



Improvement of the precision and the efficiency of the SPH method: theoretical and numerical study

Daniel Afonso Barcarolo

► To cite this version:

Daniel Afonso Barcarolo. Improvement of the precision and the efficiency of the SPH method: theoretical and numerical study. Fluids mechanics [physics.class-ph]. Ecole Centrale de Nantes (ECN), 2013. English. NNT: . tel-00904198

HAL Id: tel-00904198

<https://theses.hal.science/tel-00904198>

Submitted on 14 Nov 2013

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.

Ecole Centrale de Nantes

ÉCOLE DOCTORALE
SCIENCES POUR L'INGENIEUR, GEOSCIENCES & ARCHITECTURE

Année 2013

N° B.U. :

Thèse de DOCTORAT

Spécialité : MECANIQUE DES MILIEUX FLUIDES

Présentée et soutenue publiquement par :

DANIEL AFONSO BARCAROLO

le 24/10/2013
à Ecole Centrale de Nantes

**AMÉLIORATION DE LA PRÉCISION ET DE L'EFFICACITÉ DE LA MÉTHODE SPH:
ÉTUDE THÉORIQUE ET NUMÉRIQUE**

JURY

Président :	M. Rung Thomas	Professeur, Hamburg University of Technology
Rapporteurs :	M. Chénier Eric M. Souto-Iglesias Antonio	Maître de Conférences, H.D.R., Université Paris-Est Marne-la-Vallée Maître de Conférences, H.D.R., Technical University of Madrid
Examineurs :	M. Colagrossi Andrea M. de Vuyst Florian M. le Touzé David	Ingénieur de Recherche, Italian Ship Model Basin Professeur, Ecole Nationale Supérieure de Cachan Professeur, Ecole Centrale de Nantes

Directeur de thèse : David le Touzé

Laboratoire : Laboratoire de recherche en Hydrodynamique, Énergétique et Environnement Atmosphérique, Ecole Centrale de Nantes

Co- Encadrant : Florian de Vuyst

Laboratoire : Centre de Mathématiques et de leurs Applications, Ecole Normale Supérieure de Cachan N° ED : 498-290

Ecole Centrale de Nantes

ÉCOLE DOCTORALE
SCIENCES POUR L'INGENIEUR, GEOSCIENCES & ARCHITECTURE

Year 2013

N° B.U. :

Ph.D. Thesis

Domain : FLUID MECHANICS

Presented and defended by :

DANIEL AFONSO BARCAROLO

24/10/2013
at Ecole Centrale de Nantes

**IMPROVEMENT OF THE PRECISION AND THE EFFICIENCY OF THE SPH METHOD:
THEORETICAL AND NUMERICAL STUDY**

COMMITTEE

Chairman :	M. Rung Thomas	Professor, Hamburg University of Technology
Rapporteurs :	M. Chénier Eric M. Souto-Iglesias Antonio	Assitant Professor, Université Paris-Est Marne-la-Vallée Professor, H.D.R., Technical University of Madrid
Reviewers :	M. Colagrossi Andrea M. de Vuyst Florian M. le Touzé David	Research Engineer, Italian Ship Model Basin Professor, Ecole Nationale Supérieure de Cachan Professor, Ecole Centrale de Nantes

Supervisor : David le Touzé
Laboratory : Laboratoire de recherche en Hydrodynamique, Énergétique et Environnement Atmosphérique, Ecole Centrale de Nantes
Co- supervisor : Florian de Vuyst
Laboratory : Centre de Mathématiques et de leurs Applications, Ecole Normale Supérieure de Cachan N° ED : 498-290

