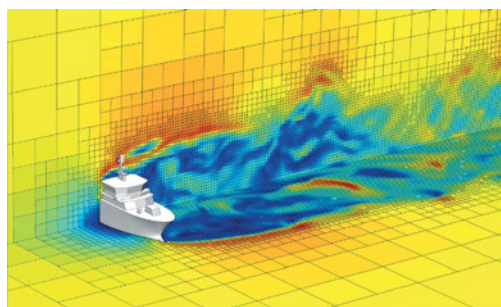


FIGURE 1.8 – Raffinement automatique de maillage sur le code GERRIS.

culaires (Vortex, Volumes Finis Particulaires, Smoothed Particle Hydrodynamics) sont très adaptées à la prise en compte de ces problèmes en raison de l'absence de règles pour la discrétisation aussi contraignante que la mise en place de points, arrêtes, facettes, suivant la géométrie ou les interfaces.

Il existe finalement assez peu de méthodes non-maillées qui ne soient pas particulière. Les seules qui n'intègrent pas le mot *particule* dans leur nom sont HP-Cloud [62], EFG [14] et DEM [168], toutes basées sur l'utilisation de moindres carrés pour l'approximation.

Une revue intéressante des méthodes dites *meshless* [13] propose une comparaison des différentes techniques d'approximation disponibles en l'absence de maillage. Une liste également très variée d'articles sur les méthodes sans maillage est proposée sur la *toile* [249].

1.2.3 Méthodes adaptatives

Les méthodes adaptatives, ou plutôt à raffinement automatique de maillage proposent une façon intelligente d'améliorer la discrétisation du calcul. En effet, le maillage peut être rendu plus fin dans les zones nécessitant des tailles de mailles plus petites (figure 1.8). A l'inverse, ces méthodes évitent une diminution injustifiée des tailles de mailles pour les zones du domaine de calcul où ce n'est pas nécessaire. Cette approche offre plusieurs avantages : le raffinement peut être effectué



(a) Définition des flux en Volumes Finis

(b) Définition des flux en Volumes Finis Particulaires

FIGURE 1.9 – Analogie entre Volumes Finis et Volumes Finis Particulaires.

sans intervention du modélisateur, qui aura juste préalablement défini ses critères de raffinement (gradient de pression, proximité d’une interface...). Le gain de temps et de mémoire permis par le raffinement spécifique des zones demandant de la précision peut accélérer les calculs de façon significative. En effet, pour une précision équivalente, moins d’éléments de calcul sont nécessaires. Par ailleurs, un corollaire à cette dernière remarque est que le nombre d’éléments dans les zones sensibles peut être augmenté et pour un temps de calcul comparable, on obtiendra une meilleure précision de calcul dans les zones sensibles.

Ainsi, un logiciel intéressant qui utilise ce type de modèle est disponible sous licence publique (GPL) : le code **GERRIS** [186, 187] [246].

1.2.4 Méthodes particulières

En méthode particulière, l’élément de contrôle où sont connues et évaluées les grandeurs physiques du système d’équations à résoudre, est la particule. Souvent initialisées sur une grille cartésienne (donc de forme carrée) puis représentée graphiquement par des disques ou des sphères selon le nombre de dimensions, la forme géométrique de ces particules n’est généralement pas connue. Elles ne sont souvent représentées numériquement que par leur centre géométrique et leur volume. Les méthodes les plus couramment employées sont les méthodes Vortex [45], les méthodes SPH, les méthodes FVPM (Volumes Finis Particulaires) ou encore les méthodes RKPM (Reproducing Kernel Particle Method) [135] basées sur une partition de l’unité pour déterminer l’influence des particules avoisinantes.

Ces méthodes sont plus récentes, et offrent une vision assez intuitive de la mécanique modélisée. Parfois elles s’appuient sur des techniques issues des méthodes maillées et adaptent ce formalisme aux particules. C’est le cas de la méthode FVPM [94, 108, 171] qui représente les variables conservatives sous la forme d’un vecteur et impose le respect des lois de conservation par l’intermédiaire de flux entre les volumes. Ces flux, initialement calculés en Volumes Finis en



fonction de la taille de facettes entre chaque élément de maillage, sont ici calculés en fonction de la distance et du rayon d'action de chaque particule (figure 1.9).

Comme mentionné précédemment, peu de méthodes non-maillées ne sont pas particulières. On peut aussi remarquer que la plupart des méthodes particulières s'appuient sur un maillage, que ce soit lors d'une étape bien spécifique du calcul, comme c'est le cas en méthode Particle In Cell (PIC) [89] où les gradients de pression sont calculés sur une grille puis appliqués aux particules qui sont ensuite advectées. C'est aussi le cas en méthode SPH [26, 29, 27] ou Vortex [45] lorsque la technique de remaillage est employée. Même si la grille en question n'apparaît pas explicitement, elle est parfois sous-jacente et permet l'application d'algorithmes d'optimisation à l'efficacité significative, c'est le cas de la technique *List Search* employée pour la recherche d'interactions et développée plus loin dans ce manuscrit (chapitre 8).

La méthode SPH est donc souvent présentée comme une méthode particulière pure. Apparue en 1977, elle est d'abord appliquée à la modélisation de corps gravitaires. L'approche particulière se montre particulièrement adaptée à ce type de modélisation du fait qu'une particule suffit à représenter un corps céleste. Gingold et Monaghan [78] et Lucy [138] sont les premiers à publier un article sur cette méthode, alors très proche de la méthode de Monte-Carlo [149].

L'écriture de la méthode a ensuite évolué pour se rapprocher d'un modèle d'interactions entre ces corps gravitaires. C'est ainsi que son utilisation s'est répandue à travers la communauté des astrophysiciens. Par exemple, la méthode a été employée avec succès pour la simulation de formation d'étoiles et de galaxies [128] ou de collision de corps célestes [21]. Encore aujourd'hui, son développement se montre particulièrement actif, tant au niveau théorique que technique.

Par son extension à l'hydrodynamique à surface libre [155], SPH a connu un développement parallèle par les mécaniciens des fluides, différent de la méthode initiale et motivé par des besoins et problématiques différents. La vision des particules, comme étant des petits éléments de liquide ou de gaz désordonnés, est à l'origine de son adaptation aux problèmes d'hydrodynamique.

SPH a également été appliquée à de nombreux autres problèmes, dominés par des phénomènes non-linéaires et de grandes déformations du milieu, où l'absence de maillage est un atout : les écoulements de matériaux élastiques, les écoulements diphasiques, les écoulements de fluides incompressibles visqueux [164], les écoulements en milieux poreux, les transferts de chaleur, les simulations d'impacts ou sur la simulation d'explosions [132]. L'utilisation de SPH peut aussi être rencontrée pour des études aussi diverses que le découpage de matériaux solides, le moulage de pièces métalliques, la diffusion d'aérosols ou encore la dynamique moléculaire.

Par l'intermédiaire de tous ces travaux, la mise en œuvre de SPH a évolué. De nombreuses améliorations ont été apportées, dont certaines seront développées dans ce manuscrit. Beaucoup de travaux ont discuté de la stabilité de la méthode et ses propriétés de convergence, avec de nombreux essais réalisés sur des cas d'écoulements modèles. Les résultats obtenus sont parfois discutables ; ce qui explique pourquoi SPH est encore le sujet de nombreuses études portant notamment sur la prise en compte des conditions limites, les écoulements à surface libre et la capacité de SPH a

fournir des résultats quantitatifs précis.

Quelques revues regroupent déjà les améliorations du modèle SPH les plus marquantes ayant été développées au cours de ces 30 dernières années : Monaghan [154], Liu et Liu [132], Monaghan [158]. Quelques comparaisons entre SPH et d'autres méthodes mettent en évidence l'atout important que peut être la discrétisation particulière pour le suivi d'interfaces et les simulations en domaine ouvert [1]. La convergence de la méthode SPH est d'ailleurs souvent contestée, en raison des nombreux paramètres employés et des instabilités souvent remarquées, si les corrections adéquates ne sont pas appliquées. Mais quelques études tendent à prouver que l'approche SPH propose des schémas convergents : [57, 58, 166, 117].

Plusieurs codes SPH ont fait leur apparition : d'abord **Gadget-2** [245] (libre) en Astrophysique (figure 1.10), puis **SPH-flow** [254] (propriétaire), **SPHysics** (libre) [256] et **ISPH** (libre) [248] en Hydrodynamique.



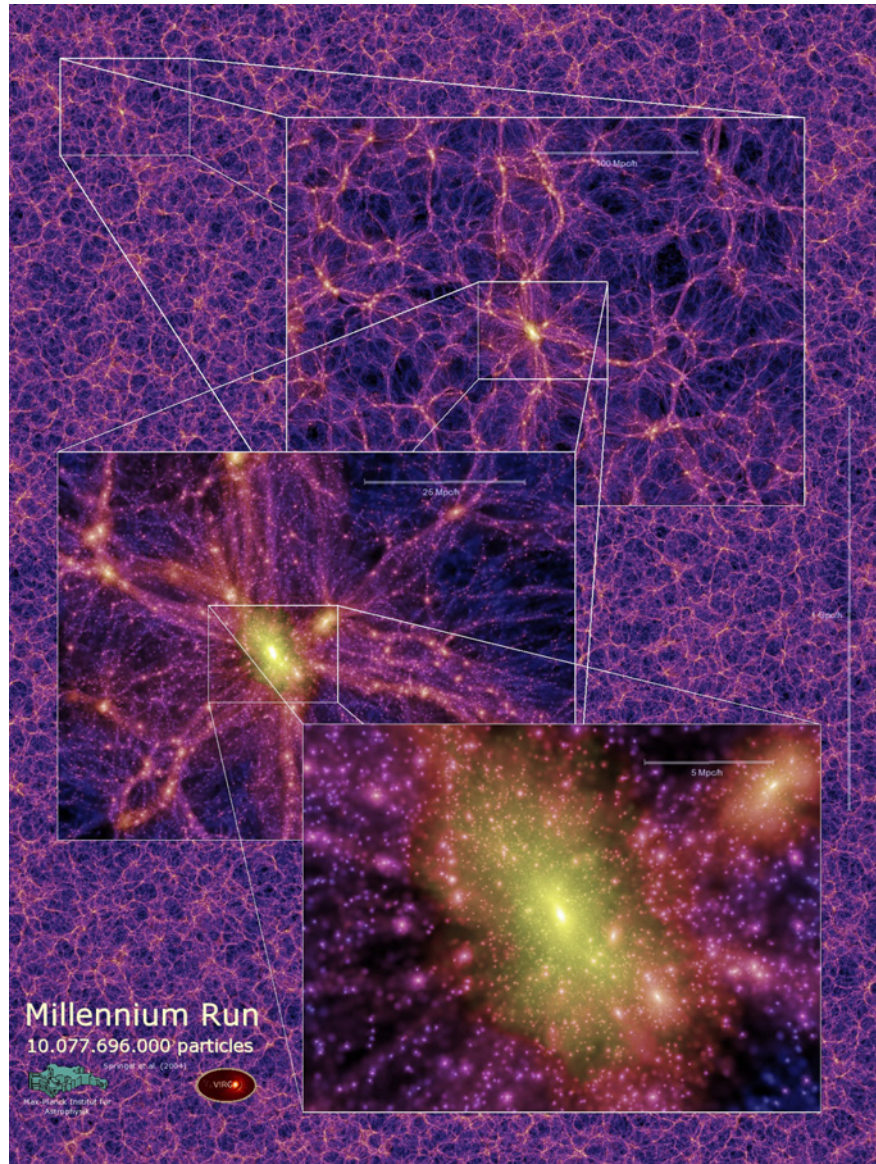


FIGURE 1.10 – Simulation de corps gravitaires avec **Gadget-2**. Chaque particule représente un corps céleste, 10 milliards de particules ont été employées.

Chapitre 2

Méthodes particulières - Outils Mathématiques

La méthode SPH fait appel à un certain nombre de notions mathématiques qu'il s'agit de décrire maintenant. Ces outils ne sont pas spécifiques à la méthode SPH. Ils sont rencontrés en réalité dans la plupart des méthodes particulières. Ils sont même issus généralement de thématiques de recherche très éloignées de la Mécanique des Fluides comme le traitement de l'image [200, 201]. Le concept d'*opérateur intégral* est essentiel à la méthode SPH, il permet de déterminer la valeur d'opérateurs différentiels appliqués à un champ donné. Les valeurs dans ce champ étant connues uniquement au niveau des particules, l'opérateur intégral permet, comme un schéma en différences finies, d'en déduire le gradient ou le laplacien par exemple. Ce chapitre passe donc en revue les notations et définitions nécessaires à la manipulation de ces opérateurs, d'abord appliqués à l'interpolation, puis à l'approximation de dérivées.

2.1 Notations et définitions

2.1.1 Approximation et discrétisation

Nous nous situons dans le cadre de la simulation numérique en Mécanique des Fluides, par les méthodes particulières. Nous cherchons à observer l'évolution spatio-temporelle de grandeurs physiques caractéristiques de l'écoulement. Toute méthode numérique nécessite une représentation du domaine de calcul en espace et en temps. Dans le cas présent, la description du domaine de calcul est lagrangienne. Chaque élément de matière i est modélisé par une particule de position \mathbf{x}_i , exprimée dans un repère cartésien, et de volume V_i . Un champ $f()$ sur le domaine est donc représenté par un nombre fini de valeurs au niveau de chaque particule : $f_i = f(\mathbf{x}_i)$. Ce champ a alors une expression analytique sur l'ensemble de domaine, appelée représentation intégrale du



champ $f()$:

$$f(\mathbf{x}) = \sum_{i=1}^N V_i f_i \eta_\epsilon^0(\mathbf{x} - \mathbf{x}_i) \quad (2.1)$$

où η_ϵ^0 est un noyau de régularisation qui a la propriété de tendre vers la mesure de Dirac quand ϵ , son pas d'interpolation, tend vers zero. C'est ici que l'interpolation en méthode particulière prend son sens, et plus particulièrement, l'interpolation par produit de convolution, déjà employée en traitement de données par Schoenberg [200, 201] en 1946. La formule (2.1) peut-être vue comme une moyenne pondérée des valeurs voisines, pour lesquelles les coefficients dépendent directement du choix du noyau d'interpolation η_ϵ^0 . L'équation (2.1) vient du fait qu'une approximation d'une fonction $f()$, supposée analytique, peut s'écrire sous la forme d'un produit de convolution entre le champ et la mesure de Dirac $\delta_0()$ [46, 112] :

$$f(\mathbf{x}) = \int_{\Omega} f(\mathbf{y}) \delta_0(\mathbf{y} - \mathbf{x}) d\mathbf{y} \quad (2.2)$$

Le passage au cas discret se fait en substituant à $\delta_0()$ un noyau de régularisation η_ϵ^0 . On retrouve alors l'expression du champ $f()$ sur le domaine de calcul discrétisé par les particules (équation (2.1)). Les calculs effectués dans ce manuscrit ont été généralement réalisés sur un ensemble de particules, initialement équidistantes, car placées sur une grille cartésienne. Ainsi on définit l'espacement inter-particule h , comme la distance séparant deux particules consécutives du domaine.

2.1.2 Opérateur différentiel

Dans la suite de ce manuscrit, nous allons voir qu'il est possible d'extraire la dérivée d'un champ connu sur un ensemble de particules. Le noyau d'interpolation doit alors respecter un certain nombre de conditions en rapport avec l'ordre de dérivation souhaité. Nous allons donc introduire les notations nécessaires à la manipulation d'opérateurs différentiels. Soit D^β , un opérateur différentiel :

$$D^\beta = \frac{\partial^{|\beta|}}{\partial x_1^{\beta_1} \partial x_2^{\beta_2} \dots \partial x_l^{\beta_l}} \quad (2.3)$$

$\beta = (\beta_1, \beta_2, \dots, \beta_l)$ est un multi-indice pour lequel on a :

$$|\beta| = \beta_1 + \beta_2 + \dots + \beta_l \quad (2.4)$$

$$\mathbf{x}^\beta = x_1^{\beta_1} x_2^{\beta_2} \dots x_l^{\beta_l} \quad (2.5)$$

et

$$\beta! = \beta_1! \beta_2! \dots \beta_l! \quad (2.6)$$

où l représente la dimension de l'espace.

Ainsi, un gradient en espace de dimension 2 sera noté :

$$\nabla = \begin{pmatrix} D^{(1,0)} \\ D^{(0,1)} \end{pmatrix} \quad (2.7)$$

L'opérateur laplacien en dimension 3 s'écrira :

$$\Delta = D^{(2,0,0)} + D^{(0,2,0)} + D^{(0,0,2)} \quad (2.8)$$

2.1.3 Opérateur d'approximation

Nous noterons $\mathcal{L}^0 f()$ l'approximation particulière de la fonction $f()$ (équation (2.1)). L'opérateur construit pour l'approximation de $D^\beta f()$ sera noté $\mathcal{L}^\beta f()$.

$$\mathcal{L}^0 f(\mathbf{x}) = \int_{\Omega} f(\mathbf{y}) \delta_0(\mathbf{x} - \mathbf{y}) d\mathbf{y} \approx \sum_{i=1}^N V_i f_i \eta_\epsilon^0(\mathbf{x} - \mathbf{x}_i) \quad (2.9)$$

2.1.4 Noyau d'interpolation

Un noyau d'interpolation est la fonction que l'on substitue à la mesure de Dirac pour construire la représentation intégrale du champ $f()$. Un noyau d'interpolation pure, utilisé au sein de l'opérateur \mathcal{L}^0 , sera noté η^0 . De même, un noyau ayant les propriétés voulues pour l'approximation \mathcal{L}^β de l'opérateur différentiel D^β sera noté η^β . Il existe une grande variété de noyaux η dans la littérature. Une liste de fonctions couramment utilisées sera donnée dans la suite de ce rapport. On note η_ϵ , la mise à l'échelle du noyau η :

$$\eta_\epsilon^\beta(\mathbf{x}) = \frac{1}{\epsilon^l} \eta^\beta\left(\frac{\mathbf{x}}{\epsilon}\right), \quad (2.10)$$

avec l la dimension de l'espace. Le paramètre ϵ est le pas d'interpolation, qui définit la taille du support de régularisation (figure 2.1). La valeur donnée à ϵ varie selon la fonction η employée. On introduit alors le rapport $\kappa = \frac{\epsilon}{h}$ reliant le pas d'interpolation et le pas de discrétisation. Ce paramètre permet de contrôler le nombre de particules à prendre en compte dans l'approximation.

Enfin, on définit le moment d'ordre α d'un noyau d'interpolation de la manière suivante :

$$M_\alpha = \int_{\Omega} \mathbf{x}^\alpha \eta^\beta(\mathbf{x}) d\mathbf{x} \quad (2.11)$$



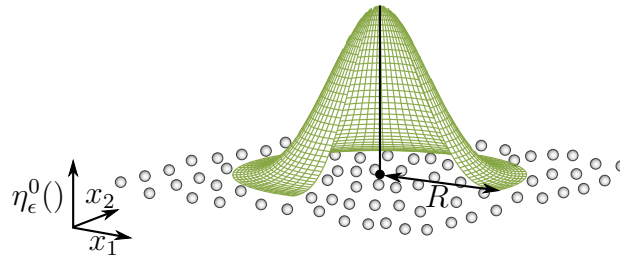


FIGURE 2.1 – Interpolation sur un ensemble de particules. Le rayon R du noyau $\eta_\epsilon^0()$ dépend de ϵ

2.1.5 Régularité d'une fonction à interpoler

Une notion importante, pour le développement des méthodes d'approximation présentées dans ce manuscrit, est la régularité de la fonction à interpoler. On dit qu'une fonction $f()$ est régulière d'ordre k , ou encore qu'elle est continue de classe C^k , si elle est k fois différentiable et que ses k premières dérivées sont continues. Cette hypothèse est normalement requise afin de pouvoir écrire les formes discrètes d'approximation intégrale. Il est clair que dans les applications envisagées, les champs interpolés ne sont pas toujours suffisamment réguliers et que dans ce cas, l'approximation intégrale ne peut pas être écrite. Nous allons donc supposer que tous les champs traités au cours de ce manuscrit sont suffisamment réguliers.

2.2 Interpolation

SPH s'appuie sur une représentation lagrangienne du fluide, et sur la méthode d'*approximation intégrale* (équation (2.1)), pour l'évaluation des champs associés aux particules. L'approximation intégrale est une forme d'interpolation déjà employée en méthodes *Vortex* pour le remaillage [45] ou en traitement de données [200, 201]. L'interpolation de grandeurs en un point à partir des données connues en d'autres points avoisinants peut être réalisée par de nombreuses méthodes, l'approximation intégrale par noyau de convolution s'inscrit dans le cadre des méthodes d'interpolation par *lissage* (*smoothing interpolation* en anglais), par opposition à l'interpolation ordinaire. Elle offre l'avantage de permettre la reconstruction d'un champ à partir de données connues en des points arbitrairement répartis sur le domaine (ici, les particules). L'opérateur intégral discret (équation (2.1)) s'appuie sur un certain nombre d'hypothèses que nous décrivons maintenant.

2.2.1 Opérateur intégral

Nous nous intéressons ici à la construction de l'opérateur d'approximation SPH, concept essentiel de la méthode. On cherche une expression du champ $f()$ en tout point \mathbf{x} à partir des valeurs

connues au niveau d'autres points \mathbf{y} . On part donc de l'équation 2.2, où on remplace la mesure de Dirac par le noyau d'interpolation $\eta_\epsilon^0(\cdot)$. On a alors une expression de l'opérateur intégral \mathcal{L}^0 , appliqué à $f(\cdot)$:

$$\mathcal{L}^0 f(\mathbf{x}) = \int_{\Omega} f(\mathbf{y}) \eta_\epsilon^0(\mathbf{y} - \mathbf{x}) d\mathbf{y} \quad (2.12)$$

Dans la littérature SPH (Liu et Liu [132], par exemple), l'approximation précédente est souvent notée $\langle f(\mathbf{x}) \rangle$. En supposant $f(\cdot)$ suffisamment régulière, on peut écrire son développement en série de Taylor en \mathbf{y} à partir du point \mathbf{x} :

$$f(\mathbf{y}) = \sum_{|k|=0}^{\infty} \frac{D^k f(\mathbf{x})}{k!} (\mathbf{y} - \mathbf{x})^k \quad (2.13)$$

En reprenant ce développement jusqu'à l'ordre 2, on a :

$$f(\mathbf{y}) = f(\mathbf{x}) + \sum_{|k|=1} D^k f(\mathbf{x}) (\mathbf{y} - \mathbf{x}) + \mathcal{O}((\mathbf{y} - \mathbf{x})^2) \quad (2.14)$$

où $\mathcal{O}((\mathbf{y} - \mathbf{x})^2)$ représente l'erreur commise en ignorant les termes d'ordre égal et supérieur. Si on remplace $f(\mathbf{y})$ dans (2.12) par l'expression (2.14), on obtient :

$$\mathcal{L}^0 f(\mathbf{x}) = \int_{\Omega} \left(f(\mathbf{x}) + \sum_{|k|=1} D^k f(\mathbf{x}) (\mathbf{y} - \mathbf{x}) + \mathcal{O}((\mathbf{y} - \mathbf{x})^2) \right) \eta_\epsilon^0(\mathbf{y} - \mathbf{x}) d\mathbf{y} \quad (2.15)$$

$$= f(\mathbf{x}) \underbrace{\int_{\Omega} \eta_\epsilon^0(\mathbf{y} - \mathbf{x}) d\mathbf{y}}_{M_0} + \sum_{|k|=1} D^k f(\mathbf{x}) \underbrace{\int_{\Omega} (\mathbf{y} - \mathbf{x}) \eta_\epsilon^0(\mathbf{y} - \mathbf{x}) d\mathbf{y}}_{M_1} + \mathcal{O}(\epsilon^2) \quad (2.16)$$

Cette dernière équation nous montre que l'approximation $\mathcal{L}^0 f(\mathbf{x})$ s'écrit comme une combinaison linéaire de la concentration $f(\cdot)$ et de ses dérivées, avec pour coefficients, les moments d'ordre 0 et 1 du noyau d'interpolation. Si ces moments prennent pour valeurs

$$M_0 = \int_{\Omega} \eta_\epsilon^0(\mathbf{x}) d\mathbf{x} = 1 \quad (2.17)$$

et

$$M_1 = \int_{\Omega} \mathbf{x} \eta_\epsilon^0(\mathbf{x}) d\mathbf{x} = 0, \quad (2.18)$$

alors l'approximation peut s'écrire :

$$\mathcal{L}^0 f(\mathbf{x}) = f(\mathbf{x}) + \mathcal{O}(\epsilon^2) \quad (2.19)$$

Ainsi, le noyau η_ϵ^0 utilisé doit satisfaire les conditions (2.17) et (2.18) afin de permettre une



approximation d'ordre 2. La condition (2.17) sur le moment d'ordre 0, également appelée condition de normalisation, impose la valeur de l'intégrale du noyau sur l'ensemble de son support. La seconde condition (2.18) sur le moment d'ordre 1, signifie que le noyau doit être à symétrie radiale. Pour augmenter l'ordre d'approximation, il faut prendre en compte plus de termes du développement de Taylor (équation (2.14)). Ainsi, pour une approximation d'ordre 4, il va falloir satisfaire 4 conditions sur les moments ($M_0 = 1$, $M_1 = 0$, $M_2 = 0$ et $M_3 = 0$, voir équation (2.11)).

2.2.2 Choix du noyau d'interpolation

L'approximation par opérateur intégral est une notion courante en méthodes particulières. Il est alors fréquent de rencontrer les mêmes types de noyaux d'interpolation dans différentes méthodes : SPH, Vortex [45]. Les choix du noyau et de la distance de régularisation ϵ sont très importants, à la fois pour la qualité et la rapidité de la simulation.

Formes de noyaux

Il existe plusieurs *formes* de noyaux dans la littérature. Une liste non-exhaustive de fonctions est présentée dans le tableau 2.1 et la figure 2.2. Les noyaux sont exprimés de la manière suivante :

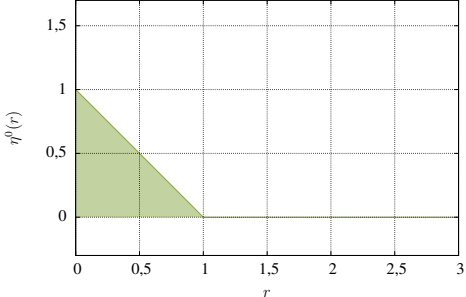
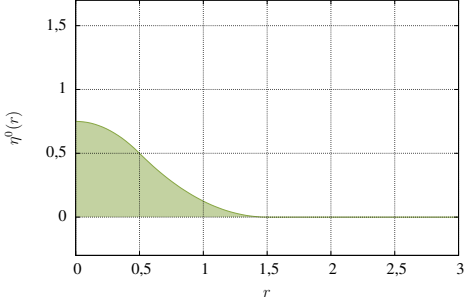
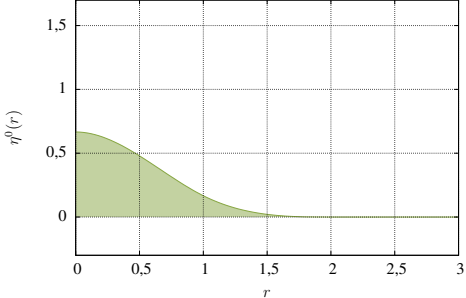
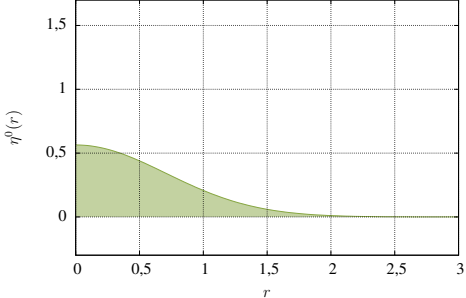
$$\eta_\epsilon^0(\mathbf{x}) = \frac{1}{\epsilon^l} \eta^0\left(\frac{\mathbf{x}}{\epsilon}\right) \quad (2.20)$$

La fonction $\eta^0(\cdot)$ fait généralement intervenir une constante α_l dépendant de l , le nombre de dimension du problème. L'ordre d'approximation présenté correspond aux conditions vérifiées sur les moments du noyau dans le cas d'une interpolation sur un champ de particules uniformément réparties sur une grille cartésienne. Il est à noter que cet ordre d'approximation n'est plus valable dès qu'elles ont abandonné cette répartition, d'où l'usage courant de techniques de (re)normalisation, présentées plus loin, afin de restaurer ces conditions.

Critères de choix d'un noyau

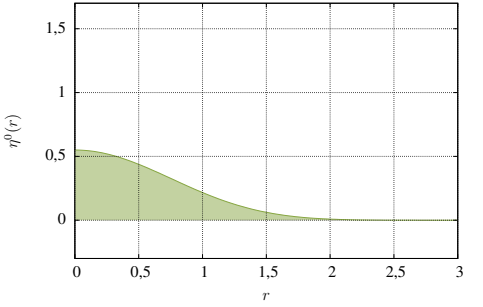
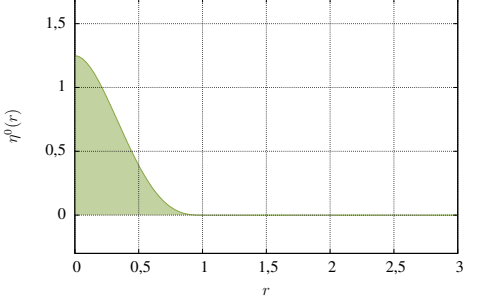
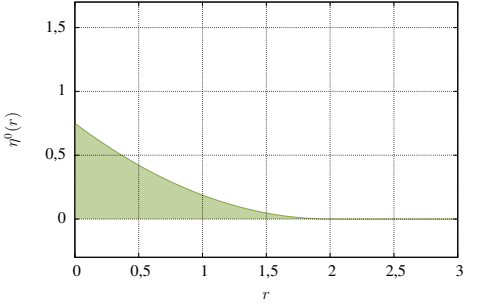
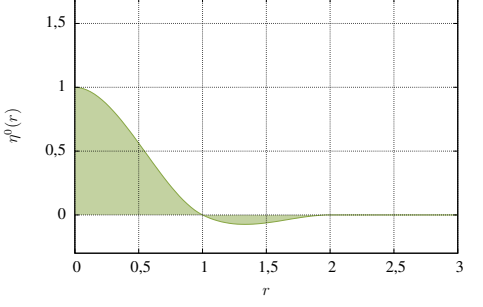
Plusieurs auteurs se sont intéressés au choix du noyau selon différents critères de *qualité*. Ainsi, Fulk et Quinn [77] proposent une analyse de 20 fonctions d'interpolation appliquées à SPH en 1D. Leur étude est basée sur le calcul de l'erreur d'interpolation effectuée lors de la mise en œuvre de l'opérateur SPH. Cette erreur se divise en deux parties :

- Une erreur d'approximation, liée à l'utilisation de l'opérateur intégral et du noyau d'interpolation. On développe effectivement le champ $f(\cdot)$ en série de Taylor (équation (2.13)), puis on ignore les termes d'ordre supérieur ou égal à 2.
- Une erreur liée à la discrétisation du domaine de calcul, qui peut être grande si la répartition des particules n'est pas suffisamment homogène.

Noyau (Ordre) Référence	$\eta(r)$	Représentation graphique en dimension 1
M_2 (1) [45]	$\begin{cases} 1-r & 0 \leq r < 1 \\ 0 & r \geq 1 \end{cases}$ <p>Le noyau en dimension 2 ou 3 s'obtient par produit des noyaux calculés dans chaque dimension</p>	
M_3 (2) [45]	$\begin{cases} \frac{1}{2} \left(r - \frac{3}{2}\right)^2 - \frac{3}{2} \left(r + \frac{1}{2}\right)^2 & 0 \leq r < \frac{1}{2} \\ \frac{1}{2} \left(\frac{3}{2} - r\right)^2 & \frac{1}{2} \leq r < \frac{3}{2} \\ 0 & r \geq \frac{3}{2} \end{cases}$ <p>Le noyau en dimension 2 ou 3 s'obtient par produit des noyaux calculés dans chaque dimension</p>	
M_4 (2) [152]	$\alpha_d \begin{cases} \frac{2}{3} - r^2 + \frac{r^3}{2} & 0 \leq r < 1 \\ \frac{1}{6} (2-r)^3 & 1 \leq r < 2 \\ 0 & r \geq 2 \end{cases}$ <p>$\alpha_1 = 1$ $\alpha_2 = \frac{15}{7\pi}$ $\alpha_3 = \frac{3}{2\pi}$</p>	
Gaussien (2) [78]	$\alpha_d e^{-r^2}$ <p>$\alpha_1 = \frac{1}{\sqrt{\pi}}$ $\alpha_2 = \frac{1}{\pi}$ $\alpha_3 = \pi^{-\frac{3}{2}}$</p>	

Suite page suivante



Noyau (Ordre) Référence	$\eta(r)$	Représentation graphique en dimension 1
Q-Splines (2) [162]	$\alpha_d \begin{cases} (3-r)^5 - 6(2-r)^5 + 15(1-r)^5 & 0 \leq r < 1 \\ (3-r)^5 - 6(2-r)^5 & 1 \leq r < 2 \\ (3-r)^5 & 2 \leq r < 3 \\ 0 & r \geq 3 \end{cases}$ $\alpha_1 = \frac{1}{120} \quad \alpha_2 = \frac{7}{478\pi} \quad \alpha_3 = \frac{3}{359\pi}$	
Lucy (2) [138]	$\alpha_d \begin{cases} (1+3r)(1-r)^3 & 0 \leq r < 1 \\ 0 & r \geq 1 \end{cases}$ $\alpha_1 = \frac{5}{4} \quad \alpha_2 = \frac{5}{\pi} \quad \alpha_3 = \frac{105}{16\pi}$	
Quadratique (2) [105]	$\alpha_d \begin{cases} \frac{3}{16}r^2 - \frac{3}{4}r + \frac{3}{4} & 0 \leq r < 2 \\ 0 & r \geq 2 \end{cases}$ $\alpha_1 = 1 \quad \alpha_2 = \frac{2}{\pi} \quad \alpha_3 = \frac{5}{4\pi}$	
M'_4 (3) [152]	$\begin{cases} 1 - \frac{5r^2}{2} + \frac{3r^3}{2} & 0 \leq r < 1 \\ \frac{1}{2}(2-r)^2(1-r) & 1 \leq r < 2 \\ 0 & r \geq 2 \end{cases}$ Le noyau en dimension 2 ou 3 s'obtient par produit des noyaux calculés dans chaque dimension	

Suite page suivante

Noyau (Ordre) Référence	$\eta(r)$			Représentation graphique en dimension 1
Gaussien (4)	$\alpha_d(r)e^{-r^2}$			
[45]	$\alpha_1(r) = \frac{3/2-r^2}{\sqrt{\pi}}$	$\alpha_2(r) = \frac{2-r^2}{\pi}$	$\alpha_3(r) = \frac{5/2-r^2}{\pi^{3/2}}$	

TABLE 2.1 – Noyaux d'interpolation

En additionnant les termes du développement de Taylor d'ordre supérieur ou égale à 2 (équation (2.13)), puis en majorant cette somme, on obtient une grandeur différente pour chaque noyau. Cette grandeur doit être minimale pour une approximation de qualité maximale. En comparant les valeurs calculées pour différents noyaux, on observe que les meilleurs résultats sont obtenus pour la classe de noyaux en cloche (Gaussien, M_3 , M_4 , Q-Splines ...). Cependant, cette première analyse ne tient pas compte de l'erreur de quadrature, commise quand on applique la formulation au cas discret. L'étude est donc approfondie en tenant compte, cette fois, de la discrétisation du domaine de calcul. L'erreur de quadrature dépend de la distance inter-particulaire h et de la distance de régularisation ϵ du noyau. ϵ définit le rayon du support du noyau, et par conséquent le nombre de particules à prendre en compte dans l'interpolation. La valeur de ϵ doit donc être fixée en tenant compte de l'espacement inter-particulaire h . En effet, si on augmente cette distance inter-particulaire pour une valeur de ϵ fixée, le nombre de particules présentes sur le support de $\eta()$ décroît. Il paraît donc raisonnable d'utiliser le rapport adimensionnel κ (équation (2.21)) pour mettre en relation la distance inter-particule et le rayon de régularisation. Le choix de la valeur de κ , qui dépend du noyau utilisé, a un impact non-négligeable sur la qualité de l'approximation.

$$\kappa = \frac{\epsilon}{h} \quad (2.21)$$

En dehors de la valeur donnée à κ , la qualité de l'interpolation dépend également des données interpolées. Ainsi, Fulk et Quinn [77] ont calculé l'erreur sur l'approximation de champs continus ou non, pour différentes valeurs du paramètre κ et sur différentes discrétisations du domaine de calcul (uniforme ou non). La synthèse de ces résultats mène à la conclusion suivante : les noyaux en cloche fournissent en général une approximation de meilleure qualité, sauf dans le cas d'un champ interpolé non-continu, où un noyau hyperbolique se montre légèrement plus précis au niveau des discontinuités, mais le faible gain en précision n'en justifie pas l'usage systématique. Liu et Liu [132] s'intéressent également à la construction de noyaux d'interpolation de type B-splines pour



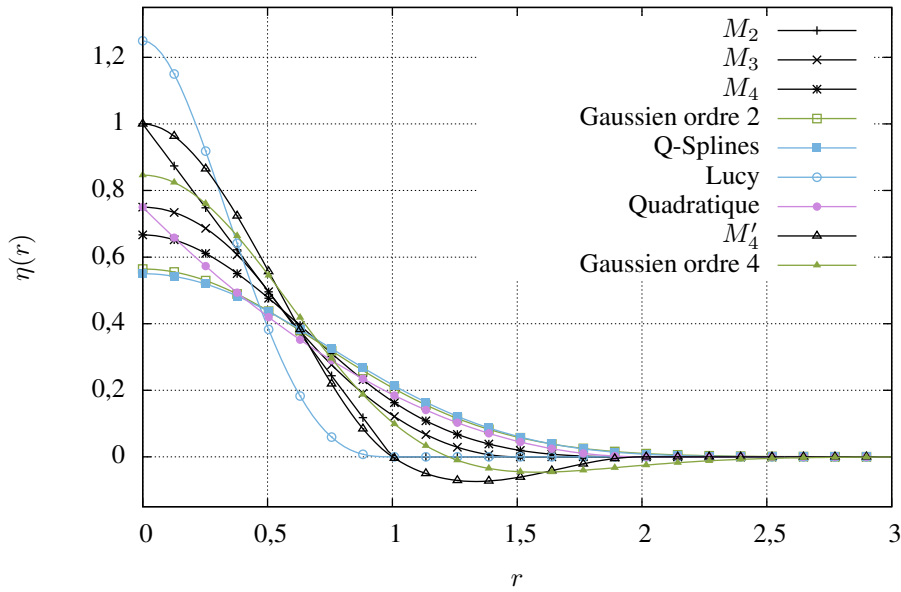


FIGURE 2.2 – Représentation graphique de l'ensemble des noyaux du tableau 2.1.

SPH. La méthode d'approximation employée étant d'ordre 2, ils fixent les valeurs des moments d'ordre 0 et 1 (équations (2.17) et (2.18)) afin de donner à leurs noyaux la forme voulue. Puis ils calculent la valeur du moment d'ordre 2, en estimant que plus elle est faible, plus l'approximation est précise. D'autres études [24, 96] s'intéressent également aux noyaux d'interpolation SPH. Plus précisément, Hongbin et Xin [96] analysent les moments d'ordre 0 et 1 calculés numériquement, tandis que Capuzzo-Dolcetta et Di Lisio [24] s'intéressent au comportement de chaque noyau si la distance de régularisation ϵ varie pour chaque particule. Les conclusions confirment celles de Fulk et Quinn [77]. Les noyaux en cloche semblent plus appropriés, en particulier les noyaux gaussiens, B-splines ou encore Q-splines. Monaghan [154] précise que si l'on souhaite donner une interprétation physique aux équations discrétisées en utilisant SPH, il faut toujours partir d'un noyau gaussien.

Temps de calcul

En dehors du comportement d'un noyau face aux variations du paramètre κ , il est important de choisir un noyau d'interpolation également en fonction des performances de calcul. Plusieurs paramètres sont à prendre en compte : complexité (nombre d'opérations nécessaires pour le calcul du noyau) et taille du support (nombre de particules prises en compte dans l'approximation). Ainsi, un noyau gaussien, plus régulier et donc plus stable qu'un noyau B-splines [96], est plus coûteux en opérations et n'a pas de support compact. Si en revanche on choisit un noyau B-splines, le temps de calcul est réduit significativement et son support compact permet l'utilisation d'algorithmes d'accélération au sein de la méthode SPH.

Amélioration de noyaux d'interpolation

Il existe différentes façons de construire des noyaux d'interpolation pour un ordre d'approximation donné. Dans tous les cas, les conditions à vérifier portent sur la valeur des moments du noyau. Chaniotis et Poulikakos [28] mettent en œuvre une technique d'approximation à l'aide de noyaux construits à partir de combinaisons linéaires de noyaux connus et de leur dérivées (extrapolation de Richardson). Quelque soit la répartition des particules, cette méthode fournit un noyau adapté permettant d'atteindre la précision souhaitée. La construction du nouveau noyau passe par la résolution d'un système linéaire de taille $n \times n$ pour une approximation d'ordre n . Pour chaque approximation, un noyau spécifique est construit, adapté à la configuration des particules voisines. Si on applique cette méthode à un ensemble de particules équidistantes, on retrouve une méthode de correction des noyaux B-splines couramment utilisée en méthode particulière [45, 152]. Il s'agit de la construction du noyau M'_4 , d'ordre 3 à partir du noyau M_4 d'ordre 2 (cf. tableau 2.1) :

$$M'_4(r) = \frac{3}{2}M_4(r) - \frac{1}{2}\frac{\partial M_4}{\partial r}(r) \quad (2.22)$$

Les noyaux d'ordres élevés sont souvent rejetés en raison du caractère non-physique des résultats fournis [132]. Cela vient du fait qu'au delà de l'ordre 2 (figure 2.2), les noyaux sont négatifs sur certains intervalles. Par exemple, près des frontières du domaine où les particules ne sont présentes que sur une partie du domaine du noyau, cela peut produire de mauvais résultats. Pour corriger ces erreurs, une solution consiste à construire des noyaux *décentrés* à utiliser dans de telles configurations [28, 65], ou encore à normaliser le noyau afin de restaurer la propriété (équation (2.17)).

2.3 Approximation de dérivées

Il existe plusieurs formes d'opérateurs-intégrales en méthode particulière. Nous nous intéressons ici aux deux formulations les plus répandues : la formulation couramment employée dans la méthode SPH [132] et la formulation *Particle Strength Exchange* (PSE) [65]. La principale différence entre les méthodes SPH et PSE vient du fait qu'on utilise, dans le premier cas, le même noyau et, dans le second cas, des noyaux spécifiques pour l'approximation du champ étudié et de ses dérivées. Le tableau 2.2 montre la forme de l'opérateur intégral utilisé dans chacune des méthodes selon le type d'approximation souhaité.

Chacune de deux formes d'opérateurs-intégrales présentées fait appel à une méthode de construction spécifique que nous allons développer maintenant.

2.3.1 Approche SPH

Si on s'intéresse au calcul de la dérivée première du champ $f()$ transporté par les particules, l'opérateur SPH se construit en remplaçant directement le champ $f()$ par son gradient $\nabla f()$ dans



Opérateur	Approximation	SPH	PSE
\mathcal{L}^0	$f(\mathbf{x})$	$\int f(\mathbf{y})\eta_\epsilon^0(\mathbf{x} - \mathbf{y})d\mathbf{y}$	\emptyset
\mathcal{L}^1	$D^1 f(\mathbf{x})$	$\int f(\mathbf{y})\nabla\eta_\epsilon^0(\mathbf{x} - \mathbf{y})d\mathbf{y}$	$\frac{1}{\epsilon} \int (f(\mathbf{y}) + f(\mathbf{x}))\eta_\epsilon^1(\mathbf{x} - \mathbf{y})d\mathbf{y}$
\mathcal{L}^2	$D^2 f(\mathbf{x})$	$\int f(\mathbf{x})\Delta\eta_\epsilon^0(\mathbf{x} - \mathbf{y})d\mathbf{y}$	$\frac{1}{\epsilon^2} \int (f(\mathbf{y}) - f(\mathbf{x}))\eta_\epsilon^2(\mathbf{x} - \mathbf{y})d\mathbf{y}$

TABLE 2.2 – Opérateurs-intégrales en méthode particulière

l'équation (2.12). Ce qui donne :

$$\mathcal{L}^1 f(\mathbf{x}) = \mathcal{L}^0 (\nabla f(\mathbf{x})) = \int (\nabla f(\mathbf{y})) \eta_\epsilon^0(\mathbf{y} - \mathbf{x})d\mathbf{y} \quad (2.23)$$

Or, on a :

$$(\nabla f(\mathbf{y}))\eta_\epsilon^0(\mathbf{y} - \mathbf{x}) = \nabla(f(\mathbf{y})\eta_\epsilon^0(\mathbf{y} - \mathbf{x})) - f(\mathbf{y})\nabla\eta_\epsilon^0(\mathbf{y} - \mathbf{x}) \quad (2.24)$$

Par intégration par parties, on peut modifier l'expression (2.23) pour obtenir :

$$\mathcal{L}^1 f(\mathbf{x}) = \underbrace{\int_{\Omega} \nabla(f(\mathbf{y})\eta_\epsilon^0(\mathbf{y} - \mathbf{x}))d\mathbf{y}}_{I_1} - \underbrace{\int_{\Omega} f(\mathbf{y})\nabla\eta_\epsilon^0(\mathbf{y} - \mathbf{x})d\mathbf{y}}_{I_2} \quad (2.25)$$

D'après le théorème de divergence de Green-Ostrogradski [132, 133, 218], l'intégrale I_1 de l'équation précédente est équivalente à une intégrale de la quantité $f()$ sur la surface $\partial\Omega$ enveloppant le domaine d'intégration Ω :

$$\mathcal{L}^1 f(\mathbf{x}) = I_S - \int_{\Omega} f(\mathbf{y})\nabla\eta_\epsilon^0(\mathbf{y} - \mathbf{x})d\mathbf{y} \quad (2.26)$$

avec

$$I_S = \int_{\partial\Omega} f(\mathbf{y})\eta_\epsilon^0(\mathbf{y} - \mathbf{x})\mathbf{n}ds = 0 \quad (2.27)$$

où \mathbf{n} est la normale unitaire à la surface $\partial\Omega$. Cette intégrale de surface est nulle si le support du noyau η_ϵ^0 est compact et s'il est totalement inclus dans le domaine de calcul. Sous ces hypothèses, on a :

$$\mathcal{L}^1 f(\mathbf{x}) = - \int_{\Omega} f(\mathbf{y})\nabla\eta_\epsilon^0(\mathbf{y} - \mathbf{x})d\mathbf{y} \quad (2.28)$$



(a) Configuration conforme à l'hypothèse (2.27). (b) Configuration non-conforme à l'hypothèse (2.27).

FIGURE 2.3 – Approximation particulière au voisinage de la frontière du domaine.

En revanche, si le support du noyau dépasse la frontière du domaine (figure 2.3), le premier terme de l'équation (2.25) n'étant pas nul, le négliger conduit à une erreur d'approximation significative. Il faut alors prendre en compte cette erreur et la corriger.

Il existe différentes manières de remédier à ce problème dans la littérature SPH [154, 132, 158]. Finalement, en exploitant la symétrie radiale du noyau :

$$\nabla_x \eta_\epsilon^0(\mathbf{y} - \mathbf{x}) = -\nabla_y \eta_\epsilon^0(\mathbf{y} - \mathbf{x}), \quad (2.29)$$

on obtient une approximation à l'ordre 2, dans de bonnes conditions discutées dans [91], de la dérivée du champ $f()$ par :

$$\mathcal{L}^1 f(\mathbf{x}) = \int_{\Omega} f(\mathbf{y}) \nabla_x \eta_\epsilon^0(\mathbf{y} - \mathbf{x}) d\mathbf{y} \quad (2.30)$$

De même, l'opérateur \mathcal{L}^2 pour l'approximation de la dérivée seconde est construit en substituant $\Delta f()$ à $f()$ dans l'équation (2.12) [132]. La formulation présentée ici est très souvent rencontrée dans les différentes variations de la méthode SPH. Il est possible d'augmenter l'ordre d'approximation en suivant la méthode exposée par Liu *et al.* [133]. Cependant, la méthode SPH utilise le même noyau d'interpolation pour les différentes formes de l'opérateur d'approximation. Ainsi, si on utilise un noyau polynomial de degré 3, sa dérivée seconde, utilisée dans l'opérateur \mathcal{L}^2 , est alors de degré 1. Dans de telles conditions, l'approximation des dérivées d'un champ $f()$ peut être très mauvaise. L'avantage de la fonction gaussienne est qu'elle ne perd pas en régularité quand on l'utilise également pour l'approximation de gradients ou laplaciens.

2.3.2 Approche PSE

La méthode présentée ici a été généralisée par Eldredge *et al.* [65] et s'inspire de la méthode PSE initialement développée par Choquin et Huberson [32] et Degond et Mas-Gallic [55], pour la



résolution de problèmes de convection-diffusion, puis appliquée aux méthodes *Vortex* en raison de ses propriétés de conservation. Cette méthode fait appel à un opérateur intégral pour approcher la valeur d'un opérateur différentiel appliqué à un champ. La version *originale* de PSE s'appliquait exclusivement à l'opérateur laplacien dans le cadre de problèmes de convection-diffusion. La version présentée ici a été généralisée au traitement de tout opérateur différentiel. Les auteurs [65] proposent également un traitement spécifique de ces opérateurs pour les particules situées près des frontières du domaine en utilisant des intégrales sur un demi-plan de manière à ne prendre en compte que la partie de l'espace contenant des particules. On définit l'opérateur \mathcal{L}^β d'approximation PSE de D^β (c.f. les notations du paragraphe 2.1.2) appliqué à f par :

$$\mathcal{L}^\beta f(\mathbf{x}, t) = \frac{1}{\epsilon^{|\beta|}} \int (f(\mathbf{y}, t) - f(\mathbf{x}, t)) \eta_\epsilon^\beta(\mathbf{x} - \mathbf{y}) d\mathbf{y} \quad (2.31)$$

Cet opérateur intégral diffère de celui présenté dans la partie précédente : l'intégrale ne fait plus intervenir la concentration elle-même, mais une différence entre les valeurs de cette concentration au niveau de la particule traitée et de toutes les autres. L'influence de deux particules données est alors réciproque et symétrique, cela assure la conservation de la grandeur traitée par la simulation. On discrétise ensuite l'intégrale sur l'ensemble du domaine occupé par les particules :

$$\mathcal{L}^\beta f_p(t) = \frac{1}{\epsilon^{|\beta|}} \sum_{q=1}^N V_q(t) (f_q(t) - f_p(t)) \eta_\epsilon^\beta(\mathbf{x}_p(t) - \mathbf{x}_q(t)) \quad (2.32)$$

Le noyau η_ϵ^β a été au préalable construit pour l'approximation de l'opérateur L^β selon la méthodologie présentée en annexe (voir Chap. A). Par un raisonnement différent, développé plus loin dans ce manuscrit (§3.2), car en rapport avec les équations traitées, [16, 232] arrivent aux mêmes équations symétrisées pour SPH.

2.4 Noyau employé

En résumé, voici toute les conditions nécessaires à la mise en œuvre d'une interpolation d'ordre 2 en méthode particulaire à l'aide de l'opérateur intégral (2.2). La première est la condition d'unicité du noyau :

$$\int_{\Omega} \eta_\epsilon^0(\mathbf{x}) d\mathbf{x} = 1 \quad (2.33)$$

Cette condition est essentielle à la cohérence de la formule d'interpolation. En effet, si on voit cette expression comme une méthode de lissage, ou de moyenne, d'un champ scalaire, il paraît évident que cette formule n'a de sens que si la somme des influences de chaque point du domaine vaut 1. La seconde est nécessaire à la convergence de l'approximation

$$\lim_{\epsilon \rightarrow 0} \eta_\epsilon^0(\mathbf{x}) = \delta_0(\mathbf{x}) \quad (2.34)$$

Une troisième condition de compacité du support de la fonction de régularisation est souvent posée, pour des raisons d'optimisation des temps de simulation :

$$\eta_\epsilon^0(\mathbf{x}) = 0 \text{ quand } |\mathbf{x}| > \delta_c, \quad (2.35)$$

où δ_c définit le rayon maximal du support du noyau. Au delà, l'influence des particules peut, sans erreur, être négligée. Dans le cas présent, le noyau utilisé est un noyau gaussien modifié proposé par Colagrossi et Landrini [41], Colagrossi [35]. En partant du noyau de base gaussien en deux dimensions, déjà utilisé par Morris [163] ou servant de noyau *générique* dans le traitement général des opérateurs différentiels en méthode particulaire proposé par Eldredge *et al.* [65] :

$$\eta_\epsilon^0(\mathbf{x}) = \frac{1}{\pi\epsilon^2} e^{-\left(\frac{|\mathbf{x}|}{\epsilon}\right)^2}, \quad (2.36)$$

il est possible de restaurer la compacité du noyau en le coupant à une distance δ_c de son centre :

$$\eta_\epsilon^0(\mathbf{x}, \delta_c) = \frac{e^{-\left(\frac{|\mathbf{x}|}{\epsilon}\right)^2} - e^{-\left(\frac{\delta_c}{\epsilon}\right)^2}}{2\pi \int_0^{\delta_c} s \left(e^{-\left(\frac{|\mathbf{x}|}{\epsilon}\right)^2} - e^{-\left(\frac{\delta_c}{\epsilon}\right)^2} \right) ds}. \quad (2.37)$$

L'intégrale au dénominateur permet de respecter la condition d'unicité (ici en deux dimensions). Elle se calcule analytiquement quand on choisit $\delta_c = n_p \epsilon$. n_p définit ainsi le nombre de pas d'interpolation à partir duquel on va effectuer la coupure. Le noyau est à décroissance rapide sur l'intervalle $[0; 3]$, il est donc raisonnable de choisir $\delta_c = 3\epsilon$.

Le choix de ce noyau se justifie par sa régularité et la plus grande stabilité (par rapport aux autres noyaux de type *splines*) de l'interpolation qu'il fournit. En effet, comme le montre Morris [163], ce type de fonction d'interpolation est moins sensible aux variations du paramètre κ , définissant le rapport entre le pas d'interpolation ϵ et l'espacement entre particules h :

$$\epsilon = \kappa h. \quad (2.38)$$

La valeur de κ détermine ainsi le nombre de particules prises en compte dans l'interpolation en un point donné. Sa valeur, couramment supérieure à 1, varie selon le noyau choisi et la précision souhaitée. En effet, quand on augmente κ , on augmente la taille du support du noyau, et donc le nombre de particules voisines à *interroger* pour effectuer l'interpolation d'une grandeur. Les erreurs liées à la forme discrète des interpolations sont alors réduites. Le coût de calcul est, quant à lui, augmenté (figure 2.4). Une petite variation du paramètre κ a d'ailleurs une influence croissante sur le nombre de particules à prendre en compte quand on passe à un problème en dimension supérieure. Dans notre cas nous posons $\kappa = 1,33$, ce qui donne en moyenne en 1 dimension



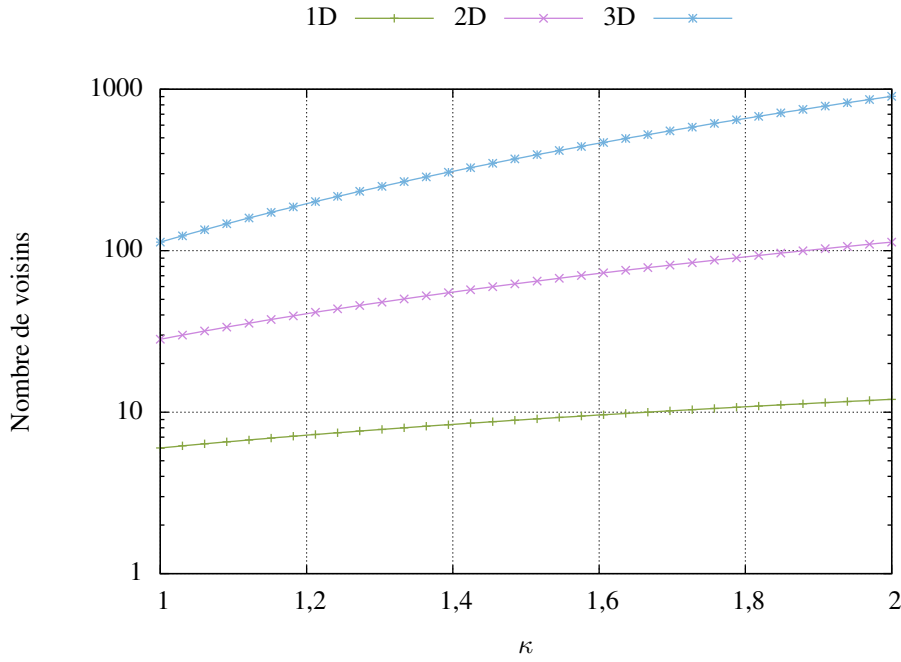


FIGURE 2.4 – Augmentation du nombre de voisins en fonction de la valeur choisie pour κ , dans le cas du noyau gaussien modifié (équation (2.37)) avec $\delta_c = 3\epsilon$ et selon le nombre de dimensions d'espace.

d'espace :

$$\frac{\text{Longueur du support}}{\text{Taille d'une particule}} = \frac{2\delta}{h} = \frac{2 * 3\kappa h}{h} = 6 * 1,33 \approx 8 \quad \text{particules.}$$

En 2 dimensions :

$$\frac{\text{Aire du support}}{\text{Aire d'une particule}} = \frac{\pi\delta^2}{h^2} = \frac{9\pi\kappa^2 h^2}{h^2} = 9\pi(1,33)^2 \approx 50 \quad \text{particules.}$$

Et en 3 dimensions :

$$\frac{\text{Volume du support}}{\text{Volume d'une particule}} = \frac{4\pi\delta^3}{3h^3} = \frac{36\pi\kappa^3 h^3}{h^3} = 36\pi(1,33)^3 \approx 266 \quad \text{particules.}$$

Si on définit $\delta_c = 3\epsilon$, le noyau gaussien (2.37) prend la forme suivante en 1D (figure 2.5) :

$$\eta_\epsilon^0(\mathbf{x}) = \frac{e^{-\left(\frac{|\mathbf{x}|}{\epsilon}\right)^2} - e^{-9}}{\epsilon(\sqrt{\pi} - 6e^{-9})}, \quad (2.39)$$

son gradient s'écrit :

$$\nabla \eta_\epsilon^0(\mathbf{x}) = \frac{-2(\mathbf{x})}{\epsilon^2} \eta_\epsilon^0(\mathbf{x}). \quad (2.40)$$

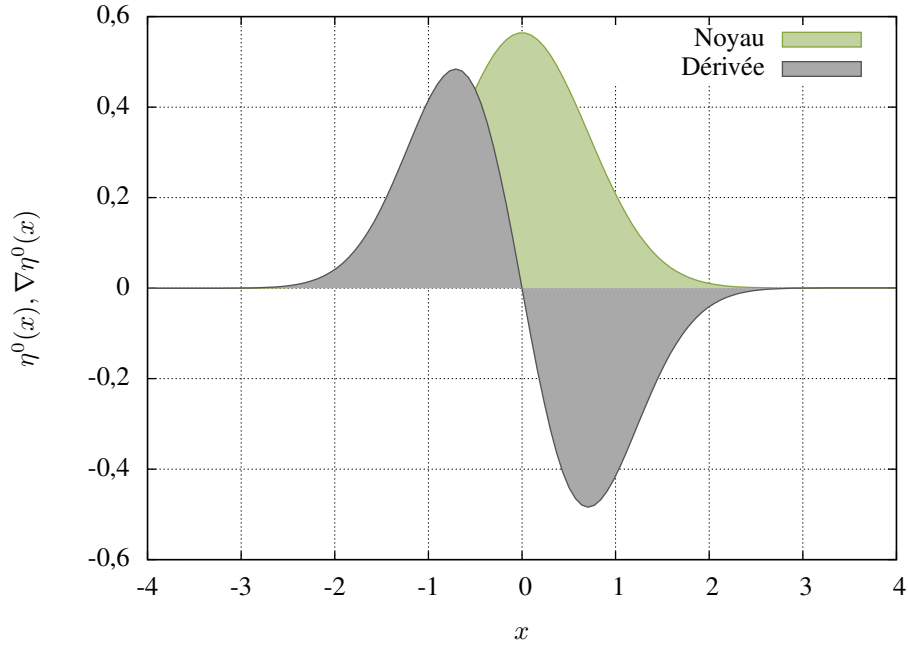


FIGURE 2.5 – Représentation graphique du noyau gaussien modifié en 1D (équation (2.39)) et de sa dérivée (équation (2.40)).

On peut observer que l'impact lié au coût de calcul de la fonction exponentielle peut être réduit si on calcule le gradient à partir de la valeur du noyau lui-même. En 2 dimensions, on a :

$$\eta_{\epsilon}^0(\mathbf{x}) = \frac{e^{-\left(\frac{|\mathbf{x}|}{\epsilon}\right)^2} - e^{-9}}{\pi\epsilon^2(1 - 10e^{-9})} \quad (2.41)$$

Et en 3D :

$$\eta_{\epsilon}^0(\mathbf{x}) = \frac{e^{-\left(\frac{|\mathbf{x}|}{\epsilon}\right)^2} - e^{-9}}{\epsilon^3(\pi^{3/2} - 36\pi e^{-9})} \quad (2.42)$$

Ainsi, pour la suite, retenons les deux formules d'interpolation SPH d'un champ et de son gradient sous forme discrète :

$$\mathcal{L}^0 f(\mathbf{x}_i) = \langle f(\mathbf{x}_i) \rangle = \sum_j f(\mathbf{x}_j) \eta_{\epsilon}^0(\mathbf{x}_i - \mathbf{x}_j) V_j \quad (2.43)$$

et

$$\mathcal{L}^1 f(\mathbf{x}_i) = \langle \nabla f(\mathbf{x}_i) \rangle = \sum_j f(\mathbf{x}_j) \nabla \eta_{\epsilon}^0(\mathbf{x}_i - \mathbf{x}_j) V_j. \quad (2.44)$$



Chapitre 3

Application aux équations de la Mécanique des Fluides

A l'aide des outils définis précédemment, de nombreux systèmes d'équations aux dérivées partielles peuvent être résolus. Le but de cette thèse étant d'appliquer la méthode SPH à l'étude d'écoulements instationnaires, nous ne passerons pas en revue tous les modèles SPH qui ont pu être écrits dans les nombreux domaines de la physique. La discrétisation des équations de Navier-Stokes par SPH est cependant le domaine de recherche le plus actif actuellement avec cette méthode. Nous allons maintenant décrire les différentes stratégies existantes pour la modélisation d'écoulements fluides.

3.1 Modélisation d'un fluide incompressible

Le but de cette étude est la simulation des équations de Navier-Stokes (ou Euler si on néglige la viscosité) pour un fluide incompressible. En description lagrangienne, formulation vitesse et pression, les équations d'Euler s'écrivent :

$$\frac{D\mathbf{u}}{Dt} = -\frac{\nabla P}{\rho} + \mathbf{g} \quad (3.1)$$

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

$$+ C.L. \quad (3.3)$$

Les variables \mathbf{u} , P et ρ définissent respectivement la vitesse, la pression et la masse volumique de la particule. $\mathbf{g} = (0, -g)$ représente l'action de la gravité. $\frac{D}{Dt}$ est la dérivée matérielle décrivant l'évolution, au cours du temps t , d'une grandeur le long de la trajectoire de la particule considérée. Un certain nombre de conditions limites peuvent être prises en compte et sont notées *C.L.* ; leur mise en œuvre sera expliquée plus loin dans ce manuscrit. L'équation (3.1) est l'équation de conservation de la quantité de mouvement, l'hypothèse d'incompressibilité est traduite par le



champ de vitesse à divergence nulle (équation (3.2)). Dans tout approche numérique, le traitement de ces équations se montre compliqué par le fait que le système ne semble pas clos. En effet, le gradient de pression dans l'équation (3.1) nécessite la connaissance du champ de pression sur l'ensemble du domaine de calcul. Or, cette pression n'est pas, a priori, déductible directement des équations (3.1), (3.2) ou (3.3).

Cependant, Chorin [33] a imaginé une procédure, en différences finies, permettant de déduire une pression en utilisant l'hypothèse d'incompressibilité du fluide. Cette approche fut reprise plus tard par Cummins et Rudman [50] pour la méthode SPH.

Par nature, SPH est plutôt adaptée à la modélisation d'un fluide compressible :

$$\frac{D\mathbf{u}}{Dt} = -\frac{\nabla P}{\rho} + \mathbf{g} \quad (3.4)$$

$$\frac{D\rho}{Dt} = -\rho \nabla \cdot \mathbf{u} \quad (3.5)$$

$$+ C.L. \quad (3.6)$$

En effet, la pression nécessaire à l'évaluation du second membre de l'équation de conservation de la quantité de mouvement peut, dans ce cas, se déduire d'une loi d'état adaptée au fluide considéré. Cette loi a pour variable la densité locale du fluide, directement déduite de l'équation de conservation de la masse (3.5). Cette approche est la plus employée dans les codes de calcul SPH, depuis la première application de SPH aux écoulements à surface libre [155].

3.1.1 Approche faiblement compressible (WCSPH)

Les équations d'Euler pour un fluide compressible s'écrivent de la manière suivante :

$$\frac{D\mathbf{x}}{Dt} = \mathbf{u} \quad (3.7)$$

$$\frac{D\mathbf{u}}{Dt} = -\frac{\nabla P}{\rho} + \mathbf{g} \quad (3.8)$$

$$\frac{D\rho}{Dt} = -\rho \nabla \cdot \mathbf{u} \quad (3.9)$$

$$+ C.L. \quad (3.10)$$

Comme mentionné plus tôt, une loi d'état pour la pression est ajoutée afin de clore le système. La littérature propose de nombreuses lois d'état, selon le domaine d'application. Dans le cas des écoulements à surface libre, la plus populaire, utilisée par Monaghan [155], est :

$$P = P_0 + \frac{\rho_0 c_0^2}{\gamma} \left(\left(\frac{\rho}{\rho_0} \right)^\gamma - 1 \right) \quad (3.11)$$

où γ est la constante polytropique du milieu modélisé, P_0 est une pression de référence à la surface libre, ρ_0 est la densité nominale et c_0 est la vitesse nominale du son. Cette équation est issue de l'équation de Laplace utilisée pour les transformations adiabatiques des gaz parfaits :

$$\frac{P}{\rho^\gamma} = cste \quad (3.12)$$

Elle a ensuite été modifiée de sorte que la pression calculée P soit la pression de référence P_0 (choisie nulle le plus souvent) quand le fluide est à sa densité nominale. Pour la modélisation des gaz parfaits, la constante polytropique γ prend pour valeur 1, 4. Pour la modélisation d'un fluide quasi-incompressible, telle que l'eau, la valeur de γ est fixée à 7. On retrouve ainsi l'équation de Cole [11].

En choisissant une pression de référence nulle, l'équation d'état, qui est celle employée dans la suite de ce manuscrit, est :

$$P = \frac{\rho_0 c_0^2}{7} \left(\left(\frac{\rho}{\rho_0} \right)^7 - 1 \right) \quad (3.13)$$

ρ_0 est fixée à sa valeur nominale 1000 kg/m^3 . La vitesse du son nominale c_0 doit correspondre à la vitesse réelle du son dans le fluide. Or en pratique, l'utilisation de la vitesse du son dans l'eau, 1480 m/s , conduirait à des simulations très coûteuses en temps de calcul, à cause des très petits pas de temps imposés par la condition CFL (Courant-Friedrich-Levy). Ainsi, comme le but de la simulation n'est pas d'observer les effets acoustiques, il est possible de définir cette vitesse du son à une valeur plus faible que la valeur réelle, de façon à avoir des pas de temps de grandeur raisonnable, tout en limitant la compressibilité du fluide.

La vitesse du son est alors déterminée par l'équivalence suivante [155] :

$$\frac{|\delta\rho|}{\rho} \approx \frac{|\mathbf{u}_{\max}|^2}{c_0^2} \quad (3.14)$$

Ainsi, en choisissant une vitesse du son égale à 10 fois la vitesse maximale du fluide, la variation de la masse volumique au cours de la simulation ne dépassera pas 1%.

On peut en déduire que le choix de la vitesse du son numérique dépend du problème traité ; plus la dynamique du système est rapide, plus la vitesse du son sera grande et les pas de temps petits. Ce point de la méthode est sujet à débat au sein de la communauté SPH. Il fait aussi souvent l'objet de critiques, qui remettent en cause la validité des résultats. L'utilisation d'une vitesse du son artificielle est pourtant la seule approche permettant d'allier validité et performances des simulations. En effet, les résultats exposés dans le chapitre suivant en démontrent la validité et les équations obtenues peuvent être résolues de façon assez aisée. Le plus grand problème de l'approche faiblement-compressible n'est pas l'utilisation d'une vitesse du son arbitraire. En effet, si on observe le champ de pression dans un cas donné, avec des conditions limites et une solution initiale bien posées, celui-ci est très proche du champ réel. Le point faible de ce modèle vient plutôt



du fait que la stabilité des simulations dépend directement du paramètre déterminant la vitesse du son numérique, qu'il faut calibrer avec attention. De plus, comme pour un fluide compressible, des ondes acoustiques numériques peuvent apparaître et perturber l'évolution de la simulation.

Néanmoins, une alternative existe, brièvement développée dans la partie suivante, qui propose un calcul plus rigoureux du champ de pression. Cependant, sa mise en œuvre requiert l'utilisation d'algorithmes plus complexes qui font actuellement l'objet d'une recherche approfondie au sein de la communauté et n'apportent pas encore de solution satisfaisante, mais prometteuse.

3.1.2 Approche incompressible (ISPH)

Dans le cas d'un fluide incompressible, les équations d'Euler deviennent alors :

$$\frac{D\mathbf{x}}{Dt} = \mathbf{u} \quad (3.15)$$

$$\frac{D\mathbf{u}}{Dt} = -\frac{\nabla P}{\rho} + \mathbf{g} \quad (3.16)$$

$$\begin{aligned} \nabla \cdot \mathbf{u} &= 0 \quad (3.17) \\ &+ C.L. \end{aligned}$$

Pour respecter la condition (3.17) dans le formalisme SPH, la technique la plus employée est la méthode de *projection*, développée initialement par Chorin [33] et adaptée à SPH par Cummins et Rudman [50]. Elle est basée sur la décomposition de Helmholtz-Hodge qui établit qu'un champ vectoriel \mathbf{w} sur un domaine Ω peut être décomposé, de façon unique, en :

$$\mathbf{w} = \mathbf{w}^* + \nabla q \quad (3.18)$$

où q est un champ scalaire et \mathbf{w}^* est un champ à divergence nulle :

$$\nabla \cdot \mathbf{w}^* = 0 \quad (3.19)$$

Ainsi, en calculant la divergence du champ initial, on a :

$$\nabla \cdot \mathbf{w} = \nabla \cdot (\mathbf{w}^* + \nabla q) = \nabla \cdot \mathbf{w}^* + \Delta q \quad (3.20)$$

En combinant les équations (3.19) et (3.20), on obtient l'équation de Poisson suivante, permettant de déduire le champ scalaire q :

$$\Delta q = \nabla \cdot \mathbf{w} \quad (3.21)$$

L'unicité de la solution dépend des conditions aux limites imposées. Si ces conditions sont uniquement des conditions de Neumann :

$$\frac{\partial q}{\partial \mathbf{n}} = C \quad (3.22)$$

où \mathbf{n} est la normale extérieure au contour $\partial\Omega$ du domaine Ω , alors il existe une infinité de solutions toutes égales à une constante additive près. En revanche, si les conditions aux limites font intervenir une condition de Dirichlet, alors la solution est unique.

Cette équation est utilisée au sein d'un algorithme de projection afin de déterminer à chaque itération un champ de vitesse à divergence nulle. Il existe différentes variantes de cette méthode :

- La plupart utilisent la divergence du champ de vitesse initial comme second membre de l'équation de Poisson [43].
- D'autres s'appuient sur un corollaire de cette condition de divergence nulle, la constance de la masse volumique sur le domaine [98].
- Ou encore sur une combinaison des deux [238].

Le problème de cette approche incompressible vient essentiellement de la résolution de l'équation de Poisson. Dans une majorité des cas, elle est effectuée sur une grille sous-jacente par un solveur externe de type Krylov (GMRES [198] ou BI-CGSTAB [227]). Une autre approche existante consiste à utiliser des multiplicateurs de Lagrange afin de limiter les variations de masse volumique [67], mais la méthode de projection reste la plus répandue. Les applications se sont d'abord limitées aux écoulements confinés [97] pour simuler, par exemple, des écoulements polyphasiques ou élastiques [189, 190].

La difficulté liée à la condition de pression à imposer à la surface libre a longtemps interdit l'utilisation de cette méthode pour l'hydrodynamique. Certaines solutions ont été proposées, qui consistent essentiellement à détecter les particules les plus proches de la surface pour leur imposer une pression nulle. Les méthodes de détection diffèrent : Shao et Lo [207] détectent les particules possédant moins de voisins que les autres et leur impose une pression nulle ; le même auteur abandonna ensuite cette technique pour identifier comme particules de surface libre celles dont le ratio masse volumique "numérique" sur masse volumique de référence devient inférieur à un certain critère [204, 6]. Tous les voisins de ces particules identifiées "de surface" sont ensuite symétrisés par rapport à la particule avec une pression opposée, afin d'obtenir une pression nulle sur la particule à l'issue de la résolution de l'équation de Poisson ; Khayyer *et al.* [109] emploient également cette approche et une variante basée sur le positionnement des voisins. Dans tous les cas, la qualité de la condition imposée est souvent contestable. Les écoulements dits rapides souffrent donc de cette approche. En revanche pour de nombreux autres écoulements, son intérêt peut être démontré [124]. Lee *et al.* [125] proposent une comparaison de l'approche faiblement compressible et de l'approche incompressible, tendant à prouver l'efficacité de la seconde par rapport à la première, mais ces résultats peuvent être remis en cause par le manque de soin apporté au traitement de la faible compressibilité pour la première. Nous avons essayé ce modèle mais les instabilités liées au traitement de la surface libre, et la difficulté liée à la résolution de cette équation de Poisson en simulation multi-processeurs, ont justifié son abandon. En revanche, et comme nous l'exposons dans la suite de ce manuscrit, certains outils développés ensuite pour le code faiblement compressible devraient nous permettre d'envisager maintenant cette possibilité (solveur de Poisson parallèle sans



grille sous-jacente, ni matrice) et l'utilisation des matrices de renormalisation devraient permettre de mieux identifier les particules proches de la surface, il reste alors à leur appliquer une condition de pression correcte, sujet encore ouvert et à explorer.

On trouve cependant des applications à des problèmes de génie côtier qui font appel à cette approche [83, 137, 206, 205].

3.2 Équations d'Euler

Dans cette partie, le schéma d'interpolation SPH est appliqué aux équations d'Euler afin de résoudre itérativement ces équations sur le champ de particules et ainsi simuler l'évolution temporelle du système. Plusieurs procédures existent pour la mise en forme des équations d'Euler discrètes. Plusieurs améliorations ont d'ailleurs été apportées au cours des années à l'écriture de ces équations. Cette partie n'est pas destinée à toutes les passer en revue, mais plutôt à expliquer comment peut être construit un système d'équations discrètes, conservatif et consistant.

Ainsi, reprenons le système d'équations continues à résoudre selon l'approche WCSPH exprimée sur une particule i :

$$\frac{D\mathbf{x}_i}{Dt} = \mathbf{u}_i \quad (3.23)$$

$$\frac{D\mathbf{u}_i}{Dt} = -\frac{\nabla P_i}{\rho_i} + \mathbf{g} \quad (3.24)$$

$$\frac{D\rho_i}{Dt} = -\rho_i(\nabla \cdot \mathbf{u}_i) \quad (3.25)$$

$$P_i = \frac{\rho_0 c_0^2}{\gamma} \left(\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (3.26)$$

$$+ C.L. \quad (3.27)$$

Dans un premier temps, l'équation de déplacement des particules ne nécessite aucune attention particulière. Intéressons-nous directement à l'équation de conservation de la quantité de mouvement (3.24). Cette équation fait intervenir le gradient de pression qui doit être évalué pour chaque particule du domaine. Ainsi en appliquant directement la formule d'interpolation (2.44) au second membre de l'équation (3.24), on obtient :

$$\frac{\nabla P_i}{\rho_i} = \frac{1}{\rho_i} \sum_j P_j \nabla \eta_\epsilon^0(\mathbf{x}_i - \mathbf{x}_j) \frac{m_j}{\rho_j} \quad (3.28)$$

De la même manière, l'équation de conservation de la masse est réécrite sous la forme :

$$\rho_i(\nabla \cdot \mathbf{u}_i) = \rho_i \sum_j \mathbf{u}_j \cdot \nabla \eta_\epsilon^0(\mathbf{x}_i - \mathbf{x}_j) \frac{m_j}{\rho_j} \quad (3.29)$$

Pour alléger l'écriture, nous allons adopter les notations suivantes :

$$\mathbf{u}_{ij} = \mathbf{u}_i - \mathbf{u}_j, \quad (3.30)$$

$$\eta_\epsilon^0(\mathbf{x}_{ij}) = \eta_\epsilon^0(\mathbf{x}_i - \mathbf{x}_j) \quad (3.31)$$

et

$$\nabla \eta_\epsilon^0(\mathbf{x}_{ij}) = \nabla \eta_\epsilon^0(\mathbf{x}_i - \mathbf{x}_j) \quad (3.32)$$

Ces équations sont les premières à avoir été employées dans les simulations SPH. On peut remarquer que dans chacune de ces équations, les sommes regroupent les influences de toutes les particules voisines sur la particule i . Il est tout à fait logique que ces influences respectent le principe d'action-réaction. Or, la formulation écrite comme ceci ne l'assure pas complètement. En effet, prenons l'influence, dans l'équation (3.28), d'une particule j sur la particule i :

$$F_{ji} = P_j \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (3.33)$$

De même, regardons l'influence que peut avoir la particule i sur la particule j :

$$F_{ij} = P_i \nabla \eta_\epsilon^0(\mathbf{x}_{ji}) \frac{m_i}{\rho_i} \quad (3.34)$$

On s'attend à avoir l'égalité :

$$F_{ji} = -F_{ij} \quad (3.35)$$

Or,

$$-F_{ij} = -P_i \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_i}{\rho_i} = P_i \nabla \eta_\epsilon^0(\mathbf{x}_{ji}) \frac{m_i}{\rho_i} \quad (3.36)$$

L'égalité (3.35) ne peut être respectée que si :

$$P_j \frac{m_j}{\rho_j} = P_i \frac{m_i}{\rho_i} \quad (3.37)$$

Ce que nous ne pouvons pas assurer avec la forme (3.28). L'intérêt de la réciprocité des actions entre particules est discuté dans la bibliographie, Vaughan [229] montre qu'elle est indispensable à la conservation de l'énergie et à la stabilité globale du schéma, tandis que Ma et Ge [140] remettent en cause l'utilisation de formes discrètes symétrisées des équations. Il paraît important que l'énergie soit conservée quand on cherche à simuler, par exemple, la propagation de houle en canal. Il faut donc reformuler ces équations afin de restaurer la réciprocité des interactions. Soient



les identités suivantes, valables $\forall \sigma \in \mathbb{R}$, [172] :

$$\frac{\nabla P}{\rho} = \frac{P}{\rho^\sigma} \nabla \left(\frac{1}{\rho^{1-\sigma}} \right) + \rho^{\sigma-2} \nabla \left(\frac{P}{\rho^{\sigma-1}} \right) \quad (3.38)$$

$$\rho(\nabla \cdot \mathbf{u}) = \frac{\nabla \cdot (\rho^{\sigma-1} \mathbf{u}) - \mathbf{u} \cdot \nabla \rho^{\sigma-1}}{\rho^{\sigma-2}} \quad (3.39)$$

En discrétisant ces deux équations, on obtient pour les équations d'Euler :

$$\frac{D\mathbf{x}_i}{Dt} = \mathbf{u}_i \quad (3.40)$$

$$\frac{D\mathbf{u}_i}{Dt} = - \sum_j \left(\frac{P_i}{\rho_i^\sigma \rho_j^{2-\sigma}} + \frac{P_j}{\rho_i^{2-\sigma} \rho_j^\sigma} \right) \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) m_j + \mathbf{g} \quad (3.41)$$

$$\frac{D\rho_i}{Dt} = \sum_j \left(\frac{\mathbf{u}_{ij}}{\rho_i^{\sigma-2} \rho_j^{2-\sigma}} \right) \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) m_j \quad (3.42)$$

$$P_i = \frac{\rho_0 c_0^2}{7} \left(\left(\frac{\rho_i}{\rho_0} \right)^7 - 1 \right) + C.L. \quad (3.43)$$

On a donc un système d'équations conservatives. Il reste à s'assurer que le choix d'une valeur de σ ne va pas conduire à deux équations de conservation de la masse et de la quantité de mouvement non consistantes. Ce travail a été réalisé par Bonet et Lok [16], qui démontre par une étude variationnelle que ces deux équations sont liées : une condition nécessaire à la consistance du schéma est le choix d'un σ identique pour les deux équations. Ainsi, dans la littérature on retrouve souvent les deux formulations qui respectent ce principe :

Formulation $\sigma = 2$:

Formulation employée par Monaghan [155] ou Dalrymple et Rogers [52] :

$$\frac{D\mathbf{u}_i}{Dt} = - \sum_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} \right) \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) m_j + \mathbf{g} \quad (3.44)$$

$$\frac{D\rho_i}{Dt} = \sum_j \mathbf{u}_{ij} \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) m_j \quad (3.45)$$

Formulation $\sigma = 1$:

Formulation employée par Oger [172] ou Colagrossi et Landrini [41] :

$$\frac{D\mathbf{u}_i}{Dt} = - \sum_j \left(\frac{P_i + P_j}{\rho_i \rho_j} \right) \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) m_j + \mathbf{g} \quad (3.46)$$

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \left(\frac{\mathbf{u}_{ij}}{\rho_j} \right) \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) m_j \quad (3.47)$$

Ces deux formulations sont incluses dans notre code SPH mais seule la dernière est employée. D'après Oger [172] et Colagrossi et Landrini [41], celle-ci est plus adaptée à la simulation d'écoulements multi-phasiques car elle ne fait pas intervenir d'approximation de $\nabla \rho$ (équation (3.39)), elle provoque donc moins d'erreurs au niveau de l'interface entre les deux phases.

Voici donc la forme des équations d'Euler employée pour les simulations présentées dans ce manuscrit :

$$\frac{D\mathbf{u}_i}{Dt} = - \frac{1}{\rho_i} \sum_j (P_i + P_j) \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} + \mathbf{g} \quad (3.48)$$

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \mathbf{u}_{ij} \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (3.49)$$

3.3 Équations de Navier-Stokes

On s'intéresse maintenant aux équations décrivant le comportement d'un fluide visqueux :

$$\frac{D\mathbf{u}_i}{Dt} = - \frac{\nabla P_i}{\rho_i} + \frac{\nabla \cdot \boldsymbol{\tau}_i}{\rho_i} + \mathbf{g} \quad (3.50)$$

$$\frac{D\rho_i}{Dt} = -\rho_i (\nabla \cdot \mathbf{u}_i) \quad (3.51)$$

$$P_i = \frac{\rho_0 c_0^2}{\gamma} \left(\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (3.52)$$

$$+ C.L. \quad (3.53)$$

avec $\boldsymbol{\tau}_i$, le tenseur des contraintes visqueuses de la particule i . La prise en compte de ce terme visqueux dans l'équation (3.51) ne modifie pas complètement le schéma numérique. La littérature propose une grande variété de solutions pour l'approximation de ce terme avec SPH, que l'on peut classer en deux catégories selon la prise en compte, ou non, de la compressibilité du fluide (faible, de l'ordre de 1% de la masse volumique de référence, rappelons-le).

Le tenseur des contraintes visqueuses, pour un fluide compressible newtonien, peut s'écrire



sous la forme suivante :

$$\boldsymbol{\tau} = \mu \left((\nabla \otimes \mathbf{u}) + (\nabla \otimes \mathbf{u})^t \right) + \lambda (\nabla \cdot \mathbf{u}) \mathbf{I} \quad (3.54)$$

où \mathbf{I} est le tenseur unité, μ est la viscosité dynamique du fluide et λ est le second coefficient de viscosité. Souvent, l'hypothèse suivante est appliquée :

$$\beta = \frac{2}{3}\mu + \lambda \quad (3.55)$$

où β est appelée la viscosité de volume. Ainsi, si on prend le terme de coordonnées α et β du tenseur, celui-ci s'écrit :

$$\tau^{\alpha\beta} = \mu \left(\frac{\partial \mathbf{u}^\alpha}{\partial \mathbf{x}^\beta} + \frac{\partial \mathbf{u}^\beta}{\partial \mathbf{x}^\alpha} \right) + \left(\beta - \frac{2}{3}\mu \right) (\nabla \cdot \mathbf{u}) \delta^{\alpha\beta} \quad (3.56)$$

avec :

$$\delta^{\alpha\beta} = \begin{cases} 1 & \text{si } \alpha = \beta \\ 0 & \text{sinon} \end{cases} \quad (3.57)$$

Si on néglige la viscosité de volume, on obtient alors cette expression du terme visqueux :

$$\tau^{\alpha\beta} = \mu \left(\frac{\partial \mathbf{u}^\alpha}{\partial \mathbf{x}^\beta} + \frac{\partial \mathbf{u}^\beta}{\partial \mathbf{x}^\alpha} \right) - \frac{2}{3}\mu (\nabla \cdot \mathbf{u}) \delta^{\alpha\beta} \quad (3.58)$$

dont le second terme dépend de la compressibilité du fluide.

Prise en compte de la compressibilité

Takeda *et al.* [223] proposent une approximation du tenseur sous cette forme (équation (3.58)). En supposant que le noyau η_ϵ^0 employé est à symétrie radiale :

$$\eta_\epsilon^0(\mathbf{x}_{ij}) = \eta_\epsilon^0(r_{ij}), \quad (3.59)$$

avec $r_{ij} = \|\mathbf{x}_{ij}\|$, ce qui est le cas avec la fonction gaussienne que nous utilisons. Alors son gradient peut s'écrire :

$$\nabla \eta_\epsilon^0(\mathbf{x}_{ij}) = \frac{\mathbf{x}_{ij}}{r_{ij}} \frac{\partial \eta_\epsilon^0(r_{ij})}{\partial r_{ij}} \quad (3.60)$$

L'approximation du tenseur par Takeda *et al.* [223] s'écrit, pour la particule i :

$$\begin{aligned}
\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i &= -\frac{\mu}{\rho_i} \sum_j \left[\left(r_{ij} \frac{\partial}{\partial r_{ij}} \left(\frac{1}{r_{ij}} \frac{\partial \eta_\epsilon^0(r_{ij})}{\partial r_{ij}} \right) \right. \right. \\
&\quad \left. \left. + \frac{2}{r_{ij}} \frac{\partial \eta_\epsilon^0(r_{ij})}{\partial r_{ij}} \right) \mathbf{u}_{ij} + \frac{1}{3} \left(\left(\frac{1}{r_{ij}} \frac{\partial \eta_\epsilon^0(r_{ij})}{\partial r_{ij}} \right) \mathbf{u}_{ij} \right. \right. \\
&\quad \left. \left. + \frac{\partial}{\partial r_{ij}} \left(\frac{1}{r_{ij}} \frac{\partial \eta_\epsilon^0(r_{ij})}{\partial r_{ij}} \right) \frac{\mathbf{x}_{ij} \cdot \mathbf{u}_{ij}}{r_{ij}} \mathbf{x}_{ij} \right) \right] \frac{m_j}{\rho_j}
\end{aligned} \tag{3.61}$$

Ce terme, assez compliqué et coûteux à calculer, est peu employé dans la littérature, en particulier en raison de la présence de la dérivée seconde du noyau.

Une autre variante, proposée par Espanol et Revenga [68] et également employée par Ferrari *et al.* [75], s'écrit :

$$\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i = - \sum_j \left(\frac{7}{3} \frac{\mu}{\rho_i} m_j \frac{\mathbf{u}_{ij}}{\rho_j} + \frac{5}{3} \frac{\mu}{\rho_i} m_j \frac{\mathbf{x}_{ij}}{\rho_j r_{ij}} \mathbf{x}_{ij} \cdot \mathbf{u}_{ij} \right) \frac{\mathbf{x}_{ij}}{r_{ij}^2} \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \tag{3.62}$$

tient également compte de la compressibilité du fluide.

Compressibilité négligée

La variation de masse volumique étant maintenue faible dans l'approche SPH, le tenseur est souvent réécrit en négligeant les effets de la compressibilité, on a alors :

$$\frac{\nabla \cdot \boldsymbol{\tau}}{\rho} = \frac{\mu}{\rho} \Delta \mathbf{u} \tag{3.63}$$

L'approximation de ce terme peut être réalisée, si on suit strictement le formalisme SPH, en employant la dérivée seconde du noyau. Cette technique, employée avec un noyau de type *B-Splines*, conduit généralement à une approximation assez mauvaise du laplacien, en raison du manque de régularité du noyau dérivé deux fois [211]. Afin de remédier à ce problème, certains emploient un noyau plus régulier, de type gaussien [223]. La technique de remaillage [29], présentée plus loin (§6.1.1), peut aussi être employée afin de ramener périodiquement la répartition des particules à une grille cartésienne. Cela a pour effet d'améliorer la précision des interpolations, et de limiter l'effet néfaste que pourrait avoir un noyau peu régulier. S'inspirant de la viscosité artificielle proposée par Monaghan [155], apportant une dissipation dans l'équation de conservation de la quantité de mouvement, Cleary [34] proposa une nouvelle forme de terme visqueux, écrite de la manière suivante :

$$\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i = - \sum_j m_j \frac{\xi}{\rho_i \rho_j} \frac{4\mu_i \mu_j}{\mu_i + \mu_j} \frac{\mathbf{u}_{ij} \cdot \mathbf{x}_{ij}}{r_{ij}^2 + \varepsilon^2} \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \tag{3.64}$$

La viscosité, dans son cas, peut être variable. La constante ε permet d'éviter toute singularité



de la fraction. Si on réécrit cette formule pour μ constant, on a :

$$\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i = - \sum_j m_j \frac{2\xi\mu}{\rho_i\rho_j} \frac{\mathbf{u}_{ij} \cdot \mathbf{x}_{ij}}{r_{ij}^2 + \varepsilon^2} \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \quad (3.65)$$

Après calibration sur un écoulement de Poiseuille, Cleary [34] établit que le paramètre ξ doit prendre pour valeur 4,963333 en précisant que cette valeur est indépendante de μ , mais limitant son usage à des écoulements dont les solutions sont connues. Sur le même écoulement, Basa *et al.* [10] proposent $\xi = 4,24$. Monaghan [158] passe en revue les études portées à cette approximation et indique que ce terme peut être utilisé sous la forme suivante :

$$\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i = - \sum_j m_j \frac{8\mu}{\rho_i\rho_j} \frac{\mathbf{u}_{ij} \cdot \mathbf{x}_{ij}}{r_{ij}^2 + \varepsilon^2} \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \quad (3.66)$$

sous réserve que l'équivalence suivante soit satisfaite :

$$\nu = \frac{\mu}{\rho_0} \sim \frac{1}{8} \alpha \epsilon c_0 \quad (3.67)$$

avec $\alpha > 0,01$. Une autre extension de terme visqueux artificiel [156] est proposée par Monaghan [159] et reprise par Kajtar et Monaghan [106].

Enfin, exploitant aussi un mélange de différences finies et de gradient SPH pour l'approximation de dérivées secondes, Morris *et al.* [164] proposent le terme suivant :

$$\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i = - \sum_j m_j \frac{\mu_i + \mu_j}{\rho_i\rho_j} \mathbf{u}_{ij} \frac{\mathbf{x}_{ij} \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij})}{r_{ij}^2} \quad (3.68)$$

En considérant à nouveau la viscosité constante, on obtient :

$$\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i = - \sum_j m_j \frac{2\mu}{\rho_i\rho_j} \mathbf{u}_{ij} \frac{\mathbf{x}_{ij} \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij})}{r_{ij}^2} \quad (3.69)$$

Basa *et al.* [10] proposent une étude précise de ces différents modèles et démontre que (3.66) conserve le moment angulaire mais peut conduire à des résultats très mauvais, tandis que (3.69) conserve le moment linéaire de façon exacte et le moment angulaire seulement de façon approximative car les forces visqueuses entre deux particules sont égales en norme et opposées mais elles ne sont pas colinéaires. Graham et Hughes [84] étudient avec précision la sensibilité des simulations d'écoulement visqueux aux variations du paramètre κ , en utilisant cependant le noyau d'interpolation M_4 , déjà connu pour être moins précis quand de faibles valeurs de κ sont employées. Il est à noter que si on développe cette dernière approximation (équation (3.69)) dans le cas d'un noyau gaussien et en exploitant (2.41), on a alors :

$$\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i = - \sum_j m_j \frac{4\mu_i}{\rho_i\rho_j\epsilon^2} \mathbf{u}_{ij} \eta_\epsilon^0(\mathbf{x}_{ij}) \quad (3.70)$$

De même, si on suit le traitement des opérateurs différentiels d'Eldredge *et al.* [66] (méthode PSE), et en utilisant (3.63), on a :

$$\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i = \left\langle \frac{\mu}{\rho} \Delta \mathbf{u} \right\rangle_i = \frac{\mu}{\rho_i} \frac{1}{\epsilon^2} \sum_j (-\mathbf{u}_{ij}) \eta_\epsilon^\Delta(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (3.71)$$

On trouve finalement que le noyau gaussien η_ϵ^Δ approchant le laplacien à l'ordre 2 peut s'écrire en fonction de η_ϵ^0 , quand on applique la procédure d'Eldredge *et al.* [66] :

$$\eta_\epsilon^\Delta(\mathbf{x}_{ij}) = 4\eta_\epsilon^0(\mathbf{x}_{ij}) \quad (3.72)$$

on obtient :

$$\left\langle \frac{\nabla \cdot \boldsymbol{\tau}}{\rho} \right\rangle_i = -\frac{\mu}{\rho_i} \frac{1}{\epsilon^2} \sum_j \mathbf{u}_{ij} 4\eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (3.73)$$

Les expressions (3.73) et (3.70) sont identiques, on aurait également pu faire cette remarque sur l'approximation du gradient de pression. Dans la suite de ce manuscrit, quand un écoulement visqueux est considéré, la forme (3.69) est employée. Finalement, les équations de Navier-Stokes complètes sont discrétisées de la manière suivante :

$$\frac{D\mathbf{x}_i}{Dt} = \mathbf{u}_i \quad (3.74)$$

$$\frac{D\mathbf{u}_i}{Dt} = -\frac{1}{\rho_i} \sum_j (P_i + P_j) \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} - \sum_j m_j \frac{2\mu}{\rho_i \rho_j} \mathbf{u}_{ij} \frac{\mathbf{x}_{ij} \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij})}{r_{ij}^2} + \mathbf{g} \quad (3.75)$$

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \mathbf{u}_{ij} \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (3.76)$$

$$P_i = \frac{\rho_0 c_0^2}{\gamma} \left(\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (3.77)$$

$$+ C.L. \quad (3.78)$$



Deuxième partie

Mise en œuvre, validations et applications



Dans la partie précédente, les équations ont été mises en forme selon la méthodologie SPH pour la modélisation d'un fluide faiblement compressible visqueux ou non. La mise en œuvre de formes discrètes requiert un certain nombre de techniques spécifiques à la méthode SPH, qu'il s'agit de développer maintenant. Pour l'illustration de ces outils, plusieurs écoulements *académiques* ont été étudiés. L'ensemble des algorithmes nécessaires à la modélisation de ces écoulements seront détaillés : modèle, prise en compte des conditions aux limites, amélioration de la précision, stabilisation ...

Le chapitre 4 porte sur l'application de SPH aux équations d'Euler. La déformation d'une goutte en deux dimensions d'espace est considérée, permettant la comparaison, puis la validation d'un modèle SPH sur un écoulement non-visqueux, en présence d'une surface libre.

Le chapitre 5 est consacré à l'application de conditions aux limites. Le modèle proposé est appliqué à l'effondrement d'une colonne d'eau, suivi d'un impact sur un mur vertical. Les résultats sont exposés et améliorés par correction du modèle établi au chapitre précédent.

Le chapitre 6 introduit la méthode de la *frontière immergée* pour les conditions aux bords de type adhérence. Le modèle visqueux combiné à la frontière immergée est validé sur l'écoulement autour d'un carré forcé dans un fluide compressible, en deux dimensions.

Le chapitre 7 montre finalement l'application du modèle, validé au cours des trois chapitres précédents, à la propagation d'ondes de gravité en canal ouvert. L'effet d'un obstacle immergé sur la houle est abordé et plusieurs résultats montrent la possibilité d'appliquer conjointement SPH et la frontière immergée dans ce contexte.



Chapitre 4

Étirement d'une goutte en deux dimensions

En deux dimensions, des particules discrétisent un domaine fluide initialement de forme circulaire. Leur vitesse et pression initiales sont définies analytiquement. L'évolution de la goutte soumise à ce champ de force est ensuite observée, en l'absence de viscosité. Ce cas ne faisant pas intervenir de conditions limites solides, il a permis la validation du modèle et l'apport des corrections adéquates pour l'amélioration des résultats. Il a également permis de tester la capacité de la méthode à suivre une interface représentée ici par la déformation du contour de la goutte.

4.1 Description du problème

Une goutte circulaire plane de rayon R est considérée (figure 4.1). Celle-ci est soumise au champ de vitesse initial suivant :

$$\begin{cases} u_0(x, y) = A_0x \\ v_0(x, y) = -A_0y \end{cases} \quad \forall (x, y), x^2 + y^2 \leq R, \quad (4.1)$$

où les composantes du champ de vitesse sont décrites en adoptant la notation $\mathbf{u} = (u, v)$ et $\mathbf{x} = (x, y)$. A_0 et R sont les deux paramètres qui régissent ce problème. La pression initiale, vérifiant les équations d'Euler en incompressible, est définie par :

$$P_0(x, y) = \frac{1}{2}\rho A_0^2 ((R^2 - (x^2 + y^2))) \quad (4.2)$$

Soumise à ces champs initiaux, la goutte se déforme au cours du temps (figure 4.2). Pour A_0 positif, la goutte subit un étirement suivant l'axe des x et une compression selon l'axe des y . L'écoulement généré est irrotationnel. L'incompressibilité du fluide a alors pour conséquence que le domaine fluide Ω conserve une forme elliptique au cours du temps. Une particule élémentaire de position



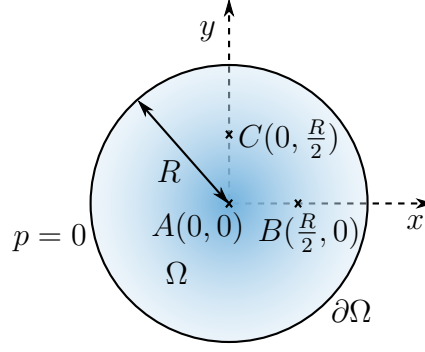


FIGURE 4.1 – Descriptions et notations pour l'étirement d'une goutte en domaine ouvert. Les points A, B et C représentent les positions des différentes mesures exposées dans la suite de ce manuscrit.

initial $\mathbf{x}_0 = (x_0, y_0)$ a alors pour position à l'instant t :

$$\begin{cases} x(t) = a(t)x_0 \\ y(t) = b(t)y_0 \end{cases}, \quad (4.3)$$

où $a(t)$ et $b(t)$ sont le demi-grand axe et le demi-petit axe de l'ellipse à l'instant t , respectivement. La surface de l'ellipse étant constante, le produit $a(t)b(t)$ l'est également. En exploitant cette propriété pour définir la fonction suivante :

$$A(t) = -\frac{\dot{a}(t)}{a(t)} = \frac{\dot{b}(t)}{b(t)}, \quad (4.4)$$

on peut, par dérivation des équations (4.3), écrire la solution du champ de vitesse :

$$\begin{cases} u(x, y, t) = A(t)x \\ v(x, y, t) = -A(t)y \end{cases} \quad (4.5)$$

En combinant les équations (4.5) aux équations d'Euler, l'expression du gradient de pression permet, par intégration et en considérant la pression nulle sur l'interface $\partial\Omega$, de définir la solution analytique du champ de pression :

$$P(x, y, t) = \frac{1}{2}\rho \left(-\dot{A}(t)(x^2 - y^2) - A(t)(x^2 + y^2) - a(t)^2(\dot{A}(t) - A(t)^2) \right) \quad (4.6)$$

L'expression de $A(t)$ s'obtient par intégration de l'équation différentielle du second ordre suivante :

$$\begin{cases} \ddot{A}(t)A(t) - 4\dot{A}(t)^2 + 2A(t)^4 = 0 \\ A(0) = A_0 \\ \dot{A}(0) = 0 \end{cases} \quad (4.7)$$

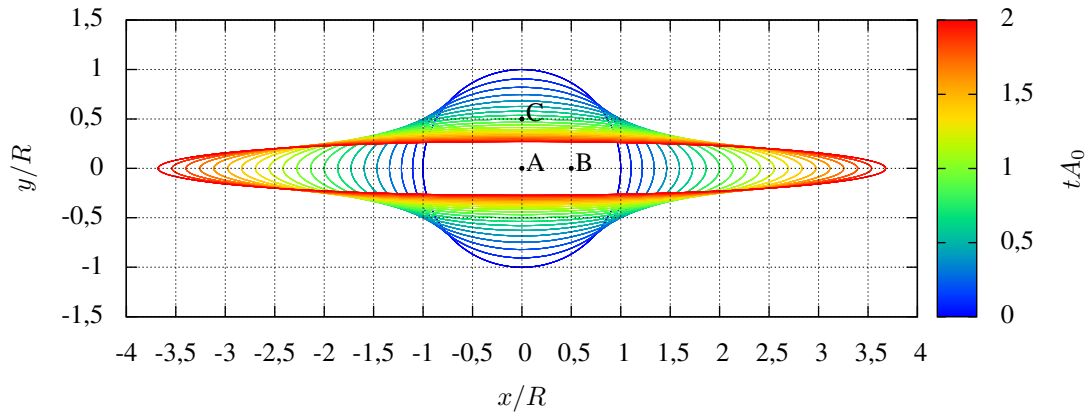


FIGURE 4.2 – Évolution analytique de la forme de la goutte au cours du temps. La couleur associée à chaque ellipse en définit l’instant.

En utilisant par exemple un schéma de Runge-Kutta, la valeur de $A(t)$ permet successivement de calculer celles de $a(t)$ et $b(t)$, puis les solutions des champs de pression et de vitesse.

4.2 État de l’art sur la modélisation de cet écoulement

Ce cas a déjà fait l’objet de modélisations par la méthode SPH faiblement compressible. Monaghan [155] compare les valeurs obtenues pour $a(t)$ par son modèle SPH aux valeurs analytiques, obtenant un très bon accord sur l’intervalle de temps $tA_0 \in [0; 0, 8]$. Ce résultat montrait l’intérêt d’une méthode lagrangienne pour le calcul d’interfaces, notamment pour des applications en hydrodynamique. L’étude de ce cas montre qu’au delà de l’intervalle de temps présenté par Monaghan [155], la méthode SPH peut avoir des difficultés à correctement représenter le bord de l’ellipse, tant celle-ci est étirée. Dans cette optique, les travaux de thèse de Colagrossi [35] montrent la capacité de son modèle à étendre l’intervalle d’étude du cas à $tA_0 \in [0; 2]$ tout en conservant la régularité du champ de pression et de l’interface. Son modèle sera détaillé, ainsi que celui de Ferrari *et al.* [75]. Ce dernier est une alternative simplifiée au schéma SPH *standard* de Monaghan [155] et au récent modèle SPH combiné à un solveur de Riemann, proposé par Vila [232]. Bien que l’intervalle de temps étudié par Ferrari *et al.* [75] soit également réduit à $tA_0 \in [0; 0, 8]$, les résultats montrent l’amélioration permise par l’utilisation de son modèle sur la régularité du champ de pression, et sur la représentation de la surface libre, à la fois. Enfin, ne portant pas sur le modèle lui-même, mais sur les techniques de *renormalisation*, visant à améliorer la précision des approximations, les travaux de Bonet et Lok [16] et Fang *et al.* [70] montrent que le modèle nécessite l’apport de corrections pour correctement reproduire le gradient de pression, particulièrement aux interfaces.



Malgré ces corrections, Bonet et Lok [16] se limite également à l'intervalle de temps $tA_0 \in [0; 0,8]$ et Fang *et al.* [70] à $tA_0 \in [0; 1]$

On trouve une étude de ce cas, par modélisation incompressible, dans un article de Ellero *et al.* [67] qui ne note pas de différence significative avec la solution faiblement compressible. De même, un autre algorithme incompressible a été proposé plus récemment par Shobeyri et Afshar [209]. La méthode est différente de SPH, il s'agit dans ce cas de la méthode DLSM (Discrete Least Squares Meshless method) où les interactions entre particules sont calculées par moindres carrés.

L'approche proposée par Vila [232], basée sur l'analogie entre Volumes Finis et SPH, n'a pas été étudiée car elle requerrait des modifications très lourdes du code de calcul et une étude précise du modèle en amont. En revanche, chacune des techniques proposées dans les autres travaux présentés dans le tableau (4.1) ont été étudiées, comparées et sont présentées maintenant.

Références par ordre chronologique	Modèle	Intervalle de temps tA_0	Interface bien représentée	Champ de pression lisse
Monaghan [155] 1994	XSPH + viscosité	[0;0,8]	✗ NON	✗ NON
Vila [232] 1999	SPH + solveur de Riemann	[0;0,8]	✗ NON	✓ OUI
Bonet et Lok [16] 1999	SPH + renormalisation	[0;0,8]	✓ OUI	✗ NON
Colagrossi [35] 2005	XSPH + densité MLS	[0;2]	✓ OUI	✓ OUI
Fang <i>et al.</i> [70] 2009	SPH + renormalisation	[0;1]	✓ OUI	✓ OUI
Ferrari <i>et al.</i> [75] 2009	SPH + flux Rusanov	[0;0,8]	✓ OUI	✓ OUI

TABLE 4.1 – Récapitulatif des modélisations avec SPH faiblement compressible de l'éirement d'une goutte.

4.3 Modélisation par SPH : schéma simple

Cet écoulement a été modélisé en employant d'abord le schéma SPH sans algorithme supplémentaire visant à en améliorer la précision ou la stabilité. Les paramètres employés sont les suivants : $\kappa = \epsilon/h = 1,33$, $R/\epsilon = 14,7$. Avec ces paramètres, la simulation fait interagir 1208 particules. Les équations d'Euler ont été intégrées en temps en utilisant un schéma *prédicteur-correcteur*, fréquemment employé en méthode SPH, avec un pas de temps $\Delta tA_0 = 10^{-3}$. La masse volumique au repos a été fixée à $\rho_0 = 1000kg/m^3$ et la vitesse du son à $c_0 = 14RA_0$ (avec $R = 1$ et $A_0 = 100$) de manière à ne pas observer de variations de la masse volumique supérieures

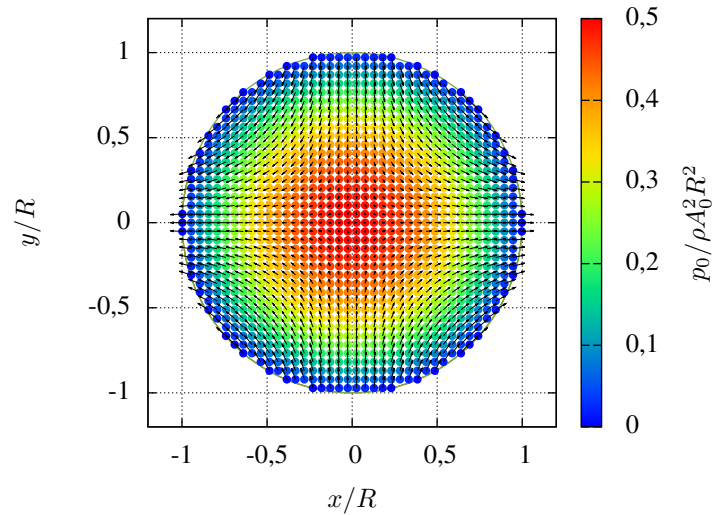


FIGURE 4.3 – Configuration initiale des particules pour la simulation de l'éirement d'une goutte en domaine ouvert. Les vecteurs symbolisent la vitesse initiale de la particule, colorée en fonction de la pression. Les particules sont positionnées suivant une grille cartésienne.

à 1%. Les particules sont initialement positionnées sur une grille cartésienne (figure 4.3). Les deux formulations les plus courantes en méthode SPH ont été comparées. Les résultats obtenus à l'aide de la formulation « $\sigma = 1$ » (équations (3.46) et (3.47) du §3.2), employée par Oger [172] ou Colagrossi et Landrini [41], sont montrés sur la figure (4.4). Ceux obtenus à l'aide de la formulation « $\sigma = 2$ » (équations (3.44) et (3.45)), employée par Monaghan [155] ou Dalrymple et Rogers [52], sont montrés sur la figure (4.5).

Les champs de pression montrent de fortes variations dues à des oscillations de très hautes fréquences qui apparaissent dès le début de la simulation. Cela avait déjà été observé par Colagrossi [35] et expliqué par la nature faiblement compressible du modèle. Aucune des deux formulations « $\sigma = 1$ » et « $\sigma = 2$ » ne se distingue d'ailleurs sur ce type d'écoulement. Comme mentionné dans la section 3.2, la formulation « $\sigma = 1$ » apporte un gain en précision significatif uniquement dans le cas d'écoulements polyphasiques, en réduisant les erreurs d'approximation au niveau des interfaces (discontinuité de masse volumique). Dans la suite de ce manuscrit, seule la formulation « $\sigma = 1$ » sera employée.

La figure (4.6) montre l'évolution de la pression au centre de la goutte (point A, voir figure (4.1)). La solution SPH brute montre effectivement de fortes oscillations. Celles-ci sont dues à la présence d'ondes de pression qui se propagent à la vitesse numérique choisie pour le son, c_0 . La fréquence de ces oscillations peut être déterminée par $f = c_0/\lambda$ avec λ , la longueur d'onde du signal de pression. Comme l'explique Colagrossi [35], la longueur d'onde du mode dominant est $\lambda = 4a(t)$, ce qui signifie qu'entre le début et la fin de la simulation, la fréquence de ce mode varie entre $3,5A_0$ et $13A_0$. Ainsi, en supprimant les oscillations de fréquences supérieures à $3,5A_0$, l'influence des ondes de pression (dues à la forme faiblement compressible du modèle)



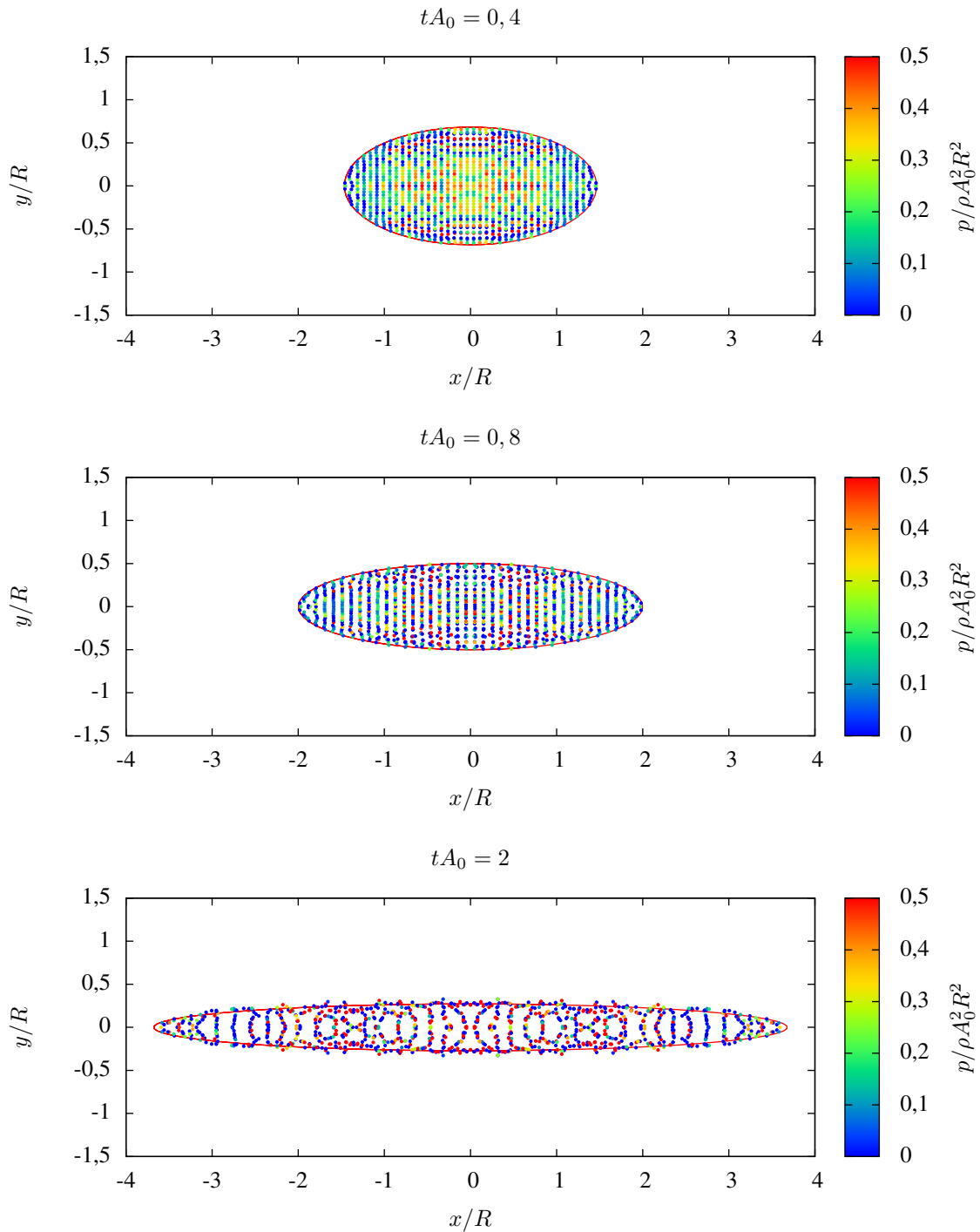


FIGURE 4.4 – Évolution de la goutte simulée à l'aide du modèle SPH simple, avec la formulation du gradient de pression de Oger [172] et Colagrossi et Landrini [41] ($\sigma = 1$, équations (3.46) et (3.47) du §3.2). Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.

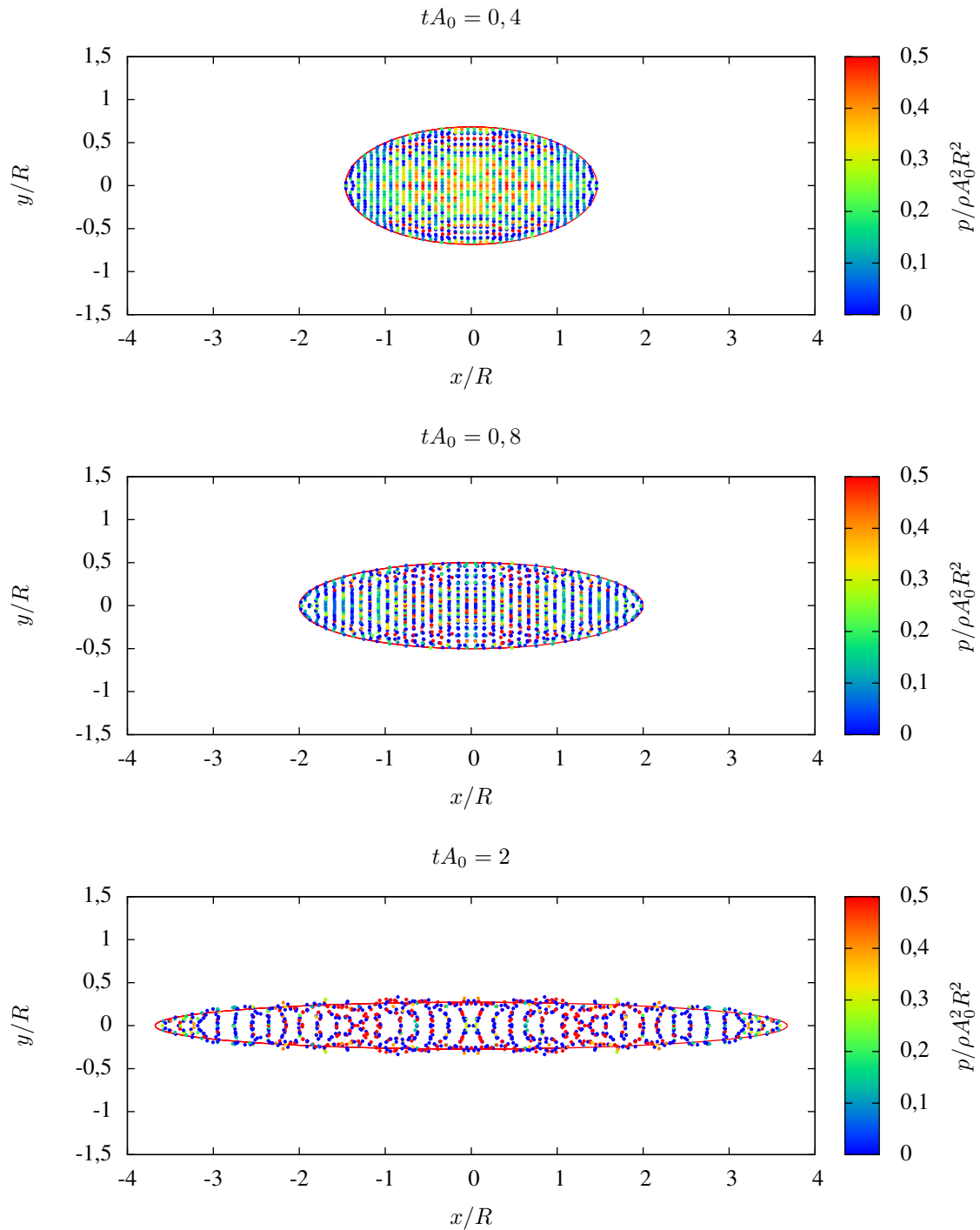


FIGURE 4.5 – Évolution de la goutte simulée à l'aide du modèle SPH simple, avec la formulation du gradient de pression de Monaghan [155] et Dalrymple et Rogers [52] ($\sigma = 2$, équations (3.44) et (3.45) du §3.2). Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.



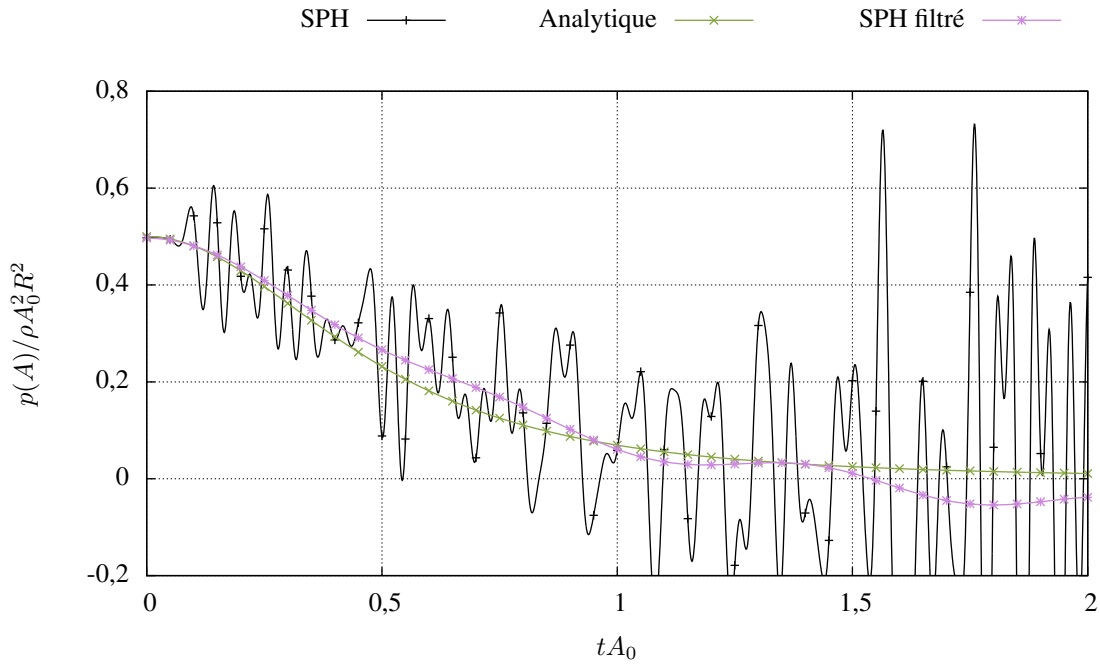


FIGURE 4.6 – Évolution de la pression calculée au point A (figure 4.1), avec la formulation du gradient de pression de Oger [172] et Colagrossi et Landrini [41] ($\sigma = 1$, équations (3.46) et (3.47) du §3.2). Comparaison entre la valeur calculée par SPH (en noir), la solution analytique (en vert) et la solution SPH filtrée (les fréquences $f > 3,5A_0$ ont été supprimées, en violet).

sur le signal mesuré en A sont retirées, et la solution filtrée se rapproche de la solution analytique incompressible.

4.4 Modélisation par SPH : schéma de Colagrossi [35]

Un certain nombre d'améliorations au schéma SPH original (simple) ont été apportées par Colagrossi au cours de ses travaux de thèse [35], combinant et améliorant certaines techniques proposées par Monaghan [155] (XSPH, viscosité artificielle) à une réinitialisation périodique de la masse volumique à l'aide de noyaux d'interpolation corrigés par *moindres carrés mobiles* (MLS). Ces différentes méthodes et les algorithmes nécessaires à leur mise en œuvre sont maintenant détaillés.

4.4.1 Correction de la vitesse « XSPH »

Initialement proposée par Monaghan [161], cette technique a depuis été très largement reprise pour la modélisation d'écoulements à surface libre ou pour des écoulements à grande vitesse. Elle

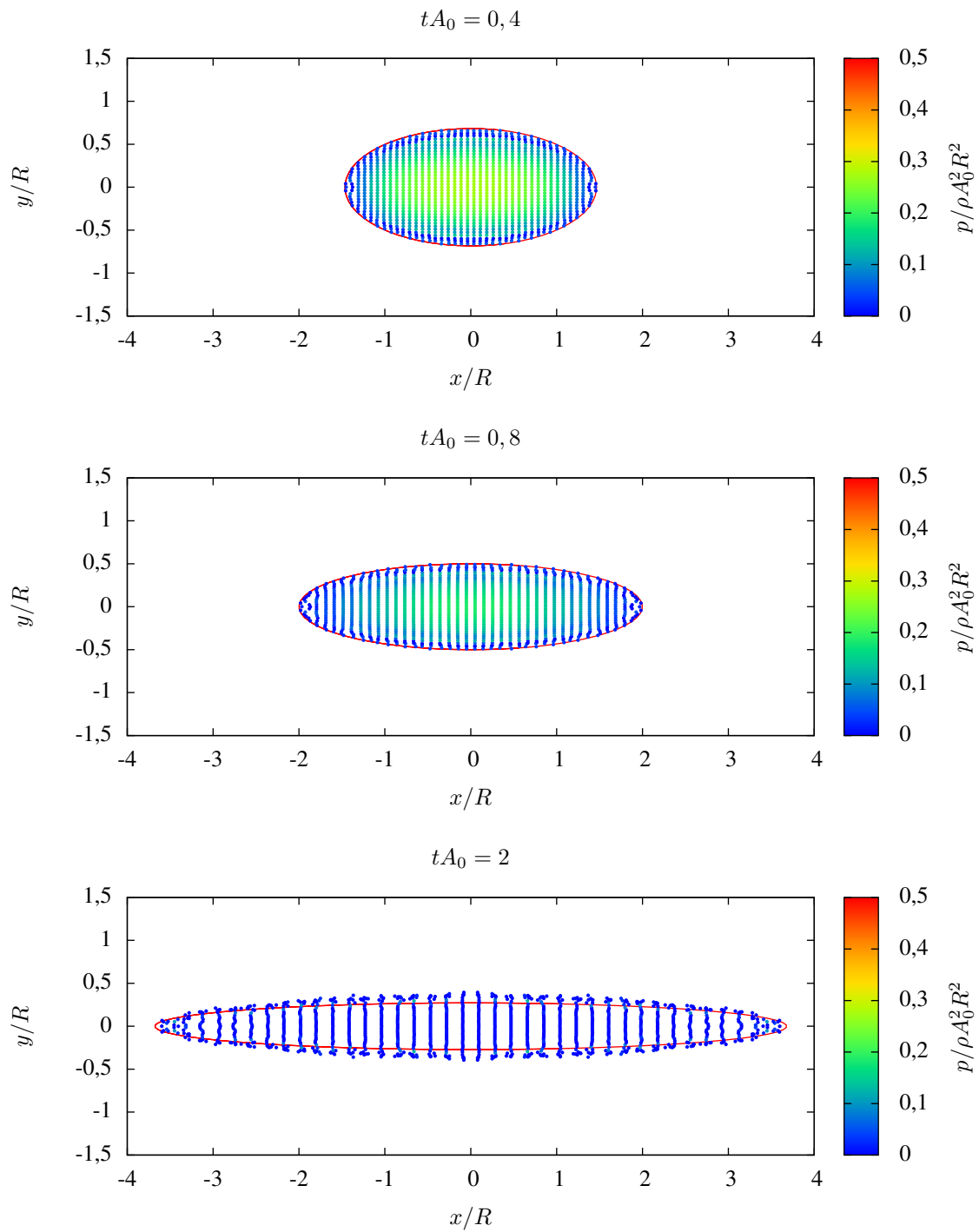


FIGURE 4.7 – Évolution de la goutte simulée à l'aide du modèle SPH de Colagrossi [35]. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.



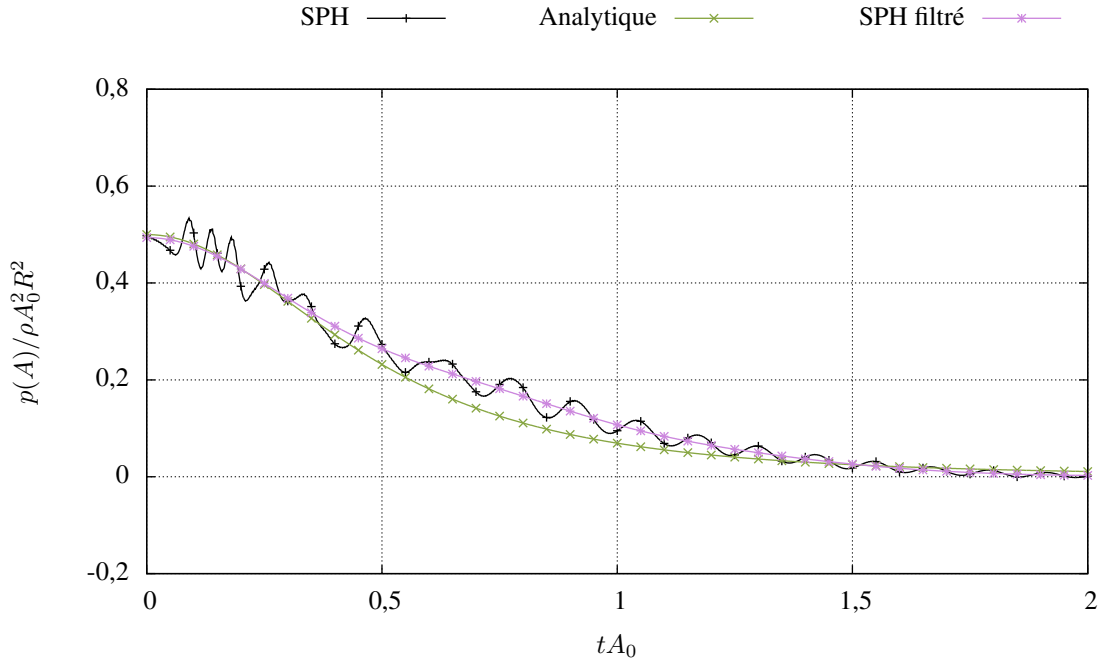


FIGURE 4.8 – Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Colagrossi [35]. Comparaison entre la valeur calculée par SPH (en noir), la solution analytique (en vert) et la solution SPH filtrée (les fréquences $f > 3,5A_0$ ont été supprimées, en violet).

consiste à modifier la vitesse de déplacement des particules de la manière suivant :

$$\mathbf{u}_i^* = \mathbf{u}_i + \epsilon_{\text{XSPH}} \sum_j \frac{m_j}{\rho_j + \rho_i} (\mathbf{u}_j - \mathbf{u}_i) \eta_\epsilon^0(r_{ij}) \quad (4.8)$$

Cela a pour effet d'éviter l'*inter-pénétration* entre particules. La vitesse étant moyennée dans le voisinage de chaque particule, celles-ci conservent effectivement une répartition plus ordonnée (figure 4.7). Cette vitesse corrigée est également employée pour l'équation de conservation de la masse (équation (3.49)), mais pas pour l'équation de conservation de la quantité de mouvement (quand elle contient un modèle de viscosité réelle ou artificielle). En XSPH, une particule a donc deux vitesses, une vitesse associée à ses déplacements \mathbf{u}_i^* , et une vitesse de déplacement du fluide qu'elle représente \mathbf{u}_i . Le paramètre ϵ_{XSPH} définit l'intensité de la correction, il est couramment fixé à 0,5.

4.4.2 Viscosité artificielle

Comme dans de nombreuses méthodes numériques, une viscosité artificielle peut être ajoutée au modèle non-visqueux, dans le but d'améliorer la stabilité numérique du code de calcul. L'équa-

tion de conservation de la quantité de mouvement est alors modifiée, par ajout d'un terme Π_{ij} :

$$\frac{D\mathbf{u}_i}{Dt} = -\frac{1}{\rho_i} \sum_j (P_i + P_j + \Pi_{ij}) \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (4.9)$$

Il existe plusieurs formes de viscosité artificielle dans la littérature SPH. La plus courante, proposée par Monaghan [154] prend la forme suivante dans le cas de notre formulation [35] :

$$\Pi_{ij} = \begin{cases} -\alpha \mu_{ij} \frac{c_i + c_j}{\frac{1}{\rho_i} + \frac{1}{\rho_j}} & \text{si } (\mathbf{u}_i - \mathbf{u}_j) \cdot (\mathbf{x}_i - \mathbf{x}_j) < 0 \\ 0 & \text{sinon} \end{cases}, \quad (4.10)$$

avec c_i la vitesse du son de la particule i et μ_{ij} , un terme basé sur le déplacement relatif des particules :

$$\mu_{ij} = \epsilon \frac{(\mathbf{u}_i - \mathbf{u}_j) \cdot (\mathbf{x}_i - \mathbf{x}_j)}{r_{ij}^2 + 0,01\epsilon^2} \quad (4.11)$$

Le terme $0,01\epsilon^2$ placé au dénominateur a pour but d'éviter que le rapport μ_{ij} soit indéfini, ce qui arrive quand deux particules sont trop proches l'une de l'autre. α définit l'influence de cette viscosité artificielle ; pour ne pas introduire trop de dissipation, sa valeur est généralement choisie inférieure à 0,03. Colagrossi propose une alternative, inspirée des travaux de Balsara [8], qui n'introduit pas de viscosité aux endroits où on observe du cisaillement. Le terme μ_{ij} précédent est remplacé par :

$$\mu_{ij} = \epsilon \frac{k_j + k_i}{2} \frac{(\mathbf{u}_i - \mathbf{u}_j) \cdot (\mathbf{x}_i - \mathbf{x}_j)}{r_{ij}^2 + 0,01\epsilon^2}, \quad (4.12)$$

avec

$$k_i = \frac{|\nabla \cdot \mathbf{u}_i|}{|\nabla \cdot \mathbf{u}_i| + \sqrt{\mathbf{E}_i \mathbf{E}_i} + 10^{-4} \frac{c_i}{\epsilon}} \quad (4.13)$$

Cette variante a pour avantage de moins dissiper l'énergie cinétique que la forme initiale de Monaghan [154]. L'algorithme associé à ce modèle de viscosité est néanmoins plus complexe, en raison de la présence de l'invariant du tenseur des déformations \mathbf{E}_i : \mathbf{E}_i , qui s'écrit, pour un problème à l dimensions d'espace :

$$\mathbf{E}_i \mathbf{E}_i = \left(\sum_{\alpha=1}^l \sum_{\beta=1}^l \frac{1}{2} \left(\left\langle \frac{\partial \mathbf{u}^\beta}{\partial \mathbf{x}^\alpha} \right\rangle_i + \left\langle \frac{\partial \mathbf{u}^\alpha}{\partial \mathbf{x}^\beta} \right\rangle_i \right) \right)^2 \quad (4.14)$$

Numériquement, ce terme est approché de la manière suivante :

$$\mathbf{E}_i \mathbf{E}_i = \left(\sum_{\alpha=1}^l \sum_{\beta=1}^l \sum_j \frac{1}{2} \left((\mathbf{u}_i^\alpha + \mathbf{u}_j^\alpha) \nabla_\beta \eta_\epsilon^0(\mathbf{x}_{ij}) + (\mathbf{u}_i^\beta + \mathbf{u}_j^\beta) \nabla_\alpha \eta_\epsilon^0(\mathbf{x}_{ij}) \right) V_j \right)^2 \quad (4.15)$$



Le surcoût engendré par ce terme est assez important, mais la réduction de pertes d'énergie (par rapport aux autres modèles de viscosité) qu'il permet sur les simulations en justifie l'utilisation.

4.4.3 Réinitialisation de la masse volumique « MLS »

Ce traitement est basé sur la formule d'interpolation initialement employée dans SPH pour la simulation de fluides compressibles, dans laquelle le noyau a été corrigé par la méthode des *moindres carrés mobiles*, selon Belytschko *et al.* [12] :

$$\rho_i = \sum_j m_j \eta_\epsilon^{0*}(\mathbf{x}_{ij}) \quad (4.16)$$

avec le noyau corrigé calculé comme suit, en 2D :

$$\eta_\epsilon^{0*}(\mathbf{x}_{ij}) = (\boldsymbol{\beta}_i \cdot (1, x_i - x_j, y_i - y_j)) \eta_\epsilon^0(\mathbf{x}_{ij}) \quad (4.17)$$

Le vecteur de correction $\boldsymbol{\beta}_i$ est calculé de la manière suivante :

$$\boldsymbol{\beta}_i = \mathbf{A}_i^{-1} \cdot (1, 0, 0) \quad (4.18)$$

où la matrice \mathbf{A}_i , de taille 3×3 prend la forme :

$$\mathbf{A}_i = \sum_j \begin{bmatrix} 1 & (x_i - x_j) & (y_i - y_j) \\ (x_i - x_j) & (x_i - x_j)^2 & (x_i - x_j)(y_i - y_j) \\ (y_i - y_j) & (x_i - x_j)(y_i - y_j) & (y_i - y_j)^2 \end{bmatrix} \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (4.19)$$

On peut remarquer que seule la première colonne de la matrice inverse est nécessaire. Ce calcul est effectué analytiquement. Bien qu'optimisé, il occasionne tout de même un surcoût en temps de calcul non-négligeable. C'est pourquoi Colagrossi et Landrini [41] ont mené une étude paramétrique sur la période de réinitialisation et en comparant la conservation de l'énergie des différents résultats obtenus. Une fréquence de réinitialisation plus grande permet de mieux conserver l'énergie mais au prix des temps de simulation dégradés. A l'inverse, des réinitialisations trop éloignées entraînent une dégradation de la précision du champ de pression et des pertes d'énergie non-négligeables. Colagrossi et Landrini ont observé qu'une période de 20 itérations était suffisante pour améliorer les résultats tout en conservant des durées de simulations raisonnables (dans le cas de la rupture de barrage que nous étudions au chapitre 5). Ce traitement ne permet pas seulement de s'assurer que seules les particules dont le domaine n'est pas totalement recouvert (donc à la surface libre) vont être à la masse volumique de référence. Il permet donc d'avoir un champ de pression plus régulier. Cette réinitialisation a l'avantage de ne pas introduire d'artifice dans le calcul, il s'agit bien là simplement d'un lissage numérique bien cohérent avec le reste de la méthode.

L'idée avait d'ailleurs déjà été retenue par Dilts [59, 60] qui utilisait les deux autres colonnes de

la matrice A pour corriger le gradient du noyau et améliorer la précision des approximations dans les équations de son modèle MLSPH (Moving Least Squares Particle Hydrodynamics). Le gain en précision n'est pas discutable dans son cas, mais contrairement à une réinitialisation périodique, la méthode MLSPH requiert l'inversion d'une matrice 3×3 (en 2D) ou 4×4 (en 3D) pour chaque particule et à chaque itération (ou sous-itération). Les temps de calcul associés à cette variante de SPH sont prohibitifs et d'autres méthodes de correction, ou renormalisation, moins coûteuses, permettent d'obtenir des résultats équivalents.

La figure (4.7) montre l'évolution de la goutte par le modèle de Colagrossi [35] associant les méthodes suivantes :

- Formulation SPH symétrique « $\sigma = 1$ ».
- Correction de la vitesse XSPH avec $\epsilon_{\text{XSPH}} = 0,5$.
- Viscosité artificielle avec $\alpha = 0,03$.
- Réinitialisation de la masse volumique par MLS tous les 20 pas de temps (fixé ici à $\Delta t = 1 \times 10^{-3}s$).
- Correction contre l'instabilité en tension présentée juste après.

Le champ de pression est, comme attendu, plus régulier que dans le cas du modèle simple. L'évolution de la pression au centre de la goutte (figure 4.8) montre d'ailleurs moins d'oscillations, et de plus faible amplitude. L'utilisation conjointe de la correction XSPH et de la viscosité artificielle permet également d'atteindre un étirement extrême de la goutte (à $tA_0 = 2$) sans que la simulation ne montre d'instabilités ou que les particules ne s'ordonnent selon des structures complexes et incohérentes comme celles observées sur les figures (4.4) et (4.5).

4.4.4 Correction de l'instabilité en tension

Communément nommée *tensile instability*, ce phénomène se manifeste, entre autres, par le regroupement de particules par petits lots solidaires, dans les zones du domaine fluide en compression. Ce regroupement de particules n'est évidemment pas physique. D'après Sweple *et al.* [219], une condition d'apparition est :

$$\Delta\eta_\epsilon^0(\mathbf{x}_{ij})\sigma_i > 0, \quad (4.20)$$

où σ_i représente le tenseur des contraintes de la particule i . Ainsi, Sweple *et al.* identifient les zones d'apparition de l'instabilité en fonction de la position des particules avoisinantes et donc de la forme de la dérivée seconde du noyau (figure 4.9). Cette instabilité est difficilement détectable dans les applications hydrodynamiques à surface libre, en raison des fortes valeurs de pression qui sont généralement rencontrées, elle apparaît plus fréquemment dans les problèmes de mécanique des solides déformables [30, 105].

Depuis que cette instabilité a été identifiée, quelques solutions ont été proposées. Monaghan [157] propose notamment d'ajouter à nouveau un terme dans l'équation de la conservation de la quantité de mouvement (équation (4.21)), modélisant une force répulsive. De cette manière, les



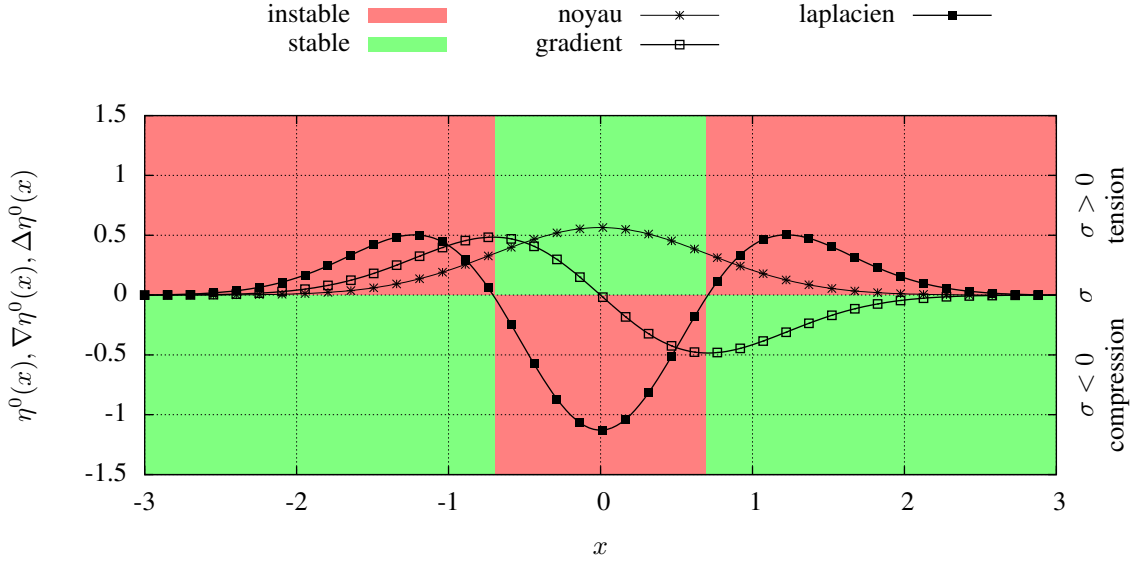


FIGURE 4.9 – Représentation graphique du critère d'apparition de l'instabilité selon Swegle et al. [219], en fonction des contraintes subies par la particule, et de la position de ses voisines.

particules sont maintenues éloignées les unes des autres.

$$\frac{D\mathbf{u}_i}{Dt} = -\frac{1}{\rho_i} \sum_j (P_i + P_j + \Pi_{ij} + R_{ij} f_{ij}^4) \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (4.21)$$

Le terme $R_{ij} f_{ij}^4$ est composé de deux facteurs, le premier, R_{ij} a pour but d'identifier si les deux particules sont en tension ou compression :

$$R_{ij} = \begin{cases} 0,02(P_i + P_j) & \text{si } P_i > 0 \text{ et } P_j > 0 \\ -0,2(P_i + P_j) & \text{si } P_i \leq 0 \text{ et } P_j \leq 0 \\ -0,2(P_i) & \text{si } P_i \leq 0 \text{ et } P_j > 0 \\ -0,2(P_j) & \text{si } P_i > 0 \text{ et } P_j \leq 0 \end{cases}, \quad (4.22)$$

où les paramètres 0,2 et 0,02 ont été proposés par Monaghan [157] et sont repris couramment dans la littérature [35]. Le second facteur f_{ij} est une mise à l'échelle de cette force répulsive et l'augmente quand les deux particules i et j se rapprochent :

$$f_{ij} = \frac{\eta_\epsilon^0(\mathbf{x}_{ij})}{\eta_\epsilon^0(h)}, \quad (4.23)$$

avec h , la distance interparticulaire initiale. Cette correction est efficace, sous réserve que le champ de pression soit assez lisse (ce qui est assuré ici par la réinitialisation MLS de la masse volumique). Une adaptation de cette force répulsive artificielle aux écoulements visqueux a été proposée par Meleán *et al.* [148], qui calculent le facteur R_{ij} en fonction des valeurs du tenseur des contraintes de chaque particule. Une illustration de l'effet de cette correction est proposée en annexe (voir

chapitre B).

L'ensemble des techniques présentées dans cette section sont actuellement couramment employées par les numériciens qui s'intéressent à l'hydrodynamique à surface libre. L'utilisation de la correction XSPH, de viscosité artificielle et le lissage de la masse volumique par MLS rendent le modèle SPH plus stable numériquement et plus précis. Les différents codes actuels ne diffèrent pas vraiment les uns des autres à ce niveau. L'utilisation de viscosité artificielle, en particulier, est quasiment systématique. La correction XSPH est très souvent utilisée quand aucune autre technique n'a été implémentée pour lisser le champ de masse volumique en modélisation faiblement compressible. Colagrossi et Landrini [41] montrent que la correction XSPH couplée à l'équation de conservation de la masse permet de maintenir la continuité des interfaces fluides en modélisation multiphasique.

Enfin, la réinitialisation de la masse volumique par MLS est une façon de régulariser le champ de pression. Les techniques de renormalisation du noyau et de son gradient [15, 16, 30, 70, 173] permettent aussi d'éviter l'apparition d'instabilités liées à l'interpolation, cela évite généralement l'apparition de bruit numérique ou de l'instabilité en tension. Molteni et Colagrossi [151] ajoutent des termes dissipatifs aux deux équations de conservation et suppriment ainsi d'éventuelles oscillations.

Dans toutes ces modifications du schéma standard, l'expérience numérique et la calibration de nombreux paramètres sont nécessaires. En effet, l'intensité de la correction XSPH, le coefficient de viscosité artificielle, le pas de réinitialisation MLS, les coefficients de termes dissipatifs [151] sont autant de valeurs à fixer avec précaution. C'est pourquoi la création d'un code SPH requiert encore maintenant une étude approfondie de tous ces paramètres et qu'encore peu d'industriels ne font confiance à des résultats pouvant varier significativement selon la configuration du modèle.

4.5 Modélisation par SPH : schéma de Ferrari *et al.* [75]

4.5.1 Flux Rusanov

L'approche développée ici a été proposée par Ferrari *et al.* [75], qui stabilise le schéma standard SPH par simple ajout d'un flux *Rusanov* à l'équation de conservation de la masse :

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \left(\mathbf{u}_{ij} + \mathbf{n}_{ij} \left(\frac{c_{ij}}{\rho_i} (\rho_j - \rho_i) \right) \right) \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j}, \quad (4.24)$$

avec $c_{ij} = \max(c_i, c_j)$, où c_i est le vitesse du son effective de la particule i , à déterminer à partir de l'équation d'état employée. Avec l'équation (3.13), on a :

$$c_i^2 = \frac{\partial P_i}{\partial \rho_i} = 7 \frac{c_0^2 \rho_0}{7} \left(\frac{\rho_i^{7-1}}{\rho_0^7} \right) = c_0^2 \left(\frac{\rho_i}{\rho_0} \right)^6 \quad (4.25)$$



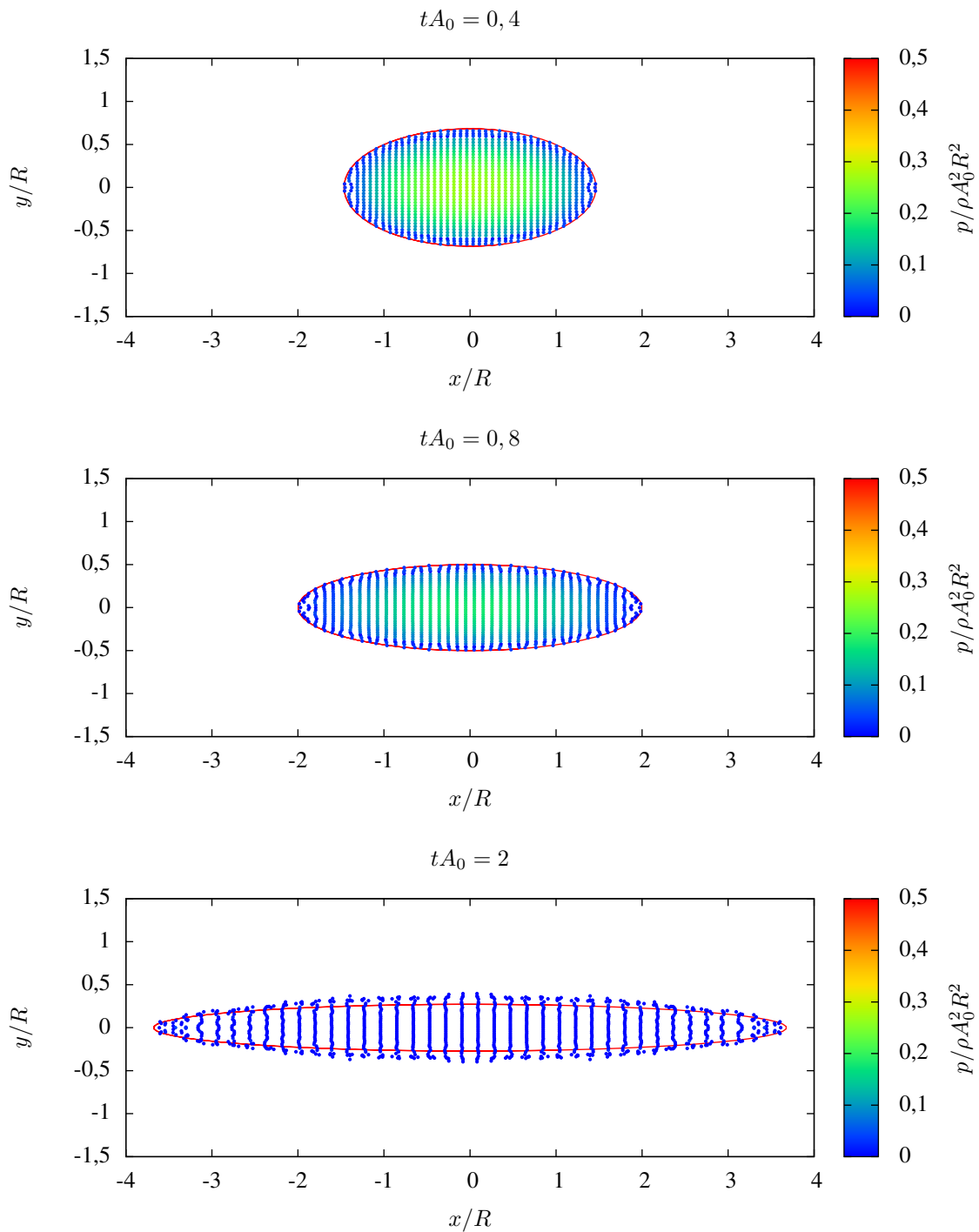


FIGURE 4.10 – Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari et al. [75]. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.

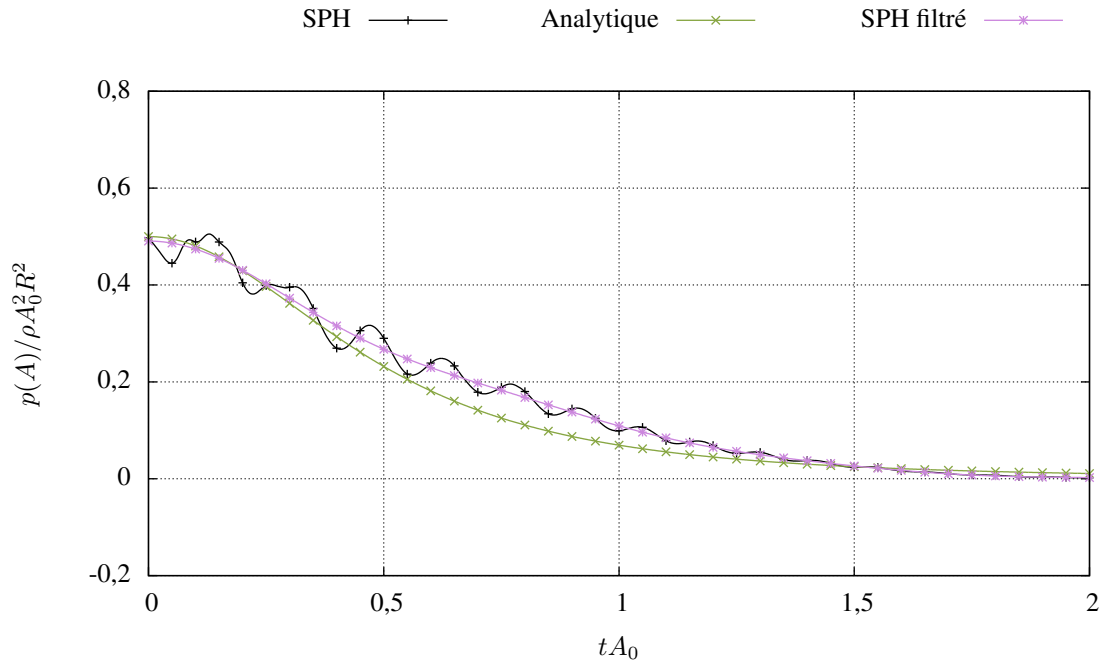


FIGURE 4.11 – Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Ferrari et al. [75]. Comparaison entre la valeur calculée par SPH (en noir), la solution analytique (en vert) et la solution SPH filtrée (les fréquences $f > 3,5A_0$ ont été supprimées, en violet).

donc

$$c_i = c_0 \left(\frac{\rho_i}{\rho_0} \right)^3 \quad (4.26)$$

\mathbf{n}_{ij} désigne le vecteur unitaire allant de la particule i vers la particule j :

$$\mathbf{n}_{ij} = \frac{\mathbf{x}_j - \mathbf{x}_i}{r_{ij}} \quad (4.27)$$

L'équation de conservation de la quantité de mouvement reste, quant à elle, inchangée.

L'idée consistant à ajouter un flux aux équations d'Euler, en vue d'en stabiliser l'intégration, s'inscrit dans une démarche plus large visant à exploiter des outils mathématiques et numériques issus d'autres méthodes. En l'occurrence, les méthodes des Volumes Finis et l'utilisation de solveurs de Riemann ont inspiré de nombreux travaux de recherche sur SPH. C'est ainsi que sont apparus les nouveaux schémas *Godunov-SPH* (par opposition aux schéma standards dits *Lax-Wendroff-SPH*). Les travaux de Inutsuka [100] sont à l'origine de cette avancée. De nombreuses variantes ont suivi, d'abord appliquées, encore un fois, à des problèmes en astrophysique [150], puis à des problèmes de mécanique des fluides compressibles [76, 101, 25] ou de mécanique des solides [177, 176]. Ainsi Monaghan [156] a proposé une nouvelle forme de viscosité artificielle s'inspirant des solveurs de Riemann. Les travaux de Vila [232], Moussa [165] ont ensuite étendus ces nouveaux modèles aux problèmes quasi-incompressibles à surface libre. Roubtsova et Kahawita [196] appliquent *Godunov-SPH* à une rupture de barrage, l'intérêt majeur étant que le solveur de



Riemann dispense de l'utilisation de viscosité artificielle. Guilcher *et al.* [88] appliquent le schéma *Godunov-SPH* à la modélisation d'une houle régulière : il montre d'ailleurs que ni le schéma *Lax-Wendroff-SPH* original de Monaghan (XSPH+viscosité artificielle [155]), ni leur nouveau schéma *Godunov-SPH* ne permettent de modéliser correctement une houle régulière sans qu'elle ne diverge progressivement de la théorie ; la renormalisation du noyau (voir §4.6.1) corrige ensuite ces erreurs de phase. L'avantage de l'approche *Godunov-SPH* employée par Guilcher *et al.* [88] est la conservation de l'énergie (point mis en défaut par Ferrari *et al.* [75]), au prix de temps de calcul quatre fois supérieurs (Oger, comm. pers.) dus :

- à la résolution du problème de Riemann exact ou approché, à l'interface entre chaque paire de particules,
- à l'intégration d'une équation supplémentaire pour la masse des particules, qui peut varier. Sur ce point, la méthode *Godunov-SPH* peut-être vue comme une méthode Arbitrairement Lagrange-Euler,
- à l'emploi d'autres techniques de montée en ordre : MUSCL...

Bien que les gains en robustesse et précision apportés par cette approche ne soient pas discutables, sa mise en œuvre n'est pas aisée et le schéma global perd en simplicité. Les domaines d'applications ne sont d'ailleurs pas encore aussi variés que pour le schéma *Lax-Wendroff-SPH*. On peut toutefois noter les tentatives récentes d'extension du modèle *Godunov-SPH* à des écoulements polyphasiques, par fraction de volume [86] ou non [123], ou l'extension du concept de flux entre particules à des flux parois-particules pour une modélisation plus précise des conditions aux limites [142, 54].

A contrario, l'approche proposée par Ferrari *et al.* [75] n'entraîne que très peu de modifications du code de calcul et s'interprète assez facilement. Ainsi le flux *Rusanov* (équation (4.24)) atténue clairement les variations de masse-volumique qui pourraient apparaître dans le domaine fluide entre particules voisines. Ce terme suffit donc à stabiliser le schéma (plus de viscosité artificielle) et à régulariser le champ de masse volumique (rôle joué, dans le cas du modèle de Colagrossi, par la correction XSPH et la réinitialisation de la masse volumique MLS).

La figure 4.10 montre l'évolution de la goutte étirée simulée par le modèle SPH standard combiné au flux *Rusanov* de Ferrari *et al.* [75] (équation (4.24)). Le champ de pression présente la même régularité que pour le modèle de Colagrossi [35]. Dans les deux cas, les particules s'arrangent en lignes verticales condensées à la fin de la simulation ($tA_0 = 2$), principalement en raison de la répartition cartésienne initiale (voir §4.6.2 sur l'effet de l'initialisation de la position des particules). L'évolution de la pression calculée au point A (figure 4.11) montre moins d'oscillations que dans le cas du modèle de Colagrossi [35]. Les deux modèles fournissent une surestimation de cette pression par rapport à la solution analytique, qui s'explique par la faible résolution employée.

4.5.2 Convergence du schéma

L'efficacité du schéma proposé par Colagrossi [35] par rapport au schéma standard a été démontrée, ce qui justifie que les différentes corrections proposées soient maintenant couramment adoptées. L'approche de Ferrari *et al.* [75], plus récente, permet de supprimer un certain nombre de paramètres dépendant du problème traité. Il reste toutefois à vérifier sa validité sur différents écoulements. Pour le cas de la goutte étirée, en suivant les paramètres de discrétisation proposés par Colagrossi [35], nous avons vérifié la convergence du modèle *Rusanov*-SPH. Ainsi, en complément de la discrétisation $R/\epsilon = 14,7$ présentée jusqu'à maintenant, deux autres cas ont été simulés avec $R/\epsilon = 29,4$ et $R/\epsilon = 58,8$, correspondant respectivement à 4800 et 19200 particules environ. Les simulations ont été réalisées avec le même schéma prédicteur-correcteur que les précédentes, mais avec un pas de temps $\Delta t\omega = 5 \times 10^{-4}$ pour la résolution $R/\epsilon = 29,4$ et $\Delta t\omega = 2,5 \times 10^{-4}$ pour $R/\epsilon = 58,8$. Les figures 4.12 et 4.13 montrent l'évolution des champs de particules associés à ces deux discrétisations, on peut noter qu'à haute résolution (figure 4.13), le contour de la goutte semble mieux reproduit que pour les plus faibles résolutions. L'évolution de la pression au point A a été calculée pour les trois résolutions, les résultats peuvent être comparés sur la figure 4.14 : aucun gain en précision significatif ne peut être observé. Cela s'explique par la déformation que subit le champ de particules et à au choix d'une grille cartésienne pour l'initialisation de leur position. Cela conduit, quelque soit la résolution à la formation de rangées verticales de particules très proches. L'espacement horizontal entre deux rangées augmente progressivement, si bien que le support du noyau d'interpolation ne contient plus suffisamment de particules pour assurer une approximation correcte des gradients. Ce problème peut être corrigé de plusieurs façons :

- En corrigeant le noyau d'interpolation directement [15, 16, 30, 70, 173], ce sont les techniques de *renormalisation*, développées après.
- En faisant évoluer le pas d'interpolation ϵ de chaque particule, au cours du temps, de manière à maintenir un nombre de particules constant dans le domaine d'interpolation [212]. Ou en utilisant des noyaux anisotropes par transformation selon chaque dimension, le pas scalaire ϵ est alors remplacé par une matrice [175]. Ces deux approches s'inscrivent dans la famille des modèles SPH à pas variables. On y distingue d'ailleurs les méthodes pour lesquelles ϵ évolue de façon continue, et d'autres pour lesquelles des sauts en ϵ peuvent être présents. Ces dernières se rapprochent des méthodes à raffinement de maillage [22, 23, 147]. D'autres difficultés sont d'ailleurs rencontrées avec l'ensemble des modèles à ϵ variable, comme la prise en compte de termes en $\nabla\epsilon$ [169], les erreurs générées si ces termes sont négligés et une recherche des interactions entre particules plus coûteuse (voir §8.2)
- En maintenant la répartition des particules aussi homogène que possible, tout au long de la simulation, par réinitialisation périodique du champ de particule : c'est la technique de *remaillage* [26, 27, 29, 45], détaillée au paragraphe 6.1.1.
- En augmentant le rapport $\kappa = \frac{\epsilon}{h}$ de manière à prendre en compte plus de particules pour chaque somme tout au long de la simulation. Cette solution est rarement retenue car son



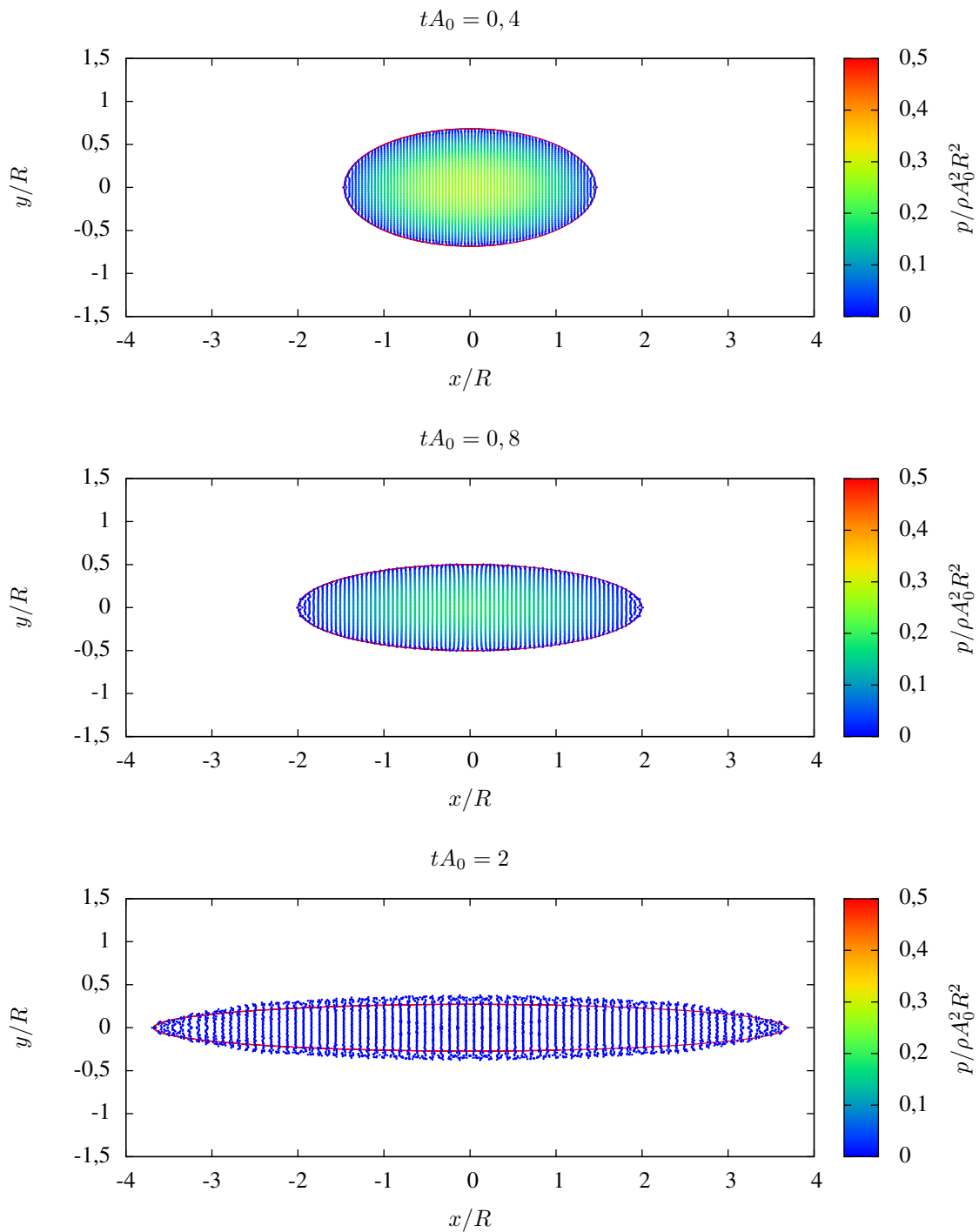


FIGURE 4.12 – Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari et al. [75], avec $R/\epsilon = 29,4$. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.

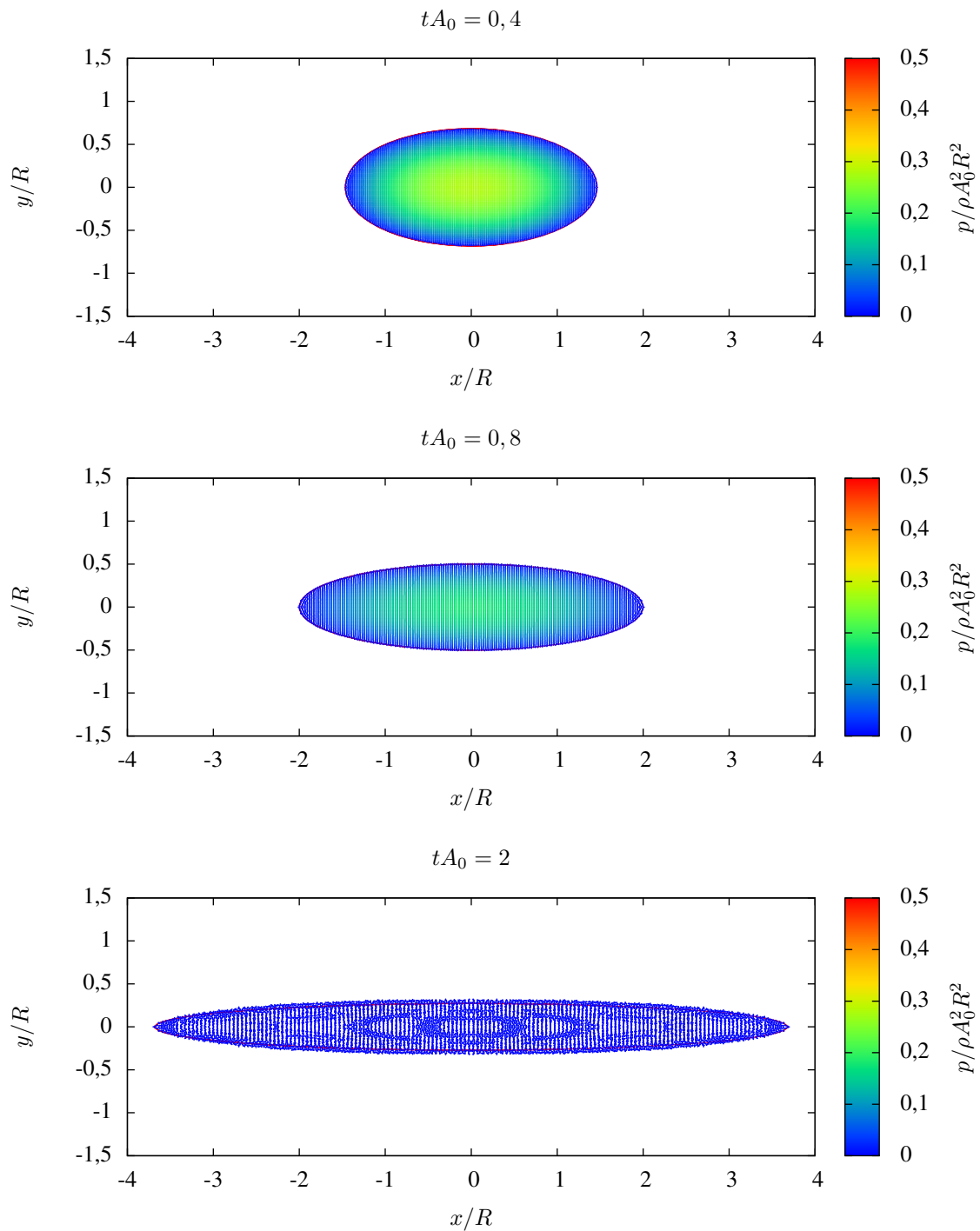
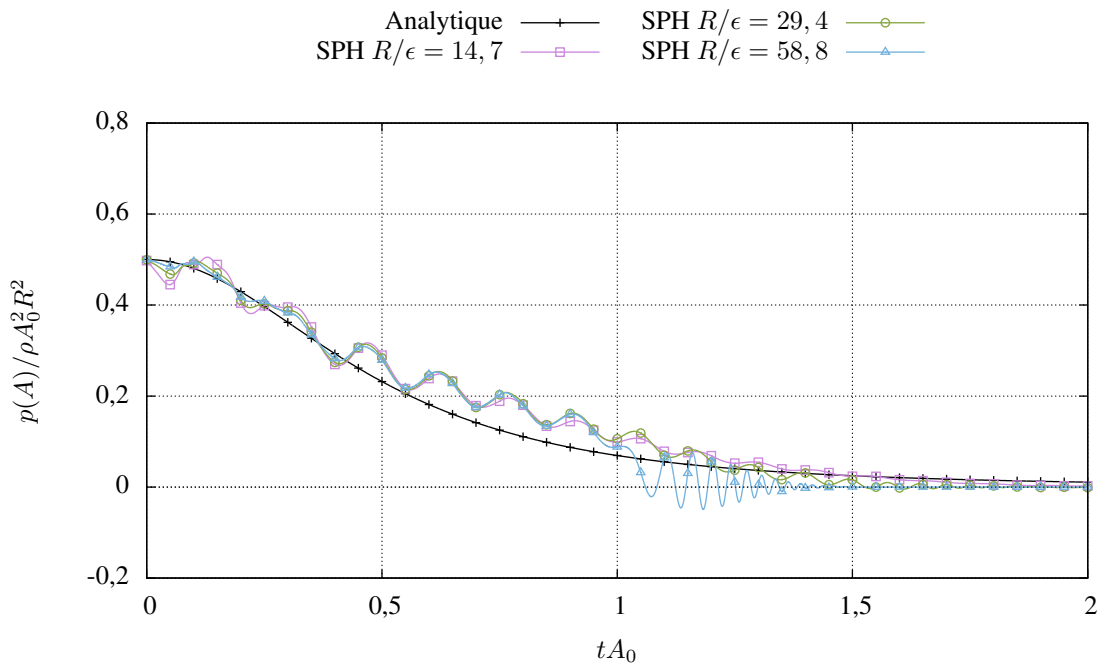
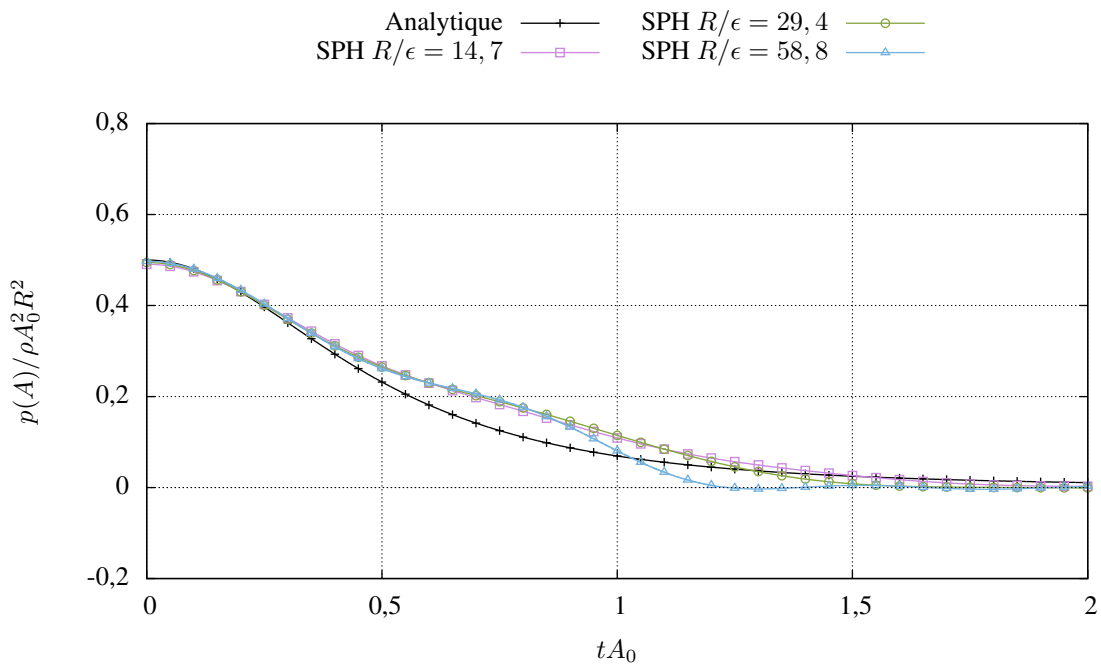


FIGURE 4.13 – Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari et al. [75], avec $R/\epsilon = 58,8$. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.





(a) Solution SPH sans filtrage.



(b) Solution SPH dont les fréquences $f > 3,5A_0$ ont été supprimées.

FIGURE 4.14 – Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Ferrari et al. [75]. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).

impact sur les temps de calcul est très important.

- En utilisant des points différents pour l'interpolation des différentes grandeurs. Ce sont les méthodes dites *duales* [230, 192].
- En choisissant une autre initialisation des particules, soit par perturbation aléatoire de la grille cartésienne, soit par choix d'une autre grille ne présentant pas de symétries, ou plus adaptée au problème posé.

4.6 Amélioration de la précision

4.6.1 Normalisation du noyau et renormalisation des gradients

Les opérateurs-intégrales présentés dans la partie précédente permettent théoriquement d'approcher la quantité interpolée à l'ordre 2 sous réserve que les particules soient maintenues sur une grille cartésienne. Cela n'est évidemment pas le cas dans la majorité des écoulements considérés ici. Cela a pour effet que les conditions établies pour la construction de l'opérateur-intégrale d'interpolation (équations (2.17) et (2.18)) ou d'approximation de dérivées ne sont plus vérifiées.

Correction du noyau

L'opérateur d'interpolation (2.43) peut fournir des résultats très imprécis si la condition de normalisation (équation (2.17)) n'est pas respectée. L'emploi d'un noyau normalisé peut permettre de restaurer cette condition [16] :

$$\mathcal{L}_{norm}^0 f(\mathbf{x}_i) = \frac{\sum_j f(\mathbf{x}_j) \eta_\epsilon^0(\mathbf{x}_{ij}) V_j}{\sum_j \eta_\epsilon^0(\mathbf{x}_{ij}) V_j} \quad (4.28)$$

Cette formule permet une interpolation à l'ordre 1 quelque soit la répartition des particules. D'autres méthodes permettent d'atteindre l'ordre 2 (la correction MLS vue précédemment, la méthode de Bonet et Lok [16] permettant de restaurer à la fois les conditions (2.17) et (2.18)), mais leur mise en œuvre est souvent plus complexe et moins robuste. L'opérateur normalisé (4.28) n'intervient pas dans le code de simulation mais il est d'une grande utilité dans le code de post-traitement que nous utilisons pour l'exploitation des résultats.

Ainsi, l'opérateur (4.28) a été employé jusqu'ici pour l'évaluation de la pression au centre de la goutte étirée. Pour noter les effets de cette normalisation, prenons les mesures de vitesse aux points B et C (figure 4.1). La figure 4.15 montre l'évolution de la composante horizontale de vitesse interpolée au point B avec ou sans normalisation de l'opérateur. Sans normalisation, le moment d'ordre 0 du noyau (équation (2.17)) montre des oscillations qui coïncident avec le passage des bandes verticales de particules observées dans les simulations précédentes. Avec normalisation, l'effet de ces variations du moment est supprimé, et la vitesse interpolée est en meilleur accord avec la solution analytique. L'augmentation du nombre de particules améliore la précision obtenue.



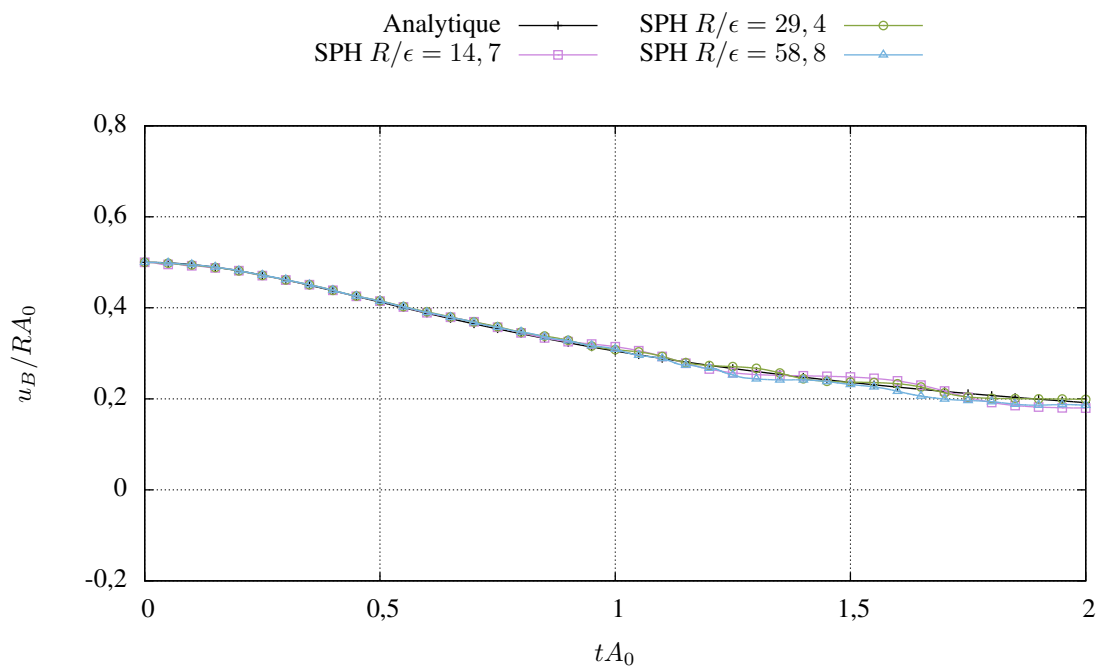
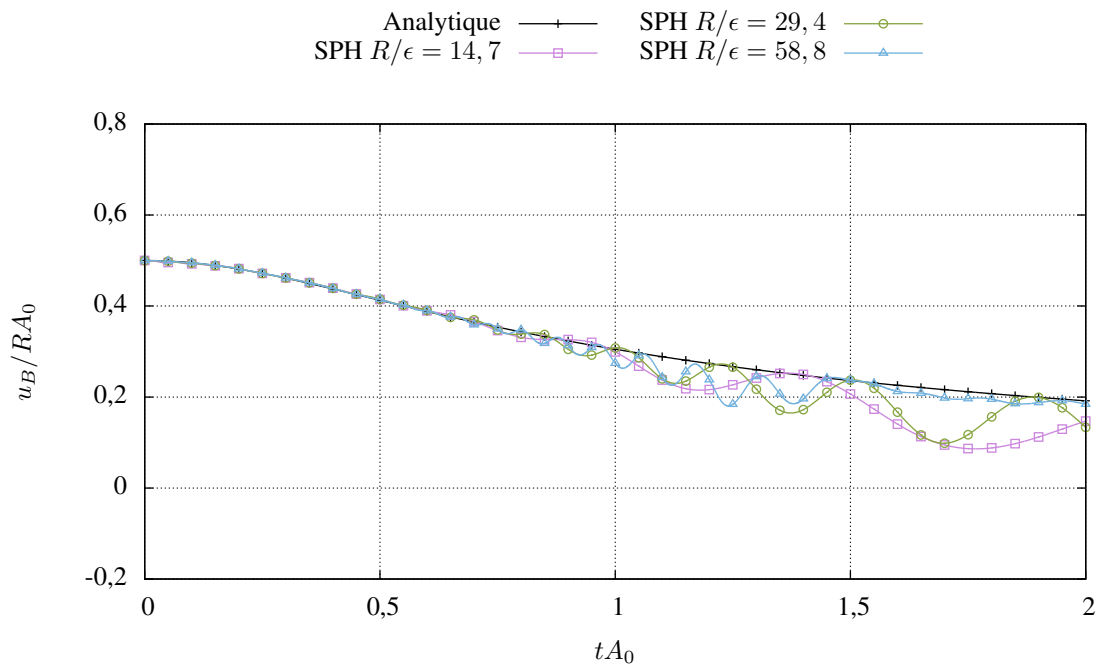
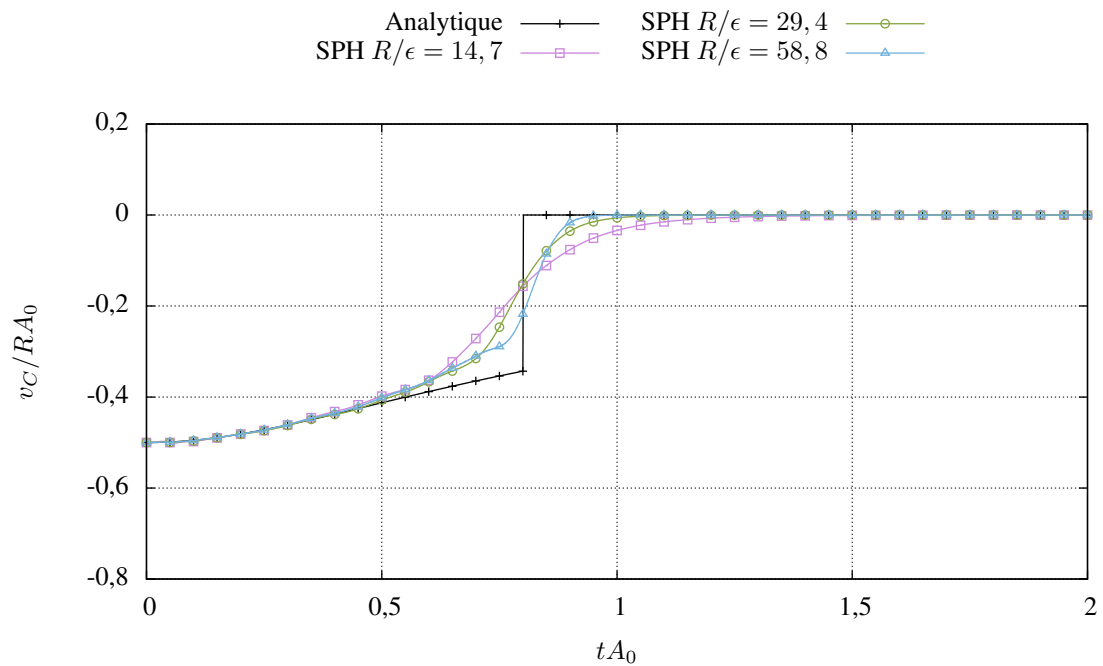
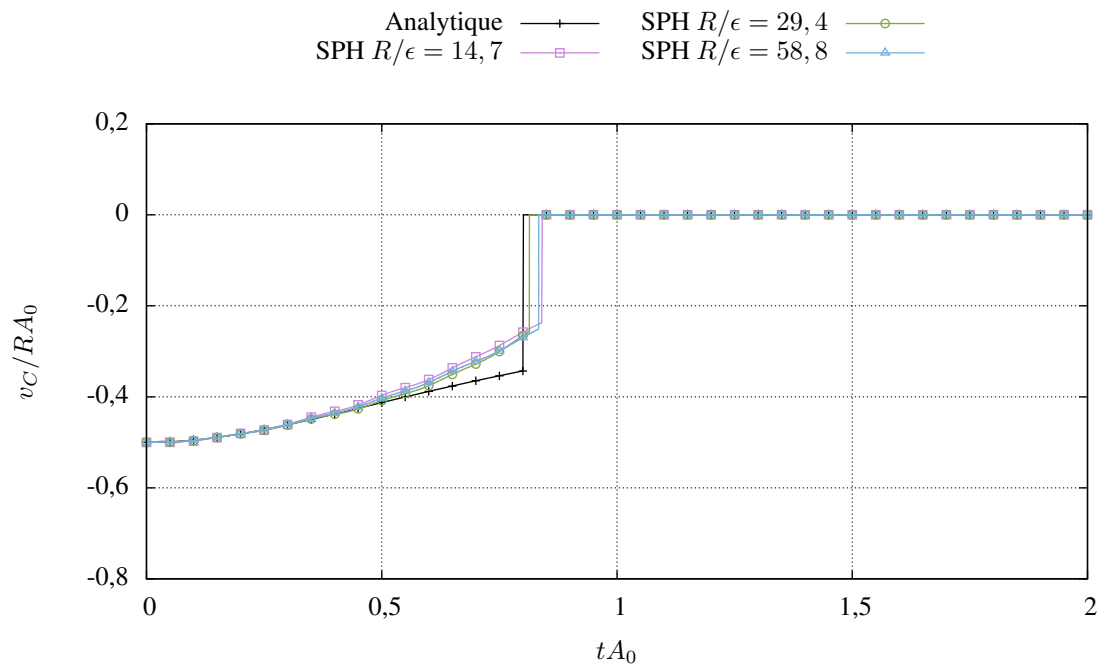


FIGURE 4.15 – Evolution de la composante horizontale de vitesse au point B (figure 4.1) par le modèle SPH de Ferrari et al. [75]. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).



(a) Sans normalisation.



(b) Avec normalisation.

FIGURE 4.16 – Evolution de la composante verticale de vitesse au point C (figure 4.1) par le modèle SPH de Ferrari et al. [75]. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).



La somme au dénominateur de l'opérateur normalisé tend vers 0 quand le nombre de particules sur le support du point d'interpolation diminue. Cela peut entraîner quelques difficultés quand il s'agit de mesurer des quantités, tout au long de la simulation, en des points qui ne sont pas en permanence entourés de fluide. C'est le cas notamment du point C, qui n'est plus dans le domaine fluide à partir de $tA_0 = 0,8$ (figure 4.2). Afin d'identifier si le point est, ou non, dans le domaine fluide, il paraît trivial d'employer le test suivant :

$$\begin{cases} \sum_j \eta_\epsilon^0(\mathbf{x}_C - \mathbf{x}_j)V_j > 0,5 & \text{alors } C \in \Omega \\ \sum_j \eta_\epsilon^0(\mathbf{x}_C - \mathbf{x}_j)V_j = 0,5 & \text{alors } C \in \partial\Omega \\ \sum_j \eta_\epsilon^0(\mathbf{x}_C - \mathbf{x}_j)V_j < 0,5 & \text{alors } C \notin \Omega \end{cases}, \quad (4.29)$$

où la valeur 0,5 correspond à une répartition homogène en particules sur la moitié du support du noyau. Le test précédent a été appliqué, en complément de l'opérateur normalisé, au calcul de la vitesse verticale au point C. La figure 4.16 représente l'évolution de cette vitesse pour les différentes résolutions, avec ou sans normalisation de l'opérateur. La solution analytique montre un saut vers 0 à $tA_0 = 0,8$ correspondant au passage de l'interface par le point C. Sans normalisation, le modèle SPH lisse ce saut sur un intervalle correspondant à la traversée complète du support du noyau (soit le diamètre 6ϵ) par l'interface. Tandis qu'en appliquant la normalisation et le test (équation (4.29)), la solution se rapproche de la valeur analytique avec, cependant, un léger décalage. Cette erreur vient de l'écart, entre l'interface simulée et l'interface exacte, observé sur les résultats du modèle SPH (figures 4.10, 4.12 et 4.13).

Renormalisation des gradients

De la même manière que pour l'interpolation, l'approximation discrète de gradients peut perdre en précision dès que la répartition des particules sur le support n'est plus suffisamment homogène. La technique de renormalisation, présentée ici, a été introduite par Vila [231], puis Lanson et Vila [116] et appliquées aux écoulements hydrodynamiques par Oger *et al.* [173]. D'autres variantes existent, encore une fois, comme la correction MLS des gradients par Belytschko *et al.* [12] et Dilts [59, 60], ou les corrections combinant normalisation du noyau et correction du gradient de Bonet et Lok [16] ou Fang *et al.* [70]. Ces différentes approches ont été testées et n'ont pas permis d'obtenir de meilleurs résultats sans sacrifier au passage la robustesse du code de calcul.

Le but est, comme pour la normalisation du noyau, de restaurer les valeurs des moments discrets d'ordre 1 du gradient de noyau, afin d'améliorer la précision des interpolations. Ces moments s'écrivent, pour une particule i , à partir de l'équation (2.11) et sous forme matricielle :

$$\sum_j \mathbf{x}_{ji} \otimes \nabla \eta_\epsilon^0(\mathbf{x}_{ij})V_j \quad (4.30)$$

En suivant, la méthodologie d'Eldredge *et al.* [66], ces moments prennent théoriquement pour

valeur la matrice identité de taille 2×2 en 2D et 3×3 en 3D. En pratique, ces moments prennent des valeurs différentes, en particulier au niveau de la surface libre si celle-ci est présente. On peut s'intéresser aux moments d'ordres supérieurs mais au prix de matrices de taille supérieure.

Une manière de restaurer ces conditions consiste à modifier le noyau par une matrice de renormalisation, de manière à imposer :

$$\sum_j \mathbf{x}_{ji} \otimes \mathbf{L}_i \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) V_j = \mathbf{I} \quad (4.31)$$

La matrice \mathbf{L}_i prend la forme suivante, en 2D :

$$\mathbf{L}_i = \left[\begin{array}{cc} \sum_j (x_j - x_i) \nabla_x \eta_\epsilon^0(\mathbf{x}_{ij}) V_j & \sum_j (x_j - x_i) \nabla_y \eta_\epsilon^0(\mathbf{x}_{ij}) V_j \\ \sum_j (y_j - y_i) \nabla_x \eta_\epsilon^0(\mathbf{x}_{ij}) V_j & \sum_j (y_j - y_i) \nabla_y \eta_\epsilon^0(\mathbf{x}_{ij}) V_j \end{array} \right]^{-1} \quad (4.32)$$

L'application de cette correction a fait l'objet d'une étude minutieuse par Oger *et al.* [173], qui suggère plusieurs formulations discrètes renormalisées pour les équations d'Euler modifiées. Il propose notamment l'utilisation de deux formes différentes pour l'équation de conservation de la quantité de mouvement, à appliquer aux particules proches de la surface libre (car elle comprend implicitement la condition à appliquer en pression) et aux particules éloignées de la surface séparément. La détection des particules de surface est d'ailleurs réalisée à l'aide des valeurs propres de la matrice de renormalisation. Ainsi, quelque soit la position de la particule par rapport à la surface libre, l'équation de conservation de la masse prend la forme suivante :

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \left(\mathbf{u}_{ij} + \mathbf{n}_{ij} \left(\frac{c_{ij}}{\rho_i} (\rho_j - \rho_i) \right) \right) \cdot \mathbf{L}_i \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (4.33)$$

Ensuite, si la valeur propre minimale de la matrice \mathbf{L}_i est inférieure à un critère défini par l'utilisateur, la particule est considérée comme une particule de surface libre. Ainsi, la quantité de mouvement de cette particule et de ces voisines est intégrée avec :

$$\frac{D\mathbf{u}_i}{Dt} = -\frac{1}{\rho_i} \sum_j (P_i + P_j) \mathbf{L}_i \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (4.34)$$

Pour les autres particules du domaine fluide, l'équation devient :

$$\frac{D\mathbf{u}_i}{Dt} = -\frac{1}{\rho_i} \sum_j (-P_i + P_j) \mathbf{L}_i \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (4.35)$$

initialement proposée par Randles et Libersky [191]. Oger *et al.* [173] montrent un gain en précision important par application de ces équations à différents écoulements à surface libre. Cependant, cette formulation hybride se montre, en pratique, très fragile pour différentes raisons :



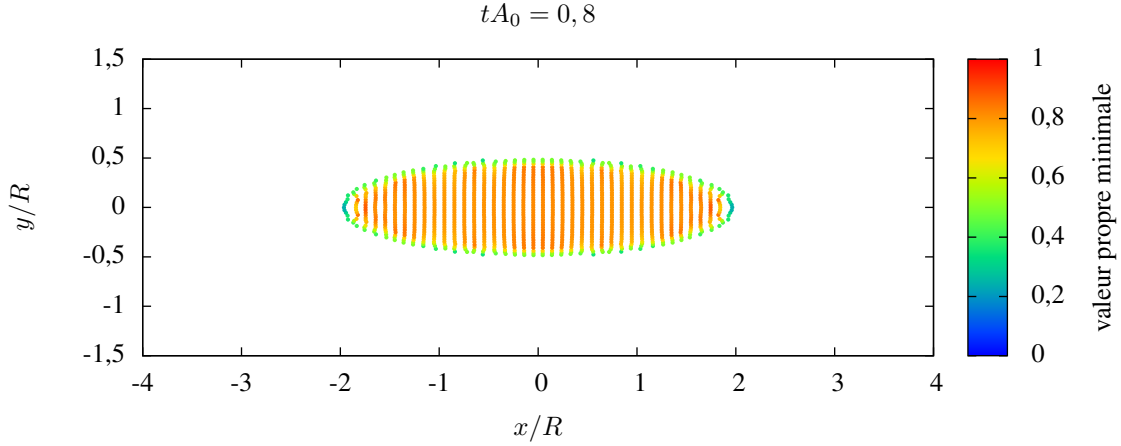


FIGURE 4.17 – Valeur propre minimale de la matrice de renormalisation de chaque particule, dans le cas de la goutte étirée.

- Les équations ne sont plus écrites sous forme symétrique. En particulier, l'équation de conservation de la masse implique :

$$\mathbf{u}_{ij} \cdot \mathbf{L}_i \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \neq \mathbf{u}_{ji} \cdot \mathbf{L}_j \nabla \eta_\epsilon^0(\mathbf{x}_{ji}) \quad (4.36)$$

ce qui rompt le principe de réciprocité des actions entre deux particules en interaction. De même, l'équation de conservation de la quantité de mouvement pour les particules éloignées de la surface libre s'écrit :

$$(-P_i + P_j) \mathbf{L}_i \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \neq (-P_j + P_i) \mathbf{L}_j \nabla \eta_\epsilon^0(\mathbf{x}_{ji}) \quad (4.37)$$

Le gain de précision apporté par la renormalisation justifie, selon Oger *et al.* [173] la rupture du principe de réciprocité. Nous n'avons pas observé de gains significatifs sans perte de robustesse.

- L'application de deux formulations différentes entraînent une augmentation importante de la complexité algorithmique du modèle.
- L'utilisation de la valeur propre minimale de la matrice est une approche fiable pour la détection des particules de la surface libre, comme le montrent Marrone *et al.* [143]. Cependant, le choix d'une valeur limite de passage d'une formulation à l'autre n'est pas trivial (figure 4.17).

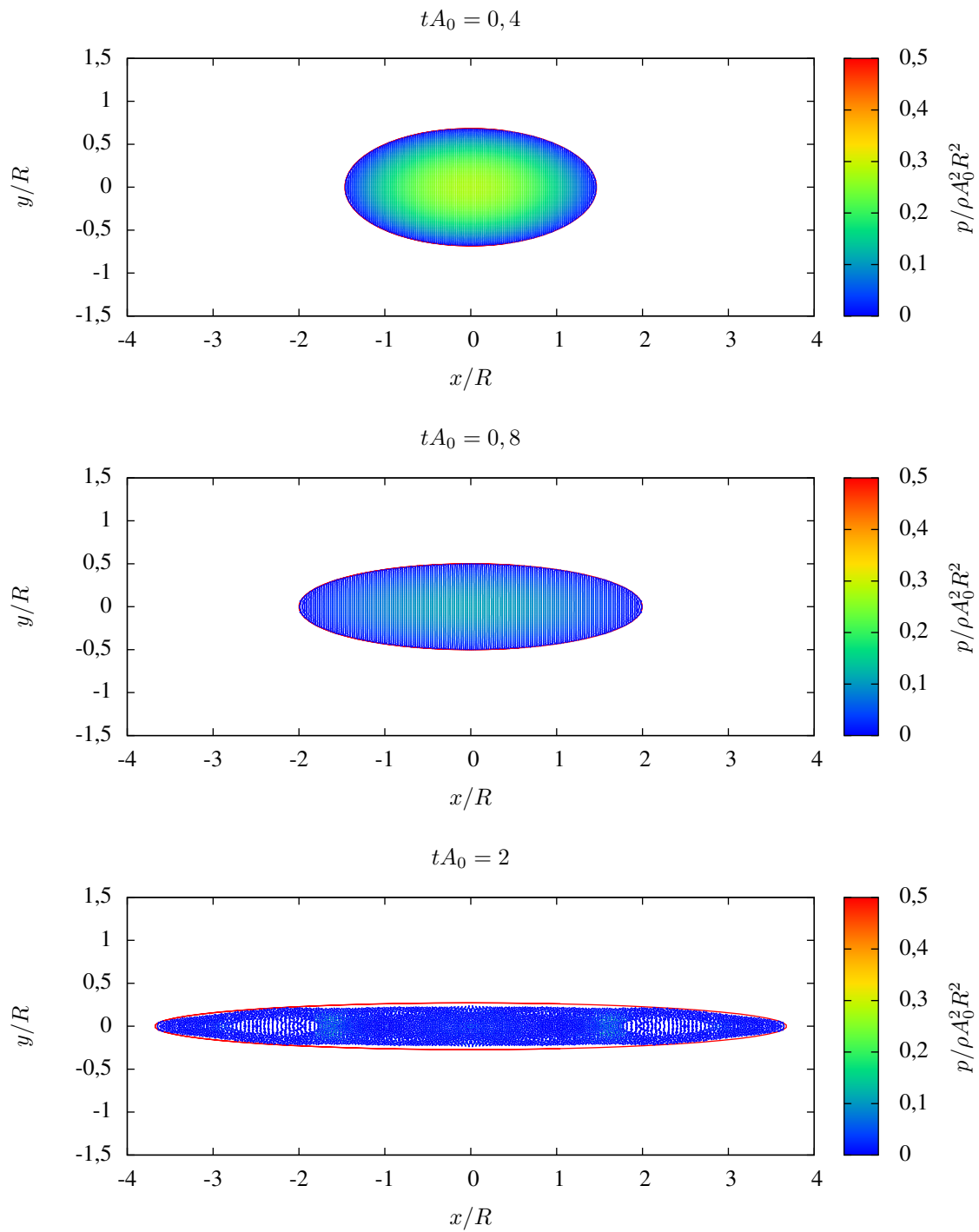


FIGURE 4.18 – Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari et al. [75] avec renormalisation des gradients et $R/\epsilon = 58,8$. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.



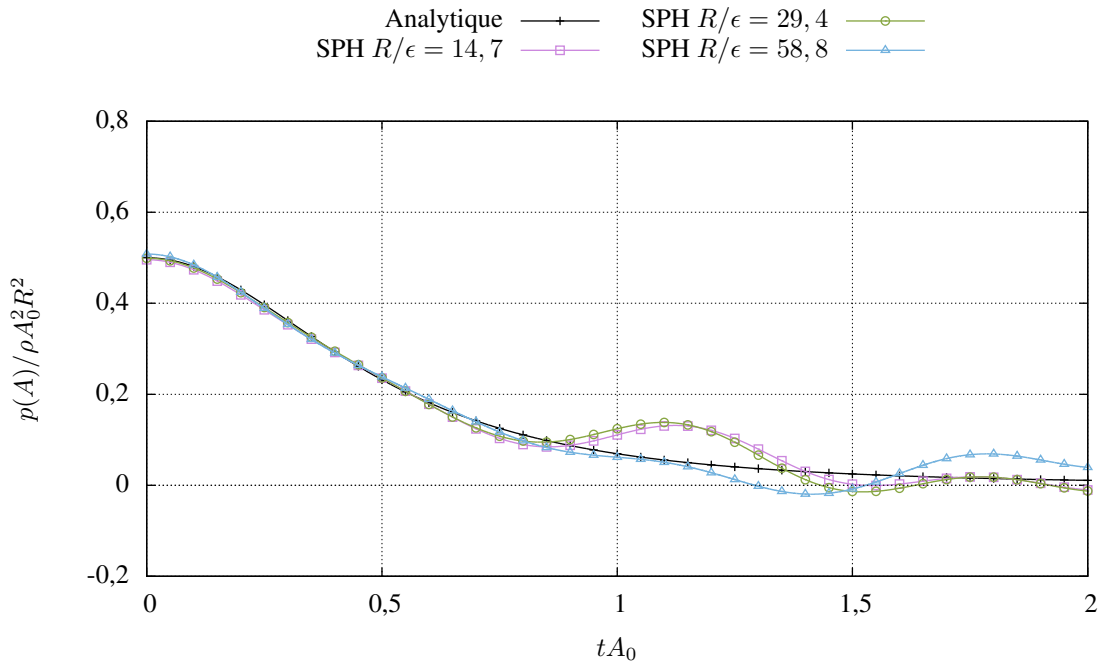


FIGURE 4.19 – Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Ferrari et al. [75] avec renormalisation des gradients. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).

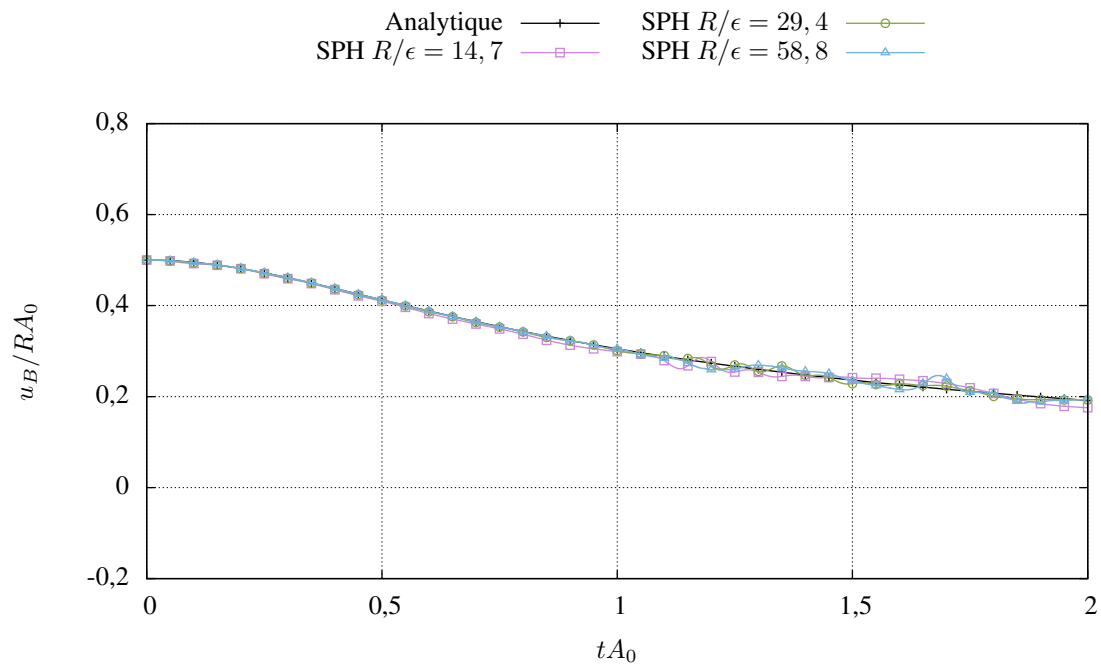


FIGURE 4.20 – Évolution de la composante horizontale de vitesse au point B (figure 4.1) par le modèle SPH de Ferrari et al. [75] avec renormalisation des gradients. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).

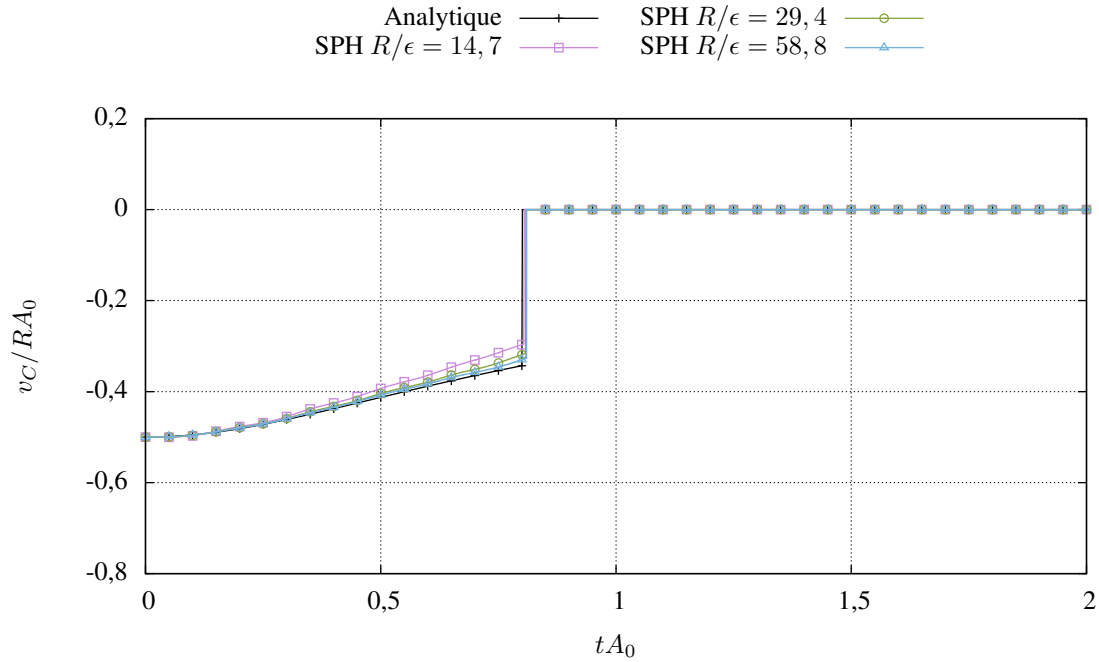


FIGURE 4.21 – Évolution de la composante verticale de vitesse au point C (figure 4.1) par le modèle SPH de Ferrari et al. [75] avec renormalisation des gradients. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).