Acknowledgements

Firstly I would like to thank my supervisors Prof. David le Touzé and Prof. Florian de Vuyst for giving the opportunity to develop this work. A special thank to Andrea Colagrossi and Nathan Quinlan for supervising my work. I would also like to thank Guillaume Oger for his help for this last 3 years. Thank you Guillaume for having left your door always opened, for always being available to help and for your patience when sometimes I just couldn't agree with what you were saying and finally I had to agree with you. I would also like to thank everyone worked in LHEEA laboratory during my thesis for the support: (I followed the office order: from the basin to the coffee machine) Guillaume Ducrozet, Felicien Bonnefoy, Nicolas Grenier, Thomas Soulard, Yves Perignon, Aurelien Babarit, Jean-Baptiste Saulnier, François Rongère, Salvatore Marrone, Maite Gouin, Pierre Bigay, Maxime Philippe, Guillaume Fourey, Majid Bhinder, Gabriel Reliquet, Amelie Bardin and Matthieu Kerhuel. Baptiste Elie, don't worry, I did not forget you. I only want to thank you for being 'co-bureau', you know how hard it can be! Thank also to the HO guys: Pierre-Michel Guilcher, Matthieu de Leffe and David Guibert.

Most importantly, I would like to address my most sincere acknowledgements to my family that has supported me since I came to live in France. Priscila, obrigado pela companhia e pelo seu amor nestes anos, especialmente nos momentos mais difíceis. Sem você eu não teria chegado aqui. Pai, mãe, Diego, Eduardo, Liliane, Erica e Tatiana, obrigado pelo apoio e carinho. Obrigado a todos.

Et finalement, merci à la France. La liste des remerciements n'est pas très longue. Il faut dire que il y a certaines choses qu'on ne trouve qu'en France et qui rendent la vie ici très agréable: le vin, le pain e le fromage.

Para você, Priscila

Contents

General Introduction	1
I State of the Art	5
1 Introduction	7
2 Navier-Stokes Equations	9
2.1 Boundary Conditions	10
2.1.1 Free-surface	11
2.1.2 Walls	11
2.1.3 Inflow/Outflow	11
3 Smoothed Particle Hydrodynamics	13
3.1 SPH integral interpolation	13
3.1.1 Continuous interpolation	13
3.1.2 Discrete interpolation	14
3.1.3 Convergence and Precision	16
3.2 Weakly-Compressible SPH	17
3.2.1 Stabilizing Techniques	18
3.2.1.1 Artificial Viscosity	19
3.2.1.2 Riemann-SPH	19
3.2.1.3 δ -SPH	20
3.3 Considerations on viscous flows	20
3.4 Boundary Conditions	21
3.5 Time step and integration	21
4 Possible Improvements	23
II Incompressible SPH	27
5 Introduction	29

6	Semi-Implicit ISPH	31
6.1	Time step	32
6.2	Boundary Conditions	33
6.2.1	Free-slip and no-slip walls, inlet and outlet	33
6.2.2	Free-surface	34
6.2.2.1	Singularity	34
6.2.2.2	Detection algorithm	34
6.3	Stabilizing method	36
6.4	System solving	37
6.5	Performance analysis	38
6.6	Discussion	39
7	Explicit ISPH	41
7.1	Principle	41
7.2	Solver procedure	43
7.3	Internal flow validations	43
7.3.1	Lid-driven cavity flow	43
7.3.2	2D flow past a circular cylinder	46
7.4	Free-surface flow results	50
7.4.1	Standing Wave	51
7.4.2	2D dam-break flow against a wall	52
7.4.2.1	Validation results	53
7.4.2.2	Convergence and energy conservation analysis	57
7.4.2.3	CFL study within the Explicit ISPH scheme	59
7.4.3	3D dam-break flow against a rectangular step	61
7.4.4	Impinging jet on a flat plate	63
7.5	Towards Naval Applications	65
7.5.1	Flow interaction with a rigid body dynamics	66
7.5.2	Sloshing phenomena	73
8	Discussion	77
III	Adaptivity in SPH	81
9	Introduction	83
10	Particle Refinement	85
10.1	Theory	85
10.2	Refinement Criteria	88
10.3	Refinement Patterns	89
10.4	General Remarks on the Particle Refinement Technique	92
10.5	Static Refinement Simulations	93