Pour ces raisons, nous appliquons la correction de la manière suivante :

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \left(\mathbf{u}_{ij} + \mathbf{n}_{ij} \left(\frac{c_{ij}}{\rho_i} (\rho_j - \rho_i) \right) \right) \cdot \frac{\mathbf{L}_i + \mathbf{L}_j}{2} \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (4.38)$$

$$\frac{D\mathbf{u}_i}{Dt} = -\frac{1}{\rho_i} \sum_j (P_i + P_j) \frac{\mathbf{L}_i + \mathbf{L}_j}{2} \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (4.39)$$

qui maintient la symétrie du schéma initial. Cette forme est également employée dorénavant par Guilcher *et al.* [88]. Malgré tout, le calcul de la valeur propre minimale reste indispensable afin d'éviter toute tentative d'inversion de matrice singulière, en affectant systématiquement la matrice identité comme renormalisation (donc aucune correction) aux particules dont cette valeur propre est inférieure à 0,5. La figure 4.18 montre l'évolution du champ de particules, pour le cas $R/\epsilon = 58,8$ de la goutte, simulée avec le modèle *Rusanov*-SPH renormalisé. On peut noter qu'à l'étape $tA_0 = 2$, les particules forment une ellipse plus proche de la solution analytique que dans le cas où aucune correction n'est appliquée (figure 4.13). En revanche, et en raison des mêmes problèmes de raréfaction en particules, la pression mesurée en A (figure 4.19) montre des perturbations, quelque soit la résolution. Celles-ci sont dues à l'abandon de la correction quand certaines particules affichent une valeur propre minimale de matrice trop faible. Cela se produit quand les particules sont arrangées en lignes trop espacées horizontalement. Les vitesses en B (figure 4.20) et C (figure 4.21) sont moins affectées par ce phénomène, en raison de la normalisation vue au



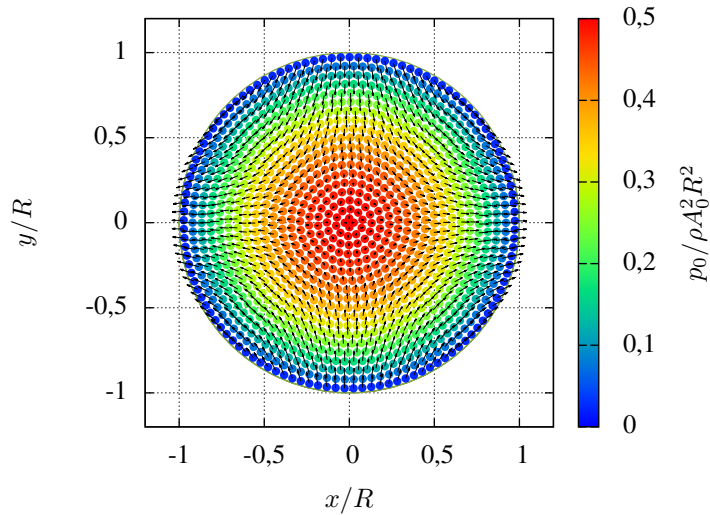


FIGURE 4.22 – Configuration initiale circulaire des particules pour la simulation de l'éirement d'une goutte en domaine ouvert. Les vecteurs symbolisent la vitesse initiale de la particule, colorée en fonction de la pression.

paragraphe précédent.

4.6.2 Influence de la disposition initiale des particules

Pour illustrer l'impact de l'initialisation de la simulation sur les résultats obtenus, nous avons réparti différemment les particules. Ainsi, les particules sont placées sur des cercles concentriques, sachant qu'un angle de rotation aléatoire est appliqué à chaque cercle, de manière à ne présenter aucune symétrie (figure 4.22). Le cas de la goutte se prête idéalement à ce genre de répartition et il est évident qu'on ne peut reproduire cela dans les cas d'écoulements étudiés plus loin. La figure 4.24 représente l'évolution du champ de particules à faible résolution ($R/\epsilon = 14, 7$), qui se rapproche plus de l'ellipse analytique que dans les cas précédents et avec un champ de pression moins perturbé que si un modèle SPH simple est employé (figure 4.23). Notamment, aucune ligne de particules ne se forme, entraînant une surestimation du petit-axe de l'ellipse.

Sur les figures 4.25, 4.26 et 4.27, la pression en A et les vitesses en B et C, respectivement, ne montrent d'ailleurs aucune oscillation et sont en bon accord avec la solution analytique. Un faible gain en précision, sur l'estimation de la vitesse verticale en C, est obtenu en augmentant la résolution (figure 4.27), juste avant le passage de l'interface. Dans ce cas précis, le schéma *Rusanov*-SPH seul a été employé. La renormalisation pourrait être ajoutée sans risque de voir apparaître les oscillations liées aux faibles valeurs propres, car la répartition des particules reste homogène tout au long de la simulation. Cependant, aucun gain significatif en précision n'est observé si la correction des gradients est appliquée. Cela montre que, si un soin particulier est apporté à l'initialisation des simulations de manière à éviter l'apparition d'instabilités numériques (ce qui était le cas avec une répartition cartésienne), il est encore une fois possible d'éviter l'ajout

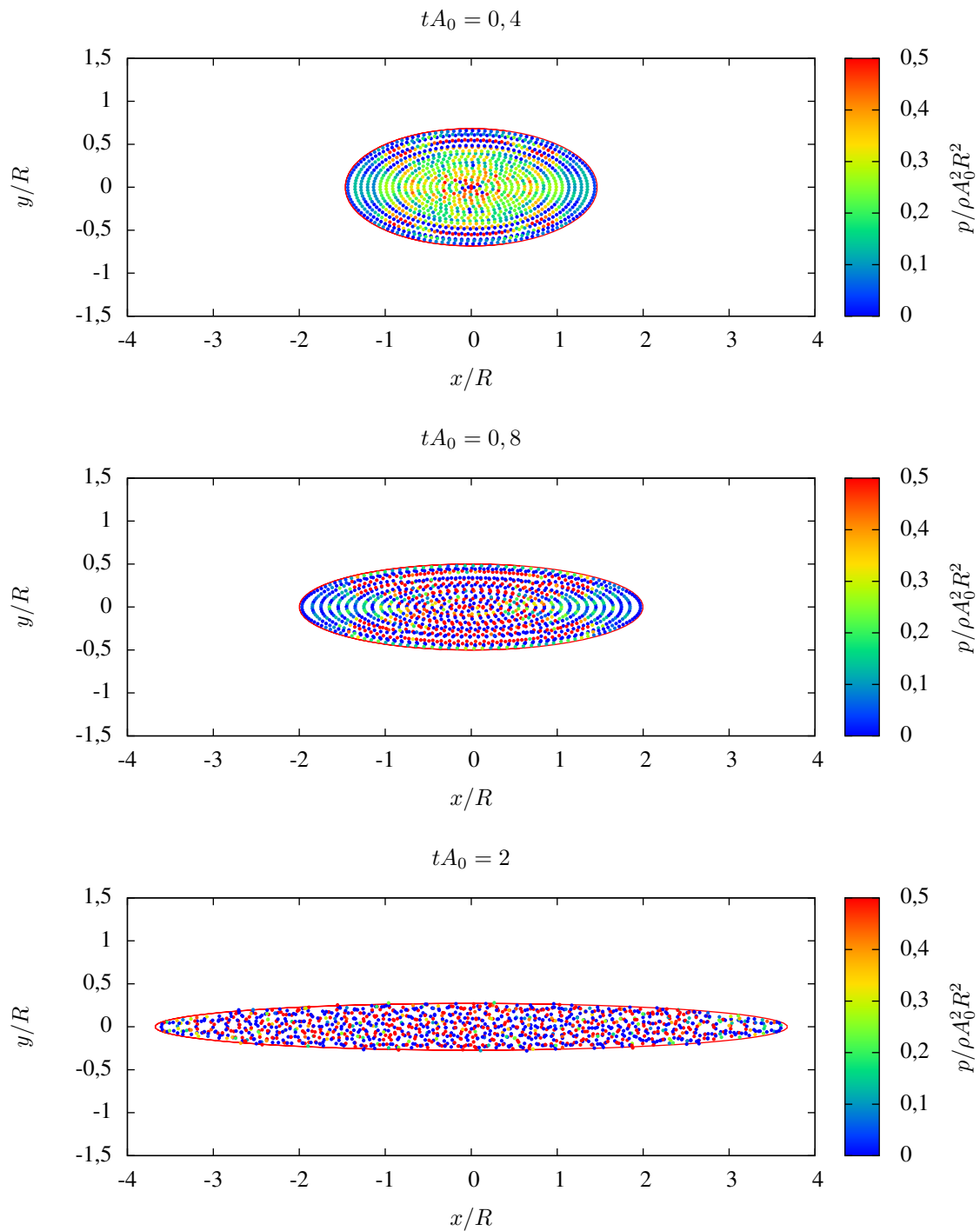


FIGURE 4.23 – Évolution de la goutte simulée à l'aide du modèle SPH simple, avec la formulation du gradient de pression de Oger [172] et Colagrossi et Landrini [41] ($\sigma = 1$, voir §3.2), $R/\epsilon = 14,7$ et une répartition circulaire des particules. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.



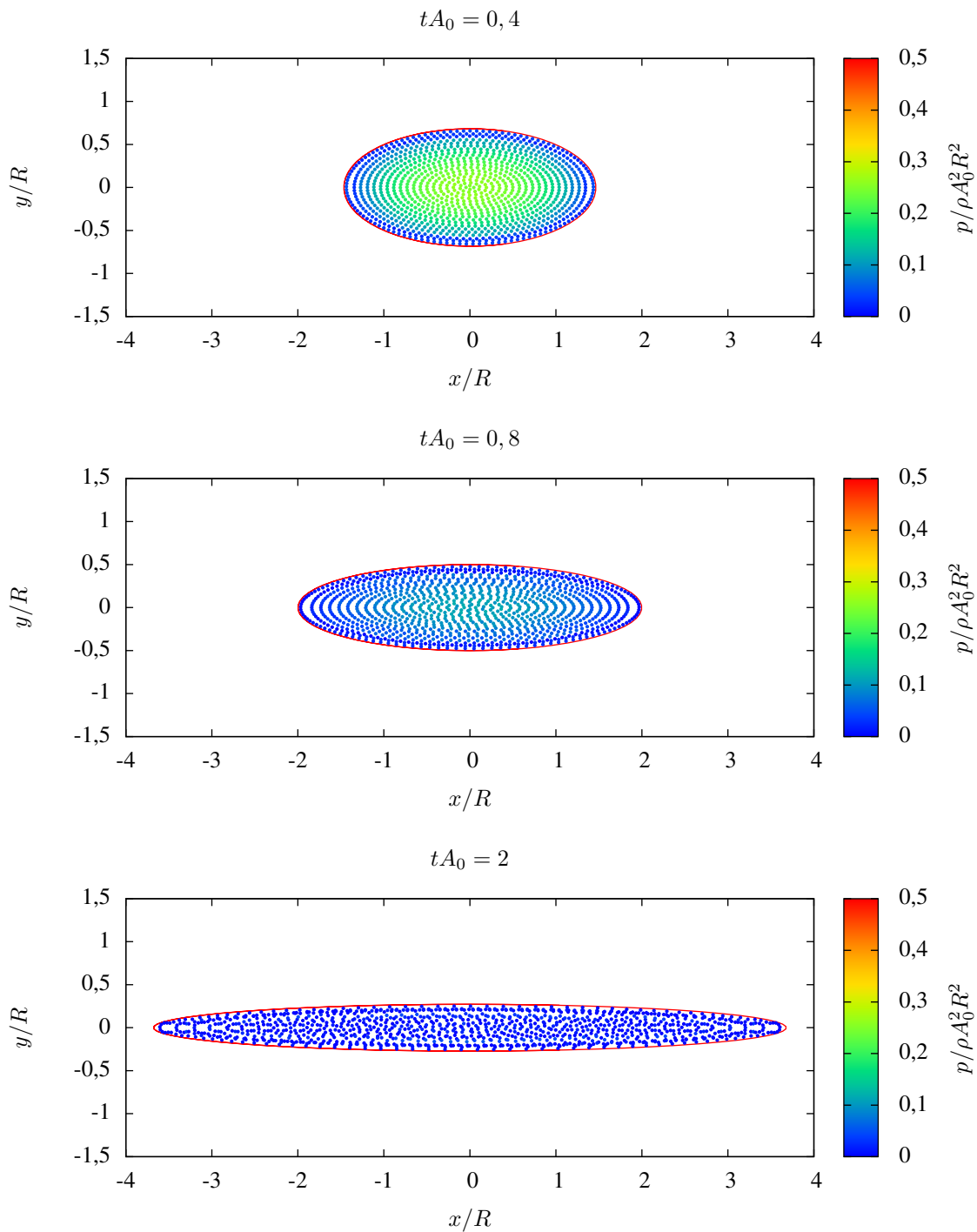


FIGURE 4.24 – Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari et al. [75], avec $R/\epsilon = 14,7$ et une répartition circulaire des particules. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.

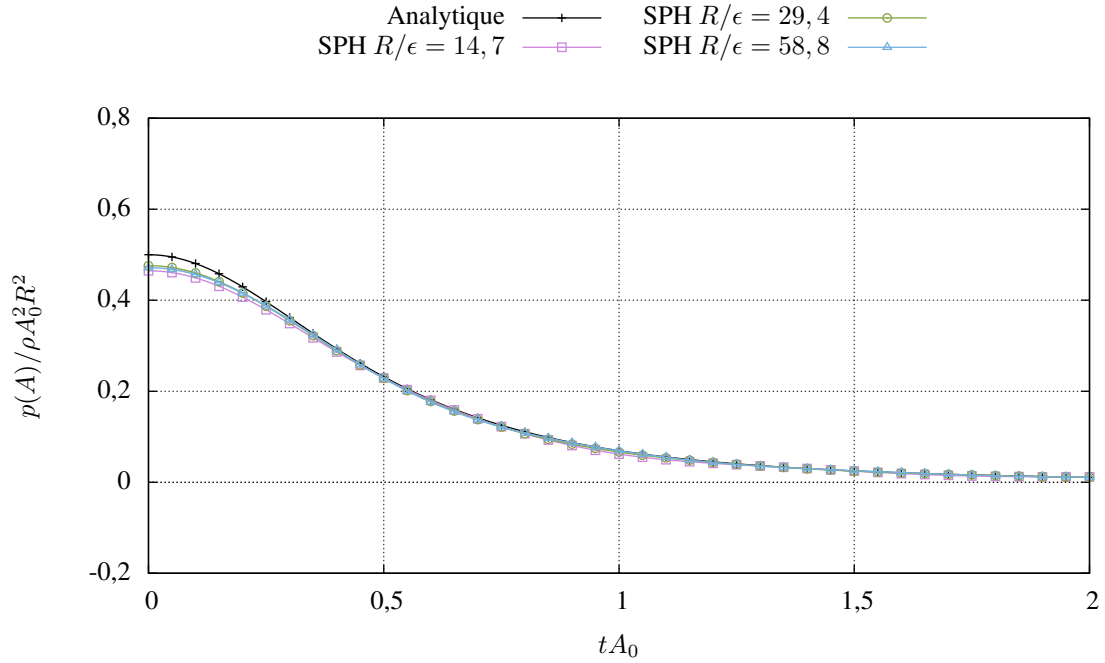


FIGURE 4.25 – Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Ferrari et al. [75] avec répartition circulaire des particules. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).

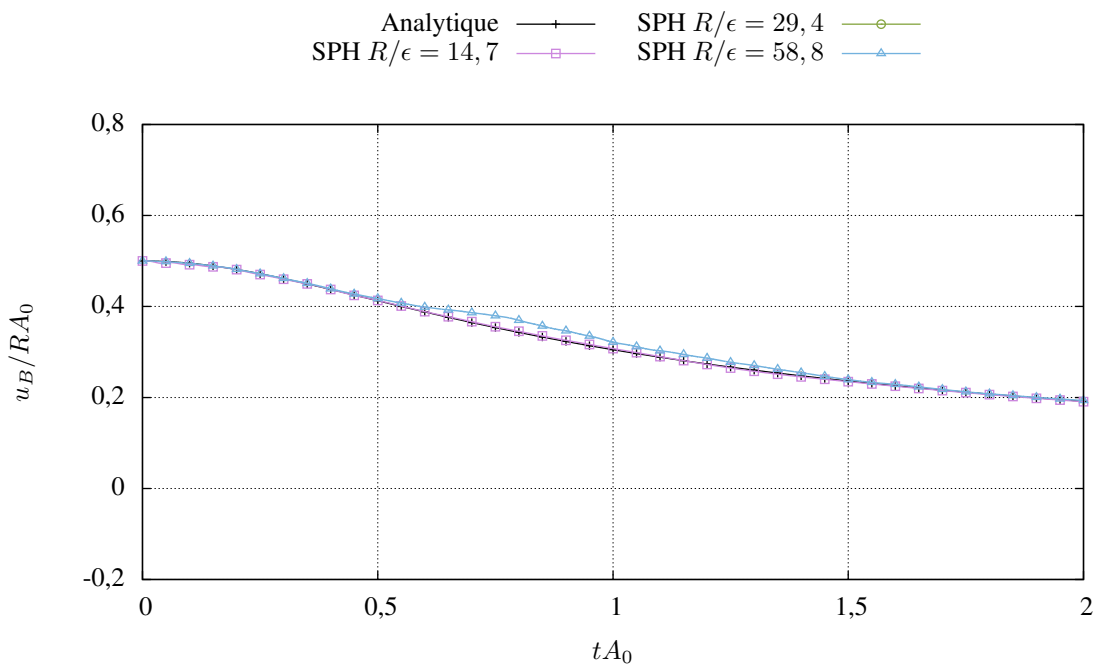


FIGURE 4.26 – Évolution de la composante horizontale de vitesse au point B (figure 4.1) par le modèle SPH de Ferrari et al. [75] avec répartition circulaire des particules. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).



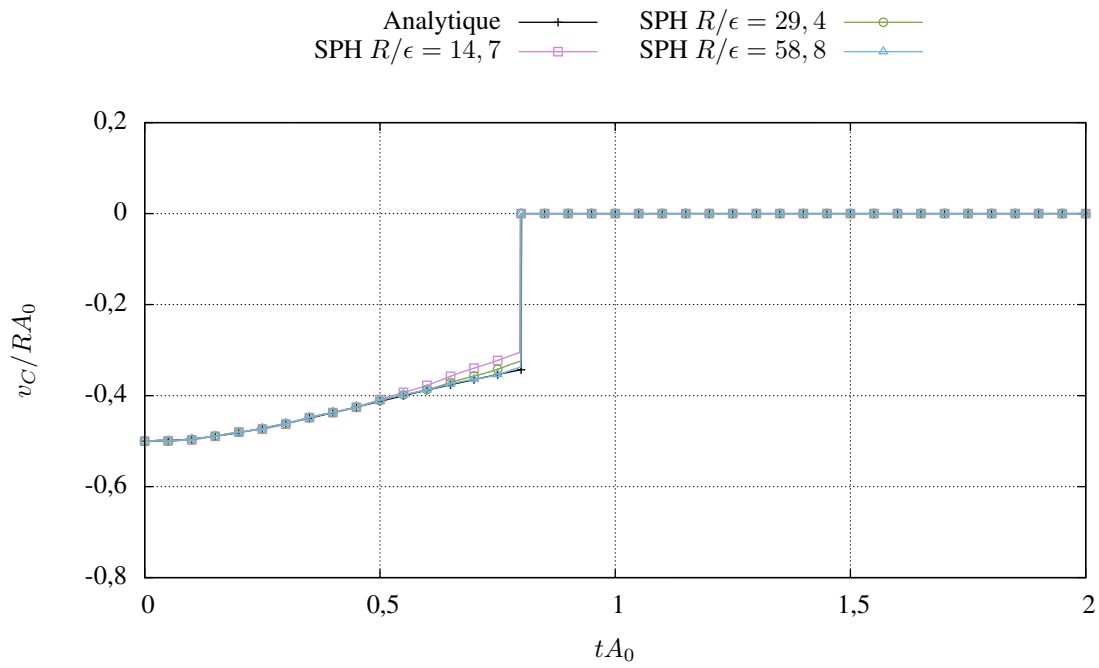


FIGURE 4.27 – Évolution de la composante verticale de vitesse au point C (figure 4.1) par le modèle SPH de Ferrari et al. [75] avec répartition circulaire des particules. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).

d'un algorithme coûteux, qui introduit un paramètre supplémentaire.

Chapitre 5

Effondrement d'une colonne d'eau en deux dimensions

Toujours en deux dimensions d'espace, ce cas, largement étudié au sein de la communauté, prévoit de laisser s'effondrer une colonne d'eau, initialement retenue par un mur vertical. Sous l'effet de la gravité, le fluide se déplace et heurte une paroi placée à l'extrémité opposée du domaine physique. Cet écoulement a été employé pour décrire et étudier les modèles standards de conditions aux parois. Il a également permis de vérifier que le modèle pouvait prendre en compte des déformations de la surface libre plus importantes que dans le cas précédent : impacts, reconnections...

5.1 Description du problème

On considère l'écoulement d'une masse fluide, initialement retenue par un mur vertical (figure 5.1). Sous l'effet de la gravité $\mathbf{g} = (0, -g)$, la colonne d'eau s'effondre en formant un front qui se déplace rapidement le long du sol. Quand un mur est placé à l'extrémité opposée du domaine de calcul, l'impact du front d'eau forme un jet vertical et s'effondre à nouveau. La surface libre subit ainsi plusieurs fragmentations et reconnections successives complexes.

Du point de vue mécanique, l'intérêt de cet écoulement *académique* réside dans sa nature instationnaire. Les phénomènes qui y sont rencontrés sont très similaires à ceux observés lors de tsunamis, ruptures de barrage, ballotement dans des réservoirs partiellement remplis de liquides, ressauts hydrauliques, déferlements en mer ou encore lors du franchissement de structures marines. Du point de vue modélisation, la méthode SPH est particulièrement intéressante sur ce cas, où le suivi des interfaces est complexe. Elle apporte d'ailleurs une solution partielle au problème de reconnections de la surface libre (impossible avec une méthode BEM [85]) et sans algorithme de détection (sauf en formulation incompressible), qui conduirait à un lissage systématique de l'interface, comme en méthodes VOF, par exemple [2, 139, 233]. Récemment, une méthode a été proposée par Dumbser [63], qui applique avec succès le modèle diphasique de Baer-Nunziato à la



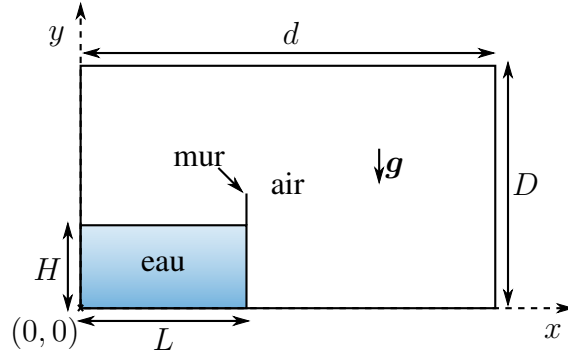


FIGURE 5.1 – Configuration initiale de l’effondrement d’une colonne d’eau. L représente la largeur du volume fluide initialement retenu, H est sa hauteur. Un mur vertical est placé à une distance d de l’origine. Le domaine de calcul est limité verticalement à une distance D de l’origine.

simulation d’une rupture de barrage. Du point de vue fondamental, l’influence de la surface libre et sa prise en compte en méthode SPH restent peu étudiés [36] mais son utilisation dans ce contexte est pourtant courante.

Des études théoriques existent sur l’écoulement considéré ici. La plus ancienne, basée sur l’hypothèse de faible profondeur, a été proposée par Ritter [193]. Il établit notamment que le front d’eau, en l’absence d’obstacle, se déplace à la vitesse constante :

$$u_{\text{front}} = 2\sqrt{gH}. \quad (5.1)$$

Plus précisément, la position $h(x, t)$ de la surface libre peut être obtenue par :

$$\sqrt{gh(x, t)} = \begin{cases} \sqrt{gH} & x \leq L - \sqrt{gH}t \\ \frac{1}{3} (2\sqrt{gH} - \frac{x-L}{t}) & L - \sqrt{gH}t < x \leq L + 2\sqrt{gH}t \\ 0 & x > L + 2\sqrt{gH}t \end{cases}, \quad (5.2)$$

et la vitesse verticale d’une particule fluide placée à la surface libre est :

$$v(x, t) = \begin{cases} 0 & x \leq L - \sqrt{gH}t \\ \frac{2}{3} (\frac{x-L}{t} + \sqrt{gH}) & L - \sqrt{gH}t < x \leq L + 2\sqrt{gH}t \\ 0 & x > L + 2\sqrt{gH}t \end{cases}. \quad (5.3)$$

La figure 5.2 montre la forme de surface libre calculée à l’aide de l’équation (5.2). Les travaux de Dressler [61] et de Whitham [237] ont eu pour objectif d’améliorer cette solution en modélisant la résistance hydraulique du liquide sur le sol. Les résultats obtenus réduisent le retard constaté des données expérimentales sur la solution analytique (5.1) pour la position du front d’eau. La vitesse de ce front n’est en effet pas constante mais elle tend vers la valeur (5.1) [180]. Au delà de cette grandeur caractéristique, peu de données théoriques sont disponibles sur ce cas, rendant la

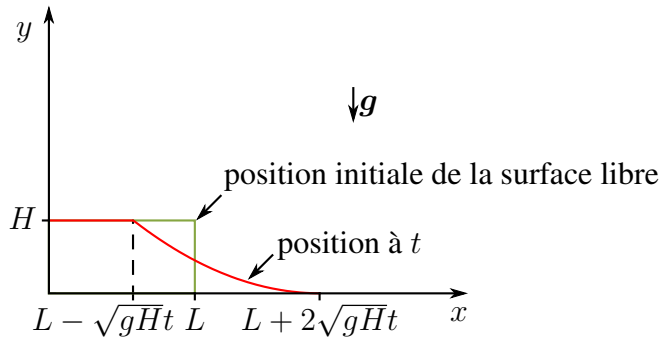


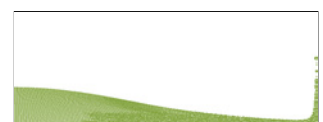
FIGURE 5.2 – Évolution de la surface libre après une rupture de barrage en hypothèse de faible profondeur, selon Ritter [193].

simulation numérique pertinente pour des applications qui sortent du cadre académique. Ce cas est donc étudié à la fois pour son intérêt physique et pour la validation de codes de calculs. Cependant les données expérimentales restent rares, tant la mise en œuvre de modèles réduits est difficile : le soulèvement du mur qui retient la masse d'eau en équilibre hydrostatique n'est jamais suffisamment précis pour obtenir une reproductibilité satisfaisante. Souvent, les résultats numériques sont simplement superposés à des vues photographiques en divers instants de l'écoulement, pour comparaison [102, 234]. Martin et Moyce [145] proposent des données sur la position du front dans le cas $L = H$. Zhou *et al.* [240] fournissent des valeurs sur l'évolution de la hauteur d'eau en différents points et sur la pression à l'impact dans le cas $L = 2H$. Stansby *et al.* [215] étudient l'évolution d'une rupture de barrage sur un court intervalle après la suppression du mur et avec un sol déjà recouvert d'une lame d'eau, l'écoulement est alors complètement modifié.

5.2 État de l'art sur la modélisation de cet écoulement

Au delà du simple calcul de la position de la surface libre, la résolution des équations d'Euler ou de Navier-Stokes est nécessaire pour prendre en compte également les problèmes d'impacts, l'influence de la condition au sol (rugosité, glissement ...) ou encore le rôle joué par l'air emprisonné dans le fluide après impact [41]. Le tableau 5.1 regroupe les différentes tentatives de modélisation de cet écoulement par SPH en formulation faiblement compressible.

Shao et Lo [207] proposent une solution SPH incompressible dans le cas d'un liquide visqueux et sans impact : la surface libre, peu déformée, y est donc bien représentée. Lee *et al.* [125] comparent WCSPH et ISPH en juxtaposant les champs de pression obtenus. La solution faiblement compressible y paraît plus mauvaise que le solution incompressible, mais le modèle WCSPH employé n'intègre aucune des corrections présentées précédemment pour le lissage de la pression. Une variante de SPH, la méthode Moving Particle Semi-implicit (MPS) a également été appliquée à ce cas par Koshizuka et Oka [111] et Ataie-Ashtiani et Farhadi [5]. Dans toutes ces variantes



Références par ordre chronologique	Modèle	Conditions aux limites	Dimensions du problème	Impact et recon-nexion	Affichage des particules			Comparaison des résultats
					position	vitesse	pression	
Monaghan [155] 1994	XSPH + viscosité artificielle	particules répulsives	2D, $L/H = 1$	✗ NON	✓ OUI	✗ NON	✗ NON	position du front avec [145]
Bonet et Lok [16] 1999	SPH + renormalisation	forçes à la frontière	2D, $L/H = 1$	✗ NON	✓ OUI	✗ NON	✗ NON	position du front avec [145]
Colagrossi et Landrini [41] 2003	diphasique + XSPH + viscosité artificielle + densité MLS	particules fantômes	2D, $L/H = 1$ et $L/H = 2$	✓ OUI	✓ OUI	✗ NON	✓ OUI	position du front avec [145], hauteur d'eau et pression à l'impact avec [240], surface libre avec [85]
Liu <i>et al.</i> [134] 2005	SPH + renormalisation (FPM)	particules répulsives	2D, $L/H = 1$	✗ NON	✓ OUI	✗ NON	✗ NON	position du front avec [145]
Roubtsova et Kahawita [196] 2006	SPH + solveur de riemann	particules fixes	2D et 3D, $L/H = 1$ et $L/H = 0.5$	✓ OUI	✓ OUI	✗ NON	✗ NON	position du front avec [145] et [111], vues photographiques pour la surface libre
Violleau et Issa [234] 2007	SPH + modèles de turbulence	particules fixes	2D, $L/H = 0.5$	✓ OUI	✓ OUI	✓ OUI	✗ NON	position du front avec [111], vues photographiques pour la surface libre
Crespo <i>et al.</i> [48] 2007	XSPH + viscosité artificielle	particules dynamiques	2D, $L/H = 0.5$	✓ OUI	✓ OUI	✓ OUI	✗ NON	position du front avec [111]
Ferrari <i>et al.</i> [75] 2009	SPH + flux Rusanov	points de frontières et symétries multiples	3D, $L/H = 2$	✓ OUI	✓ OUI	✗ NON	✓ OUI	hauteur d'eau et pression à l'impact avec [240]
Molteni et Colagrossi [151] 2009	δ -SPH (viscosité artificielle + diffusion de la masse volumique)	particules fantômes	2D, $L/H = 2$	✓ OUI	✓ OUI	✗ NON	✓ OUI	pression à l'impact avec [127]

TABLE 5.1 – Récapitulatif des modélisations avec SPH faiblement compressible de l'effondrement de colonnes d'eau.

incompressibles, la détection des particules de surface libre et l'application précise de la condition de pression reste problématique. Marrone *et al.* [144] propose un algorithme de détection basé sur la matrice de renormalisation, qu'il applique à la rupture de barrage pour illustration.

Par ailleurs, l'application de l'approximation SPH aux équations de Saint-Venant donne des résultats proches de la théorie [4, 194]. L'interaction fluide-structure entre un obstacle flexible et la masse fluide en effondrement a également été étudiée en méthode SPH par Antoci *et al.* [3] et Rafiee et Thiagarajan [190].

5.3 Modélisation par SPH

En se basant sur les résultats obtenus au cours du chapitre précédent, la forme suivante des équations d'Euler a été appliquée à la rupture de barrage.

$$\frac{D\mathbf{u}_i}{Dt} = -\frac{1}{\rho_i} \sum_j (P_i + P_j) \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} + \mathbf{g} \quad (5.4)$$

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \left(\mathbf{u}_{ij} + \mathbf{n}_{ij} \left(\frac{c_{ij}}{\rho_i} (\rho_j - \rho_i) \right) \right) \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (5.5)$$

$$\frac{D\mathbf{x}_i}{Dt} = \mathbf{u}_i \quad (5.6)$$

Ce cas impose également la mise en œuvre de conditions aux limites pour la modélisation des parois solides. La validation des résultats nécessite le calcul de la position du front, de l'évolution de la hauteur d'eau en un point précis en fonction du temps, et de la pression en un point de la paroi. Ces calculs requièrent une attention particulière et l'utilisation de techniques spécifiques, que nous allons aborder maintenant. En outre, une donnée importante du calcul pour l'évaluation de la fiabilité du modèle est la conservation de l'énergie totale au cours du temps. Le modèle employé ici étant faiblement compressible, une partie de l'énergie d'une particule est associée à la compressibilité. Cette énergie *interne* est calculée par intégration d'une équation supplémentaire, qui sera présentée.

5.3.1 Conditions aux limites : parois solides

Le tableau 5.1 contient une colonne précisant comment chaque auteur a modélisé les parois solides qui retiennent le fluide. On y remarque différentes terminologies. Les stratégies développées pour imposer ces conditions limites sont en effet très variées.

Particules répulsives

La première approche proposée fût celle de Monaghan [155] : la technique dite des *particules répulsives*, qui permet d'imposer une condition de non-pénétration. Les frontières fixes ou mobiles



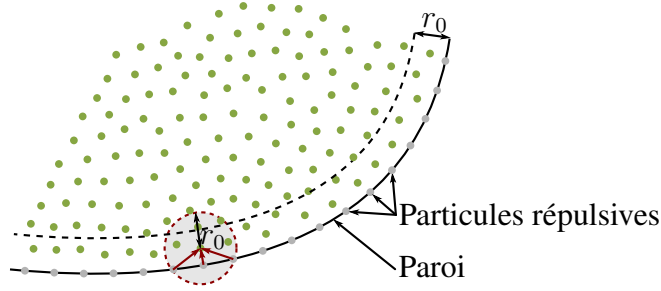


FIGURE 5.3 – Conditions aux limites imposées à l'aide de particules répulsives selon Monaghan [155].

sont représentées par une rangée de points, exerçant une force répulsive sur les particules fluides (figure 5.3). L'écriture de cette force répulsive est inspirée par celle du potentiel de Lennard-Jones en dynamique moléculaire (équation (5.7)). Monaghan précise que cette approche *fonctionne mieux* que la frontière immergée de Peskin [183], présentée plus loin, où les forces exercées par la paroi sur le fluide sont décrites comme une fonction $\delta_0()$ approchée.

$$f_{ij} = \begin{cases} D \left[\left(\frac{r_0}{r_{ij}} \right)^{p_1} - \left(\frac{r_0}{r_{ij}} \right)^{p_2} \right] \frac{\mathbf{x}_{ij}}{r_{ij}^2} & \text{si } \left(\frac{r_0}{r_{ij}} \right) \leq 1 \\ 0 & \text{si } \left(\frac{r_0}{r_{ij}} \right) > 1 \end{cases} \quad (5.7)$$

La valeur de D dépend du problème traité, elle est généralement choisie de l'ordre du carré de la vitesse maximale. Le seuil (de distance par rapport à la paroi) d'application de cette force, r_0 , est également fixé de façon arbitraire à la distance inter-particulaire initiale h . Les constantes p_1 et p_2 doivent satisfaire l'inégalité $p_1 > p_2$. Les valeurs courantes choisies sont $p_1 = 4$ et $p_2 = 2$, ou encore $p_1 = 12$ et $p_2 = 6$.

Ce modèle de conditions aux limites est encore très couramment employé, en raison de sa simplicité et de la souplesse qu'il autorise, malgré la présence de paramètres à calibrer. En effet, des formes géométriques très complexes peuvent être facilement décrites quand il suffit de placer des points de façon *assez* régulière sur des parois, que l'on peut imaginer directement issues d'un logiciel de construction de maillage. De plus, elle demande peu de ressources de calcul supplémentaires, si la détection des particules proches de la paroi est intégrée à la recherche globale des interactions (voir §8.2). En revanche, elle ne permet pas de modéliser précisément une condition du type $\mathbf{u} \cdot \mathbf{n} = 0$ (glissement) ou $\mathbf{u} = 0$ (adhérence). Il s'agit simplement d'une condition d'anti-pénétration sans réelle interprétation physique ; selon l'espacement de ces particules répulsives, la force ressentie par les particules fluides n'est pas constante si la particule se déplace parallèlement à la paroi. Elle présente aussi l'inconvénient de ne pas palier au problème de troncature du noyau à la paroi (absence de particules à l'extérieur du domaine) (figure 2.3). Cette technique souffre donc d'un manque de précision évident sur la condition imposée, et génère même des erreurs au

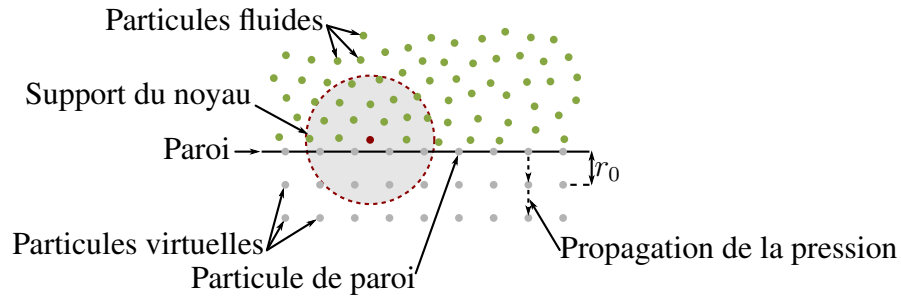


FIGURE 5.4 – Conditions aux limites imposées à l'aide de particules virtuelles.

sein du schéma d'interpolation. Une variante de cette méthode a été proposée ultérieurement par Monaghan et Kos [160], où la force est définie normale à la paroi.

Particules virtuelles

Cette méthode de modélisation des conditions aux bords est une extension de l'approche précédente, visant à apporter une solution au problème de l'absence de particules au delà de la paroi. Initialement proposée par Shao et Lo [207], elle a ensuite été reprise par Issa [102], Issa *et al.* [103] ou encore Lee *et al.* [125]. Comme le montre la figure 5.4, des particules sont placées sur les parois à modéliser, de façon régulière avec un espacement choisi identique à celui des particules fluides, initialement. La masse volumique associée aux particules de la paroi est calculée par intégration de l'équation de conservation de la masse (équation (5.5)), la loi d'état permet ensuite d'en déduire la pression. Des rangées de particules virtuelles sont placées au delà de la frontière, par copies successives des particules de paroi, jusqu'à avoir suffisamment de particules pour éviter des lacunes de discrétisation dans le voisinage des particules fluides. Ces particules virtuelles ont donc la même pression que les particules de paroi dont elles sont issues. La vitesse de ces particule est nulle, ce qui en fait un modèle de condition d'adhérence. Peu de détails supplémentaires sont disponibles dans la bibliographie au sujet de l'application de ce procédé à une condition de glissement.

Particules dynamiques

Développée par Dalrymple et Knio [51], ce modèle de parois solides a également pour but d'éviter la troncature du support d'intégration des particules fluides. A la différence des particules virtuelles, aucune particule n'est placée sur la paroi, mais uniquement au delà et selon un maillage caractéristique, formé par une succession de rangées de particules décalées les unes par rapport aux autres (figure 5.5). Comme dans le cas précédent, la masse volumique est déduite par résolution de l'équation de conservation de la masse. La pression obtenue est prise en compte dans le calcul



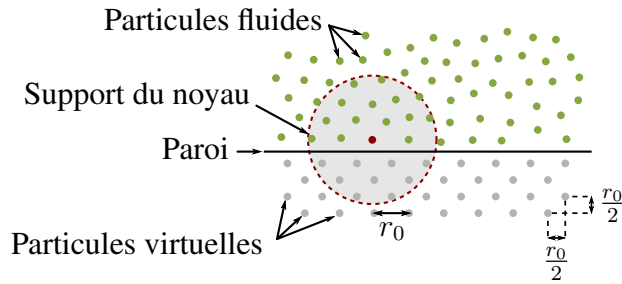


FIGURE 5.5 – Conditions aux limites imposées à l'aide de particules dynamiques.

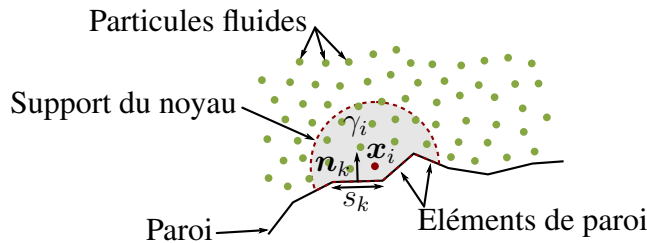


FIGURE 5.6 – Conditions aux limites imposées à l'aide d'intégrales de surface.

de l'accélération des particules fluides, exerçant sur celles-ci une force répulsive les empêchant de traverser la paroi. Cette approche a été employée par Gomez-Gesteira *et al.* [81], Dalrymple et Rogers [52] ou encore Gomez-Gesteira *et al.* [82]. Le mécanisme de répulsion a été détaillé par Crespo *et al.* [48], mais aucune précision n'est donnée sur la nature de la condition en vitesse imposée.

Intégrales de surface

Cette approche regroupe en réalité un certain nombre de techniques différentes, qui visent toutes à imposer les conditions aux limites en tirant partie de la troncature du noyau par la paroi (figure 2.3), et de l'intégrale de surface qui apparaît dans ce cas (équation (2.27)). Contrairement aux deux modèles précédents, aucune particule fictive n'est placée au delà du bord. Cela a pour avantage d'éviter la programmation d'algorithmes complexes pour la modélisation de bords tels que des angles droits, parois courbes... L'idée est apparue dans les travaux de Kulasegaram *et al.* [113] et nécessite une réécriture complète des équations sous forme discrète, qui fait intervenir des termes γ_i pour chaque particule. Ces termes γ_i correspondent à l'intégrale du noyau d'interpolation

sur le support (figure 5.6) :

$$\gamma_i = \sum_j \eta_\epsilon^0(\mathbf{x}_{ij}) V_j \quad (5.8)$$

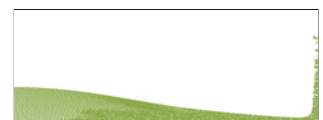
Un exemple d'approximation du gradient de pression, proposé par De Leffe *et al.* [54], selon cette méthode, est :

$$\nabla P_i = \sum_j (P_i + P_j) \frac{\nabla \eta_\epsilon^0(\mathbf{x}_{ij})}{\gamma_i} \frac{m_j}{\rho_j} + \sum_k (P_i + P_k) \frac{\eta_\epsilon^0(\mathbf{x}_{ik})}{\gamma_i} \mathbf{n}_k s_k \quad (5.9)$$

Ils agissent d'une part comme une normalisation des opérateurs-intégrales quand le support de la particule est tronqué, ils pondèrent d'autre part l'influence de l'intégrale de surface. L'écriture du gradient de pression sous cette forme rompt le principe de réciprocité des interactions entre particules fluides, la conservation des quantités n'est donc plus assurée. De plus, on peut remarquer que l'évaluation du second terme nécessite de connaître la pression à la paroi. Ce problème est résolu soit par interpolation depuis les valeurs connues sur les particules fluides [54] ou bien en l'extrapolant à l'aide de la méthode des *caractéristiques* [141]. Un inconvénient de cette approche est le calcul des termes γ_i (équation (5.8)), que certains auteurs préfèrent évaluer analytiquement ou par approximation polynomiale [72, 74]. Un avantage, lorsque cette méthode est couplée aux solveurs de Riemann, est la possibilité de calculer des flux aux parois, de manière à imposer des conditions de non-réflexion des ondes [54, 142]. Ce mode de représentation des conditions limites est prometteur car il permet de représenter des parois aussi complexes qu'avec la méthode des particules répulsives, tout en corrigeant les problèmes liés à l'absence de particules hors du domaine. En revanche, le calcul des termes γ_i et les nouvelles équations discrètes requièrent un temps de calcul non négligeable. Une particule, dont le support serait tronqué à la fois par une paroi solide et par la surface libre, pose problème avec ce type de méthode. L'évaluation de γ_i est en effet ambiguë.

Particules fantômes

La méthode des particules fantômes, ou condition-miroir, est assez courante en méthodes particulières. Son application à la méthode SPH a été proposée par Libersky *et al.* [129], puis reprise par Takeda *et al.* [223] pour des conditions de type adhérence. C'est d'ailleurs la technique employée par Colagrossi et Landrini [41] pour l'étude de l'effondrement d'une colonne d'eau. Cette approche consiste à appliquer une symétrie par rapport au bord modélisé. Ainsi, toutes les particules placées à une distance inférieure à 3ϵ de la paroi sont dupliquées (figure 5.7) et leurs propriétés sont ajustées en fonction de la conditions à imposer. Ces particules sont prises en compte dans toutes les équations du modèle, ainsi que dans les éventuelles améliorations plus techniques que peut contenir le code. Ce point constitue un réel avantage par rapport aux autres méthodes car il n'est pas nécessaire de tester le type de particule pour choisir le type de traitement à réaliser. Ensuite, bien que plus coûteuse en espace mémoire et temps de calcul, elle est la seule qui permet



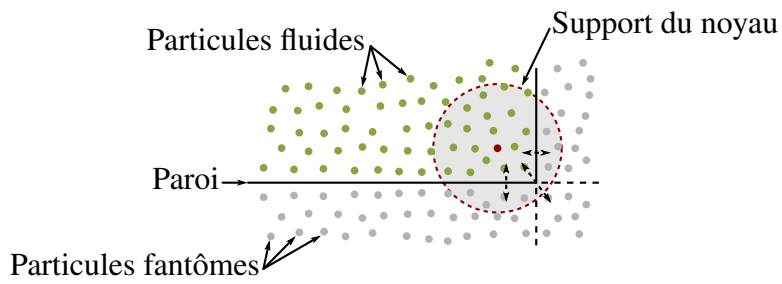


FIGURE 5.7 – Conditions aux limites imposées à l'aide de particules fantômes. Traitement d'un angle : symétrie axiale pour les segments adjacents et symétrie centrale par rapport au coin.

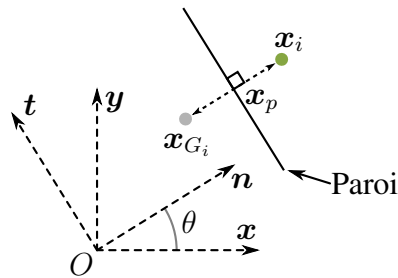


FIGURE 5.8 – Symétrie d'une particule par rapport à une paroi.

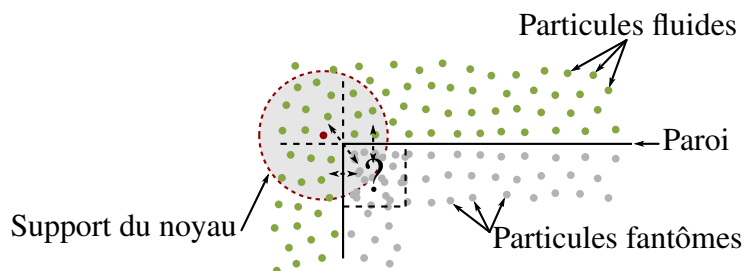


FIGURE 5.9 – Exemple de configuration problématique avec la méthode des particules fantômes : le coin doit faire l'objet de trois symétries successives, superposant trois couches de particules fantômes au même endroit.

d'appliquer des conditions limites physiques.

Ainsi, sur la base des travaux de Oger [172], prenons l'exemple d'une condition de glissement sur une paroi plane. Pour une particule libre \mathbf{x}_i à une distance de la paroi inférieure à 3ϵ , on calcule sa projection orthogonale sur la paroi. Si \mathbf{x}_p est la position de cette projection (figure 5.8), alors une particule fantôme est placée à la position \mathbf{x}_{G_i} , définie par :

$$\mathbf{x}_{G_i} = 2\mathbf{x}_p - \mathbf{x}_i \quad (5.10)$$

Quand la paroi est horizontale ou verticale, le calcul informatique est bien évidemment simplifié. Il reste à définir les propriétés physiques de cette particule. Cela doit être réalisé de façon rigoureuse. On peut cependant raisonner de différentes manières pour finalement arriver aux mêmes conclusions. Si on cherche quelle vitesse attribuer à la particule fantôme, on peut partir de sa définition :

$$\mathbf{u}_{G_i} = \frac{D\mathbf{x}_{G_i}}{Dt} \quad (5.11)$$

En exprimant la dérivée dans le repère $(O, \mathbf{n}, \mathbf{t})$ lié à la paroi, on a :

$$\mathbf{u}_{G_i} = \frac{D_n \mathbf{x}_{G_i}}{Dt} + \boldsymbol{\Omega}_n \wedge \mathbf{x}_{G_i} \quad (5.12)$$

avec $\boldsymbol{\Omega}_n$, le vecteur rotation instantanée de la base $(O, \mathbf{n}, \mathbf{t})$ par rapport au repère absolu. Dans le cas bidimensionnel, on a : $\boldsymbol{\Omega}_n = \dot{\theta}\mathbf{z}$. La vitesse de la particule fantôme s'écrit alors :

$$\mathbf{u}_{G_i} = \left(2u_{pn} - \dot{\theta}(2x_{pt} - x_{it}) - u_{in}\right) \mathbf{n} + \left(2u_{pt} - \dot{\theta}(2x_{pn} - x_{in}) - u_{it}\right) \mathbf{t} \quad (5.13)$$

où les indices n et t indiquent que les coordonnées sont exprimées dans le repère $(O, \mathbf{n}, \mathbf{t})$. Dans le cas d'une condition de glissement, pour laquelle la vitesse tangentielle à la paroi est égale à la vitesse de la particule à symétriser, on a donc :

$$u_{pt} = u_{it} \quad (5.14)$$

$$x_{pt} = x_{it} \quad (5.15)$$

La vitesse se réduit alors à :

$$\mathbf{u}_{G_i} = \left(2u_{pn} - \dot{\theta}x_{pt} - u_{in}\right) \mathbf{n} + \left(u_{it} - \dot{\theta}(2x_{pn} - x_{in})\right) \mathbf{t} \quad (5.16)$$

Et si la paroi est fixe, on alors :

$$\mathbf{u}_{G_i} = -u_{in}\mathbf{n} + u_{it}\mathbf{t} \quad (5.17)$$

Pour une condition d'adhérence, on aura :

$$\mathbf{u}_{G_i} = -u_{in}\mathbf{n} - u_{it}\mathbf{t} \quad (5.18)$$



Une étude de Colagrossi *et al.* [40] remet en cause l'usage d'une vitesse opposée pour la particule fantôme, par analyse des contraintes visqueuses à l'approche de la paroi. Pour le cas présenté ici, la condition à imposer étant de type glissement, ce point n'a pas fait l'objet d'une étude plus approfondie. Il reste à définir le volume, la masse volumique (et donc la masse) ainsi que la pression de la particule fantôme. La condition de non-pénétration s'écrit :

$$(\mathbf{u}_i - \mathbf{u}_p) \cdot \mathbf{n} = 0 \quad (5.19)$$

Par dérivation, on a :

$$\frac{D\mathbf{u}_i}{Dt} \cdot \mathbf{n} = (\mathbf{u}_i - \mathbf{u}_p) \cdot (\boldsymbol{\Omega}_n \wedge \mathbf{n}) + \frac{D\mathbf{u}_p}{Dt} \cdot \mathbf{n} \quad (5.20)$$

En remplaçant $\frac{D\mathbf{u}_i}{Dt}$ par l'équation de conservation de la quantité de mouvement, en utilisant la loi d'état (3.13) et en intégrant, on obtient l'expression suivante pour la masse volumique de la particule symétrisée :

$$\rho_{G_i} = \left[\rho_i^{\gamma-1} + \frac{\rho_0^{\gamma-1}(\gamma-1)}{c_0^2} \left((\mathbf{u}_i - \mathbf{u}_p) \cdot (\boldsymbol{\Omega}_n \wedge \mathbf{n}) + \frac{D\mathbf{u}_p}{Dt} \cdot \mathbf{n} + \mathbf{g} \cdot \mathbf{n} \right) (x_{G_{in}} - x_{in}) \right]^{\frac{1}{\gamma-1}} \quad (5.21)$$

Pour une paroi fixe horizontale, cette expression se réduit à :

$$\rho_{G_i} = \left[\rho_i^{\gamma-1} + \frac{\rho_0^{\gamma-1}(\gamma-1)}{c_0^2} g(x_{G_{in}} - x_{in}) \right]^{\frac{1}{\gamma-1}} \quad (5.22)$$

Cette dernière équation s'interprète comme une correction hydrostatique à appliquer aux particules quand celles-ci servent à modéliser le fond d'une cuve, par exemple. Dans le cadre d'une formulation faiblement compressible, il paraît cohérent que la masse volumique soit modifiée de manière à reproduire une condition en pression du type Neumann, *via* la loi d'état :

$$\frac{\partial P}{\partial \mathbf{n}} = -\rho g. \quad (5.23)$$

Si la paroi est fixe et verticale, on a :

$$\rho_{G_i} = \rho_i \quad (5.24)$$

Les relations (5.21)-(5.24) sont utiles pour définir correctement la masse volumique de la particule fantôme, du cas simple d'une paroi fixe verticale ou horizontale, à celui d'une paroi oblique en mouvement (comme un batteur-pivot en canal à houle, par exemple). La pression est ensuite déduite de la loi d'état grâce à la masse volumique que l'on vient de calculer. Le volume de la particule fantôme étant choisi identique à celui de la particule fluide, sa masse sera calculée par :

$$m_{G_i} = \rho_{G_i} V_{G_i} \quad (5.25)$$

Au cours de cette thèse, nous n'avons pas tenté de modéliser des parois courbes. Morris *et al.* [164] proposent un traitement spécifique en modifiant les vitesses des particules fantômes en fonction de la courbure locale. Oger [172] propose, quant à lui, d'employer les relations précédentes, mais dans ce cas particulier, de modifier le volume de la particule symétrisée de manière à maintenir une répartition suffisante de masse au delà de la frontière. Il donne la relation suivante :

$$V_{G_i} = \frac{R - d}{R + d} V_i, \quad (5.26)$$

où R est le rayon de courbure local et d , la distance de la particule à la paroi. Une possibilité reste cependant de discrétiser une paroi courbe par des segments, comme dans la plupart des méthodes maillées. La qualité de la condition dépend alors de la discrétisation employée. La méthode des particules fantômes permet de traiter de façon précise des parois simples, mais son extension à des formes plus complexes reste problématique, plus particulièrement quand il s'agit de modéliser des angles, comme le montre la figure 5.9. L'application d'une symétrie par rapport aux deux côtés adjacents et par rapport au coin résulte dans l'accumulation de 3 ensembles de particules sur une zone de taille restreinte, cet excès de masse génère des instabilités et n'a d'ailleurs aucun sens physique. Ce problème a déjà été souligné par Le Touzé *et al.* [122] et Colagrossi *et al.* [39]. Plusieurs artifices géométriques ont été testés pour remédier à ce problème, comme la technique dite des *visibilités* [13] initialement prévue pour tenir compte des fractures dans les solides modélisés par SPH ; sa mise en œuvre reste hasardeuse quand il s'agit de l'associer à la technique des particules fantômes. Une évolution combinée des particules fantômes et des particules virtuelles a été proposée par Colagrossi *et al.* [37], qui place des particules fixes derrière les parois. Leurs propriétés d'une particule fixe donnée sont définies en fonction des valeurs interpolées dans le fluide, par symétrie de cette particule par rapport à la paroi. Cette approche permet d'améliorer la prise en compte de formes variées mais l'algorithme de positionnement de ces particules fixes au delà de la paroi peut s'avérer complexe.

Toutes ces méthodes apportent une solution à la modélisation de conditions limites extérieures, plus ou moins précise. Les particules fantômes sont préférées dans le cas de parois simples. Les intégrales de surface offrent la possibilité de modéliser des parois aux formes plus complexes, moyennant un plus grand nombre d'opérations. A l'inverse, bien que plus simples, les particules répulsives et virtuelles ne permettent pas d'imposer de condition physique. Sur la base des résultats observés dans la bibliographie, nous avons choisi la méthode des particules fantômes. Plusieurs tentatives d'extension aux formes complexes ont été menées, sans résultat satisfaisant. Nous avons préféré développer et intégrer un second modèle de condition aux limites, spécifique aux corps immergés dans le fluide. Cette méthode, inspirée de la méthode de *frontière immergée* [183], permet d'imposer une condition d'adhérence sur toute paroi rigide ou déformable, de forme quelconque. Une description plus détaillée est donnée dans le chapitre 6.



5.3.2 Conditions aux limites : surface libre

En formulation faiblement compressible, la surface libre ne fait pas l'objet d'un traitement spécifique. En théorie, deux conditions sont à appliquer :

Condition cinématique

La conditions cinématique impose qu'une particule élémentaire de fluide initialement située à la surface libre doit y rester pendant toute la durée de l'écoulement. La méthode SPH, lagrangienne, remplit implicitement cette condition. Les écarts éventuellement observés ne peuvent être attribuer qu'au pas de discrétisation choisi et au fait qu'aucune particule n'est placée exactement sur la surface libre. En réalité, la surface est située à environ $\frac{h}{2}$ avec h , la distance inter-particulaire initiale. Il n'y a donc qu'en raffinant le maillage de particules que l'on peut s'assurer d'améliorer la prise en compte de cette condition.

Condition dynamique

La condition dynamique impose quant à elle la continuité des contraintes à l'interface. La seule manière d'imposer cette condition avec précision, serait de discrétiser et simuler également l'évolution de la phase air au dessus du fluide. Cela aurait pour effet d'augmenter considérablement le nombre de particules pour des simulations en domaine ouvert où, justement, la méthode SPH trouve son intérêt. En effet, la forme des équations discrètes employées ici, conduit la masse volumique des particules à tendre vers la masse volumique de référence quand celles-ci s'approchent de la surface libre. Leur pression tend alors vers zéro, choisie comme pression atmosphérique. En toute rigueur, les particules dont le support est tronqué à l'interface devraient faire intervenir un terme de surface, analogue à celui employé avec la technique des intégrales de surface pour les frontières solides, citée plus haut (§5.3.1). Cependant, un calcul précis de la forme de la surface libre et de la contrainte à imposer nécessiterait des algorithmes complexes. Cela n'assurerait pas pour autant d'améliorer la condition imposée actuellement. Même lorsque la phase air est prise en compte, l'interface requiert des corrections supplémentaires, absentes ici, pour maintenir sa continuité [41].

En formulation incompressible, l'application de la condition dynamique nécessite encore de nettes avancées pour réduire les instabilités actuellement observées quand il y a fragmentation ou reconnexion de la surface libre. Ces deux points sont eux-mêmes des sujets d'études difficilement accessibles par l'approche SPH.

5.3.3 Initialisation

La colonne d'eau (figure 5.1) est initialement au repos et est discrétisé par des particules placées sur une grille cartésienne uniforme. Le champ de pression de ces particules devrait, en toute

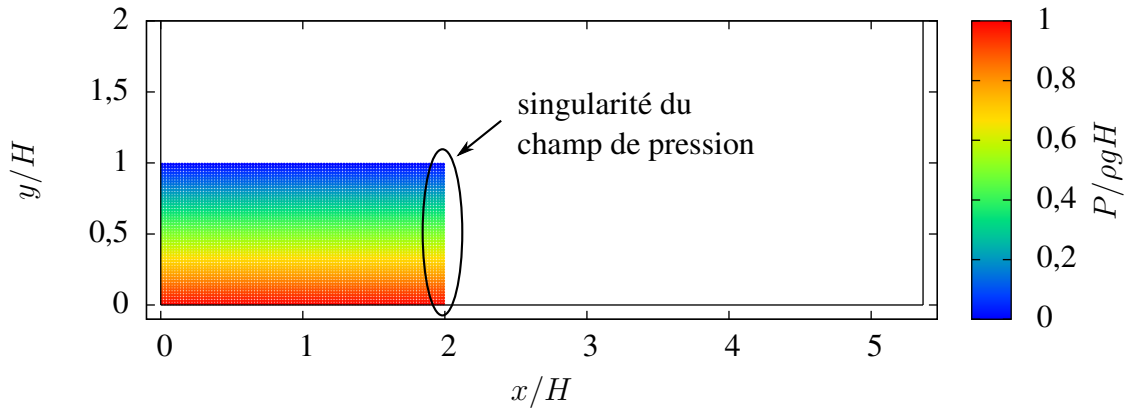


FIGURE 5.10 – Singularité du champ de pression initial quand celui-ci est choisi hydrostatique.

rigueur, respecter une répartition conforme à l'équilibre hydrostatique du fluide, soumis à l'action de la gravité. A savoir :

$$P(y) = \rho_0 g y \quad (5.27)$$

Or, si cette loi est appliquée lors de la mise en place des particules, une singularité apparaît sur la partie verticale de la surface libre initiale, quand la masse d'eau est libérée (figure 5.10). En effet, quand ce bord est libéré, la condition limite change *instantanément* pour se transformer en bord libre (pression nulle). Cette singularité du champ de pression est à l'origine d'une onde qui perturbe ensuite l'écoulement simulé. La nature faiblement compressible du modèle employé ici accentue les effets en la propageant indéfiniment dans le fluide. Il apparaît nécessaire de *lisser* ce saut de pression afin d'obtenir une solution acceptable pour cet écoulement. Colagrossi [35] (en méthode SPH) et Greco [85] (en méthode BEM) proposent d'initialiser le champ de pression en utilisant la relation suivante :

$$P(x, y) = \rho_0 g (H - y) - \frac{8\rho_0 g H}{\pi^2} \sum_{n=0}^{\infty} \left(\frac{1}{(2n+1)^2} e^{\frac{(2n+1)\pi}{2H}(x-L)} \cos\left(\frac{(2n+1)\pi}{2H}y\right) \right) \quad (5.28)$$

Cette série, à convergence rapide (une dizaine de termes suffisent à établir une approximation correcte de la solution initiale), permet de définir le champ de pression dans le cas particulier d'une masse fluide rectangulaire en deux dimensions d'espace. Elle est en fait une solution du problème de Laplace suivant (et illustré par la figure 5.11), permettant d'initialiser le champ de pression sur



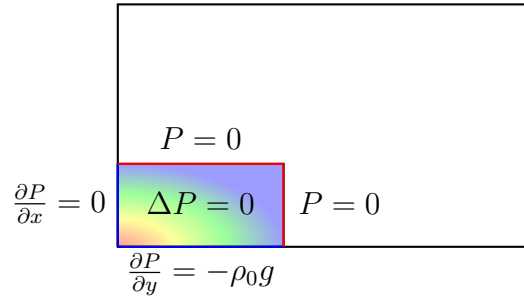


FIGURE 5.11 – Problème de Laplace pour le calcul du champ de pression initial.

les particules :

$$\left\{ \begin{array}{ll} \Delta P(x, y) = 0 & \forall (x, y) \in]0; L[\times]0; H[\\ P(L, y) = 0 & \forall y \in [0; H] \\ P(x, H) = 0 & \forall x \in [0; L] \\ \frac{\partial P}{\partial x}(0, y) = 0 & \forall y \in [0; H] \\ \frac{\partial P}{\partial y}(x, 0) = -\rho_0 g & \forall x \in [0; L] \end{array} \right. \quad (5.29)$$

Pour permettre la simulation de cas plus variés (potentiellement en trois dimensions), un solveur spécifique a été développé et validé, pour la résolution du problème (5.29) (stage ingénieur de Paul Mycek [167]). Basé sur l'approximation particulière de l'opérateur laplacien en méthode PSE, et sans maillage sous-jacent, cet algorithme emploie la méthode de Jacobi pour la résolution en parallèle du système linéaire obtenu (de taille $(N + P) \times (N + P)$ avec N , le nombre de particules à initialiser et P , le nombre de particules fantômes employées pour décrire les conditions aux bords du problème). Quand la solution (5.28) est valable, celle-ci est préférée, car plus rapide à mettre en œuvre que la résolution complète du système (5.29).

5.4 Application au cas $L/H = 2$

5.4.1 Résultats préliminaires

La figure 5.12 montre l'effondrement de la colonne d'eau, simulé à l'aide des équations (5.4)-(5.6) pour le cas $L/H = 2$, $D = \infty$ et $d/H = 5,366$. La méthode des particules fantômes a été choisie pour imposer la condition de glissement aux parois. Le rapport entre la longueur de lissage et le pas de discrétisation a été fixé à $\kappa = \epsilon/h = 1,33$ (comme pour le cas précédent). La masse fluide a été représentée par 5000 particules ($L/\epsilon = 100$ et $H/\epsilon = 50$). La vitesse numérique du son a été définie à $c_0 = 10,9\sqrt{gH}$ (identique à [41]). Le pas de temps employé est $\Delta t\sqrt{g/H} = 4,04 \times 10^{-4}$, soit un coefficient CFL de 0,166 avec un schéma d'intégration en temps

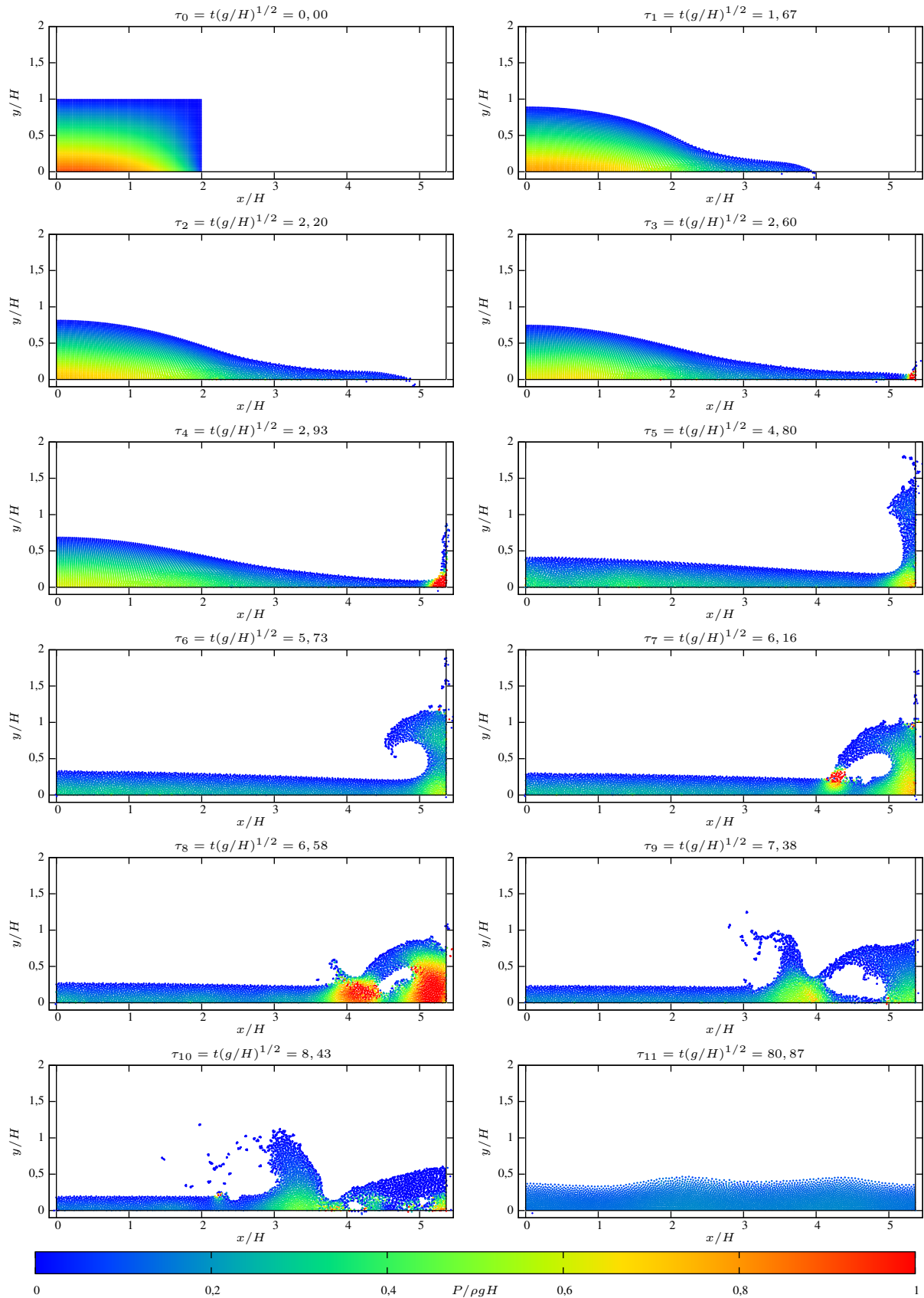


FIGURE 5.12 – Effondrement d'une colonne d'eau en deux dimensions par SPH. Vues instantanées du champ de pression.



du type Runge-Kutta 4 (voir chapitre C).

L'analyse de ces résultats préliminaires (figure 5.12) montre que le modèle permet la simulation de ce problème d'impact. Pendant la première phase de l'écoulement, le front d'eau se forme et se propage rapidement jusqu'à son impact sur le mur opposé, à $\tau_3 = t\sqrt{g/H} = 2,60$. Le front est alors dévié de sa trajectoire initiale, formant un jet vertical (τ_3, τ_4). L'effet de gravité ralentit la propagation de ce jet, qui se renverse progressivement, tel un déferlement (τ_5, τ_6). La reconnexion du front sur le fluide amont (τ_7) crée une violente onde de pression. Une *poche sans particules* est alors formée, dont l'effet non-négligeable sur la suite de l'écoulement n'est pas modélisé ici. Suivent alors une série de reconnections rapides et l'effondrement de cette *poche* (τ_8 à τ_{10}) jusqu'au retour progressif vers l'équilibre hydrostatique (ultérieur à τ_{11}).

5.4.2 Corrections

Une modélisation plus rigoureuse de cet écoulement et l'obtention de valeurs physiques telles que la vitesse de propagation du front ou la pression à l'impact nécessitent pourtant la correction de deux instabilités, visibles sur la figure 5.12.

Non-pénétration des parois

Malgré le petit pas de temps employé et l'usage d'une technique réputée fiable pour la modélisation de parois solides rectilignes, des particules parviennent à traverser les parois. D'abord, pendant la phase d'accélération du front d'eau (τ_1 à τ_3), puis, après impact sur la paroi verticale opposée (τ_4 à τ_{10}). Nous avons donc mis en place une correction supplémentaire de la condition de non-pénétration aux parois.

Le problème du respect des conditions de non-pénétration aux parois est souvent rencontré dans la communauté. Il fait l'objet de nombreuses discussions au cours des conférences. Chaque numéricien a en général sa propre solution qui consiste soit au choix d'une des méthodes de modélisation présentées plus haut plutôt qu'une autre, ou bien à l'ajout d'un artifice algorithmique permettant de restaurer cette condition. Les méthodes des forces répulsives, des particules fixes et dynamiques assurent qu'aucune particule ne puisse traverser la paroi. Dans le cas des forces répulsives, à croissance exponentielle quand les particules approchent de la paroi, aucune d'entre elles ne peut effectivement la traverser. En revanche, cette méthode souffre d'un manque de précision évident sur la condition imposée, et on observe d'ailleurs souvent une élévation artificielle des particules qui représentent le front quand celui-ci se forme en début de simulation. Avec les particules fixes ou dynamiques, la présence de masse particulaire de l'autre côté de la paroi empêche effectivement les particules de la traverser ; mais encore une fois, la condition en vitesse n'est qu'approximative.

Dans le cas des particules fantômes, la masse est répartie en fonction des particules fluides, à chaque itération. Une conséquence est qu'une particule se rapprochant de la paroi se voit attribuer

une image par symétrie, elle-aussi de plus en plus proche de la paroi. Si le pas de temps est suffisamment petit, cela permet d'éviter la traversée du bord par la particule. En revanche, si le pas de temps est trop grand, cela peut violer la condition au bord. La particule n'est alors plus symétrisée et l'absence subite d'image pour cette particule produit des instabilités, qui se matérialisent souvent sous la forme d'ondes de pression qui se propagent dans le fluide. Les solutions à ce problème sont variées, et souvent passées sous silence. Certains proposent de simplement réduire drastiquement le pas de temps de façon à s'assurer qu'aucune particule ne traverse ; l'impact de cette approche sur les temps de simulations est très important. D'autres proposent d'échanger la particule fluide et son image quand la première a traversé la paroi (la particule fluide est alors considérée comme une image et inversement), la mise en œuvre algorithmique de ce procédé peut être plus ou moins compliquée selon la manière dont les symétries ont été codées. On peut aussi envisager d'annuler la composante normale de la vitesse d'une particule jugée trop proche de la paroi, l'effet sur la conservation de l'énergie peut être significatif.

Dans le cas présent, nous avons choisi d'appliquer une force répulsive entre une particule fluide et son image quand celles-ci se trouve à une distance inférieure à $\epsilon/2$ l'une de l'autre. Cette correction, inspirée de l'équation (5.7), s'écrit donc :

$$\mathbf{f}_{iG_i} = \begin{cases} gH \left[\left(\frac{\epsilon}{2r_{iG_i}} \right)^4 - \left(\frac{\epsilon}{2r_{iG_i}} \right)^2 \right] \frac{\mathbf{x}_{iG_i}}{r_{iG_i}^2} & \text{si } \left(\frac{\epsilon}{2r_{iG_i}} \right) \leq 1 \\ 0 & \text{si } \left(\frac{\epsilon}{2r_{iG_i}} \right) > 1 \end{cases} \quad (5.30)$$

où r_{iG_i} est la distance entre la particule fluide i et son image G_i . Contrairement à la méthode des forces répulsives pour laquelle une particule fluide subit l'action de plusieurs marqueurs de la paroi, ici, une seule force est calculée entre la particule fluide et son image. La force ainsi appliquée est donc normale à la paroi, et n'influence la particule que si celle-ci se trouve à une distance inférieure à $\epsilon/4$. Ce procédé, bien que sans interprétation physique, est simple à mettre en œuvre, efficace et intervient rarement lors d'une simulation. Cependant, son effet est suffisant pour éviter toute traversée des parois, qui influencerait directement la suite de la simulation et la conservation de l'énergie totale.

Flux Rusanov

Après un certain temps, la masse fluide retrouve un équilibre hydrostatique que nous n'avons pas atteint en raison du nombre très important d'itérations nécessaires. Cependant, nous nous sommes rendus compte que, au fur et à mesure que les vitesses observées dans le fluide se réduisaient, un flux vertical de masse se mettait progressivement en place. Partant du fond de la cuve vers les particules de surface, ce flux tendait à égaliser le champ de pression, sans raison physique (τ_{11}). Une conséquence directe est l'accroissement du volume des particules de surface, qui semblent *léviter* au dessus des autres particules. Ce phénomène a été attribué au Flux Rusanov



employé pour stabiliser le schéma, qui ne tenait pas compte de l'effet de la gravité sur la masse volumique et tendait à surestimer la correction intervenant dans l'équation (équation (5.5)).

Comme mentionné dans le §4.5.1, ce flux lisse les différences de masse volumique entre particules voisines. Or, en vision faiblement compressible, ce sont justement ces petites variations de la masse volumique qui permettent d'obtenir la valeur du champ de pression. D'où l'effet observé ici du champ de pression peu à peu égalisé en même temps que les sauts de masse volumique le sont. Nous avons remédié à ce problème en limitant le flux Rusanov.

Quand une particule fluide est symétrisée selon la technique des particules fantômes, une compensation hydrostatique intervient afin d'appliquer la condition adéquate en pression à la paroi (équation (5.22)). De la même manière, pour le cas présent, nous avons considéré qu'un saut de masse volumique entre deux particules voisines ne devait être lissé que si celui-ci était supérieur à la *différence hydrostatique* suivante :

$$\Delta\rho_{ij} = \frac{\delta\rho_{ij} + \delta\rho_{ji}}{2}, \quad (5.31)$$

avec

$$\delta\rho_{ij} = \left[\rho_i^{\gamma-1} + \frac{\rho_0^{\gamma-1}(\gamma-1)}{c_0^2} g(y_j - y_i) \right]^{\frac{1}{\gamma-1}} - \rho_i \quad (5.32)$$

L'équation de conservation de la masse devient alors :

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \left(\mathbf{u}_{ij} + \mathbf{n}_{ij} \left(\frac{c_{ij}}{\rho_i} (\rho_j - \rho_i - \Delta\rho_{ij}) \right) \right) \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (5.33)$$

Les deux corrections précédentes ont été appliquées à l'effondrement de la colonne d'eau. La figure 5.13 montre qu'aucune particule ne traverse les parois (τ_1 à τ_{10}) et que le retour à l'équilibre hydrostatique n'est pas perturbé (τ_{11}). Les données simulées sont maintenant exploitables pour le calcul de valeurs plus ciblées, telles que la position du front, la pression à l'impact ...

Influence de la discrétisation

Dans le cas étudié précédemment (de la goutte étirée en deux dimensions), une initialisation des particules en cercles concentriques était plus adaptée à la dynamique de l'écoulement et permettait de limiter la dégradation de la solution au cours de la simulation. Dans le cas présent, par sa forme rectangulaire, la masse fluide initiale est discrétisée par une grille cartésienne uniforme. Le seul paramètre influent sur l'issue de la simulation est donc le nombre de particules employées.

Pour les résultats exposés dans le paragraphe suivant (§5.4.3), la rupture de barrage a été simulée à l'aide de trois discrétisations différentes. Outre l'impact que cette augmentation du nombre de particules peut générer sur la précision des calculs effectués et des données obtenues par post-traitement dans le fluide, c'est aussi la forme de la surface libre qui est directement influencée

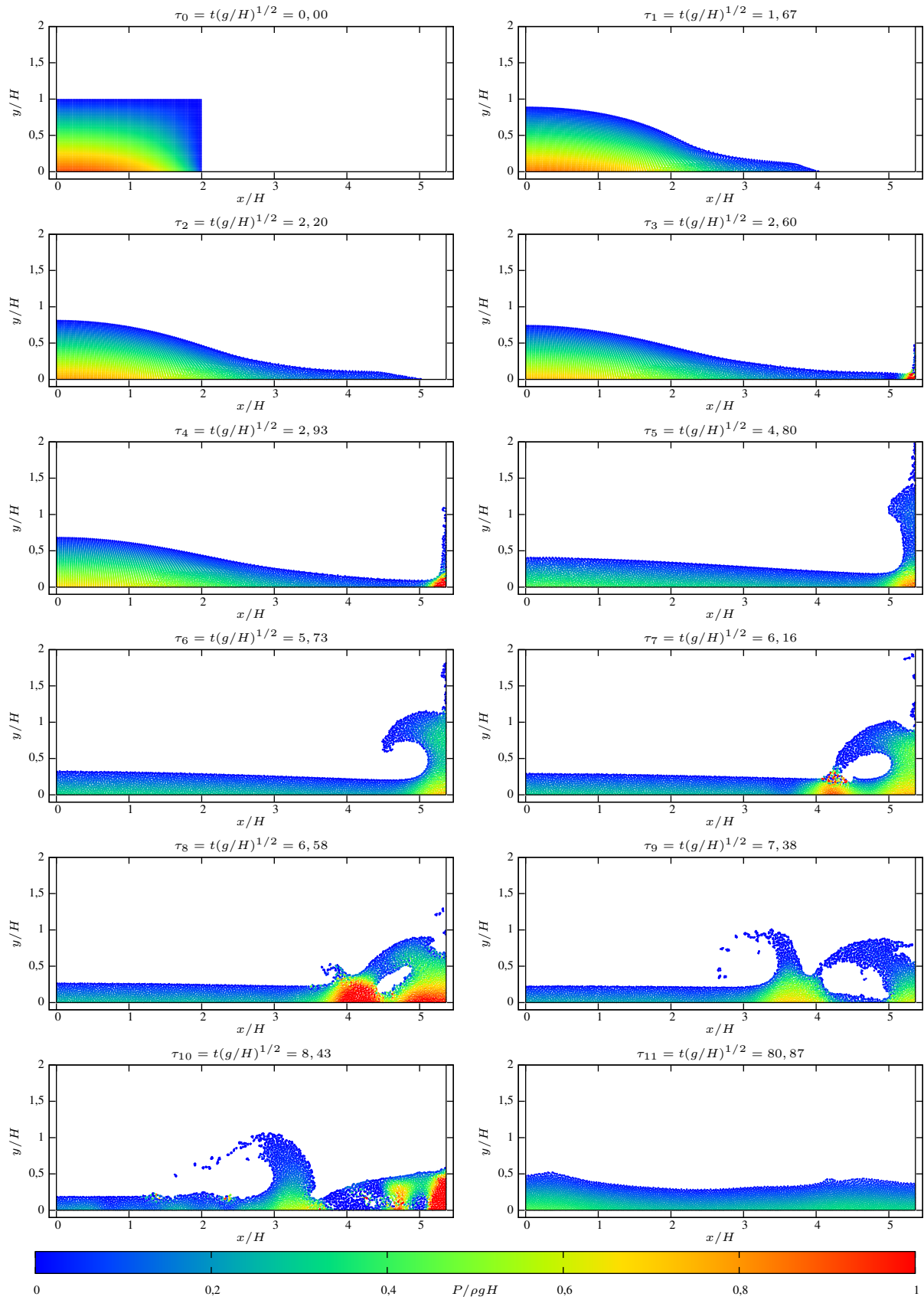


FIGURE 5.13 – Effondrement d’une colonne d’eau en deux dimensions par SPH intégrant la correction de la condition au bord et du flux Rusanov. Vues instantanées du champ de pression.



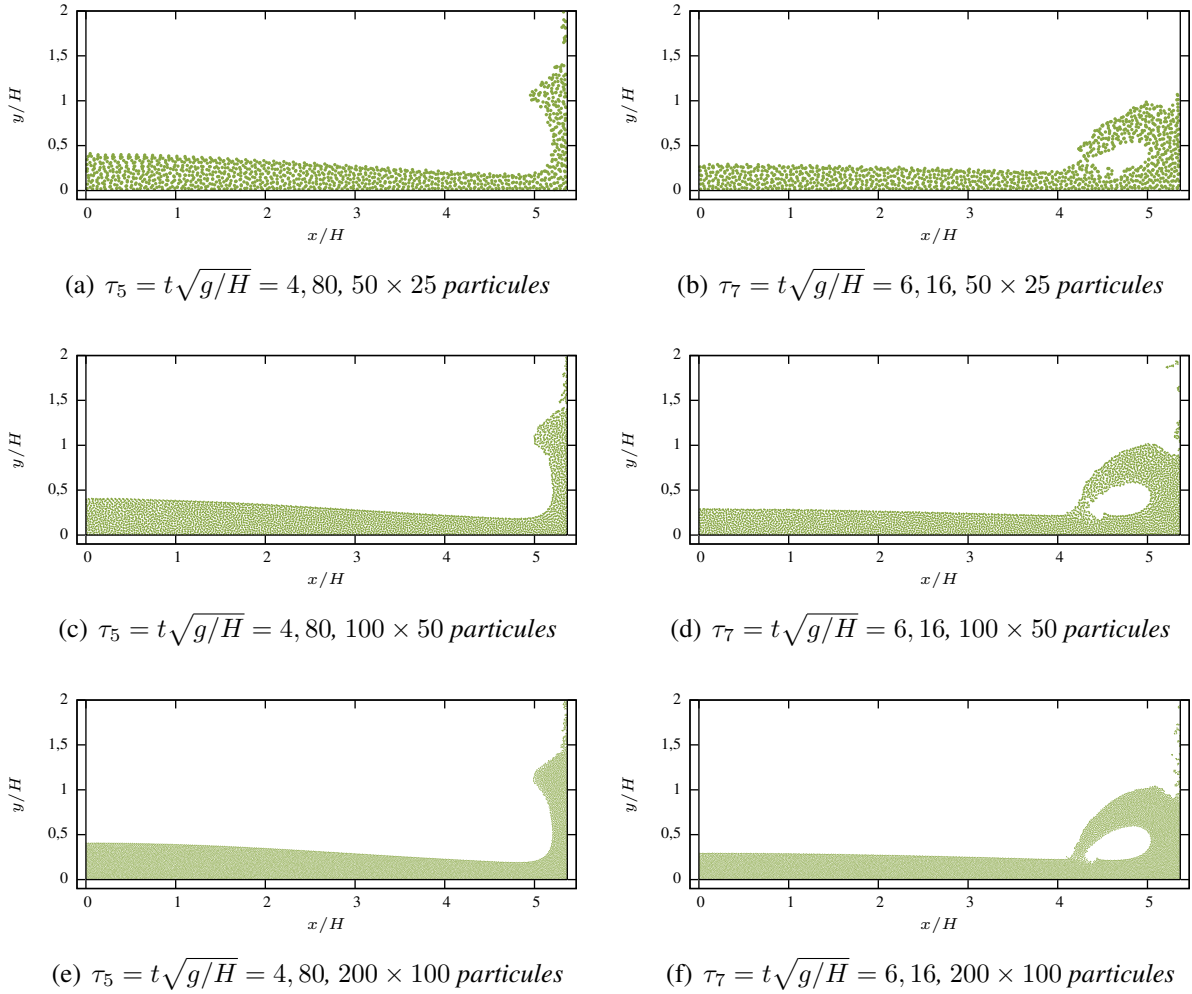


FIGURE 5.14 – Effondrement d’une colonne d’eau en deux dimensions par SPH. Influence de la discrétisation sur la forme de la surface libre.

par ce raffinement. La figure 5.14 montre la forme de la surface libre aux instants $\tau_5 = 4, 80$ et $\tau_7 = 6, 16$, pour trois tailles de particules différentes. Les résultats obtenus avec une grille grossière de 50×25 particules montrent une déformation importante du jet vertical (Fig. 5.14-(a)), puis de la poche juste après reconnexion (Fig. 5.14-(b)), par rapport aux résultats obtenus avec des grilles plus fines (Fig. 5.14-(c-f)). Aucune différence significative sur la position des particules n’est observable entre la simulation avec 100×50 particules et celle impliquant 200×100 particules.

Conservation de l’énergie

Les méthodes SPH, par définition, lissent les variables transportées à chaque interpolation ou approximation d’opérateur différentiel. Bien qu’écrites sous forme conservative, les équations discrètes nécessitent d’être stabilisées par l’adjonction d’un terme visqueux artificiel (§4.4.2) ou du

flux Rusanov employé ici (§4.5.1). L'effet indésirable d'une viscosité artificielle est la dissipation de l'énergie du fluide simulé. C'est pour remédier à ce problème que Colagrossi et Landrini [41] ont proposé l'usage combiné d'une nouvelle forme de viscosité artificielle (équation (4.12)) et d'une réinitialisation périodique de la masse volumique (équation (4.16)). Correctement paramétré, son modèle réduit considérablement la dissipation de l'énergie. Afin, d'évaluer et comparer cette donnée pour le modèle présent, l'énergie totale a été calculée :

$$E_{\text{TOT}} = \underbrace{\frac{1}{2} \sum_i m_i \|\mathbf{u}_i\|^2}_{\text{énergie cinétique}} + \underbrace{g \sum_i m_i y_i}_{\text{énergie potentielle}} + \underbrace{\sum_i m_i e_i}_{\text{énergie interne}} \quad (5.34)$$

Les deux premiers termes, correspondant respectivement à l'énergie cinétique et à l'énergie potentielle, sont aisément calculables à partir des valeurs connues sur les particules. En revanche, le dernier terme, lié à la compressibilité (même faible) du fluide requiert l'intégration d'une équation supplémentaire :

$$\frac{De_i}{Dt} = -\frac{P_i}{\rho_i} (\nabla \cdot \mathbf{u}_i), \quad (5.35)$$

discrétisée de la manière suivante :

$$\frac{De_i}{Dt} = \frac{P_i}{\rho_i} \sum_j \mathbf{u}_{ij} \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j}, \quad (5.36)$$

L'intégration de cette équation nécessite le stockage d'une variable supplémentaire pour chaque particule. Le surcoût en mémoire et temps de calcul se justifie cependant par l'intérêt lié à l'observation de cette variable, qui est un bon indicateur de l'état d'une simulation en cours.

La figure 5.15 montre la variation au cours du temps de l'énergie totale du système, adimensionnée de la façon suivante (proposée par Colagrossi et Landrini [41]) :

$$\frac{E_{\text{TOT}} - E_{\text{TOT0}}}{\Delta E} \quad (5.37)$$

avec ΔE , la différence entre l'énergie potentielle à l'état initial et à l'état final (retour à l'équilibre hydrostatique). Globalement, la dissipation de l'énergie est inférieure à 3%, quelque soit la discrétisation employée, ce qui est déjà bien inférieur à la dissipation observée avec une viscosité artificielle *classique* (équation (4.10)), avec laquelle la dissipation atteint 20% à $t\sqrt{g/H} = 8,0$ (non représentée ici). A résolution comparable (100×50), le modèle SPH employé ici est légèrement moins dissipatif que celui de Colagrossi et Landrini [41]. L'augmentation du nombre de particules réduit significativement cette dissipation ($< 0,5\%$ pour 200×100 particules).



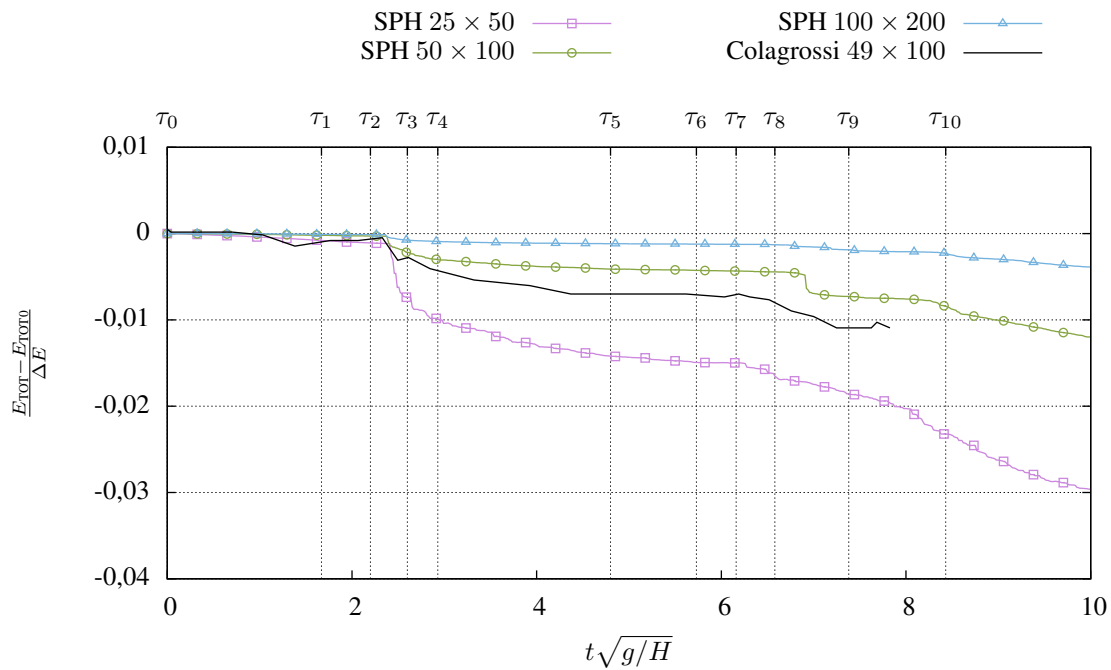


FIGURE 5.15 – Évolution de l'énergie totale au cours de l'effondrement d'une colonne d'eau en deux dimensions. Comparaison entre la meilleure solution de Colagrossi et Landrini [41] (en noir), qui emploie une grille de 100×49 particules et la solution présente avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).

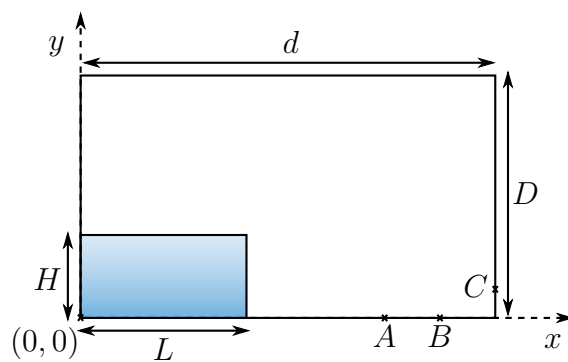


FIGURE 5.16 – Positions de capture des données de comparaison. Les points A $(3,721H; 0)$ et B $(4,542H, 0)$ correspondent aux mesures de hauteur d'eau. Le point C $(d; 0, 192H)$.

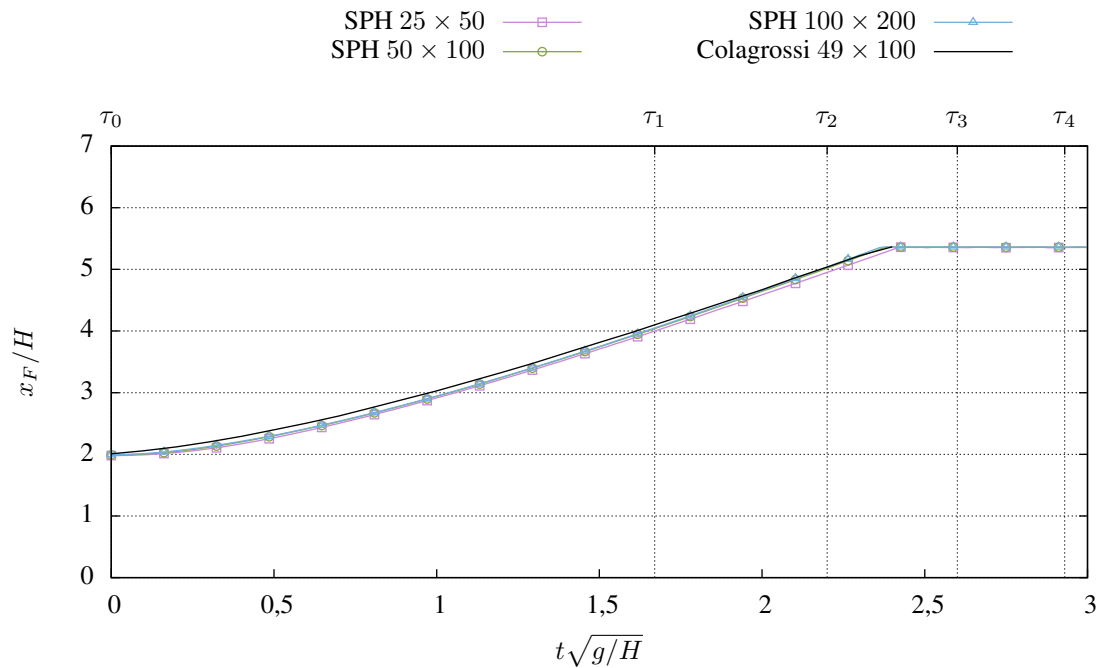


FIGURE 5.17 – Évolution de la position du front d'eau. Comparaison entre la solution SPH de référence fournie par Colagrossi et Landrini [41] (en noir) et la solution selon le modèle SPH présent avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).

5.4.3 Post-traitement et validation

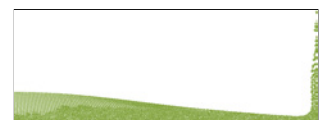
Pour ce cas particulier d'un effondrement d'une masse avec un rapport d'aspect $L/H = 2$, les seules données expérimentales employées couramment dans la littérature sont issues des travaux de Zhou *et al.* [240]. La solution SPH de référence a été proposée par Colagrossi et Landrini [41], qui la compare également à ces mesures. La figure 5.16 montre le placement des différentes mesures fournies par Zhou *et al.* :

- La hauteur d'eau mesurée aux points *A* et *B*.
- La pression sur le mur, mesurée au point *C*.

Toutes ces données sont calculées après simulation, par un code de post-traitement spécifique, dont chaque algorithme est détaillé ci-après. Les techniques de post-traitement diffèrent souvent, il existe d'ailleurs plusieurs travaux spécifiquement consacrés aux méthodes d'extraction de données et à leur visualisation [104, 143, 144, 174].

Position du front

La position du front d'eau peut être approchée en déterminant, à chaque étape du post-traitement, la particule dont la composante x de la position est maximale. L'erreur entre cette approximation et la position réelle du front est inférieure à h , l'espace initial et dépend de l'étirement de cette particule. Cette approche est très simple et rapide. Nous aurions pu exploiter la même technique de détection de position de l'interface que celle proposée par l'équation (4.29) (basée sur la norme du



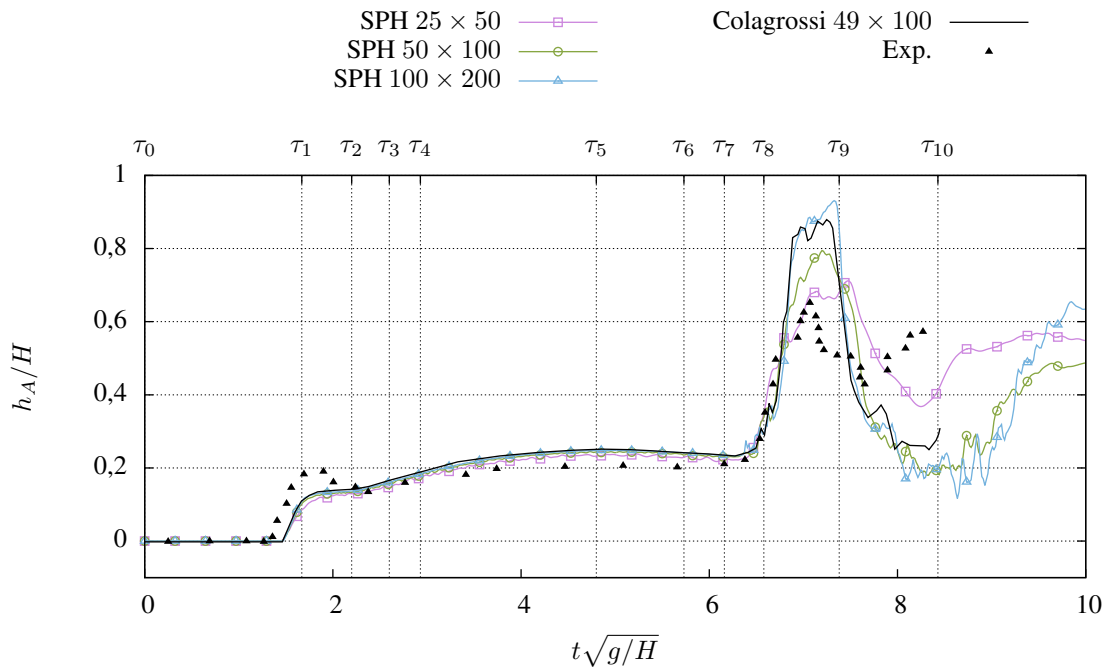


FIGURE 5.18 – Évolution de la hauteur d'eau calculée au point A (figure 5.16). Comparaison entre les données expérimentales de Zhou et al. [240] (triangles noirs), la solution SPH de référence fournie par Colagrossi et Landrini [41] (en noir) et la solution selon le modèle SPH présent avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).

noyau), mais la forme en *pointe* du front d'eau aurait conduit à une sous-estimation systématique de la position du front. L'évolution de cette grandeur est représentée par la figure 5.17, pour trois résolutions différentes comparées à la solution de référence de Colagrossi et Landrini [41]. Aucune donnée expérimentale n'est disponible dans le cas $L/H = 2$. Les résultats fournis par le modèle actuel convergent vers une solution unique qui affiche un léger retard par rapport à la solution de référence. Les données présentées ici ont été dérivées afin d'accéder à la vitesse de propagation du front et de la comparer avec sa limite théorique \sqrt{gH} , les résultats sont exposés en annexe (§D).

Hauteur d'eau

La hauteur d'eau a été relevée, au cours du temps, en deux points *A* et *B* (figure 5.16). Un test basé sur l'équation (4.29) pourrait suffire à déterminer la position de la surface libre et à comparer cette valeur à des données expérimentales issues de sondes résistives. Or, pour les mesures de Zhou *et al.* [240], les sondes employées sont capacitives. Ces sondes fournissent une longueur "immergée" dans le fluide. Ainsi, toute bulle éventuellement emprisonnée dans le fluide n'est pas comptée. Le calcul de cette hauteur immergée par le code de post-traitement est réalisé en discrétisant la verticale au point de mesure en petits segments h . Au milieu de chacun de ces segments, la somme (5.38) est évaluée. On additionne ensuite les longueurs de tous les segments pour lesquels

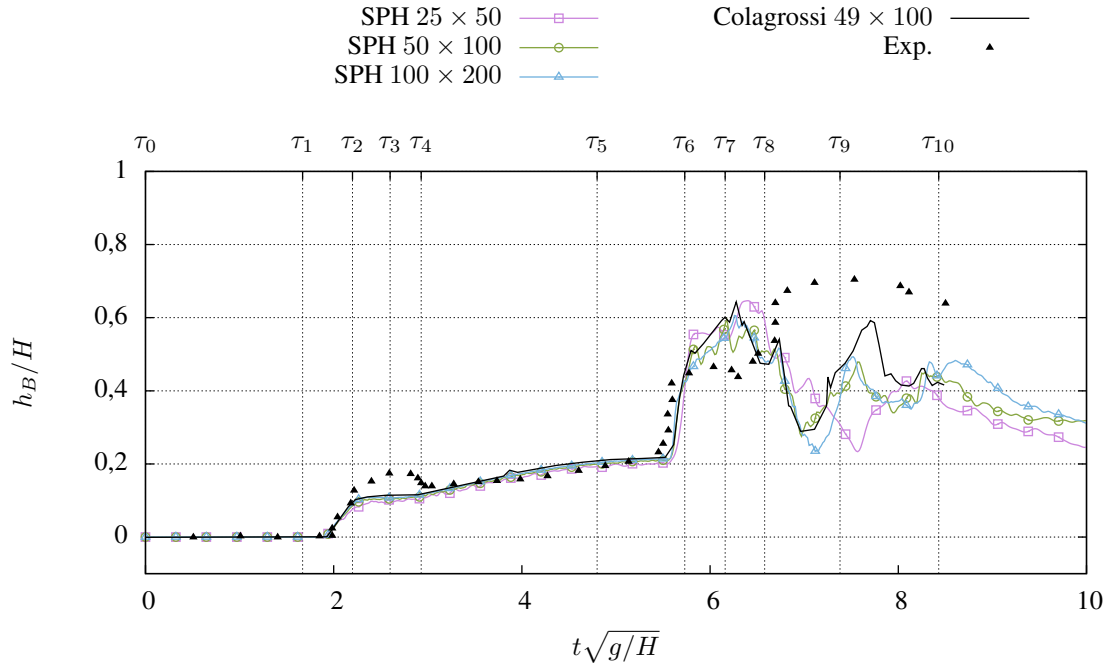


FIGURE 5.19 – Évolution de la hauteur d'eau calculée au point B (figure 5.16). Comparaison entre les données expérimentales de Zhou et al. [240] (triangles noirs), la solution SPH de référence fournie par Colagrossi et Landrini [41] (en noir) et la solution selon le modèle SPH présent avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).

cette somme dépasse 0,5.

$$S(\mathbf{x}) = \sum_j \eta_e^0(\mathbf{x} - \mathbf{x}_j) V_j \quad (5.38)$$

Les figures 5.18 et 5.19 montrent l'évolution des hauteurs aux points A et B , pour trois résolutions différentes. Au point A , on peut noter un bon accord de toutes les solutions entre les instants τ_2 et τ_8 . Avant τ_2 , les données de Zhou *et al.* [240] montrent une surélévation par rapport aux simulations, due certainement aux conditions initiales expérimentales ; l'auteur précise d'ailleurs que d'un essai à l'autre, les mesures varient sensiblement. A partir de τ_8 , la forme du jet vertical (figure 5.13) diffère selon la discrétisation employée. A 100×50 particules, notre solution sous-estime la hauteur prédite par Colagrossi et Landrini [41] (obtenue avec une discrétisation équivalente). En augmentant le nombre de particules, seule la hauteur de ce jet (entre τ_8 et τ_9) est modifiée ; la tendance globale, elle, reste inchangée. Au point B , le même écart entre données expérimentales et simulations peut être observé au passage du front (τ_1 à τ_4). La suite de la simulation montre un bon accord entre le modèle actuel et celui de Colagrossi et Landrini [41]. Les mesures de Zhou *et al.* [240] sont différentes, quelque soit la position de la sonde capacitive, à partir de τ_8 .



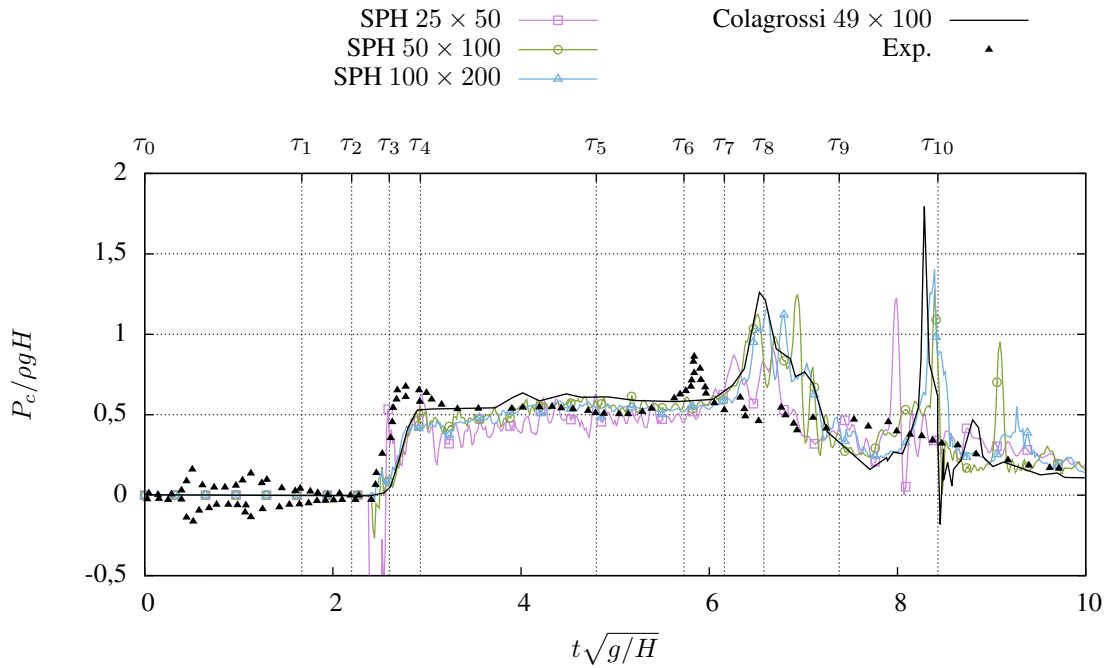


FIGURE 5.20 – Évolution de la pression calculée au point $C(d; 0, 192H)$ (figure 5.16). Comparaison entre les données expérimentales de Zhou et al. [240] (triangles noirs), la solution SPH de référence fournie par Colagrossi et Landrini [41] (en noir) et la solution selon le modèle SPH présent avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).

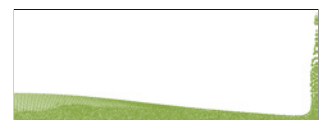
Pression à l'impact

Plusieurs approches existent pour extraire la pression en un point donné à partir des résultats d'une simulation [172]. Nous avons choisi ici une simple interpolation normalisée, en tenant compte des particules fantômes :

$$P(\mathbf{x}_c) = \frac{\sum_j P_j \eta_\epsilon^0(\mathbf{x}_c - \mathbf{x}_j) V_j}{\sum_j \eta_\epsilon^0(\mathbf{x}_c - \mathbf{x}_j) V_j} \quad (5.39)$$

Les résultats obtenus sont représentés sur la figure 5.20. La pression sur le mur est d'abord nulle (ou égale à la pression atmosphérique) jusqu'au moment de l'impact et du passage du jet vertical au niveau de la sonde (τ_3 à τ_4). La pression augmente alors brutalement jusqu'à une valeur maintenue qui décroît progressivement au fur et à mesure que les vitesses au coin diminuent (à partir de τ_4). On remarque cependant que plusieurs pics apparaissent ensuite sur les résultats des simulations numériques. À τ_8 , le retournement du jet vertical sur la surface libre incidente crée un violent choc de pression qui se propage jusqu'au mur et apparaît sur les données issues de la sonde. De même, à τ_{10} , la poche (contenant en pratique de l'air, mais non modélisé ici) disparaît, formant un second pic violent. Les expériences ne montrent pas ces pics de pression (ou un léger pic entre τ_6 et τ_7). Cela pourrait s'expliquer par la présence de ces cavités, remplies d'air, qui amortissent chacun des deux impacts mentionnés précédemment. Colagrossi et Landrini [41]

montrent d'ailleurs que la modélisation de la phase air en SPH change significativement les résultats et que ces pics de pression disparaissent alors. La valeur obtenue pour le palier entre τ_4 et τ_7 est en bon accord avec les données de comparaison. Il est à noter que pour le post-traitement des résultats présentés ici, comme pour la solution de référence, le point d'interpolation C de coordonnées $(d; 0, 192H)$ diffère légèrement de la position réelle de la sonde dans les expériences de Zhou *et al.* [240]. Le transducteur employé est placé à $(d; 0, 267H)$ et a un diamètre de $9\text{cm} \approx 0,15H$ (avec $H = 60\text{cm}$). Nous avons donc placé la sonde numérique au point le plus bas de la sonde expérimentale, de manière à se placer dans les mêmes conditions que la solution de référence de Colagrossi et Landrini [41]. La position des pics diffère légèrement entre la solution du modèle présent et la solution de référence. Nous attribuons ce décalage au schéma d'intégration en temps employé, à pas variable dans le cas de la solution de référence tandis que le schéma employé ici est à pas constant. Quelques oscillations violentes apparaissent aussi entre τ_2 et τ_3 malgré le fait que la courbe présentée soit lissée par moyenne glissante sur 5 points. Le terme de normalisation (équation (5.39)), dont les valeurs sur cet intervalle sont très faibles, amplifie ces oscillations.



Chapitre 6

Méthode de la frontière immergée et application aux écoulements visqueux

Comme nous l'avons évoqué dans le chapitre 5, les modèles de conditions aux limites proposés pour les frontières solides du domaine se montrent souvent inadaptés à la prise en compte de corps immergés, ou insuffisamment précis pour une étude quantitative des efforts subis par ces objets dans le fluide. Un modèle de conditions aux limites a donc été développé, basé sur les méthodes de *frontières immergées* en différences finies, permettant la prise en compte de corps fixes ou mobiles, aux formes potentiellement complexes. Ce modèle a été inspiré par les travaux de Hieber et Koumoutsakos [93], qui proposent une adaptation de la méthode originale des frontières immergées à la méthode SPH remaillée (RSPH) [26], où les particules sont repositionnées sur une grille cartésienne à chaque itération de manière à maintenir une précision acceptable lors de l'approximation des équations simulées. Le modèle associant *frontière immergée* et *remaillage*, proposé par Hieber et Koumoutsakos [93], a donc été développé. La technique du remaillage, également détaillée dans ce qui suit, s'est avérée, dans un premier temps, indispensable à la méthode de frontière immergée, en raison des nombreuses interpolations successives des particules vers les points définissant la frontière à modéliser et inversement. Or, notre objectif étant la modélisation de la surface libre, atout certainement le plus intéressant de la méthode SPH, le remaillage remettait en cause l'utilisation d'un tel modèle. On se retrouvait avec les mêmes difficultés que les autres méthodes quand il s'agissait de capter la surface libre sur une grille. Une variante de la méthode de Hieber et Koumoutsakos [93] a donc été développée, permettant de s'affranchir du remaillage lors de la mise en œuvre de la frontière immergée. Ainsi, le modèle que nous proposons ici permet, comme nous allons le voir, de conserver le caractère lagrangien de la méthode et ses qualités pour le suivi des déformations de la surface libre, et de modéliser de façon précise une condition d'adhérence sur des corps immergés dans le fluide, sans remaillage.

L'écoulement d'un carré forcé dans un fluide visqueux incompressible, en deux dimensions d'espace, a été étudié pour validation de ce modèle. Pour ce cas, la méthode SPH est donc appliquée aux équations de Navier-Stokes.



6.1 Méthode de la *frontière immergée*

La méthode dite de la *frontière immergée* a été proposée, en 1977, par Peskin [183] qui l'emploie pour la modélisation de l'écoulement sanguin dans le coeur. Cette application montre les nombreuses possibilités offertes par ce nouveau modèle de condition aux limites en différences finies. Elle permet en effet de mettre en œuvre des conditions aux parois flexibles sans qu'elles ne coïncident avec le maillage cartésien sous-jacent. Le bord γ est modélisé par un ensemble de points, associés à une surface sur laquelle est calculée l'action du fluide. Une loi de comportement de la paroi permet alors de calculer le mouvement effectif de celle-ci sous l'action du fluide. La vitesse en chaque point de frontière permet alors de calculer l'action que doit avoir la paroi sur le fluide pour satisfaire le principe de réciprocité. La *frontière immergée* est donc à l'origine un modèle de couplage fluide-structure en méthode eulérienne où le corps est représenté par des marqueurs lagrangiens. Les forces sont transmises du maillage aux marqueurs par l'intermédiaire d'une approximation de la mesure de Dirac $\delta_0()$.

De nombreux travaux de recherche ont suivi, essentiellement dans le domaine de la simulation de fluides en biologie. Dans un premier temps, les études ont porté sur la mise en œuvre mathématique du modèle, sur les approximations de la fonction $\delta_0()$, les critères de stabilité et les lois de comportement des parois . . . [182, 220]. La bibliographie est très abondante sur cette méthode et ses variantes. On trouve notamment les travaux de Roma *et al.* [195], puis Taira et Colonius [221], sur la combinaison d'une méthode frontière immergée à un schéma d'intégration en temps par projection ; le gain en précision y est important. Lai et Peskin [114] proposent également un schéma en temps spécifique permettant d'atteindre le second ordre en précision. La méthode a ensuite été adaptée aux nouvelles techniques de modélisation en différences finies par raffinement automatique de maillage [195, 87, 213], la principale difficulté est alors de mettre en œuvre un modèle de transfert précis des grandeurs de la frontière vers la grille et *vice versa*, imposant un choix souvent arbitraire (et pas toujours justifié) des mailles du support pour ces interpolations.

L'extension de la méthode à la simulation d'écoulements autour de corps solides a également fait l'objet de plusieurs travaux. Ainsi, Goldstein *et al.* [80] s'y sont intéressés et ont proposé une approche dite à *rétro-action*. Ensuite, la méthode de *forçage direct* proposée par Fadlun *et al.* [69], a été adoptée par d'autres auteurs [225, 7, 226, 216, 235].

Un point faible de la méthode appliquée aux corps rigides serait le fait que seule une condition en vitesse est imposée à la frontière immergée. Ainsi, dans les résultats obtenus, il y a continuité des contraintes à travers la paroi modélisée, ce qui n'est évidemment pas toujours le cas, en réalité. La combinaison de la méthode aux schémas de projection et la prise en compte de *sauts* ont donné naissance à une nouvelle méthode, nommée *interface immergée* [119, 120, 239]

Enfin, la *frontière immergée* a été étendue à d'autres classes de méthodes numériques : éléments finis [136], [110], VOF [208] puis SPH [93]. On trouve également des extensions de la méthodes aux formulations vitesse-tourbillon [44, 64, 185].

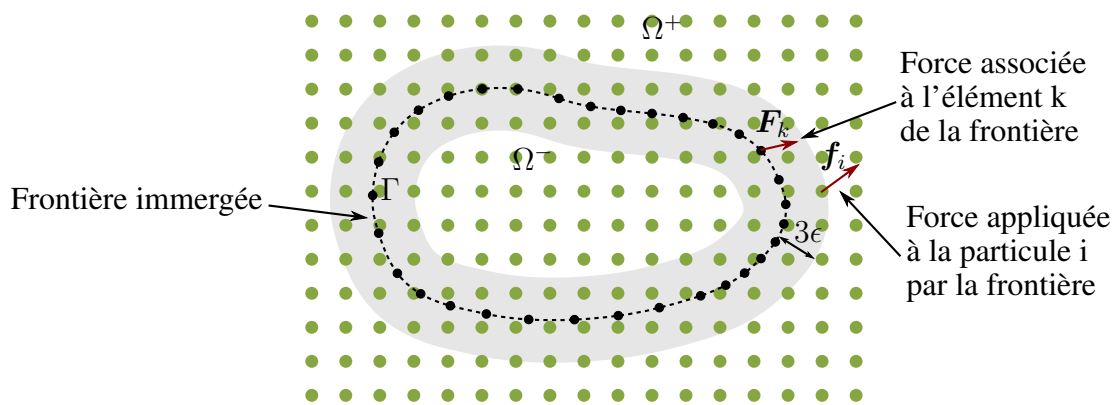


FIGURE 6.1 – Représentation d'un corps immergés dans le fluide par la méthode de la frontière immergée.

6.1.1 SPH remaillée et *frontière immergée*, méthode de Hieber et Koumoutsakos [93] : RSPH-IB

Frontière immergée et forçage direct

L'utilisation de la *frontière immergée* dans SPH a été proposée par Hieber et Koumoutsakos [93], dans la lignée directe de leurs travaux sur les méthodes *level-set* [92], la méthode SPH remaillée (RSPH) [29] et les simulations en particule-maillage [199]. La prise en compte des conditions aux limites de type adhérence s'inspire directement des travaux de Peskin [183]. Dans [93], l'interface est définie par des marqueurs lagrangiens dans le fluide et une loi de comportement leur permet de déterminer la vitesse à imposer sur la paroi. Ici, l'objectif étant l'application de la méthode à des corps rigides au déplacement prédéfini, nous présentons la version adaptée à ce contexte, c'est à dire sans la simulation du comportement mécanique de la paroi. De plus, afin de n'observer que les effets de la mise en œuvre numérique de la *frontière immergée* sur les résultats obtenus, nous avons choisi de conserver l'écriture des équations de Navier-Stokes telle qu'elle a été établie au cours des chapitres précédents. Dans la méthode originale de Hieber et Koumoutsakos [93], un noyau spécifique est employé pour la diffusion de la quantité de mouvement et la loi d'état adoptée est celle correspondant au comportement d'un gaz parfait.

Leur méthode de *frontière immergée*, dite par *spreading* (qui traduit le fait qu'on emploie deux interpolations successives des particules vers la frontière et inversement), s'inspire du forçage direct [69] : elle consiste à calculer des forces de corrections, ou *pénalités*, nécessaires à l'application de la condition souhaitée en vitesse (figure 6.1). Ainsi, la paroi Γ séparant le domaine fluide en une partie extérieure Ω^+ et intérieure Ω^- est représentée par un ensemble de marqueurs x_k sur lesquels une force F_k doit être calculée à partir des particules du fluide et de la vitesse à imposer. Cette force F_k est ensuite répartie sur les particules voisines x_i sous la forme de termes de forçage notés f_i . Ces forces de volume f_i interviennent directement dans les équations de Navier-Stokes :



$$\frac{D\mathbf{x}_i}{Dt} = \mathbf{u}_i \quad (6.1)$$

$$\frac{D\mathbf{u}_i}{Dt} = \mathbf{rhs}_i + \mathbf{f}_i \quad (6.2)$$

$$\frac{D\rho_i}{Dt} = \rho_i \sum_j \left(\mathbf{u}_{ij} + \mathbf{n}_{ij} \left(\frac{c_{ij}}{\rho_i} (\rho_j - \rho_i - \Delta\rho_{ij}) \right) \right) \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} \quad (6.3)$$

$$P_i = P_0 + \frac{\rho_0 c_0^2}{\gamma} \left(\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (6.4)$$

$$+ C.L. \quad (6.5)$$

L'équation d'état 6.4 fait intervenir une pression de référence P_0 , généralement choisie nulle lors de simulations d'écoulements à surface libre ; la valeur attribuée à ce paramètre est évoquée plus loin. L'équation de conservation de la masse (équation (6.3)) intègre le flux Rusanov corrigé dans le chapitre précédent (§5.4.2). L'équation de conservation de la quantité de mouvement (équation (6.2)) a été écrite de manière à distinguer la partie du membre droit propre à la partie fluide de l'écoulement, \mathbf{rhs}_i , du terme de forçage \mathbf{f}_i symbolisant l'influence de la condition limite. Le terme \mathbf{rhs}_i s'écrit de la manière suivante :

$$\mathbf{rhs}_i = -\frac{1}{\rho_i} \sum_j (P_i + P_j) \nabla \eta_\epsilon^0(\mathbf{x}_{ij}) \frac{m_j}{\rho_j} - \sum_j m_j \frac{2\mu}{\rho_i \rho_j} \mathbf{u}_{ij} \frac{\mathbf{x}_{ij} \cdot \nabla \eta_\epsilon^0(\mathbf{x}_{ij})}{r_{ij}^2} + \mathbf{g} \quad (6.6)$$

où la première somme correspond à l'approximation du gradient de pression et la seconde au terme visqueux établi au §3.3. En discrétisant la dérivée temporelle de l'équation (6.2) par un simple quotient différentiel, on a :

$$\frac{\mathbf{u}_i^{n+1} - \mathbf{u}_i^n}{\Delta t} = \mathbf{rhs}_i + \mathbf{f}_i \quad (6.7)$$

La force appliquée à l'élément k de la frontière Γ , notée \mathbf{F}_k , par l'ensemble des particules voisines du fluide $\Omega^+ \cup \Omega^-$ est :

$$\mathbf{F}_k = \sum_i \mathbf{f}_i \eta_\epsilon^0(\mathbf{x}_{ik}) \frac{m_i}{\rho_i} \quad (6.8)$$

Nous avons choisi ici d'adopter l'approximation de la fonction $\delta_0()$ par le noyau $\eta_\epsilon^0()$ déjà utilisé (fonction gaussienne tronquée : équation (2.41)). Ce n'est pas la stratégie adoptée habituellement en différences finies, où la fonction $\delta_0()$ est approchée par d'autres noyaux, devant satisfaire, pourtant, les mêmes conditions (§2.4). Les fonctions proposées dans la littérature ont généralement un support de rayon plus petit [182]. D'ailleurs, Hieber et Koumoutsakos [93] ne précisent pas l'approximation qu'ils emploient (au delà du fait qu'il s'agisse de noyaux de type

splines). En intégrant, de la même manière, l'équation (6.7) autour du marqueur k , on obtient :

$$\underbrace{\sum_i \mathbf{f}_i \eta_\epsilon^0(\mathbf{x}_{ik}) \frac{m_i}{\rho_i}}_{\mathbf{F}_k} = \sum_i \left(-\mathbf{r} \mathbf{h} \mathbf{s}_i - \frac{\mathbf{u}_i^n}{\Delta t} \right) \eta_\epsilon^0(\mathbf{x}_{ik}) \frac{m_i}{\rho_i} + \frac{1}{\Delta t} \underbrace{\sum_i \mathbf{u}_i^{n+1} \eta_\epsilon^0(\mathbf{x}_{ik}) \frac{m_i}{\rho_i}}_{\mathbf{U}_k} \quad (6.9)$$

La dernière somme de l'équation (6.9) correspond à la prédiction de la vitesse au marqueur k en fin d'itération. En le remplaçant par la vitesse désirée \mathbf{U}_k (principe du forçage direct [69]), on obtient la force à imposer à la frontière, par combinaison des équations (6.9) et (6.8) :

$$\mathbf{F}_k = \sum_i \left(-\mathbf{r} \mathbf{h} \mathbf{s}_i - \frac{\mathbf{u}_i^n}{\Delta t} \right) \eta_\epsilon^0(\mathbf{x}_{ik}) \frac{m_i}{\rho_i} + \frac{\mathbf{U}_k}{\Delta t} \quad (6.10)$$

Enfin, les forces \mathbf{F}_k sont réparties sur les particules voisines par :

$$\mathbf{f}_i = \sum_k \mathbf{F}_k \eta_\epsilon^0(\mathbf{x}_{ki}) S_k \quad (6.11)$$

L'équation (6.11) ne peut être correcte (du point de vue dimensionnel) que si S_k est une surface (en 2D) ou un volume (en 3D), ce qui peut paraître étonnant quand on considère respectivement des marqueurs sensés représenter une courbe (en 2D) ou une surface (en 3D). C'est pourtant la façon dont est traitée cette répartition des forces habituellement en *frontière immergée* [182]. S_k permet à l'origine de définir la marge de flexibilité de la paroi (il ne s'agit là que d'une interprétation parmi tant d'autres proposées dans la littérature). Dans les résultats exposés ensuite, nous avons choisi de définir $S_k = l_k^2$ avec l_k , la longueur du segment associé au marqueur k (les applications étudiées ici sont en deux dimensions d'espace). De plus, la frontière a été discrétisée avec le même raffinement que le champ de particules, si bien que $l_k = h$. L'équation (6.11) devient donc :

$$\mathbf{f}_i = \sum_k \mathbf{F}_k \eta_\epsilon^0(\mathbf{x}_{ki}) h^2 \quad (6.12)$$

Aucune précision n'est apportée sur ce point par Hieber et Koumoutsakos [93]. Pour les résultats présentés ensuite, en suivant cette méthode, le champ de particules est remaillé à chaque itération, comme le proposent Hieber et Koumoutsakos [93]. Dans ce contexte, l'utilisation de l'équation (6.12) ne peut conduire qu'à des résultats équivalents à ceux obtenus en différences finies, puisque toutes les particules et les marqueurs se voient attribués une surface uniforme h^2 , avec h le pas de remaillage.

Remaillage

La technique de remaillage, couramment employée en méthode vortex [45], consiste à replacer les particules sur une grille cartésienne uniforme. Une grandeur Φ associée aux nouvelles particules



est obtenue à partir des anciennes particules par interpolation à l'aide du noyau M'_4 (§2.2.2) :

$$\Phi_i^{\text{nouveau}} = \frac{h^2}{\sum_j \frac{m_j}{\rho_j} M'_4(\mathbf{x}_{ij}/h)} \sum_j \Phi_j^{\text{ancien}} M'_4(\mathbf{x}_{ij}/h) \quad (6.13)$$

On peut mettre en doute l'intérêt de remailler à chaque itération quand on considère que l'atout majeur de la méthode SPH est justement le suivi lagrangien des particules. Cependant, dans les applications choisies par Hieber et Koumoutsakos [93], l'absence de surface libre ou d'interfaces entre d'éventuelles phases, justifie l'utilisation du remaillage comme une amélioration de la précision. L'utilisation de techniques de renormalisation est alors inutile car les particules sont constamment réparties idéalement pour réaliser des approximations des opérateurs différentiels avec une précision à l'ordre 2. La *frontière immergée* est d'ailleurs particulièrement adaptée à ce contexte car il n'est pas nécessaire de faire coïncider la paroi à modéliser, avec les particules. RSPH est souvent apparentée, à tort, à une méthode différences finies *masquée* ; les dérivées restent lagrangiennes : le terme convectif de l'équation de quantité de mouvement est une simple dérivée temporelle, ce qui conduit à un meilleur calcul des trajectoires des particules.

Dans le cas présent, l'objectif étant la modélisation d'écoulements à surface libre, le remaillage devient un point problématique. Il existe des techniques de coupure permettant de redéfinir la surface libre après ce remaillage, mais leur mise en œuvre reste hasardeuse et une fréquence de remaillage trop grande *gène* l'évolution de l'interface. Nous avons tenté de suivre la méthode présentée dans ce paragraphe en ne remaillant pas systématiquement à chaque itération, sans succès en raison des erreurs d'interpolation générées par l'utilisation de l'équation (6.12). Nous avons également essayé de la remplacer par un calcul plus précis des forces nécessaires, en suivant l'approche de Su *et al.* [216]. Si cette technique permet d'abaisser la fréquence des remaillages, elle nécessite l'inversion d'une matrice de taille $M \times M$ à chaque itération (cette matrice dépendait des positions des particules voisines), avec M le nombre de marqueurs sur la *frontière immergée*. De manière à se rapprocher au plus près de la méthode originale sans remailler et pour limiter les opérations sur la matrice à une inversion en début de simulation, nous avons mis en place une variante *multi-domaine* [197, 236], où une grille cartésienne solidaire de la frontière immergée permet de définir une répartition spatiale des forces autour de la frontière. Cette approche finalement assez coûteuse en temps de calcul, nécessitant 4 interpolations successives (particules \rightarrow maillage \rightarrow frontière \rightarrow maillage \rightarrow particules) mais une seule inversion de matrice en début de simulation, ne dispensait pas pour autant de remailler pour maintenir une précision acceptable lors des échanges particules-maillage local. Finalement, un procédé permettant de s'affranchir de tout remaillage a été adopté et est maintenant détaillé.

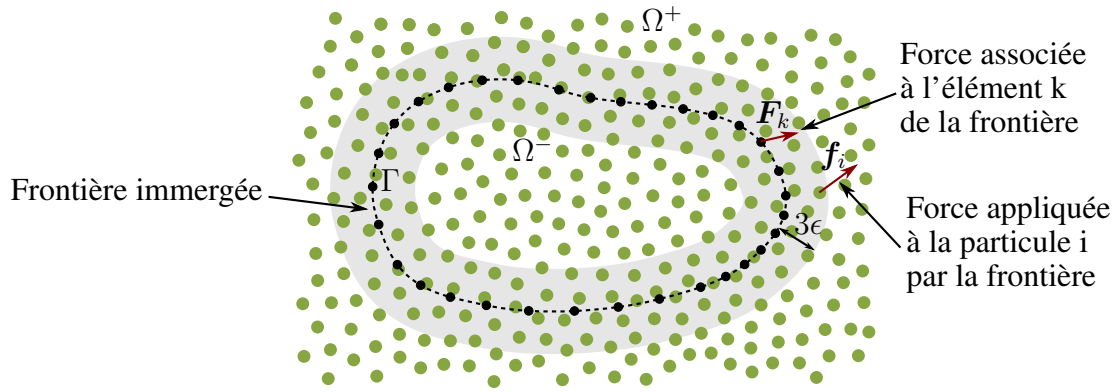


FIGURE 6.2 – Représentation d'un corps immergés dans le fluide par la méthode de la frontière immergée.

6.1.2 SPH sans remaillage et *frontière immergée corrigée par moindres carrés* : SPH-IBMLS

Frontière immergée corrigée par moindres carrés

La méthode présentée ici est proche de celle de Hieber et Koumoutsakos [93] et permet de s'affranchir totalement du remaillage (figure 6.2). Elle consiste également à transférer le second membre des équations de Navier-Stokes sur la frontière, pour y calculer les forces qui sont ensuite réparties sur le fluide. La différence vient de l'utilisation de moindres carrés pour l'approximation de la fonction $\delta_0()$, comme le proposent Vanella et Balaras [228]. Cette amélioration des interpolations a déjà été abordée précédemment (§4.4.3) : elle a été adoptée par Colagrossi et Landrini [41] pour la réinitialisation périodique de la masse volumique des particules ou par Dilts [59] pour la correction des approximations d'opérateurs différentiels.

Le noyau d'interpolation utilisé dans les équations (6.8)-(6.11) est modifié :

$$\eta_\epsilon^{0*}(\mathbf{x}_{ik}) = (\boldsymbol{\beta}_k \cdot (1, x_i - x_k, y_i - y_k)) \eta_\epsilon^0(\mathbf{x}_{ik}) \quad (6.14)$$

où le vecteur de correction $\boldsymbol{\beta}_k$ est calculé comme suit :

$$\boldsymbol{\beta}_k = \mathbf{A}_k^{-1} \cdot (1, 0, 0) \quad (6.15)$$

La matrice \mathbf{A}_k , de taille 3×3 (en deux dimensions d'espace) prend la forme :

$$\mathbf{A}_k = \sum_j \begin{bmatrix} 1 & (x_i - x_k) & (y_i - y_k) \\ (x_i - x_k) & (x_i - x_k)^2 & (x_i - x_k)(y_i - y_k) \\ (y_i - y_k) & (x_i - x_k)(y_i - y_k) & (y_i - y_k)^2 \end{bmatrix} \eta_\epsilon^0(\mathbf{x}_{ik}) \frac{m_j}{\rho_j} \quad (6.16)$$

La mise en œuvre de ces interpolations MLS nécessite l'inversion de M matrices 3×3 à chaque itération. Le temps nécessaire à ces inversions reste très faible par rapport aux autres opérations.



Un gain significatif en précision est donc obtenu lors de la première étape de la méthode, à savoir le transfert des informations des particules vers la frontière. L'étape finale, qui consiste à répartir les forces \mathbf{F}_k sur les particules voisines de la frontière (équation (6.11)) requiert également quelques modifications en raison de l'abandon du remaillage.

Répartition des forces sur les particules sans remaillage

Les particules voisines de la frontière doivent subir l'influence de la frontière immergée, comme dans la méthode précédente. Cependant, l'absence de remaillage et le modèle faiblement compressible employé ont pour conséquence que la surface de ces particules n'est plus constante. L'utilisation de la relation (6.12) conduit à des résultats incorrects. Elle est remplacée par l'équation suivante :

$$\mathbf{f}_i = \sum_k \mathbf{F}_k \eta_\epsilon^{0*}(\mathbf{x}_{ik}) c_k \quad (6.17)$$

où c_k , le facteur associé au marqueur k , est homogène à une surface en 2D et à un volume en 3D. Il est à définir de manière à avoir équivalence entre la force totale sur la frontière par les particules et la force exercée par la frontière sur les particules. En négligeant les effets de la faible compressibilité du fluide ($|\delta\rho|/\rho < 1\%$), on peut approcher cette équivalence de la manière suivante :

$$\sum_i \mathbf{f}_i \frac{m_i}{\rho_i} = \sum_k \mathbf{F}_k S_k \quad (6.18)$$

avec S_k , la *surface* associée à l'élément k de la frontière. Afin d'exprimer cette équivalence des forces totales sur la frontière, combinons les équations (6.17) et (6.18) :

$$\sum_i \sum_k \mathbf{F}_k \eta_\epsilon^{0*}(\mathbf{x}_{ik}) c_k \frac{m_i}{\rho_i} = \sum_k \mathbf{F}_k S_k \quad (6.19)$$

En réarrangeant les sommes du membre de gauche, on a :

$$\sum_k \mathbf{F}_k c_k \sum_i \eta_\epsilon^{0*}(\mathbf{x}_{ik}) \frac{m_i}{\rho_i} = \sum_k \mathbf{F}_k S_k \quad (6.20)$$

Ainsi, le facteur c_k doit être défini à :

$$c_k = \frac{S_k}{\sum_i \eta_\epsilon^{0*}(\mathbf{x}_{ik}) \frac{m_i}{\rho_i}} \quad (6.21)$$

Or, le principe de la correction par moindres carrés était de satisfaire la condition :

$$\sum_i \eta_\epsilon^{0*}(\mathbf{x}_{ik}) \frac{m_i}{\rho_i} = 1 \quad (6.22)$$

On a donc :

$$c_k = S_k \quad (6.23)$$

A la différence de la méthode RSPH-IB, pour laquelle $S_k = h^2$ (équation (6.12)), nous choisissons ici de tenir compte des variations de taille des particules voisines assimilées à des cellules de forme carrée, de côté $\sqrt{\frac{m_i}{\rho_i}}$ [228]. La surface attachée au marqueur k de la frontière est donnée par :

$$S_k = l_k d_k \quad (6.24)$$

avec d_k la moyenne pondérée des tailles associées à chaque particule dans le voisinage du marqueurs k , on a :

$$d_k = \sum_i \sqrt{\frac{m_i}{\rho_i}} \eta_\epsilon^{0*}(\mathbf{x}_{ik}) \frac{m_i}{\rho_i} \quad (6.25)$$

Ainsi,

$$c_k = l_k \sum_i \sqrt{\frac{m_i}{\rho_i}} \eta_\epsilon^{0*}(\mathbf{x}_{ik}) \frac{m_i}{\rho_i} \quad (6.26)$$

Les simulations réalisées à l'aide de cette variante de la frontière immergée n'ont nécessité aucun remaillage pour maintenir une précision acceptable, tant dans le fluide que dans le voisinage de la frontière. Le paragraphe suivant montre justement les résultats de validation obtenus sur le cas d'un carré forcé dans une boîte rectangulaire.

Enfin, la mise en œuvre algorithmique de ce modèle est assez aisée. Elle ne nécessite que la détection d'interactions supplémentaires entre les marqueurs \mathbf{x}_k et les particules \mathbf{x}_i , qui peut être réalisée par une méthode analogue à celle exposée au §8.2.2. Le temps de calcul consacré à cette partie du code de calcul est infime mais dépend directement du nombre de marqueurs sur la frontière. Dans les applications suivantes, le corps immergé étant de petite taille, peu de points ont été nécessaires (et toujours choisi de même espacement que les particules). L'absence de remaillage est d'ailleurs un gain précieux en temps de calcul, surtout quand celui-ci est réalisé à chaque itération. De plus, dans le cadre de calcul en parallèle, le remaillage nécessite souvent une réinitialisation complète des particules entre les processeurs (*cf.* §9).

6.2 Écoulement autour d'un carré forcé dans une boîte rectangulaire

Le cas étudié à présent est celui de l'écoulement généré par le forçage d'un cylindre de section carré, dans un liquide visqueux newtonien et incompressible, en l'absence de gravité. Le domaine



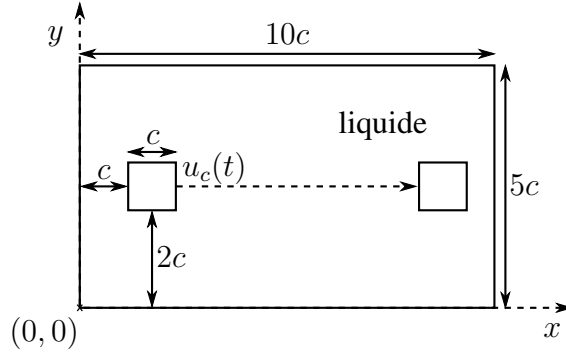


FIGURE 6.3 – Configuration initiale de l'écoulement autour d'un carré forcé, de côté c , dans une boîte rectangulaire de dimensions $10c \times 5c$.

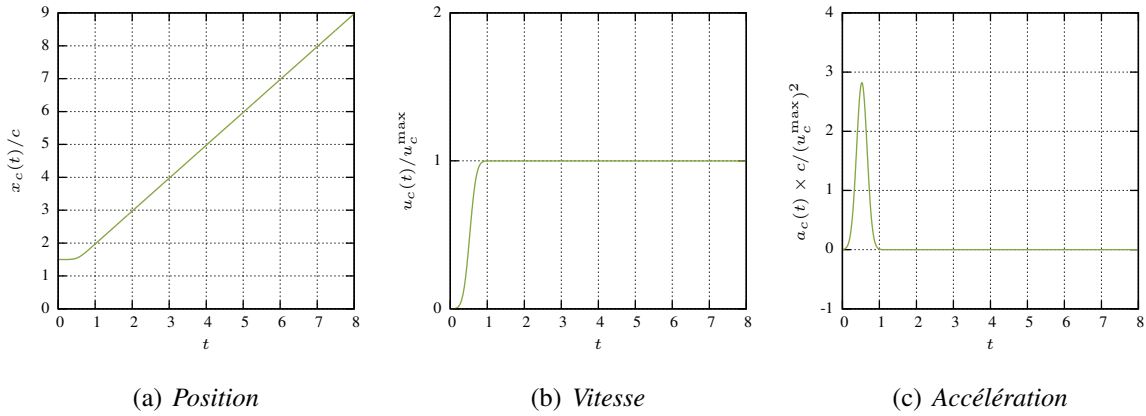


FIGURE 6.4 – Déplacement du centre du carré au cours du temps.

de calcul est limité par une boîte de forme rectangulaire (figure 6.3). Le carré initialement immobile, subit une forte accélération et atteint une vitesse stationnaire, u_c^{\max} , jusqu'à la fin de la simulation (figure 6.4(c)). Cet écoulement a été proposé par le Groupement d'Intérêt Scientifique (GIS) SPHERIC [255] pour comparaison des résultats fournis par les différents codes SPH développés au sein de la communauté. Les différentes tentatives sont décrites dans le paragraphe suivant. Les résultats sont généralement comparés à une solution de référence, issue de la simulation numérique par un code de résolution des équations de Navier-Stokes en différences finies, couplé à une méthode *Level-Set* pour le suivi des interfaces. Cette solution a été proposée par Colicchio *et al.* [42], qui a étudié l'écoulement pour trois nombres de Reynolds différents (50, 100 et 150), calculé par :

$$\text{Re} = \frac{cu_c^{\max}}{\nu} \quad (6.27)$$

Le maillage cartésien employé pour ces simulations est de taille 600×300 . Les figures 6.5, 6.6 et 6.7 montrent respectivement les champs de pression, module de vitesse et vorticité obtenus par le

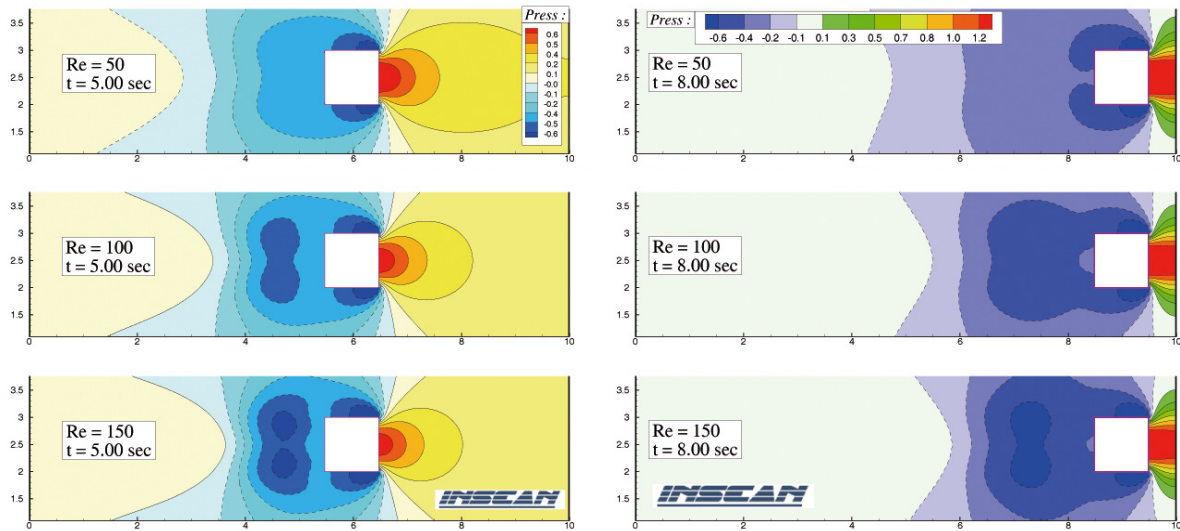


FIGURE 6.5 – Solution de référence [42] pour l'écoulement visqueux autour d'un carré forcé. Pression à $t = 5$ et $t = 8$ pour 3 valeurs du nombre de Reynolds : 50, 100 et 150.

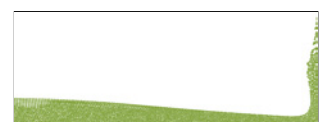
modèle de référence. A $t = 5$ (images de gauche), le cylindre de section carrée translate à vitesse constante. Les dépressions du sillage sont accentuées (figure 6.5) et la forme des structures de vorticit  se complexifie avec l'augmentation du nombre de Reynolds (figure 6.7). A $t = 8$, quelque soit le nombre de Reynolds, les sillages sont plus  tir s qu'  $t = 5$. Le carr  approchant rapidement l'extr mit  du domaine, les pressions mesur es sur la *face avant* sont nettement sup rieures.

6.2.1  tat de l'art sur la mod lisation de cet  coulement

Ce cas a permis, entre autres :

-   Colagrossi *et al.* [39] de mettre en  vidence les difficult s d'une des approches conventionnelles   correctement mod liser des obstacles   g om trie complexe. C'est le cas de la technique des *particules fant mes* d velopp e dans le chapitre pr c dent.
-   Lee *et al.* [126] de comparer les r sultats obtenus par les approches faiblement compressible et incompressible mentionn es pr c demment.
-   Nestor *et al.* [170] de valider une m thode de prise en compte de fronti res mobiles en volumes finis particuliers.
-   Ferrand *et al.* [74] de montrer la possibilit  de traiter des fronti res mobiles avec son nouveau traitement des conditions limites solides au sein du code de calcul SPARTACUS-2D [253].

La figure 6.8 est extraite des travaux de Colagrossi *et al.* [39], qui a propos  ce cas-test au sein de la communaut  SPHERIC [255]. Le premier graphique (figure 6.8(a)) montre une comparaison du champ de pression obtenu en fonction du type d'interpolation mise en  uvre dans le mod le SPH (interpolation classique ou corrig e par moindres carr s [59]). Des cavit s sont form es dans



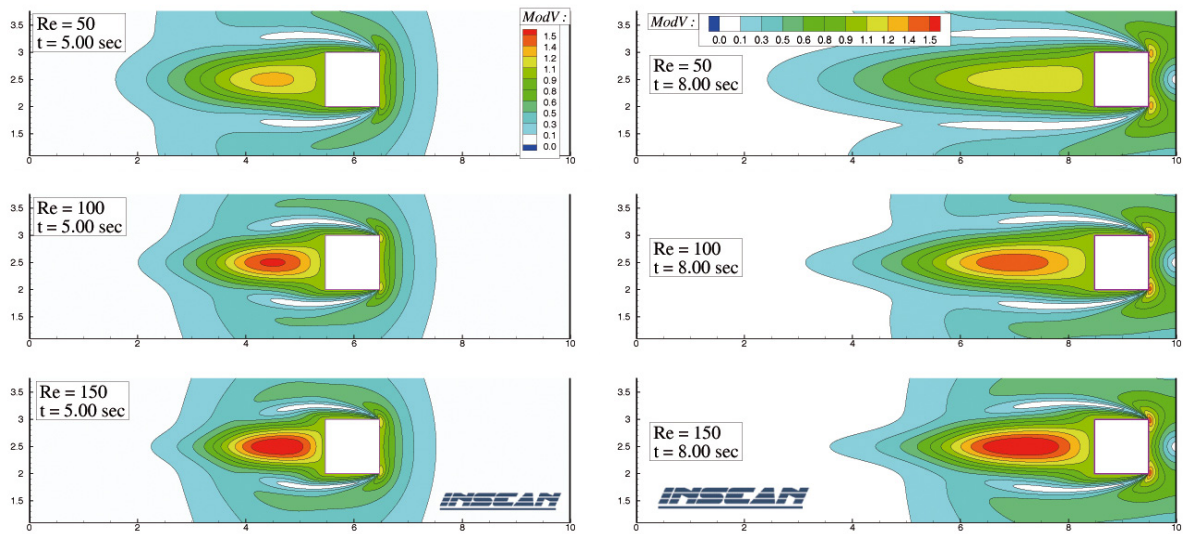


FIGURE 6.6 – Solution de référence [42] pour l'écoulement visqueux autour d'un carré forcé. Module de vitesse à $t = 5$ et $t = 8$ pour 3 valeurs du nombre de Reynolds : 50, 100 et 150.

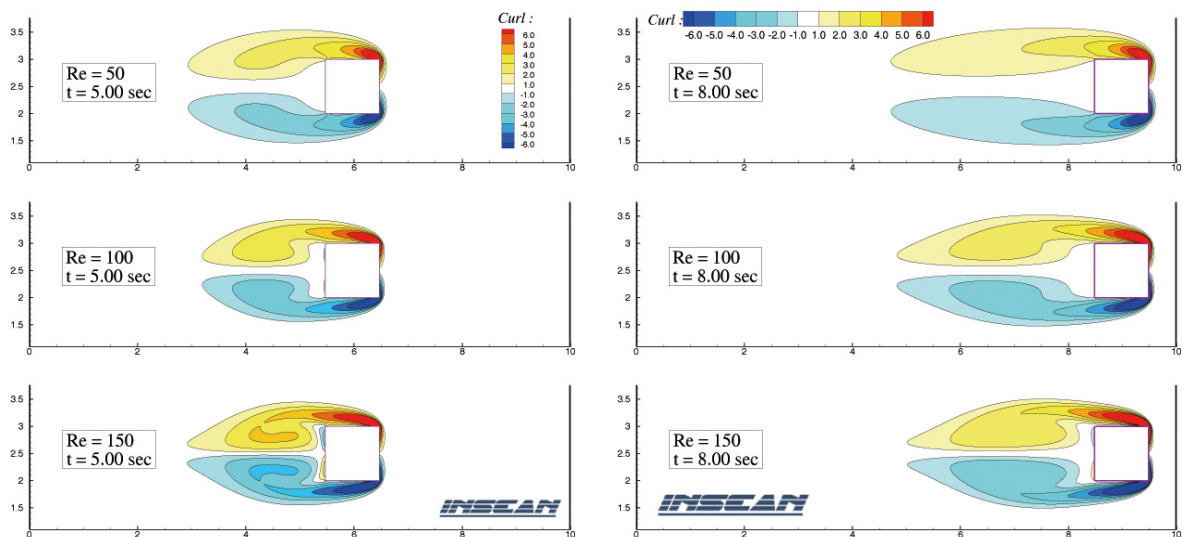
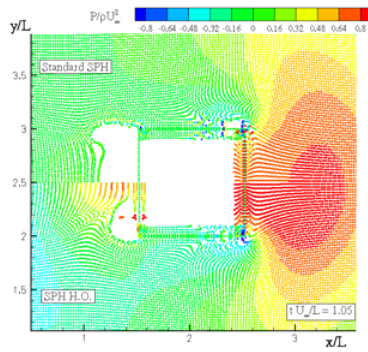
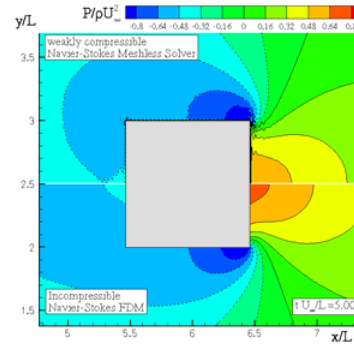


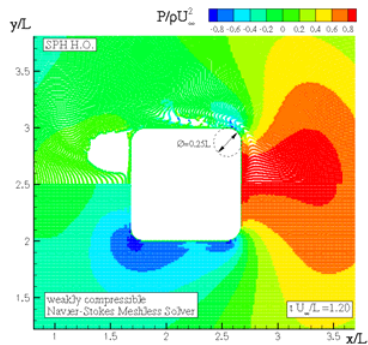
FIGURE 6.7 – Solution de référence [42] pour l'écoulement visqueux autour d'un carré forcé. Vorticité à $t = 5$ et $t = 8$ pour 3 valeurs du nombre de Reynolds : 50, 100 et 150.



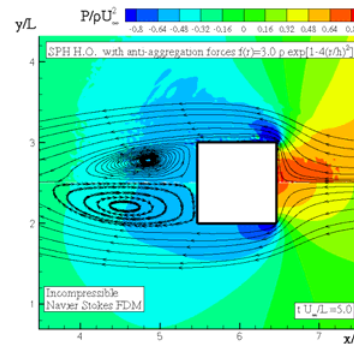
(a) Champ de pression au voisinage du carré forcé, à $t = 1,05L/u_c^{\max}$ et $Re = 50$: moitié supérieure, résultat obtenu à l'aide d'un schéma SPH classique ; moitié inférieure, résultat obtenu à l'aide de noyaux corrigés par moindres carrés (MLS).



(b) Champ de pression au voisinage du carré forcé, à $t = 5,00L/u_c^{\max}$ et $Re = 50$: moitié supérieure, résultat obtenu à l'aide d'un schéma SPH eulérien ; moitié inférieure, solution de référence [42].



(c) Champ de pression au voisinage du carré forcé aux coins arrondis, à $t = 1,20L/u_c^{\max}$ et $Re = 50$: moitié supérieure, résultat obtenu à l'aide d'un schéma SPH corrigé par MLS ; moitié inférieure, schéma SPH eulérien.



(d) Champ de pression au voisinage du carré forcé, à $t = 5,00L/u_c^{\max}$ et $Re = 50$: moitié supérieure, résultat obtenu à l'aide d'un schéma SPH corrigé par MLS, intégrant des forces répulsives et une solution potentielle locale pour la suppression des cavités ; moitié inférieure, schéma SPH eulérien.

FIGURE 6.8 – Etude de l'écoulement autour d'un carré forcé avec SPH par Colagrossi et al. [39].



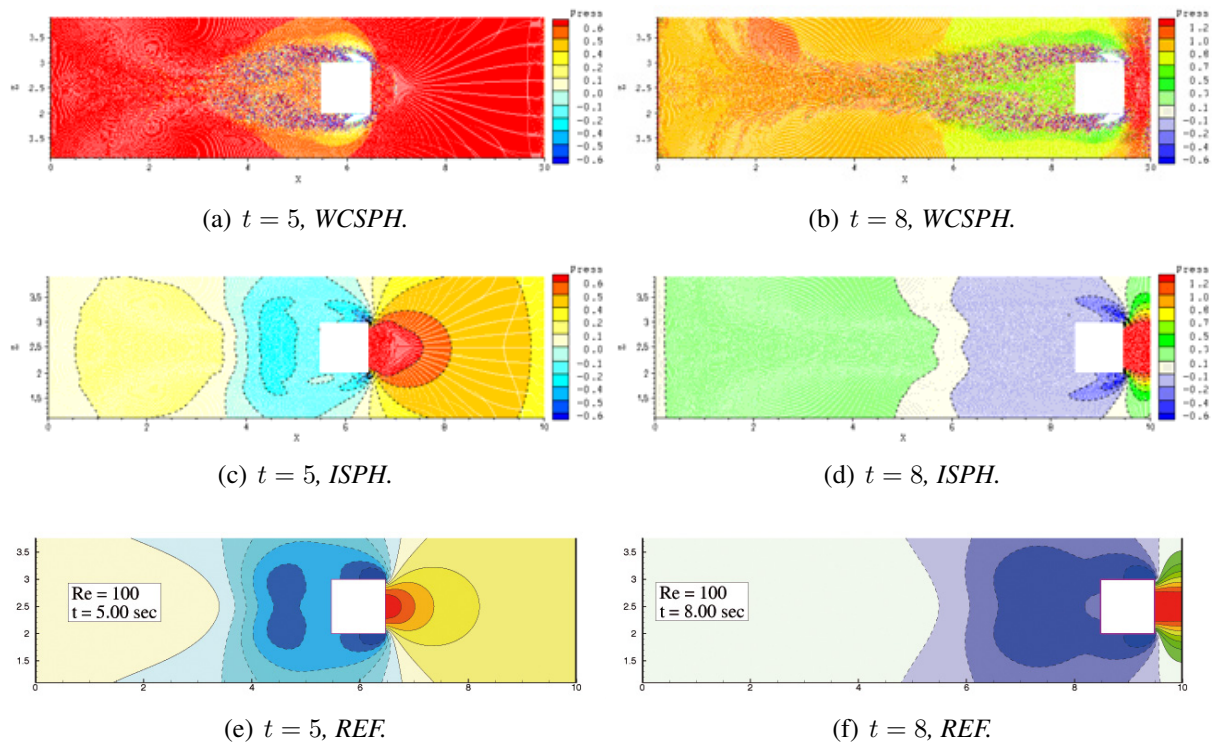


FIGURE 6.9 – Pression autour du carré forcé simulé par les modèles SPH faiblement compressible (WCSPH) et incompressible (ISPH) de Lee et al. [126] à $Re = 100$, la solution de référence (REF) est présentée pour comparaison.

le sillage du carré et près des coins, aux endroits où la pression est sensée être négative. Ces anomalies sont dues à la combinaison de deux facteurs sensibles en méthode SPH : la modélisation d'angles droits par la technique des *particules fantômes* et l'instabilité en tension (§4.4.4). Les deux graphiques suivants (Fig. 6.8(b) et 6.8(c)) montrent que ces cavités peuvent être supprimées si les équations de Navier-Stokes sont résolues à l'aide d'un schéma SPH, qu'ils disent *eulérien*, pour lequel les particules sont fixes et le terme convectif est discrétisé. Sur le graphique 6.8(b), les deux solutions affichent quelques différences, principalement dues au fait que le modèle SPH employé est faiblement compressible. Le graphique 6.8(c) montre que même en arrondissant les coins, un schéma lagrangien ne parvient pas à correctement reproduire les trajectoires. Les meilleurs résultats (figure 6.8(d)) ont été obtenus en combinant un modèle SPH basé sur une approximation MLS des gradients, des forces répulsives évitant la formation de cavités et l'introduction d'une solution potentielle à proximité des angles droits. Selon cette analyse, les difficultés liées à la simulation de cet écoulement viendraient essentiellement de la nature lagrangienne de la méthode.

Ce n'est pas l'avis de Lee *et al.* [126] qui mettent en cause le modèle faiblement compressible (WCSPH) et tente de démontrer que seul un schéma SPH incompressible (ISPH) peut fournir des résultats satisfaisants sur ce cas. En effet, les résultats exposés par Lee *et al.* [126] montrent que le modèle WCSPH employé fournit des champs de pression très bruités, à la validité contestable

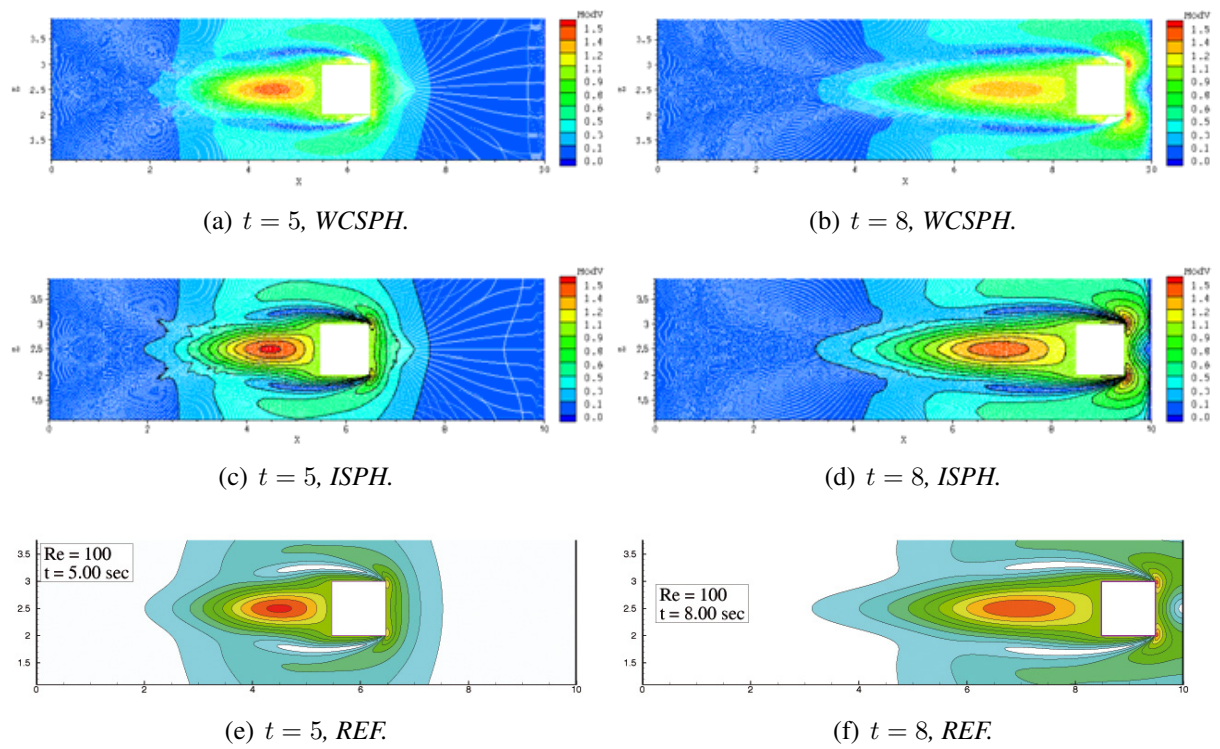


FIGURE 6.10 – Module de vitesse autour du carré forcé simulé par les modèles SPH faiblement compressible (WSPH) et incompressible (ISPH) de Lee et al. [126] à $Re = 100$, la solution de référence (REF) est présentée pour comparaison.

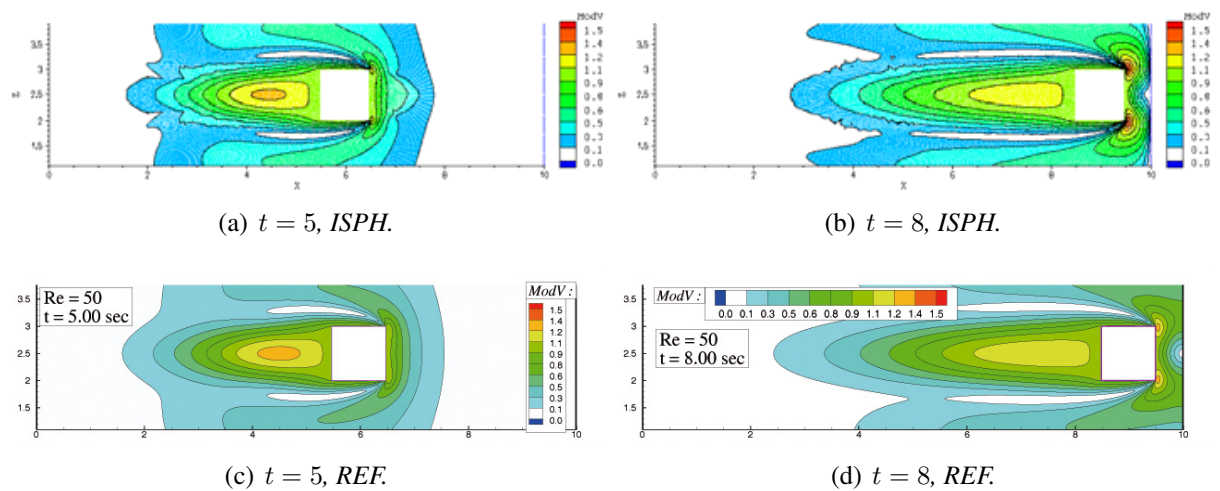
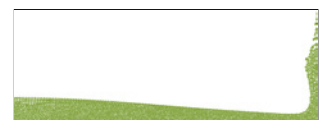


FIGURE 6.11 – Module de vitesse autour du carré forcé simulé par le modèle SPH incompressible (ISPH) de Lee et al. [126] à $Re = 50$, la solution de référence (REF) est présentée pour comparaison.



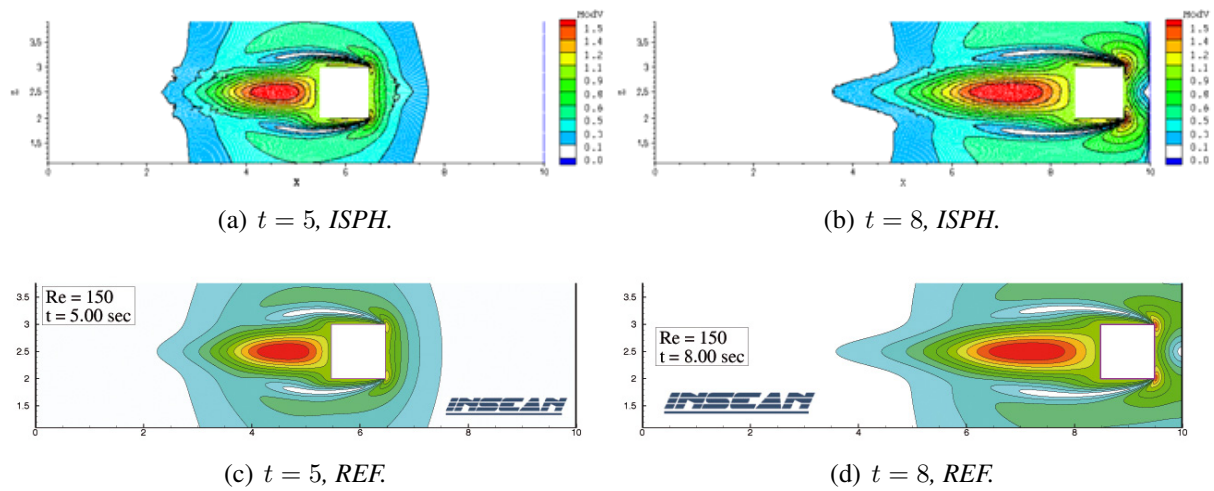


FIGURE 6.12 – Module de vitesse autour du carré forcé simulé par le modèle SPH incompressible (ISPH) de Lee et al. [126] à $Re = 150$, la solution de référence (REF) est présentée pour comparaison.

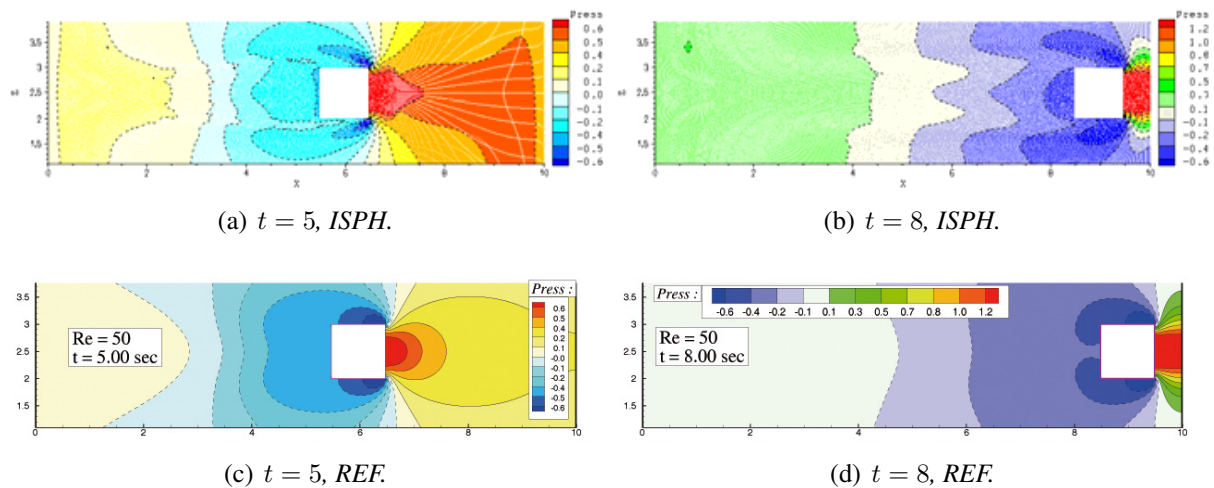


FIGURE 6.13 – Pression autour du carré forcé simulé par le modèle SPH incompressible (ISPH) de Lee et al. [126] à $Re = 050$, la solution de référence (REF) est présentée pour comparaison.

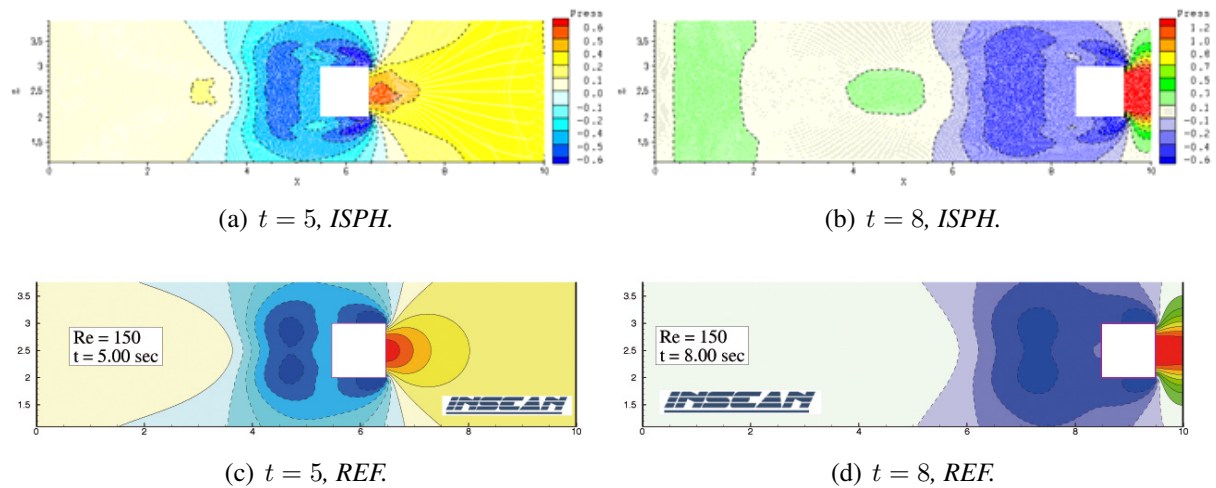
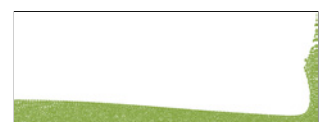


FIGURE 6.14 – Pression autour du carré forcé simulé par le modèle SPH incompressible (ISPH) de Lee *et al.* [126] à $Re = 150$, la solution de référence (REF) est présentée pour comparaison.

(figure 6.9(a)). Les contours de champs de pression obtenus à $Re = 100$ par le modèle incompressible sont plus proches de la solution de référence. Le code de calcul ISPH a aussi l'avantage de fournir ces résultats avec un pas de temps 5 fois plus grands, les temps de simulations avec chacun des modèles ne sont pas précisés. On peut noter que finalement, même si le champ de pression semble plus exploitable par l'emploi d'un modèle incompressible, les champs de vitesse obtenus sont très similaires, si on omet l'absence de contours (soulignés en noirs) sur le résultat WCSPH par rapport à ISPH. Le résultat du modèle WCSPH montre également deux cavités qui prennent naissance sur les coins avant du carré. Ces cavités sont les mêmes que celles observées par Colagrossi *et al.* [39] (figure 6.8(a)) et sont absentes avec le modèle ISPH. Lee *et al.* [126] valident ensuite le modèle ISPH sur les cas $Re = 50$ et $Re = 150$, en comparant les modules de vitesse (figures 6.11 et 6.12) et les champs de pression (figures 6.12 et 6.14) avec les résultats de référence. Les solutions WCSPH ne sont pas montrées pour ces cas. Par ailleurs, les courbes de coefficient de traînée présentées par l'auteur montrent des oscillations de grande amplitude, sans justification aucune, dans le cadre d'un modèle incompressible. Chacune des approches profitant régulièrement de nouvelles avancées, il pourrait être intéressant que les meilleurs résultats obtenus avec chacune d'elles soient de nouveau confrontés.

Un résultat intéressant a également été proposé par Nestor *et al.* [170] avec la méthode Volumes Finis Particulaires (FVPM). Ses travaux portent sur l'extension de cette méthode particulière, proche de la méthode SPH, aux écoulements visqueux en présence de corps en mouvement. Par la mise en œuvre de conditions aux limites appropriées et en corrigeant les trajectoires des particules (le modèle devient semi-lagrangien), l'auteur propose une solution très prometteuse. Les figures 6.15 et 6.17 montrent les modules de vitesse obtenus avec, respectivement, $Re = 50$ et $Re = 150$. Les résultats sont encore une fois très proches de la solution de référence avec, pourtant, seulement 20000 particules (contre 180000 mailles pour la solution de référence). Les champs de



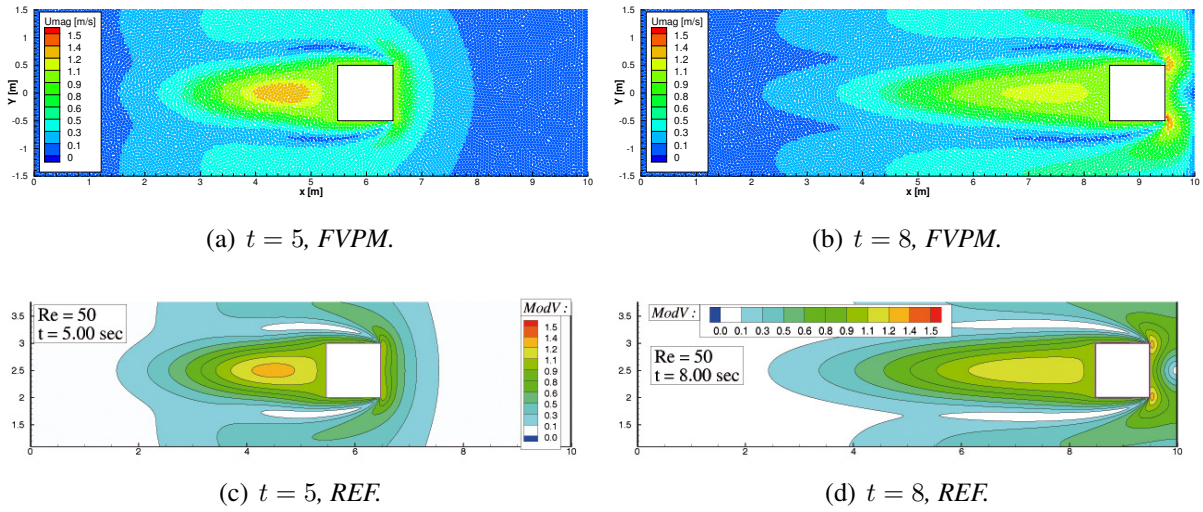


FIGURE 6.15 – Module de vitesse autour du carré forcé simulé par le modèle volumes finis particulaires (FVPM) de Nestor et al. [170] à $Re = 50$, la solution de référence (REF) est présentée pour comparaison.

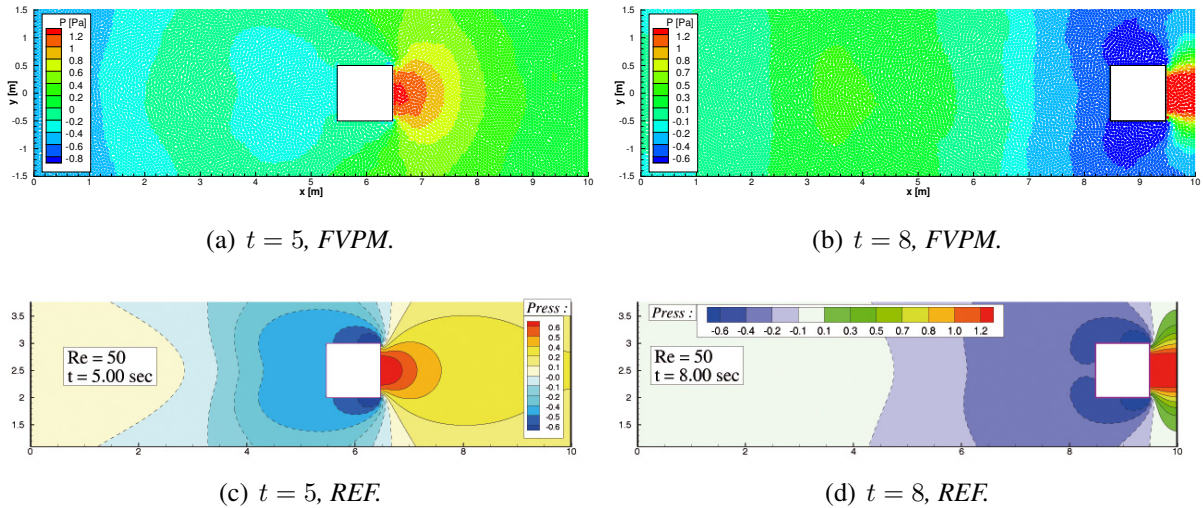
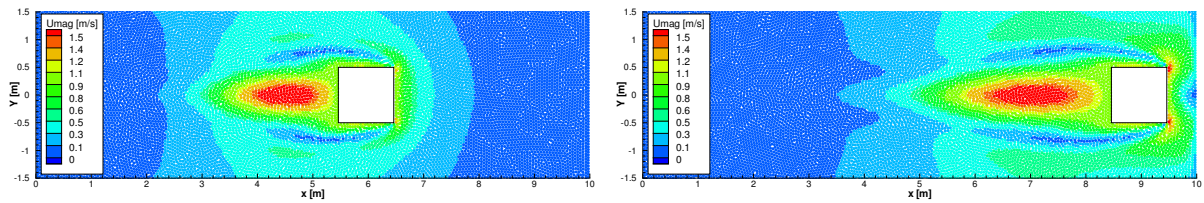
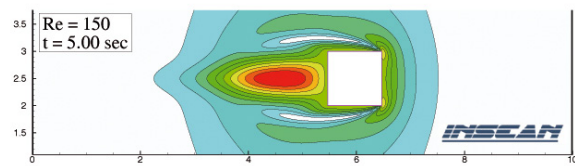


FIGURE 6.16 – Pression autour du carré forcé simulé par le modèle volumes finis particulaires (FVPM) de Nestor et al. [170] à $Re = 50$, la solution de référence (REF) est présentée pour comparaison.

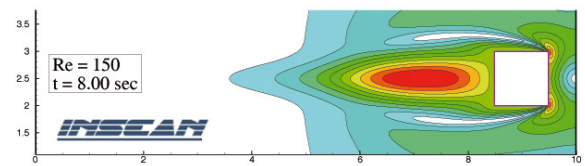


(a) $t = 5$, FVPM.

(b) $t = 8$, FVPM.

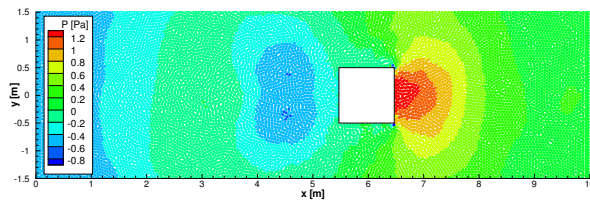


(c) $t = 5$, REF.



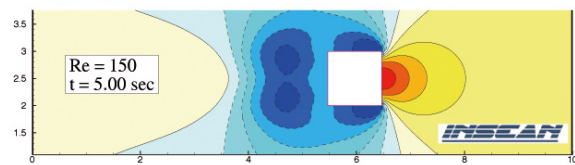
(d) $t = 8$, REF.

FIGURE 6.17 – Module de vitesse autour du carré forcé simulé par le modèle volumes finis particulaires (FVPM) de Nestor et al. [170] à $Re = 150$, la solution de référence (REF) est présentée pour comparaison.

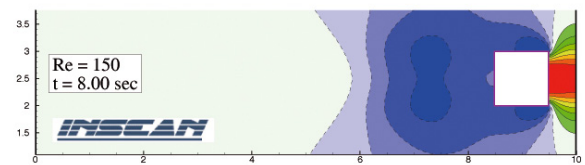


(a) $t = 5$, FVPM.

(b) $t = 8$, FVPM.

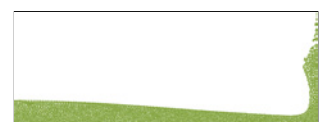


(c) $t = 5$, REF.



(d) $t = 8$, REF.

FIGURE 6.18 – Pression autour du carré forcé simulé par le modèle volumes finis particulaires (FVPM) de Nestor et al. [170] à $Re = 150$, la solution de référence (REF) est présentée pour comparaison.



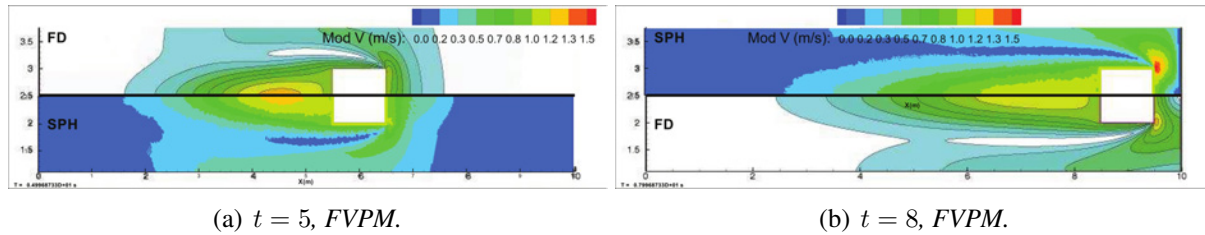


FIGURE 6.19 – Module de vitesse autour du carré forcé simulé par SPARTACUS-2D et le modèle de conditions aux limites de Ferrand *et al.* [74] à $Re = 50$ (SPH) comparé à la solution de référence (FD).

pression présentés (figures 6.16 et 6.18) diffèrent légèrement de la solution de référence, mais les zones en dépression et surpression sont correctement situées, compte tenu de la nature faiblement compressible de ce modèle. La méthode de correction du mouvement des particules à d'ailleurs été reprise par Xu *et al.* [238], qui améliorent ainsi la précision de leur modèle ISPH. Si le modèle, corrigé de cette manière, n'est plus totalement lagrangien, cette approche reste une alternative intéressante au remaillage.

Enfin, une récente application d'un modèle SPH à ce cas-test a été proposée par Ferrand *et al.* [74]. Il propose une nouvelle méthode de modélisation des conditions limites, abordée au cours du chapitre précédent (§5.3.1), dans le cas d'écoulements non-visqueux. L'auteur propose plusieurs perspectives pour cette approche, comme son extension aux écoulements visqueux et présente un résultat préliminaire du cas $Re = 50$, sans qu'aucune précision ne soit apportée sur la conditions appliquée en vitesse. La figure 6.19 montre le champ de vitesse obtenu, effectivement très proche de la solution de référence.

Les résultats proposés pour cet écoulement dans la bibliographie tendent donc à démontrer qu'un modèle faiblement compressible ne peut pas reproduire correctement les trajectoires dans le voisinage du carré sans l'apport de corrections du déplacement des particules (pour éviter la formation de cavités) et/ou la mise en œuvre de conditions limites plus fiables que les approches conventionnelles. Lee *et al.* [126] est plus radical vis à vis des modèles WCSPH, en montrant des résultats plus mauvais que ceux obtenus par ISPH. Nous avons justement choisi, au cours de ce manuscrit, l'approche WCSPH pour la condition implicite de surface libre qu'elle propose (contrairement à la méthode ISPH) ; nous avons également mis au point un modèle de frontière immergée permettant de s'affranchir de remaillage (ou de tout autre méthode de correction artificielle des trajectoires des particules permettant de maintenir la précision des simulations : XSPH, méthode de Nestor *et al.* [170]) dans le but de conserver le caractère lagrangien de la méthode. Nous allons donc maintenant comparer les résultats obtenus avec les deux méthodes présentées au §6.1, et montrer qu'une approche faiblement compressible combinée à une modélisation précise des conditions limites peut apporter une solution très satisfaisante à ce type d'écoulement.

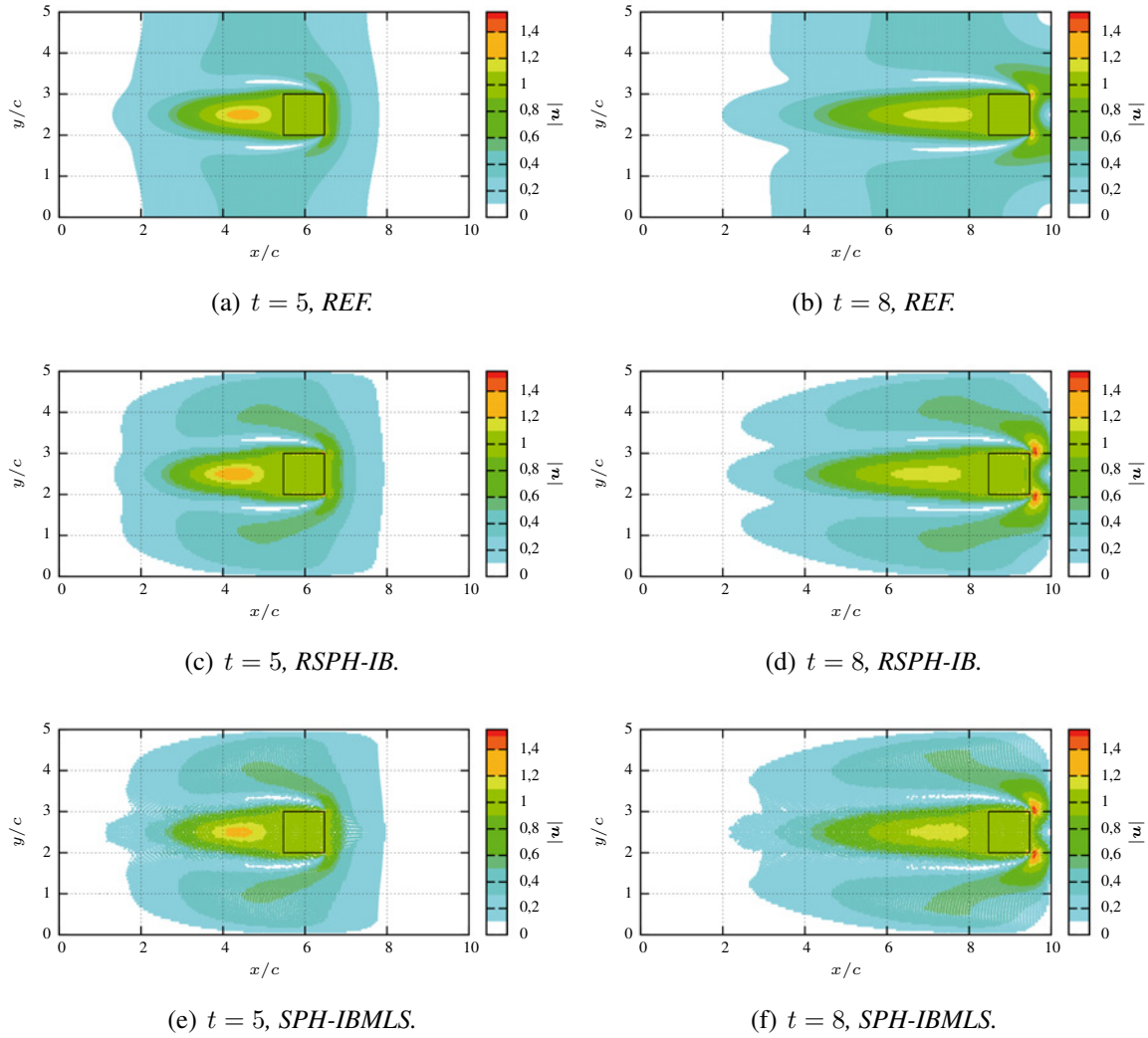
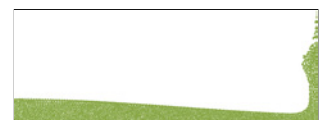


FIGURE 6.20 – Module de vitesse autour du carré forcé à $Re = 50$ simulé à l’aide du modèle de Hieber et Koumoutsakos [93] (RSPH-IB) et par le nouveau modèle (SPH-IBMLS), la solution de référence (REF) est présentée pour comparaison.

6.2.2 Comparaison des modèles RSPH-IB et SPH-IBMLS à $Re = 50$

Les deux méthodes de *frontière immergée* présentées au §6.1 ont été confrontées sur ce cas-test à $Re=50$. Les simulations ont été réalisées avec autant de particules dans les deux cas, à savoir 200×100 particules dans le domaine fluide. La frontière immergée a été discrétisée de façon similaire, avec un total de 80 marqueurs répartis uniformément sur le périmètre du carré. Les bords du domaine ont, quant à eux, été modélisés par la méthode des particules fantômes, abordée au chapitre précédent. Rappelons les différences entre les deux modèles employés :



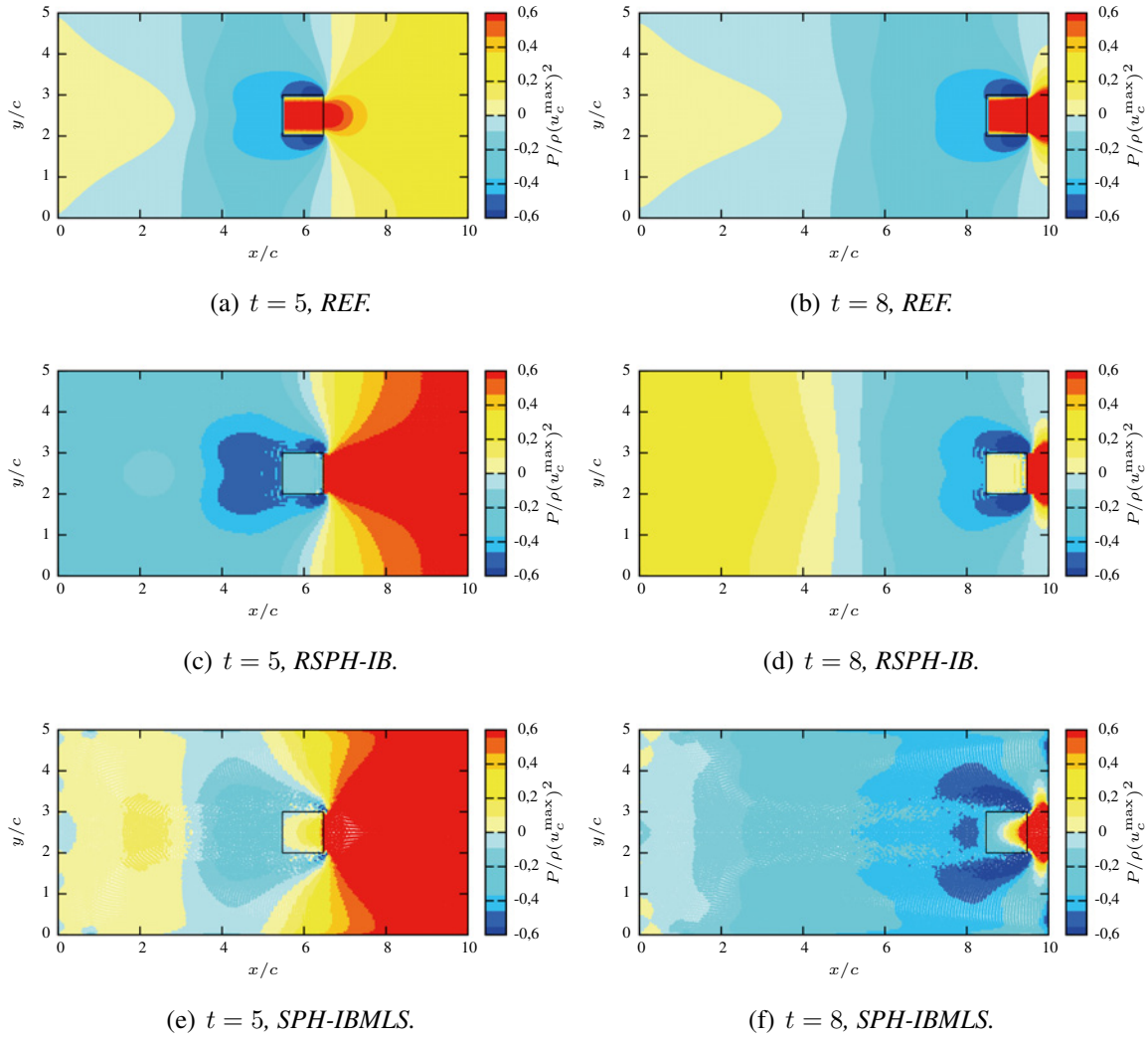


FIGURE 6.21 – Pression autour du carré forcé à $Re = 50$ simulé à l'aide du modèle de Hieber et Koumoutsakos [93] (RSPH-IB) et par le nouveau modèle (SPH-IBMLS), la solution de référence (REF) est présentée pour comparaison.

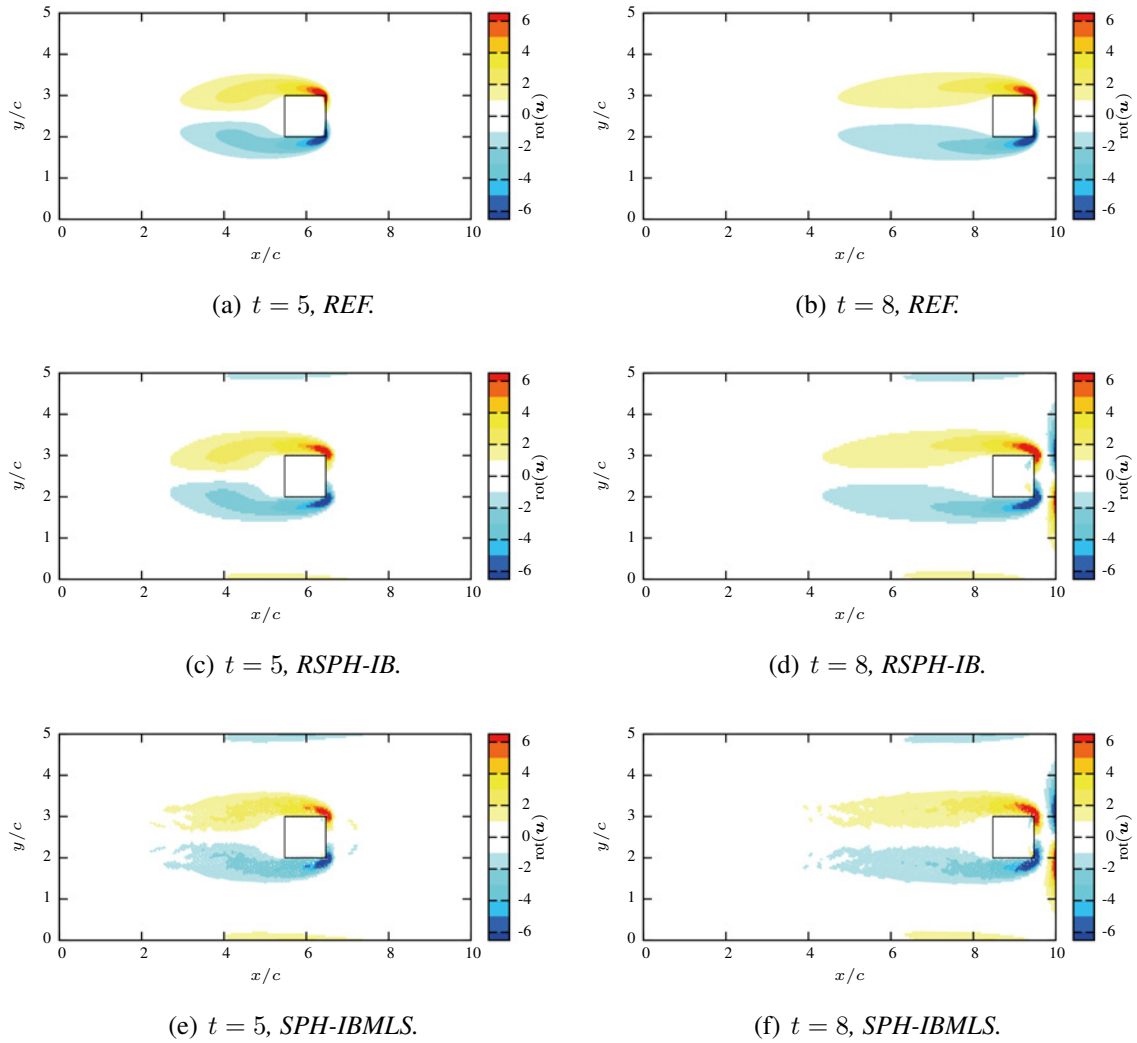
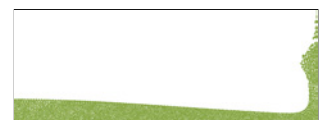


FIGURE 6.22 – Vorticité autour du carré forcé à $Re = 50$ simulé à l'aide du modèle de Hieber et Koumoutsakos [93] (RSPH-IB) et par le nouveau modèle (SPH-IBMLS), la solution de référence (REF) est présentée pour comparaison.



RSPH-IB

- Équations (6.1) à (6.6) simulées par SPH faiblement compressible et remaillée à tous les pas de temps.
- Frontière immergée classique.
- Pas de temps : $\Delta t = 0,0002$.

SPH-IBMLS

- Équations (6.1) à (6.6) simulées par SPH faiblement compressible, sans remaillage.
- Frontière immergée corrigée par MLS.
- Pas de temps : $\Delta t = 0,001$ ($5\times$ plus grand qu'avec le modèle RSPH-IB).

La masse volumique du fluide simulé est $\rho_0 = 1$. Dans les deux cas, la pression de référence P_0 de la loi d'état (équation (6.4)) a été fixée à la pression minimale calculée dans le fluide. Par ce procédé, aucune cavité comme celles observées par Lee *et al.* [126] (figure 6.9) ou Colagrossi *et al.* [39] (figure 6.8(a)) n'apparaît. Une variante, qui consiste à prendre la pression moyenne dans le voisinage de la particule, a été proposée par Basa *et al.* [10]. La vitesse du son a été fixée à $c_0 = 30m.s^{-1}$, dans les deux cas. Le pas de temps a pu être défini à une valeur 5 fois supérieure pour la méthode sans remaillage, par rapport à la méthode remaillée à chaque pas de temps, pour un temps de calcul plus de 10 fois inférieur (essentiellement dû au plus grand pas de temps et à l'absence de remaillage).

La figure 6.20 montre le module de vitesse calculé par la méthode RSPH-IB de Hieber et Koumoutsakos [93], la méthode SPH-IBMLS proposée ici et la solution de référence de Colicchio *et al.* [42] (calculée sur un maillage 600×300). L'intégralité du domaine de calcul est affichée ($\Omega = [0; 10] \times [0; 5]$), contrairement aux résultats de la littérature étudiés dans le paragraphe précédent pour lesquels les bords supérieurs ($y/c \in [4; 5]$) et inférieurs ($y/c \in [0; 1]$) sont masqués (par souci de cohérence avec les représentations de la solution de référence (figures 6.5 à 6.7)). Les données brutes issues des simulations du code de calcul Navier-Stokes en différences finies Level-Set de Colicchio *et al.* [42] ont été gracieusement mises à notre disposition par l'auteur, ce qui nous a permis de comparer les champs obtenus avec la même échelle de couleurs. La description proposée par SPHERIC [255] pour la simulation de cet écoulement stipulent que les conditions sur les bords du domaine doivent être de type adhérence. Or, à la vue des champs de référence, il semblerait que la condition appliquée ait été, en réalité, de type glissement, la vitesse n'étant pas nulle à la paroi (figures 6.20(a) et 6.20(b)). Au delà, des zones influencées par les conditions aux bords du domaine, on observe peu de différences entre les trois solutions. Les sillages secondaires (sur les côtés du carré) ont une forme différente si on compare les modèles SPH et la solution de référence. Les deux solutions SPH sont proches, on remarque néanmoins une déformation du dernier niveau de vitesse ($|\mathbf{u}| = 0,1$) dans le cas du modèle SPH-IBMLS par rapport aux deux autres solutions. On retrouve le même type de distorsion du champ quand on observe la vorticit  (figure 6.22), alors

que les solutions de référence et la solution RSPH-IB sont très proches. Cela vient du déplacement purement lagrangien des particules et aux erreurs induites lors de l'approximation du terme visqueux, qui ne diffuse pas aussi bien le *sillage*. L'approximation du rotationnel de la vitesse, même si ce n'est qu'une étape de représentation graphique, est aussi influencée par ce désordre. La renormalisation, évoquée au §4.6.1, pourrait permettre d'en réduire les effets. Comme le remaillage, la renormalisation entraîne quelques difficultés pour la simulation de la surface libre, même si elle apporte, pourtant, une correction des erreurs dans le domaine fluide. Nous avons donc préféré nous dispenser de cette correction. Les champs de pression (figure 6.21) montrent des écarts plus importants entre la solution de référence et les méthodes SPH. Cela vient surtout d'oscillations qui apparaissent dans le domaine fluide dès le début de la simulation, quand le carré est mis en mouvement : un onde de pression est générée par l'accélération du carré et est réfléchié alternativement par le bord droit ($x/c = 10$) et gauche ($x/c = 0$) du domaine fluide. Malgré ces perturbations, les zones de faible pression, sur les côtés du carré et dans son sillage, sont correctement situées et sont du bon ordre de grandeur. Le champ de pression est, d'ailleurs, bien moins bruité que ceux présentés par Lee *et al.* [126]. Au delà des perturbations constatées sur les champ de rotationnel et de vitesse, la méthode SPH sans remaillage parvient à reproduire correctement l'écoulement autour de la frontière immergée.

6.2.3 Validation du modèle SPH-IBMLS à $Re = 50$, $Re = 100$ et $Re = 150$

Les trois cas ($Re = 50$, $Re = 100$ et $Re = 150$) ont été simulés pour trois résolutions différentes : 200×100 comme précédemment, 400×200 et 600×300 . Le pas de temps des simulations a été réduit proportionnellement à la taille des particules.

On constate que l'augmentation du nombre de particules semble réduire les perturbations des contours, générées par le déplacement lagrangien des particules (figure 6.23). Cette observation est d'autant plus vraie quand on augmente le nombre de Reynolds (figures 6.25 et 6.27). La forme des sillages est très proche de la solution de référence, que ce soit en termes de niveaux de vorticit  (contours de couleurs) ou si on observe les dimensions des structures. Les zones de vorticit  pr sentes sur les bords du domaine sont g n r es par la condition de type adh rence qui a  t  appliqu e. Le module de vitesse (figures 6.24, 6.26 et 6.28) est  galement en bon accord avec la solution de r f rence. De la m me mani re, l'augmentation du nombre de Reynolds am liore cet accord, ce qui semble indiquer que le mod le SPH pour les  quations de Navier-Stokes, se comporte mieux   nombre de Reynolds mod r .

Une donn e importante de cet  coulement est l' volution de la tra n e hydrodynamique g n r e par le carr  au cours du temps. La fronti re immerg e permet d'acc der directement aux forces appliqu es au carr  par le fluide environnant : ces forces sont les termes f_i de l' quation 6.2 pour toutes les particules fluides situ es   l'ext rieur du carr  (ensemble Ω^+ sur la figure 6.2). Le



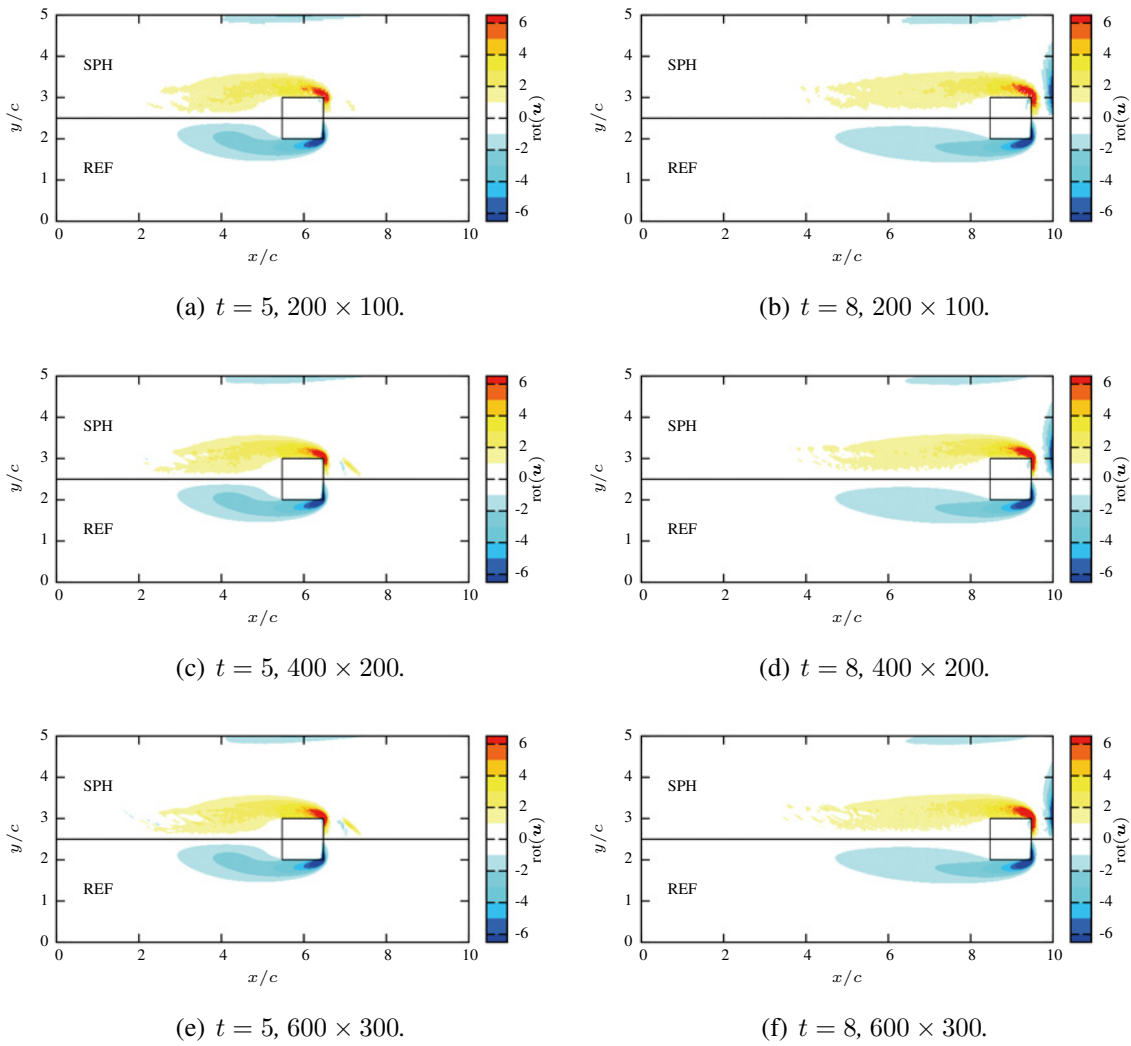


FIGURE 6.23 – Vorticité autour du carré forcé à $Re = 50$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour trois résolutions différentes 200×100 , 400×200 et 600×300 .

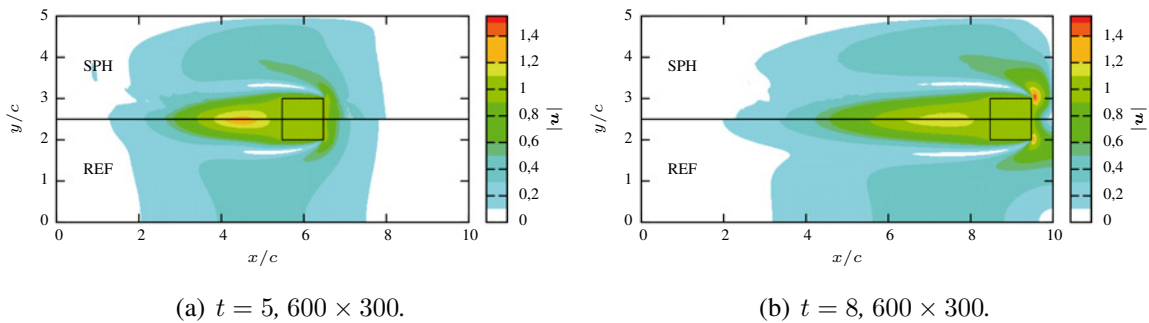


FIGURE 6.24 – Vitesse autour du carré forcé à $Re = 50$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour 600×300 particules.

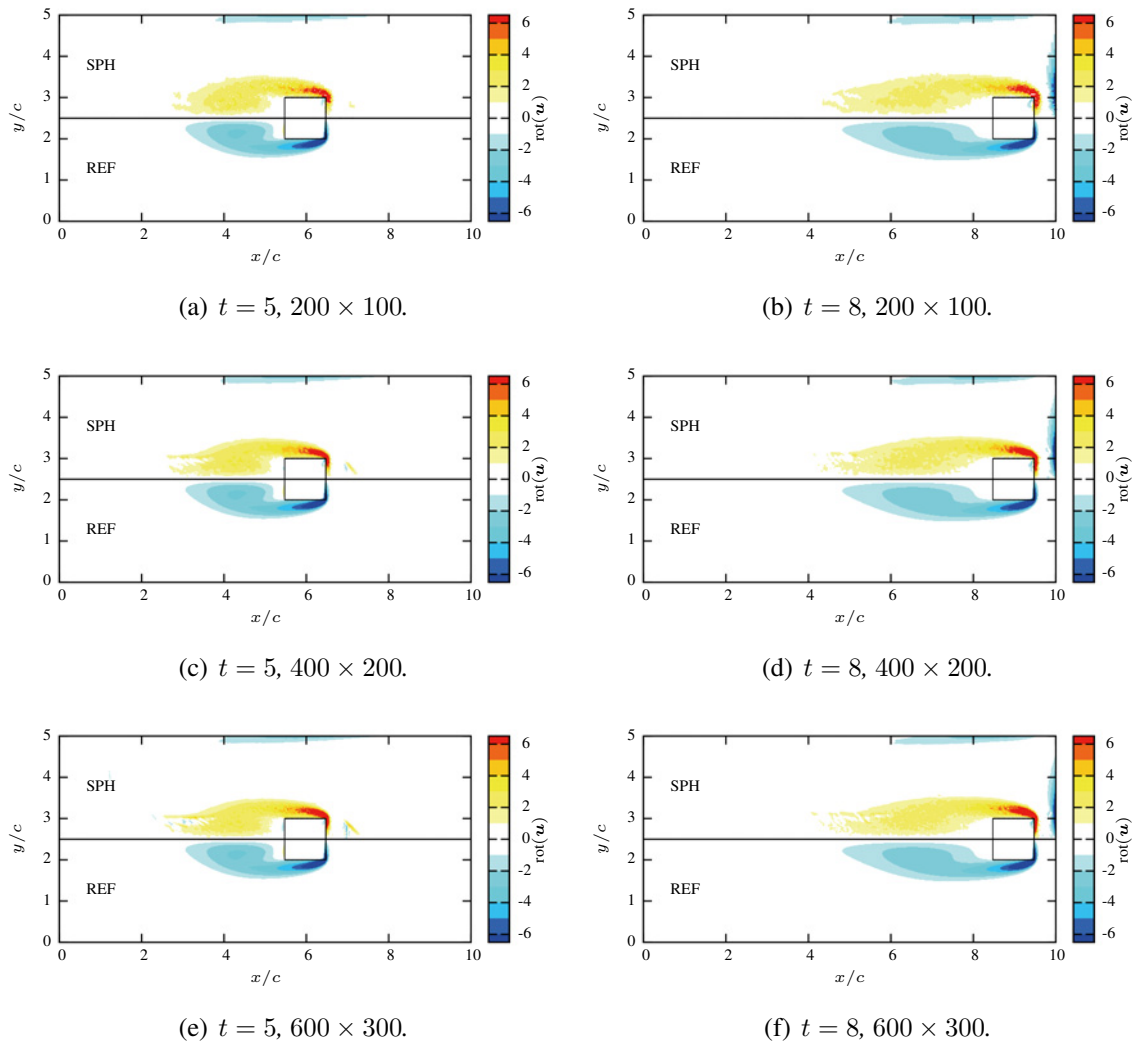


FIGURE 6.25 – Vorticité autour du carré forcé à $Re = 100$ simulé à l’aide du modèle SPH, comparée à la solution de référence (REF) pour trois résolutions différentes 200×100 , 400×200 et 600×300 .