10.6	Dynamic refinement simulations	95
10.6.1	Flow separation through funnel	96
10.6.2	2D dam-break flow against a wall	99
10.7	Discussion	104
11	Particle Derefinement	105
11.1	Theory	106
11.2	Validation results	109
11.2.1	2D dam-break flow against a wall	109
11.2.2	Asymmetric mass distribution body	112
11.2.3	Impinging jet on a flat plate	116
12	Towards Naval and Offshore Problems	119
12.1	Flooding of a Roll-on Roll-off ship section	119
12.2	Wave Breaking	123
13	Discussion	125
IV	Hybrid-SPH: Proof of Concept	127
14	Introduction	129
15	Voronoi Finite Volumes Method	131
15.1	The Voronoi volume	132
15.1.1	Voronoi derivatives	133
15.1.2	Convergence of the Voronoi discrete operators	135
15.2	Discrete Euler equations in the Voronoi-FVM context	136
15.2.1	MUSCL Scheme	136
16	Voronoi-SPH coupling	139
16.1	Description	139
16.2	Questions on the Voronoi-SPH coupling	141
16.2.1	Volume time integration	141
16.2.2	Grid regularity	142
16.3	Algorithm	144
16.4	Results	145
16.4.1	Hydrostatic test case	145
16.4.2	2D dam-break flow	146
16.4.3	Gresho's Vortex	156
17	Discussion	159
V	Final Considerations and Perspectives	161

List of Figures

1	Applications of the SPH method: complex interfaces (top-left), impact phenomena (top-right), shallow water (down-left) and multi-fluid flow (down-right).	2
2.1	Fluid domain and its boundary conditions.	11
6.1	Ghost particles (red) are mirrored from the fluid particles (blue) to enforce: on the left a free slip boundary condition and on the right a no-slip boundary condition.	34
6.2	Search region for the 2D algorithm [1].	36
6.3	Free-surface particles (blue) detected by Marrone <i>et al.</i> [1] algorithm on a dam-break test case	36
6.4	Pressure signal comparison between semi-implicit ISPH_DF and experimental results for 5 different CFL numbers: (a) Non smoothed signal and (b) smoothed signal.	39
7.1	Lid-driven cavity flow sketch.	44
7.2	Comparison between FVM [2] (black triangles), Explicit ISPH_DF (red line) and WC-SPH (green dashed line) with a particle space resolution of $\Delta x = 0.005\text{m}$ for: (a) velocity profile in the x-direction and (b) velocity profile yn the y-direction.	44
7.3	Comparison between FVM (STARCCM+, blue dashed line), Explicit ISPH_DF (red line) and WCSPH (green dashed line) with a particle space resolution of $\Delta x = 0.005\text{m}$ for: (a) pressure profile on the x-direction and (b) pressure profile on the y-direction.	45
7.4	Comparison between FVM [2] (black triangles), Explicit ISPH_DF (red line) and WC-SPH (green dashed line) with a particle space resolution of $\Delta x = 0.005\text{m}$ for: (a) velocity profile on the x-direction and (b) velocity profile on the y-direction.	46
7.5	2D flow past a cylinder simulation sketch.	46
7.6	Comparison between WCSPH and Explicit ISPH_DF for $Re = 40$ with $\Delta x = 0.005\text{m}$	47
7.7	Pressure coefficient around the cylinder for $Re = 40$	48
7.8	Drag coefficient for Reynolds number from 10 to 100. The red line represents the Explicit ISPH_DF solutions, the green points correspond to the experimental results [3].	48
7.9	Plot of the cylinder wake for each method tested and for an experimental result [4]	50
7.10	Comparison of the analytical and explicit ISPH energy decay obtained for the standing wave test case.	51
7.11	Pressure distribution and velocity field at $t = 7.0\text{s}$ with dynamic free-surface condition imposed	52