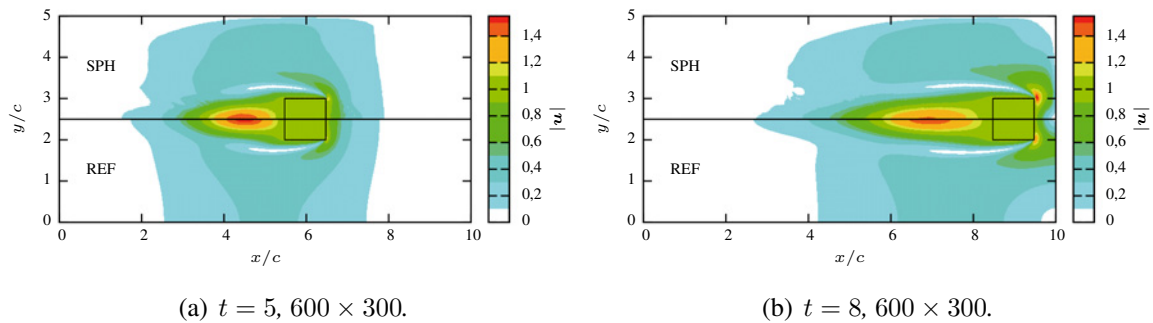
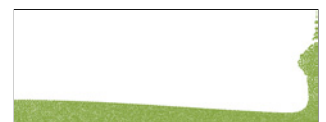


FIGURE 6.26 – Vitesse autour du carré forcé à $Re = 100$ simulé à l’aide du modèle SPH, comparée à la solution de référence (REF) pour 600×300 particules.



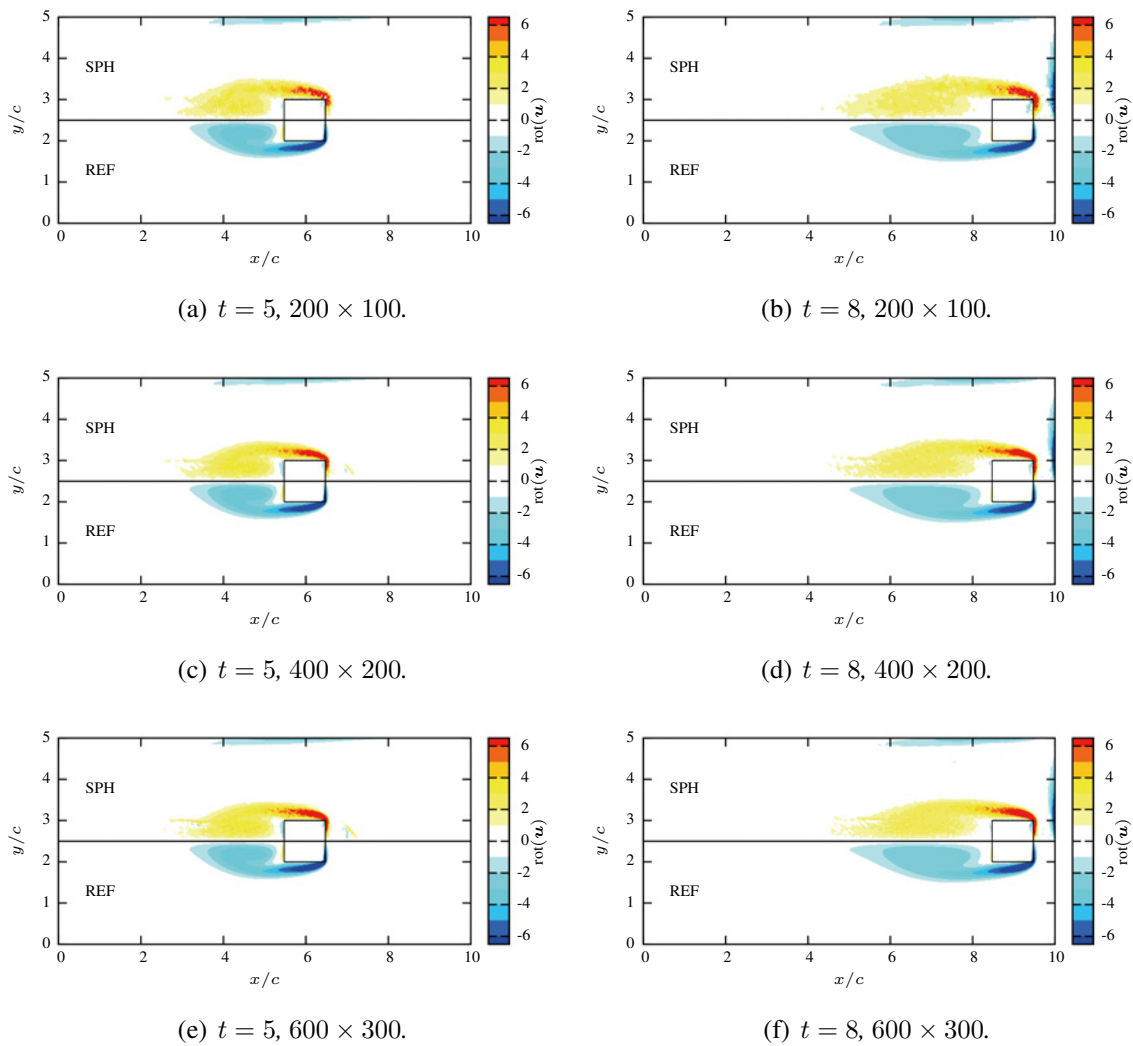


FIGURE 6.27 – Vorticité autour du carré forcé à $Re = 150$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour trois résolutions différentes 200×100 , 400×200 et 600×300 .

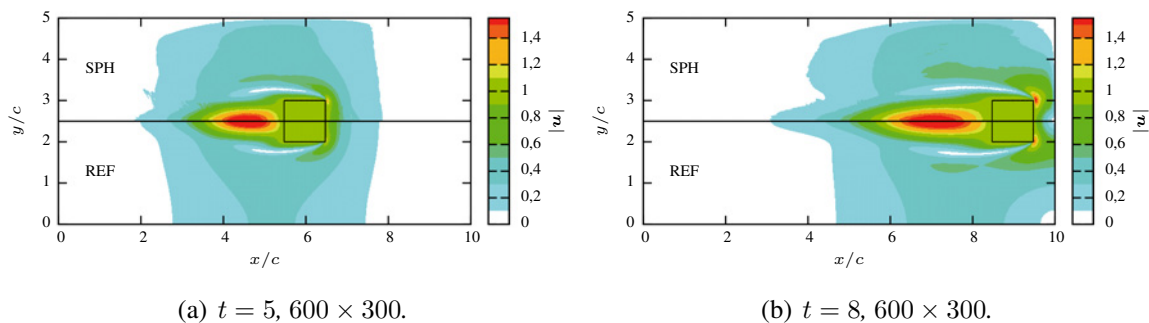


FIGURE 6.28 – Vitesse autour du carré forcé à $Re = 150$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour 600×300 particules.

coefficient de traînée s'obtient alors simplement par :

$$C_d = \frac{F_d}{\frac{1}{2}\rho_0 (u_c^{\max})^2 S} = \frac{\sum_{i \in \Omega^+} \mathbf{f}_i \cdot \mathbf{x} m_i}{\frac{1}{2}\rho_0 (u_c^{\max})^2 S} \quad (6.28)$$

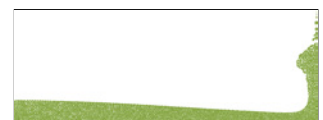
Cet accès direct aux forces sur la frontière immergée est un atout pour la modélisation de couplage fluide-solide. En revanche, cela ne permet pas de distinguer la partie du coefficient de traînée issue des forces de pression en amont et en aval du carré de la partie visqueuse (qui agit principalement sur le cisaillement latéral). Nous avons tenté plusieurs approches pour évaluer autrement ce coefficient. Notamment, il est possible de le calculer en intégrant le tenseur de déformation sur le contour de la frontière immergée. Bien que proches, les résultats obtenus par cette méthode sont plus éloignés de la solution de référence (6.28), principalement en raison de la continuité des contraintes au passage de la frontière. C'est à ce problème que la méthode de l'*interface immergée* tente d'apporter des réponses, par la prise en compte des sauts de part et d'autre de la frontière.

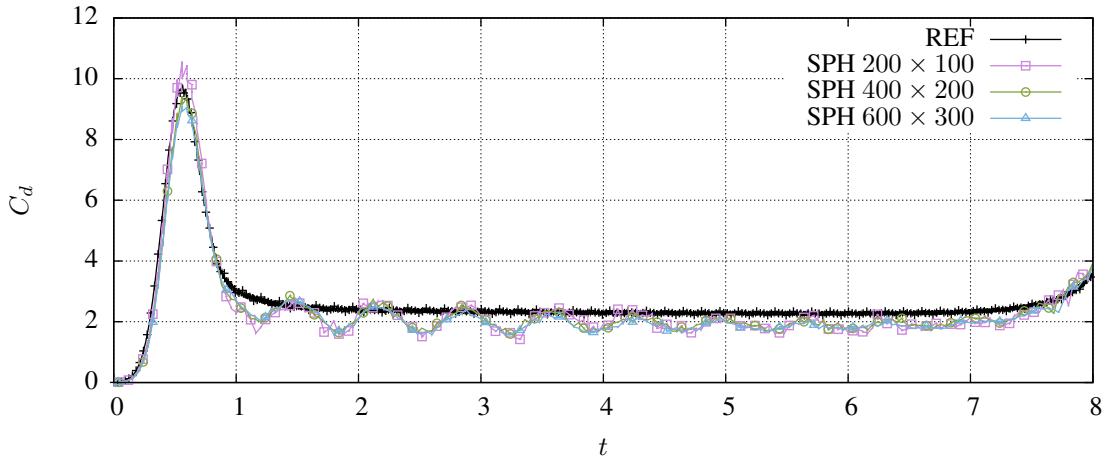
Ce point nécessite d'être approfondi, mais les coefficient de traînée obtenus sont très proches de ceux issus du modèle de référence, comme le montre la figure 6.29. La solution SPH converge d'ailleurs vers cette solution de référence quelque soit le nombre de Reynolds (surtout au niveau du pic à $t \approx 0,5$, correspondant à la phase d'accélération maximale), quand on augmente la discrétisation. L'accord est meilleur quand le nombre de Reynolds augmente. Durant la phase de translation à vitesse constante, la solution SPH oscille autour de la solution de référence. Ces oscillations correspondent au passage de l'onde de pression déjà évoquée, sur le carré. Elles sont donc entretenues par la faible compressibilité du fluide. Ces oscillations restent cependant très inférieures à celles obtenues avec le modèle incompressible de Lee *et al.* [126]. Il existe des travaux sur la modélisation de parois perméables et/ou non-réfléctives en SPH [118] qui pourrait permettre d'*évacuer* ces ondes.

6.2.4 Résultats à $Re = 300$ et $Re = 600$

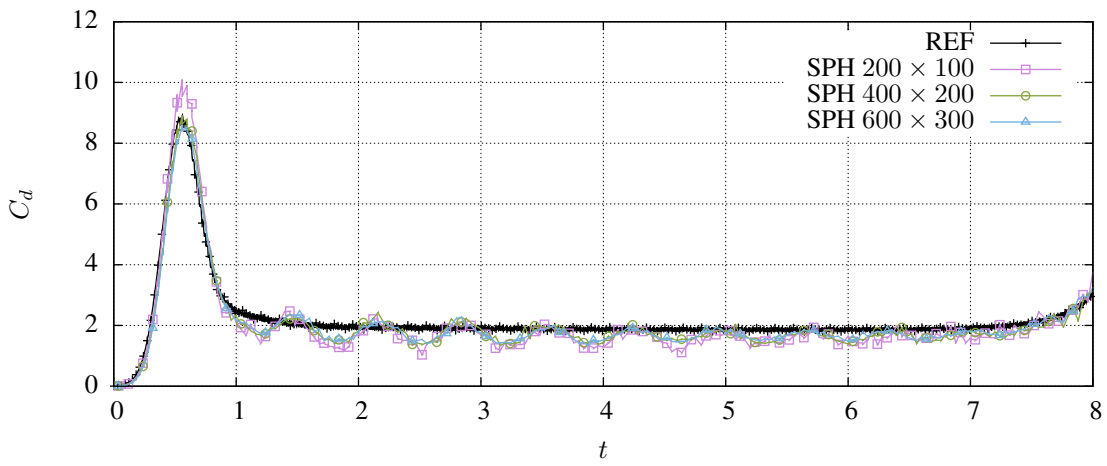
A titre illustratif, les cas $Re = 300$ et $Re = 600$ ont également été simulés à l'aide de la méthode SPH combinée à la *frontière immergée*, toujours sans remaillage, et avec amélioration des aller-retour particules-marqueurs de frontière par moindres carrés. Le cas $Re = 300$ a été simulé avec un champ de 1200×600 particules et 480 marqueurs de frontière uniformément répartis sur le carré. Les résultats obtenus sont représentés sur les figures 6.30 (module de vitesse) et 6.31 (vorticité). Les contours de vitesse restent très lisses malgré l'absence totale de remaillage. La forme des structures tourbillonnaires est également plus lisse, on observe d'ailleurs la formation de couches de vorticité *opposée* entre les tourbillons de sillage et la face arrière du carré. A $Re = 600$, simulé avec $2400 \times 1200 = 2,88 \times 10^6$ particules et 960 marqueurs de *frontière immergée*, ces structures tourbillonnaires prennent des formes encore plus complexes. Plusieurs pics de vorticité sont générés dans le sillage, qui restent persistant en aval du carré.

En raison de l'absence de données de comparaison, la validité du modèle pour des nombres

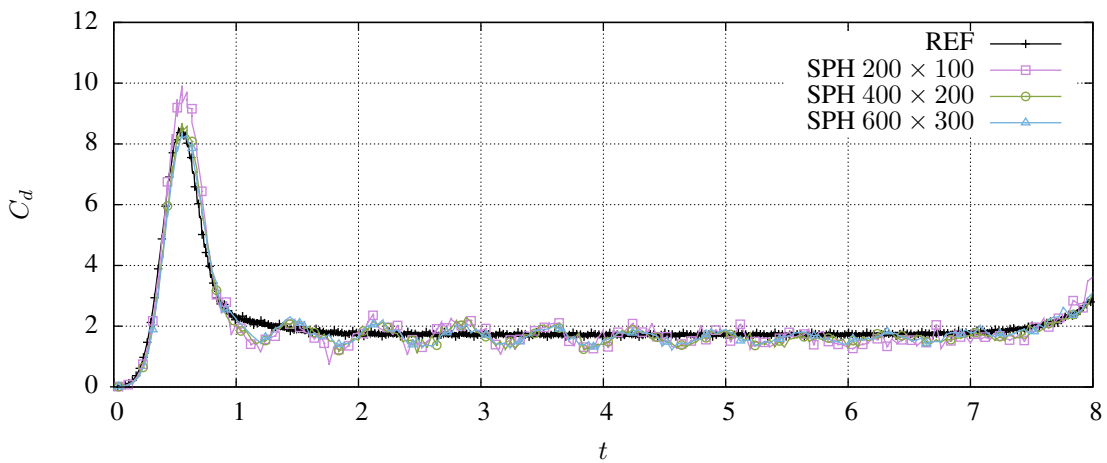




(a) $Re = 50$.



(b) $Re = 100$.



(c) $Re = 150$.

FIGURE 6.29 – Coefficient de traînée du carré forcé calculé à l'aide du modèle SPH, pour trois résolutions différentes 200×100 , 400×200 et 600×300 . La solution de référence (REF) est représentée pour comparaison.

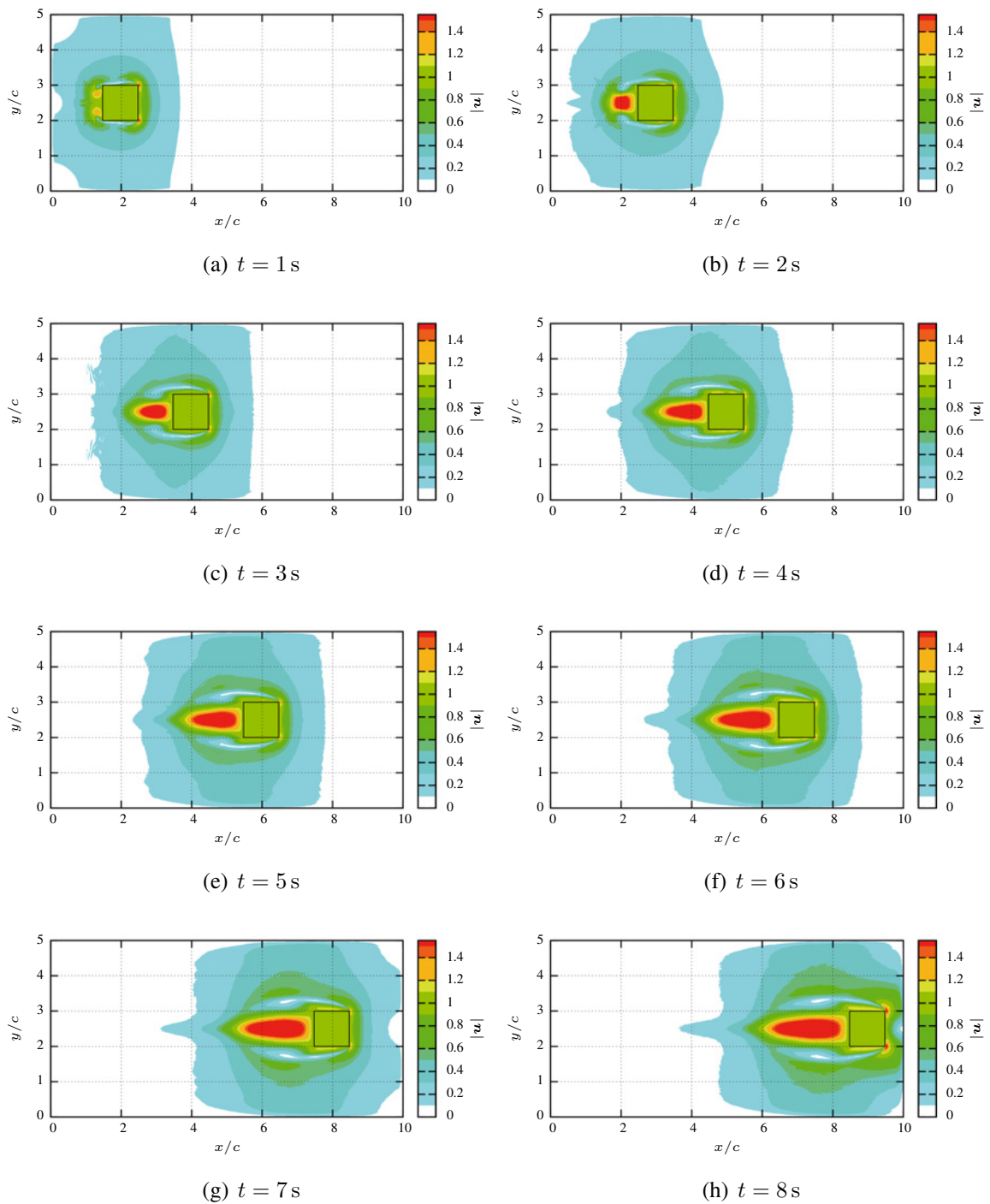
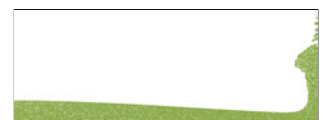


FIGURE 6.30 – Module de vitesse autour du carré forcé à $Re = 300$ simulé à l'aide du modèle SPH.



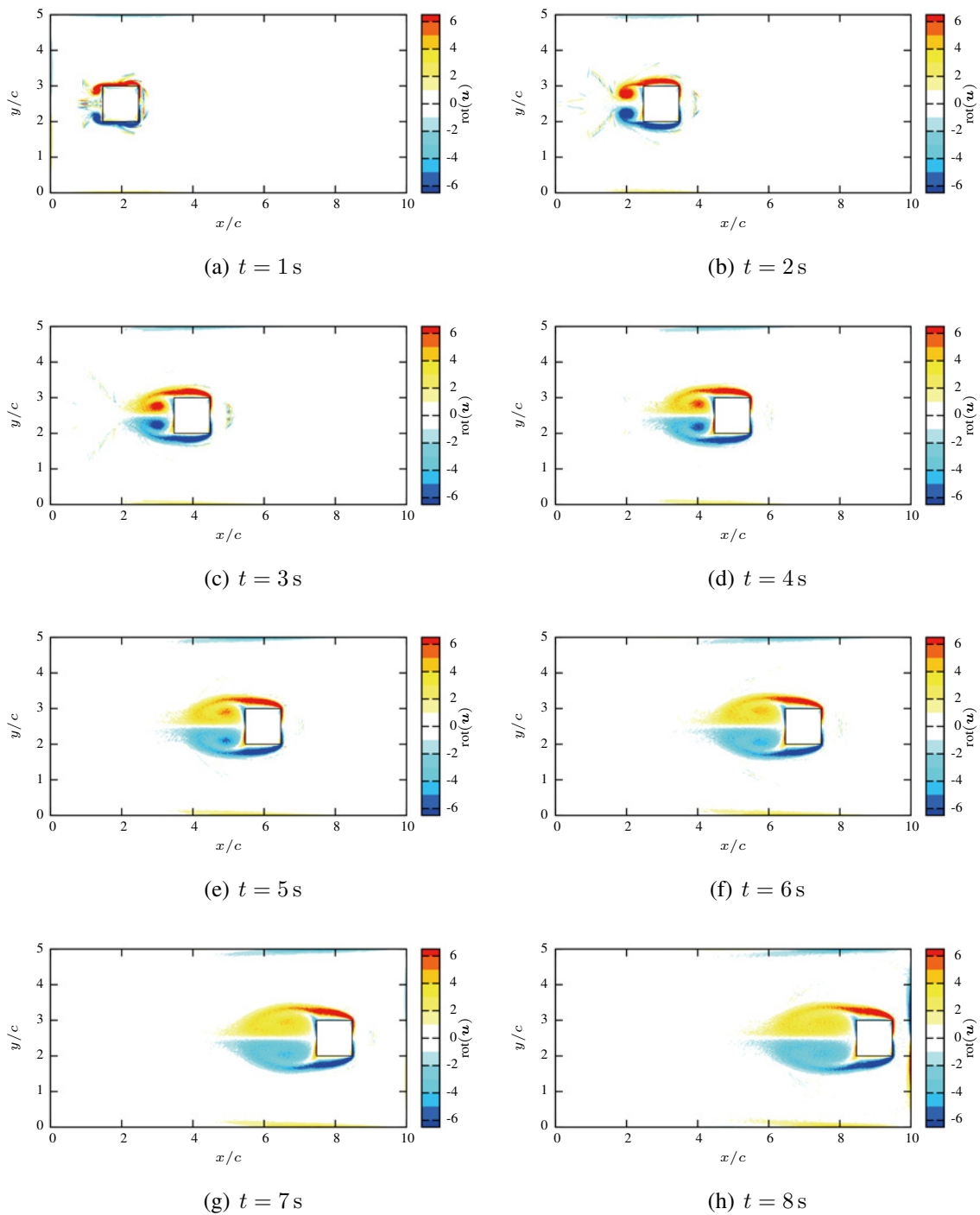


FIGURE 6.31 – Vorticité autour du carré forcé à $Re = 300$ simulé à l'aide du modèle SPH.

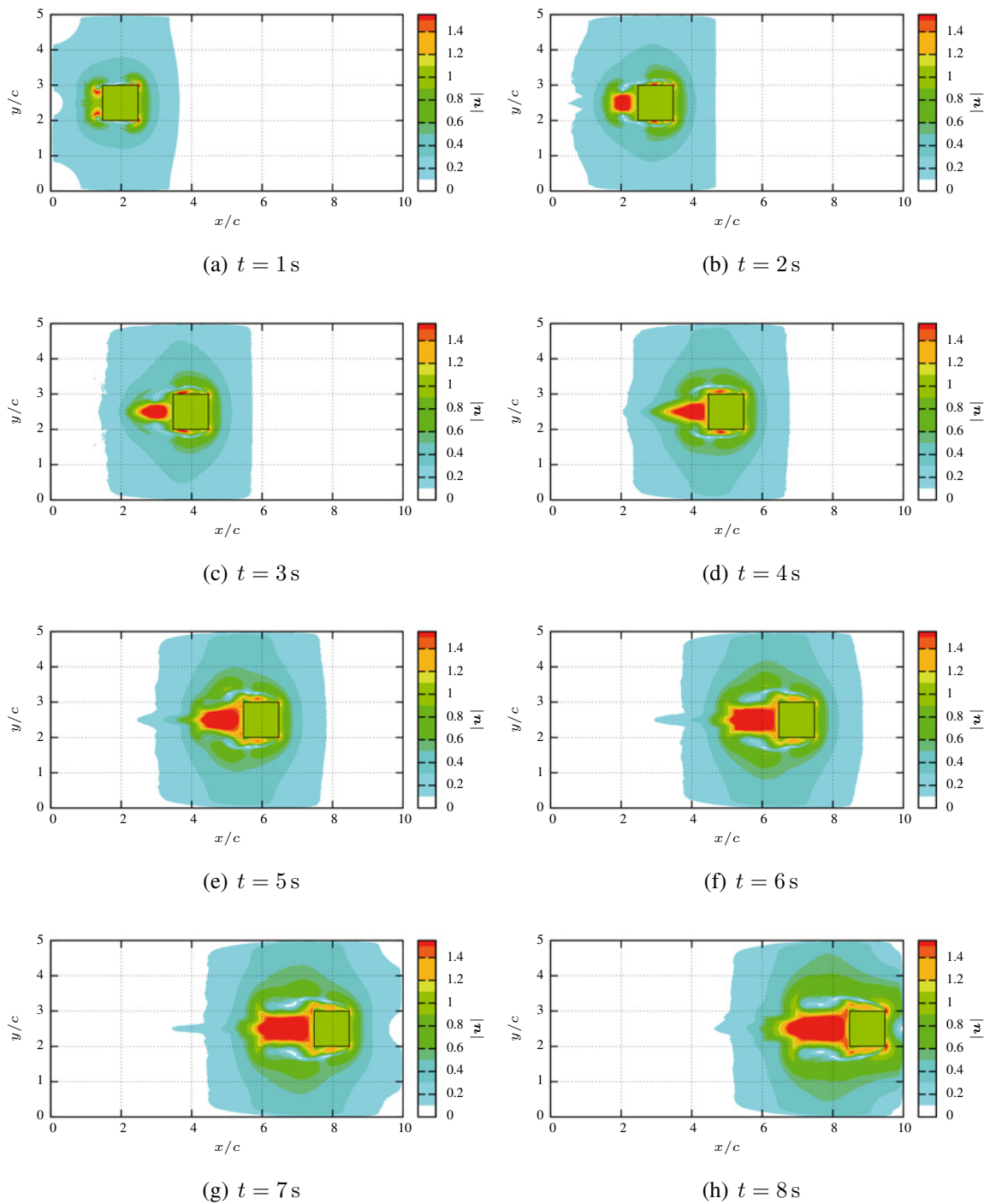
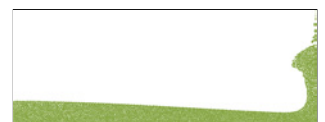


FIGURE 6.32 – Module de vitesse autour du carré forcé à $Re = 600$ simulé à l'aide du modèle SPH.



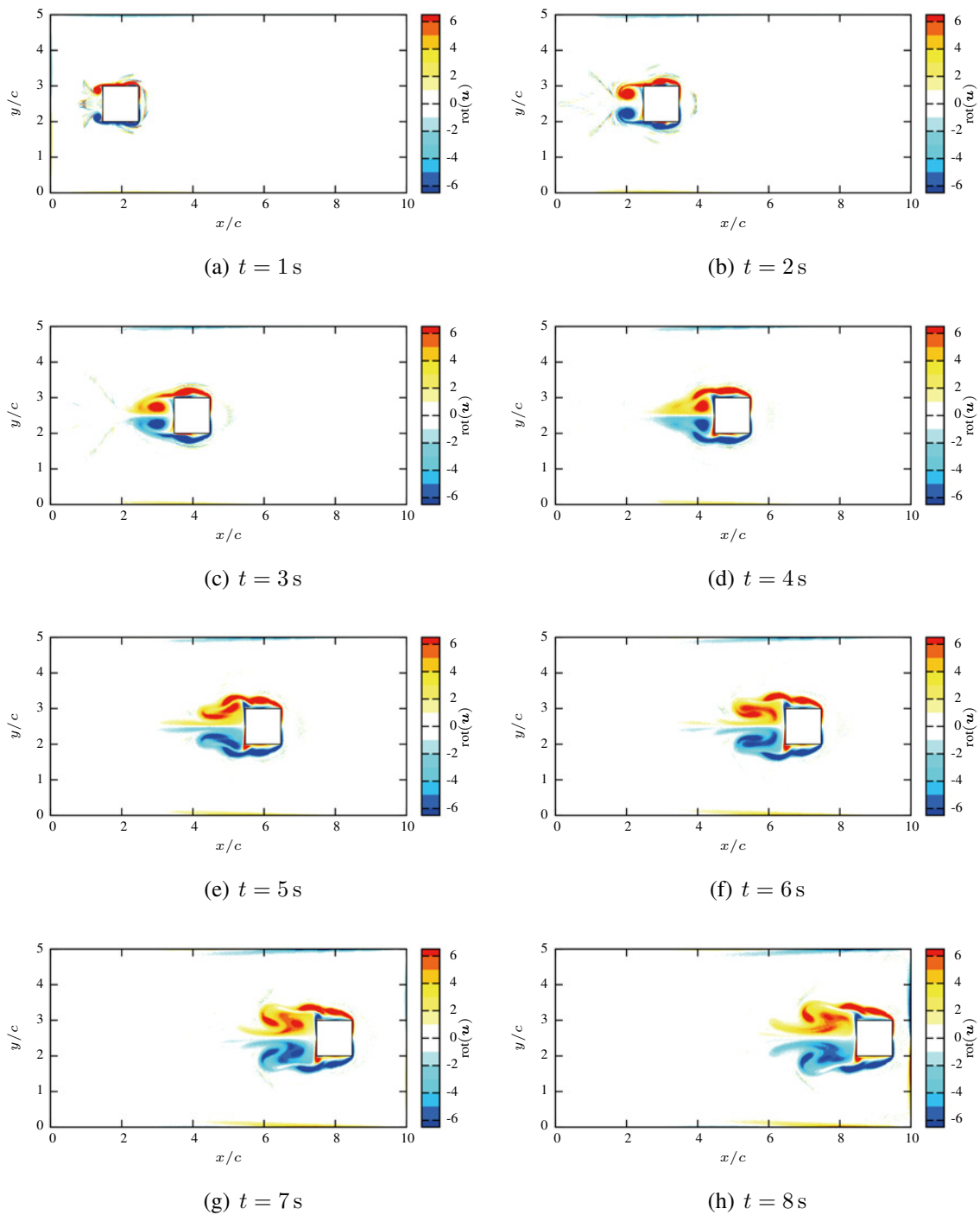


FIGURE 6.33 – Vorticité autour du carré forcé à $Re = 600$ simulé à l'aide du modèle SPH.

de Reynolds supérieurs reste à démontrer. D'autres cas-test, plus courants, peuvent également permettre de valider le code à *haut* Reynolds, comme celui du sillage d'un cylindre de section circulaire, bien plus documenté. La mise en œuvre de conditions limites du type entrées/sorties est à l'étude.



Chapitre 7

Propagation d'ondes de gravité en canal à houle

La méthode SPH, développée dans les chapitres précédents, permet désormais de modéliser :

- des écoulements à surface libre,
- la condition limite de type adhérence sur des corps solides indéformables par la méthode de la *frontière immergée*.

L'application initiale du modèle concernait l'action de la houle sur des ouvrages marins. Pour illustrer les capacités de la méthode, nous reprenons le problème évoqué dans l'introduction, de la plaque immergée soumise à l'action d'une houle monochromatique.

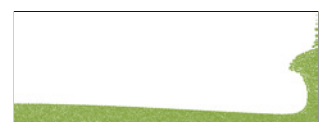
La propagation de la houle dans une boîte de type canal de longueur finie équipée d'un batteur plan est d'abord analysé. On y étudie l'évolution d'une onde de gravité et l'effet d'une plage d'amortissement *numérique* sur sa réflexion.

Dans ce même canal, les effets de la plaque sur l'écoulement et sur la propagation de l'onde sont ensuite étudiés.

7.1 Description du problème

7.1.1 Génération de la houle

Le domaine de calcul considéré est un canal à houle ouvert, comme ceux utilisés actuellement au LOMC pour l'étude expérimentale de problèmes d'hydrodynamique côtière. La figure 7.1 montre une vue schématique d'un canal en deux dimensions. Un batteur piston permet de générer une houle régulière de fréquence $f = \frac{1}{T} = \frac{\omega}{2\pi}$ (T est la période et ω la pulsation correspondante), l'amplitude du batteur est notée a . Les parois *extérieures* (batteur, fond et extrémité droite) du canal sont traitées numériquement par la technique des *particules fantômes*, que ce soit pour appliquer une condition de glissement ou d'adhérence. Le déplacement $x_b(t)$ du batteur est défini par la loi



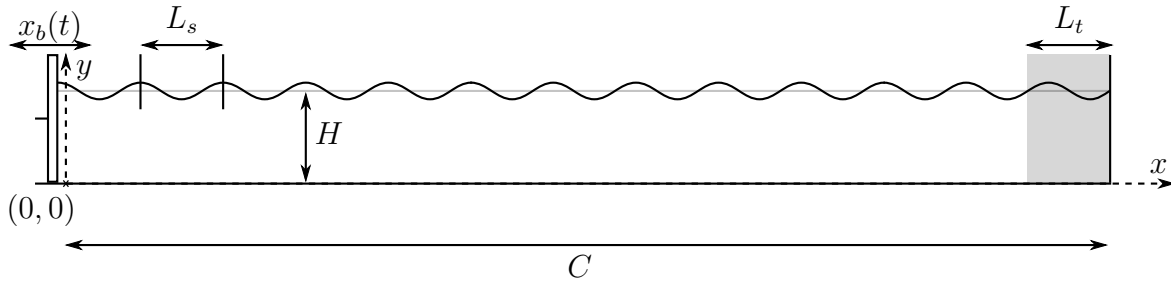


FIGURE 7.1 – Vue schématique du canal à houle de longueur C , rempli d'eau jusqu'à la hauteur H . Le batteur-piston, de position horizontale $x_b(t)$ (dépendant de sa fréquence d'oscillation f et de son amplitude a), génère une houle régulière de longueur d'onde principale L_s . Une zone d'amortissement de la houle est définie à l'extrémité droite du canal, de longueur L_t , la longueur d'onde théorique (issue de l'équation (7.5)).

suivante :

$$x_b(t) = \begin{cases} a \sin\left(\frac{\omega}{4k}t\right) \sin(\omega t) & \text{si } t < kT \\ a \sin(\omega t) & \text{sinon} \end{cases} \quad (7.1)$$

Par dérivation, on obtient la vitesse de déplacement horizontal du batteur, utile pour la définition des vitesses des *particules fantômes* (équation (5.16)). La figure 7.2 montre le déplacement de batteur obtenu en appliquant l'équation (7.1). La rampe de vitesse, d'une durée de k périodes (généralement fixée à $k = 2$), permet de mettre en mouvement les particules en évitant de leur imposer une accélération *trop brutale* en début de simulation.

7.1.2 Condition d'amortissement

Pour l'étude de la houle et de son action sur des obstacles, que ce soit en approche expérimentale ou numérique, la houle est souvent amortie en extrémité de canal afin d'éviter que l'onde réfléchi ne vienne perturber la zone d'observation. Nous avons choisi de définir une zone d'amortissement de longueur L_t , la longueur d'onde fondamentale de houle simulée, à l'extrémité droite du canal (figure 7.1). Pour toutes les particules situées dans cette zone, l'accélération est modifiée, en fin d'itération, par l'équation suivante :

$$\frac{D\mathbf{u}_i^*}{Dt} = A(x_i) \frac{D\mathbf{u}_i}{Dt} \quad (7.2)$$

avec

$$A(x_i) = \begin{cases} 1 & \text{si } x_i < C - L_t \\ \sin^2\left(\frac{(C-x_i)\pi}{2L_t}\right) & \text{sinon} \end{cases} \quad (7.3)$$

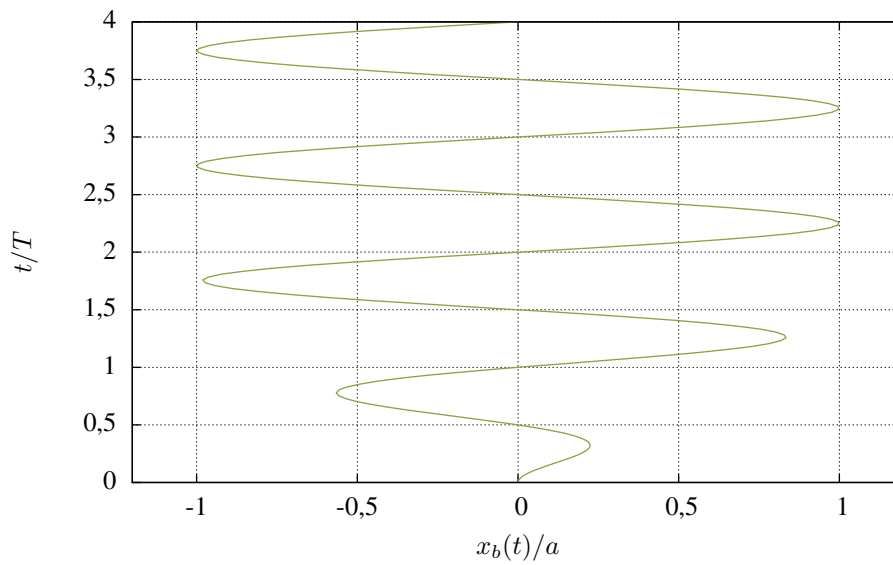


FIGURE 7.2 – Représentation graphique du mouvement du batteur-piston $x_b(t)$ en fonction du temps. La rampe a été appliquée sur 2 périodes ($k = 2$).

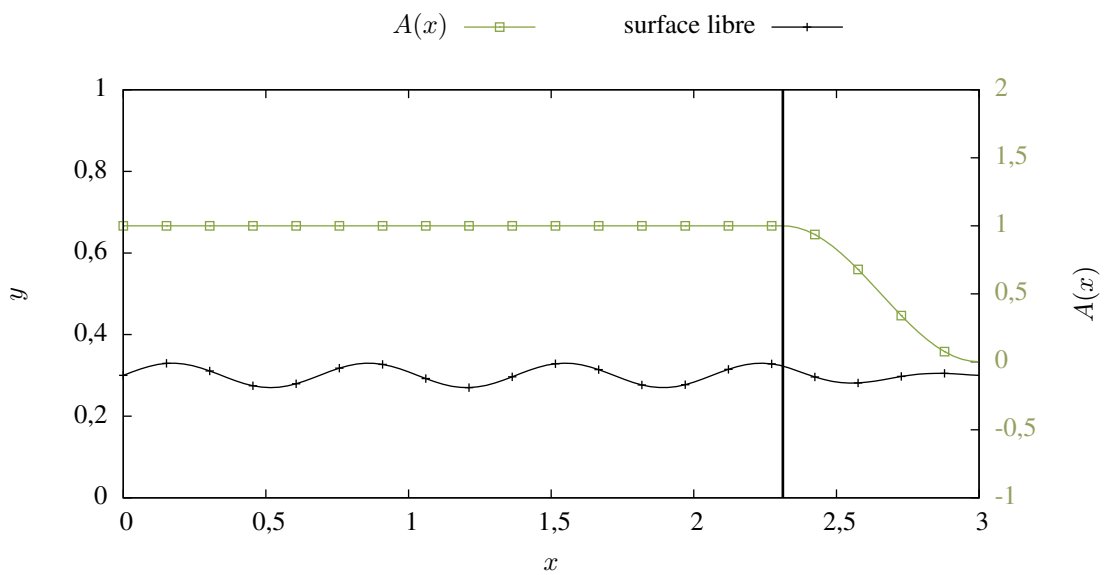
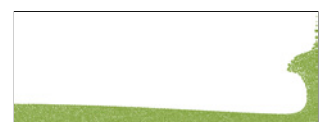


FIGURE 7.3 – Condition d’amortissement sur le canal par modification directe de l’accélération des particules.



La figure 7.3 représente la fonction $A(x)$ appliquée aux particules en fonction de leur position dans le canal (x étant la première composante du vecteur position \boldsymbol{x}). Elle décroît continument vers 0 quand la particule se rapproche de l'extrémité droite du canal, réduisant progressivement leur déplacement.

7.2 Propagation de la houle par SPH

Cette technique d'amortissement de la houle, peu coûteuse, a été appliquée à 4 différents champs de houle. Les fréquences associées d'oscillation du batteur-piston sont 1,25 Hz, 1,5 Hz, 1,75 Hz et 2 Hz. La relation de dispersion, issue du modèle de houle de Stokes, permet le calcul de la longueur d'onde pour une période de houle T et une hauteur d'eau H données. Au second ordre, on a :

$$L_t = \frac{gT^2}{2\pi} \tanh\left(2\pi \frac{H}{L_t}\right) \quad (7.4)$$

et au troisième ordre, elle fait intervenir un terme supplémentaire dépendant de la cambrure $a_h/L_{t,3}$ (a_h est l'amplitude de la houle émise) :

$$L_{t,3} = \frac{gT^2}{2\pi} \tanh\left(2\pi \frac{H}{L_{t,3}}\right) \left[1 + \left(\frac{2\pi a_h}{L_{t,3}}\right)^2 \frac{9 + 8 \cosh^4\left(2\pi \frac{H}{L_{t,3}}\right) - 8 \cosh^2\left(2\pi \frac{H}{L_{t,3}}\right)}{8 \sinh^4\left(2\pi \frac{H}{L_{t,3}}\right)} \right] \quad (7.5)$$

Les autres paramètres du canal sont : $a = 0,01$ m, $H = 0,3$ m et $C = 3$ m. Le code SPH a été exécuté avec les paramètres numériques suivants : un schéma d'intégration de type RK4, 50×500 particules pour un pas de temps $\Delta t = 0,0002$ s et une vitesse du son $c_0 = 10\sqrt{gH}$.

La figure 7.4 montre, sous la forme de diagrammes spatiotemporels, l'évolution de la houle pour les 4 cas simulés. L'intervalle de temps considéré est $[0; 20T]$ et la déformation de surface libre est représentée sur la longueur *utile* du canal, soit $x \in [0; C - L_t]$. Ces figures ne montrent pas de superposition d'une onde réfléchie sur l'onde incidente (comme cela est observable dans le cas de la réflexion par une plaque immergée, étudiée au paragraphe suivant, sur la figure 7.13).

Les longueurs d'ondes effectives L_s de houle calculées à partir des données numériques diffèrent légèrement des prédictions issues de l'équation de dispersion au second ordre (7.4). Les valeurs de L_t , $L_{t,3}$ et L_s sont données dans le tableau 7.1 pour comparaison. Une représentation graphique de ces valeurs est également fournie par la figure 7.5. Le modèle SPH vérifie donc l'équation de dispersion au second ordre, mais s'en écarte progressivement quand la longueur d'onde simulée diminue. Cette erreur peut être attribuée à deux facteurs :

- Les 4 houles simulées ici ayant une amplitude presque identique, la cambrure augmente avec la fréquence (on observe d'ailleurs une légère surélévation de la surface libre à proximité du batteur pour $f = 1,75$ Hz et $f = 2$ Hz, sur la figure 7.13). Nous avons donc simulé le

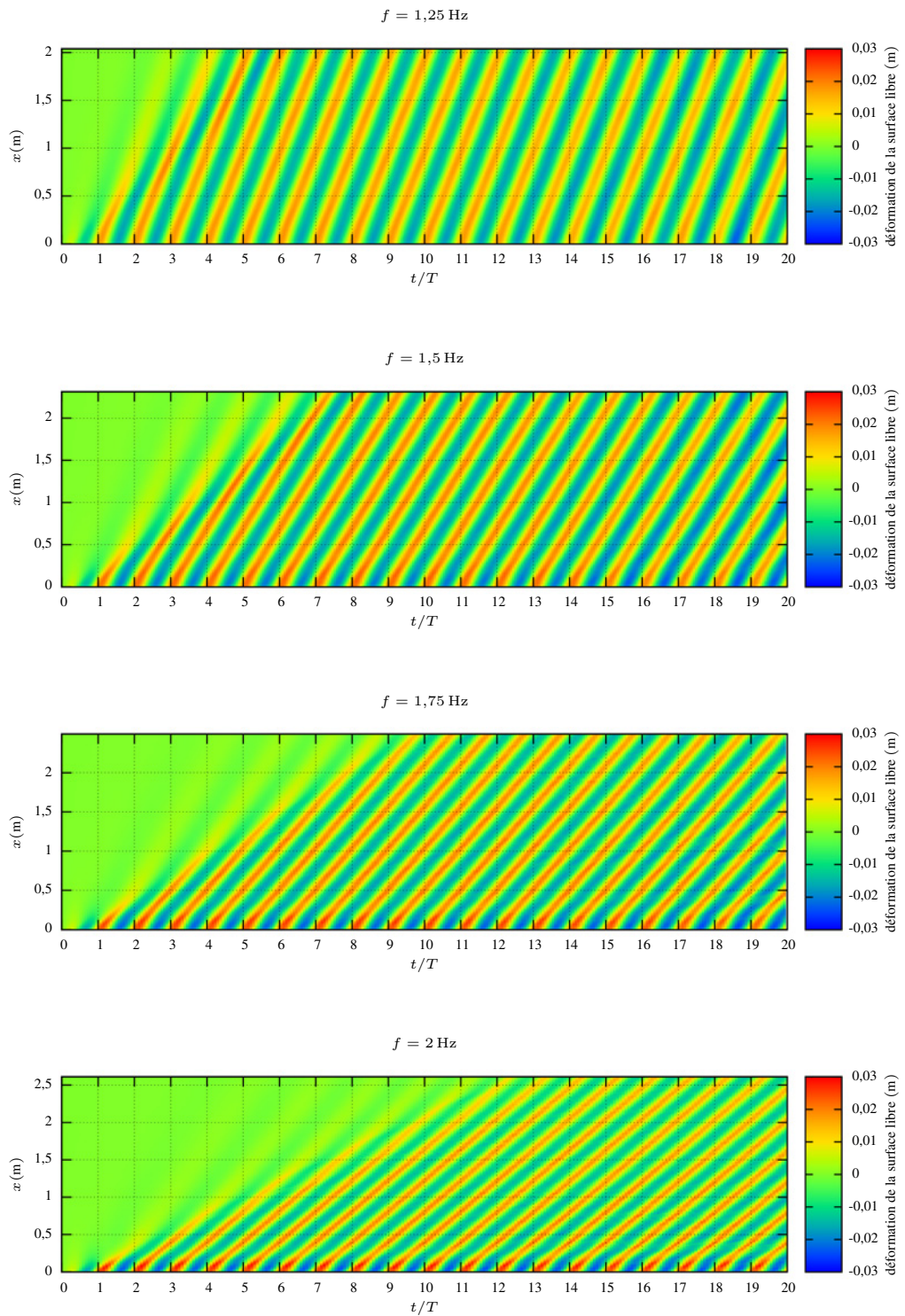
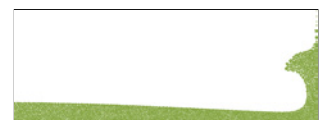


FIGURE 7.4 – Déformation spatio-temporelle de la surface libre, $t/T \in [0; 20]$ et $x \in [0; C - L_t]$, pour 4 périodes de houle différentes.



f	L_s	L_t	$L_{t,3}$	$\frac{L_s-L_t}{L_t}$	$\frac{L_s-L_{t,3}}{L_{t,3}}$	$\frac{a_h}{L_s}$
1,25 Hz	0,967 m	0,962 m	0,978 m	0,6 %	-1,2 %	0,021
1,5 Hz	0,697 m	0,689 m	0,711 m	1,1 %	-2 %	0,029
1,75 Hz	0,523 m	0,509 m	0,54 m	2,7 %	-3,2 %	0,038
2 Hz	0,412 m	0,39 m	0,431 m	5,6 %	-4,4 %	0,049

TABLE 7.1 – Longueurs d’ondes théoriques (L_t , $L_{t,3}$) et simulées L_s pour chacun des 4 cas de houle considérés. Erreurs par rapport aux longueurs d’ondes théoriques et cambrure (l’amplitude de la houle générée a été estimée à $a_h = 0,02$ m quelque soit le cas).

cas $f = 1,75$ Hz pour une amplitude de batteur réduite à $a = 0,005$ m. La longueur d’onde simulée devient alors $L_s = 0,51$ m, soit une erreur de 0,26 % (tous les autres paramètres restant inchangés) pour une amplitude de houle estimée à $a_h = 0,01$ m. La relation au troisième ordre a été représentée pour illustrer l’effet de la cambrure sur la houle (figure 7.5), les résultats de la méthode SPH sont situées dans la zone délimitée par les deux courbes théoriques.

- Le nombre total de particules dans le canal est identique quelque soit la fréquence du batteur. Ainsi, le nombre de particules par longueur d’onde (soit le rapport L_s/h) diminue quand la fréquence augmente, ce qui peut également être à l’origine d’une atténuation de l’amplitude de l’onde au cours de sa propagation. Une analyse de l’impact du rapport L_s/h sur la conservation de la houle est en cours.

La figure 7.7 représente les enveloppes de surface libre obtenues par SPH entre 15 et 20 périodes de houle pour les 4 longueurs d’ondes choisies. En observant l’enveloppe de surface libre, il est possible d’évaluer le taux de réflexion d’un obstacle. En calculant l’épaisseur de l’enveloppe à un ventre (zone de plus grande déformation maximale de la surface libre) H_{\max} et à un nœud (zone de plus faible déformation maximale) H_{\min} en amont de l’obstacle (figure 7.6), une approximation du taux de réflexion est possible. Un ventre et un nœud sont espacés d’une distance $L_s/4$ en l’absence d’harmoniques du mode fondamental. La forme des enveloppes calculées à l’aide du modèle SPH rendent difficile la distinction des ventres et des nœuds. Cette méthode n’est donc pas applicable, en raison des très faibles variations observées. Cependant, une enveloppe d’amplitude constante indique que la réflexion est nulle, si cette amplitude est égale à celle de la houle incidente, ce qui est présentement le cas. La condition d’amortissement appliquée ici permet donc d’éviter en quasi-totalité la réflexion de la houle simulée. Une faible réflexion (estimée inférieure à 5%) peut être mesurée dans le cas $f = 1,25$ Hz. Dans les autres cas, la dissipation de la houle incidente ne peut plus être négligée et celle-ci est également due à la diminution du nombre de

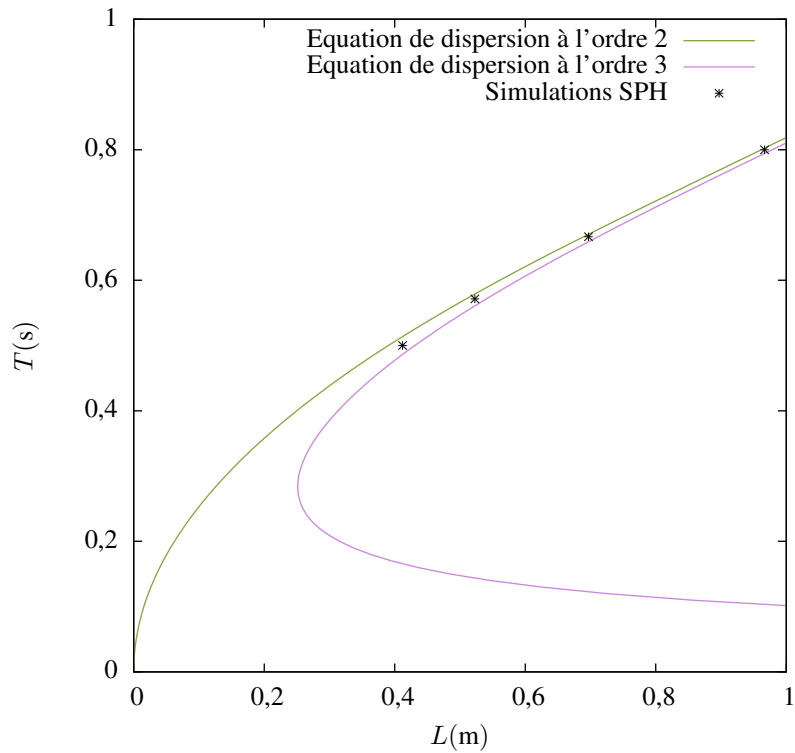


FIGURE 7.5 – Comparaison entre les équations de dispersion de la houle de Stokes au second ordre et au troisième ordre ($a_h = 0,02$ m) et les longueurs d’ondes obtenues par le modèle SPH.

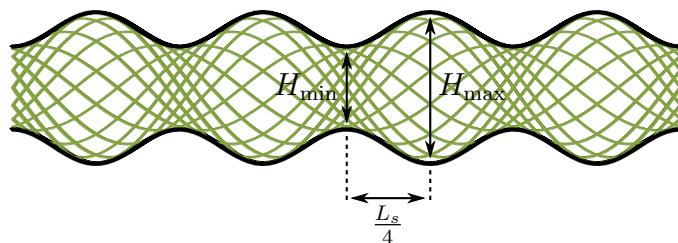
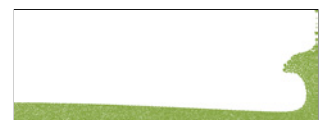


FIGURE 7.6 – Calcul du taux de réflexion par la méthode des ventres et des nœuds.



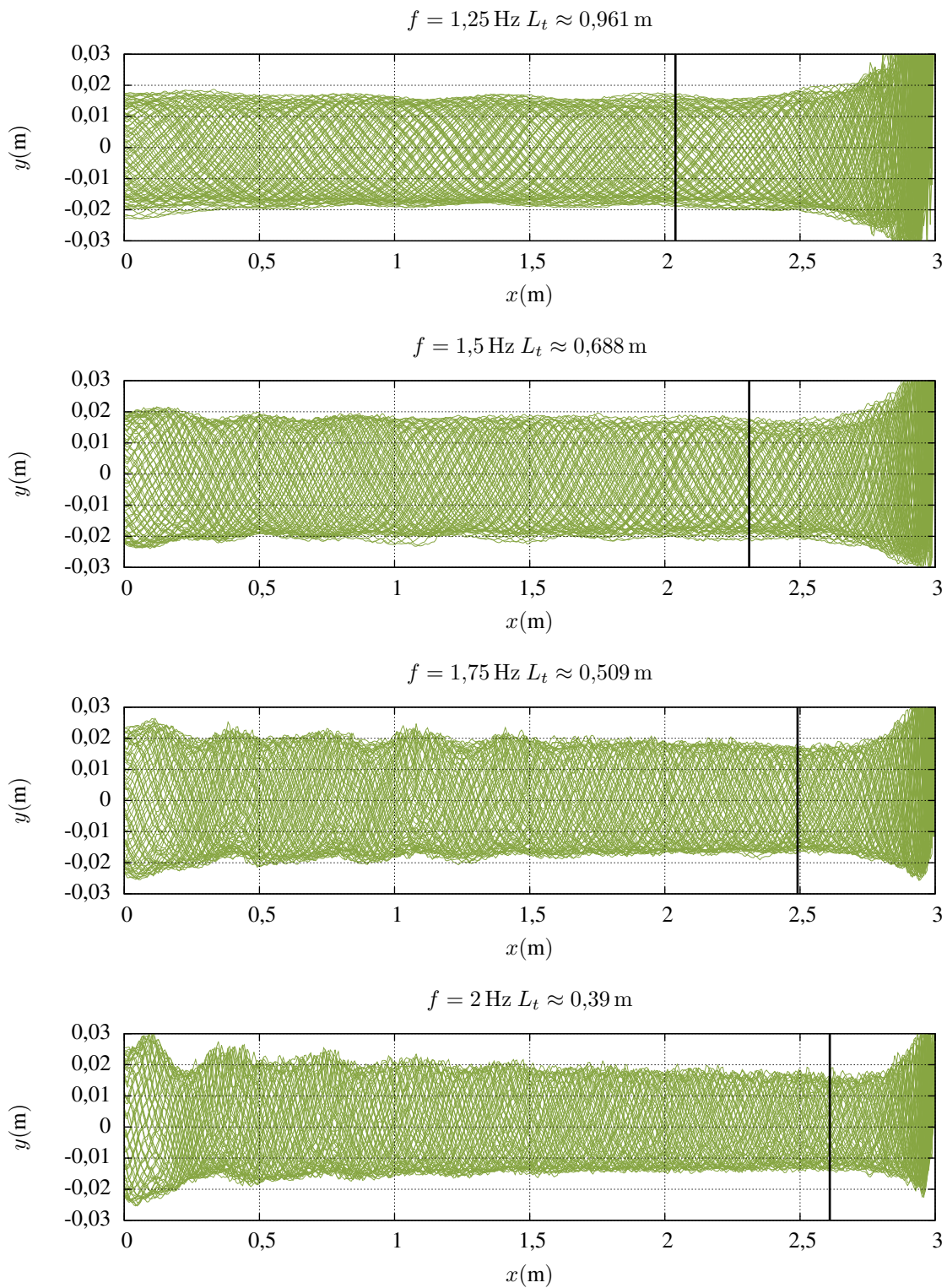


FIGURE 7.7 – Enveloppes de surface libre entre $t/T = 15$ et $t/T = 20$, pour 4 longueurs d'onde de houle, amorties à partir du trait noir épais à l'aide de l'équation (7.2)

particules par longueur d'onde.

D'autres approches pour la mesure de taux de réflexion sont également à l'étude, que ce soit par analyse spectrale de la surface libre ou en s'inspirant des techniques expérimentales [79, 19]. L'accès aux données complètes, telles que les diagrammes spatio-temporels, est ici un atout et devrait faciliter la mise en œuvre d'algorithmes de post-traitements efficaces.

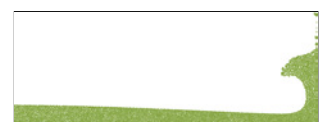
7.3 Plaque horizontale immergée en canal à houle

Une condition d'amortissement, pour la suppression des ondes réfléchies en canal, a été introduite au paragraphe précédent. Elle permet d'exploiter le modèle pour l'étude d'écoulements autour d'obstacles subissant l'action de la houle. Les applications sont multiples : amortisseurs de houle, structures *offshore* ... Plus particulièrement, nous nous sommes intéressés aux effets d'une plaque horizontale immergée sur la propagation de la houle.

Pour la protection des rivages, les plaques immergées, qui jouent le rôle d'amortisseurs de houle, sont une solution économique et discrète. La houle incidente est partiellement réfléchie en amont de la plaque et transmise en aval [18]. Le dimensionnement de ces ouvrages nécessite néanmoins une bonne connaissance de l'écoulement complexe entourant l'obstacle. En particulier, le cisaillement généré aux extrémités est à l'origine d'échappements tourbillonnaires qui peuvent interagir avec le lit du canal et la surface libre selon la position de la plaque [17, 181].

Quelques études théoriques et numériques existent sur le sujet. On trouve notamment les travaux de Patarapanich et Cheong [179] portant sur le taux de réflexion d'une plaque immergée dans un canal modélisé par la méthode des éléments finis. Par ailleurs, Boulier [17] s'est intéressé à la dynamique tourbillonnaire aux extrémités de la plaque en employant une méthode *Vortex*. Plusieurs approximations linéaires de ce problème existent également, en hypothèse de faible profondeur [210, 178], permettant de prédire l'amortissement de la plaque dans des conditions simples. Des solutions analytiques plus élaborées prennent en compte les discontinuités aux extrémités de la plaque en introduisant des modes évanescents [121]. L'écoulement autour d'une plaque immergée est d'ailleurs souvent ramené à celui autour d'un bloc rectangulaire posé au fond et recouvert par le fluide [222]. Il existe également des études numériques de ce cas [217, 130] et les résultats sont effectivement assez proches, pour certaines configurations géométriques. Au sein du laboratoire LOMC, l'étude de cet écoulement est actuellement réalisée à la fois par des approches expérimentales [188] et numériques (*Vortex*, ou SPH dans le cas présent).

La figure 7.8 montre la configuration du canal à houle. La plaque immergée de longueur B est placée à une hauteur h_i du fond.



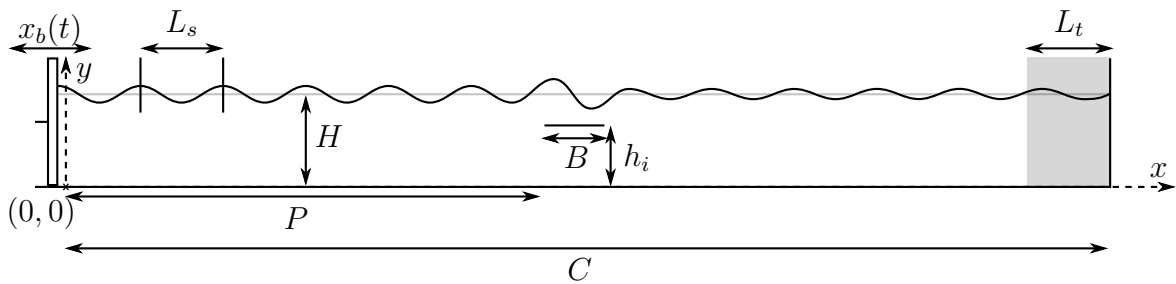


FIGURE 7.8 – Vue schématique du canal à houle de longueur C , rempli d'eau jusqu'à la hauteur H . Le batteur-piston, de position horizontale $x_b(t)$ (dépendant de sa fréquence f et de son amplitude a), génère une houle régulière de longueur d'onde L_s . Une plaque horizontale de longueur B est placée à une distance P de la position initiale du batteur et à une distance h_i du fond.

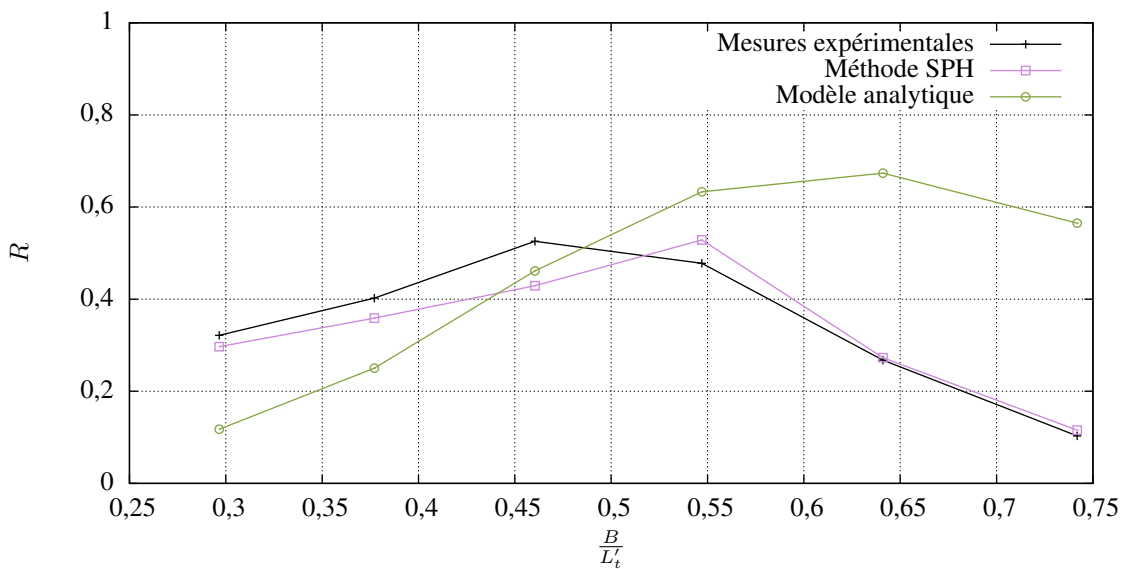
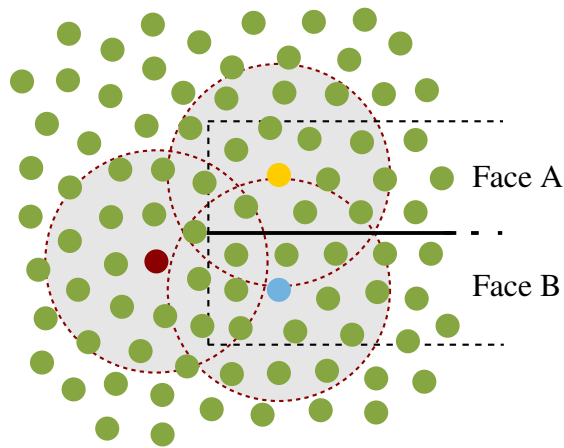
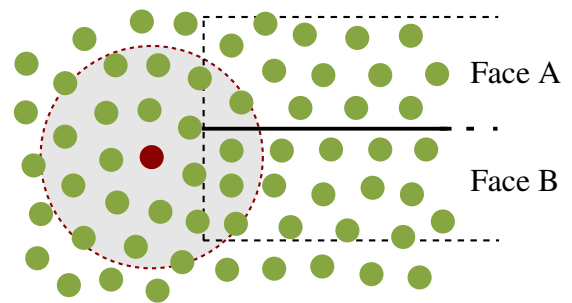


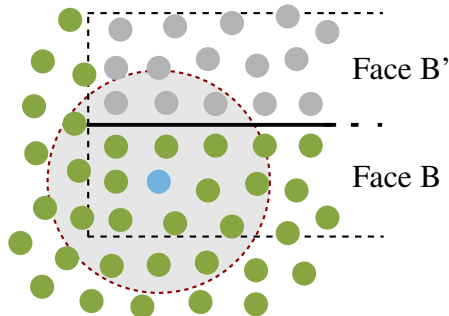
FIGURE 7.9 – Taux de réflexion R de la houle par la plaque immergée en fonction du rapport B/L_t , B étant la longueur de la plaque immergée et L_t la longueur d'onde théorique au second ordre (issue de l'équation 7.4 au dessus de la plaque).



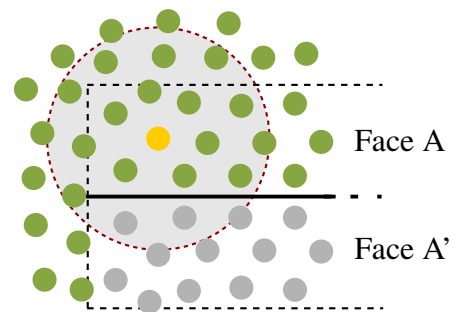
(a) Positions des particules du fluide autour de la plaque. Trois cas différents peuvent apparaître symbolisés par les trois couleurs.



(b) Cas de la particule rouge : on lui montre toutes les particules du fluide et aucune fantôme.

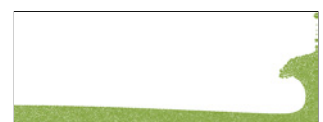


(c) Cas de la particule bleue : on lui montre toutes les particules du fluide sauf celles à une distance inférieure à 3ϵ de la face A, qui sont remplacées par les images des particules de la face B.



(d) Cas de la particule jaune : on lui montre toutes les particules du fluide sauf celles à une distance inférieure à 3ϵ de la face B, qui sont remplacées par les images des particules de la face A.

FIGURE 7.10 – Modélisation d'une paroi mince par la technique des particules fantômes et des visibilitées.



7.3.1 Viscosité négligée

Dans un premier temps, la propagation d'une houle régulière en canal a été simulée, à l'aide de notre modèle, en négligeant les effets de la viscosité. Les paramètres géométriques sont les suivants : $H = 0,2$ m, $C = 5$ m, $P = 2$ m, $B = 0,25$ m et $h_i = 0,15$ m. La déformation de la surface libre a été calculée pour 6 fréquences de houle différentes ($f = 0,814$ Hz, 1,02 Hz, 1,224 Hz, 1,428 Hz, 1,633 Hz et 1,839 Hz) et une amplitude de batteur $a = 0,005$ m. Les paramètres du code de calcul sont identiques au cas précédent. L'enveloppe de surface libre a permis, par la méthode des ventres et des nœuds (figure 7.6), de déterminer le taux de réflexion de la plaque en utilisant l'équation suivante :

$$R = \frac{H_{\max} - H_{\min}}{H_{\max} + H_{\min}} \quad (7.6)$$

Les résultats obtenus sont représentés sur la figure 7.9, montrant un bon accord avec les données expérimentales de Perret [181]. Un maximum est obtenu quand la longueur d'onde de houle estimée au dessus de la plaque (L'_i calculée par l'équation (7.4) en prenant $H - h_i$ comme hauteur d'eau) correspond à deux fois la longueur de celle-ci ($B/L'_i = 0,5$). La solution analytique présentée ici est basée sur une approximation potentielle de l'écoulement (Chagdali, comm. pers.). Elle a tendance à surestimer le taux de réflexion pour les faibles longueurs d'ondes de houle incidente et à le sous-estimer dans le cas des plus grandes longueurs d'ondes. La simulation numérique est donc intéressante sur ce type d'écoulement complexe, qu'il semble difficile de linéariser. Des résultats similaires ont été obtenus avec d'autres méthodes (Patarapanich et Cheong [179]), permettant de prédire les performances de telles structures marines. Mais la simulation pourrait aussi permettre de mieux comprendre les mécanismes de dissipation de l'énergie de la houle incidente et le rôle joué par le fond (sédimentaire, par exemple) et la surface dans ce processus. C'est dans ce contexte que l'application d'une méthode particulière, comme SPH, peut prendre tout son sens.

Les résultats présentés ici ont été obtenus en modélisant la plaque immergée par la technique des *particules fantômes*. Pour représenter une paroi mince, séparant deux zones du fluide distinctes n'interagissant pas, il est assez facile, par détection des positions relatives à la plaque, d'éliminer les interactions entre les particules placées au dessus et au dessous de l'obstacle. Cette approche conduirait naturellement à une condition de saut en vitesse et pression à travers la plaque, et à l'apparition de cisaillement aux extrémités (dans le cas visqueux), se rapprochant ainsi d'une condition de *Kutta-Joukowski*. Or, le traitement des extrémités s'avère, justement, en méthode SPH, problématique, comme ce fût le cas lors de la modélisation d'angles droits (chapitre 6).

La figure 7.10 illustre la technique des *visibilités* qui a été mise en œuvre dans le voisinage des extrémités de la plaque. Cette approche a permis d'obtenir des résultats satisfaisants au niveau de la surface libre et de l'estimation du coefficient de réflexion, mais l'écoulement a été fortement perturbé dans le voisinage de la plaque. En effet, lorsqu'une particule se déplace d'une zone à une autre, elle voit subitement un voisinage complètement différent, ce qui conduit à l'apparition de pics brutaux du champ de pression. La méthode SPH est assez robuste pour absorber ces pics,

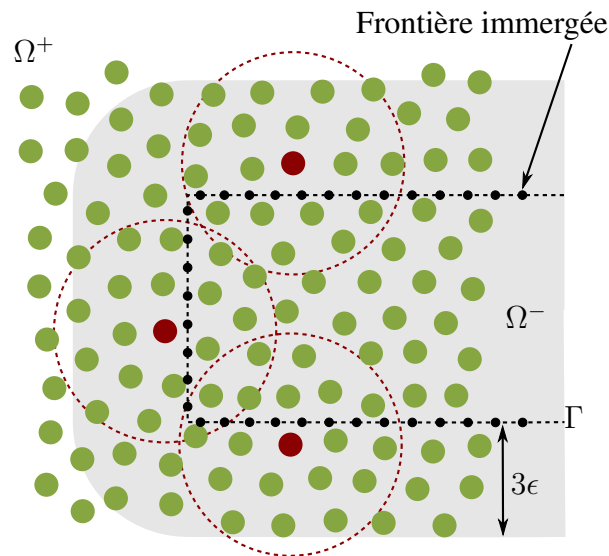


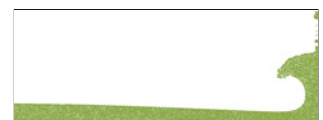
FIGURE 7.11 – Vue schématique du voisinage de l’extrémité de la plaque immergée. La taille des particules est définie de manière à avoir l’épaisseur de la plaque supérieure à 6ϵ , soit 8 particules quand $\kappa = 1,33$.

qui ont cependant des conséquences non-négligeables sur l’évolution de l’écoulement ; la solution obtenue est acceptable en termes de données issues de la surface libre, mais néanmoins contestable. Nous avons tenté d’améliorer le traitement spécifique des extrémités (par des approches similaires à celles de Colagrossi *et al.* [39] pour les angles droits ou à celle de Belytschko *et al.* [13] pour la modélisation de fractures dans les milieux solides), sans succès. C’est une des raisons qui nous a incité à développer une méthode des *frontières immergées* qui permet, par nature, de s’affranchir de ce type de situation problématique et permet d’imposer, nous l’avons vu au chapitre précédent, une condition d’adhérence de façon efficace, en vue d’étudier les effets de la viscosité sur l’écoulement.

7.3.2 Prise en compte de la viscosité

La méthode de la *frontière immergée* a été appliquée uniquement à la plaque. Le batteur et les autres parois du canal sont toujours représentées par des *particules fantômes*. Le fluide considéré est visqueux et la condition en vitesse sur la plaque est maintenant de type adhérence alors qu’elle est toujours de type glissement sur les autres parois solides. La plaque étant fixe, la vitesse U_k à imposer sur les marqueurs de la frontière (équation (6.10)) est constante et nulle. Contrairement au cas précédent, la plaque modélisée n’est plus d’épaisseur nulle, dans le but d’éviter que des particules situées au dessus de la plaque n’interagissent directement avec les particules et les marqueurs du dessous de la plaque (figure 7.11). L’épaisseur minimale de la plaque doit donc être supérieure ou égale à 6ϵ (avec ϵ , la distance de régularisation du noyau, définie au paragraphe 2.1.4).

Les paramètres géométriques employés pour ce résultat préliminaire de la *frontière immergée* appliquée à une configuration proche de celles étudiées au laboratoire, sont : $f = 1,75$ Hz, $a = 0,01$ m, $C = 2,5$ m, $H = 0,25$ m, $P = 1,2$ m (distance entre la position initiale du batteur et



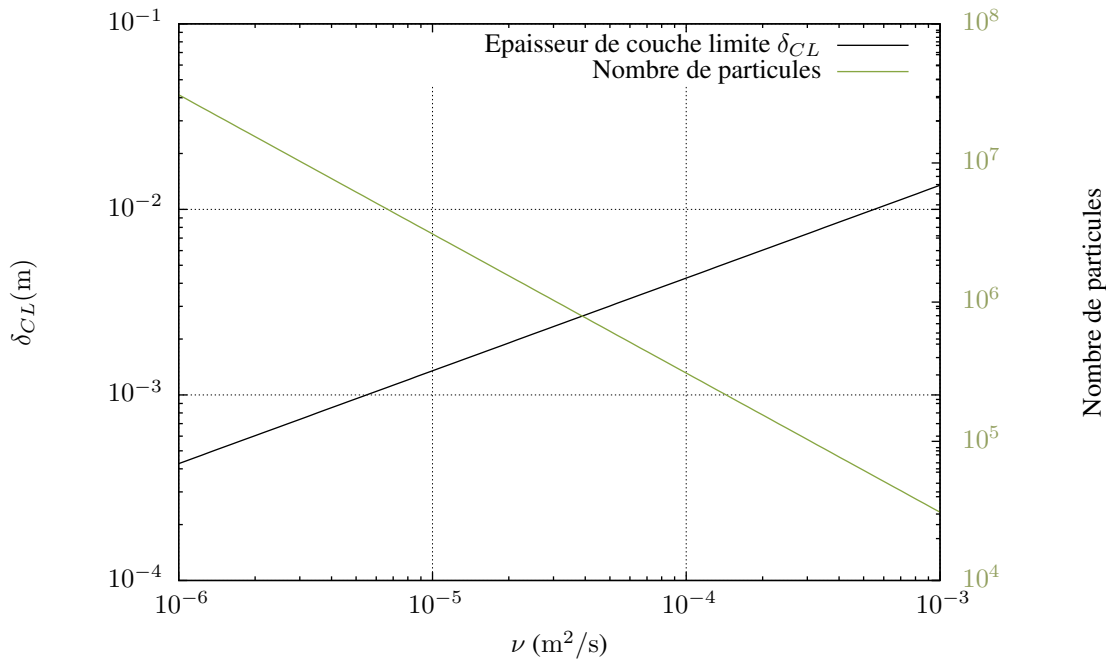


FIGURE 7.12 – Épaisseur de couche limite et nombre de particules nécessaires à la discrétisation du canal en fonction de la viscosité du fluide.

l'extrémité gauche de la plaque), $h_i = 0,15$ m (distance entre le fond et le centre géométrique de la plaque) et $B = 0,125$ m. Sous ces paramètres, la longueur d'onde fondamentale de houle prévue est $L_t \approx 0,508$ m.

D'importantes similarités ont été observées entre l'écoulement, considéré ici, autour d'une plaque immergée soumise à la houle et celui mettant en œuvre un bloc rectangulaire posé sur le fond du canal. Dans les deux cas, l'écoulement est de type oscillant, et les échappements tourbillonnaires constatés sont cadencés par la fréquence f de la houle incidente. Ici, nous cherchons à modéliser les effets de la viscosité sur l'écoulement, il est donc important de discrétiser suffisamment la masse fluide pour reproduire correctement le développement de la couche limite autour de la plaque, dont l'épaisseur peut être estimée selon Batchelor [11, p. 354], par $\delta_{CL} \approx \sqrt{2\nu/\omega}$. En considérant que 3 particules suffisent à discrétiser cette couche limite, on obtient les courbes du nombre total de particules dans le canal et d'épaisseur de couche limite représentées sur la figure 7.12.

Ainsi, si le fluide modélisé était de l'eau ($\nu = 1 \times 10^{-6}$ m²/s), le nombre de particules nécessaire serait de $2,9 \times 10^8$. Pour cette première approche, nous avons fixé la viscosité à $\nu = 1 \times 10^{-5}$ m²/s (10 fois supérieure à la viscosité réelle de l'eau) pour une épaisseur de couche limite estimée à $1,3 \times 10^{-3}$ m. Nous avons défini le pas de discrétisation en espace à $h = 6,25 \times 10^{-4}$ m pour avoir un nombre de particules sur cette épaisseur de couche limite $\delta_{CL}/h \approx 2$. Ainsi, la masse fluide dans le canal est représentée par 400 particules sur la hauteur d'eau H et 4000 particules sur la longueur C , soit un total de $1,6 \times 10^6$ particules. L'épaisseur de la plaque a été fixée à

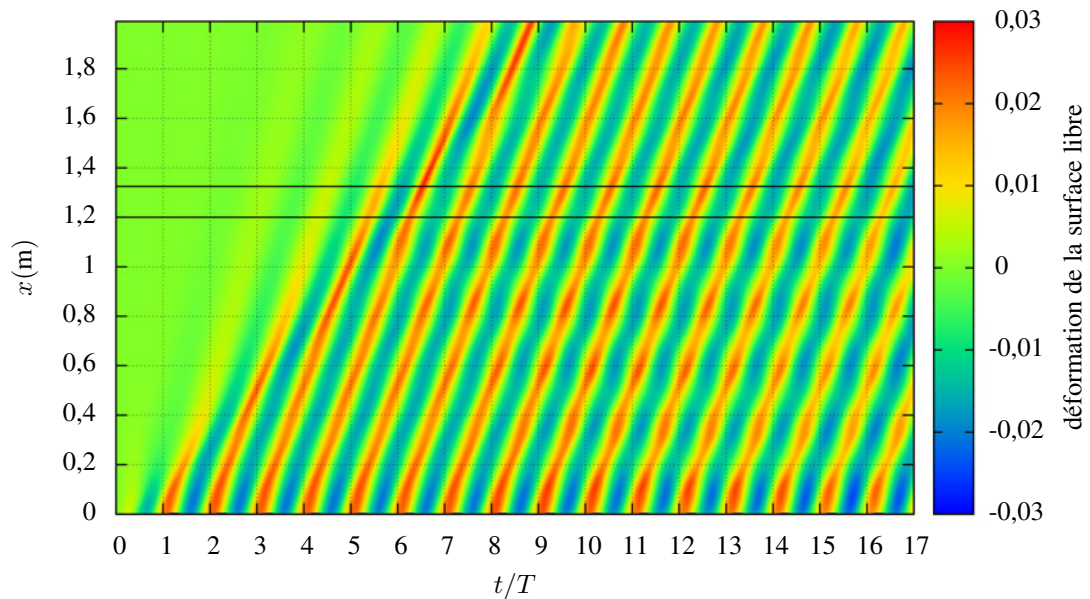


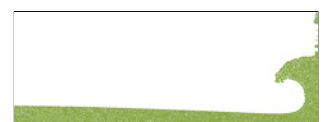
FIGURE 7.13 – Déformation de la surface libre en fonction du temps et de la position dans le canal sur les 17 premières périodes et sur la longueur utile du canal ($x \in [0; C - L_t]$). Les deux lignes horizontales symbolisent la position de la plaque immergée.

5×10^{-3} m de manière à avoir 8 particules (6ϵ) sur cette distance (figure 7.11). Le pas de temps a été fixé, en conséquence, à $\Delta t = 2 \times 10^{-5}$ s. Le nombre de Reynolds, basé sur la longueur de la plaque et sur la vitesse maximale du batteur, a été estimé à :

$$\text{Re} = \frac{u_{\max}^b B}{\nu} = \frac{a\omega B}{\nu} = \frac{a2\pi f B}{\nu} \approx 1200 \quad (7.7)$$

L'évolution de la surface libre sur les 17 premières périodes est représentée sur le diagramme spatiotemporel de la figure 7.13. On remarque que la houle en amont de la plaque est déformée par une onde réfléchiée dès la 6^{ème} période et que l'onde transmise est, en conséquence, atténuée. L'enveloppe de surface libre, une fois la houle établie ($t/T \in [10; 13]$), est représentée sur la figure 7.14. En appliquant la méthode des ventres et des nœuds, nous avons estimé le taux de réflexion à $R = 0,13$ (tableau 7.2). Encore une fois, l'analyse du diagramme spatiotemporel de la surface libre (figure 7.13) doit permettre d'enrichir les informations concernant la propagation de la houle.

La figure 7.15 représente le champ de vorticit  au voisinage de la plaque. Le mod le SPH coupl    la m thode de la *fronti re immerg e* parvient   simuler la formation des couches limites. La vorticit  sur les faces inf rieures et sup rieures de la plaque semble atteindre une valeur absolue maximale d s qu'une cr te ou un creux de houle passe au dessus de celle-ci. Deux c urs tourbillonnaires contra-rotatifs sont d tach s   chaque p riode en aval de la plaque (c t e plage



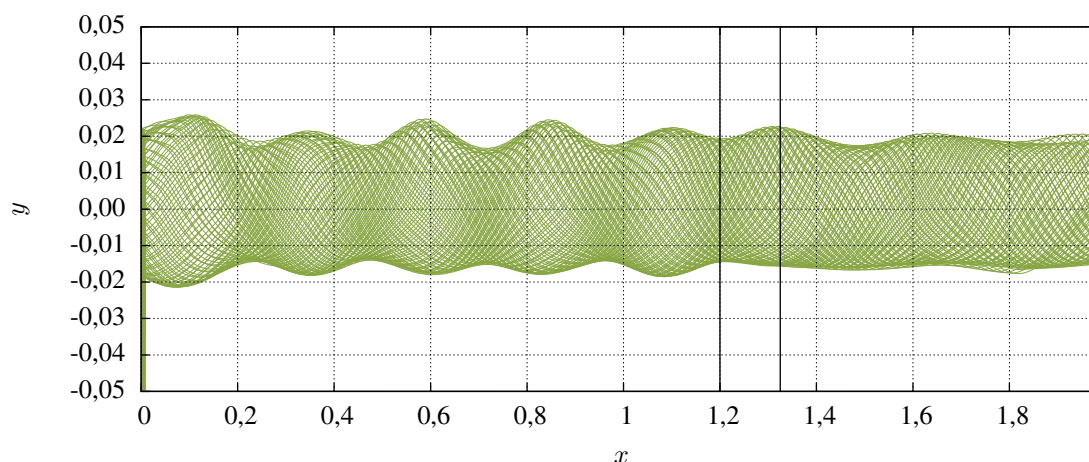


FIGURE 7.14 – Enveloppes de surface libre entre la 10^{ème} et la 13^{ème} période..

$x(m)$	$H_{\max}(m)$	$H_{\min}(m)$
$4,78 \times 10^{-1}$		$3,16 \times 10^{-2}$
$5,94 \times 10^{-1}$	$4,28 \times 10^{-2}$	
$7,16 \times 10^{-1}$		$3,18 \times 10^{-2}$
$8,50 \times 10^{-1}$	$4,24 \times 10^{-2}$	
$9,67 \times 10^{-1}$		$3,18 \times 10^{-2}$
$1,09 \times 10^0$	$4,09 \times 10^{-2}$	
Moyenne	$4,20 \times 10^{-2}$	$3,18 \times 10^{-2}$

TABLE 7.2 – Amplitudes maximales et minimales de l’enveloppe de surface libre

d’amortissement). Deux cœurs moins intenses sont également émis à l’extrémité amont (côté batteur), et plus rapidement diffusés au dessus de la plaque. La dynamique de ces tourbillons est un phénomène complexe influencé par la géométrie et la position de la plaque, la houle incidente et le fluide considéré. Ces résultats seront à confronter aux différentes approches numériques développées actuellement au laboratoire.

La méthode de la *frontière immergée* permet, nous l’avons vu au chapitre précédent, d’accéder directement aux efforts subis par la plaque. Les figures 7.16 et 7.17 montrent les efforts horizontaux F_x et verticaux F_y en fonction du temps. Ces efforts ont été calculés et adimensionnés de la manière

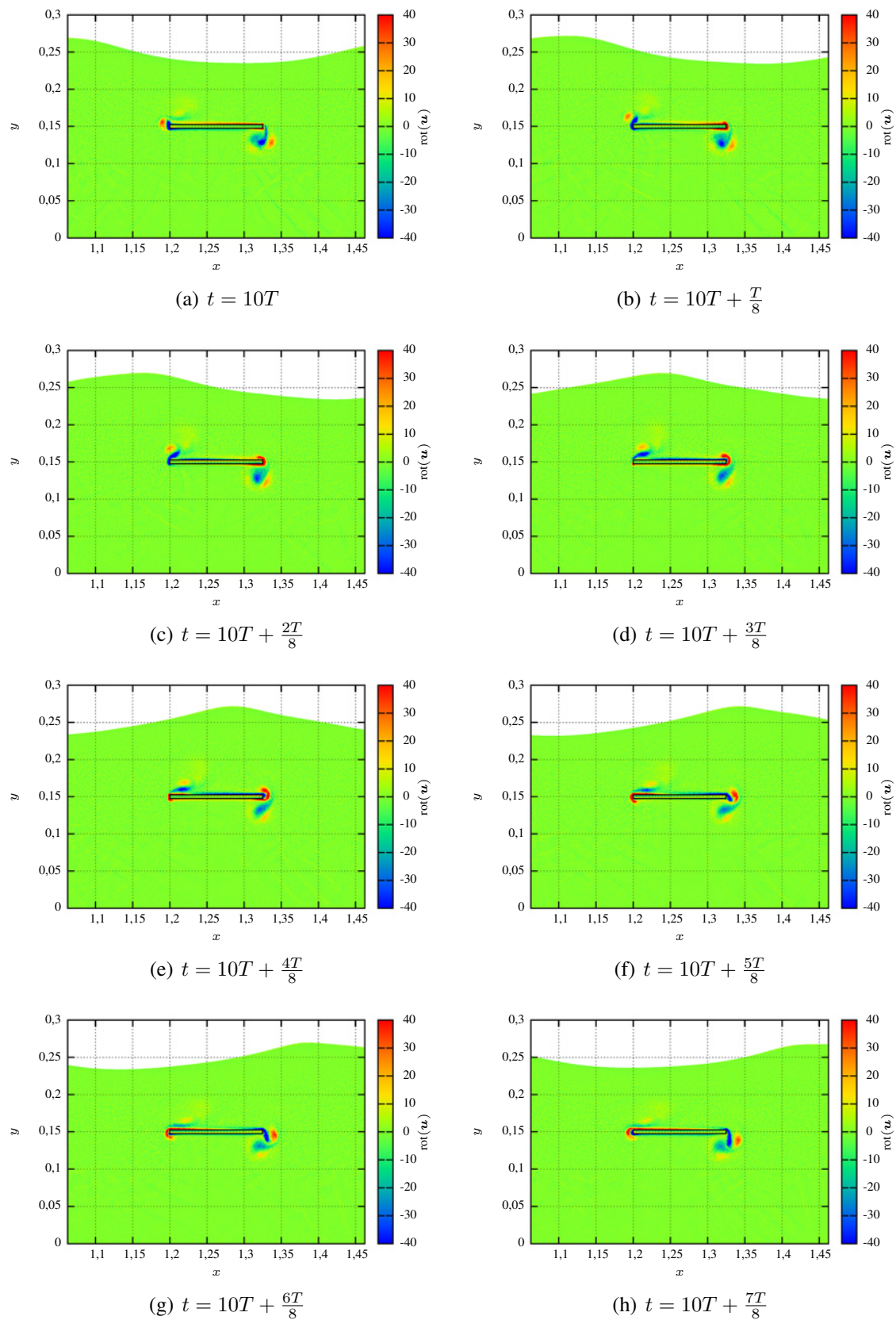
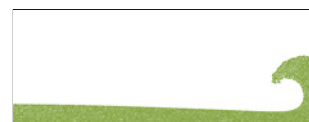


FIGURE 7.15 – Champ de tourbillon autour de la plaque immergée soumise à l'action de la houle en 8 instants de la 10^{ème} période.



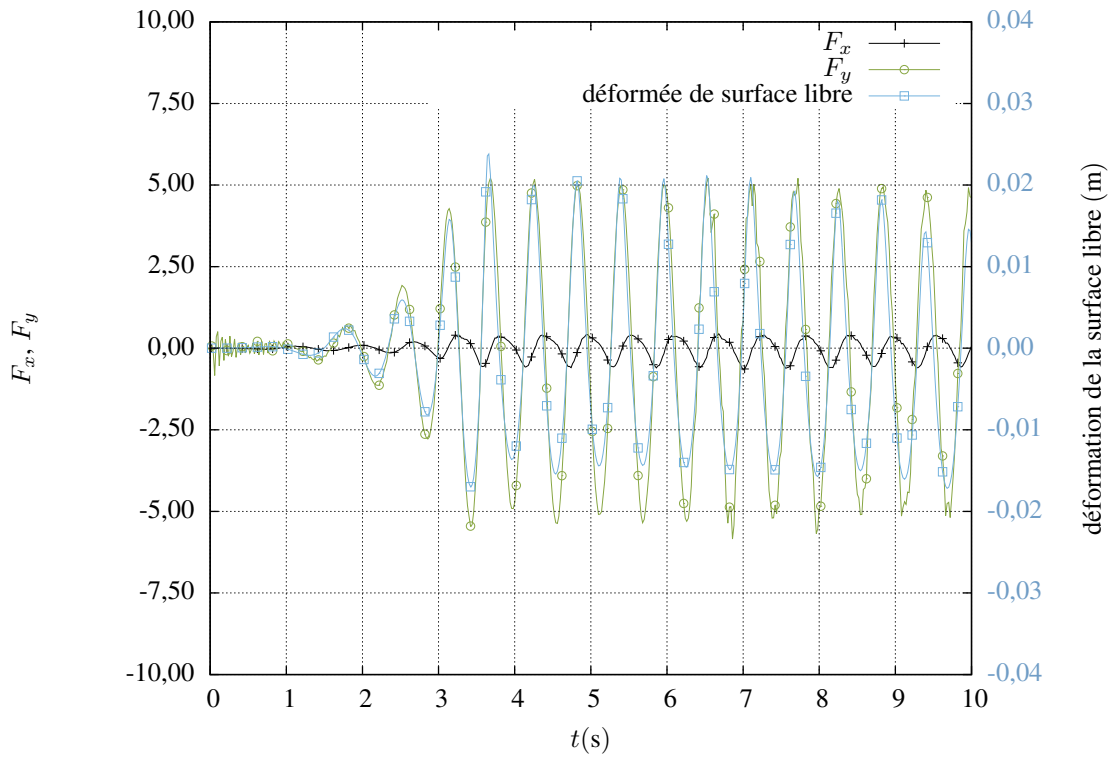


FIGURE 7.16 – Efforts horizontaux et verticaux sur la plaque en fonction du temps.

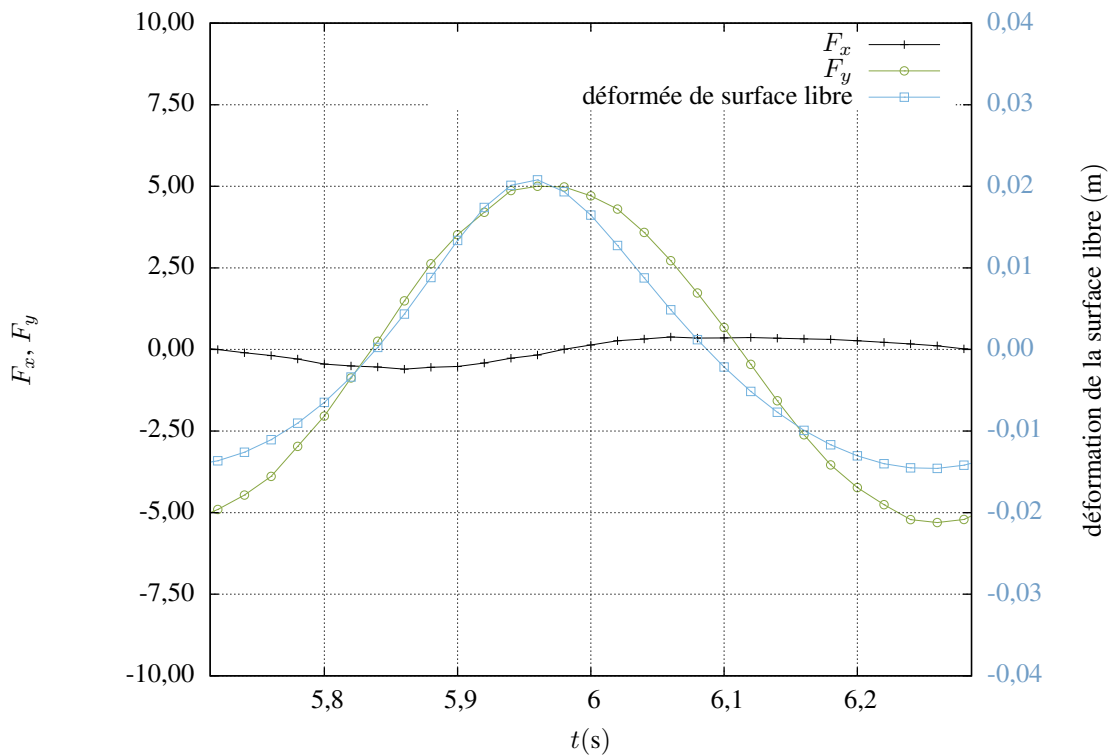
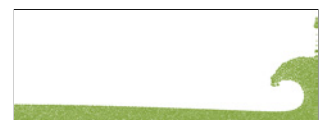


FIGURE 7.17 – Efforts horizontaux et verticaux sur la plaque en fonction du temps, fenêtre réduite à la 10^{ème} période de houle.

suivante :

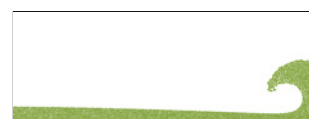
$$\mathbf{F} = \begin{pmatrix} F_x \\ F_y \end{pmatrix} = \frac{\sum_{i \in \Omega^+} \mathbf{f}_i m_i}{\frac{1}{2} \rho_0 B (w_{\max}^b)^2} \quad (7.8)$$

La déformation de surface libre en $x = P + B/2 = 1,2625$ (au dessus du centre géométrique de la plaque) a également été représentée. Les maxima d'efforts de portance sont synchrones au passage d'une crête, de même, les minima sont synchrones aux creux. La figure 7.17 représente ces courbes sur un intervalle de temps réduit à la 10^{ème} période de houle, confirmant cette corrélation entre passage d'une crête et effort maximal de portance. Les efforts de traînée sont d'un ordre de grandeur inférieur à ceux de portance et s'annulent au passage d'une crête ou d'un creux. Ces données vont être également confrontées aux résultats issus d'autres modèles, actuellement à l'étude.

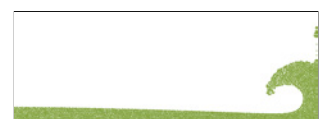


Troisième partie

Aspects informatiques



Le code de calcul, qui a permis de simuler les écoulements décrits dans les chapitres 4, 5, 6 et 7, va maintenant être présenté. Le chapitre 8 justifie le choix du langage de programmation et présente les particularités des algorithmes utilisés en méthode SPH. Le chapitre 9 est consacré à la parallélisation du code SPH, indispensable quand le nombre de particules augmente. Le choix du type de parallélisation est justifié et les performances sont indiquées.



Chapitre 8

Optimisation

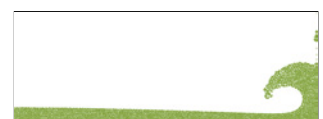
Un code de calcul a été créé pour la simulation d'écoulements à grandes échelles. Tous les écoulements précédents ont également été modélisés à l'aide de ce logiciel, écrit en *Fortran 90*. Le choix de ce langage a été justifié par les faits suivants :

- Le *Fortran 90* est encore très couramment utilisé pour le calcul scientifique. Il existe ainsi de nombreux compilateurs et bibliothèques performantes, généralement installés sur toutes les plateformes de calcul (calculateur du Centre de Ressources Informatiques de HAute-Normandie (CRIHAN), calculateur du laboratoire).
- Historiquement, tous les codes particuliers du laboratoire ont été programmés en *Fortran*, il était donc préférable d'écrire ce code de calcul dans un langage bien connu des autres membres de l'équipe.
- Ce langage est simple et relativement souple, notamment en raison des nouvelles normes introduites à partir de 1990. Ce qui rend la programmation rapide, si toutefois la base du code de calcul le permet.

Dans le chapitre présent, plusieurs *points-clés* de la programmation d'un tel code de calcul sont étudiés. Le développement logiciel a été réalisé autour des cas simples précédents, pour lesquels chaque amélioration du code n'entraînait pas de surcoût significatif en temps de calcul. Toutefois, en prévision des simulations à grandes échelles de temps et d'espace, un soin particulier a été apporté au maintien de performances optimales. En effet, un surcoût très léger, occasionné par une nouvelle fonctionnalité, dans le cas de la rupture de barrage, peut se révéler très significatif sur la durée de simulation de plusieurs périodes de houle en canal, par exemple.

8.1 Programmation objet ou tableaux ?

Le concept de particule, porteuse d'informations et interagissant avec ses voisines, conduit naturellement n'importe quel informaticien actuel à envisager une programmation par *objets* (voir figure 8.1). Avec une grande souplesse, ce concept d'objet permet de manipuler les particules comme des *entités* informatiques propres, munies de *méthodes* ou fonctions agissant sur elles ou



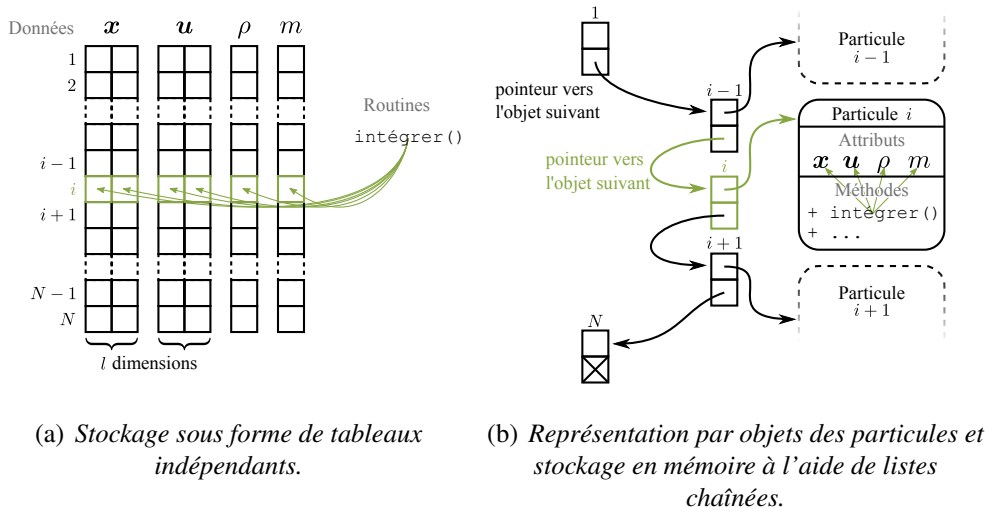


FIGURE 8.1 – Représentations des données en mémoire.

sur d'autres objets. Des langages informatiques permettent de modéliser des particules de cette manière, comme le langage *Java* dans lequel chaque algorithme est forcé à être intégré à un objet ou classe. D'autres langages permettent une *programmation orientée objet*, en combinant le concept d'objet à une vision plus classique de l'organisation d'un programme informatique (programme principal appelant des routines indépendantes). C'est le cas du langage *C++*. Le *Fortran* n'intégrait pas initialement toutes les notions du paradigme *objet*. Les normes 90, 95 et 2003 ont proposé successivement une extension de plus en plus orientée vers ce mode de programmation, mais les outils mis en place ne permettent pas de mettre en œuvre des objets aussi aisément qu'avec des langages tels que le *Java* ou le *C++*.

Au delà des possibilités proposées par chaque langage, et le *Fortran 90* ayant été choisi comme base de programmation, le choix restait ouvert sur l'emploi d'objets ou de simples tableaux pour le stockage des données associées aux particules. Ce choix a donc été réalisé en considérant deux critères :

- La manipulation des données en mémoire. Autrement dit, la facilité d'accès à une donnée précise, la suppression et la création de particules.
- Les performances associées à chaque approche quand un grand volume de données doit être employé.

Dans le cas d'un tableau, qu'il soit unique ou que des tableaux différents soient créés pour chaque grandeur physique, les données sont naturellement stockées de façon contiguë en mémoire. Cela présente l'avantage d'être adapté aux processeurs actuels qui fonctionnent par niveaux successifs de mémoire, appelés *caches* : toutes les données sont initialement stockées en mémoire vive, puis déplacées vers la mémoire cache du processeur quand celles-ci sont nécessaires à un calcul donné. Quand un chargement du cache est effectué, celui-ci est généralement rempli entièrement, avec les données de mémoire vive suivantes, même si celles-ci ne sont pas nécessaires. Par

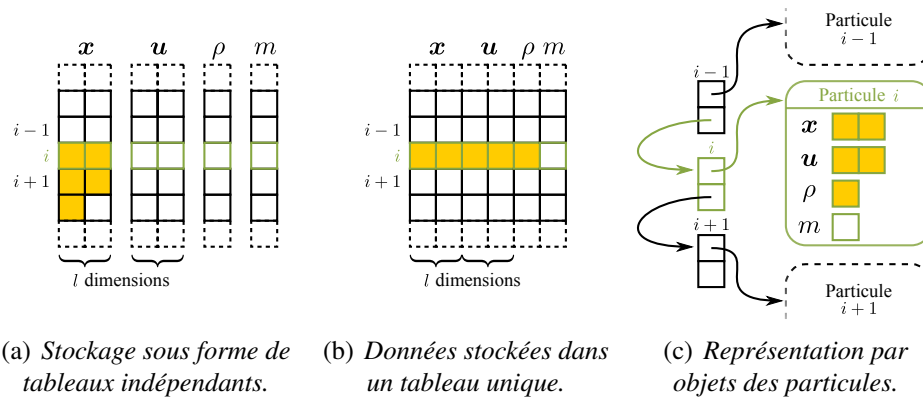


FIGURE 8.2 – Accès aux données en mémoire. Lors du chargement d’une nouvelle ligne de cache (5 cases ici, pour l’exemple), les données ne sont pas mises en cache dans le même ordre selon le choix de représentation (cases jaunes).

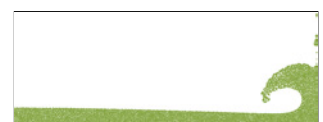
souci d’optimisation, un développeur va donc tenter d’exploiter totalement les données chargées en cache avant de déclencher un nouveau chargement.

Ainsi, quand une boucle est réalisée sur l’indice de position dans un tableau, le chargement de celui-ci n’est effectué que périodiquement (selon sa taille) car les données suivantes peuvent déjà y être présentes. En revanche, dans le cas d’une programmation par objet, comme pour un tableau unique (voir figure 8.2), l’ensemble des valeurs associées à une particule sont chargées en cache, même si on ne cherche pas à traiter ces données à cet instant. Bien sûr, il est possible d’optimiser l’utilisation du cache pour une représentation objet des particules, mais cela entraîne une vision complètement différente de l’algorithme, pas toujours en accord avec les équations modélisées.

L’utilisation de tableaux indépendants permet ainsi d’optimiser naturellement l’usage du cache, en réduisant les appels (ou *cache miss*). Cela permet également l’ajout de tableaux supplémentaires pour les évolutions futures du code de calcul, sans que les boucles itératives sur ces tableaux n’aient à être radicalement modifiées, comme par exemple :

- la longueur de lissage ϵ , si à l’avenir cette longueur est à faire varier pour chaque particule.
- un entier identifiant le *type* de particules pour des simulations multiphasiques.
- ...

Cela serait possible aussi dans le cadre d’une vision par objets du code, les appels de procédure ne seraient d’ailleurs pas modifiés, contrairement aux tableaux indépendants (une solution est toutefois proposée dans le chapitre suivant). En revanche, si un tableau unique est employé, l’ajout d’un tableau peut entraîner une modification de toutes les routines du code. On peut noter aussi que dans le cas d’un stockage par tableaux séparés ou par objets, le passage de simulations à deux dimensions (2D) à trois dimensions (3D) est plus aisé que dans le cas de tableaux séparés. Le tableau 8.1 résume, de manière simplifiée, les avantages et inconvénients de ces trois modèles de données, et confirme que l’usage de tableaux indépendants reste l’approche optimale pour l’écriture d’un code de calcul performant en méthode SPH.



	Tableaux séparés	Tableau unique	Objets
Déclaration (nombre de lignes de code)	✗ NON	✓ OUI	✗ NON
Souplesse (manipulation des données)	✓ OUI	✗ NON	✓ OUI
Performances (avec le langage Fortran 90)	✓ OUI	✗ NON	✗ NON

TABLE 8.1 – Avantages et inconvénients de chaque représentation des données.

Une combinaison des approches *objet* et *tableaux séparés* reste envisageable mais n’a pas motivé d’étude plus approfondie pour ces travaux de thèse.

L’utilisation d’objets n’aurait d’ailleurs d’intérêt que dans le cadre d’une vision *bio-inspirée* des phénomènes à étudier. On trouve d’ailleurs quelques travaux où des *systèmes multi-agents* (SMA) ont été mis en œuvre pour la modélisation d’écoulements : Schweitzer [202] propose un modèle de *particules actives* dont la dynamique repose sur le mouvement brownien. Servat [203] s’inspire de la méthode SPH pour la création d’un SMA adapté à l’étude des phénomènes de ruissellement, d’infiltration et d’érosion.

Ainsi, un tableau différent est déclaré pour chaque grandeur physique transportée par les particules. Un soin particulier doit être apporté au parcours des tableaux à deux dimensions, utilisés pour le stockage des vecteurs (\mathbf{x} , \mathbf{u} , $\frac{D\mathbf{u}}{Dt}$...). En effet, en Fortran et contrairement au langage C, les tableaux sont stockés en mémoire par colonnes. Quand un tableau à deux dimensions doit être parcouru, un accès par colonnes (code 8.1 (a)) permet de réduire le nombre de *cache miss* par rapport à un accès par lignes (code 8.1 (b)).

<pre> INTEGER :: I, D, ND, NP REAL :: DT REAL, DIMENSION (1:ND, 1:NP) :: X, U DO I=1, NP DO D=1, ND X(D, I) = X(D, I) + U(D, I) * DT ENDDO ENDDO </pre>	<pre> INTEGER :: I, D, ND, NP REAL :: DT REAL, DIMENSION (1:ND, 1:NP) :: X, U DO D=1, ND DO I=1, NP X(D, I) = X(D, I) + U(D, I) * DT ENDDO ENDDO </pre>
---	---

(a) Accès par colonnes

(b) Accès par lignes

CODE 8.1 – Parcours de tableaux à deux dimensions en Fortran.

8.2 Recherche des interactions

La forme des opérateurs utilisés en méthode SPH et la condition de compacité du noyau (équation (2.35)) impliquent que deux particules ne peuvent interagir que si la distance qui les sépare est inférieure au rayon du noyau, 3ϵ dans le cas présent (équation (2.37)). Ainsi, quand les dérivées temporelles sont intégrées, il est possible d'éviter un parcours de toutes les particules, simple du point de vue algorithmique mais très coûteux en temps de calcul (Code 8.2). En effet, cette recherche *directe* requiert un nombre d'opérations de l'ordre de $\mathcal{O}(N^2)$ avec N , le nombre de particules.

```
REAL :: R
INTEGER :: I, J, NP, ND
REAL, DIMENSION (1:ND, 1:NP) :: X
DO I=1, NP
  DO J=1, NP
    R=DISTANCE(X, I, J, ND)
    IF (R<=3*EPSILON) THEN

        !CALCUL DES INTERACTIONS

    ENDIF
  ENDDO
ENDDO
```

CODE 8.2 – Recherche directe des interactions.

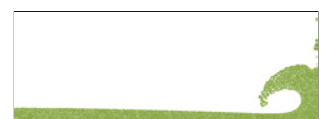
Deux algorithmes courants ont été développés pour détecter, avant toute intégration, les paires de particules en interaction :

- La recherche par arborescence ou *Tree Search*.
- La recherche par maillage sous-jacent ou *Linked List*.

Le principe, commun aux deux algorithmes, est d'éviter de calculer la distance entre chaque couple de particules, en éliminant progressivement chaque couple ne pouvant interagir, à l'aide de tests moins coûteux en temps. Ces techniques ne sont pas apparues avec la méthode SPH, mais ont plutôt été développées dans le cadre de la recherche en dynamique moléculaire. Leur utilisation en méthode particulière, pour la recherche de voisins, a donc été naturelle.

8.2.1 Algorithme de recherche *Tree Search*

L'idée consistant à employer une arborescence pour la recherche *globale* de toutes les interactions dans un nuage de points est courante en informatique. La notion même d'arborescence est très couramment abordée en programmation orientée objet. Tous les algorithmes *Treecode* [9] s'en inspirent pour l'intégration d'interactions sur l'ensemble du domaine de calcul. Pour les interac-



tions les plus lointaines, il permet de ne prendre en compte qu'une approximation de l'influence des particules au lieu de calculer, de façon précise, ces interactions peu significatives [131, 184]. L'approche rencontrée en méthode SPH est plus simple, car le noyau employé est le plus souvent à support compact (équation (2.35)). L'absence d'interactions lointaines réduit l'usage de l'arborescence à une recherche des particules proches. C'est Hernquist et Katz [90] qui ont introduit cette méthode, qu'ils nomment *TreeSPH* et pour laquelle chaque particule a une longueur de lissage ϵ distincte. Le *Tree Search* permet de détecter de façon optimale les interactions dans ce cas précis d'un schéma ϵ -variable. Il est évidemment applicable à un schéma où ϵ est constant, mais comme nous le verrons dans la suite, l'algorithme *Tree Search* se montre moins performant que l'alternative *Linked List*.

Deux parties sont à distinguer pour la détection des interactions par la méthode *Tree Search* :

- construction d'un arbre représentant la répartition des particules,
- parcours de l'arbre pour repérer les particules voisines.

Construction de l'arbre

L'arbre est construit sur la base d'une structure décrite dans l'extrait de code 8.3 (on se place donc dans le cadre de la programmation orientée objet). La manipulation de ce type de structure de données est, comme nous l'avons déjà mentionné, possible en *Fortran 90* bien que peu aisée. Une première cellule (ou *racine* de l'arbre) est créée, de forme carrée, de manière à contenir l'ensemble des particules. A chaque étape de la création de l'arbre (effectuée récursivement), la cellule considérée est découpée en au plus 2^l cellules-filles (avec l , le nombre de dimensions physiques) (figure 8.3). Les cellules de l'arbre sont choisies de forme carrée car l'utilisation de cellules au rapport d'aspect variable complexifie l'algorithme. De plus, des cellules de dimensions variables ne sont pas optimales pour la seconde étape de détection des interactions. La découpe récursive est réalisée tant que les feuilles de l'arbre (dernières cellules créées) contiennent plus de $NMAX$ particules. Le choix d'un test d'arrêt basé sur le nombre de particules a été motivé pour l'utilisation potentielle de cet algorithme au sein d'un schéma ϵ -variable. Un test basé sur la taille de la cellule (dont le seuil serait par exemple 3ϵ) paraît mieux approprié à un schéma ϵ -constant, cependant l'algorithme *Linked List* reste optimal pour ce cas précis.

Le choix d'une valeur pour le nombre $NMAX$ de particules contenues dans une feuille de l'arbre a été étudié par Croquet [49] au cours de son stage *ingénieur* au LOMC. Les temps de recherche de toutes les interactions ont été mesurés pour différentes valeurs de $NMAX$ (sur une courte simulation de canal à houle). Les résultats obtenus ont été reportés dans le tableau 8.2 et montrent qu'un choix judicieux serait $NMAX \approx 250$. Ce résultat n'est pas très précis mais il pourra être ajusté s'il s'avérait que le *Tree search* soit plus rapide que le *Linked List*.

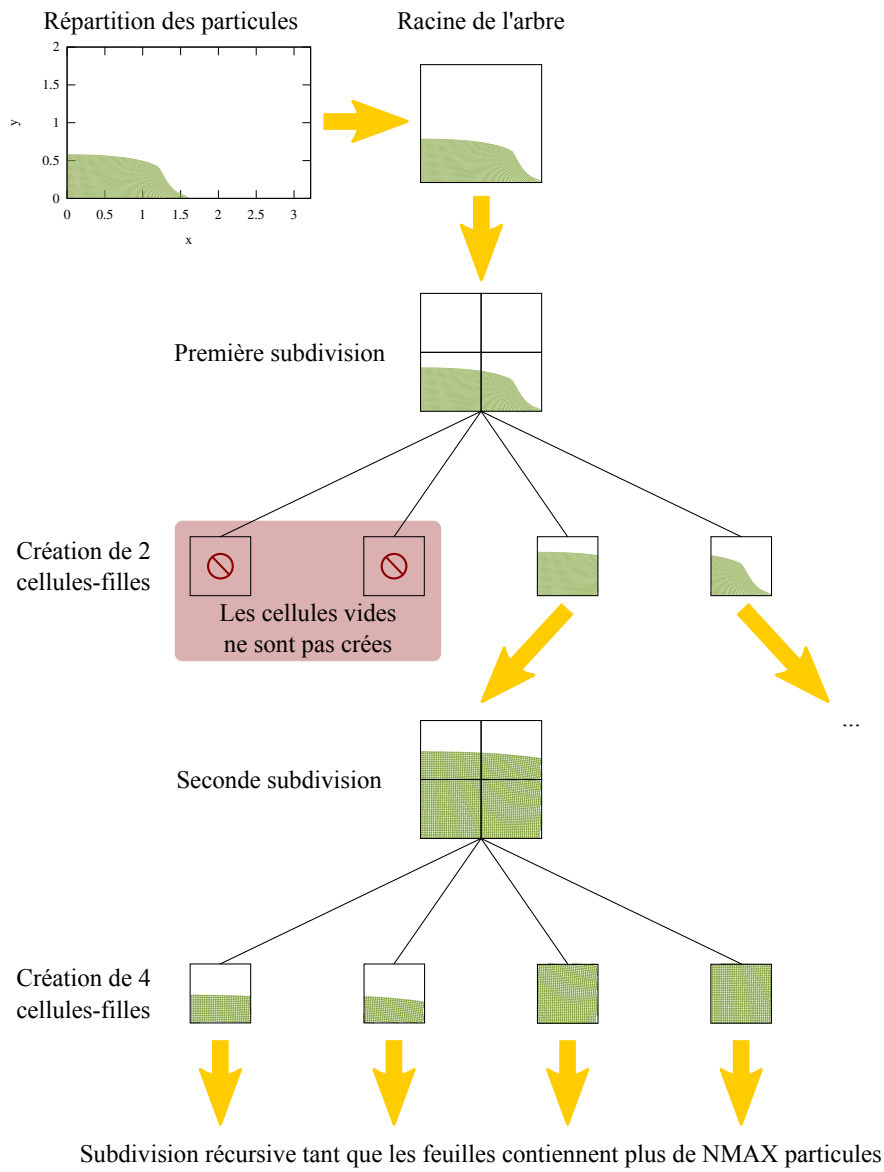
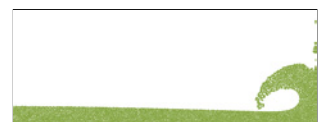


FIGURE 8.3 – Exemple de construction de l'arborescence en 2D.



```

MODULE MOD_TREE

TYPE PCELLULE
  TYPE (CELLULE), POINTER :: CELL
END TYPE PCELLULE

TYPE CELLULE
!NOMBRE DE FILLES
  INTEGER :: NBFILLES
!NOMBRE DE PARTICULES
  INTEGER :: NBPART
!COORDONNEES DE LA CELLULE
  REAL, DIMENSION (:), ALLOCATABLE :: BMIN, BMAX
!INDICES DANS LES TABLEAUX DES PARTICULES CONTENUES
  INTEGER, DIMENSION (:), ALLOCATABLE :: PARTIN
!POINTEURS VERS LES CELLULES-FILLES
  TYPE (PCELLULE), DIMENSION (:), ALLOCATABLE :: FILLES
END TYPE CELLULE

END MODULE MOD_TREE

```

CODE 8.3 – Structure Fortran 90 pour la mise en œuvre de l’algorithme Tree Search.

		NMAX					
		10	50	100	250	500	750
Nombre de particules	24000	71,80	27,57	23,40	20,28	31,56	31,06
	54000	556,16	140,10	65,98	61,73	61,28	60,85
	96000	962,90	278,15	126,66	126,26	140,91	140,85

TABLE 8.2 – Temps moyen de recherche des interactions par l’algorithme Tree Search en fonction la valeur du paramètre NMAX et du nombre de particules.

Recherche des interactions

Cette étape de l’algorithme *Tree Search* consiste à parcourir l’arbre précédemment créé afin de construire une liste de toutes les paires de particules en interactions. Ainsi, pour chaque particule i , un parcours récursif de l’arborescence est réalisé en appliquant les règles suivantes :

- Si la distance de la particule i à la cellule considérée est inférieure ou égale à 3ϵ , on explore les cellules filles.
- Si la cellule considérée est une feuille de l’arbre, alors on calcule pour chaque particule j de

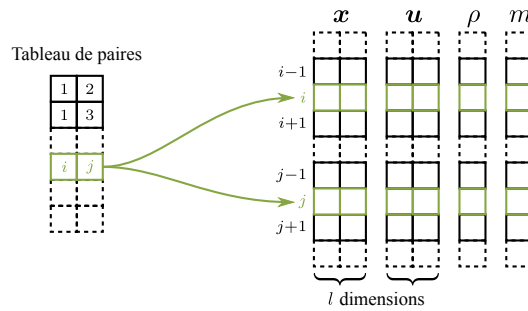


FIGURE 8.4 – Stockage des paires de particules en interaction.

la cellule la distance r_{ij} en ne testant que les particules dont l'indice $j > i$.

Cette dernière règle exploite le principe de réciprocité des actions et évite de stocker des doublons dans le tableau de paires de particules ainsi construit (figure 8.4).

Ainsi, les couples de particules en interaction connus, il suffit d'effectuer une boucle sur le tableau de paires pour intégrer les équations (code 8.4).

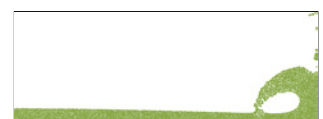
```

REAL :: FIJ
INTEGER :: I, J, K, NPAIR
REAL, DIMENSION(1:NP) :: DRHO_DT
DO K=1, NPAIR
  I=PAIR(1, K)
  J=PAIR(2, K)
  !CALCUL DU TERME EN SECOND MEMBRE DE
  !L'EQUATION DE CONSERVATION DE LA MASSE
  !RELATIF A CETTE PAIRE DE PARTICULES
  FIJ=CONSERVATION_DE_LA_MASSE(I, J, ...)
  !APPLICATION AUX DEUX PARTICULES CONCERNEES
  DRHO_DT(I)=DRHO_DT(I)-FIJ
  DRHO_DT(J)=DRHO_DT(J)-FIJ
ENDDO

```

CODE 8.4 – Intégration de l'équation de conservation de la masse par déroulement du tableau de paires de particules en interaction.

On peut noter que cette façon de stocker en mémoire les couples de particules fait appel à la notion de *double-indices*, souvent rejetée et à laquelle la notion de pointeur est préférée. Mais encore une fois, ce concept de pointeur n'étant pas *natif* au Fortran 90, il n'a pas été observé de gain de temps significatif quand ce double-indices est remplacé par l'usage de pointeurs. En outre, cette façon d'intégrer les équations garantit la conservation numérique en évitant l'accumulation d'écarts entre les deux approximations séparées de l'influence de la particule i sur la particule j , puis de la particule j sur la particule i .



Un soin particulier a ainsi été apporté à la programmation de cet algorithme *Tree Search*, justifié par la part prépondérante qu’occupe la détection des interactions dans le temps total d’exécution d’une simulation (de l’ordre de 60%). Si un schéma ϵ -variable devait être employé, en vue de réduire le nombre de particules et la précision dans certaines zones moins intéressantes du domaine de calcul, cette méthode serait adaptée. De plus, ces schémas posent la question du critère à retenir pour le pas d’interpolation (ϵ_i, ϵ_j ou $\frac{\epsilon_i + \epsilon_j}{2}$ [172, 90]), chacune de ces interprétations pourrait aisément être intégrée au code, puis testée, ultérieurement.

8.2.2 Algorithme de recherche *Linked List*

Cette méthode de recherche rapide a été adaptée à SPH par Monaghan [153]. L’algorithme *Linked List* est très couramment employé au sein des modèles SPH appliqués à l’hydrodynamique. Comme pour le *Tree Search*, l’algorithme s’appuie sur une structure de données spécifique (code 8.5). Il se déroule également en deux étapes (figure 8.5) :

- Calcul des coordonnées des particules dans une grille cartésienne et construction pour chaque cellule d’une liste des particules contenues.
- Détection des paires de particules en interaction.

```

MODULE MOD_LIST

TYPE LIST
!NOMBRE DE PARTICULES DANS LA CELLULE
  INTEGER :: NBPART
!INDICES DE CES PARTICULES DANS LES TABLEAUX
  INTEGER, DIMENSION(:), ALLOCATABLE :: PARTIN
END TYPE LIST

END MODULE MOD_LIST

```

CODE 8.5 – Structure Fortran 90 pour la mise en œuvre de l’algorithme *Linked List*.

Mise en place des listes

L’idée de cet algorithme est de sélectionner plus rapidement que par simple recherche directe, toutes les particules potentiellement en interaction avec une particule donnée. Ainsi, la position de chaque particule est repérée par ses coordonnées entières dans une grille cartésienne de pas 3ϵ . Ce calcul se fait assez rapidement de la manière suivante :

$$\mathbf{X}_i = \left\lceil \frac{\mathbf{x}_i - \mathbf{x}_O}{3\epsilon} \right\rceil, \quad (8.1)$$

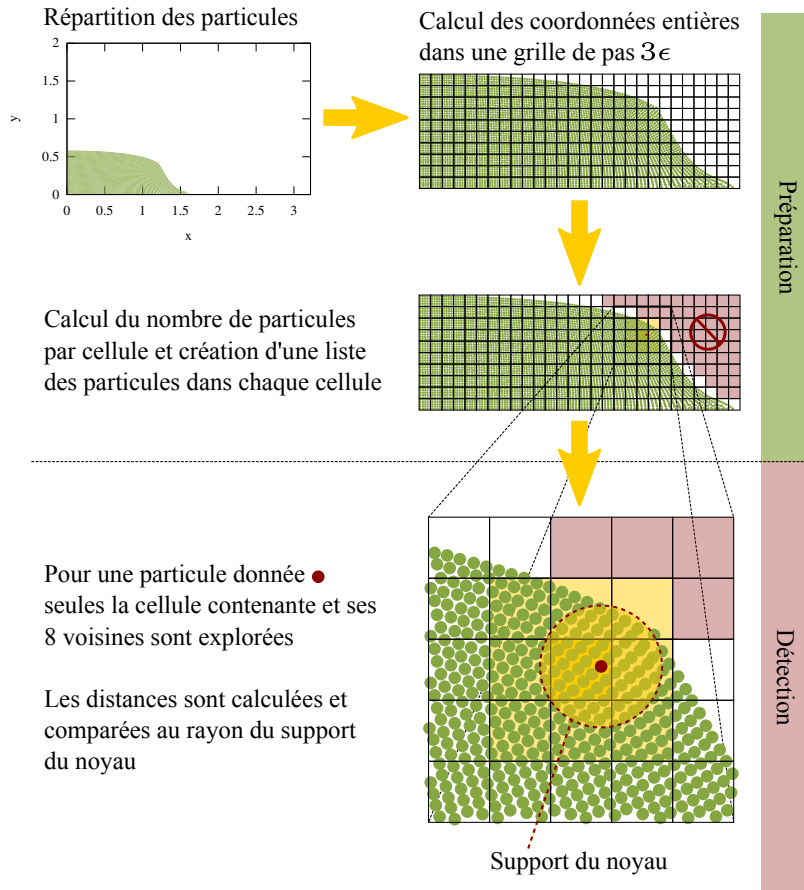
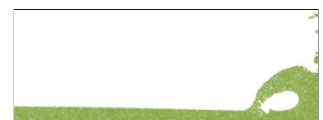


FIGURE 8.5 – Exemple de mise en œuvre de l’algorithme *Linked List* en 2D.

où X_i désigne la position de la particule i dans la grille d’origine x_0 . $[\cdot]$ désigne la partie entière par excès, qui s’obtient aisément en *Fortran* par appel de la fonction intrinsèque `ceiling()`. Un seul passage sur toutes les particules suffit donc à déterminer leurs positions dans la grille. Cela représente déjà une diminution de la complexité algorithmique par rapport au *Tree Search*, pour lequel plusieurs tests sont réalisés sur chaque particule à chaque niveau de développement de l’arbre.

Au cours de cette boucle sur les particules, un compteur est incrémenté sur chaque case du maillage cartésien afin de connaître le nombre final de particules par maille. Ainsi, par souci de gain de place en mémoire, une liste des particules présentes (structure décrite dans l’extrait 8.5) est allouée pour chaque maille, uniquement si celle-ci contient des particules. En parcourant à nouveau les particules, leur indice dans les tableaux est ajouté à la case correspondante. Sur ce point, on note une nouvelle fois l’usage de *double indiciage* ; une étude plus approfondie des outils de pointage direct en mémoire serait utile, mais nécessiterait de profondes modifications du code de calcul.



Détection des interactions

La position de chaque particule étant connue, il suffit alors de parcourir les listes de particules contenues dans les 9 cases voisines (en 2D, 27 en 3D) et de calculer chaque distance par rapport à la particule considérée. Encore une fois, seules les particules dont l'indice est supérieur à celui de la particule considérée sont testées afin d'empêcher le stockage de doublons. La mise en œuvre de cet algorithme se montre donc assez simple, la complexité algorithmique correspondante est $\mathcal{O}(N)$, donc plus faible que l'algorithme *Tree Search* (en $\mathcal{O}(N \log(N))$, avec N le nombre de particules). Ce résultat a d'ailleurs été vérifié en mesurant le temps consacré à la détection des interactions sur une courte simulation. Le tableau 8.3 et la figure 8.6 montrent les résultats obtenus. On remarque que l'algorithme *Linked List* est bien plus efficace que l'algorithme *Tree Search*. Ainsi, ce mode de recherche a été adopté pour toutes les simulations présentées dans ce manuscrit.

		<i>Tree Search</i>	<i>Linked List</i>
Nombre de particules	24000	20,28	11,50
	54000	61,73	26,02
	96000	126,26	46,85

TABLE 8.3 – Comparaison du temps de détection des interactions en fonction de la méthode de recherche et du nombre de particules.

8.2.3 Accélération par la méthode de Verlet

Une autre technique courante est souvent combinée aux deux algorithmes précédents. La méthode, dite de Verlet, consiste à n'effectuer une recherche de toutes ces interactions que tous les N_v pas de temps, au lieu de la réaliser à tous les pas de temps (et sous-itérations pour le schéma de Runge-Kutta). Afin de correctement prendre en compte toutes les interactions et de ne pas omettre celles qui pourraient apparaître entre deux recherches consécutives, le rayon de recherche de chaque particule r_s , égal à 3ϵ jusqu'à présent, doit être augmenté :

$$r_s = 3\epsilon + r_v, \quad (8.2)$$

où r_v correspond au déplacement maximale que peut avoir une particule au cours des N_v itérations qui séparent deux recherches d'interactions :

$$r_v = N_v |\mathbf{u}_{\max}| \Delta t \quad (8.3)$$

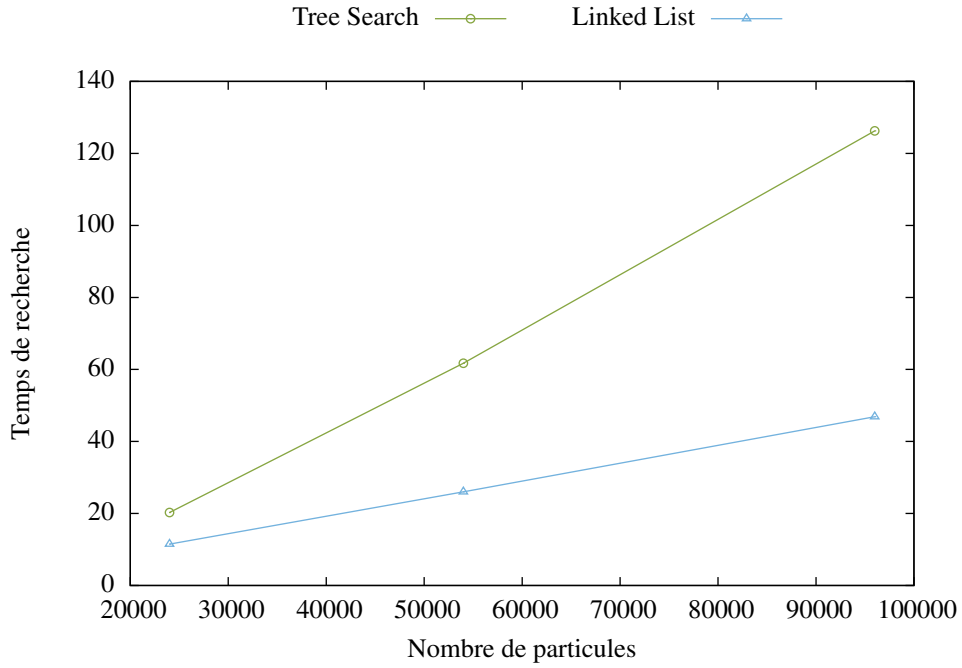


FIGURE 8.6 – Comparaison du temps de détection des interactions en fonction de la méthode de recherche et du nombre de particules.

La vitesse \mathbf{u}_{\max} maximale pouvant être observée dans la simulation est employée afin de s’assurer qu’aucune nouvelle interaction n’est omise. La figure 8.7 illustre l’importance de cette augmentation du rayon de recherche. Or, dans l’approche faiblement compressible adoptée ici, la vitesse maximale et la vitesse du son sont liées par la relation suivante :

$$c_0 = 10|\mathbf{u}_{\max}| \quad (8.4)$$

et si la condition Courant-Friedrichs-Lewy (CFL) [47] est respectée pour le pas temps, on a en pratique :

$$\Delta t \leq \frac{\epsilon}{c_0} \quad (8.5)$$

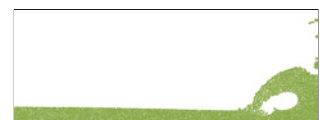
Ainsi, l’équation (8.3) devient :

$$r_v = N_v \frac{\epsilon}{10} \quad (8.6)$$

et, de l’équation (8.2), on déduit, le nombre maximal d’itérations que l’on peut réaliser, sans risque, pour une valeur donnée du nouveau rayon r_s :

$$N_v = 30 \left(\frac{r_s}{3\epsilon} - 1 \right) = 30 (\alpha_v - 1) \quad (8.7)$$

où α_v est le rapport entre le rayon r_s et le rayon du noyau 3ϵ . En effectuant une recherche avec, par exemple l’algorithme *Linked List* et un maillage de pas r_s , l’ensemble des paires de particules à prendre en compte à $t + N_v \Delta t$ sont forcément incluses dans l’ensemble des interactions détectées.



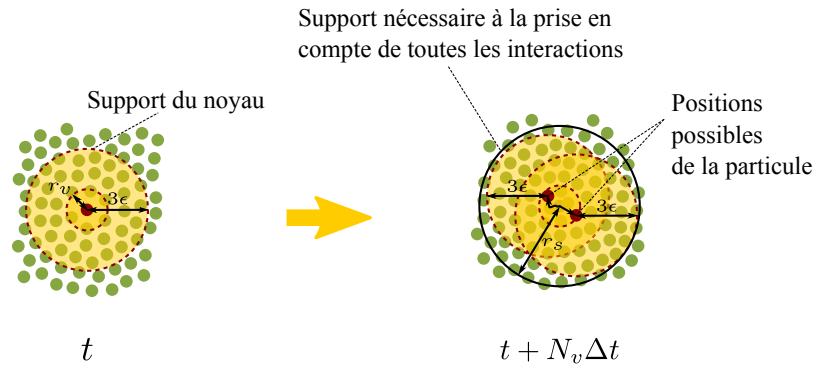


FIGURE 8.7 – Définition du nouveau rayon de recherche des interactions pour la méthode de Verlet.

Cet algorithme dispense donc de construire un nouveau maillage ou une nouvelle arborescence à chaque pas de temps. En revanche, les valeurs du noyau et de son gradient doivent être recalculées systématiquement. Le tableau 8.4 montre le rapport entre le temps d'exécution avec Verlet (pour différentes valeurs de α_v) et le temps d'exécution sans Verlet ($\alpha_v = 1$). Le meilleur résultat a été obtenu pour $\alpha_v = 1,10$ (soit une recherche des interactions tous les 3 pas de temps) avec une réduction du temps d'exécution du code de 5%.

	α			
Nombre de particules	1,00	1,10	1,20	1,30
54000	1,00	0,95	1,04	1,28
216000	1,00	0,95	1,04	1,28
N_v	0	3	6	9

TABLE 8.4 – Comparaison

Pour cet essai, un schéma d'intégration de type *Runge-Kutta 4* a été employé. Ainsi, avec la méthode de Verlet et une recherche complète tous les 3 pas de temps, on réalise l'économie de 12 détections des interactions (1 par sous-itération). Le gain obtenu paraît donc très inférieur à ce qui pouvait être attendu. Cela peut s'expliquer par le fait que ces 12 détections ne sont pas simplement supprimées car les calculs du noyau et de son gradient pour chaque paire de particules sont maintenus. Seule la phase de préparation disparaît (figure 8.5), qui pourtant n'occupe que 50% du temps consacré à la détection. Une autre raison est aussi l'augmentation rapide du nombre d'interactions stockées en mémoire quand on agrandit la zone de recherche. Cet algorithme nécessite deux tableaux de paires d'entiers comme celui de la figure 8.4 : un pour le stockage de toutes les

paires *potentiellement* en interaction selon la recherche complète réalisée tous les N_v pas de temps, qui est ensuite parcouru pour la construction du tableau de paires *effectives* à chaque mise à jour des positions. Cette augmentation de la mémoire nécessaire n'explique qu'en partie les faibles gains de temps observés. Au regard des grandes modifications du code nécessaires, des faibles gains de temps obtenus et de la mémoire supplémentaire requise (qui pourrait rapidement devenir restrictive sur de grandes applications), la méthode de Verlet n'a pas été maintenue au sein du code ; privilégiant ainsi robustesse et efficacité à la faible amélioration des performances.

8.3 Autres améliorations et perspectives

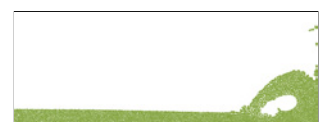
L'idée de la méthode de Verlet reste cependant très intéressante, et dans le cadre de la programmation par objets, une approche plus radicale pourrait être tentée : le graphe d'interactions serait créé une seule fois à l'initialisation du code, puis mis à jour dynamiquement au cours de la simulation. Cette mise à jour éviterait la réinitialisation complète de la liste des paires de particules à chaque appel de la méthode de recherche des interactions. Aucune technique ne permet actuellement de réaliser une mise à jour dynamique de cette liste assez rapidement pour concurrencer la méthode employée ici et les espaces en mémoire requis restent prohibitifs. Mais cet axe de travail ne doit cependant pas être abandonné trop vite car il est concevable qu'une méthode relativement simple puisse être mise en œuvre pour ajouter et retirer des interactions de la liste à la volée sans la recalculer complètement. Ce point est à l'étude.

Colagrossi [35] suggère une autre amélioration qui consisterait à construire des sous-cellules dans chaque maille de l'algorithme *Linked List*, de manière à éviter encore quelques tests inutiles de distance avec les particules des sous-cellules sans intersection avec le support de noyau de la particule considérée. En théorie, cette technique, dite *Linked List with Subcells*, combinée à la méthode de Verlet, semble intéressante mais elle n'a pas encore été testée.

Lorsque des particules fantômes sont employées pour modéliser des conditions aux limites, les interactions entre ces particules et les particules réelles doivent également être détectées. Initialement, lorsque la routine de recherche était appelée, toutes les interactions sur l'ensemble du domaine (particules réelles et fantômes) étaient comptabilisées. Ainsi le tableau de paires de particules pouvait contenir au final un certain nombre de couples fantôme-fantôme sans utilité puisqu'on ne calcule pas de dérivée temporelle pour ces particules virtuelles. Une réduction significative de la taille de cette liste d'interaction a été obtenue en ne cherchant d'abord que les paires de particules réelles, puis quand celles-ci sont proches de la paroi, les éventuelles interactions avec les particules virtuelles associées (figure 8.8).

Nous l'avons déjà mentionné auparavant, mais l'utilisation d'un schéma ϵ -variable pourrait permettre d'accélérer significativement les simulations à grand nombre de particules, pour lesquelles seules certaines zones du domaine de calcul nécessiteraient une précision accrue.

Du point de vue optimisation scalaire, un soin particulier a été apporté à l'écriture de chaque boucle



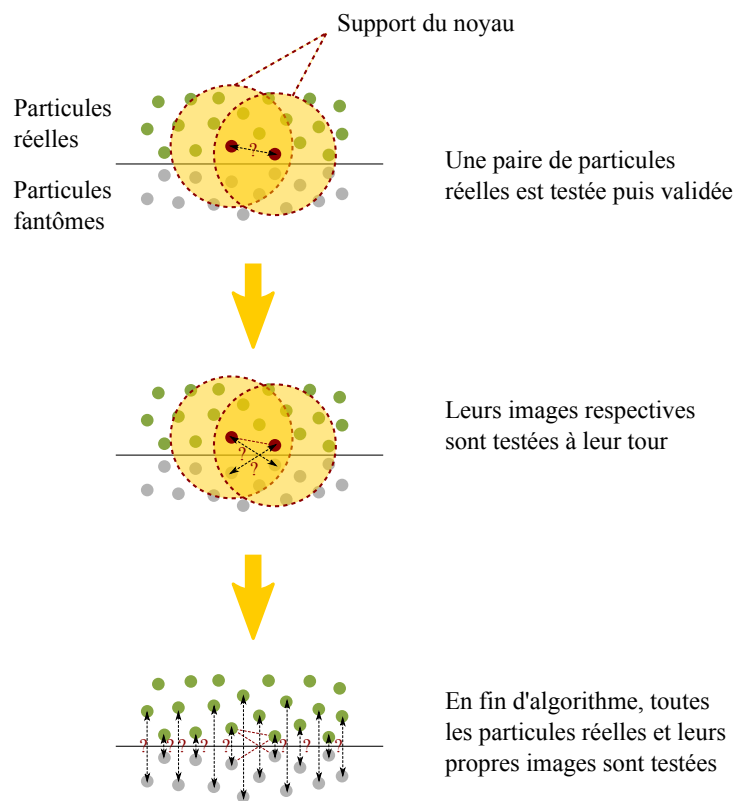


FIGURE 8.8 – *Prise en compte de particules virtuelles lors de la détection des interactions.*

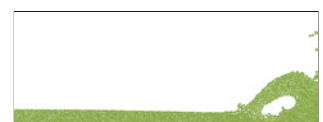
du code de manière à exploiter, de la façon la plus efficace possible, la représentation en mémoire des particules et les différents niveaux de cache (les options de compilateur permettent très souvent de mettre en œuvre ces optimisations (déroulement des boucles. . .) sans modification du code.

Une discipline émergente en calcul scientifique profite de la montée en puissance actuelle des processeurs de cartes graphiques (GPU). Ces processeurs disposent d'une architecture très différente des processeurs de calcul (CPU), conçue pour privilégier la rapidité des opérations en virgule flottante. Comme le montre la figure 8.9, les performances atteintes par les GPU de ces dernières années sont très largement supérieures aux capacités des processeurs CPU classiques (la chaleur dégagée et la consommation électrique de ces GPU sont, elles aussi, très largement supérieures).

Ainsi, depuis le début des années 2000, de nombreux numériciens s'intéressent à de nouvelles plates-formes de calcul hybrides combinant GPU et CPU (citons, par exemple, les machines **Tesla** [250]). Les outils logiciels ont considérablement évolué. La norme **OpenCL** [251] propose un langage unifié, en complément du C++, permettant de programmer de façon transparente un code, pouvant s'exécuter à la fois sur GPU ou CPU. Différemment, les compilateurs commencent à intégrer la possibilité de porter le code vers le GPU, c'est le cas du compilateur (payant) **PGI** [252] avec lequel il est maintenant possible de spécifier par un mot-clé au sein du code de calcul, quelles parties doivent être exécutées sur le GPU. Le développement dans ce domaine est très actif et l'utilisation de ces GPU devient de plus en plus simple. Une extension du code à ce type de processeur pourrait être envisageable. Cependant quelques réserves peuvent être émises.

D'une part, d'importantes modifications du code SPH seraient nécessaires pour transposer les parties les plus lourdes du calcul sur GPU. La représentation en mémoire étant totalement différente d'un CPU, chaque déclaration est à modifier et un tri systématique des tableaux est requis afin d'éviter tout *cache miss*, qui serait fatal à l'exécution sur le GPU. L'utilisation d'un tableau de paires de particules (figure 8.4) devient inefficace sur GPU, imposant l'utilisation de listes complètes de particules proches pour chaque particule. Cela a pour effet de séparer le calcul de l'action de la particule i sur la particule j de celui de la particule j sur la particule i . Une conséquence immédiate est l'abandon de la symétrie *informatique* des équations. Deux évaluations séparées de la même quantité conduisent à deux accumulations différentes d'erreurs de troncature. Finalement, la comparaison de résultats obtenus sur GPU ou CPU peut montrer des écarts significatifs entre les deux solutions.

D'autre part, au delà des considérations de performance et précision, ces GPU ont une consommation électrique et un dégagement de chaleur proportionnels aux performances obtenues. Le prix attractif n'inclut donc pas le surcoût occasionné par leur utilisation.



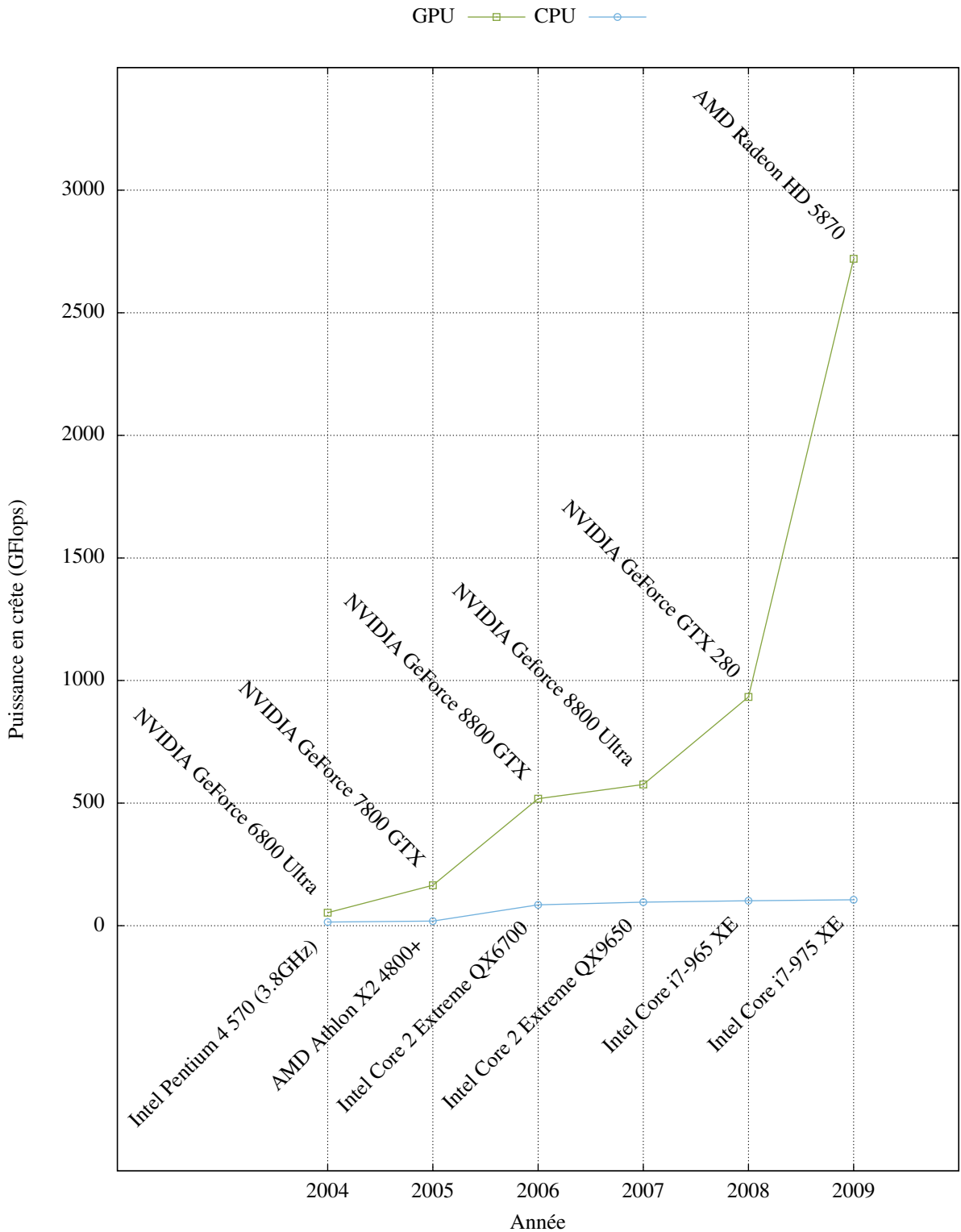


FIGURE 8.9 – Comparaisons des puissances maximales de processeurs GPU et CPU entre 2004 et 2009.

Chapitre 9

Parallélisation

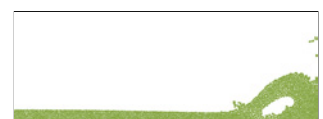
Au delà des différentes optimisations dont peut faire l'objet d'un code de calcul, la parallélisation est un moyen efficace de réduire les temps de simulations en divisant l'ensemble des opérations sur un nombre fini de processeurs. La charge de calcul peut être répartie sur les processeurs selon plusieurs modèles : Single Process Multiple Data (SPMD), SIMD. ... Il ne s'agit pas de passer en revue ici toutes les méthodes de parallélisation d'un code, tant celles-ci sont variées. L'approche employée dépend d'une part du code de calcul lui-même et des applications visées, d'autre part de la plate-forme de calcul. Ainsi, si le code doit être exécuté sur une machine massivement parallèle, un cluster de calcul ou une grille de calcul, l'approche choisie ne sera pas la même.

Dans le cas présent, les moyens informatique mis à notre disposition sont (par ordre de puissance) :

- Une station de travail comprenant un processeur Intel Xeon (à quatre cœurs et deux unités logiques par cœur, soit la possibilité de lancer jusqu'à huit processus), pour le développement du code, son débogage et le lancement de petites simulations.
- Un cluster de calcul au laboratoire, comptant sept nœuds de deux processeurs Intel Xeon (à quatre cœurs mais une seule unité logique par cœur), autorisant le lancement de 56 processus.
- Le calculateur du CRIHAN, regroupant 22 nœuds de calcul avec huit processeurs Power5 par nœud.

Chaque nœud dispose d'une quantité de mémoire commune aux processeurs du nœud. Ainsi, il aurait été possible d'exploiter les options d'auto-parallélisation (OpenMP) du compilateur, qui auraient simplement segmenté chaque boucle itérative du code en autant de boucles que de processeurs. Bien qu'efficace, ce type d'auto-parallélisation ne prend pas en charge la répartition de la mémoire et limite l'exécution du code à un seul nœud de la plate-forme employée.

Nous avons donc choisi de transformer explicitement le code par ajout d'instructions du paradigme Message Passing Interface (MPI), de manière à pouvoir exécuter le code sur une grande variété de machines de calcul (stage ingénieur de Remi Croquet [49]).



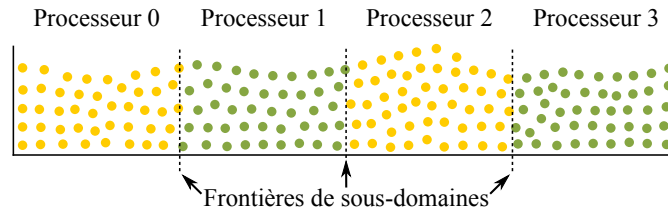


FIGURE 9.1 – Décomposition du domaine en bandes verticales sur 4 processeurs.

9.1 Décomposition de domaine

L'approche adoptée ici consiste à diviser le domaine de calcul en autant de sous-domaines qu'il y a de processeurs. Cela offre plusieurs avantages :

- La propriété de compacité du noyau (2.35) nous assure que les seules informations à transmettre d'un processeur à l'autre se situent toutes dans une zone de largeur 3ϵ le long de chaque interface entre processeur.
- Chaque processeur ne se charge que d'un seul sous-domaine, le nombre de particules et l'espace mémoire à allouer sur chaque unité de calcul sont donc divisés par le nombre de processeurs. Cela permet d'envisager une augmentation globale du nombre de particules (sous réserve que le code soit parallélisé dans son intégralité).

Dans un premier temps, il a été imaginé que ce découpage de sous-domaines soit le plus arbitraire possible, de façon à pouvoir simuler le plus grand nombre de problèmes sans se préoccuper de la géométrie du domaine. Nous nous sommes vite rendus compte que cette approche nécessitait de chaque processeur qu'il place d'abord la totalité des particules avant d'éliminer celles qui ne le concernaient pas. Bien que très générale, cette procédure a été abandonnée car elle entraînait directement en contradiction avec notre remarque précédente. C'est pourtant l'approche employée par Oger [172] qui ne précise pas comment se déroule l'initialisation de chaque simulation. De la même manière, Ferrari *et al.* [75] utilisent la librairie Metis [107] pour le partitionnement de graphes qui lui permet de ne pas gérer explicitement chaque transfert de particules pour l'équilibrage de la charge, mais aucune information n'est donnée sur l'initialisation des simulations. Par modification de l'algorithme *Tree Search*, Davé *et al.* [53] exploite l'arborescence pour la détection des zones d'interactions entre ses sous-domaines (également défini par découpage récursif du domaine global) ; l'arborescence est alors exploitée tout au long de l'algorithme et pas uniquement au moment de la recherche des paires en interaction, mais encore une fois, aucune précision n'est apportée quant à la répartition initiale des particules. La bibliographie est donc assez pauvre sur la parallélisation de la méthode SPH, en particulier appliquée aux écoulements à surface libre. Maruzewski *et al.* [146] décrivent l'exécution d'un code SPH à très grand nombre de particules (de l'ordre de 10^8) sur une machine massivement parallèle (IBM Blue Gene/L avec 8192 processeurs). Les difficultés liées à la gestion de la mémoire de telles machines sont décrites.

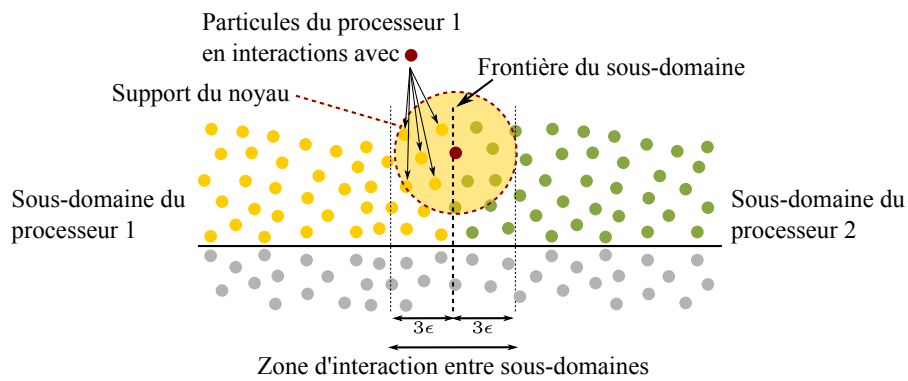


FIGURE 9.2 – Zone d'interaction entre deux sous-domaines : les informations concernant les particules jaunes de cette zone sont nécessaires au processeur 2 pour l'intégration de la particule ●.

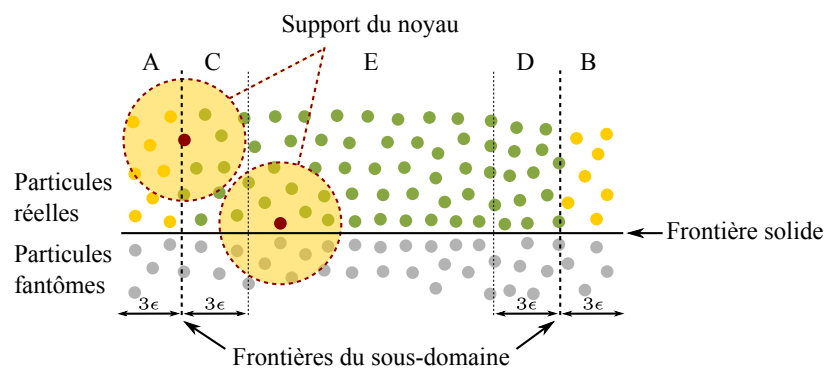
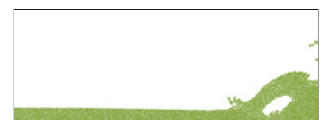


FIGURE 9.3 – Sous-domaine tel qu'il est vu par le processeur correspondant.

Dans notre cas, l'application visée étant principalement des simulations numériques de canal à houle, nous avons exploité le fait que le domaine soit de forme *allongée*. Ainsi, un simple découpage en autant de bandes verticales que de processeurs (figure 9.1) nous a paru optimal :

- Ce découpage ne nécessite qu'un test sur la coordonnée en x de chaque particule pour déterminer son processeur d'appartenance.
- L'initialisation de chaque cas présenté jusqu'ici et du canal à houle est très simple car la taille de chaque sous-domaine peut être déterminée analytiquement avant toute création de particule.
- Les particules situées dans la zone d'interaction (figure 9.2), font l'objet d'appels répétés du processeur voisin à chaque mise à jour des informations. Ces bandes verticales assurent que la taille des zones d'interaction est minimale, réduisant ainsi le volume d'informations à transférer.
- Ce mode de découpage reste compatible avec les cas de simulation étudiés pour cette thèse, autres que le canal à houle (effondrement d'une masse d'eau, carré entraîné, goutte étirée...) (figure 9.6).



En utilisant ce mode de découpage, on s'assure donc une répartition simple et équilibrée de la charge initiale, la possibilité de paralléliser le code de calcul dans son intégralité, de réduire la mémoire nécessaire à chaque processeur et de minimiser le nombre de communications. La figure 9.3 montre la vision qu'a un processeur du sous-domaine qui lui est associé.

9.2 Modifications du code séquentiel

La tâche de parallélisation d'un code séquentiel nécessite de profondes modifications : ajout de directives, création et manipulation de variables spécifiques à la parallélisation. Nous aurions pu préférer un développement parallèle dès le début de l'écriture du code (déploiement des boucles sur les processeurs d'un même nœud en OpenMP, puis gestion des nœuds en MPI) plutôt que de procéder à cette modification. Mais la priorité a été donnée à l'apprentissage de la méthode et du modèle à employer, par l'écriture de plusieurs codes de calcul simples que nous avons finalement unifiés avant de procéder à la parallélisation. Toutes les modifications ont été regroupées dans une librairie indépendante et les données déclarées au sein d'une structure spécifique afin de faciliter le développement. La figure 9.4 montre les différentes étapes de simulation d'un écoulement (la rupture de barrage a été choisie comme exemple mais tous les cas présentés ici suivent la même procédure) en séquentiel et en parallèle. Des routines spécifiques (en **jaune**) ont été ajoutées pour que l'intégralité de la simulation soit effectuée en parallèle. Chacune de ces procédures est détaillée dans la suite de ce chapitre.

L'extrait de code 9.1 définit la structure employée pour la manipulation des données relatives à la parallélisation. De cette manière, le développement du code de calcul est accéléré et simplifié (code 9.2) car un seul argument suffit à transmettre l'ensemble des données nécessaires. Il serait d'ailleurs utile de créer un type de données équivalent regroupant tous les tableaux des grandeurs associées aux particules, pour les mêmes raisons. Un autre moyen de simplifier l'écriture du code de calcul serait de définir toutes ces variables de façon globale (par opposition à la définition locale à chaque procédure employée ici) comme cela a été décidé pour le code **SPHysics** [256]. Les variables globales étant connues en tout point du code de calcul, il n'est plus nécessaire de les passer en argument. Cependant, ce mode de programmation est très souvent rejeté car le débogage de tels codes devient extrêmement difficile car il nécessite de connaître le code dans son intégralité avant de pouvoir déterminer à quel endroit une variable est incorrectement modifiée. L'usage de variables locales et de tableaux alloués dynamiquement impose, en *Fortran 90*, la déclaration d'une interface pour chaque procédure. Quelques lignes de code supplémentaires sont donc requises pour gérer ces variables, mais ces interfaces permettent au *débogueur* de situer très précisément toute erreur éventuelle. Nous préférons donc un regroupement par *thèmes*, en petits ensembles, aisément *traçables* dans le déroulement du code de calcul.

Les particules sont dorénavant classées en 3 types distincts : particules réelles, particules réelles des processeurs voisins, particules fantômes dont les quantités sont respectivement notées N_{part} ,

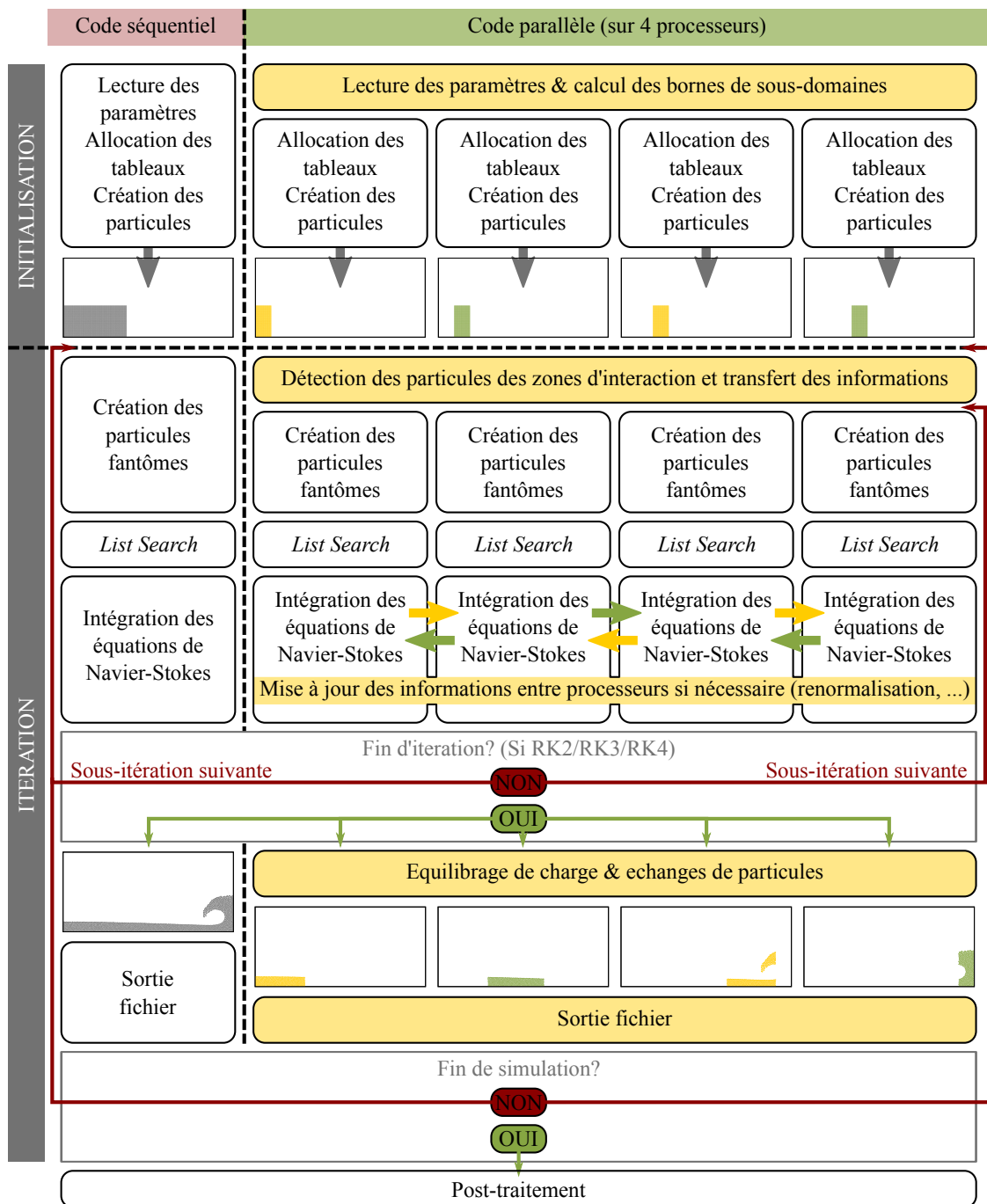
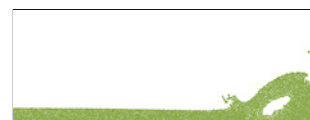


FIGURE 9.4 – Comparaison entre les exécutions séquentielle et parallèle du code de calcul dans le cas de la rupture de barrage. Les boîtes jaune symbolisent les procédures qui ont du être créées.



```

MODULE MOD_PARA
  TYPE INFO_PARA
    !NOMBRE DE PARTICULES DANS LA ZONE A
    INTEGER::NFLUIDEG
    !NOMBRE DE PARTICULES DANS LA ZONE B
    INTEGER::NFLUIDED
    !NOMBRE DE PARTICULES DANS LA ZONE C
    INTEGER::NPARTBORDG
    !NOMBRE DE PARTICULES DANS LA ZONE D
    INTEGER::NPARTBORDD
    !INDICES DES PARTICULES EN INTERACTION
    !AVEC LES VOISINS (ZONES C ET D)
    INTEGER, DIMENSION(:), POINTER :: LEBORDD, LEBORDG
    !NOMBRE DE PARTICULE A TRANSFERER VERS LE PROCESSEUR
    !DE DROITE (ZONE D VERS ZONE B) EN FIN D'ITERATION
    INTEGER::NLISTED
    !NOMBRE DE PARTICULE A TRANSFERER VERS LE PROCESSEUR
    !DE GAUCHE (ZONE C VERS ZONE A) EN FIN D'ITERATION
    INTEGER::NLISTEG
    !INDICES DES PARTICULES A TRANSFERER VERS LES
    !PROCESSEURS VOISINS EN FIN D'ITERATION
    INTEGER, DIMENSION(:), POINTER :: LISTED, LISTEG
    !POSITIONS DES FRONTIERES DU SOUS-DOMAIN
    REAL::BMIN, BMAX
    !NOMBRE DE PROCESSEURS
    INTEGER::NPROC
    !RANG DU PROCESSEUR (ENTRE 0 ET NPROC-1)
    INTEGER::RANG
    !ECART ENTRE LE NOMBRE DE PARTICULES ACTUEL
    !ET LE NOMBRE INITIAL
    INTEGER::DELTA_NPART
  END TYPE INFO_PARA
END MODULE MOD_PARA

```

CODE 9.1 – Structure Fortran 90 pour la mise en œuvre de la parallélisation.

```

INTEGER :: NFLUIDEG, NFLUIDED, NPARTBORDG, NPARTBORDD, NLISTED, NLISTEG
INTEGER, DIMENSION (:), POINTER :: LEBORDD, LEBORDG, LISTED, LISTEG
REAL :: BMIN, BMAX
INTEGER :: NPROC, RANG
INTEGER :: DELTA_NPART
INTEGER :: I, D, ND, NP
REAL, DIMENSION (1:ND, 1:NP) :: X, U
REAL, DIMENSION (1:NP) :: RHO, M
!APPEL D'UNE PROCEDURE
CALL PROCEDURE_APELEE (X, U, RHO, M, NFLUIDEG, NFLUIDED, NPARTBORDG,
    NPARTBORDD, LEBORDD, LEBORDG, NLISTED, NLISTEG, LISTED, LISTEG, BMIN
    , BMAX, RANG, DELTA_NPART)

```

(a) Sans type de données.

```

TYPE (INFO_PARA) :: SOUSDOMAINE
INTEGER :: I, D, ND, NP
REAL, DIMENSION (1:ND, 1:NP) :: X, U
REAL, DIMENSION (1:NP) :: RHO, M
!APPEL D'UNE PROCEDURE
CALL PROCEDURE_APELEE (X, U, RHO, M, SOUSDOMAINE)

```

(b) Avec le type de données INFO_PARA.

```

TYPE (INFO_PARA) :: SOUSDOMAINE
INTEGER :: I, D, ND, NP
TYPE (INFO_PART) :: PARTICULES
!APPEL D'UNE PROCEDURE
CALL PROCEDURE_APELEE (PARTICULES, SOUSDOMAINE)

```

(c) Avec un nouveau type de données INFO_PART.

CODE 9.2 – Utilisation de types de données supplémentaires pour la simplification des appels de procédures.



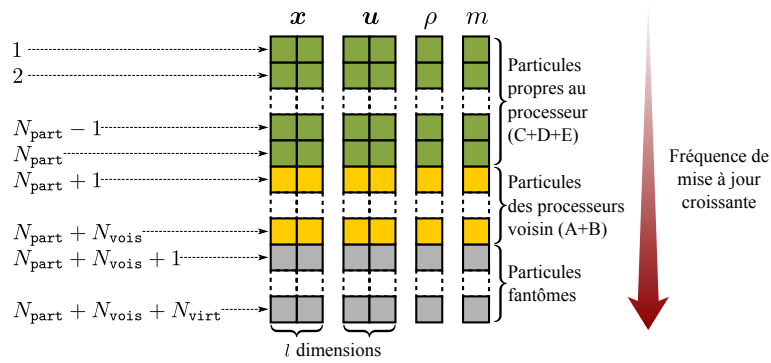


FIGURE 9.5 – Stockage des informations relatives aux particules des processeurs voisins, les ensembles A, B, C, D et E sont définis sur la figure 9.3.

N_{vois} et N_{virt} au sein du code. Toutes les particules sont stockées dans les mêmes tableaux mais en respectant l'ordre représenté sur la figure 9.5. Cet ordre ne requiert pas de tri systématique des tableaux car il a été établi selon la *durée de vie* des particules concernées. En effet, les particules *réelles* sont conservées sur un processeur donné du début à la fin de la simulation. Quelques échanges peuvent avoir lieu quand la charge de calcul est équilibrée entre les processeurs, ce qui n'arrive qu'en fin d'itération. Dans ce cas bien précis, il se peut que des particules *réelles* soient supprimées et qu'une copie du reste du tableau (jusqu'à N_{part}) soit nécessaire pour combler le vide ainsi créé. Les particules des processeurs voisins sont effacées à chaque fin d'itération ; au début de l'itération suivante, les particules de la zone d'interaction (figure 9.2) sont à nouveau détectées (leur position a changé) et stockées à la suite des particules *réelles*. Enfin, les particules fantômes sont effacées puis recrées plusieurs fois par itération, dès que les informations concernant les particules *réelles* ont été modifiées.

9.2.1 Initialisation des simulations

Comme nous l'avons expliqué précédemment, pour la mise en œuvre de simulations à grand nombre de particules (de l'ordre du million), il est important d'économiser l'espace mémoire, et en particulier de pouvoir répartir équitablement les particules entre les processeurs sans avoir à toutes les créer au préalable. Cela implique de connaître, avant toute allocation mémoire, l'emplacement des bornes de sous-domaine. Pour les cas présentés ici, la configuration géométrique permet de calculer ces bornes analytiquement de manière à avoir un nombre égal (ou presque) de particules sur chaque processeur. La figure 9.6 montre le découpage obtenu (avec 4 processeurs) sur chacun des écoulements présentés jusqu'ici en utilisant cette approche.

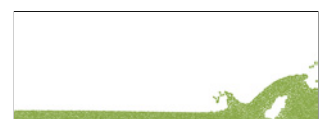
Pour les configurations géométriques plus complexes, il peut s'avérer difficile de prédire la forme des sous-domaines. Une procédure itérative devrait alors permettre de calculer, sans les stocker en mémoire, le nombre de particules que chaque processeur aurait à prendre en charge pour une partition initiale arbitraire du domaine de calcul. Avec une certaine tolérance, il serait alors possible

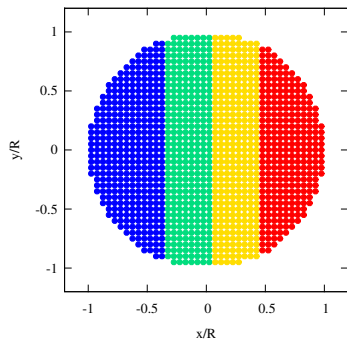
de déplacer les limites de sous-domaines de façon à réduire l'écart à la moyenne sur le nombre de particules gérées par chaque processeur. Après plusieurs itérations et compensations de ces écarts, une répartition acceptable serait obtenue et le nombre de particules à allouer serait connu, permettant ainsi de procéder à l'allocation des tableaux. Cette approche n'a pas encore été testée mais est à l'étude (pour l'importation de cas de simulation préalablement *dessinées* à l'aide d'un logiciel de maillage, par exemple). En revanche, pour la reprise de simulation interrompues prématurément, une routine a été mise en place. Celle-ci permet d'effectuer la reprise sur un nombre différents de processeurs que la simulation d'origine. Deux lectures successives du dernier fichier de sauvegarde sont réalisées sur le nombre total $N_{\text{part}}^{\text{TOTAL}}$ de particules :

- La première ne parcourt que la coordonnée x des particules afin d'en établir un classement selon l'axe correspondant.
- La seconde permet à chaque processeur de stocker toutes les informations relatives aux $N_{\text{part}}^{\text{TOTAL}}/N_{\text{proc}}$ particules qui lui ont été affectées après le premier passage.

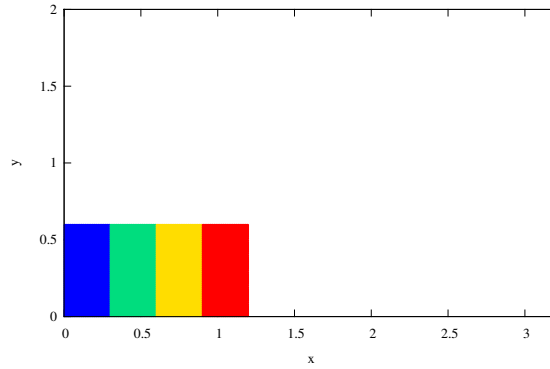
9.2.2 Entrées et sorties fichier

Le code de calcul emploie des fichiers écrits en norme *ASCII* pour la lecture des paramètres de simulation. Ces fichiers, de petite taille, ne requièrent que très peu de temps CPU pour être lus. Ainsi, nous avons choisi de *faire* lire ces fichiers à chaque processeur successivement, plutôt que de centraliser la lecture des paramètres sur un processeur, qui aurait ensuite envoyé un message contenant toutes ces informations aux autres processeurs. Ce choix, simple mais moins performant, est aussi justifié par le fait que les paramètres ne sont lus qu'une fois, en début de simulation. Il est envisagé de ne plus utiliser des fichiers *ASCII* et de les remplacer par d'autres formats (*xml*, par exemple) permettant une gestion plus simple (les fichiers *ASCII* obligent à respecter un format d'écriture fixe et un ordre bien précis ; une erreur d'un simple espace, dans ces fichiers modifiés manuellement pour la préparation des simulations, peut empêcher le code de s'exécuter). Il existe des bibliothèques permettant la manipulation de ces fichiers en langage *Fortran 90*. Les fichiers *ASCII* ont aussi l'inconvénient d'atteindre des tailles prohibitives si un grand nombre de particules doivent être écrites. Actuellement, ces fichiers sont encore utilisés pour la sauvegarde des données (les processeurs écrivent l'un après l'autre leurs données propres) mais l'utilisation d'un format de fichier compressé et manipulable en parallèle est à l'étude : le format **HDF-5** [247]. Bien qu'il existe, en paradigme *MPI-2*, des commandes permettant de lire et écrire des fichiers *ASCII* ou binaire (mais non portable d'un ordinateur à l'autre), la taille de ces fichiers reste trop importante pour une utilisation courante, nous n'avons donc pas jugé nécessaire d'améliorer la gestion parallèle de ces fichiers.

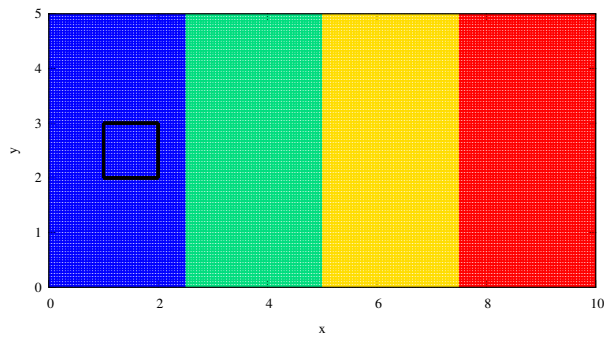




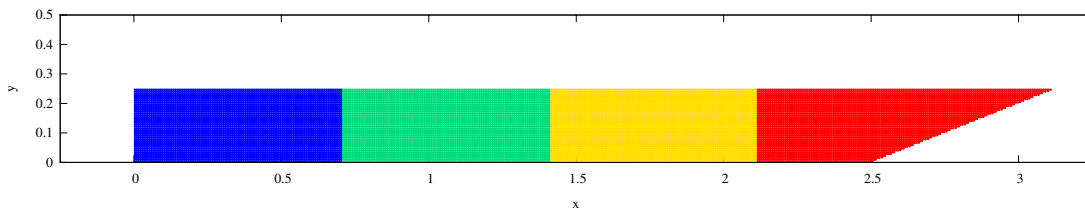
(a) *Goutte étirée.*



(b) *Rupture de barrage.*



(c) *Carré forcé immergé.*



(d) *Canal à houle.*

FIGURE 9.6 – Configuration initiale des particules en parallèle pour différents écoulements, la couleur symbolise le processeur d'affectation de la particule.

9.2.3 Détection des zones d'interaction

Nous avons expliqué que si les domaines gérés par chaque processus sont presque indépendants, il existe des bandes de particules situées de part et d'autre des sous-domaines qui doivent être communiquées aux processeurs voisins. Il est donc nécessaire de détecter les particules qui seraient susceptibles d'interagir avec ces processeurs et de stocker leurs indices dans les tableaux `LEBORDD` et `LEBORDG` de la structure `INFOPARA` (code 9.1) prévue à cet effet.

La détection de ces particules est, encore une fois, simplifiée par le choix du découpage des sous-domaines en bandes verticales. Ainsi, une seule boucle sur la coordonnée x des particules permet de déterminer en début d'itération les particules interagissant avec les processeurs voisins. La figure 9.2 illustre l'importance de cette étape de calcul pour l'intégration des particules proches d'une frontière de sous-domaines. Le choix d'une largeur de $3h$ pour ces zones vient directement du rayon du noyau. Or, en pratique, nous avons choisi de détecter ces particules une seule fois, au début de chaque itération, même si un schéma d'ordre supérieur au schéma d'Euler est employé (Runge-Kutta ...). La description lagrangienne des équations implique que la position des particules soit également intégrée à chaque sous-itération de ces schémas, si bien que certaines particules peuvent sortir de la zone d'interaction ou y entrer entre deux sous-pas de temps successifs. Nous avons donc choisi d'agrandir la taille de ces zones d'interaction à $3\epsilon + \epsilon = 4\epsilon$, sachant qu'une particule ne se déplace pas d'une distance supérieure à ϵ par pas de temps (condition CFL). Entre chaque sous-itération, il suffit alors de mettre à jour uniquement (sans les détecter à nouveau) ces particules entre processeurs voisins, selon le protocole défini dans le paragraphe suivant.

9.2.4 Communications

Les informations concernant les particules des zones d'interaction doivent régulièrement être mises à jour afin d'assurer la cohérence de l'ensemble de la simulation. Cet échange de messages a lieu (figure 9.4) :

- En début d'itération, une fois les particules des zones d'interaction détectées, avant de commencer le calcul des équations de Navier-Stokes proprement dit.
- A la fin de chaque sous-itération, pour la mise à jour des variables intégrées avant le début de la sous-itération suivante.
- Au cours de chaque sous-itération, pour certaines variables, comme la matrice de renormalisation, par exemple, si les formules (4.38) et (4.39) sont employées.

Sachant qu'un processeur ne peut effectuer plus d'une action à la fois et que les informations relatives à une zone d'interaction *n'intéressent* que les deux processeurs adjacents, l'ordre proposé sur la figure 9.7 nous a paru optimal. Ainsi les quatre étapes suivantes sont respectées :

Etape 1 : Les processeurs de rang pair envoient les messages vers leur voisin de rang supérieur.

Etape 2 : Les processeurs de rang impair envoient les messages vers leur voisin de rang inférieur.



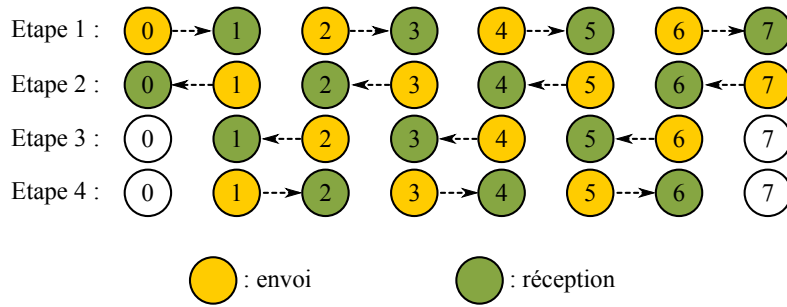


FIGURE 9.7 – Ordonnancement des envois et réceptions de messages entre 8 processeurs.

Etape 3 : Les processeurs de rang pair envoient les messages vers leur voisin de rang inférieur, sauf le processeur de rang 0.

Etape 4 : Les processeurs de rang impair envoient les messages vers leur voisin de rang supérieur, sauf le processeur de rang $N_{\text{proc}} - 1$.

Les particules concernées (c'est à dire située dans une zone d'interaction) ne sont pas stockées de façon contiguë en mémoire sur leur processeur d'appartenance. Cela nous a obligé à utiliser des tableaux intermédiaires pour l'envoi des informations afin de les regrouper avant le départ du message. Nous nous sommes également intéressé à l'utilisation de communications non-bloquantes avec pour objectif de réduire le temps consacré à l'ensemble des communications. Ainsi les informations sont envoyées ou attendues par chaque processeur pendant la recopie du tableau-*tampon* relatif aux données précédentes. Le gain apporté par ces différents choix n'a pas pu être quantifié de façon significative. En effet, le temps consacré aux communications est très petit devant le temps de calcul des équations.

9.2.5 Équilibrage de la charge

Toutes les étapes de mise en œuvre de cette parallélisation s'appuient sur le découpage en bandes verticales du domaine de calcul. Or, après plusieurs intégrations en temps des positions des particules, il peut arriver que certaines d'entre-elles se déplacent progressivement hors du domaine géré initialement par leur processeur. Quand ce cas se présente et pour maintenir le découpage du domaine, nous avons choisi de transférer la gestion de la particule (et donc toutes les informations la concernant) au processeur voisin. Cette technique est assez simple à mettre en place, en revanche une conséquence directe est alors que le nombre de particules dans chaque sous-domaine peut varier. Après plusieurs itérations, on peut même observer un déséquilibre important dans la répartition des particules entre les différents processeurs. Selon la dynamique de l'écoulement simulé, ce déséquilibre peut apparaître plus ou moins rapidement mais dégrade dans tous les cas les performances du code de calcul. La figure 9.8 illustre le déséquilibre obtenu dans le cas de la rupture de barrage simulée sur quatre processeurs. On peut noter que le dernier processeur récupère

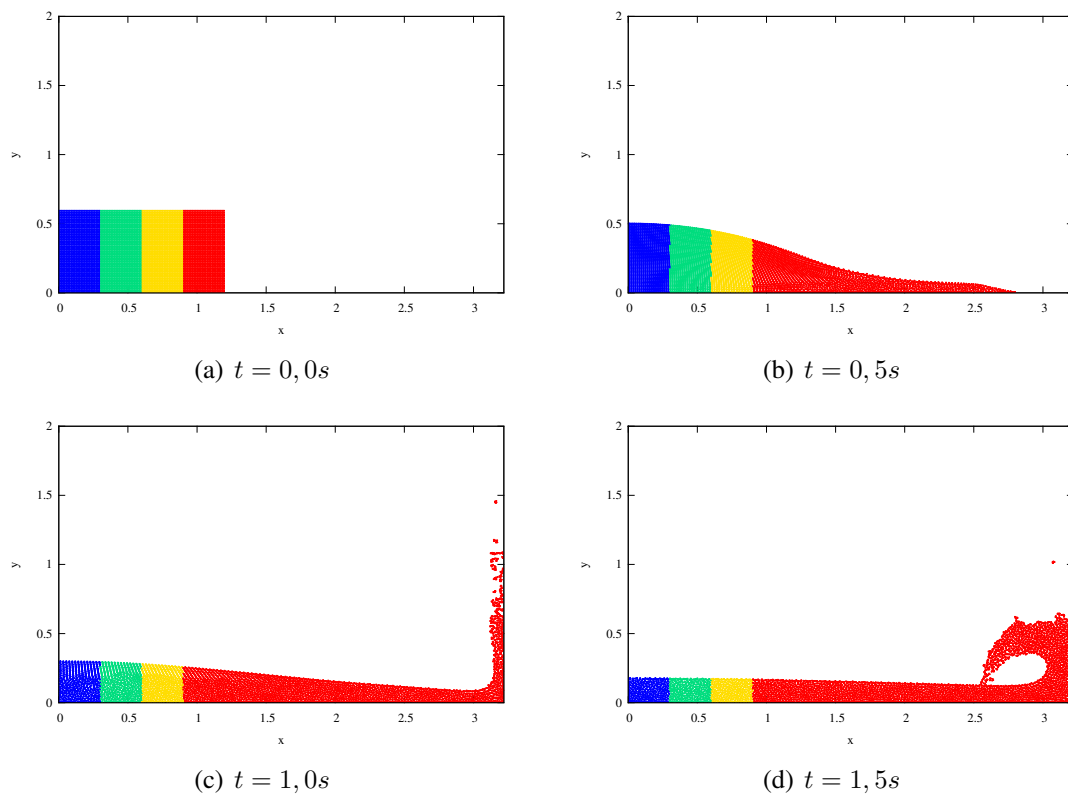
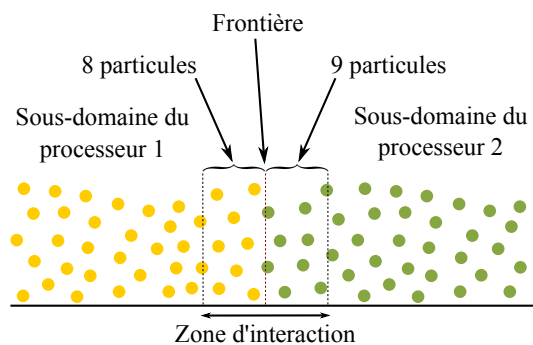
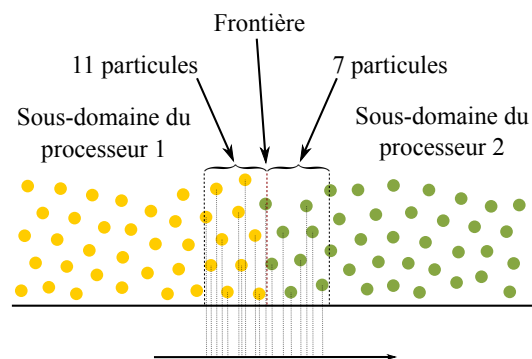


FIGURE 9.8 – Évolution de la répartition des particules sur quatre processeurs dans le cas d'une rupture de barrage, sans équilibrage de charge. La couleur symbolise le processeur d'appartenance.

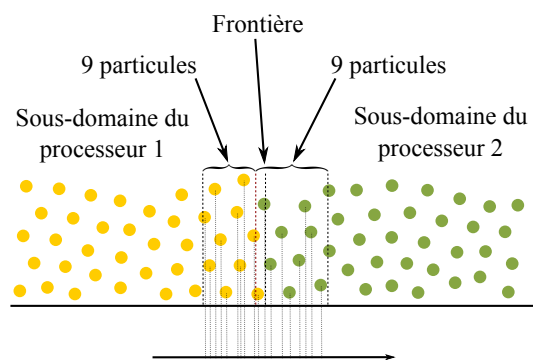




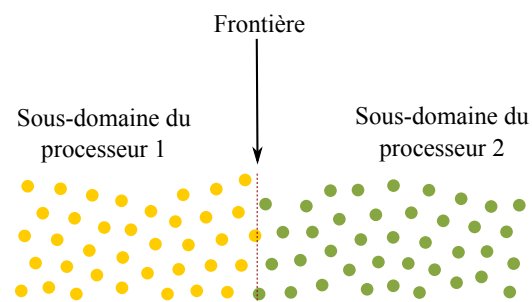
(a) Début d'itération et détection de la zone d'interaction



(b) Fin d'itération, les particules ont été déplacées créant un déséquilibre entre les deux parties de la zone d'interaction.



(c) Calcul de la nouvelle position de la frontière.



(d) Transfert des particules.

FIGURE 9.9 – Tapes de l'équilibrage de la charge de calcul entre deux processeurs adjacents.

une grande partie de l'effort de calcul en fin de simulation. Cet exemple bien qu'extrême (ce déséquilibre est plus difficile à observer au cours de simulations de houle en canal) montre la nécessité de réévaluer la position des limites de sous-domaines au cours de la simulation afin de maintenir une répartition équitable des particules entre les processeurs.

Ainsi, à chaque fin d'itération, une procédure a été ajoutée permettant de déplacer les frontières entre chaque processeur. Tous les processeurs, sauf le dernier en rang, déterminent l'ensemble de particules qui changent de sous-domaine pour celui du processeur voisin. Ce flux de particules est symétrique (ce qui quitte un processeur, entre dans le suivant), ce qui justifie le fait d'employer un seul processeur pour évaluer la nouvelle position de la frontière de droite (et donc de ne pas utiliser le dernier). La figure 9.9 montre les différentes étapes de ce processeur d'équilibrage de charge. Les étapes sont les suivantes :

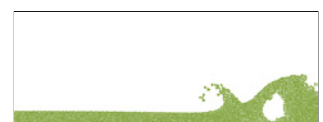
Fig. 9.9-(a) Au début de l'itération, les particules de la zone d'interaction sont détectées.

Fig. 9.9-(b) En fin d'itération, une nouvelle détection de ces particules est nécessaire après intégration de leur position. On note un déséquilibre entre les parties de processeur 1 et du processeur 2 de cette zone d'interaction. Cette différence en nombre de particules est un indicateur de la dynamique locale de l'écoulement. On regroupe les coordonnées x de chaque particule dans un même tableau.

Fig. 9.9-(c) A l'aide de l'algorithme `QuickSelect()` (code 9.3), on calcule une valeur pour x , parmi les coordonnées regroupées à l'étape précédente, permettant d'avoir autant de particules dans la zone de gauche (`NPARTBORDD`, voir l'extrait de code 9.1) qu'au début de l'itération (ou avec un écart maximal d'une particule), déterminant ainsi une nouvelle position de la frontière.

Fig. 9.9-(d) Les particules *mal* positionnées sont transférées d'un processeur à l'autre. L'itération suivante peut commencer.

Après quelques essais, nous avons observé l'accumulation d'écarts d'une particule à chaque itération. La conséquence est un déséquilibre de la charge entre les processeurs qui peut devenir important, compte tenu du nombre total d'itérations lors d'une simulation (plusieurs milliers voire centaines de milliers). Nous avons donc introduit une *mémoire* dans la structure 9.1, `DELTA_NPART`, dans le but de stocker en mémoire l'écart entre le nombre de particules actuel du processeur et le nombre initial. Ainsi, en plaçant la nouvelle frontière à la $(NPARTBORDD + DELTA_NPART/2)^{\text{ème}}$ valeur en x des particules de la zone d'interaction, on s'assure qu'un écart généré à une itération donnée sera corrigé à l'itération suivante. Cette procédure d'équilibrage est rapide et efficace. Les figures 9.8 et 9.10 montrent le partitionnement du domaine au cours du temps, obtenu respectivement sans et avec la mise en œuvre de cette technique.



```

RECURSIVE FUNCTION QUICKSELECT (A,K) RESULT (QS)
  REAL::QS
  INTEGER::K
  REAL,DIMENSION(:)::A
  REAL,DIMENSION(:), POINTER::A1,A2,A3
  REAL::PIVOT
  !CHOISIR UNE VALEUR PIVOT PARMIS LES ELEMENTS DU TABLEAU A
  !PARTITIONNER A EN 3 SOUS-TABLEAUX: A1<P,A2=P,A3>P
  IF (K<=LONGUEUR (A1) ) THEN
    QS=QUICKSELECT (A1,K)
  ELSE
    IF (K>LONGUEUR (A1) +LONGUEUR (A2) ) THEN
      QS=QUICKSELECT (A3,K-LONGUEUR (A1) -LONGUEUR (A2) )
    ELSE
      QS=PIVOT
    ENDIF
  ENDIF
  RETURN
END FUNCTION QUICKSELECT

```

CODE 9.3 – Fonction QuickSelect ().

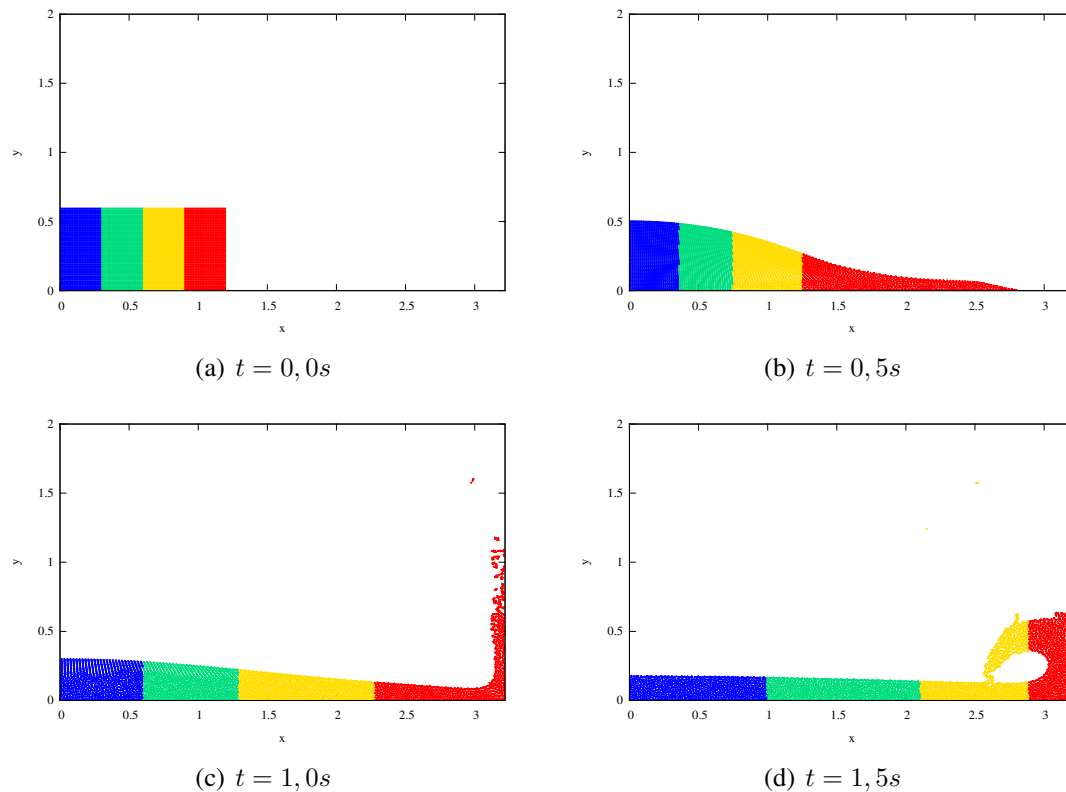


FIGURE 9.10 – Évolution de la répartition des particules sur quatre processeurs dans le cas d'une rupture de barrage, avec équilibrage de charge. La couleur symbolise le processeur d'appartenance.

9.3 Analyse des performances

A l'aide des algorithmes décrits précédemment, la simulation d'écoulements en parallèle a été réalisée et les résultats obtenus sont identiques à ceux produits par la version séquentielle du code de calcul. Nous avons souhaité quantifier le temps de calcul consacré à la parallélisation, à savoir le temps passé dans chacune des procédures de la librairie de parallélisation. A cet effet, une simulation de rupture de barrage avec 5000 particules réparties sur 4 processeurs sur 10000 itérations a été réalisée. Le tableau 9.1 montre le temps (en ms) passé sur les différentes tâches liées à la parallélisation et aux calculs. On remarque que la détection des particules des zones d'interaction et l'équilibrage de la charge entre les processeurs occupent une part négligeable du temps global. Les communications entre processeurs constituent l'étape la plus coûteuse de cette parallélisation, mais reste très inférieure au temps consacré à la résolution des équations de Navier-Stokes.

Une étude a d'ailleurs été réalisée à l'aide des outils de profilage de code fournis par le CRIHAN [49]. Nous avons constaté que les processeurs mis en œuvre sur un calcul sont exploités à leur capacité maximale. La seule période de *latence* qui a pu être observée a lieu pendant les deux dernières étapes des transmissions de données entre processeurs (figure 9.7), où le premier et le dernier processeur sont inactifs car ils n'ont qu'un seul processeur voisin avec lequel échanger des informations.

	Détection des bords	Communications	Équilibrage de la charge	Calculs
Temps d'exécution (en ms)	0,95	8,72	1,03	1622,47
Fraction du temps total (en %)	0,06	0,53	0,06	98,35

TABLE 9.1 – Temps d'exécution de chacune des étapes du calcul d'une rupture de barrage (5000 particules sur 10000 itérations).

Une façon de mesurer la performance d'un code parallèle consiste à calculer l'accélération permise par l'utilisation de N_p processeurs :

$$A(N_p) = \frac{T_1}{T_{N_p}}, \quad (9.1)$$

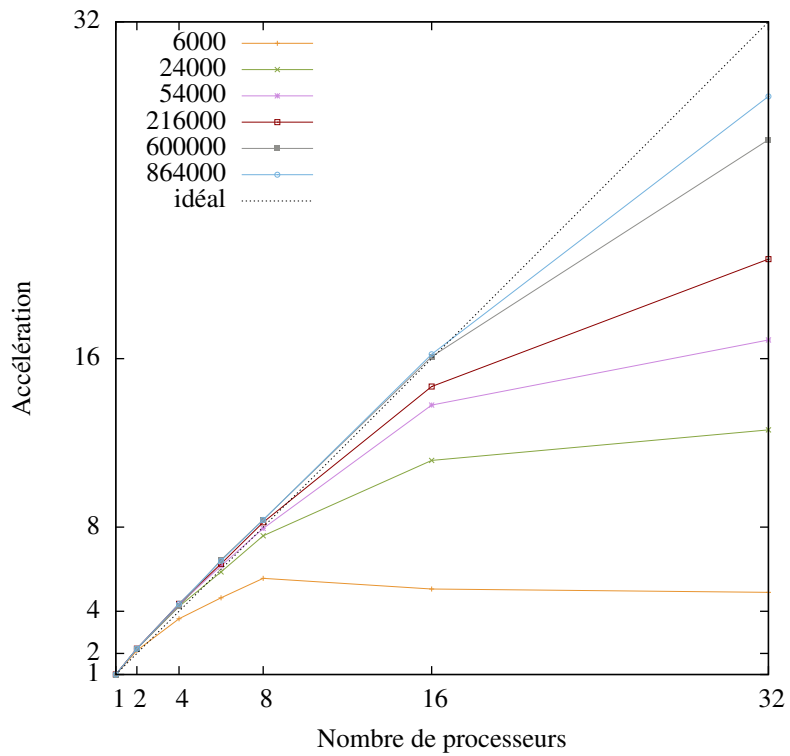
où T_1 est le temps nécessaire à la simulation d'un écoulement donné par le code séquentiel et T_{N_p} , le temps de simulation du même écoulement sur N_p processeurs. $A(N_p)$ correspond à un nombre effectif de processeurs. Idéalement, l'accélération serait $A(N_p) = p$ si le calcul était simplement réparti entre les processeurs sans que des zones d'interaction ne nécessitent le partage d'informations. La figure 9.11 représente l'accélération ainsi mesurée pour deux écoulements différents, un canal à houle et une rupture de barrage, en fonction du nombre de particules. Ces mesures font



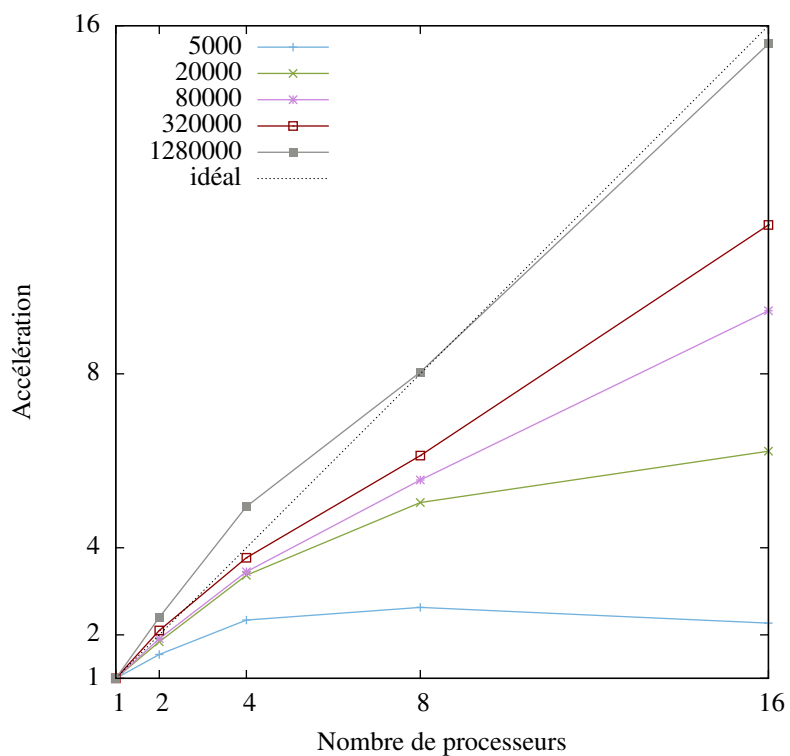
intervenir le temps de simulation du code séquentiel, qui ne peut être décemment obtenu quand le nombre de particules devient trop grand. Cela explique que ces résultats sont limités à environ un million de particules. Des mesures pourraient être réalisées à nombre de particules supérieur, en calculant une approximation du temps séquentiel, mais les résultats seraient contestables.

Quelque soit l'écoulement simulé, on constate qu'à faible nombre de particules, le gain permis par la parallélisation est limité. Il n'est effectivement pas nécessaire d'employer beaucoup de processeurs, car le temps consacré aux communications de données deviendrait supérieur au calcul lui-même. En revanche, l'accélération s'approche rapidement de la courbe idéale quand on augmente le nombre de particules. A faible nombre de processeurs et grand nombre de particules, l'accélération parvient même à dépasser la courbe idéale. Ce résultat inattendu s'explique par la réduction de la taille des tableaux du code parallèle par rapport au code séquentiel, et ainsi à une diminution des temps d'accès à la mémoire.

Par ailleurs, on peut constater que l'effondrement d'une masse d'eau est simulée de façon moins efficace en parallèle que la propagation de houle en canal (à nombre équivalent de particules). Cela vient du temps consacré à l'équilibrage de la charge, qui est nettement supérieur dans le cas d'une rupture de barrage, où le domaine fluide se déforme beaucoup plus que dans le cas du canal à houle. Le rapport d'aspect du domaine fluide est aussi moins avantageux dans le cas de la rupture de barrage, ce qui conduit à des zones d'interactions plus *hautes* que dans le cas du canal à houle.



(a) Canal à houle.



(b) Rupture de barrage.

FIGURE 9.11 – Accélération du code parallèle par rapport au code séquentiel dans le cas d'un canal à houle et d'une rupture de barrage en fonction du nombre de processeurs employés et du nombre de particules.



Conclusion

Une méthode SPH a été développée pour la simulation d'écoulements visqueux à surface libre. En s'appuyant sur les différentes approches proposées dans la littérature, nous avons construit une méthode spécifique, basé sur une vision faiblement compressible de l'écoulement. Ainsi, les équations, discrétisées au cours du chapitre I, ont été appliquées à un écoulement à surface libre académique, celui de la goutte étirée (chapitre 4). L'apport de corrections aux équations, telles que le *flux Rusanov*, a permis d'obtenir des résultats précis tout en diminuant le nombre de paramètres numériques du modèle (§4.5.1). Nous avons également noté qu'il était nécessaire de mettre en œuvre les méthodes de post-traitement appropriées, avec la normalisation des noyaux d'interpolation par exemple (§4.6.1), pour extraire les grandeurs pertinentes du champ de particules. Ensuite, le modèle a été étendu aux problèmes d'hydrodynamique à surface libre, sans viscosité, en présence de parois solides. La méthode des *particules fantômes*, adoptée pour les conditions de non-pénétration, s'est montrée adaptée à des configurations géométriques simples. L'étude de l'effondrement d'une colonne d'eau en 2D a mis en évidence la nécessité d'apporter de nouvelles corrections, à la condition de non-pénétration (§5.4.2) et au flux Rusanov (§5.4.2). Le modèle a été, de nouveau, validé dans ce contexte et ses propriétés de conservation ont été établies. Pour la simulation de corps solides immergés dans un fluide visqueux, une méthode différente a été développée, basée sur la *frontière immergée* (chapitre 6). L'usage d'un noyau corrigé par moindres carrés (§6.1.2) a permis d'accroître la précision au voisinage des parois et de dispenser les simulations de remaillages périodiques du champ de particules, comme proposé antérieurement par Hieber et Koumoutsakos [93]. Le forçage d'un cylindre de section carré dans un fluide visqueux incompressible a été simulé avec succès par ce modèle SPH de fluide visqueux faiblement compressible combiné à la *frontière immergée*. Les perturbations numériques de l'écoulement au voisinage des angles droits, habituellement observées, ont été supprimées, et les efforts calculés sont en accord avec la solution de référence. Les ondes de pression générées par l'accélération du carré perturbent légèrement la simulation. Ces oscillations sont inhérentes au modèle faiblement compressible et sont particulièrement mises en évidence sur ce cas. Le développement d'autres conditions aux limites serait nécessaire pour achever la validation du modèle ou le compléter, soit en imposant la non-réflexion de ces ondes de pression aux parois, soit en simulant l'écoulement autour d'un cylindre de section circulaire en domaine *infini* à l'aide de conditions d'entrée/sortie.

Une application du modèle à la propagation d'ondes de gravité a été proposée. Le modèle non-



visqueux a d'abord été employé pour la simulation d'ondes en canal à houle, muni d'une plage d'amortissement numérique. La *frontière immergée*, combinée au modèle visqueux, a ensuite été appliqué à l'amortissement de la houle par une plaque immergée. Ce dernier exemple montre la capacité de la *frontière immergée* à modéliser l'action d'objets sur le fluide avec robustesse et précision, et sans que le remaillage ne soit indispensable. En l'état, le code de calcul n'est applicable qu'aux corps indéformables (fixes ou forcés) totalement immergés dans un fluide visqueux, mais elle apporte déjà une solution aux problèmes souvent rencontrés lors de la modélisation de parois de forme complexe (angles droits, parois courbes, par exemple). Plusieurs perspectives s'offrent donc pour le développement futur de ce modèle.

D'une part, l'application du modèle à des corps partiellement immergés est à l'étude. L'apport de nouvelles corrections est nécessaire, en particulier, quand le support d'un marqueur de frontière n'est que partiellement rempli de particules. La *frontière immergée* couplée à une méthode SPH prendrait alors tout son sens, appliquée par exemple à l'étude d'un écoulement à surface libre dans un domaine quelconque (en présence d'obstacles). On pourrait également envisager de modéliser les parois extérieures du domaine de calcul avec la *frontière immergée* et de supprimer les écoulements internes lors de la simulation d'objets *clos*. Le gain en précision et efficacité serait significatif. Son extension à une condition en vitesse de type glissement est d'ailleurs à l'étude, en employant un schéma spécifique, inspiré par ceux employés en méthode des intégrales de surface (développée au paragraphe 5.3.1).

D'autre part, malgré les optimisations apportées au chapitre 8 et la parallélisation proposée au chapitre 9, la simulation de houle en canal à l'aide du modèle visqueux nécessite une discrétisation très fine du domaine et des temps de calcul très importants. De même, l'étude d'écoulements en trois dimensions, bien que possible en l'état (Annexe D.5), demande également d'importantes ressources informatiques. Des schémas SPH à pas d'espace variable ont déjà été développés dans ce sens, en réservant la plus fine discrétisation aux zones d'intérêt accru. Une précision comparable peut ainsi être obtenue avec moins de particules. L'attrait de ces schémas est cependant limité par les erreurs dues aux variations du pas d'intégration. C'est pourquoi ces variations ne sont que progressives actuellement. La correction de ces erreurs et l'écriture de schémas adaptés à une diminution *brutale* du pas pourrait permettre de raffiner automatiquement le champ de particules, en fonction de critères prédéfinis. La pertinence de ce raffinement serait bien sur progressivement dégradée par les déplacements des particules. Un remaillage périodique pourrait permettre de maintenir la précision souhaitée, et de restaurer la pertinence du raffinement local. Ce remaillage vient en contradiction avec la démarche adoptée pour l'application de la *frontière immergée*, mais il serait alors indispensable. Dans le cas précis du carré forcé notamment, avec la méthode de Hieber et Koumoutsakos [93] (§6.1.1), un remaillage était nécessaire à chaque itération. La fréquence de remaillage adaptatif serait inférieure et dépendrait essentiellement des vitesses de déplacement des particules. Par ailleurs, les difficultés liées au remaillage cartésien uniforme d'une surface libre pourraient être, en partie, réduites dans le cas d'un remaillage adaptatif, en choisissant de raffiner

le maillage également au voisinage de cette interface. L'erreur sur la conservation du volume global, en particulier, serait inférieure. La mise en œuvre de cette méthode particulière à raffinement automatique dépend essentiellement du schéma d'interpolation et d'approximation des opérateurs différentiels.

Enfin, la modélisation SPH d'écoulements visqueux, comme celui autour de la plaque immergée, quand la viscosité demeure faible, doit être approfondie. Le choix du terme visqueux peut avoir une influence significative qu'il s'agit d'analyser. L'ajout d'un modèle de turbulence pourrait également étendre les capacités du code sur les applications futures.



Annexes



Annexe A

Traitement des opérateurs différentiels en méthode PSE

Nous nous sommes intéressés à l'interpolation et l'approximation d'opérateurs différentiels en méthode particulière. Les conclusions issues de ces travaux ont été présentées dans la première partie de cette thèse (chapitre 2). Une partie de cette thèse a été consacrée à la construction de noyaux spécifiques pour l'approximation de dérivées avec une précision accrue. En suivant la méthode proposée par Eldredge *et al.* [65], il est possible d'écrire des noyaux permettant d'atteindre une précision supérieure à celle obtenue en SPH. L'expérience numérique a révélé que l'usage des noyaux proposés par Eldredge *et al.* ou de leurs variantes basées sur des fonctions *splines* n'est profitable que dans des configurations où les particules restent peu désordonnées. Nous leur avons donc finalement préféré le noyau gaussien modifié (équation (2.41)) pour un usage courant.

A.1 Méthodologie

Pour construire l'approximation de l'opérateur différentiel D^β appliqué à f , on utilise le développement de Taylor en \mathbf{y} de la fonction f , supposée analytique, à partir du point \mathbf{x} :

$$f(\mathbf{y}) = f(\mathbf{x}) + \sum_{|\alpha|=1}^{\infty} \frac{1}{\alpha!} (\mathbf{y} - \mathbf{x})^\alpha D^\alpha f(\mathbf{x}) \quad (\text{A.1})$$

On retranche $f(\mathbf{x})$ de chaque côté de l'égalité et on multiplie les deux membres par le noyau, pour l'instant inconnu, η_ϵ^β . On obtient :

$$(f(\mathbf{y}) - f(\mathbf{x}))\eta_\epsilon^\beta(\mathbf{x} - \mathbf{y}) = \sum_{|\alpha|=1}^{\infty} \frac{1}{\alpha!} (\mathbf{y} - \mathbf{x})^\alpha D^\alpha f(\mathbf{x})\eta_\epsilon^\beta(\mathbf{x} - \mathbf{y}) \quad (\text{A.2})$$



On intègre l'expression précédente par rapport à \mathbf{y} :

$$\int (f(\mathbf{y}) - f(\mathbf{x})) \eta_\epsilon^\beta(\mathbf{x} - \mathbf{y}) d\mathbf{y} = \int \sum_{|\alpha|=1}^{\infty} \frac{1}{\alpha!} (\mathbf{y} - \mathbf{x})^\alpha D^\alpha f(\mathbf{x}) \eta_\epsilon^\beta(\mathbf{x} - \mathbf{y}) d\mathbf{y} \quad (\text{A.3})$$

En remplaçant le premier membre de l'égalité par la définition de l'opérateur PSE (2.31) et en appliquant le théorème de la convergence dominée au second membre (on admet que toutes les hypothèses nécessaires sont réunies), on a :

$$\epsilon^{|\beta|} \mathcal{L}^\beta f(\mathbf{x}) = \sum_{|\alpha|=1}^{\infty} \frac{1}{\alpha!} D^\alpha f(\mathbf{x}) \int (\mathbf{y} - \mathbf{x})^\alpha \eta_\epsilon^\beta(\mathbf{x} - \mathbf{y}) d\mathbf{y} \quad (\text{A.4})$$

Intéressons-nous à l'intégrale présente dans la somme en second membre, qu'on applique sur l'espace entier (1D, 2D ou 3D) :

$$I = \int (\mathbf{y} - \mathbf{x})^\alpha \eta_\epsilon^\beta(\mathbf{x} - \mathbf{y}) d\mathbf{y} = \int_{\mathbf{R}} (\mathbf{y} - \mathbf{x})^\alpha \eta_\epsilon^\beta(\mathbf{x} - \mathbf{y}) d\mathbf{y} \quad (\text{A.5})$$

Posons $\mathbf{z} = \frac{\mathbf{x} - \mathbf{y}}{\epsilon}$, on alors $d\mathbf{y} = (-\epsilon)^l d\mathbf{z}$ où l est le nombre de dimensions physiques du domaine. On a aussi $\eta_\epsilon^\beta(\mathbf{x} - \mathbf{y}) = \frac{1}{\epsilon^l} \eta^\beta(\mathbf{z})$.

$$\begin{aligned} I &= (-1)^l \int_{\mathbf{R}} (-z\epsilon)^\alpha \frac{1}{\epsilon^l} \eta^\beta(\mathbf{z}) (-\epsilon)^l d\mathbf{z} \\ &= \int_{\mathbf{R}} (-z\epsilon)^\alpha \eta^\beta(\mathbf{z}) d\mathbf{z} \\ &= (-1)^\alpha \epsilon^\alpha \int_{\mathbf{R}} z^\alpha \eta^\beta(\mathbf{z}) d\mathbf{z} \end{aligned} \quad (\text{A.6})$$

Or $(-1)^\alpha = (-1)^{\alpha_1} (-1)^{\alpha_2} \dots (-1)^{\alpha_l} = (-1)^{|\alpha|}$, de même, $\epsilon^\alpha = \epsilon^{|\alpha|}$.

En remplaçant dans (A.4), on obtient alors :

$$\mathcal{L}^\beta f(\mathbf{x}) = \sum_{|\alpha|=1}^{\infty} \frac{(-1)^{|\alpha|} \epsilon^{|\alpha| - |\beta|}}{\alpha!} D^\alpha f(\mathbf{x}) M_\alpha \quad (\text{A.7})$$

Avec M_α , le moment d'ordre α du noyau η .

De la même manière que l'opérateur SPH, l'opérateur d'approximation PSE s'écrit comme une combinaison linéaire des dérivées, il suffit alors de choisir la valeur de β correspondant à l'opérateur différentiel visé que l'on veut isoler. On impose alors que tous les moments d'ordres différents de β soient nuls et tous les termes de (A.7) non désirés disparaissent. On peut tout à fait envisager de construire une approximation d'une combinaison linéaire de plusieurs opérateurs, il suffit

d'imposer les valeurs des moments en conséquence.

$$\begin{aligned} \mathcal{L}^\beta f(\mathbf{x}) &= \frac{(-1)^{|\beta|}}{\beta!} M_\beta D^\beta f(\mathbf{x}) \\ &+ \sum_{\substack{|\alpha|=1 \\ |\alpha| \neq |\beta|}}^{\infty} \frac{(-1)^{|\alpha|}}{\alpha!} \epsilon^{|\alpha|-|\beta|} M_\alpha D^\alpha f(\mathbf{x}) \end{aligned} \quad (\text{A.8})$$

$$+ \sum_{\substack{|\alpha|=|\beta| \\ \alpha \neq \beta}} \frac{(-1)^{|\alpha|}}{\alpha!} M_\alpha D^\alpha f(\mathbf{x}) \quad (\text{A.9})$$

Ainsi, pour limiter notre approximation à l'opérateur différentiel D^β , tous les termes présents dans les sommes (A.8) et (A.9) doivent être annulés, on en déduit une série de conditions sur les moments M_α , pour une approximation d'ordre r :

$$M_\alpha = \begin{cases} (-1)^{|\beta|} \beta! & \alpha = \beta \\ 0 & |\alpha| = |\beta|, \alpha \neq \beta \\ 0 & |\alpha| \in [1, |\beta| - 1] \cup [|\beta| + 1, |\beta| + r - 1] \end{cases} \quad (\text{A.10})$$

On aura également :

$$\int |\mathbf{y}|^{|\beta|+r} |\eta^\beta(\mathbf{y})| d\mathbf{y} < \infty \quad (\text{A.11})$$

En utilisant la méthode décrite précédemment, on peut construire facilement une approximation d'opérateurs différentiels, en voici, maintenant quelques exemples.

A.2 Application au gradient

On cherche à approcher la dérivée première à l'ordre 2 selon la première coordonnée dans un domaine à deux dimensions. L'opérateur différentiel correspondant s'écrit :

$$D^{(1,0)} = \frac{\partial}{\partial x_1} \quad (\text{A.12})$$

L'opérateur d'approximation PSE s'écrit alors :

$$\mathcal{L}^{(1,0)} f(\mathbf{x}) = \frac{1}{\epsilon} \int (f(\mathbf{y}) - f(\mathbf{x})) \eta_\epsilon^{(1,0)}(\mathbf{x} - \mathbf{y}) d\mathbf{y} \quad (\text{A.13})$$

En utilisant le noyau d'interpolation η_ϵ^1 construit à partir des conditions suivantes :

$$\begin{cases} M_{(1,0)} = -1 & M_{(0,1)} = 0 \\ M_{(2,0)} = 0 & M_{(1,1)} = 0 & M_{(0,2)} = 0 \end{cases} \quad (\text{A.14})$$



La construction du noyau approprié peut être réalisée en suivant plusieurs démarches. La partie suivante montre deux formes de noyaux pouvant être utilisés. Ces noyaux ont généralement des propriétés de symétrie permettant d'éliminer certaines conditions facilement. Ainsi, dans l'exemple ci-dessus, l'utilisation d'un noyau impair selon la première dimension et pair dans le sens de la seconde élimine les conditions suivantes $M_{(0,1)} = 0$, $M_{(2,0)} = 0$, $M_{(1,1)} = 0$ et $M_{(0,2)} = 0$. Au final, le noyau ne doit respecter qu'une seule condition : $M_{(1,0)} = -1$.

On a donc le choix entre :

- Un noyau quelconque ayant 5 degrés de liberté.
- Un noyau ayant les deux propriétés de symétrie citées plus haut et un seul degré de liberté.

A.3 Application à l'opérateur laplacien

Pour illustrer la possibilité d'approcher une combinaison linéaire d'opérateurs différentiels, voici l'exemple du laplacien, à l'ordre 2 :

$$\Delta = \nabla^2 = D^{(2,0)} + D^{(0,2)} = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} \quad (\text{A.15})$$

On utilise :

$$\mathcal{L}^\Delta f(\mathbf{x}) = \frac{1}{\epsilon^2} \int (f(\mathbf{y}) - f(\mathbf{x})) \eta_\epsilon^\Delta(\mathbf{x} - \mathbf{y}) d\mathbf{y} \quad (\text{A.16})$$

Avec pour conditions :

$$\begin{cases} M_{(1,0)} = 0 & M_{(0,1)} = 0 \\ M_{(2,0)} = 2 & M_{(1,1)} = 0 & M_{(0,2)} = 2 \end{cases} \quad (\text{A.17})$$

Dans ce cas, l'utilisation d'un noyau à symétrie radiale permet de n'avoir qu'une seule condition à satisfaire : $M_{(2,0)} = 2$. En revanche, si on utilise un tel noyau, il devient impossible d'isoler la dérivée seconde partielle selon une des deux coordonnées. On peut alors partir d'un noyau adapté présentant un degré de régularité supérieur dans la direction voulue.

A.4 Noyaux de dérivation

Une fois l'opérateur d'approximation PSE construit et les conditions sur les moments M_α établies, il reste à définir le noyau d'interpolation requis pour l'approximation d'ordre r voulue. Les noyaux utilisés sont identiques à ceux de la méthode SPH et la méthode de construction est proche (seules les conditions sur les moments différent). Dans la suite, nous développons la méthode de création des noyaux d'interpolation pour deux familles courantes : les noyaux gaussiens proposés par Eldredge et al. [65] et une nouvelle famille de noyaux construits pour PSE à l'aide de *B-Splines*.

A.4.1 Noyaux gaussiens

Prenons l'exemple de l'opérateur $D^{(2)} = \frac{\partial^2}{\partial x^2}$ approché à l'ordre 4 dans un espace à une dimension :

$$D^{(2)}f(x) \approx \mathcal{L}^{(2)}f(x) = \frac{1}{\epsilon} \int (f(y) - f(x))\eta_\epsilon^{(2)}(x-y)dy \quad (\text{A.18})$$

Les conditions sont :

$$M_1 = 0 \quad M_2 = 2 \quad M_3 = 0 \quad M_4 = 0 \quad (\text{A.19})$$

Pour simplifier la construction du noyau, il faut réduire le nombre de degrés de libertés. Les auteurs proposent de partir d'une forme générique symétrique :

$$\eta^\beta(-\mathbf{x}) = (-1)^{|\beta|}\eta^\beta(\mathbf{x}) \quad (\text{A.20})$$

Ainsi, si $\beta = (\beta_1, \beta_2)$ avec β_1 impair et β_2 pair, le noyau doit également être impair et pair dans les directions respectives. Dans le cas présent, avec $\beta = 2$, on a alors :

$$\eta^{(2)}(-x) = \eta^{(2)}(x) \quad (\text{A.21})$$

Tous les moments M_α , avec α impair sont alors nuls. Si on choisit un noyau de base gaussien tel que :

$$\eta^{(2)}(x) = \frac{1}{\sqrt{\pi}} \left(\sum_{j=0}^m \gamma_j x^{2j} \right) e^{-x^2} \quad (\text{A.22})$$

Il vérifie effectivement la condition de symétrie (A.20), on a alors :

$$M_1 = \int_{-\infty}^{+\infty} z\eta^{(2)}(z)dz = \int_{-\infty}^{+\infty} z \frac{1}{\sqrt{\pi}} (\gamma_0 + \gamma_1 z^2) e^{-z^2} dz = 0 \quad (\text{A.23})$$

et

$$M_3 = \int_{-\infty}^{+\infty} z^3\eta^{(2)}(z)dz = \int_{-\infty}^{+\infty} z^3 \frac{1}{\sqrt{\pi}} (\gamma_0 + \gamma_1 z^2) e^{-z^2} dz = 0 \quad (\text{A.24})$$

Il ne reste qu'à satisfaire $M_2 = 2$ et $M_4 = 0$. Les calculs détaillés sont expliqués dans l'annexe A. On en déduit un système de deux équations à deux inconnues :

$$\begin{cases} 2\gamma_0 + 3\gamma_1 = 8 \\ 2\gamma_0 + 5\gamma_1 = 0 \end{cases} \quad (\text{A.25})$$



La solution nous donne alors la forme complète du noyau :

$$\eta^{(2)}(x) = \frac{1}{\sqrt{\pi}}(\gamma_0 + \gamma_1 x^2)e^{-x^2} = \frac{1}{\sqrt{\pi}}(10 - 4x^2)e^{-x^2} \quad (\text{A.26})$$

A.4.2 Noyaux B-Splines

Dans cette partie, nous nous intéressons à la construction d'une nouvelle famille de noyaux d'interpolation pour la méthode PSE. Les noyaux présentés par les auteurs de [65] sont des noyaux gaussiens qui ont l'avantage d'être très réguliers et de fournir des résultats très précis à l'interpolation. Leur calcul demande cependant plus de temps que celui de fonctions polynomiales. Par ailleurs, le support de ces noyaux n'étant pas compact, chaque approximation pour une particule donnée demande le calcul d'une somme sur l'ensemble de toutes les particules présentes. Cela conduit à un algorithme général de complexité de l'ordre de N^2 si N est le nombre de particules. Les fonctions B-Splines sont des fonctions polynomiales par morceaux définies à l'aide de points d'interpolation. Une fonction B-Splines de degré n peut être construite de manière à être de classe C^{n-1} en tout point de son domaine de définition, en particulier au niveau de ses points d'interpolation [152]. Ainsi, on peut définir un noyau d'interpolation de la manière suivante :

$$\eta(x) = \begin{cases} b_1(R_1 - x)^3 + b_2(R_2 - x)^3 & 0 \leq x < R_1 \\ b_2(R_2 - x)^3 & R_1 \leq x < R_2 \\ 0 & R_2 \leq x \end{cases} \quad (\text{A.27})$$

où R_1 et R_2 sont les points d'interpolation et b_1 et b_2 sont des paramètres à définir en fonction des propriétés à respecter pour $\eta(\cdot)$. Ce noyau de classe C^2 , à deux morceaux, a un support compact de rayon R_2 centré en 0 (cas 1D).

Nous avons vu précédemment une forme générique de noyaux gaussiens, utilisable comme point de départ à la construction de noyaux pour l'approximation de dérivées. La démarche est identique pour la mise au point de noyaux B-Splines adaptés à PSE. Le noyau est construit de manière à satisfaire les conditions établies sur les moments M_α . Nous avons vu également qu'en raison des propriétés de symétrie des noyaux gaussiens, il suffisait de $r/2$ coefficients pour une approximation d'un opérateur à l'ordre r . Dans le cas des fonctions B-Splines, ces propriétés de symétrie ne sont pas naturelles, il faut donc les imposer. Prenons l'exemple d'un noyau construit pour l'approximation de la dérivée première en dimension 1. Ce noyau doit être impair, on peut donc partir de la forme (A.27), paire, en la modifiant pour obtenir la fonction B-splines de degré

$n :$

$$\eta(x) = x \left\{ \begin{array}{llll} b_1(R_1 - |x|)^n + & b_2(R_2 - |x|)^n + & \cdots & +b_m(R_m - |x|)^n & 0 \leq |x| < R_1 \\ & b_2(R_2 - |x|)^n + & \cdots & +b_m(R_m - |x|)^n & R_1 \leq |x| < R_2 \\ & & \ddots & \vdots & \vdots \\ & & & b_m(R_m - |x|)^n & R_{m-1} \leq |x| < R_m \\ & & & 0 & R_m \leq |x| \end{array} \right. \quad (\text{A.28})$$

En ajoutant x en facteur, le noyau devient impair. Dans le cas de noyaux destinés à des problèmes 2D ou 3D, le noyau peut-être rendu impair dans la direction x_i voulue :

$$\eta(\mathbf{x}) = x_i \left\{ \begin{array}{llll} b_1(R_1 - |\mathbf{x}|)^n + & b_2(R_2 - |\mathbf{x}|)^n + & \cdots & +b_m(R_m - |\mathbf{x}|)^n & 0 \leq |\mathbf{x}| < R_1 \\ & b_2(R_2 - |\mathbf{x}|)^n + & \cdots & +b_m(R_m - |\mathbf{x}|)^n & R_1 \leq |\mathbf{x}| < R_2 \\ & & \ddots & \vdots & \vdots \\ & & & b_m(R_m - |\mathbf{x}|)^n & R_{m-1} \leq |\mathbf{x}| < R_m \\ & & & 0 & R_m \leq |\mathbf{x}| \end{array} \right. \quad (\text{A.29})$$

De même, dans le cas du calcul du laplacien, le noyau doit être pair et de degré identique dans toutes les directions :

$$\eta(\mathbf{x}) = \left\{ \begin{array}{llll} b_1(R_1 - |\mathbf{x}|)^n + & b_2(R_2 - |\mathbf{x}|)^n + & \cdots & +b_m(R_m - |\mathbf{x}|)^n & 0 \leq |\mathbf{x}| < R_1 \\ & b_2(R_2 - |\mathbf{x}|)^n + & \cdots & +b_m(R_m - |\mathbf{x}|)^n & R_1 \leq |\mathbf{x}| < R_2 \\ & & \ddots & \vdots & \vdots \\ & & & b_m(R_m - |\mathbf{x}|)^n & R_{m-1} \leq |\mathbf{x}| < R_m \\ & & & 0 & R_m \leq |\mathbf{x}| \end{array} \right. \quad (\text{A.30})$$

Comme montré précédemment, un noyau à symétrie radiale ne permet pas d'isoler une dérivée seconde partielle. On peut en revanche définir le noyau de la manière suivante :

$$\eta(\mathbf{x}) = x_i^2 \left\{ \begin{array}{llll} b_1(R_1 - |\mathbf{x}|)^n + & b_2(R_2 - |\mathbf{x}|)^n + & \cdots & +b_m(R_m - |\mathbf{x}|)^n & 0 \leq |\mathbf{x}| < R_1 \\ & b_2(R_2 - |\mathbf{x}|)^n + & \cdots & +b_m(R_m - |\mathbf{x}|)^n & R_1 \leq |\mathbf{x}| < R_2 \\ & & \ddots & \vdots & \vdots \\ & & & b_m(R_m - |\mathbf{x}|)^n & R_{m-1} \leq |\mathbf{x}| < R_m \\ & & & 0 & R_m \leq |\mathbf{x}| \end{array} \right. \quad (\text{A.31})$$

Revenons à l'exemple de la dérivée première en dimension 1, la forme générique du noyau à



utiliser est écrite en (A.28). Posons :

$$W_i(x) = b_i(R_i - |x|)^n \quad (\text{A.32})$$

On a alors :

$$\eta(x) = x \begin{cases} W_1(x) + W_2(x) + \cdots + W_m(x) & 0 \leq |x| < R_1 \\ W_2(x) + \cdots + W_m(x) & R_1 \leq |x| < R_2 \\ \vdots & \vdots \\ W_m(x) & R_{m-1} \leq |x| < R_m \\ 0 & R_m \leq |x| \end{cases} \quad (\text{A.33})$$

Rappelons la définition du moment d'ordre α :

$$M_\alpha = \int_{-\infty}^{+\infty} z^\alpha \eta(z) dz \quad (\text{A.34})$$

On a alors :

$$\begin{aligned} M_\alpha &= \int_{-R_m}^{-R_{m-1}} z^{\alpha+1} W_m(z) dz + \cdots + \int_{-R_1}^0 \sum_{i=1}^m z^{\alpha+1} W_i(z) dz \\ &+ \int_0^{R_1} \sum_{i=1}^m z^{\alpha+1} W_i(z) dz + \cdots + \int_{R_{m-1}}^{R_m} z^{\alpha+1} W_m(z) dz \end{aligned} \quad (\text{A.35})$$

Par linéarité de l'intégrale, on obtient :

$$M_\alpha = \sum_{i=1}^m \left(\int_{-R_i}^0 z^{\alpha+1} W_i(z) dz + \int_0^{R_i} z^{\alpha+1} W_i(z) dz \right) \quad (\text{A.36})$$

$W_i(z)$ étant une fonction paire, on peut déjà vérifier que les moments d'ordre pair sont nuls. Dans le cas où α est impair, on a :

$$M_\alpha = 2 \sum_{i=1}^m \int_0^{R_i} z^{\alpha+1} W_i(z) dz \quad (\text{A.37})$$

En remplaçant $W_i(z)$ par sa définition, on obtient :

$$M_\alpha = 2 \sum_{i=1}^m \int_0^{R_i} z^{\alpha+1} b_i (R_i - |z|)^n dz \quad (\text{A.38})$$

ou encore :

$$M_\alpha = 2 \sum_{i=1}^m b_i \int_0^{R_i} z^{\alpha+1} \sum_{j=0}^n \binom{n}{j} R_i^{n-j} (-|z|)^j dz \quad (\text{A.39})$$

avec $\binom{n}{j} = C_n^j = \frac{n!}{(n-j)!j!}$. Appliquons à nouveau le principe de linéarité de l'intégrale :

$$M_\alpha = 2 \sum_{i=1}^m b_i \sum_{j=0}^n \binom{n}{j} (-1)^{j+1} \int_0^{R_i} z^{\alpha+j+1} R_i^{n-j} dz \quad (\text{A.40})$$

En calculant l'intégrale, on obtient finalement :

$$M_\alpha = 2 \sum_{i=1}^m b_i \sum_{j=0}^n \binom{n}{j} (-1)^{j+1} \frac{R_i^{n+\alpha+2}}{\alpha+j+2} \quad (\text{A.41})$$

On a alors la forme générale des moments dans le cas du calcul de dérivées premières. Il reste à définir le nombre de morceaux nécessaires et le degré du polynôme. Si on souhaite mettre en œuvre une approximation d'ordre r , il suffit de satisfaire $\frac{r}{2}$ conditions sur les moments d'ordre impair. Le noyau final a donc, au moins $\frac{r}{2}$, morceaux. Il faut également choisir le degré n du polynôme selon l'ordre d'approximation choisi. On augmente ainsi la régularité du noyau avec l'ordre d'approximation. Il est important de noter que si le noyau est de degré n , chaque morceau est de la forme :

$$W_i(x) = b_i (R_i - |x|)^n = b_i \sum_{j=0}^n \binom{n}{j} (-1)^j R_i^{n-j} |x|^j \quad (\text{A.42})$$

Le noyau présente donc des monômes de degré impair qui peuvent poser des problèmes de continuité en 0. On choisit donc d'annuler ces monômes en 0, i.e. pour le premier morceau du noyau. On obtient alors de nouvelles conditions dont le nombre augmente avec le degré du noyau. Ainsi, pour un noyau de degré 3, on a deux équations supplémentaires :

$$\sum_{i=1}^m b_i R_i^{n-1} = 0 \quad (\text{A.43})$$

et

$$\sum_{i=1}^m b_i R_i^{n-3} = 0 \quad (\text{A.44})$$

Le tableau, ci-dessous, montre le degré des B-Splines et le nombre de conditions supplémentaires pour un ordre d'approximation donné :

Pour l'approximation de la dérivée première en 1D à l'ordre 2, on a donc une seule condition sur les moments :

$$2b_2 + 24b_3 = -1 \quad (\text{A.45})$$



Ordre r	Conditions sur les moments M_α	Degré n des $W_i(x)$	Conditions supplémentaires	Nombre total d'équations
2	1	3	2	3
4	2	4	2	4
6	3	5	3	6
8	4	6	3	7

TABLE A.1 – Paramètres de construction des noyaux B-splines

ainsi que deux équations en 0 :

$$b_1 + 4b_2 + 9b_3 = 0 \quad (\text{A.46})$$

et

$$b_1 + b_2 + b_3 = 0 \quad (\text{A.47})$$

Après résolution, on a le noyau suivant :

$$\eta(x) = x \begin{cases} -\frac{5}{56}(1 - |x|)^3 + \frac{1}{7}(2 - |x|)^3 - \frac{3}{56}(3 - |x|)^3 & 0 \leq |x| < 1 \\ \frac{1}{7}(2 - |x|)^3 - \frac{2}{56}(3 - |x|)^3 & 1 \leq |x| < 2 \\ -\frac{3}{56}(3 - |x|)^3 & 2 \leq |x| < 3 \\ 0 & 3 \leq |x| \end{cases} \quad (\text{A.48})$$

La démarche à suivre est identique pour les dérivées secondes. On peut également étendre cette technique pour la construction de noyaux pour les problèmes à 2 ou 3 dimensions. L'expression des moments est alors modifiée mais les conditions de raccord en 0 restent les mêmes.

Annexe B

Correction de l'instabilité en tension

Au cours de notre étude de l'étirement d'une goutte en deux dimensions (§4), nous avons mentionné une méthode de correction d'instabilité en tension, souvent rencontré en méthode SPH et qui a fait l'objet de nombreux travaux de recherche. Cette annexe reprend un cas test proposé par Colagrossi [35], où cette instabilité est particulièrement présente. Nous comparons également l'effet des corrections existantes sur les différents modèles étudiés au cours du chapitre 4.

B.1 Présentation du cas test

Pour observer brièvement les effets de cette force artificielle, une variante du cas étudié dans ce chapitre, proposée par Colagrossi [35], reprise par Oger *et al.* [173] et Fang *et al.* [70], a été simulée. Dans ce cas la goutte prend une forme carrée, et l'initialisation des vitesses est la suivante :

$$\begin{cases} u_0(x, y) = \omega y \\ v_0(x, y) = -\omega x \end{cases}, \quad (\text{B.1})$$

ω est le rotationnel de la vitesse, constant à l'intérieur du carré de côté L et nul à l'extérieur. La pression associée à cette distribution de vitesse initiale est la suivante :

$$p_0(x, y) = -\rho \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \frac{32\omega^2}{(2m+1)(2n+1)\pi^2} \frac{\sin\left(\frac{(2m+1)(x+\frac{L}{2})\pi}{L}\right) \sin\left(\frac{(2n+1)(y+\frac{L}{2})\pi}{L}\right)}{\left(\frac{(2n+1)\pi}{L}\right)^2 + \left(\frac{(2m+1)\pi}{L}\right)^2} \quad (\text{B.2})$$

Cette série converge très rapidement, quelques termes ($3 \sim 5$) suffisent à obtenir une approximation précise. La figure (B.1) montre le champ de particules initial pour cet écoulement. Le carré entre en rotation et se déforme pour former quatre bras de fluide issus des coins initiaux.



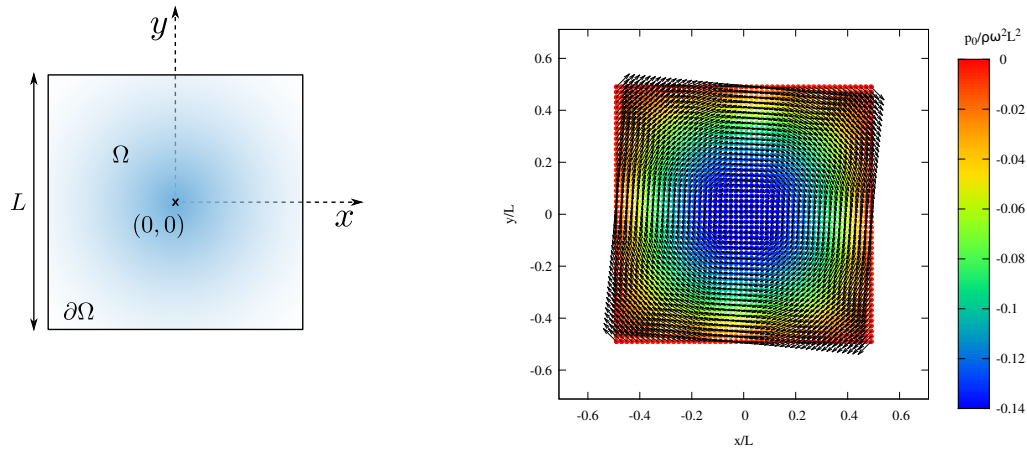


FIGURE B.1 – Descriptions et notations pour l'étirement d'une tourbillon carré en domaine ouvert (gauche). Configuration initiale des particules pour la simulation, les vecteurs symbolisent la vitesse initiale associée à la particule colorée en fonction de la pression (droite).

B.2 Comparaison des modèles de Colagrossi [35] et de Ferrari *et al.* [75]

Les simulations ont été réalisées avec une grille de 50×50 particules, la vitesse du son a été fixée à $c_0 = 7\omega L$. L'intégration par schéma prédicteur-correcteur a été réalisée avec $\Delta t\omega = 10^{-3}$. La figure (B.2) montre l'évolution du domaine fluide sans correction de l'instabilité en tension ni réinitialisation de la masse volumique par MLS (seules la vitesse XSPH et la viscosité artificielle sont utilisées). On observe l'apparition de petits lots de particules dès $t\omega = 1, 2$. Si la réinitialisation MLS de la densité est ajoutée, on obtient les résultats représentés par la figure (B.3). Les premiers lots de particules apparus durant la simulation précédente ont disparu ; en revanche, les effets de l'instabilité sont toujours observables à $t\omega = 2$. Enfin, si la force répulsive est ajoutée, alors la répartition des particules reste homogène jusqu'à $t\omega = 2$ (figure B.4). D'autres instabilités apparaissent ensuite, mais dues au manque de particules dans certaines zones du domaine fluide.

La figure (B.5) montre que le schéma *Rusanov*-SPH est également sensible à l'instabilité en tension. Cela s'explique par le fait que l'équation de conservation de la quantité de mouvement ne subit aucune modification par rapport à l'équation du schéma standard, ainsi aucune viscosité artificielle ne permet de réduire l'instabilité. L'ajout de la force répulsive (équation (4.21)) permet d'en limiter un peu les effets (figure B.5), mais pas autant que dans le cas du modèle de Colagrossi (figure B.4). La suppression de cette instabilité, peu observée dans les exemples des chapitres suivants, n'a pas été déterminante dans le choix d'une formulation.

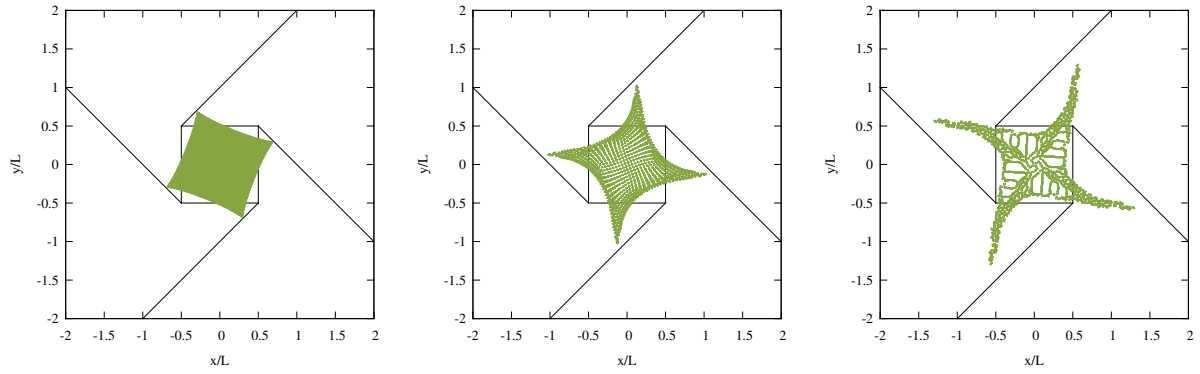


FIGURE B.2 – Évolution du carré fluide sans correction de l'instabilité en tension, ni réinitialisation MLS. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0, 4$, $t\omega = 1, 2$ et $t\omega = 2, 0$.

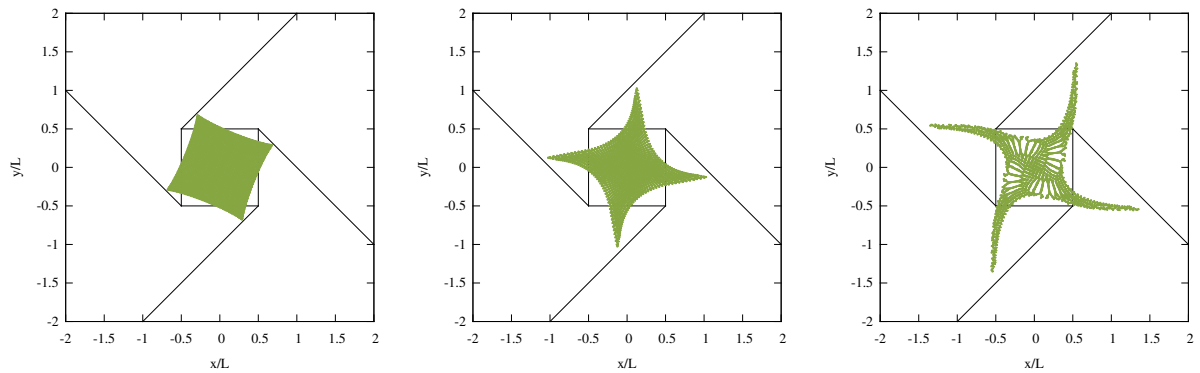


FIGURE B.3 – Évolution du carré fluide avec réinitialisation MLS mais sans correction de l'instabilité en tension. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0, 4$, $t\omega = 1, 2$ et $t\omega = 2, 0$.

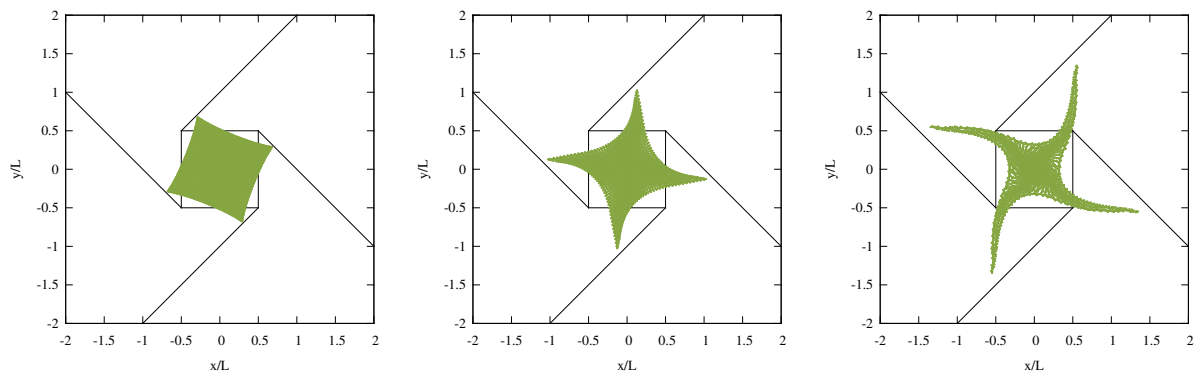


FIGURE B.4 – Évolution du carré fluide avec correction de l'instabilité en tension et réinitialisation MLS. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0, 4$, $t\omega = 1, 2$ et $t\omega = 2, 0$.



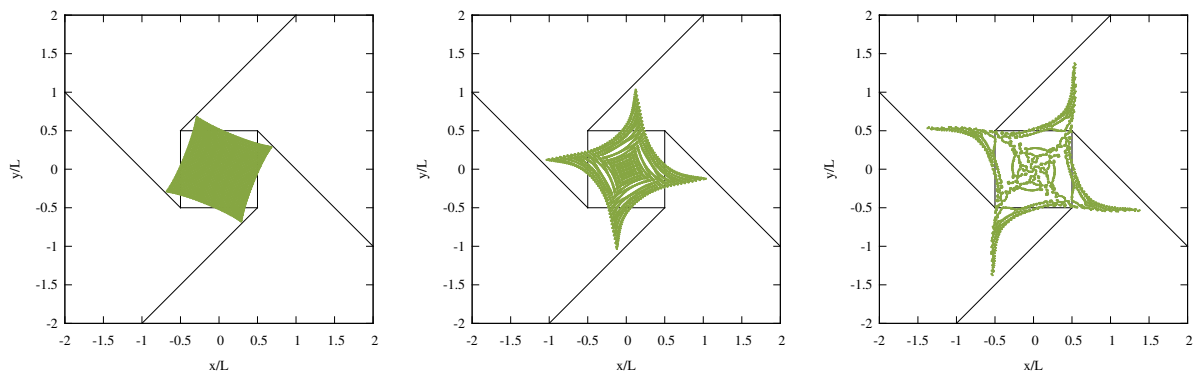


FIGURE B.5 – Évolution du carré fluide avec le flux Rusanov sans correction de l'instabilité en tension. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0,4$, $t\omega = 1,2$ et $t\omega = 2,0$.

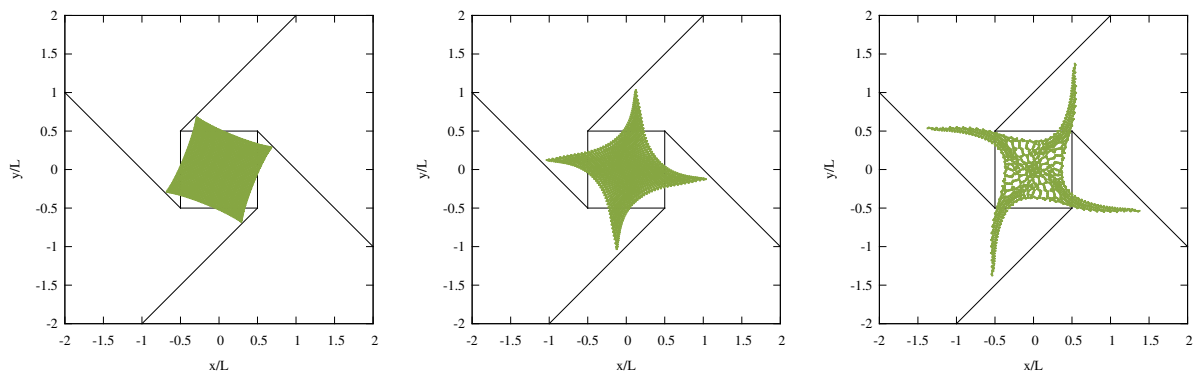


FIGURE B.6 – Évolution du carré fluide avec le flux Rusanov et la correction de l'instabilité en tension. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0,4$, $t\omega = 1,2$ et $t\omega = 2,0$.

Annexe C

Intégration en temps

L'intégration en temps des équations de Navier-Stokes (ou Euler) en méthode SPH est un point très souvent abordé dans les différents travaux (articles ou thèses) relatifs à la méthode. De nombreux schémas différents sont employés mais assez peu de détails techniques sont disponibles sur l'écriture de cette intégration en langage informatique. Ainsi, cette annexe est consacrée à cet aspect fondamental de la simulation.

C.1 Variables d'intégration

Quelque soit le modèle à faible compressibilité employé, les variables essentielles associées aux particules sont :

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{u} \\ \rho \end{pmatrix} \quad (\text{C.1})$$

La masse des particules étant constante au cours du temps, leur volume est systématiquement déduit du rapport $\frac{m}{\rho}$. De même, la pression est calculée à l'aide de la loi d'état en ne faisant intervenir aucune dérivée temporelle. Par la nature lagrangienne de la méthode, la position \mathbf{x} des particules doit être intégrée.

Souvent, en méthodes particulières, différents schémas d'intégration sont employés pour traiter chacune des équations. Ainsi, il arrive que les positions des particules soient intégrées par un simple schéma d'Euler d'ordre 1 tandis que les équations de conservation de la masse et de la quantité de mouvement le sont par un schéma d'ordre supérieur. C'est l'approche qui a été adoptée dans les premières versions du code que nous avons développé. Nous avons finalement choisi d'utiliser un schéma identique à l'intégration des trois variables de l'équation (C.1), pour les raisons suivantes :

- Pour les applications envisagées, où la dynamique de l'écoulement peut être complexe, la trajectoire des particules requiert un soin particulier.



- L'approximation des dérivées spatiales (gradient de pression, divergence de la vitesse...) dans les équations de conservation dépend directement des positions des particules. Si celles-ci sont intégrées à l'aide d'un schéma à plusieurs sous-itérations, alors que la position ne change à aucun d'entre eux, le gain espéré par l'emploi d'un schéma d'ordre supérieur ne peut pas être atteint.

C.2 Schémas

Ainsi, l'ensemble des schémas est basé sur la même fonction de sous-itération Ψ , qui fournit les dérivées temporelles adéquates sur l'intervalle de temps souhaité :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix} = \Psi \begin{pmatrix} \mathbf{x} \\ \mathbf{u} \\ \rho \\ t_{\text{initial}} \\ t_{\text{final}} \end{pmatrix} \quad (\text{C.2})$$

Cette fonction Ψ se déroule ainsi :

1. Placement des marqueurs de frontière immergée (uniquement dans le cas de corps en mouvement, dépendant alors de t_{initial}).
2. Placement des particules fantômes (dépendant de t_{final} également si des parois mobiles sont modélisées).
3. Détection des interactions (entre particules puis entre particules et marqueurs de frontière immergées).
4. Calcul du noyau et de sa dérivée.
5. Renormalisation (si activé).
6. Calcul du second membre des équations de Navier-Stokes (\mathbf{rhs}_i , pour la quantité de mouvement et rus_i , pour la masse (avec flux Rusanov)).
7. Calcul des forces de frontière immergée (si activé) \mathbf{f}_i dépendant de t_{initial} et t_{final} , pour le calcul de la force à imposer.

A la fin de ces étapes, on fixe :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix} = \begin{pmatrix} \mathbf{u} \\ \mathbf{rhs}_i + \mathbf{f}_i \\ rus_i \end{pmatrix} \quad (\text{C.3})$$

Considérons maintenant les différents schémas d'intégration en temps qui ont été employés au cours de cette thèse. On se place à l'instant t_n où les variables \mathbf{x}_n , \mathbf{u}_n et ρ_n sont connues. Le pas de temps est constant et vaut Δt .

Euler d'ordre 1

Ce schéma se déroule en une seule sous-itération :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix} = \Psi \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \\ t_n \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.4})$$

On intègre finalement les variables :

$$\begin{pmatrix} \mathbf{x}_{n+1} \\ \mathbf{u}_{n+1} \\ \rho_{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix} \times \Delta t \quad (\text{C.5})$$

Prédicteur/correcteur

Le schéma prédicteur-correcteur se décompose en deux sous-itérations. Une première, comme son nom l'indique, de prédiction :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 = \Psi \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \\ t_n \\ t_n + \frac{\Delta t}{2} \end{pmatrix} \quad (\text{C.6})$$

et

$$\begin{pmatrix} \mathbf{x}_{n+1/2} \\ \mathbf{u}_{n+1/2} \\ \rho_{n+1/2} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 \times \frac{\Delta t}{2} \quad (\text{C.7})$$

Suivie d'une étape de correction :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 = \Psi \begin{pmatrix} \mathbf{x}_{n+1/2} \\ \mathbf{u}_{n+1/2} \\ \rho_{n+1/2} \\ t_n + \frac{\Delta t}{2} \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.8})$$



et

$$\begin{pmatrix} \mathbf{x}_{n+1} \\ \mathbf{u}_{n+1} \\ \rho_{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 \times \Delta t \quad (\text{C.9})$$

Runge-Kutta 2

Le schéma Runge-Kutta 2 diffère du précédent par le placement de l'étape intermédiaire en fin d'itération au lieu du milieu :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 = \Psi \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \\ t_n \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.10})$$

et

$$\begin{pmatrix} \mathbf{x}_{n+1/2} \\ \mathbf{u}_{n+1/2} \\ \rho_{n+1/2} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 \times \Delta t \quad (\text{C.11})$$

La deuxième étape se déroule au temps $t_n + \Delta t$, la frontière immergée n'a alors aucun effet :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 = \Psi \begin{pmatrix} \mathbf{x}_{n+1/2} \\ \mathbf{u}_{n+1/2} \\ \rho_{n+1/2} \\ t_n + \Delta t \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.12})$$

et on intègre finalement en effectuant une moyenne des dérivées temporelles :

$$\begin{pmatrix} \mathbf{x}_{n+1} \\ \mathbf{u}_{n+1} \\ \rho_{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \left(\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 \right) \times \frac{\Delta t}{2} \quad (\text{C.13})$$

Runge-Kutta 3 TVD

Le schéma Runge-Kutta 3 TVD (Total Variation Diminishing) se déroule en 3 étapes :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 = \Psi \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \\ t_n \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.14})$$

et

$$\begin{pmatrix} \mathbf{x}_{n+1/4} \\ \mathbf{u}_{n+1/4} \\ \rho_{n+1/4} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 \times \Delta t \quad (\text{C.15})$$

La deuxième étape se déroule au temps $t_n + \frac{\Delta t}{4}$:

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 = \Psi \begin{pmatrix} \mathbf{x}_{n+1/4} \\ \mathbf{u}_{n+1/4} \\ \rho_{n+1/4} \\ t_n + \frac{\Delta t}{4} \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.16})$$

$$\begin{pmatrix} \mathbf{x}_{n+2/3} \\ \mathbf{u}_{n+2/3} \\ \rho_{n+2/3} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 \times \frac{\Delta t}{4} \quad (\text{C.17})$$

La dernière étape se déroule à l'instant $t_n + \frac{2\Delta t}{3}$:

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_3 = \Psi \begin{pmatrix} \mathbf{x}_{n+1/4} \\ \mathbf{u}_{n+1/4} \\ \rho_{n+1/4} \\ t_n + \frac{2\Delta t}{3} \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.18})$$

Intégration finale :

$$\begin{pmatrix} \mathbf{x}_{n+1} \\ \mathbf{u}_{n+1} \\ \rho_{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \left(\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 + 4 \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_3 \right) \times \frac{\Delta t}{6} \quad (\text{C.19})$$



Runge-Kutta 4

Première étape :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 = \Psi \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \\ t_n \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.20})$$

et

$$\begin{pmatrix} \mathbf{x}_{n+1/2} \\ \mathbf{u}_{n+1/2} \\ \rho_{n+1/2} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 \times \frac{\Delta t}{2} \quad (\text{C.21})$$

La deuxième étape se déroule au temps $t_n + \frac{\Delta t}{2}$:

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 = \Psi \begin{pmatrix} \mathbf{x}_{n+1/2} \\ \mathbf{u}_{n+1/2} \\ \rho_{n+1/2} \\ t_n + \frac{\Delta t}{2} \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.22})$$

$$\begin{pmatrix} \mathbf{x}_{n+1/2}^* \\ \mathbf{u}_{n+1/2}^* \\ \rho_{n+1/2}^* \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 \times \frac{\Delta t}{2} \quad (\text{C.23})$$

La troisième étape se déroule toujours à l'instant $t_n + \frac{1\Delta t}{2}$:

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_3 = \Psi \begin{pmatrix} \mathbf{x}_{n+1/2}^* \\ \mathbf{u}_{n+1/2}^* \\ \rho_{n+1/2}^* \\ t_n + \frac{\Delta t}{2} \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.24})$$

$$\begin{pmatrix} \mathbf{x}_{n+1/2}^{**} \\ \mathbf{u}_{n+1/2}^{**} \\ \rho_{n+1/2}^{**} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_3 \times \Delta t \quad (\text{C.25})$$

La dernière étape se déroule à l'instant $t_n + \Delta t$, la frontière immergée n'a donc aucun effet :

$$\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_4 = \Psi \begin{pmatrix} \mathbf{x}_{n+1/2}^{**} \\ \mathbf{u}_{n+1/2}^{**} \\ \rho_{n+1/2}^{**} \\ t_n + \Delta t \\ t_n + \Delta t \end{pmatrix} \quad (\text{C.26})$$

Intégration finale :

$$\begin{pmatrix} \mathbf{x}_{n+1} \\ \mathbf{u}_{n+1} \\ \rho_{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_n \\ \mathbf{u}_n \\ \rho_n \end{pmatrix} + \left(\begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_1 + 2 \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_2 + 2 \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_3 + \begin{pmatrix} \frac{D\mathbf{x}}{Dt} \\ \frac{D\mathbf{u}}{Dt} \\ \frac{D\rho}{Dt} \end{pmatrix}_4 \right) \times \frac{\Delta t}{6} \quad (\text{C.27})$$

C.3 Condition CFL

Tous ces schémas d'intégration ont été testés. Le choix d'un schéma par rapport à un autre dépend grandement de l'application traitée. Leur stabilité est soumise au critère CFL (Courant-Friedrichs-Lewy [47]) qui impose d'avoir :

$$\Delta t \leq \beta \frac{\epsilon}{c_0} \quad (\text{C.28})$$

avec ϵ , la distance de régularisation du noyau et c_0 , la vitesse du son dans la loi d'état. Pour les écoulements considérés dans cette thèse, le paramètre β n'a jamais été pris supérieur à 0,3 pour le schéma prédicteur-correcteur ou le schéma de Runge-Kutta d'ordre 2. La majorité des simulations sont maintenant réalisées à l'aide des schémas de Runge-Kutta d'ordre 3 ou 4 avec un paramètre $\beta < 0,5$. Il existe d'autres formes de conditions CFL souvent employées en SPH, qui font intervenir les effets visqueux [50], particulièrement lorsque la discrétisation en particules est fine, où elle peut devenir plus restrictive que la condition sur la convection (C.28). Ce point est également à l'étude.



Annexe D

Application et étude de l'influence des paramètres géométriques

Le modèle présenté dans cette thèse est adapté à la simulation d'effondrements de colonnes d'eau et de problèmes d'impacts. L'application d'un tel outil à des cas concrets, mettant en œuvre des mécanismes similaires, est donc envisageable. D'ores et déjà, des études, réalisées à l'aide de codes SPH, existent sur le *sloshing* [214, 56] ou les vagues d'étrave [38]. Une autre application directe de ce modèle est l'optimisation des bassins de rétention construits autour de cuves de stockage de fluides (hydrocarbures, produits chimiques...). Souvent, ces bassins contiennent plusieurs cuves de grande capacité. La réglementation en vigueur [244, Art. 29] impose qu'une cuvette de rétention aie une capacité au moins égale à la contenance de la plus grande cuve (ou, selon le type d'hydrocarbures stockées, entre 20% et 50% de la somme des volumes de toutes les cuves du bassin). Cependant, les circonstances d'un éventuel accident et la dynamique de l'écoulement font que ces mesures ne sont pas suffisantes pour contenir la vague qui suit l'effondrement de tels volumes de fluide. Ce fut le cas, par exemple, lors de l'incident du 25 décembre 1988 à l'Étang de Berre [243] (figure D.1), où la rupture d'une cuve de stockage a causé l'effondrement de 2 bacs voisins. Le risque environnemental et humain associé à de telles installations est important.

Un autre exemple récent, parmi les nombreux où la simulation pourrait permettre de prédire les risques encourus, est celui de la coulée de boues rouges qui s'est formée après la rupture d'un bassin de stockage à Ajka en Hongrie, le 4 octobre 2010 [242] (figure D.2). L'impact écologique d'un tel désastre est considérable.

Nous nous sommes intéressés à l'influence que pouvait avoir les dimensions de la colonne d'eau initiale sur la suite de l'écoulement, notamment, sur la forme du jet vertical qui suit l'impact du front sur le mur vertical opposé, dont la position a également été variée (figure 5.1). Sur la base des travaux de Colagrossi [35], nous avons choisi d'étudier l'influence du rapport L/H en maintenant la hauteur de cette colonne, H , constante. Ainsi, la vitesse du front atteignant une valeur maximale théorique de $2\sqrt{gH}$, nous avons également fait varier la distance entre la colonne et le mur opposé, à savoir le rapport $(d - L)/H$. Ainsi, le tableau D.1 montre l'ensemble des



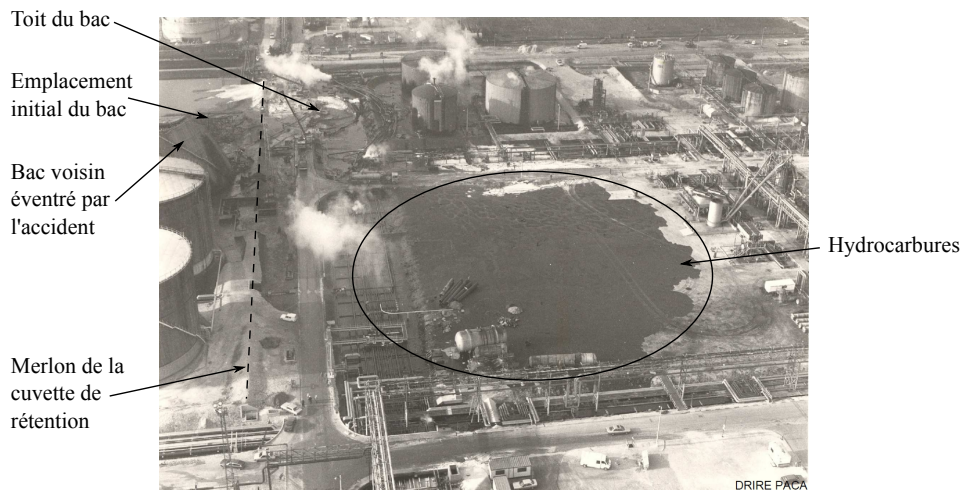


FIGURE D.1 – Rupture d'un bac d'hydrocarbures chauffés, le 25 décembre 1988 à l'Étang de Berre. Vue aérienne des dégâts causés par la vague d'hydrocarbures.



FIGURE D.2 – Marée de boues rouges toxiques issues d'un réservoir de l'usine d'aluminium d'Ajka en Hongrie, le 4 octobre 2010.

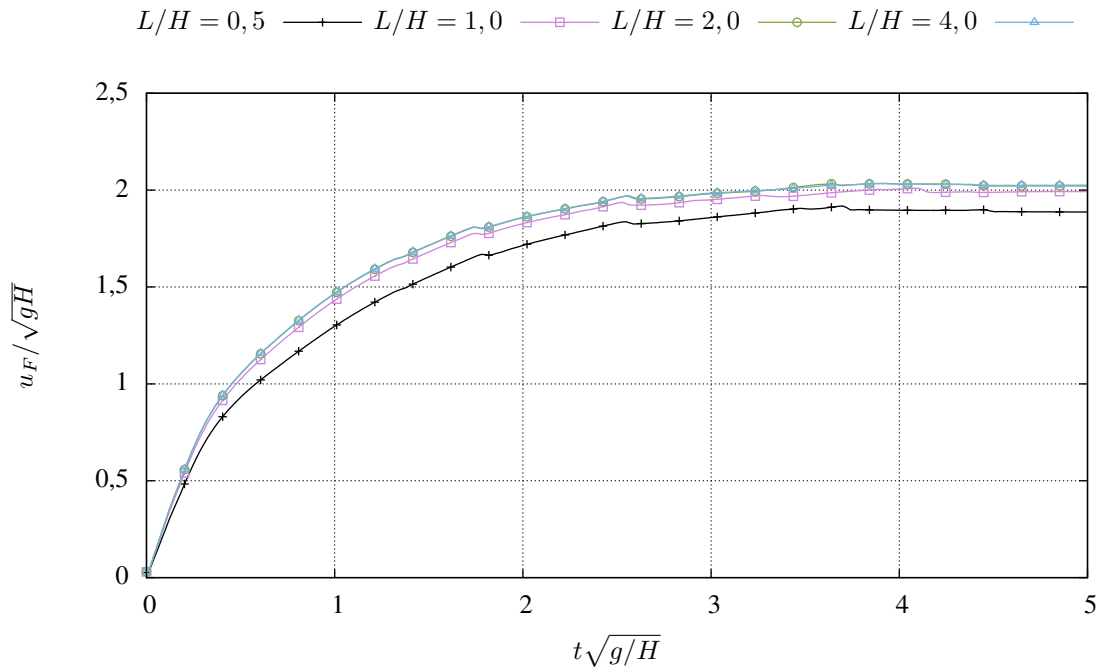


FIGURE D.3 – Évolution de la vitesse du front d'eau en fonction du temps. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu), pour $(d - L)/H = 10$.

simulations qui ont été réalisées pour cette étude.

Bien sur, les calculs effectués ici ne permettent pas de recommander une configuration fiable, tant celle-ci dépend de la rhéologie du fluide. De plus, les simulations ont été réalisées en deux dimensions d'espaces alors qu'il faudrait trois dimensions pour étudier la validité des réglementations mentionnées précédemment [244, Art. 29]. Cependant, cette étude montre une application potentielle et directe d'un tel modèle particulier.

D.1 Influence du rapport L/H sur la vitesse du front

La théorie, vue précédemment, dit que la vitesse maximale du front d'eau vaut $2\sqrt{gH}$ avec H , la hauteur initiale de la colonne d'eau. Nous avons suivi la position du front d'eau pour différentes valeurs du rapport L/H . Seuls les cas où le mur d'impact est le plus éloigné de la colonne (c'est à dire la dernière ligne du tableau D.1) ont été étudiés, car dans les autres configurations, la vitesse maximale ne pouvait pas être atteinte au moment de l'impact. La figure D.3 montre la vitesse de ce front (déduite de la position par différences centrées). Comme attendu, pour les rapport $L/H = 1,0$, $L/H = 2,0$ et $L/H = 4,0$, le front atteint la vitesse théorique maximale de $2\sqrt{gH}$. On observe un légère tendance à l'augmentation de cette vitesse quand le rapport L/H augmente. Pour le rapport $L/H = 0,5$, la vitesse obtenue est inférieure ; dans ce cas précis, l'hypothèse de faible profondeur, établie pour l'estimation de la vitesse maximale, n'est plus respectée.



$\frac{d-L}{H}$	$\frac{L}{H}$			
	0,5	1,0	2,0	4,0
	25 × 50 particules	50 × 50 particules	100 × 50 particules	200 × 50 particules
0,2				
0,5				
1,0				
2,0				
3,0				
4,0				
6,0				
8,0				
10,0				

TABLE D.1 – Représentations graphiques des configurations géométriques simulées, correspondant aux valeurs des paramètres étudiés (figure 5.1).

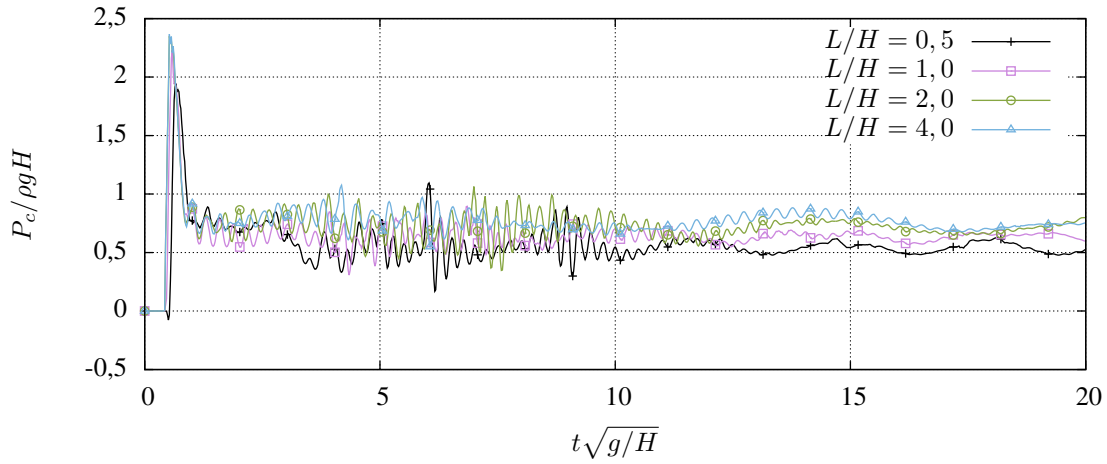
D.2 Pression à l'impact

Les figures D.4 à D.6 montrent la pression mesurée au point $C(d; 0, 192H)$ en fonction du temps. Les courbes ont été regroupées par valeurs du rapport $(d - L)/H$, car la forme de la courbe enregistrée dépend grandement de cette distance au mur. Sur chaque graphique, la pression obtenue pour les quatre valeurs de L/H est représentée. L'interprétation de ces résultats reste délicate et leur précision dépend directement du nombre de particules présentes dans le voisinage du point d'interpolation C . Ainsi, pour de grandes valeurs de $(d - L)/H$ (figure D.6), l'étirement du domaine fluide est tel que seules quelques particules sont présentes dans le support d'interpolation de la pression (un remaillage de cette zone serait utile et entièrement justifié par l'étirement important du champ de particules). Il est donc difficile de conclure sur le lien entre le rapport L/H et le niveau de pression atteint. En effet, respectivement, pour les rapports $L/H = 0,5$, $L/H = 1,0$, $L/H = 2,0$ et $L/H = 4,0$; 2500, 5000, 10000 et 20000 particules ont été employées. Donc, pour les cas où le mur d'impact est le plus éloigné ($(d - L)/H$ grand), la valeur de la pression obtenue gagne en précision quand le ratio la masse fluide initiale est grand. Cela expliquerait que l'écart entre la pression prédite pour chaque rapport d'aspect L/H augmente avec la distance du mur d'impact. Cet écart est effectivement plus réduit quand $(d - L)/H$ est petit (figure D.4). Pour les positions de mur intermédiaires (figure D.5), la discrétisation de la nappe fluide qui atteint le mur est correcte quelque soit le ratio employé; on observe toujours une tendance à l'augmentation de la pression à l'impact quand ce ratio L/H croît. Le *palier* de pression conserve une valeur similaire, pour un ratio donné, quand la distance au mur varie. Cette valeur de palier dépend du ratio L/H et tend vers une valeur asymptotique quand celui-ci augmente. Avec un mur très rapproché ($(d - L)/H$ petit) (figure D.4), ce palier est d'autant plus marqué; la courbe oscille d'ailleurs autour d'une valeur proche de celle qui serait obtenue à l'équilibre statique, à savoir $P_c/\rho g H = 1 - 0,192 = 0,808$. Dans le cas $(d - L)/H = 0,2$ (Fig. D.4-a, pour $t(g/H)^{1/2} > 10$, on distingue nettement deux modes d'oscillation: le premier, le plus rapide, correspond à la réflexion de l'onde acoustique artificielle (dont la célérité est celle issue de la loi d'état); le second, de fréquence plus faible, se rapproche des observations de pression dans des simulations de *sloshing*. Globalement, sur les trois figures D.4 à D.6, on remarque qu'avec l'augmentation du rapport $(d - L)/H$, le pic de pression associé à l'impact sur le mur s'estompe, et la valeur-*palier* est atteinte de façon plus progressive. En revanche, le pic lié au déferlement de la vague qui suit l'impact est plus prononcé (bien que non physique, puisque ce pic est en réalité amorti par l'air des cavités, non modélisé ici).

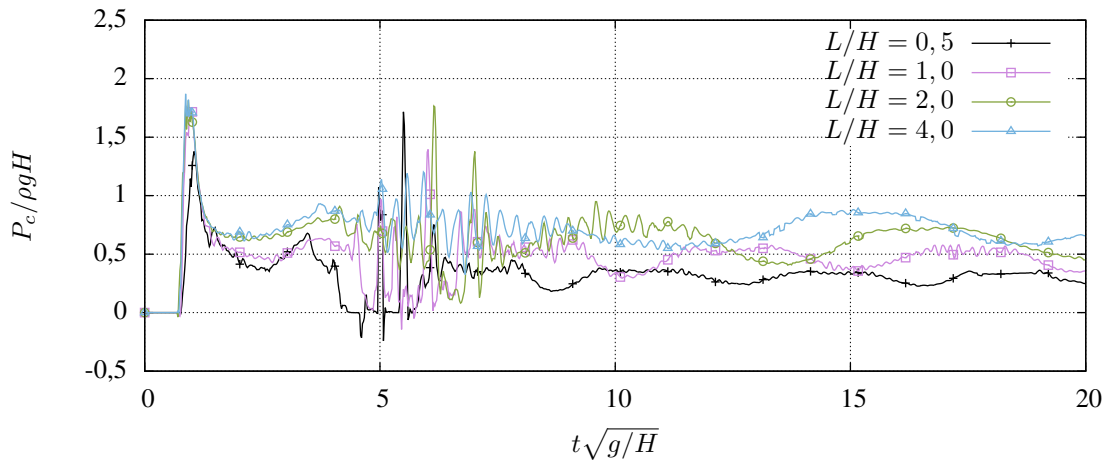
D.3 Débit du jet vertical post-impact

A partir des observations précédentes, il est difficile d'extraire une distance au mur suffisante pour minimiser l'effet de l'impact sur le mur opposé. Afin d'apporter des éléments de réponse à la

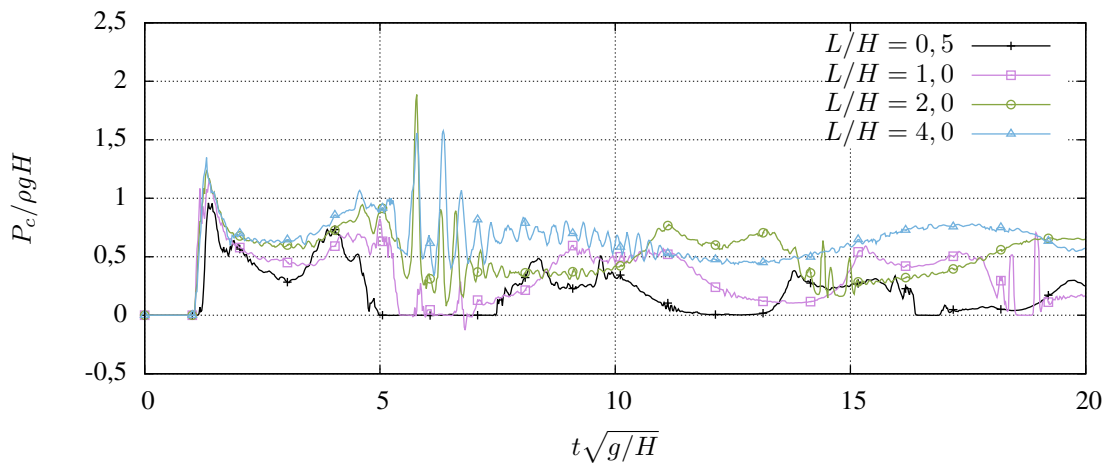




(a) $(d - L)/H = 0,2$

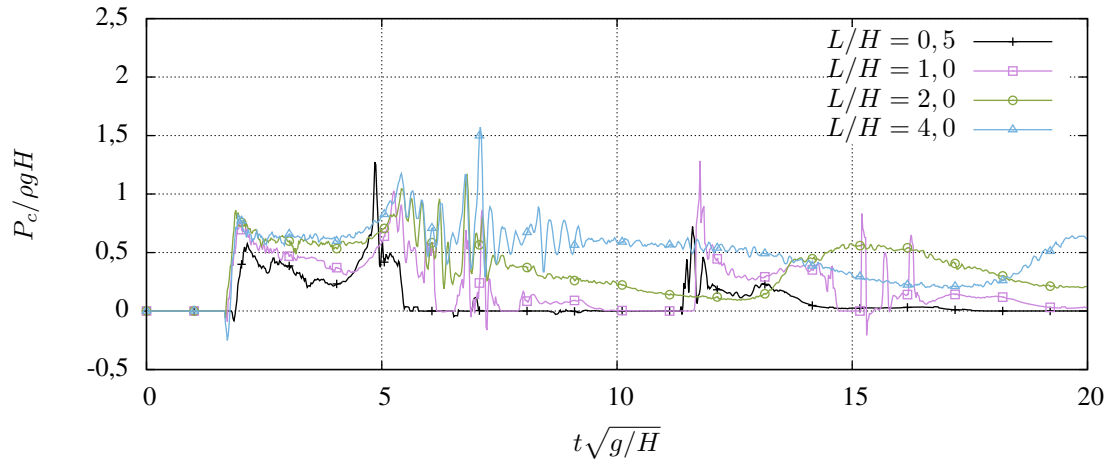


(b) $(d - L)/H = 0,5$

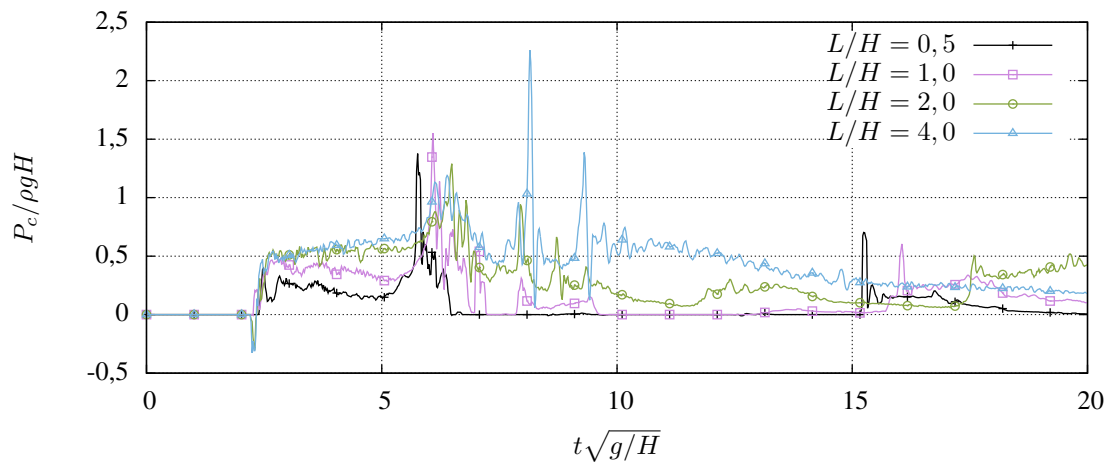


(c) $(d - L)/H = 1,0$

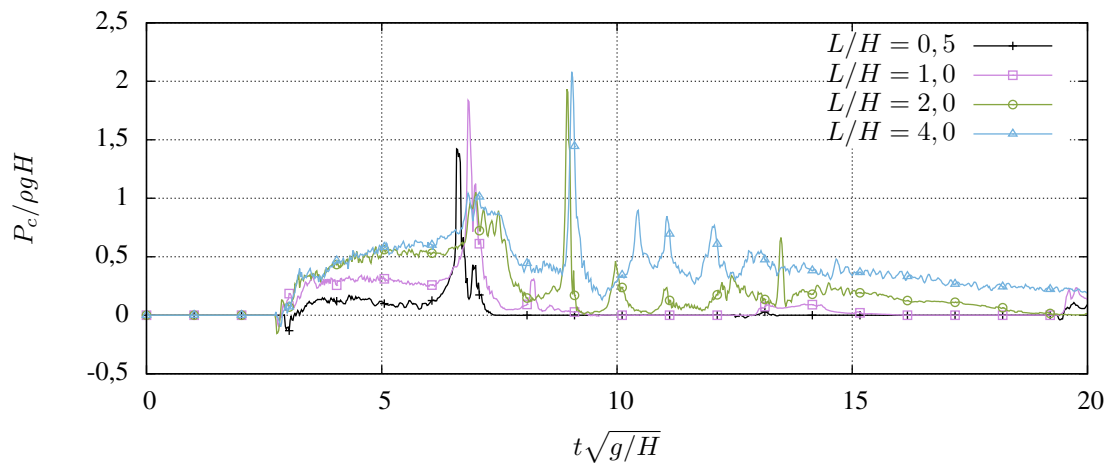
FIGURE D.4 – Évolution de la pression au point $C(d; 0, 192H)$ en fonction du temps pour différentes valeurs de la distance au mur $(d - L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).



(a) $(d - L)/H = 2,0$



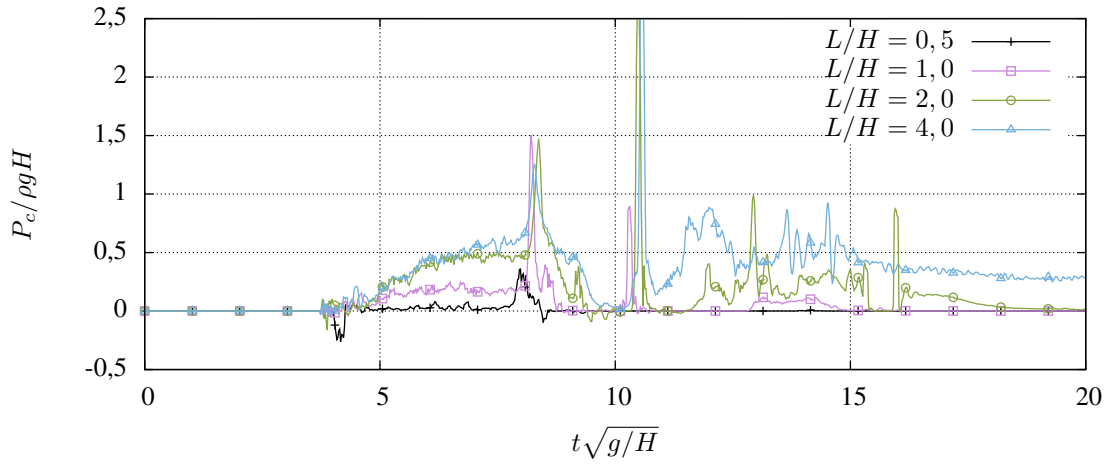
(b) $(d - L)/H = 3,0$



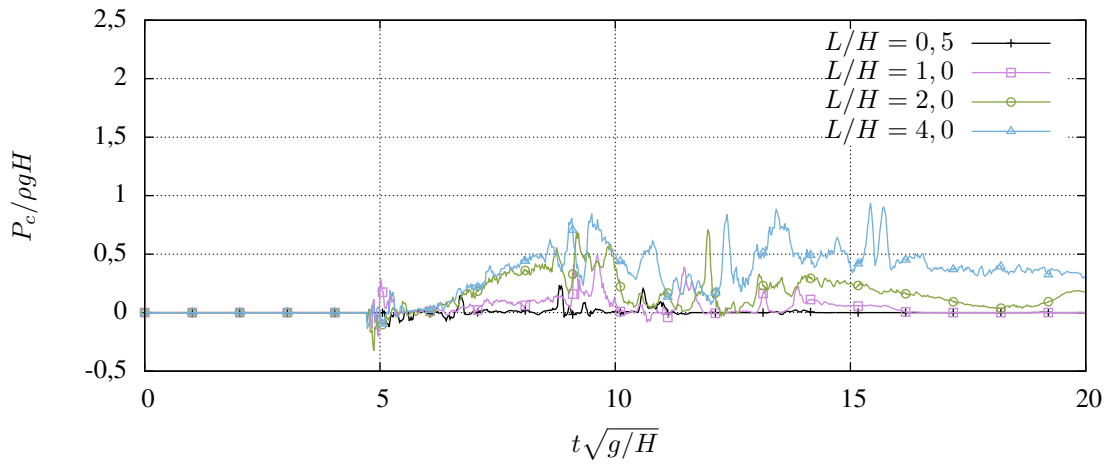
(c) $(d - L)/H = 4,0$

FIGURE D.5 – Évolution de la pression au point $C(d; 0, 192H)$ en fonction du temps pour différentes valeurs de la distance au mur $(d - L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).

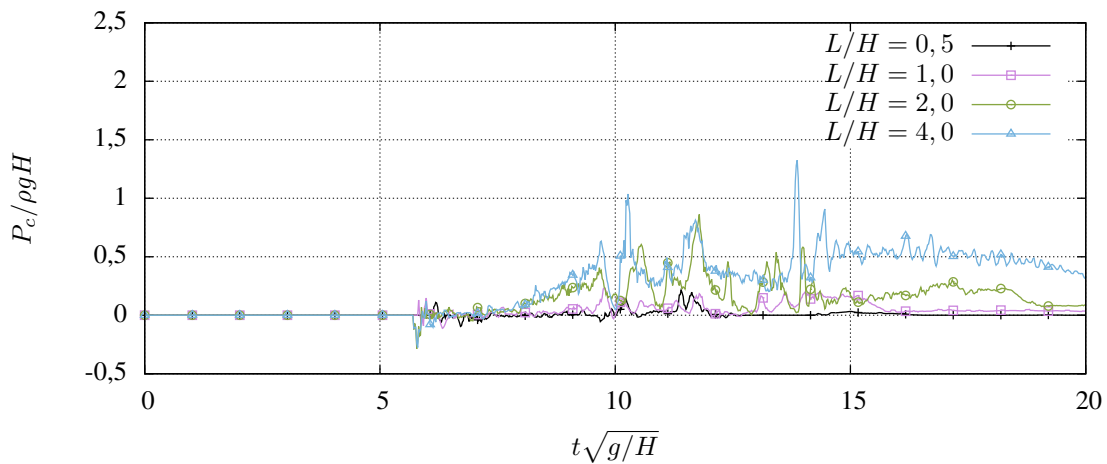




(a) $(d - L)/H = 6,0$



(b) $(d - L)/H = 8,0$



(c) $(d - L)/H = 10,0$

FIGURE D.6 – Évolution de la pression au point $C(d; 0, 192H)$ en fonction du temps pour différentes valeurs de la distance au mur $(d - L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).

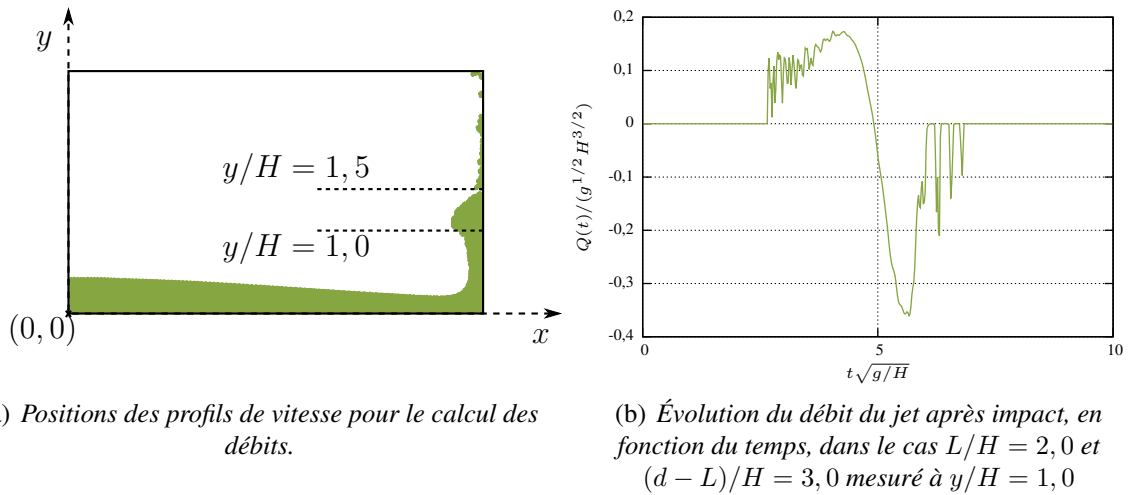


FIGURE D.7 – Calcul des débits sur le mur d’impact.

question du dimensionnement d’un mur de retenue pour un risque minimal de dépassement, nous avons quantifié le débit du fluide le long de cette paroi opposée. Afin de mesurer les débits au niveau du mur après impact, la composante verticale de vitesse du fluide a été interpolée le long de profils horizontaux (figure D.7) à l’aide de l’opérateur habituel :

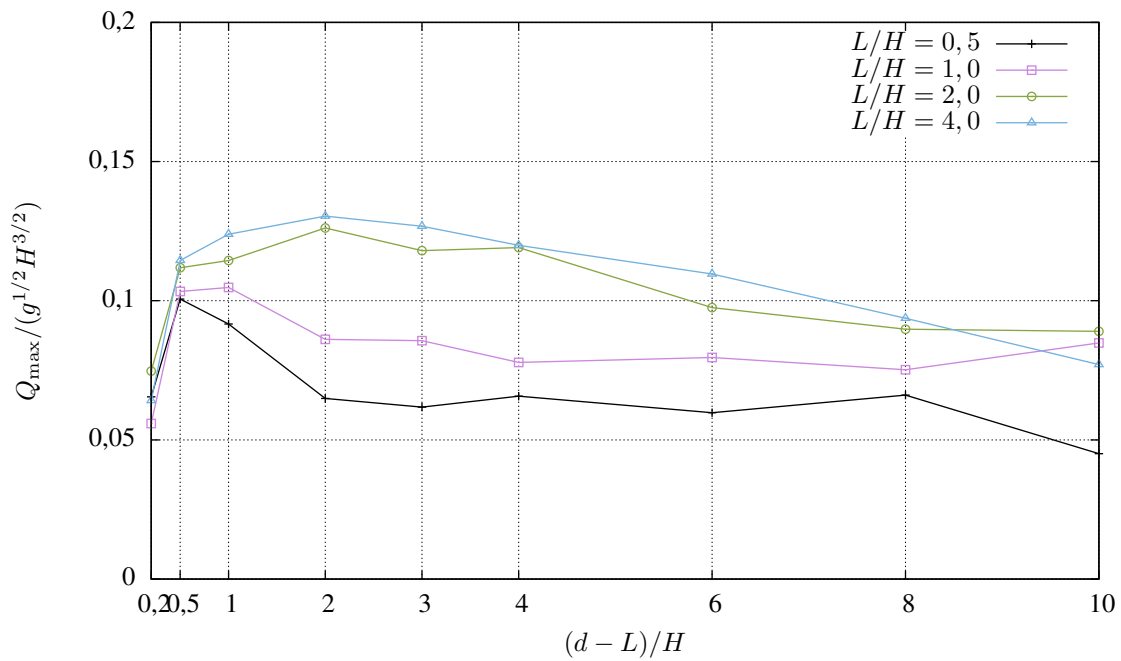
$$v(\mathbf{x}_p) = \frac{\sum_j v_j \eta_\epsilon^0(\mathbf{x}_p - \mathbf{x}_j) V_j}{\sum_j \eta_\epsilon^0(\mathbf{x}_p - \mathbf{x}_j) V_j} \quad (\text{D.1})$$

Ensuite, le débit a été calculé par intégration de la vitesse verticale le long de chaque profil :

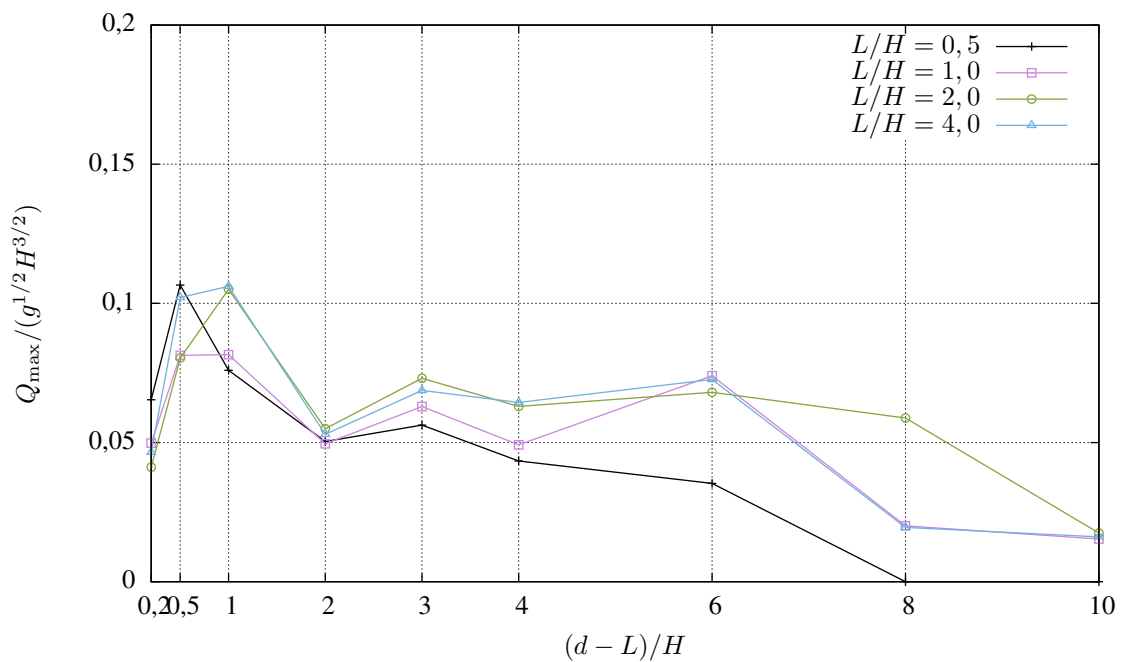
$$Q(y_p, t) = \sum_{x_p} v(x_p, y_p) l_{x_p} \quad (\text{D.2})$$

Ainsi, l’évolution du débit du jet à son intersection avec les profils horizontaux est connue (figure D.7). La courbe obtenue montre d’abord une augmentation du débit puis un changement de signe correspondant respectivement à la montée du jet et à son effondrement. Le débit maximal obtenu a été extrait pour chacune des configurations géométriques étudiées, ainsi que l’instant auquel ce débit maximal a été atteint. La figure D.8 montre le débit maximal obtenu, en fonction de la distance au mur, pour chaque ratio initial de la masse fluide. D’une part, on peut observer qu’entre les profils à $y_{Q_{\max}}/H = 1,0$ et $y_{Q_{\max}}/H = 1,5$, les débits sont significativement réduits, car cet intervalle correspond généralement à celui où le jet se *retourne* pour s’effondrer à nouveau (figure D.7). Si les débits sont mesurés à $y_{Q_{\max}}/H = 1,0$, on peut remarquer que quand le mur est très proche, les débits sont plus faibles (alors que les pressions d’impact étaient les plus violentes (figure D.4)). Les plus forts débits sont atteints pour les valeurs de $(d - L)/H$ intermédiaires ($(d - L)/H = 0,5$ et $1,0$). Ensuite, pour des murs de plus en plus éloignés, les débits maximaux décroissent doucement. En augmentant le rapport L/H , le débit maximal croît également. Dans le



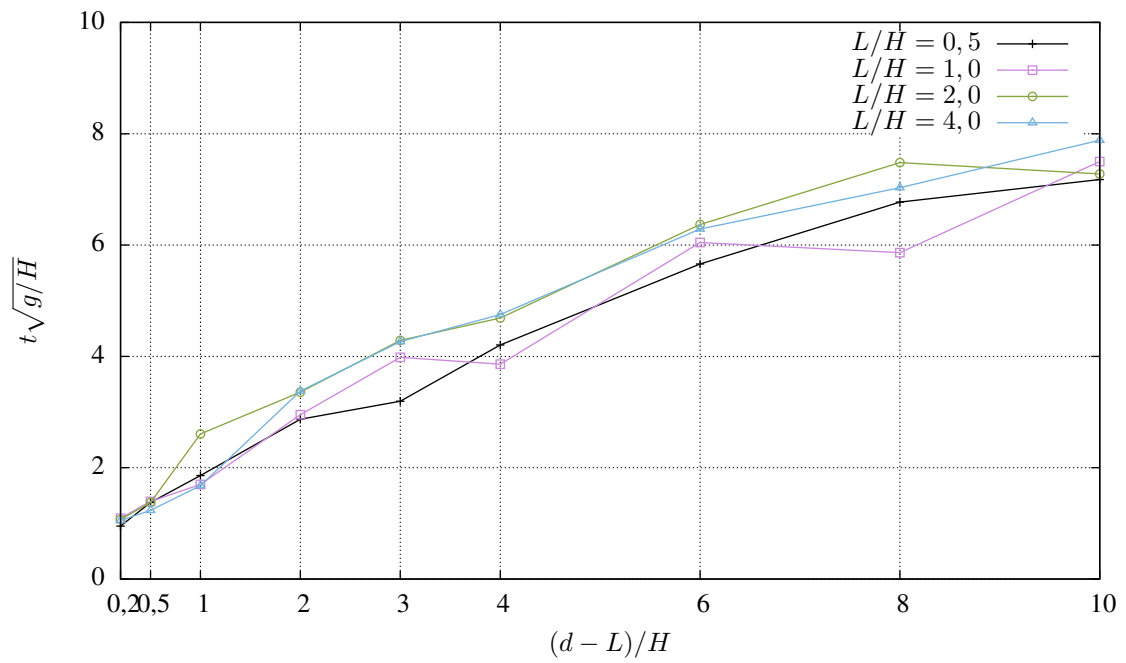


(a) $y_{Q_{\max}}/H = 1,0$

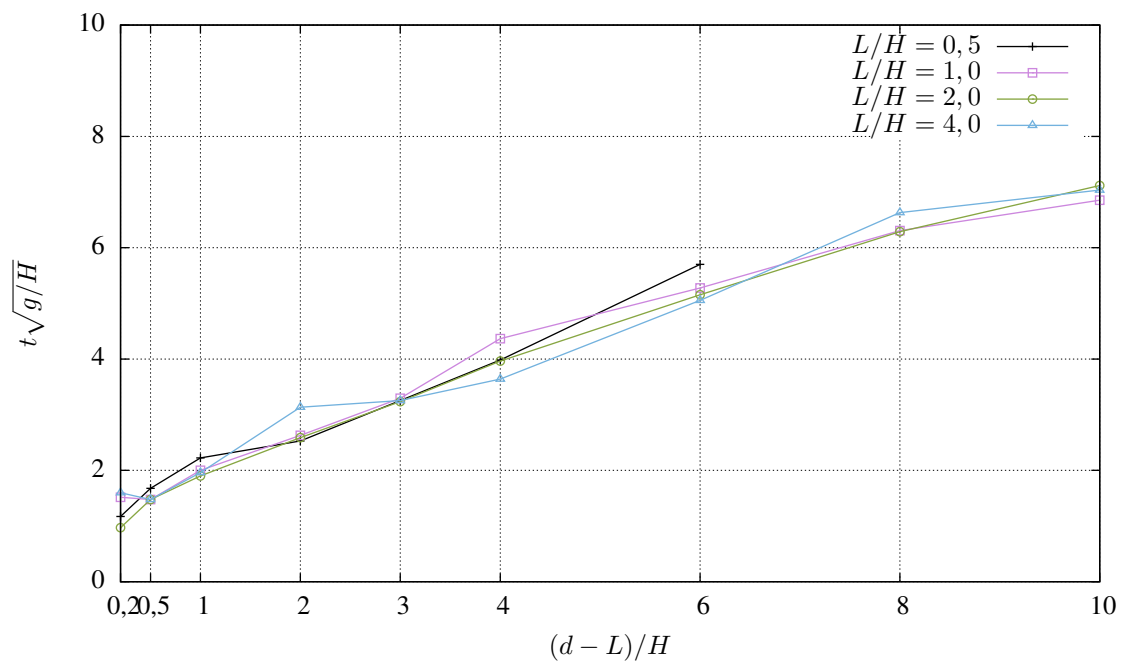


(b) $y_{Q_{\max}}/H = 1,5$

FIGURE D.8 – Débit maximal mesuré à $y_{Q_{\max}}/H = 1,0$ (haut) et $y_{Q_{\max}}/H = 1,5$ (bas) en fonction de la distance au mur $(d-L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).



(a) $y_{Q_{\max}}/H = 1,0$



(b) $y_{Q_{\max}}/H = 1,5$

FIGURE D.9 – Instant ou le débit maximal (mesuré à $y_{Q_{\max}}/H = 1,0$ (haut) et $y_{Q_{\max}}/H = 1,5$ (bas)) est atteint, en fonction de la distance au mur $(d-L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).



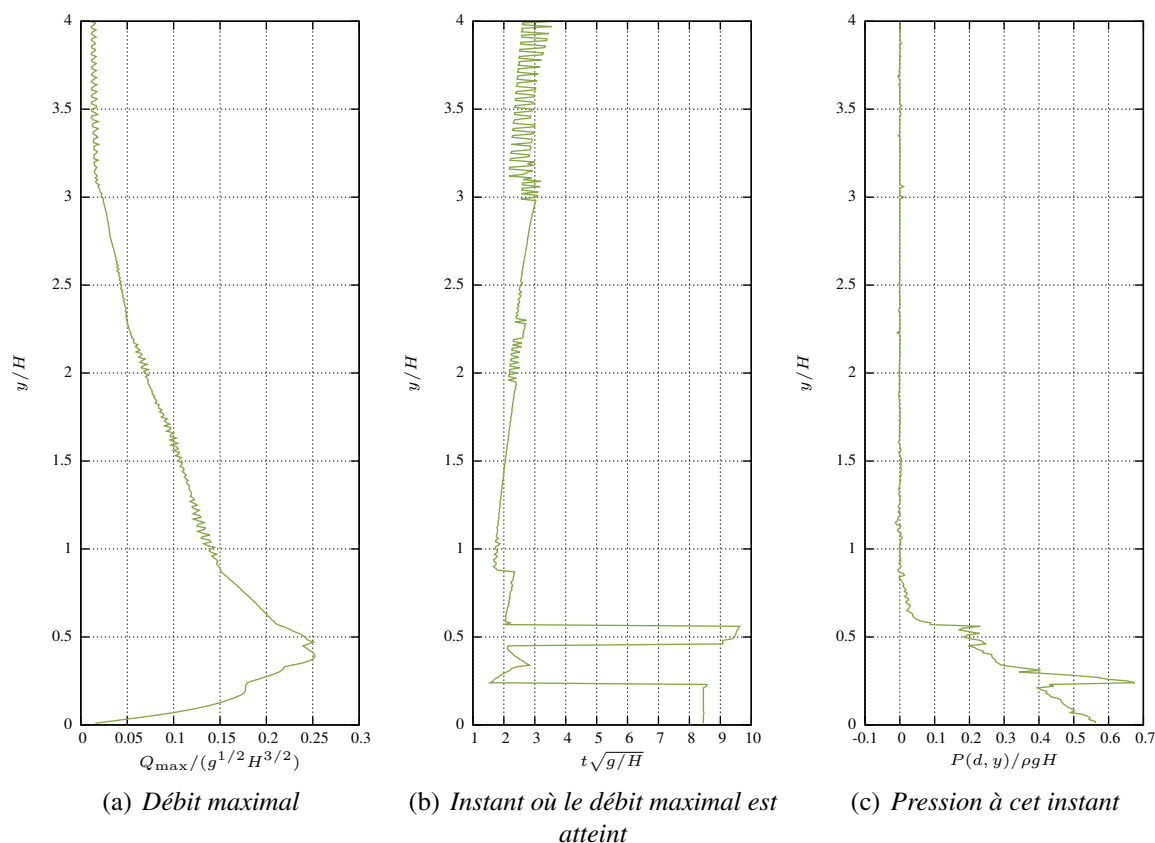


FIGURE D.10 – Cas $L/H = 1,0$ et $(d - L)/H = 1,0$ en deux dimensions d'espace. Évolution du débit maximal, de l'instant où le débit est maximal et de la pression mesurée à cet instant, le long du mur d'impact.

cas $y_{Q_{\max}}/H = 1,5$, les débits les plus forts sont également obtenus à $(d - L)/H = 0,5$ ou $1,0$.

La figure D.9 montre l'instant auquel ce débit maximal est atteint. La tendance linéaire des courbes montre que ce débit est toujours atteint à un instant proche, quelque soit le ratio. Cet instant dépend essentiellement du temps mis par le front à parcourir la distance le séparant du mur d'impact, il est donc cohérent d'observer cette augmentation linéaire. A $y_{Q_{\max}}/H = 1,5$, l'écart entre les instants de débit maximal pour les différents ratios est encore réduit (les valeurs pour la courbe $L/H = 0,5$ à $(d - L)/H > 8$ ne sont pas représentées car les débits y sont nuls).

D.4 Cas $L/H = 1,0$ et $(d - L)/H = 1,0$ en deux dimensions d'espace

Pour confirmer l'observation du paragraphe précédent concernant l'important écart de débit maximal entre les mesures à $y_{Q_{\max}}/H = 1,5$ et $y_{Q_{\max}}/H = 1,0$, nous nous sommes intéressés plus précisément au cas $L/H = 1,0$ et $(d - L)/H = 1,0$. Les débits ont été mesurés en de plus nombreuses coupes horizontales, réparties le long du mur d'impact. La figure D.10(a) montre

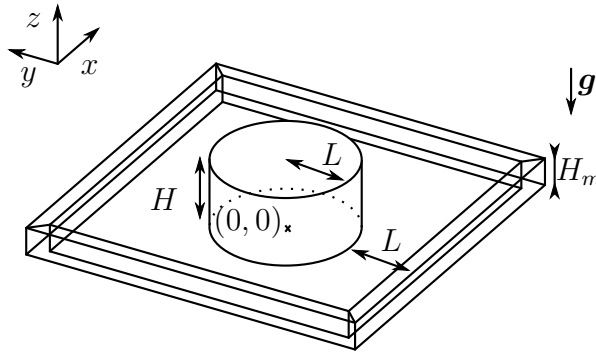


FIGURE D.11 – Configuration initiale de l’effondrement d’une colonne d’eau en 3D. L représente le rayon du cylindre fluide, H est sa hauteur. Un merlon de hauteur H_m est placé à une distance L du cylindre, formant un bassin de rétention de côté $4L$ centré autour du cylindre.

les débits maximaux obtenus en fonction de la hauteur à laquelle ils ont été mesurés. La courbe montre que ce débit est maximal pour $y_{Q_{\max}}/H = 0,5$ et qu’il décroît progressivement quand on élève le point de mesure. Ces débits maximaux ont été mesurés en des instants différents, reportés sur le graphe D.10(b), qui montre que le pic repéré à $y_{Q_{\max}}/H = 0,5$ arrive en fin de simulation contrairement aux valeurs obtenues à hauteur supérieure, qui sont mesurées en des instants qui suivent l’élévation de ce jet le long de la paroi. Nous avons également calculé la pression au mur au moment où le débit maximal a été atteint en chaque point. La figure D.10(c) montre que la pression et les débits maximaux semblent décorrélés. En effet, pour $y_{Q_{\max}}/H > 1,0$, la pression mesurée est nulle. Sous la surface initiale ($y_{Q_{\max}}/H < 1,0$), la pression suit une augmentation quasi-linéaire quand le point de mesure se rapproche du sol.

D.5 Cas $L/H = 1,0$ et $(d - L)/H = 1,0$ à trois dimensions d’espace

Le code de calcul a été étendu aux simulations en trois dimensions d’espace au cours du stage ingénieur de Paul Mycek [167]. Pour illustrer ces travaux, nous avons simulé l’effondrement d’une colonne d’eau (cylindre vertical de section circulaire et de rayon L égal à la hauteur H) dans un bassin de rétention, de forme carrée, délimité par un merlon. Les dimensions de ce merlon ont été calculées en se basant sur la réglementation en vigueur [244, Art. 29]. Les grandeurs ont été définies à $L = 1$ m et $H = 1$ m pour un volume total $V_{\text{fluide}} = \pi L^2 H = \pi$. Le volume du bassin de rétention devant être égal à celui du cylindre de fluide, on a $(4L)^2 H_m = \pi$, soit $H_m = \frac{\pi}{16} \approx 2 \times 10^{-1}$ m. Le volume fluide a été discrétisé par 201 840 placées sur des cercles concentriques autour de l’axe de révolution du cylindre ($\frac{L}{h} = 40$). Les propriétés du fluide sont identiques au cas précédent. L’intégration en temps a été réalisée à l’aide d’un schéma de type Runge-Kutta 3 TVD [75] et un pas de temps $\Delta t = 2 \times 10^{-4}$ s.



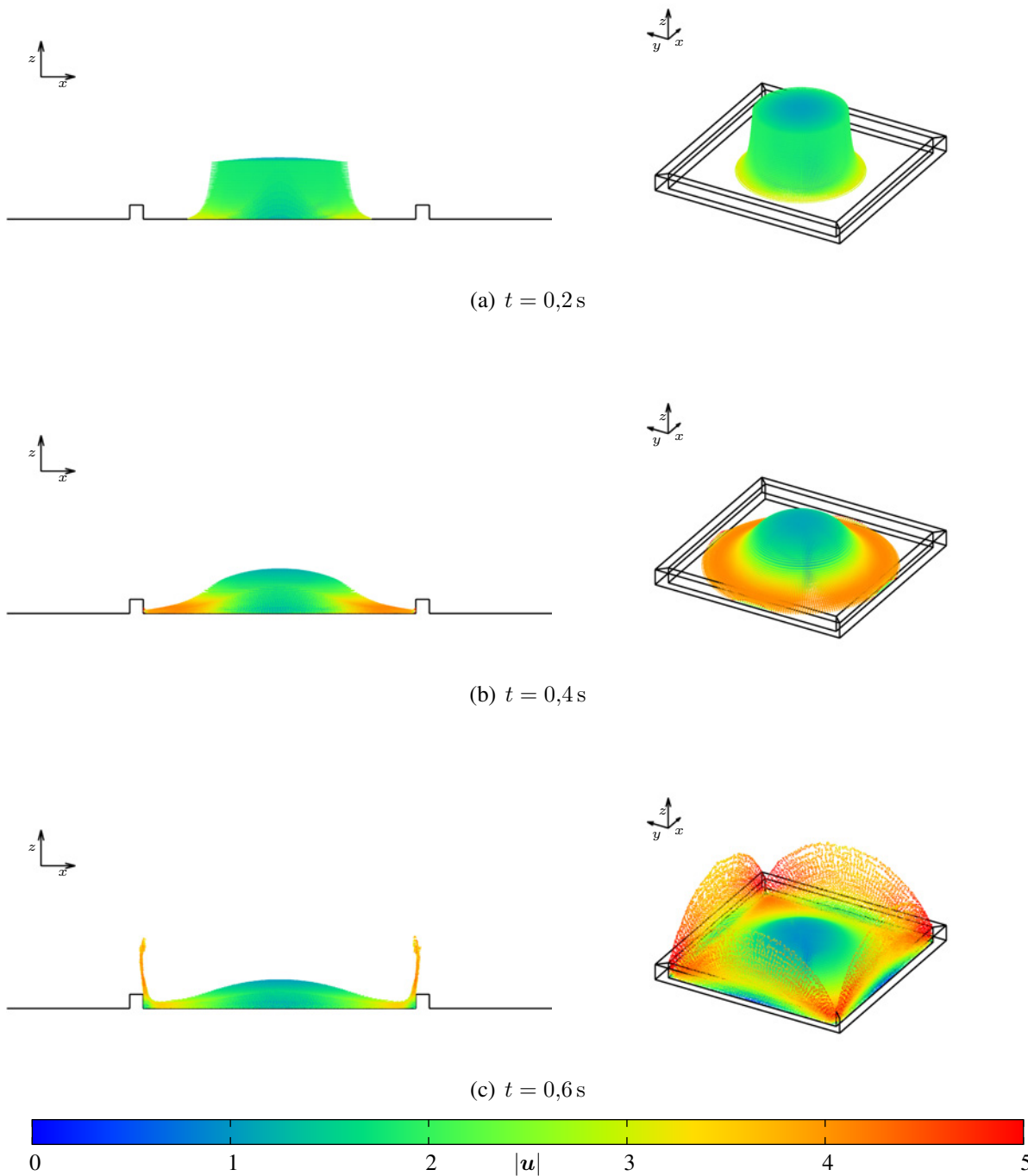


FIGURE D.12 – Effondrement d’une colonne d’eau en trois dimensions par SPH. Vue latérale dans le plan (x, z) des particules situées à $y \in [-1; 1]$ (à gauche) et vue isométrique de l’ensemble des particules (à droite), colorées en fonction du module de vitesse.

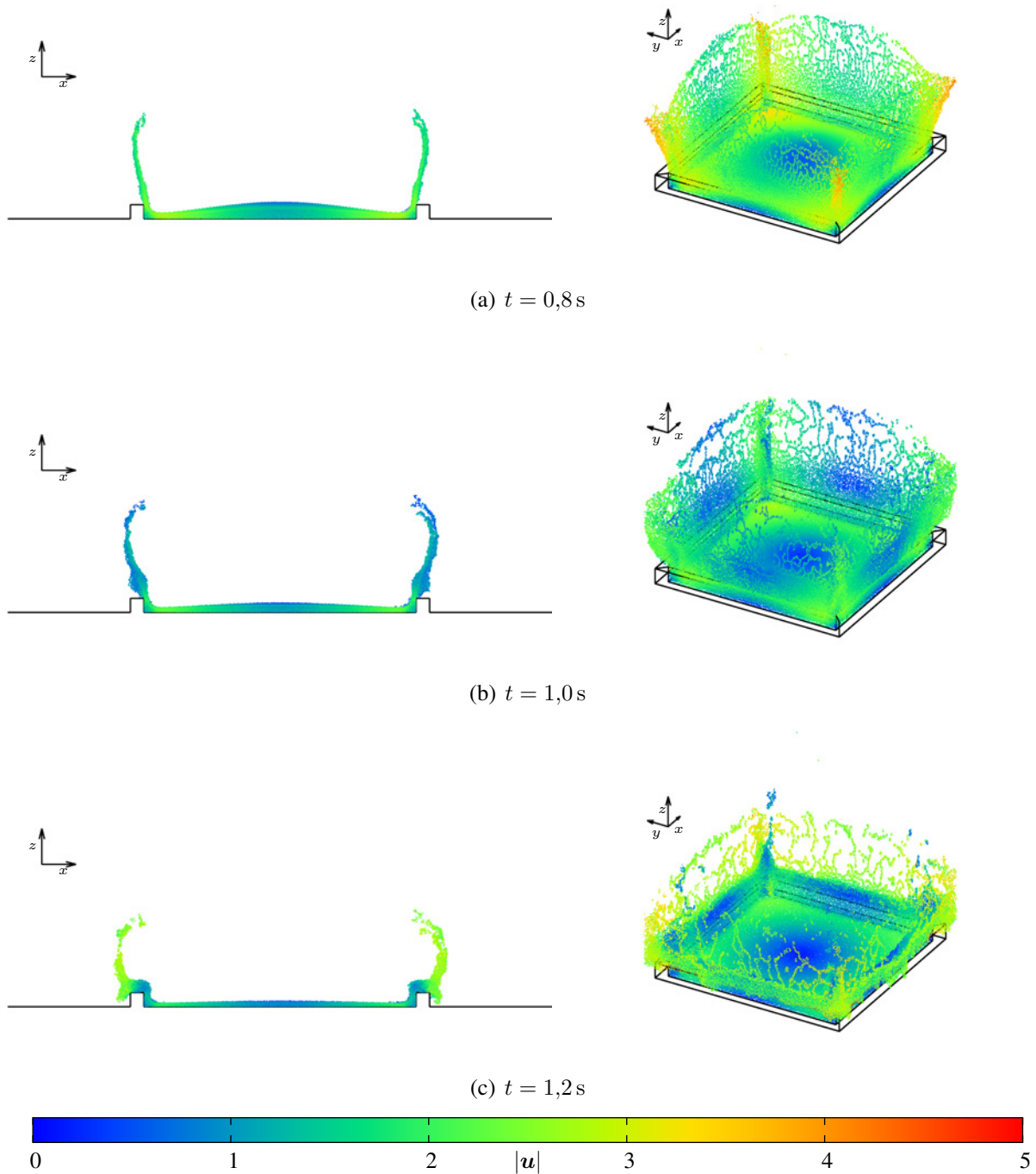


FIGURE D.13 – Effondrement d’une colonne d’eau en trois dimensions par SPH. Vue latérale dans le plan (x, z) des particules situées à $y \in [-1; 1]$ (à gauche) et vue isométrique de l’ensemble des particules (à droite), colorées en fonction du module de vitesse.



Le champ de particules est représenté sur les figures D.12 et D.13 à différents instants. La masse fluide s'effondre (figure D.12(b)) et l'impact sur le merlon de rétention est à l'origine de la création d'une *nappe* verticale assimilable à un jet sur la coupe transversale (figure D.12(c)). L'impact est d'ailleurs plus violent aux coins du bassin de rétention, où des jets se forment également (figure D.13(a)). Les nappes retombent ensuite à l'extérieur du bassin de rétention (figures D.13(b) et D.13(c)). Ces résultats montrent que des merlons, mêmes verticaux (souvent de section trapézoïdale à petite base vers le haut, en réalité), ne suffisent pas à retenir la masse fluide. L'évolution des jets après impact peut évidemment être mise en doute, tant le nombre de particules dans l'épaisseur de ces nappes devient faible. Les coûts de simulation 3D sont largement supérieurs (plus de 8 fois) à ceux d'une simulation 2D, pour un nombre de particules équivalent. Une faible augmentation du nombre de particules a un impact très significatif sur la durée d'une simulation.

Bibliographie

- [1] **O. AGERTZ, B. MOORE, J. STADEL, D. POTTER, F. MINIATI, J. READ, L. MAYER, A. GAWRYSZCZAK, A. KRAVTSOV, J. J. MONAGHAN, A. NORDLUND, F. PEARCE, V. QUILIS, D. RUDD, V. SPRINGEL, J. STONE, E. TASKER, R. TEYSSIER, J. WAD-SLEY ET R. WALDER** : Fundamental differences between sph and grid methods. *Monthly Notices Royal Astronomical Society*, 380:963–978, 2007. **Lien**.
- Cité page 17
- [2] **Y. ANDRILLON** : *Simulation d'écoulements à surface libre par une méthode de capture d'interface en formulation totalement couplée*. Thèse de doctorat, Ecole Centrale de Nantes, 2004. **Lien**.
- Cité page 91
- [3] **C. ANTOCI, M. GALLATI ET S. SIBILLA** : Numerical simulation of fluid-structure interaction by sph. *Computers & Structures*, 85(11-14):879–890, 2007. **Lien**.
- Cité page 95
- [4] **R. ATA ET A. SOULAÏMANI** : A stabilized sph method for inviscid shallow water flows. *International Journal for Numerical Methods in Fluids*, 47(2):139–159, 2005. **Lien**.
- Cité page 95
- [5] **B. ATAIE-ASHTIANI ET L. FARHADI** : A stable moving-particle semi-implicit method for free surface flows. *Fluid Dynamics Research*, 38(4):241–256, 2006. **Lien**.
- Cité page 93
- [6] **B. ATAIE-ASHTIANI, G. SHOBEYRI ET L. FARHADI** : Modified incompressible sph method for simulating free surface problems. *Fluid Dynamics Research*, 40(9):637–661, 2008. ISSN 0169-5983. **Lien**.
- Cité page 41
- [7] **E. BALARAS** : Modeling complex boundaries using an external force field on fixed cartesian grids in large-eddy simulations. *Computers & Fluids*, 33(3):375–404, 2004. ISSN 0045-7930. **Lien**.
- Cité page 122



-
- [8] **DINSHAW S. BALSARA** : von neumann stability analysis of smoothed particle hydrodynamics - suggestions for optimal algorithms. *Journal of Computational Physics*, 121(2):357–372, 1995. ISSN 0021-9991. **Lien**.
Cité page 65
- [9] **J. E. BARNES ET P. HUT** : Error analysis of a tree code. *The Astrophysical Journal Supplement Series*, 70:389–417, 1989. **Lien**.
Cité page 185
- [10] **M. BASA, N. J. QUINLAN ET M. LASTIWKA** : Robustness and accuracy of sph formulations for viscous flow. *International Journal for Numerical Methods in Fluids*, 60(10):1127–1148, 2009. **Lien**.
2 citations pages 48 et 144
- [11] **G. K. BATCHELOR** : *An Introduction to Fluid Dynamics*. Cambridge University Press, 1967.
3 citations pages 9, 39, et 170
- [12] **T. BELYTSCHKO, Y. KRONGAUZ, J. DOLBOW ET C. GERLACH** : On the completeness of meshfree particle methods. *International Journal for Numerical Methods in Engineering*, 43(5):785–819, 1998. **Lien**.
2 citations pages 66 et 80
- [13] **T. BELYTSCHKO, Y. KRONGAUZ, D. ORGAN, M. FLEMING ET P. KRYSL** : Meshless methods : An overview and recent developments. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):3–47, 1996. **Lien**.
3 citations pages 14, 103, et 169
- [14] **T. BELYTSCHKO, Y. Y. LU ET L. GU** : Element-free galerkin methods. *International Journal for Numerical Methods in Engineering*, 37(2):229–256, 1994. **Lien**.
2 citations pages 13 et 14
- [15] **J. BONET ET S. KULASEGARAM** : Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations. *International Journal for Numerical Methods in Engineering*, 47(6):1189–1214, 2000. **Lien**.
2 citations pages 69 et 73
- [16] **J. BONET ET T. S. L. LOK** : Variational and momentum preservation aspects of smooth particle hydrodynamic formulations. *Computer Methods in Applied Mechanics and Engineering*, 180(1-2):97–115, 1999. ISSN 0045-7825. **Lien**.
9 citations pages 32, 44, 57, 58, 69, 73, 77, 80, et 94
-

[17] **B. BOULIER** : *Evolution spatio-temporelle de structures tourbillonnaires dans la houle*. Thèse de doctorat, Université du Havre, 1996.

Cité page 165

[18] **J. BROSSARD ET M. CHAGDALI** : Experimental investigation of the harmonic generation by waves over a submerged plate. *Coastal Engineering*, 42(4):277–290, 2001. ISSN 0378-3839. **Lien**.

Cité page 165

[19] **J. BROSSARD, A. HÉMON ET E. RIVOALEN** : Improved analysis of regular gravity waves and coefficient of reflexion using one or two moving probes. *Coastal Engineering*, 39(2-4):193–212, 2000. ISSN 0378-3839. **Lien**.

Cité page 165

[20] **J. BROSSARD, G. PERRET, L. BLONCE ET A. DIEDHIOU** : Higher harmonics induced by a submerged horizontal plate and a submerged rectangular step in a wave flume. *Coastal Engineering*, 56(1):11–22, 2009. ISSN 0378-3839. **Lien**.

Cité page 2

[21] **L. S. BRUESCH ET E. ASPHAUG** : Modeling global impact effects on middle-sized icy bodies : applications to saturn’s moons. *Icarus*, 168(2):457–466, 2004. **Lien**.

Cité page 16

[22] **S. BØRVE, M. OMANG ET J. TRULSEN** : Regularized smoothed particle hydrodynamics : A new approach to simulating magnetohydrodynamic shocks. *The Astrophysical Journal*, 561(1):82–93, 2001. ISSN 0004-637X. **Lien**.

Cité page 73

[23] **S. BØRVE, M. OMANG ET J. TRULSEN** : Regularized smoothed particle hydrodynamics with improved multi-resolution handling. *Journal of Computational Physics*, 208(1):345–367, 2005. ISSN 0021-9991. **Lien**.

Cité page 73

[24] **R. CAPUZZO-DOLCETTA ET R. DI LISIO** : A criterion for the choice of the interpolation kernel in smoothed particle hydrodynamics. *Applied Numerical Mathematics*, 34(4):363–371, 2000. ISSN 0168-9274. **Lien**.

Cité page 28

[25] **S. H. CHA ET A. P. WHITWORTH** : Implementations and tests of godunov-type particle hydrodynamics. *Monthly Notice of the Royal Astronomical Society*, 340(1):73–90, 2003. **Lien**.

Cité page 71



[26] **A. K. CHANIOTIS** : *Remeshed Smoothed Particle Hydrodynamics For The Simulation Of Compressible, Viscous, Heat Conducting, Reacting & Interfacial Flows*. Thèse de doctorat, Swiss Federal Institute of Technology Zurich, 2002. **Lien**.

3 citations pages 16, 73, et 121

[27] **A. K. CHANIOTIS, C. E. FROUZAKIS, J. C. LEE, A. G. TOMBOULIDES, D. POULIKAKOS ET K. BOULOCHOS** : Remeshed smoothed particle hydrodynamics for the simulation of laminar chemically reactive flows. *Journal of Computational Physics*, 191(1):1–17, 2003. **Lien**.

2 citations pages 16 et 73

[28] **A. K. CHANIOTIS ET D. POULIKAKOS** : High order interpolation and differentiation using b-splines. *Journal of Computational Physics*, 197(1):253–274, 2004. **Lien**.

Cité page 29

[29] **A. K. CHANIOTIS, D. POULIKAKOS ET P. KOUMOUTSAKOS** : Remeshed smoothed particle hydrodynamics for the simulation of viscous and heat conducting flows. *Journal of Computational Physics*, 182(1):67–90, 2002. **Lien**.

4 citations pages 16, 47, 73, et 123

[30] **J. K. CHEN, J. E. BERAUN ET C. J. JIH** : An improvement for tensile instability in smoothed particle hydrodynamics. *Computational Mechanics*, 23(4):279–287, 1999. **Lien**.

3 citations pages 67, 69, et 73

[31] **SHIYI CHEN ET GARY D. DOOLEN** : Lattice boltzmann method for fluid flows. *Annual Review of Fluid Mechanics*, 30(1):329–364, 1998. ISSN 0066-4189. **Lien**.

Cité page 10

[32] **J. P. CHOQUIN ET S. HUBERSON** : Particles simulation of viscous flow. *Computers & Fluids*, 17(2):397–410, 1989. ISSN 0045-7930. **Lien**.

Cité page 31

[33] **A. J. CHORIN** : Numerical solution of the navier-stokes equations. *Mathematics of Computation*, 22(104):745–762, 1968. **Lien**.

2 citations pages 38 et 40

[34] **P. W. CLEARY** : Modelling confined multi-material heat and mass flows using sph. *Applied Mathematical Modelling*, 22(12):981–993, 1998. ISSN 0307-904X. **Lien**.

2 citations pages 47 et 48

[35] **A. COLAGROSSI** : *A Meshless Lagrangian Method For Free-Surface And Interface Flows With Fragmentation*. Thèse de doctorat, University Of Roma La Sapienza, 2005. **Lien**.

20 citations pages ii, iv, 33, 57, 58, 59, 62, 63, 64, 65, 67, 68, 72, 73, 105, 195, 235, 236, 247, et 294

-
- [36] **A. COLAGROSSI, M. ANTUONO ET D. LE TOUZÉ** : Theoretical considerations on the free-surface role in the smoothed-particle-hydrodynamics model. *Phys. Rev. E*, 79(5): 056701–13, 2009. **Lien.**

Cité page 92

- [37] **A. COLAGROSSI, M. ANTUONO ET S. MARRONE** : A 2d+t sph model with enhanced solid boundary treatment. *In 4th International SPHERIC SPH Workshop, Nantes, 2009.* **Lien.**

Cité page 103

- [38] **A. COLAGROSSI ET G. COLICCHIO** : Sph method applied to naval-hydrodynamic problems. *Communications to SIMAI Congress ; Vol 1 (2006)*, 1:–, 2007. **Lien.**

Cité page 247

- [39] **A. COLAGROSSI, G. COLICCHIO ET D. LE TOUZÉ** : Enforcing boundary conditions in sph applications involving bodies with right angles. *In 2nd International SPHERIC SPH Workshop, Madrid, 2007.* **Lien.**

7 citations pages 103, 131, 133, 137, 144, 169, et 298

- [40] **A. COLAGROSSI, L. DELORME ET A. SOUTO IGLESIAS** : Reynolds number and shallow depth sloshing. *In 3rd International SPHERIC SPH Workshop, Lausanne, 2008.*

Cité page 102

- [41] **A. COLAGROSSI ET M. LANDRINI** : Numerical simulation of interfacial flows by smoothed particle hydrodynamics. *Journal of Computational Physics*, 191(2):448–475, 2003. **Lien.**

25 citations pages 33, 45, 59, 60, 62, 66, 69, 87, 93, 94, 99, 104, 106, 113, 114, 115, 116, 117, 118, 119, 127, 294, 296, 297, et 298

- [42] **G. COLICCHIO, M. GRECO ET O.M. FALTINSEN** : Fluid-body interaction on a cartesian grid : dedicated studies for a cfd validation. *In Proc. 21st International Workshop on Water Waves and Floating Bodies, IWWWFB, 2006.*

6 citations pages 130, 131, 132, 133, 144, et 298

- [43] **F. COLIN, R. EGLI ET F. Y. LIN** : Computing a null divergence velocity field using smoothed particle hydrodynamics. *Journal of Computational Physics*, 217(2):680–692, 2006. **Lien.**

Cité page 41

- [44] **R. CORTEZ** : A vortex/impulse method for immersed boundary motion in high reynolds number flows. *Journal of Computational Physics*, 160(1):385–400, 2000. ISSN 0021-9991. **Lien.**

Cité page 122



[45] **G. H. COTTET ET P. KOUMOUTSAKOS** : *Vortex Methods : Theory And Applications*. Cambridge University Press, 2000. **Lien**.

10 citations pages 10, 15, 16, 22, 24, 25, 27, 29, 73, et 125

[46] **G. H. COTTET ET P. A. RAVIART** : Particle methods for the one-dimensional vlasov–poisson equations. *SIAM J. Numer. Anal.*, 21(1):52–76, 1984. **Lien**.

Cité page 20

[47] **R. COURANT, K. FRIEDRICHS ET H. LEWY** : Über die partiellen differenzgleichungen der mathematischen physik. *Mathematische Annalen*, 100(1):32–74, 1928. **Lien**.

3 citations pages 1, 193, et 245

[48] **A. J. C. CRESPO, M. GOMEZ-GESTEIRA ET R. A. DALRYMPLE** : Boundary conditions generated by dynamic particles in sph methods. *Computers, Materials & Continua*, 5:173–184, 2007. **Lien**.

2 citations pages 94 et 98

[49] **R. CROQUET** : Parallélisation et optimisation d’un code de calcul en hydrodynamique marine. Rapport de stage ingénieur insa, Laboratoire Ondes & Milieux Complexes, 2008.

3 citations pages 186, 199, et 215

[50] **S. J. CUMMINS ET M. RUDMAN** : An sph projection method. *Journal of Computational Physics*, 152(2):584–607, 1999. **Lien**.

3 citations pages 38, 40, et 245

[51] **R. A. DALRYMPLE ET O. KNIO** : Sph modelling of water waves. pages 80–80, Lund, Sweden, juin 2001. ASCE. **Lien**.

Cité page 97

[52] **R. A. DALRYMPLE ET B. D. ROGERS** : Numerical modeling of water waves with the sph method. *Coastal Engineering*, 53(2-3):141–147, 2006. **Lien**.

5 citations pages 44, 59, 61, 98, et 294

[53] **R. DAVÉ, J. DUBINSKI ET L. HERNQUIST** : Parallel treesph. *New Astronomy*, 2(3):277–297, 1997. **Lien**.

Cité page 200

[54] **M. DE LEFFE, D. LE TOUZÉ ET B. ALESSANDRINI** : Normal flux method at the boundary for sph. *In 4th International SPHERIC SPH Workshop, Nantes, 2009*. **Lien**.

2 citations pages 72 et 99

[55] **P. DEGOND ET S. MAS-GALLIC** : The weighted particle method for convection-diffusion equations. part 1 : The case of an isotropic viscosity. *Mathematics of Computation*, 53

(188):485–507, 1989. ISSN 00255718. **Lien.**

Cité page 31

- [56] **L. DELORME, A. COLAGROSSI, A. SOUTO-IGLESIAS, R. ZAMORA-RODRÍGUEZ ET E. BOTÍA-VERA** : A set of canonical problems in sloshing, part i : Pressure field in forced roll–comparison between experimental results and sph. *Ocean Engineering*, 36(2):168–178, 2009. ISSN 0029-8018. **Lien.**

Cité page 247

- [57] **R. DI LISIO, E. GRENIER ET M. PULVIRENTI** : On the regularization of the pressure field in compressible euler equations. *Annali della Scuola Normale Superiore di Pisa - Classe di Scienze, Sér. 4*, 24(2):227–238, 1997. **Lien.**

Cité page 17

- [58] **R. DI LISIO, E. GRENIER ET M. PULVIRENTI** : The convergence of the sph method. *Computers & Mathematics with Applications*, 35(1-2):95–102, 1998. ISSN 0898-1221. **Lien.**

Cité page 17

- [59] **G. A. DILTS** : Moving-least-squares-particle hydrodynamics - i. consistency and stability. *International Journal for Numerical Methods in Engineering*, 44(8):1115–1155, 1999. **Lien.**

4 citations pages 66, 80, 127, et 131

- [60] **G. A. DILTS** : Moving least-squares particle hydrodynamics ii : conservation and boundaries. *International Journal for Numerical Methods in Engineering*, 48(10):1503–1524, 2000. **Lien.**

2 citations pages 66 et 80

- [61] **R. F. DRESSLER** : Comparison of theories and experiments for hydraulic dam-break wave. *Proceedings of the International Association of Scientific Hydrology*, 3:319–328, 1954.

Cité page 92

- [62] **C. A. DUARTE ET J. T. ODEN** : An h-p adaptive method using clouds. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):237–262, 1996. ISSN 0045-7825. **Lien.**

Cité page 14

- [63] **MICHAEL DUMBSER** : A simple two-phase method for the simulation of complex free surface flows. *Computer Methods in Applied Mechanics and Engineering*, 200(9-12):1204–1219, février 2011. ISSN 0045-7825. **Lien.**

Cité page 91



[64] **J. D. ELDREDGE** : Numerical simulation of the fluid dynamics of 2d rigid body motion with the vortex particle method. *Journal of Computational Physics*, 221(2):626–648, 2007. ISSN 0021-9991. **Lien**.

Cité page 122

[65] **J. D. ELDREDGE, T. COLONIUS ET A. LEONARD** : A dilating vortex particle method for compressible flow. *Journal of Turbulence*, 3:N36, 2002. **Lien**.

7 citations pages 29, 31, 32, 33, 225, 228, et 230

[66] **J. D. ELDREDGE, A. LEONARD ET T. COLONIUS** : A general deterministic treatment of derivatives in particle methods. *Journal of Computational Physics*, 180(2):686–709, 2002. ISSN 0021-9991. **Lien**.

2 citations pages 49 et 80

[67] **M. ELLERO, M. SERRANO ET P. ESPAÑOL** : Incompressible smoothed particle hydrodynamics. *Journal of Computational Physics*, 226(2):1731–1752, 2007. ISSN 0021-9991. **Lien**.

2 citations pages 41 et 58

[68] **P. ESPANOL ET M. REVENGA** : Smoothed dissipative particle dynamics. *Phys. Rev. E*, 67(2):026705, 2003. **Lien**.

Cité page 47

[69] **E. A. FADLUN, R. VERZICCO, P. ORLANDI ET J. MOHD-YUSOF** : Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *Journal of Computational Physics*, 161(1):35–60, 2000. ISSN 0021-9991. **Lien**.

3 citations pages 122, 123, et 125

[70] **J. FANG, A. PARRIAUX, M. RENTSCHLER ET C. ANCEY** : Improved sph methods for simulating free surface flows of viscous fluids. *Applied Numerical Mathematics*, 59(2):251–271, 2009. ISSN 0168-9274. **Lien**.

6 citations pages 57, 58, 69, 73, 80, et 235

[71] **M. FARGE** : L’approche numérique en physique. *Fundamenta Scientiae*, 7:155–175, 1986.

Cité page 1

[72] **J. FELDMAN ET J. BONET** : Dynamic refinement and boundary contact forces in sph with applications in fluid flow problems. *International Journal for Numerical Methods in Engineering*, 72(3):295–324, 2007. **Lien**.

Cité page 99

[73] **S. FERMI, J. PASTA ET S. ULAM** : Studies of nonlinear problems. Rapport technique, Los Alamos, 1955.

Cité page 1

[74] **M. FERRAND, D. LAURENCE ET VIOLEAU D. ROGERS, B. D.** : Improved time scheme integration approach for dealing with semi-analytical boundary conditions in spartacus2d. *In 5th International SPHERIC SPH Workshop, Manchester, 2010. Lien.*

4 citations pages 99, 131, 140, et 299

[75] **A. FERRARI, M. DUMBSER, E. F. TORO ET A. ARMANINI** : A new 3d parallel sph scheme for free surface flows. *Computers & Fluids*, 38(6):1203–1217, 2009. ISSN 0045-7930. **Lien.**

28 citations pages ii, iv, 47, 57, 58, 69, 70, 71, 72, 73, 74, 75, 76, 78, 79, 83, 84, 85, 88, 89, 90, 94, 200, 236, 259, 294, 295, et 296

[76] **D. A. FULK ET D. W. QUINN** : Hybrid formulations of smoothed particle hydrodynamics. *International Journal of Impact Engineering*, 17(1-3):329–340, 1995. ISSN 0734-743X. **Lien.**

Cité page 71

[77] **D. A. FULK ET D. W. QUINN** : An analysis of 1-d smoothed particle hydrodynamics kernels. *Journal of Computational Physics*, 126(1):165–180, 1996. ISSN 0021-9991. **Lien.**

3 citations pages 24, 27, et 28

[78] **R. A. GINGOLD ET J. J. MONAGHAN** : Smoothed particle hydrodynamics : Theory and application to non-spherical stars. *Monthly Notices Royal Astronomical Society*, 181:375–389, 1977. **Lien.**

3 citations pages 7, 16, et 25

[79] **Y GODA ET Y SUZUKI** : Estimation of incident and reflected waves in random wave experiments. *In Proceedings of 15th ICCE, ASCE*, pages 828–845, 1976.

Cité page 165

[80] **D. GOLDSTEIN, R. HANDLER ET L. SIROVICH** : Modeling a no-slip flow boundary with an external force field. *Journal of Computational Physics*, 105(2):354–366, avril 1993. ISSN 0021-9991. **Lien.**

Cité page 122

[81] **M. GOMEZ-GESTEIRA, D. CERQUEIRO, C. CRESPO ET R. A. DALRYMPLE** : Green water overtopping analyzed with a sph model. *Ocean Engineering*, 32(2):223–238, 2005. ISSN 0029-8018. **Lien.**

Cité page 98



[82] **M. GOMEZ-GESTEIRA, B. D. ROGERS, R. A. DALRYMPLE ET A. C. CRESPO** : State-of-the-art of classical sph for free-surface flows. *Journal of Hydraulic Research*, 48:6–27, 2010. **Lien.**

Cité page 98

[83] **H. GOTOH, S. SHAO ET T. MEMITA** : Sph-les model for numerical investigation of wave interaction with partially immersed breakwater. *Coastal Engineering Journal*, 46:39–63, 2004. **Lien.**

Cité page 42

[84] **D. I. GRAHAM ET J. P. HUGHES** : Accuracy of sph viscous flow models. *International Journal for Numerical Methods in Fluids*, 56(8):1261–1269, 2008. **Lien.**

Cité page 48

[85] **M. GRECO** : *A Two-dimensional Study of Green-Water Loading*. Thèse de doctorat, Norwegian University of Science and Technology, 2001. **Lien.**

3 citations pages 91, 94, et 105

[86] **N. GRENIER, D. LE TOUZÉ, P. FERRANT ET J. P. VILA** : Two-phase flow simulations using a volume fraction sph scheme with a riemann solver. *In 3rd International SPHERIC SPH Workshop, Lausanne, 2008.*

Cité page 72

[87] **B. E. GRIFFITH, R. D. HORNING, DAVID M. MCQUEEN ET C. S. PESKIN** : An adaptive, formally second order accurate version of the immersed boundary method. *Journal of Computational Physics*, 223(1):10–49, 2007. ISSN 0021-9991. **Lien.**

Cité page 122

[88] **P. M. GUILCHER, G. DUCROZET, B. ALESSANDRINI ET P. FERRANT** : Water wave propagation using sph models. *In 2nd International SPHERIC SPH Workshop, Madrid, 2007.* **Lien.**

2 citations pages 72 et 85

[89] **F. H. HARLOW** : The particle-in-cell computing method for fluid dynamics. *Methods in Computational Physics*, 3:319–343, 1964.

Cité page 16

[90] **L. HERNQUIST ET N. KATZ** : Treesph - a unification of sph with the hierarchical tree method. *Astrophys. J. Supp.*, 70:419–446, 1989. **Lien.**

2 citations pages 186 et 190

[91] **D. L. HICKS ET L. M. LIEBROCK** : Lanczos' generalized derivative : Insights and applications. *Applied Mathematics and Computation*, 112(1):63–73, 2000. ISSN 0096-3003. **Lien.**

Cité page 31

[92] **S. E. HIEBER ET P. KOUMOUTSAKOS** : A lagrangian particle level set method. *Journal of Computational Physics*, 210(1):342–367, 2005. ISSN 0021-9991. **Lien.**

Cité page 123

[93] **S. E. HIEBER ET P. KOUMOUTSAKOS** : An immersed boundary method for smoothed particle hydrodynamics of self-propelled swimmers. *Journal of Computational Physics*, 227(19):8636–8654, 2008. ISSN 0021-9991. **Lien.**

15 citations pages iii, 121, 122, 123, 124, 125, 126, 127, 141, 142, 143, 144, 219, 220, et 299

[94] **D. HIETEL, K. STEINER ET STRUCKMEIER J.** : A finite volume particle method for compressible flows. *Mathematical Models and Methods in Applied Science*, 10:1363–1382, 2000. **Lien.**

Cité page 15

[95] **C. W. HIRT ET B. D. NICHOLS** : Volume of fluid (vof) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, 1981. ISSN 0021-9991. **Lien.**

Cité page 10

[96] **J. HONGBIN ET D. XIN** : On criterions for smoothed particle hydrodynamics kernels in stable field. *Journal of Computational Physics*, 202(2):699–709, 2005. ISSN 0021-9991. **Lien.**

Cité page 28

[97] **X. Y. HU ET N. A. ADAMS** : An incompressible multi-phase sph method. *Journal of Computational Physics*, 227(1):264–278, 2007. ISSN 0021-9991. **Lien.**

Cité page 41

[98] **X. Y. HU ET N. A. ADAMS** : A constant-density approach for incompressible multi-phase sph. *Journal of Computational Physics*, 228(6):2082–2091, 2009. ISSN 0021-9991. **Lien.**

Cité page 41

[99] **J. HYMERS** : *A treatise on differential equations and on the calculus of finite differences*. Cambridge, 1839.

Cité page 10



[100] **S. I. INUTSUKA** : Godunov-type sph. *Memorie della Societa Astronomia Italiana*, 65: 1027–1032, 1994. **Lien**.

Cité page 71

[101] **S. I. INUTSUKA** : Reformulation of smoothed particle hydrodynamics with riemann solver. *Journal of Computational Physics*, 179(1):238–267, 2002. ISSN 0021-9991. **Lien**.

Cité page 71

[102] **R. ISSA** : *Numerical assessment of the Smoothed Particle Hydrodynamics gridless method for incompressible flows and its extension to turbulent flows*. Thèse de doctorat, University of Manchester, 2004. **Lien**.

2 citations pages 93 et 97

[103] **R. ISSA, E. S. LEE, D. VIOLEAU ET D. R. LAURENCE** : Incompressible separated flows simulations with the smoothed particle hydrodynamics gridless method. *International Journal for Numerical Methods in Fluids*, 47(10-11):1101–1106, 2005. **Lien**.

Cité page 97

[104] **Y. JANG, J. C. MARONGIU ET PARKINSON E.** : Analysis of sph and mesh based simulations using point based post processing tool. *In 3rd International SPHERIC SPH Workshop, Lausanne, 2008*.

Cité page 115

[105] **G. R. JOHNSON ET S. R. BEISSEL** : Normalized smoothing functions for sph impact computations. *International Journal for Numerical Methods in Engineering*, 39(16):2725–2741, 1996. **Lien**.

2 citations pages 26 et 67

[106] **J. KAJTAR ET J. J. MONAGHAN** : Sph simulations of swimming linked bodies. *Journal of Computational Physics*, 227(19):8568–8587, 2008. ISSN 0021-9991. **Lien**.

Cité page 48

[107] **GEORGE KARYPIS ET VIPIN KUMAR** : Multilevelk-way partitioning scheme for irregular graphs,. *Journal of Parallel and Distributed Computing*, 48(1):96–129, janvier 1998. ISSN 0743-7315. **Lien**.

Cité page 200

[108] **R. KECK ET D. HIETEL** : A projection technique for incompressible flow in the meshless finite volume particle method. *Advances in Computational Mathematics*, 23(1):143–169, 2005. **Lien**.

Cité page 15

-
- [109] **A. KHAYYER, H. GOTOH ET S. SHAO** : Enhanced predictions of wave impact pressure by improved incompressible sph methods. *Applied Ocean Research*, 31(2):111–131, 2009. ISSN 0141-1187. **Lien.**
Cité page 41
- [110] **JUNGWOO KIM, DONGJOO KIM ET HAECHEON CHOI** : An immersed-boundary finite-volume method for simulations of flow in complex geometries. *Journal of Computational Physics*, 171(1):132–150, juillet 2001. ISSN 0021-9991. **Lien.**
Cité page 122
- [111] **S. KOSHIZUKA ET Y. OKA** : Moving-particle semi-implicit method for fragmentation of incompressible fluid. *Nuclear Science and Engineering*, 123:421–434, 1996.
2 citations pages 93 et 94
- [112] **P. KOUMOUTSAKOS** : Multiscale flow simulations using particles. *Annual Review of Fluid Mechanics*, 37(1):457–487, 2005. **Lien.**
Cité page 20
- [113] **S. KULASEGARAM, J. BONET, R. W. LEWIS ET M. PROFIT** : A variational formulation based contact algorithm for rigid boundaries in two-dimensional sph applications. *Computational Mechanics*, 33(4):316–325, 2004. **Lien.**
Cité page 98
- [114] **M. C. LAI ET C. S. PESKIN** : An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *Journal of Computational Physics*, 160(2):705–719, 2000. ISSN 0021-9991. **Lien.**
Cité page 122
- [115] **H. LAMB** : *Hydrodynamics*. Cambridge University Press, 1994. **Lien.**
Cité page 9
- [116] **N. LANSON ET J. P. VILA** : Meshless methods for conservation laws. *Mathematics and Computers in Simulation*, 55(4-6):493–501, 2001. ISSN 0378-4754. **Lien.**
Cité page 80
- [117] **N. LANSON ET J. P. VILA** : Convergence des méthodes particulières renormalisées pour les systèmes de friedrichs. *Comptes Rendus Mathématique*, 340(6):465–470, 2005. ISSN 1631-073X. **Lien.**
Cité page 17
- [118] **M. LASTIWKA, N. QUINLAN ET M. BASA** : Permeable and non-reflecting boundary conditions in sph. *In 3rd International SPHERIC SPH Workshop, Lausanne, 2008.*
Cité page 149



[119] **D. V. LE** : *An Immersed Interface Method for Solving Viscous Incompressible Flows Involving Rigid and Flexible Boundaries*. Thèse de doctorat, HCMC University of Technology, Singapore, 2005.

Cité page 122

[120] **D. V. LE, B. C. KHOO ET J. PERAIRE** : An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries. *Journal of Computational Physics*, 220(1):109–138, 2006. ISSN 0021-9991. **Lien**.

Cité page 122

[121] **N. LE THI MINH** : *Etudes théorique et expérimentale du passage des ondes longues sur des obstacles immergés*. Thèse de doctorat, Institut National Polytechnique de Grenoble, 1989.

Cité page 165

[122] **D. LE TOUZÉ, A. COLAGROSSI ET G. COLICCHIO** : Ghost technique for right angles applied to the solution of benchmarks 1 and 2. *In 1st International SPHERIC SPH Workshop, Rome, 2006*. **Lien**.

Cité page 103

[123] **J. LEDUC, J. C. MARONGIU, F. LEBOEUF, M. LANCE ET PARKINSON E.** : Multi-phase sph : A new model based on acoustic riemann solver. *In 4th International SPHERIC SPH Workshop, Nantes, 2009*. **Lien**.

Cité page 72

[124] **E. S. LEE** : *Truly incompressible approach for computing incompressible flow in SPH and comparisons with the traditional weakly compressible approach*. Thèse de doctorat, University of Manchester, 2007. **Lien**.

Cité page 41

[125] **E. S. LEE, C. MOULINEC, R. XU, D. VIOLEAU, D. LAURENCE ET P. STANSBY** : Comparisons of weakly compressible and truly incompressible algorithms for the sph mesh free particle method. *Journal of Computational Physics*, 227(18):8417–8436, 2008. ISSN 0021-9991. **Lien**.

3 citations pages 41, 93, et 97

[126] **E. S. LEE, D. VIOLEAU, D. LAURENCE, P. STANSBY ET C. MOULINEC** : Spheric test case 6 : 2-d incompressible flow around a moving square inside a rectangular box. *In 2nd International SPHERIC SPH Workshop, Madrid, 2007*. **Lien**.

10 citations pages 131, 134, 135, 136, 137, 140, 144, 145, 149, et 298

[127] **TZUNG-HANG LEE, ZHENGQUAN ZHOU ET YUSONG CAO** : Numerical simulations of hydraulic jumps in water sloshing and water impacting. *Journal of Fluids Engineering*, 124(1):215–226, mars 2002. **Lien.**

Cité page 94

[128] **C. LIA ET G. CARRARO** : A parallel treesph code for galaxy formation. *Monthly Notices of the Royal Astronomical Society*, 314(1):145–161, 2000. **Lien.**

Cité page 16

[129] **LARRY D. LIBERSKY, ALBERT G. PETSCHKE, THEODORE C. CARNEY, JIM R. HIPPE ET FIROOZ A. ALLAHDADE** : High strain lagrangian hydrodynamics : A three-dimensional sph code for dynamic material response. *Journal of Computational Physics*, 109(1):67–75, novembre 1993. ISSN 0021-9991. **Lien.**

Cité page 99

[130] **M. LIN ET L. H. HUANG** : Vortex shedding from a submerged rectangular obstacle attacked by a solitary wave. *Journal of Fluid Mechanics*, 651(-1):503–518, 2010. **Lien.**

Cité page 165

[131] **K. LINDSAY ET R. KRASNY** : A particle method and adaptive treecode for vortex sheet motion in three-dimensional flow. *Journal of Computational Physics*, 172(2):879–907, 2001. ISSN 0021-9991. **Lien.**

Cité page 186

[132] **G. R. LIU ET M. B. LIU** : *Smoothed particle hydrodynamics : a meshfree particle method*. World Scientific Publishing, 2003. ISBN 9812384561. **Lien.**

7 citations pages 16, 17, 23, 27, 29, 30, et 31

[133] **M. B. LIU, G. R. LIU ET K. Y. LAM** : Constructing smoothing functions in smoothed particle hydrodynamics with applications. *Journal of Computational and Applied Mathematics*, 155(2):263–284, 2003. ISSN 0377-0427. **Lien.**

2 citations pages 30 et 31

[134] **M. B. LIU, W. P. XIE ET G. R. LIU** : Modeling incompressible flows using a finite particle method. *Applied Mathematical Modelling*, 29(12):1252–1270, 2005. ISSN 0307-904X. **Lien.**

Cité page 94

[135] **W. K. LIU, S. LI ET T. BELYTSCHKO** : Moving least-square reproducing kernel methods (i) methodology and convergence. *Computer Methods in Applied Mechanics and Engineering*, 143(1-2):113–154, 1997. ISSN 0045-7825. **Lien.**

Cité page 15



-
- [136] **WING KAM LIU, YALING LIU, DAVID FARRELL, LUCY ZHANG, X. SHELDON WANG, YOSHIO FUKUI, NEELESH PATANKAR, YONGJIE ZHANG, CHANDRAJIT BAJAJ, JUNGHOOON LEE, JUHEE HONG, XINYU CHEN ET HUAYI HSU** : Immersed finite element method and its applications to biological systems. *Computer Methods in Applied Mechanics and Engineering*, 195(13-16):1722–1749, février 2006. ISSN 0045-7825. **Lien.**
Cité page 122
- [137] **E. Y. LO ET S. SHAO** : Simulation of near-shore solitary wave mechanics by an incompressible sph method. *Applied Ocean Research*, 24(5):275–286, 2002. ISSN 0141-1187. **Lien.**
Cité page 42
- [138] **L. B. LUCY** : A numerical approach to the testing of the fission hypothesis. *Astronomical Journal*, 82:1013–1024, 1977. **Lien.**
3 citations pages 7, 16, et 26
- [139] **R. LÖHNER, C. YANG ET E. OÑATE** : Simulation of flows with violent free surface motion and moving objects using unstructured grids. *International Journal for Numerical Methods in Fluids*, 53(8):1315–1338, 2007. **Lien.**
Cité page 91
- [140] **J. MA ET W. GE** : Is standard symmetric formulation always better for smoothed particle hydrodynamics ? *Computers & Mathematics with Applications*, 55(7):1503–1513, 2008. ISSN 0898-1221. **Lien.**
Cité page 43
- [141] **J. C. MARONGIU, F. LEBOEUF ET PARKINSON E.** : A new treatment of solid boundaries for the sph method. *In 2nd International SPHERIC SPH Workshop, Madrid, 2007.*
Cité page 99
- [142] **J. C. MARONGIU, F. LEBOEUF ET PARKINSON E.** : Riemann solvers and efficient boundary treatments : and hybrid sph-finite volume numerical method. *In 3rd International SPHERIC SPH Workshop, Lausanne, 2008.*
2 citations pages 72 et 99
- [143] **S. MARRONE, A. COLAGROSSI, D. LE TOUZÉ ET G. GRAZIANI** : A fast algorithm for free-surface particles detection in 2d and 3d sph methods. *In 4th International SPHERIC SPH Workshop, Nantes, 2009. Lien.*
2 citations pages 82 et 115
- [144] **S. MARRONE, A. COLAGROSSI, D. LE TOUZÉ ET G. GRAZIANI** : Fast free-surface detection and level-set function definition in sph solvers. *Journal of Computational Physics*,
-

- [145] **J. C. MARTIN ET W. J. MOYCE** : An experimental study of the collapse of liquid columns on a rigid horizontal plane. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 244:312–324, 1952. **Lien.**

2 citations pages 93 et 94

- [146] **PIERRE MARUZEWSKI, DAVID LE TOUZÉ, GUILLAUME OGER ET FRANÇOIS AVELLAN** : Sph high-performance computing simulations of rigid solids impacting the free-surface of water. *Journal of Hydraulic Research*, 48(Extra Issue (2010)):126–134, 2010. ISSN 0022-1686. **Lien.**

Cité page 200

- [147] **Z. MEGLICKI, D. WICKRAMASINGHE ET G. V. BICKNELL** : Three-dimensional structure of truncated accretion discs in close binaries. *Monthly Notice of the Royal Astronomical Society*, 264:691–704, 1993. **Lien.**

Cité page 73

- [148] **Y. MELEÁN, L. D. G. SIGALOTTI ET A. HASMY** : On the sph tensile instability in forming viscous liquid drops. *Computer Physics Communications*, 157(3):191–200, 2004. ISSN 0010-4655. **Lien.**

Cité page 68

- [149] **N. METROPOLIS ET S. ULAM** : The monte carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949. ISSN 01621459. **Lien.**

Cité page 16

- [150] **D. MOLTENI ET C. BILELLO** : Riemann solver in sph. *Memorie Della Societa Astronomica Italiana Supplement*, 1:36–45, 2003. **Lien.**

Cité page 71

- [151] **D. MOLTENI ET A. COLAGROSSI** : A simple procedure to improve the pressure evaluation in hydrodynamic context using the sph. *Computer Physics Communications*, 180(6):861–872, 2009. ISSN 0010-4655. **Lien.**

2 citations pages 69 et 94

- [152] **J. J. MONAGHAN** : Extrapolating b splines for interpolation. *Journal of Computational Physics*, 60(2):253–262, 1985. ISSN 0021-9991. **Lien.**

4 citations pages 25, 26, 29, et 230



[153] **J. J. MONAGHAN** : Particle methods for hydrodynamics. *Computer Physics reports*, 3 (2):71–124, octobre 1985. ISSN 0167-7977. **Lien.**

Cité page 190

[154] **J. J. MONAGHAN** : Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30(1):543–574, 1992. ISSN 0066-4146. **Lien.**

4 citations pages 17, 28, 31, et 65

[155] **J. J. MONAGHAN** : Simulating free surface flows with sph. *Journal of Computational Physics*, 110(2):399–406, 1994. ISSN 0021-9991. **Lien.**

17 citations pages 3, 16, 38, 39, 44, 47, 57, 58, 59, 61, 62, 72, 94, 95, 96, 294, et 296

[156] **J. J. MONAGHAN** : Sph and riemann solvers. *Journal of Computational Physics*, 136 (2):298–307, 1997. ISSN 0021-9991. **Lien.**

2 citations pages 48 et 71

[157] **J. J. MONAGHAN** : Sph without a tensile instability. *Journal of Computational Physics*, 159(2):290–311, 2000. ISSN 0021-9991. **Lien.**

2 citations pages 67 et 68

[158] **J. J. MONAGHAN** : Smoothed particle hydrodynamics. *Reports on Progress in Physics*, 68(8):1703–1759, 2005. ISSN 0034-4885. **Lien.**

3 citations pages 17, 31, et 48

[159] **J. J. MONAGHAN** : Smoothed particle hydrodynamic simulations of shear flow. *Monthly Notices of the Royal Astronomical Society*, 365(1):199–213, 2006. **Lien.**

Cité page 48

[160] **J. J. MONAGHAN ET A. KOS** : Solitary waves on a cretan beach. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 125(3):145–155, 1999. **Lien.**

Cité page 97

[161] **J.J. MONAGHAN** : On the problem of penetration in particle methods. *Journal of Computational Physics*, 82(1):1–15, 1989. ISSN 0021-9991. **Lien.**

Cité page 62

[162] **J. P. MORRIS** : *Analysis Of Smoothed Particle Hydrodynamics With Applications*. Thèse de doctorat, Monash University, 1996.

Cité page 26

[163] **J. P. MORRIS** : A study of the stability properties of smooth particle hydrodynamics. *Publications Astronomical Society Of Australia*, 13:97–102, 1996. **Lien.**

Cité page 33

-
- [164] **J. P. MORRIS, P. J. FOX ET YI ZHU** : Modeling low reynolds number incompressible flows using sph. *Journal of Computational Physics*, 136(1):214–226, 1997. ISSN 0021-9991. **Lien**.
3 citations pages 16, 48, et 103
- [165] **B. B. MOUSSA** : On the convergence of sph method for scalar conservation laws with boundary conditions. *Methods and Applications of Analysis*, 13:29–62, 2006. **Lien**.
Cité page 71
- [166] **B. B. MOUSSA ET J. P. VILA** : Convergence of sph method for scalar nonlinear conservation laws. *SIAM Journal on Numerical Analysis*, 37(3):863–887, 2000. **Lien**.
2 citations pages 11 et 17
- [167] **P. MYCEK** : Extension 3d et validation d’un code de simulation en hydrodynamique marine. Rapport de stage ingénieur insa, Laboratoire Ondes & Milieux Complexes, 2009.
2 citations pages 106 et 259
- [168] **B. NAYROLES, G. TOUZOT ET P. VILLON** : Generalizing the finite element method : Diffuse approximation and diffuse elements. *Computational Mechanics*, 10(5):307–318, 1992. **Lien**.
2 citations pages 13 et 14
- [169] **R. P. NELSON ET J. C. B. PAPALOIZOU** : Variable smoothing lengths and energy conservation in smoothed particle hydrodynamics. *Monthly Notice of the Royal Astronomical Society*, 270:1–20, 1994. **Lien**.
Cité page 73
- [170] **R. NESTOR, M. BASA ET N. QUINLAN** : Moving boundary problems in the finite volume particle method. *In 3rd International SPHERIC SPH Workshop, Lausanne, 2008*.
6 citations pages 131, 137, 138, 139, 140, et 299
- [171] **R. M. NESTOR, M. BASA, M. LASTIWKA ET N. J. QUINLAN** : Extension of the finite volume particle method to viscous flow. *Journal of Computational Physics*, 228(5):1733–1749, 2009. ISSN 0021-9991. **Lien**.
2 citations pages 11 et 15
- [172] **G. OGER** : *Aspects théoriques de la méthode SPH et applications à l’hydrodynamique à surface libre*. Thèse de doctorat, Ecole Centrale de Nantes, 2006. **Lien**.
13 citations pages 44, 45, 59, 60, 62, 87, 101, 103, 118, 190, 200, 294, et 296
- [173] **G. OGER, M. DORING, B. ALESSANDRINI ET P. FERRANT** : An improved sph method : Towards higher order convergence. *Journal of Computational Physics*, 225(2):1472–



- [174] **G. OGER, C. LEROY, E. JACQUIN, D. LE TOUZÉ ET B. ALESSANDRINI** : Specific pre/post treatments for 3-d sph applications through massive hpc simulations. *In 4th International SPHERIC SPH Workshop, Nantes, 2009.* **Lien.**

Cité page 115

- [175] **J. M. OWEN, J. V. VILLUMSEN, P. R. SHAPIRO ET H. MARTEL** : Adaptive smoothed particle hydrodynamics : Methodology. ii. *The Astrophysical Journal Supplement Series*, 116(2):155, 1998. **Lien.**

Cité page 73

- [176] **A. N. PARSHIKOV ET S. A. MEDIN** : Smoothed particle hydrodynamics using interparticle contact algorithms. *Journal of Computational Physics*, 180(1):358–382, 2002. ISSN 0021-9991. **Lien.**

Cité page 71

- [177] **A. N. PARSHIKOV, S. A. MEDIN, IGOR I. LOUKASHENKO ET V. A. MILEKHIN** : Improvements in sph method by means of interparticle contact algorithm and analysis of perforation tests at moderate projectile velocities. *International Journal of Impact Engineering*, 24(8):779–796, 2000. ISSN 0734-743X. **Lien.**

Cité page 71

- [178] **MANA PATARAPANICH** : Maximum and zero reflection from submerged plate. *Journal of Waterway, Port, Coastal and Ocean Engineering*, 110(2):171–181, mai 1984. **Lien.**

Cité page 165

- [179] **MANA PATARAPANICH ET HIN-FATT CHEONG** : Reflection and transmission characteristics of regular and random waves from a submerged horizontal plate. *Coastal Engineering*, 13(2):161–182, juillet 1989. ISSN 0378-3839. **Lien.**

2 citations pages 165 et 168

- [180] **W. G. PENNEY, C. K. THORNHILL, L. FOX ET E. T. GOODWIN** : Part iii. the dispersion, under gravity, of a column of fluid supported on a rigid horizontal plane. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 244(882):285–311, 1952. **Lien.**

Cité page 92

- [181] **G. PERRET** : Mesures expérimentales du coefficient de réflexion d'une plaque immergée dans la houle. Rapport technique, Laboratoire Ondes & Milieux Complexes, 2009.

2 citations pages 165 et 168

[182] **C. S. PESKIN** : The immersed boundary method. *Acta Numerica*, 11(-1):479–517, 2002. **Lien.**

3 citations pages 122, 124, et 125

[183] **CHARLES S PESKIN** : Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25(3):220–252, 1977. ISSN 0021-9991. **Lien.**

5 citations pages 3, 96, 103, 122, et 123

[184] **G. PINON** : *Modélisation numérique des jets transverses : application au cas des inverseurs de poussée d'un avion en phase d'atterrissage*. Thèse de doctorat, Université du Havre - Editions Universitaires Europeennes (ISBN 6131508658), 2005. **Lien.**

Cité page 186

[185] **P. PONCET** : Analysis of an immersed boundary method for three-dimensional flows in vorticity formulation. *Journal of Computational Physics*, 228(19):7268–7288, 2009. ISSN 0021-9991. **Lien.**

Cité page 122

[186] **S. POPINET** : Gerris : a tree-based adaptive solver for the incompressible euler equations in complex geometries. *Journal of Computational Physics*, 190(2):572–600, 2003. ISSN 0021-9991. **Lien.**

Cité page 15

[187] **S. POPINET ET G. RICKARD** : A tree-based solver for adaptive ocean modelling. *Ocean Modelling*, 16(3-4):224–249, 2007. ISSN 1463-5003. **Lien.**

Cité page 15

[188] **A. POUPARDIN, N. BOURNETON, G. PERRET, G. PINON, E. RIVOALEN ET J. BROSSARD** : Sillages tourbillonnaires des structures immergées. In PARALIA, éditeur : *XIèmes Journées Nationales Génie Côtier - Génie Civil, Les Sables d'Olonne*, 2010. **Lien.**

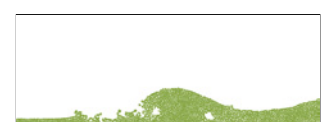
Cité page 165

[189] **A. RAFIEE, M. T. MANZARI ET M. HOSSEINI** : An incompressible sph method for simulation of unsteady viscoelastic free-surface flows. *International Journal of Non-Linear Mechanics*, 42(10):1210–1223, 2007. ISSN 0020-7462. **Lien.**

Cité page 41

[190] **A. RAFIEE ET K. P. THIAGARAJAN** : An sph projection method for simulating fluid-hypoelastic structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 198(33-36):2785–2795, 2009. ISSN 0045-7825. **Lien.**

2 citations pages 41 et 95



[191] **P. W. RANDLES ET L. D. LIBERSKY** : Smoothed particle hydrodynamics : Some recent improvements and applications. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):375–408, 1996. ISSN 0045-7825. **Lien**.

Cité page 81

[192] **P. W. RANDLES ET L. D. LIBERSKY** : Boundary conditions for a dual particle method. *Computers & Structures*, 83(17-18):1476–1486, 2005. ISSN 0045-7949. **Lien**.

Cité page 77

[193] **A. RITTER** : Die fortpflanzung der wasserwellen. *Zeitschrift des Vereins deutscher Ingenieure*, 36:947–954, 1892.

3 citations pages 92, 93, et 296

[194] **M. RODRIGUEZ-PAZ ET J. BONET** : A corrected smooth particle hydrodynamics formulation of the shallow-water equations. *Computers & Structures*, 83(17-18):1396–1410, 2005. ISSN 0045-7949. **Lien**.

Cité page 95

[195] **A. M. ROMA, C. S. PESKIN ET M. J. BERGER** : An adaptive version of the immersed boundary method. *Journal of Computational Physics*, 153(2):509–534, 1999. ISSN 0021-9991. **Lien**.

Cité page 122

[196] **V. ROUBTSOVA ET R. KAHAWITA** : The sph technique applied to free surface flows. *Computers & Fluids*, 35(10):1359–1371, 2006. ISSN 0045-7930. **Lien**.

2 citations pages 71 et 94

[197] **Y. ROUX** : *Etude de l'amortissement visqueux dans les problèmes de tenue à la mer d'un flotteur immergé*. Thèse de doctorat, Université du Havre, 1999.

Cité page 126

[198] **Y. SAAD ET M. H. SCHULTZ** : Gmres : A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. and Stat. Comput.*, 7(3):856–869, 1986. **Lien**.

Cité page 41

[199] **I. F. SBALZARINI, J. H. WALTHER, M. BERGDORF, S.E. HIEBER, E. M. KOTSALIS ET P. KOUMOUTSAKOS** : Ppm - a highly efficient parallel particle-mesh library for the simulation of continuum systems. *Journal of Computational Physics*, 215(2):566–588, 2006. ISSN 0021-9991. **Lien**.

Cité page 123

[200] **I. J. SCHOENBERG** : Contributions to the problem of approximation of equidistant data by analytic functions. part a—on the problem of smoothing or graduation. a first class of analytic approximation formulae. *Quarterly Of Applied Mathematics*, 4:45–99, 1946.

3 citations pages 19, 20, et 22

[201] **I. J. SCHOENBERG** : Contributions to the problem of approximation of equidistant data by analytic functions. part b—on the problem of osculatory interpolation. a second class of analytic approximation formulae. *Quarterly Of Applied Mathematics*, 4:112–141, 1946.

3 citations pages 19, 20, et 22

[202] **FRANK SCHWEITZER** : Active brownian particles : Artificial agents in physics. *Stochastic Dynamics*, 484:358–371, 1997. **Lien.**

Cité page 184

[203] **D. SERVAT** : *Modélisation de dynamiques de flux par agents. Application aux processus de ruissellement, infiltration et érosion.* Thèse de doctorat, Université de Paris 6, 2000. **Lien.**

Cité page 184

[204] **S. SHAO** : Incompressible sph simulation of wave breaking and overtopping with turbulence modelling. *International Journal for Numerical Methods in Fluids*, 50(5):597–621, 2006. **Lien.**

Cité page 41

[205] **S. SHAO** : Incompressible sph simulation of water entry of a free-falling object. *International Journal for Numerical Methods in Fluids*, 59(1):91–115, 2009. **Lien.**

Cité page 42

[206] **S. SHAO, C. JI, D. I. GRAHAM, DOMINIC E. REEVE, P. W. JAMES ET A. J. CHADWICK** : Simulation of wave overtopping by an incompressible sph model. *Coastal Engineering*, 53(9):723–735, 2006. ISSN 0378-3839. **Lien.**

Cité page 42

[207] **S. SHAO ET E. Y. M. LO** : Incompressible sph method for simulating newtonian and non-newtonian flows with a free surface. *Advances in Water Resources*, 26(7):787–800, 2003. ISSN 0309-1708. **Lien.**

3 citations pages 41, 93, et 97

[208] **L. SHEN ET E. S. CHAN** : Numerical simulation of fluid-structure interaction using a combined volume of fluid and immersed boundary method. *Ocean Engineering*, 35(8-9):939–952, 2008. ISSN 0029-8018. **Lien.**

Cité page 122



[209] **G. SHOBEYRI ET M.H. AFSHAR** : Simulating free surface problems using discrete least squares meshless method. *Computers & Fluids*, 39(3):461–470, 2010. ISSN 0045-7930. **Lien.**

Cité page 58

[210] **P. F. SIEW ET D. G. HURLEY** : Long surface waves incident on a submerged horizontal plate. *Journal of Fluid Mechanics*, 83(01):141–151, 1977. **Lien.**

Cité page 165

[211] **L. D. G. SIGALOTTI, J. KLAPP, E. SIRA, Y. MELEÁN ET A. HASMY** : Sph simulations of time-dependent poiseuille flow at low reynolds numbers. *Journal of Computational Physics*, 191(2):622–638, 2003. ISSN 0021-9991. **Lien.**

Cité page 47

[212] **L. D. G. SIGALOTTI ET H. LÓPEZ** : Adaptive kernel estimation and sph tensile instability. *Computers & Mathematics with Applications*, 55(1):23–50, 2008. ISSN 0898-1221. **Lien.**

Cité page 73

[213] **R. SINGH ET W. SHYY** : Three-dimensional adaptive cartesian grid method with conservative interface restructuring and reconstruction. *Journal of Computational Physics*, 224(1):150–167, 2007. ISSN 0021-9991. **Lien.**

Cité page 122

[214] **A. SOUTO-IGLESIAS, L. DELORME, L. PÉREZ-ROJAS ET S. ABRIL-PÉREZ** : Liquid moment amplitude assessment in sloshing type problems with smooth particle hydrodynamics. *Ocean Engineering*, 33(11-12):1462–1484, 2006. ISSN 0029-8018. **Lien.**

Cité page 247

[215] **P. K. STANSBY, A. CHEGINI ET T. C. D. BARNES** : The initial stages of dam-break flow. *Journal of Fluid Mechanics*, 374(-1):407–424, 1998. **Lien.**

Cité page 93

[216] **S. W. SU, M. C. LAI ET C. A. LIN** : An immersed boundary technique for simulating complex flows with rigid boundary. *Computers & Fluids*, 36(2):313–324, 2007. ISSN 0045-7930. **Lien.**

2 citations pages 122 et 126

[217] **Y.C. SUE, M.J. CHERN ET R.R. HWANG** : Interaction of nonlinear progressive viscous waves with a submerged obstacle. *Ocean Engineering*, 32(8-9):893–923, juin 2005. ISSN 0029-8018. **Lien.**

Cité page 165

[218] **J. W. SWEGLE ET S. W. ATTAWAY** : On the feasibility of using smoothed particle hydrodynamics for underwater explosion calculations. *Computational Mechanics*, 17(3): 151–168, 1995. **Lien.**

Cité page 30

[219] **J. W. SWEGLE, D. L. HICKS ET S. W. ATTAWAY** : Smoothed particle hydrodynamics stability analysis. *Journal of Computational Physics*, 116(1):123–134, 1995. ISSN 0021-9991. **Lien.**

3 citations pages 67, 68, et 294

[220] **K. TAIRA** : *The immersed boundary projection method and its application to simulation and control of flows around low-aspect-ratio wings*. Thèse de doctorat, California Institute of Technology, 2008. **Lien.**

Cité page 122

[221] **K. TAIRA ET T. COLONIUS** : The immersed boundary method : A projection approach. *Journal of Computational Physics*, 225(2):2118–2137, 2007. ISSN 0021-9991. **Lien.**

Cité page 122

[222] **K. TAKANO** : Effets d'un obstacle parallépipédique sur la propagation de la houle. *La Houille Blanche*, 3:247–267, 1960.

Cité page 165

[223] **H. TAKEDA, S. M. MUYAMA ET M. SEKIYA** : Numerical simulation of viscous flow by smoothed particle hydrodynamics. *Progress Of Theoretical Physics*, 92:939–960, 1994. **Lien.**

3 citations pages 46, 47, et 99

[224] **E. F. TORO** : *Riemann Solvers and Numerical Methods for Fluid Dynamics - A Practical Introduction*. Springer, 1999.

Cité page 10

[225] **Y. H. TSENG ET J. H. FERZIGER** : A ghost-cell immersed boundary method for flow in complex geometry. *Journal of Computational Physics*, 192(2):593–623, 2003. ISSN 0021-9991. **Lien.**

Cité page 122

[226] **M. UHLMANN** : An immersed boundary method with direct forcing for the simulation of particulate flows. *Journal of Computational Physics*, 209(2):448–476, 2005. ISSN 0021-9991. **Lien.**

Cité page 122



[227] **H. A. VAN DER VORST** : Bi-cgstab : A fast and smoothly converging variant of bi-cg for the solution of nonsymmetric linear systems. *SIAM J. Sci. and Stat. Comput.*, 13(2):631–644, 1992. **Lien.**

Cité page 41

[228] **M. VANELLA ET E. BALARAS** : A moving-least-squares reconstruction for embedded-boundary formulations. *Journal of Computational Physics*, 228(18):6617–6628, 2009. ISSN 0021-9991. **Lien.**

2 citations pages 127 et 129

[229] **G. L. VAUGHAN** : The sph equations for fluids. *International Journal for Numerical Methods in Engineering*, 79(11):1392–1418, 2009. **Lien.**

Cité page 43

[230] **R. VIGNJEVIC, J. CAMPBELL ET L. LIBERSKY** : A treatment of zero-energy modes in the smoothed particle hydrodynamics method. *Computer Methods in Applied Mechanics and Engineering*, 184(1):67–85, 2000. ISSN 0045-7825. **Lien.**

Cité page 77

[231] **J. P. VILA** : Méthodes particulières régularisées. développements récents et nouvelles applications. *ESAIM : Proc. CANum'97*, 3:131–146, 1998. **Lien.**

Cité page 80

[232] **J. P. VILA** : On particle weighted methods and smooth particle hydrodynamics. *Mathematical Models and Methods in Applied Sciences*, 9:161–209, 1999. **Lien.**

4 citations pages 32, 57, 58, et 71

[233] **D. VINCENT, G. BALMIGÈRE, J. P. CALTAGIRONE ET E. MEILLOT** : Eulerian-lagrangian multiscale methods for solving scalar equations - application to incompressible two-phase flows. *Journal of Computational Physics*, 229(1):73–106, 2010. ISSN 0021-9991. **Lien.**

Cité page 91

[234] **D. VIOLEAU ET R. ISSA** : Numerical modelling of complex turbulent free-surface flows with the sph method : an overview. *International Journal for Numerical Methods in Fluids*, 53(2):277–304, 2007. **Lien.**

2 citations pages 93 et 94

[235] **Z. WANG, J. FAN ET K. LUO** : Combined multi-direct forcing and immersed boundary method for simulating flows with moving particles. *International Journal of Multiphase Flow*, 34(3):283–302, 2008. ISSN 0301-9322. **Lien.**

Cité page 122

[236] **L. WEYNANS** : *Methode particuliere multiniveaux pour la dynamique des gaz, application au calcul d'écoulements multifluides*. Thèse de doctorat, Université Joseph-Fourier - Grenoble I, 2006. **Lien**.

Cité page 126

[237] **G. B. WHITHAM** : The effects of hydraulic resistance in the dam-break problem. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 227 (1170):399–407, 1955. **Lien**.

Cité page 92

[238] **R. XU, P. STANSBY ET D. LAURENCE** : Accuracy and stability in incompressible sph (isph) based on the projection method and a new approach. *Journal of Computational Physics*, 228(18):6703–6725, 2009. ISSN 0021-9991. **Lien**.

2 citations pages 41 et 140

[239] **S. XU** : The immersed interface method for simulating prescribed motion of rigid objects in an incompressible viscous flow. *Journal of Computational Physics*, 227(10):5045–5071, 2008. ISSN 0021-9991. **Lien**.

Cité page 122

[240] **Z. Q. ZHOU, J. O. DE KAT ET B. BUCHNER** : A nonlinear 3-d approach to simulate green water dynamics on deck. *In Proceedings of the 7th International Conference on Numerical Ship Hydrodynamics, Nantes, France*, pages 1–15, 1999. **Lien**.

9 citations pages 93, 94, 115, 116, 117, 118, 119, 297, et 298

[241] **O. C. ZIENKIEWICZ, R. L. TAYLOR ET J. Z. ZHU** : *The Finite Element Method Its Basis and Fundamentals*. Elsevier, 2005.

Cité page 10



Ressources électroniques

[242] **ACCIDENT INDUSTRIEL EN HONGRIE : DE L'ALUMINIUM AUX COULÉES DE BOUES ROUGES :**

http://www.cite-sciences.fr/francais/ala_cite/science_actualites/sitesactu/question_actu.php?langue=fr&id_article=16003

[243] **ARIA :**

http://www.aria.developpement-durable.gouv.fr/barpi_4965.jsp

[244] **ARRÊTÉ DU 4 SEPTEMBRE 1967 RELATIF AUX RÈGLES D'AMÉNAGEMENT ET D'EXPLOITATION DES USINES DE TRAITEMENT DE PÉTROLE BRUT, DE SES DÉRIVÉS ET RÉSIDUS :**

<http://droit-finances.commentcamarche.net/textes-de-loi/396253-arrete-du-4-septembre-1967>

[245] **GADGET-2 :**

<http://www.mpa-garching.mpg.de/galform/gadget/index.shtml>

[246] **GERRIS :**

<http://gfs.sourceforge.net/>

[247] **HDF-5 :**

<http://www.hdfgroup.org/HDF5/>

[248] **ISPH :**

<http://isph.sourceforge.net/>

[249] **MÉTHODES SANS MAILLAGE :**

<http://www.isces.org/meshless.html>

[250] **NVIDIA TESLA :**

http://www.nvidia.fr/page/tesla_computing_solutions.html

[251] **OPENCL :**

<http://www.khronos.org/opencv/>



[252] **PORTLAND GROUP : CUDA FORTRAN :**

<http://www.pgroup.com/resources/cudafortran.htm>

[253] **SPARTACUS-2D :**

http://www.telemacsystem.com/index.php?option=com_content&view=article&id=28&Itemid=15&lang=fr

[254] **SPH-FLOW :**

<http://www.sph-flow.com/>

[255] **SPHERIC :**

http://wiki.manchester.ac.uk/spheric/index.php/SPHERIC_Home_Page

[256] **SPHYSICS :**

<http://wiki.manchester.ac.uk/sphysics/>

Table des figures

Page de garde : <i>La Grande Vague de Kanagawa</i> , estampe par Hokusai (1830)	1
1 Bassin de traction B600 du Bassin d’Essais des Carènes de Val de Reuil (gauche) et essai de résistance sur un modèle réduit dans le même bassin. (Photos : Bassin d’Essais des Carènes)	4
2 Écoulement oscillant autour d’une plaque immergée (gauche). Tourbillon généré au bord de la plaque en canal à houle (droite).	4
1.1 Description eulérienne d’un écoulement.	10
1.2 Description lagrangienne d’un écoulement.	11
1.3 Description ALE d’un écoulement.	11
1.4 Maillage cartésien, uniforme (à gauche) ou non (à droite).	12
1.5 Maillage structuré (à gauche) et non-structuré (à droite).	12
1.6 Maillage conforme (à gauche) et non-conforme (à droite, certaines mailles triangulaires ont un nombre de mailles voisines supérieur à 3).	13
1.7 Maillage déformé par la prise en compte de frontières mobiles	14
1.8 Raffinement automatique de maillage sur le code <i>GERRIS</i>	14
1.9 Analogie entre Volumes Finis et Volumes Finis Particulaires.	15
1.10 Simulation de corps gravitaires avec Gadget-2 . Chaque particule représente un corps céleste, 10 milliards de particules ont été employées.	18
2.1 Interpolation sur un ensemble de particules. Le rayon R du noyau $\eta_\epsilon^0()$ dépend de ϵ	22
2.2 Représentation graphique de l’ensemble des noyaux du tableau 2.1.	28
2.3 Approximation particulière au voisinage de la frontière du domaine.	31
2.4 Augmentation du nombre de voisins en fonction de la valeur choisie pour κ , dans le cas du noyau gaussien modifié (équation (2.37)) avec $\delta_c = 3\epsilon$ et selon le nombre de dimensions d’espace.	34
2.5 Représentation graphique du noyau gaussien modifié en 1D (équation (2.39)) et de sa dérivée (équation (2.40)).	35



4.1	Descriptions et notations pour l'étirement d'une goutte en domaine ouvert. Les points A,B et C représentent les positions des différentes mesures exposées dans la suite de ce manuscrit.	56
4.2	Évolution analytique de la forme de la goutte au cours du temps. La couleur associée à chaque ellipse en définit l'instant.	57
4.3	Configuration initiale des particules pour la simulation de l'étirement d'une goutte en domaine ouvert. Les vecteurs symbolisent la vitesse initiale de la particule, colorée en fonction de la pression. Les particules sont positionnées suivant une grille cartésienne.	59
4.4	Évolution de la goutte simulée à l'aide du modèle SPH simple, avec la formulation du gradient de pression de Oger [172] et Colagrossi et Landrini [41] ($\sigma = 1$, équations (3.46) et (3.47) du §3.2). Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.	60
4.5	Évolution de la goutte simulée à l'aide du modèle SPH simple, avec la formulation du gradient de pression de Monaghan [155] et Dalrymple et Rogers [52] ($\sigma = 2$, équations (3.44) et (3.45) du §3.2). Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.	61
4.6	Évolution de la pression calculée au point A (figure 4.1), avec la formulation du gradient de pression de Oger [172] et Colagrossi et Landrini [41] ($\sigma = 1$, équations (3.46) et (3.47) du §3.2). Comparaison entre la valeur calculée par SPH (en noir), la solution analytique (en vert) et la solution SPH filtrée (les fréquences $f > 3, 5A_0$ ont été supprimées, en violet).	62
4.7	Évolution de la goutte simulée à l'aide du modèle SPH de Colagrossi [35]. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.	63
4.8	Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Colagrossi [35]. Comparaison entre la valeur calculée par SPH (en noir), la solution analytique (en vert) et la solution SPH filtrée (les fréquences $f > 3, 5A_0$ ont été supprimées, en violet).	64
4.9	Représentation graphique du critère d'apparition de l'instabilité selon Swegle <i>et al.</i> [219], en fonction des contraintes subies par la particule, et de la position de ses voisines.	68
4.10	Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari <i>et al.</i> [75]. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.	70

4.11	Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75]. Comparaison entre la valeur calculée par SPH (en noir), la solution analytique (en vert) et la solution SPH filtrée (les fréquences $f > 3,5A_0$ ont été supprimées, en violet).	71
4.12	Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari <i>et al.</i> [75], avec $R/\epsilon = 29,4$. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.	74
4.13	Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari <i>et al.</i> [75], avec $R/\epsilon = 58,8$. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.	75
4.14	Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75]. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).	76
4.15	Evolution de la composante horizontale de vitesse au point B (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75]. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).	78
4.16	Evolution de la composante verticale de vitesse au point C (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75]. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).	79
4.17	Valeur propre minimale de la matrice de renormalisation de chaque particule, dans le cas de la goutte étirée.	82
4.18	Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari <i>et al.</i> [75] avec renormalisation des gradients et $R/\epsilon = 58,8$. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.	83
4.19	Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75] avec renormalisation des gradients. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).	84
4.20	Évolution de la composante horizontale de vitesse au point B (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75] avec renormalisation des gradients. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).	84



4.21	Évolution de la composante verticale de vitesse au point C (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75] avec renormalisation des gradients. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).	85
4.22	Configuration initiale circulaire des particules pour la simulation de l'étirement d'une goutte en domaine ouvert. Les vecteurs symbolisent la vitesse initiale de la particule, colorée en fonction de la pression.	86
4.23	Évolution de la goutte simulée à l'aide du modèle SPH simple, avec la formulation du gradient de pression de Oger [172] et Colagrossi et Landrini [41] ($\sigma = 1$, voir §3.2), $R/\epsilon = 14,7$ et une répartition circulaire des particules. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.	87
4.24	Évolution de la goutte simulée à l'aide du modèle SPH de Ferrari <i>et al.</i> [75], avec $R/\epsilon = 14,7$ et une répartition circulaire des particules. Les particules sont colorées en fonction de la pression. Le trait rouge symbolise la position de l'interface selon la solution analytique.	88
4.25	Évolution de la pression calculée au point A (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75] avec répartition circulaire des particules. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).	89
4.26	Évolution de la composante horizontale de vitesse au point B (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75] avec répartition circulaire des particules. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).	89
4.27	Évolution de la composante verticale de vitesse au point C (figure 4.1) par le modèle SPH de Ferrari <i>et al.</i> [75] avec répartition circulaire des particules. Comparaison entre la solution analytique (en noir), la solution SPH avec $R/\epsilon = 14,7$ (en violet), $R/\epsilon = 29,4$ (en vert) et $R/\epsilon = 58,8$ (en bleu).	90
5.1	Configuration initiale de l'effondrement d'une colonne d'eau. L représente la largeur du volume fluide initialement retenu, H est sa hauteur. Un mur vertical est placé à une distance d de l'origine. Le domaine de calcul est limité verticalement à une distance D de l'origine.	92
5.2	Évolution de la surface libre après une rupture de barrage en hypothèse de faible profondeur, selon Ritter [193].	93
5.3	Conditions aux limites imposées à l'aide de particules répulsives selon Monaghan [155].	96
5.4	Conditions aux limites imposées à l'aide de particules virtuelles.	97
5.5	Conditions aux limites imposées à l'aide de particules dynamiques.	98

5.6	Conditions aux limites imposées à l'aide d'intégrales de surface.	98
5.7	Conditions aux limites imposées à l'aide de particules fantômes. Traitement d'un angle : symétrie axiale pour les segments adjacents et symétrie centrale par rapport au coin.	100
5.8	Symétrie d'une particule par rapport à une paroi.	100
5.9	Exemple de configuration problématique avec la méthode des particules fantômes : le coin doit faire l'objet de trois symétries successives, superposant trois couches de particules fantômes au même endroit.	100
5.10	Singularité du champ de pression initial quand celui-ci est choisi hydrostatique. . .	105
5.11	Problème de Laplace pour le calcul du champ de pression initial.	106
5.12	Effondrement d'une colonne d'eau en deux dimensions par SPH. Vues instantanées du champ de pression.	107
5.13	Effondrement d'une colonne d'eau en deux dimensions par SPH intégrant la correction de la condition au bord et du flux Rusanov. Vues instantanées du champ de pression.	111
5.14	Effondrement d'une colonne d'eau en deux dimensions par SPH. Influence de la discrétisation sur la forme de la surface libre.	112
5.15	Évolution de l'énergie totale au cours de l'effondrement d'une colonne d'eau en deux dimensions. Comparaison entre la meilleure solution de Colagrossi et Landrini [41] (en noir), qui emploie une grille de 100×49 particules et la solution présente avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).	114
5.16	Positions de capture des données de comparaison. Les points A ($3, 721H; 0$) et B ($4, 542H, 0$) correspondent aux mesures de hauteur d'eau. Le point C ($d; 0, 192H$).	114
5.17	Évolution de la position du front d'eau. Comparaison entre la solution SPH de référence fournie par Colagrossi et Landrini [41] (en noir) et la solution selon le modèle SPH présent avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).	115
5.18	Évolution de la hauteur d'eau calculée au point A (figure 5.16). Comparaison entre les données expérimentales de Zhou <i>et al.</i> [240] (triangles noirs), la solution SPH de référence fournie par Colagrossi et Landrini [41] (en noir) et la solution selon le modèle SPH présent avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).	116
5.19	Évolution de la hauteur d'eau calculée au point B (figure 5.16). Comparaison entre les données expérimentales de Zhou <i>et al.</i> [240] (triangles noirs), la solution SPH de référence fournie par Colagrossi et Landrini [41] (en noir) et la solution selon le modèle SPH présent avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).	117



5.20	Évolution de la pression calculée au point $C(d; 0, 192H)$ (figure 5.16). Comparaison entre les données expérimentales de Zhou <i>et al.</i> [240] (triangles noirs), la solution SPH de référence fournie par Colagrossi et Landrini [41] (en noir) et la solution selon le modèle SPH présent avec 50×25 particules (en violet), 100×50 particules (en vert) et 200×100 particules (en bleu).	118
6.1	Représentation d'un corps immergés dans le fluide par la méthode de la frontière immergée.	123
6.2	Représentation d'un corps immergés dans le fluide par la méthode de la frontière immergée.	127
6.3	Configuration initiale de l'écoulement autour d'un carré forcé, de côté c , dans une boîte rectangulaire de dimensions $10c \times 5c$	130
6.4	Déplacement du centre du carré au cours du temps.	130
6.5	Solution de référence [42] pour l'écoulement visqueux autour d'un carré forcé. Pression à $t = 5$ et $t = 8$ pour 3 valeurs du nombre de Reynolds : 50, 100 et 150.	131
6.6	Solution de référence [42] pour l'écoulement visqueux autour d'un carré forcé. Module de vitesse à $t = 5$ et $t = 8$ pour 3 valeurs du nombre de Reynolds : 50, 100 et 150.	132
6.7	Solution de référence [42] pour l'écoulement visqueux autour d'un carré forcé. Vorticité à $t = 5$ et $t = 8$ pour 3 valeurs du nombre de Reynolds : 50, 100 et 150.	132
6.8	Etude de l'écoulement autour d'un carré forcé avec SPH par Colagrossi <i>et al.</i> [39].	133
6.9	Pression autour du carré forcé simulé par les modèles SPH faiblement compressible (WCSPH) et incompressible (ISPH) de Lee <i>et al.</i> [126] à $Re = 100$, la solution de référence (REF) est présentée pour comparaison.	134
6.10	Module de vitesse autour du carré forcé simulé par les modèles SPH faiblement compressible (WCSPH) et incompressible (ISPH) de Lee <i>et al.</i> [126] à $Re = 100$, la solution de référence (REF) est présentée pour comparaison.	135
6.11	Module de vitesse autour du carré forcé simulé par le modèle SPH incompressible (ISPH) de Lee <i>et al.</i> [126] à $Re = 050$, la solution de référence (REF) est présentée pour comparaison.	135
6.12	Module de vitesse autour du carré forcé simulé par le modèle SPH incompressible (ISPH) de Lee <i>et al.</i> [126] à $Re = 150$, la solution de référence (REF) est présentée pour comparaison.	136
6.13	Pression autour du carré forcé simulé par le modèle SPH incompressible (ISPH) de Lee <i>et al.</i> [126] à $Re = 050$, la solution de référence (REF) est présentée pour comparaison.	136
6.14	Pression autour du carré forcé simulé par le modèle SPH incompressible (ISPH) de Lee <i>et al.</i> [126] à $Re = 150$, la solution de référence (REF) est présentée pour comparaison.	137

6.15	Module de vitesse autour du carré forcé simulé par le modèle volumes finis particuliers (FVPM) de Nestor <i>et al.</i> [170] à $Re = 50$, la solution de référence (REF) est présentée pour comparaison.	138
6.16	Pression autour du carré forcé simulé par le modèle volumes finis particuliers (FVPM) de Nestor <i>et al.</i> [170] à $Re = 50$, la solution de référence (REF) est présentée pour comparaison.	138
6.17	Module de vitesse autour du carré forcé simulé par le modèle volumes finis particuliers (FVPM) de Nestor <i>et al.</i> [170] à $Re = 150$, la solution de référence (REF) est présentée pour comparaison.	139
6.18	Pression autour du carré forcé simulé par le modèle volumes finis particuliers (FVPM) de Nestor <i>et al.</i> [170] à $Re = 150$, la solution de référence (REF) est présentée pour comparaison.	139
6.19	Module de vitesse autour du carré forcé simulé par SPARTACUS-2D et le modèle de conditions aux limites de Ferrand <i>et al.</i> [74] à $Re = 50$ (SPH) comparé à la solution de référence (FD).	140
6.20	Module de vitesse autour du carré forcé à $Re = 50$ simulé à l'aide du modèle de Hieber et Koumoutsakos [93] (RSPH-IB) et par le nouveau modèle (SPH-IBMLS), la solution de référence (REF) est présentée pour comparaison.	141
6.21	Pression autour du carré forcé à $Re = 50$ simulé à l'aide du modèle de Hieber et Koumoutsakos [93] (RSPH-IB) et par le nouveau modèle (SPH-IBMLS), la solution de référence (REF) est présentée pour comparaison.	142
6.22	Vorticité autour du carré forcé à $Re = 50$ simulé à l'aide du modèle de Hieber et Koumoutsakos [93] (RSPH-IB) et par le nouveau modèle (SPH-IBMLS), la solution de référence (REF) est présentée pour comparaison.	143
6.23	Vorticité autour du carré forcé à $Re = 50$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour trois résolutions différentes 200×100 , 400×200 et 600×300	146
6.24	Vitesse autour du carré forcé à $Re = 50$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour 600×300 particules.	146
6.25	Vorticité autour du carré forcé à $Re = 100$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour trois résolutions différentes 200×100 , 400×200 et 600×300	147
6.26	Vitesse autour du carré forcé à $Re = 100$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour 600×300 particules.	147
6.27	Vorticité autour du carré forcé à $Re = 150$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour trois résolutions différentes 200×100 , 400×200 et 600×300	148



6.28	Vitesse autour du carré forcé à $Re = 150$ simulé à l'aide du modèle SPH, comparée à la solution de référence (REF) pour 600×300 particules.	148
6.29	Coefficient de traînée du carré forcé calculé à l'aide du modèle SPH, pour trois résolutions différentes 200×100 , 400×200 et 600×300 . La solution de référence (REF) est représentée pour comparaison.	150
6.30	Module de vitesse autour du carré forcé à $Re = 300$ simulé à l'aide du modèle SPH.	151
6.31	Vorticité autour du carré forcé à $Re = 300$ simulé à l'aide du modèle SPH.	152
6.32	Module de vitesse autour du carré forcé à $Re = 600$ simulé à l'aide du modèle SPH.	153
6.33	Vorticité autour du carré forcé à $Re = 600$ simulé à l'aide du modèle SPH.	154
7.1	Vue schématique du canal à houle de longueur C , rempli d'eau jusqu'à la hauteur H . Le batteur-piston, de position horizontale $x_b(t)$ (dépendant de sa fréquence d'oscillation f et de son amplitude a), génère une houle régulière de longueur d'onde principale L_s . Une zone d'amortissement de la houle est définie à l'extrémité droite du canal, de longueur L_t , la longueur d'onde théorique (issue de l'équation (7.5)).	158
7.2	Représentation graphique du mouvement du batteur-piston $x_b(t)$ en fonction du temps. La rampe a été appliquée sur 2 périodes ($k = 2$).	159
7.3	Condition d'amortissement sur le canal par modification directe de l'accélération des particules.	159
7.4	Déformation spatio-temporelle de la surface libre, $t/T \in [0; 20]$ et $x \in [0; C - L_t]$, pour 4 périodes de houle différentes.	161
7.5	Comparaison entre les équations de dispersion de la houle de Stokes au second ordre et au troisième ordre ($a_h = 0,02$ m) et les longueurs d'ondes obtenues par le modèle SPH.	163
7.6	Calcul du taux de réflexion par la méthode des ventres et des nœuds.	163
7.7	Enveloppes de surface libre entre $t/T = 15$ et $t/T = 20$, pour 4 longueurs d'onde de houle, amorties à partir du trait noir épais à l'aide de l'équation (7.2)	164
7.8	Vue schématique du canal à houle de longueur C , rempli d'eau jusqu'à la hauteur H . Le batteur-piston, de position horizontale $x_b(t)$ (dépendant de sa fréquence f et de son amplitude a), génère une houle régulière de longueur d'onde L_s . Une plaque horizontale de longueur B est placée à une distance P de la position initiale du batteur et à une distance h_i du fond.	166
7.9	Taux de réflexion R de la houle par la plaque immergée en fonction du rapport B/L'_t , B étant la longueur de la plaque immergée et L'_t la longueur d'onde théorique au second ordre (issue de l'équation 7.4 au dessus de la plaque).	166
7.10	Modélisation d'une paroi mince par la technique des <i>particules fantômes</i> et des <i>visibilités</i>	167

7.11	Vue schématique du voisinage de l'extrémité de la plaque immergée. La taille des particules est définie de manière à avoir l'épaisseur de la plaque supérieure à 6ϵ , soit 8 particules quand $\kappa = 1,33$	169
7.12	Épaisseur de couche limite et nombre de particules nécessaires à la discrétisation du canal en fonction de la viscosité du fluide.	170
7.13	Déformation de la surface libre en fonction du temps et de la position dans le canal sur les 17 premières périodes et sur la longueur <i>utile</i> du canal ($x \in [0; C - L_t]$). Les deux lignes horizontales symbolisent la position de la plaque immergée.	171
7.14	Enveloppes de surface libre entre la 10 ^{ème} et la 13 ^{ème} période.. . . .	172
7.15	Champ de tourbillon autour de la plaque immergée soumise à l'action de la houle en 8 instants de la 10 ^{ème} période.	173
7.16	Efforts horizontaux et verticaux sur la plaque en fonction du temps.	174
7.17	Efforts horizontaux et verticaux sur la plaque en fonction du temps, fenêtre réduite à la 10 ^{ème} période de houle.	174
8.1	Représentations des données en mémoire.	182
8.2	Accès aux données en mémoire. Lors du chargement d'une nouvelle ligne de cache (5 cases ici, pour l'exemple), les données ne sont pas mises en cache dans le même ordre selon le choix de représentation (cases jaunes).	183
8.3	Exemple de construction de l'arborescence en 2D.	187
8.4	Stockage des paires de particules en interaction.	189
8.5	Exemple de mise en œuvre de l'algorithme Linked List en 2D.	191
8.6	Comparaison du temps de détection des interactions en fonction de la méthode de recherche et du nombre de particules.	193
8.7	Définition du nouveau rayon de recherche des interactions pour la méthode de Verlet.	194
8.8	Prise en compte de particules virtuelles lors de la détection des interactions.	196
8.9	Comparaisons des puissances maximales de processeurs GPU et CPU entre 2004 et 2009.	198
9.1	Décomposition du domaine en bandes verticales sur 4 processeurs.	200
9.2	Zone d'interaction entre deux sous-domaines : les informations concernant les particules jaunes de cette zone sont nécessaires au processeur 2 pour l'intégration de la particule ●.	201
9.3	Sous-domaine tel qu'il est vu par le processeur correspondant.	201
9.4	Comparaison entre les exécutions séquentielle et parallèle du code de calcul dans le cas de la rupture de barrage. Les boîtes jaune symbolisent les procédures qui ont du être créées.	203
9.5	Stockage des informations relatives aux particules des processeurs voisins, les ensembles A,B,C,D et E sont définis sur la figure 9.3.	206



9.6	Configuration initiale des particules en parallèle pour différents écoulements, la couleur symbolise le processeur d'affectation de la particule.	208
9.7	Ordonnancement des envois et réceptions de messages entre 8 processeurs.	210
9.8	Évolution de la répartition des particules sur quatre processeurs dans le cas d'une rupture de barrage, sans équilibrage de charge. La couleur symbolise le processeur d'appartenance.	211
9.9	Tapes de l'équilibrage de la charge de calcul entre deux processeurs adjacents.	212
9.10	Évolution de la répartition des particules sur quatre processeurs dans le cas d'une rupture de barrage, avec équilibrage de charge. La couleur symbolise le processeur d'appartenance.	214
9.11	Accélération du code parallèle par rapport au code séquentiel dans le cas d'un canal à houle et d'une rupture de barrage en fonction du nombre de processeurs employés et du nombre de particules.	217
B.1	Descriptions et notations pour l'étirement d'une tourbillon carré en domaine ouvert (gauche). Configuration initiale des particules pour la simulation, les vecteurs symbolisent la vitesse initiale associée à la particule colorée en fonction de la pression (droite).	236
B.2	Évolution du carré fluide sans correction de l'instabilité en tension, ni réinitialisation MLS. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0, 4$, $t\omega = 1, 2$ et $t\omega = 2, 0$	237
B.3	Évolution du carré fluide avec réinitialisation MLS mais sans correction de l'instabilité en tension. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0, 4$, $t\omega = 1, 2$ et $t\omega = 2, 0$	237
B.4	Évolution du carré fluide avec correction de l'instabilité en tension et réinitialisation MLS. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0, 4$, $t\omega = 1, 2$ et $t\omega = 2, 0$	237
B.5	Évolution du carré fluide avec le flux Rusanov sans correction de l'instabilité en tension. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0, 4$, $t\omega = 1, 2$ et $t\omega = 2, 0$	238
B.6	Évolution du carré fluide avec le flux Rusanov et la correction de l'instabilité en tension. Les lignes obliques montrent les trajectoires analytiques des coins du carré initial. Les instants correspondant à chaque image sont (de gauche à droite) : $t\omega = 0, 4$, $t\omega = 1, 2$ et $t\omega = 2, 0$	238

D.1	Rupture d'un bac d'hydrocarbures chauffés, le 25 décembre 1988 à l'Étang de Berre. Vue aérienne des dégâts causés par la vague d'hydrocarbures.	248
D.2	Marée de boues rouges toxiques issues d'un réservoir de l'usine d'aluminium d'Ajka en Hongrie, le 4 octobre 2010.	248
D.3	Évolution de la vitesse du front d'eau en fonction du temps. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu), pour $(d - L)/H = 10$	249
D.4	Évolution de la pression au point $C(d; 0, 192H)$ en fonction du temps pour différentes valeurs de la distance au mur $(d - L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).	252
D.5	Évolution de la pression au point $C(d; 0, 192H)$ en fonction du temps pour différentes valeurs de la distance au mur $(d - L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).	253
D.6	Évolution de la pression au point $C(d; 0, 192H)$ en fonction du temps pour différentes valeurs de la distance au mur $(d - L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).	254
D.7	Calcul des débits sur le mur d'impact.	255
D.8	Débit maximal mesuré à $y_{Q_{\max}}/H = 1,0$ (haut) et $y_{Q_{\max}}/H = 1,5$ (bas) en fonction de la distance au mur $(d - L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).	256
D.9	Instant où le débit maximal (mesuré à $y_{Q_{\max}}/H = 1,0$ (haut) et $y_{Q_{\max}}/H = 1,5$ (bas)) est atteint, en fonction de la distance au mur $(d - L)/H$. Comparaison entre les rapports d'aspect de la masse d'eau $L/H = 0,5$ (en noir), $L/H = 1,0$ (en violet), $L/H = 2,0$ (en vert) et $L/H = 4,0$ (en bleu).	257
D.10	Cas $L/H = 1,0$ et $(d - L)/H = 1,0$ en deux dimensions d'espace. Évolution du débit maximal, de l'instant où le débit est maximal et de la pression mesurée à cet instant, le long du mur d'impact.	258
D.11	Configuration initiale de l'effondrement d'une colonne d'eau en 3D. L représente le rayon du cylindre fluide, H est sa hauteur. Un merlon de hauteur H_m est placé à une distance L du cylindre, formant un bassin de rétention de côté $4L$ centré autour du cylindre.	259
D.12	Effondrement d'une colonne d'eau en trois dimensions par SPH. Vue latérale dans le plan (x, z) des particules situées à $y \in [-1; 1]$ (à gauche) et vue isométrique de l'ensemble des particules (à droite), colorées en fonction du module de vitesse. . .	260



D.13 Effondrement d'une colonne d'eau en trois dimensions par SPH. Vue latérale dans le plan (x, z) des particules situées à $y \in [-1; 1]$ (à gauche) et vue isométrique de l'ensemble des particules (à droite), colorées en fonction du module de vitesse. . . 261

Liste des tableaux

2.1	Noyaux d'interpolation	27
2.2	Opérateurs-intégrales en méthode particulière	30
4.1	Récapitulatif des modélisations avec SPH faiblement compressible de l'étirement d'une goutte.	58
5.1	Récapitulatif des modélisations avec SPH faiblement compressible de l'effondrement de colonnes d'eau.	94
7.1	Longueurs d'ondes théoriques (L_t , $L_{t,3}$) et simulées L_s pour chacun des 4 cas de houle considérés. Erreurs par rapport aux longueurs d'ondes théoriques et cambrure (l'amplitude de la houle générée a été estimée à $a_h = 0,02$ m quelque soit le cas).	162
7.2	Amplitudes maximales et minimales de l'enveloppe de surface libre	172
8.1	Avantages et inconvénients de chaque représentation des données.	184
8.2	Temps moyen de recherche des interactions par l'algorithme Tree Search en fonction la valeur du paramètre $NMAX$ et du nombre de particules.	188
8.3	Comparaison du temps de détection des interactions en fonction de la méthode de recherche et du nombre de particules.	192
8.4	Comparaison	194
9.1	Temps d'exécution de chacune des étapes du calcul d'une rupture de barrage (5000 particules sur 10000 itérations).	215
A.1	Paramètres de construction des noyaux B-splines	234
D.1	Représentations graphiques des configurations géométriques simulées, correspondant aux valeurs des paramètres étudiés (figure 5.1).	250



LES AVENTURES DE LE CARNET

Ben moi je saute sur mon vélo et je vais

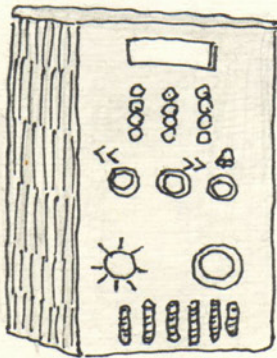
17.03.11



LABORATOIRE ONDES ET
MILIEUX COMPLEXES

là
←
Un truc de
l'université
et lié au
C.N.R.S....

À l'entrée y a un interphone à code



Dedans y a
des portes
à code...



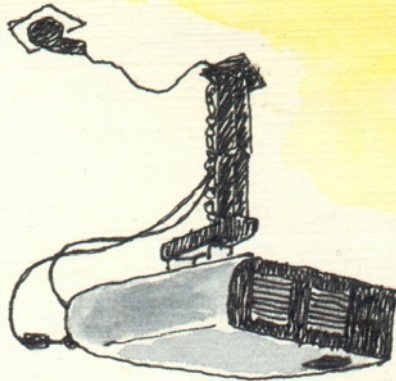
Faut faire gaffe... →



Y a aussi des canaux à fibre,
des caméras, des câbles, des
machines électroniques, tout ça
est moderne, scientifique, et
dans un couloir, une petite
case en bois (sans code) semble
à sa place...

LES AVENTURES DE LE CARNET

J'arrive dans un
Amphi où le rétro
projecteur tourne à
plein régime car...



JM, alias Jean Jean, joueur de saxo et tronche,
soutient sa thèse de doctorat, fringé comme
à un mariage...



Titre de la thèse:
"Développements et applications
de la méthode SPH aux
écoulements visqueux
à surface libre."

Y a des jolis trucs qui
passent à l'écran, sinon, j'en trouve que dalle....

LES AVENTURES DE LE CARNET

Tout en regardant les petits dessins animés, je
chope des bribes de thèse...

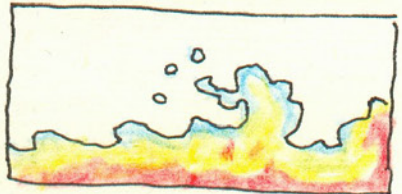


"...L'ennui, avec la méthode des
particules fantômes..."



"...approximation d'un gradient
de la fonction F ..."

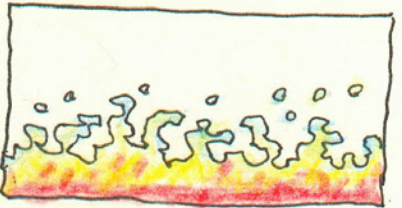
"...opérateur d'interpolation..."



"...vorticité parasitaire..."

"...équation de poisson..."

"...écoulements interfaciaux..."



etc...

Et, enfin, le pot aux roses!

"...Tout sa pour propager de la Roule..." Haaa, Bon....

Membre du Jury
De Face, le sosie de Jean Louis
Borloo!!

Jolie Brune
à coiffure
complexe...



Membre du jury à chemise
fripée (connait pas l'équation
du fer à repasser...)

LES AVENTURES DE LE CARNET



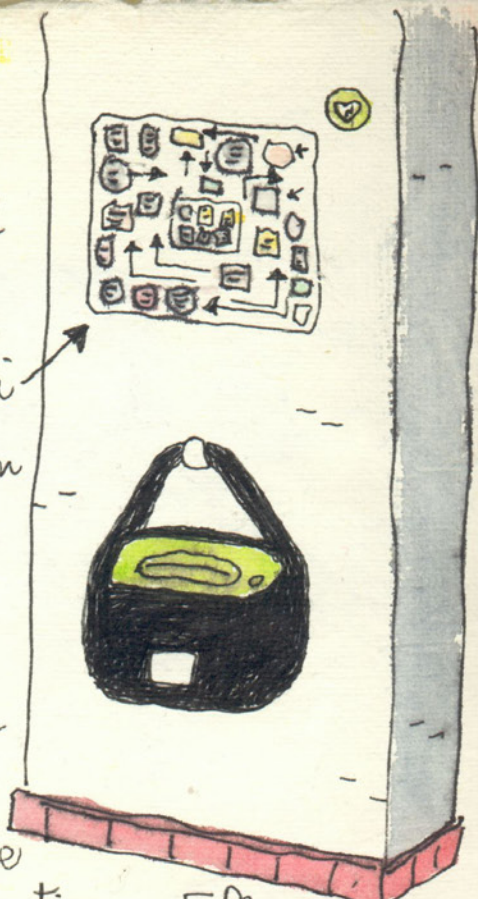
$$\Pi_{ij} = \sum m_j n_i$$

$$\nabla_{\theta}^{\alpha} (x_{ij})^{\frac{m_j}{p_j}}$$

LES AVENTURES DE LE CARNET

Bon, après le jury pose des questions je suis prendre l'air - Accroché dans le Hall y a un défibrillateur, avec un mode d'emploi dont la complexité n'a rien à envier à la thèse de Jean-Jean...

Mais V. qui travaille ici m'apprend que le défibrillateur parle tout seul, aussi, y a qu'à suivre les instructions...



Elle me montre le canal à Houle de 35 M...



SPLOCH!
SPLOCH

En gros c'est un aquarium où ils font des vagues et ils étudient ça.

Des fois, quand ils se remplissent, on n'a plus d'eau aux W.C!

Retour à L'Amphi, 2 types du jury commencent à s'embrouiller "le pourcentage, là!" "quoi, le pourcentage?!" - puis, ils vont délibérer...

LES AVENTURES DE LE CARNET

Le Buffet s'installe, le jwuy revient donner son verdict. "du très bon travail!"

"tu devrais publier"

Voilà, Jean Jean est docteur de l'université, on applaudit et on boit

un verre ou deux... les parents & frères sont là...

Pourtain, bientôt des vacances!

(4 ans de boulot...)

il est pas très ressemblant, bon, tant pis...

il remercie ses 2 Atagjains au passage...



La maman a concocté des chouquettes & très bons gâteaux...

Et puis, quand on a fini tout le Buffet, avec Jean Jean et les ROCKEUX on grimpe tous dans le camion à Francesco mais c'est une autre histoire...



Ouais!
DOCTEUR
JEAN JEAN!
HA! HA! HA!

À Plus, J.