7.12	2D dam-break simulation sketch.	53
7.13	Comparison of the pressure signal at p_1 for the coarse particle distribution: (a) Riemann SPH, Explicit ISPH and Experimental Results; (b) Semi-Implicit ISPH, Explicit ISPH and Experimental Results	53
7.14	Comparison of the pressure signal at p_1 for the fine particle distribution.	54
7.15	Pressure field obtained with the Explicit ISPH formulation.	55
7.16	Comparison of the maximum value found in the flow for the velocity divergence between semi-implicit and explicit ISPH	56
7.17	Mechanical energy and pressure signal evolution for the five different simulations of the ISPH_DF variant without free-surface detection procedure.	58
7.18	Mechanical energy and pressure signal evolution for the five different simulations of the ISPH_DI variant without free-surface detection procedure.	58
7.19	Mechanical energy and pressure signal evolution for the five different simulations of the ISPH_DF variant with free-surface detection procedure.	59
7.20	Mechanical energy and pressure signal evolution for the five different simulations of the ISPH_DI variant with free-surface detection procedure.	59
7.21	Pressure signal comparison between Explicit ISPH_DF and experimental results for five different CFL conditions.	60
7.22	3D dam-break simulation sketch with the pressure sensors on the right.	61
7.23	Pressure signal comparison for the three pressure sensors.	62
7.24	Sketch of the impinging jet flow test case	63
7.25	Pressure coefficient at $x/L=0.2$: the red line shows the Riemann-SPH result, the green one the SPH result, the blue dashed line the Explicit-SPH result and finally the pink line is the analytical reference value	64
7.26	Pressure field for each SPH variant tested	65
7.27	Sketch of the simulation with an asymmetric mass distribution body	65
7.28	Center of Gravity position (x-direction) comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 with free-surface	67
7.29	Center of Gravity position (y-direction) comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 with free-surface	68
7.30	Body's roll angle comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 with free-surface	69
7.31	Center of Gravity position (x-direction) comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 without free-surface	70
7.32	Center of Gravity position (y-direction) comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 without free-surface	71
7.33	Body's roll angle comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 without free-surface	72
7.34	Sloshing problem sketch (figure from [5]).	73
7.35	Results obtained for different SPH formulations for the sloshing test case with water filling.	74

7.36	Results obtained for different SPH formulations for the sloshing test case with oil filling.	75
8.1	Void formation due to tensile instability for $Re = 3200$ with renormalization.	78
10.1	Illustration of the refinement process and on the change of the particles inside the kernel: <i>mother</i> (red) particles are split into a finite number of <i>daughter</i> (blue) particles.	88
10.2	Feldman <i>et al.</i> [6] first 2D distribution pattern.	89
10.3	Feldman <i>et al.</i> [6] second 2D distribution pattern.	90
10.4	Feldman <i>et al.</i> [6] 3D distribution pattern (figure from [7]).	90
10.5	Lopez <i>et al.</i> [8] 2D distribution pattern.	91
10.6	Graphic for the kernel gradient error following Lopez <i>et al.</i> (read α_{PR} and ϵ_{PR}).	91
10.7	Uneven particle inter-particle space when using $\epsilon_{PR} = 0.55$.	93
10.8	Hydrostatic test case: the central domain highlighted with dark lines will have a different particle refinement.	94
10.9	Evolution of particle distribution and pressure contours for simulation without any particular treatment (left columns) and with the dynamic refinement treatment by Lopez <i>et al.</i> (right columns) at: (a) $t=0.0s$; (b) $t=1.0s$; (c) $t=2.0s$; (d) $t=3.0s$; (e) $t=4.0s$ and (f) $t=5.0s$.	95
10.10	Flow separation through a funnel simulation sketch.	96
10.11	Particles passing through the funnel at $t=0.13s$: right column shows (a) Riemann-SPH, (b) δ -SPH and (c) AV-SPH present results and the left column shows Feldman's results (AV-WCSPH only).	97
10.12	Particles passing through the funnel at $t=0.21s$: right column shows (a) Riemann-SPH, (b) δ -SPH and (c) AV-SPH present results and the left column shows Feldman's results (AV-SPH only).	98
10.13	Riemann-SPH comparison results for the 2D dam-break flow using a particle refinement technique.	100
10.14	Riemann-SPH pressure fields for the Lopez <i>et al.</i> [8] refinement pattern for different time instants (blue line represents the begging of the refined area): (a) $t=0.4s$; (b) $t=0.6s$; (c) $t=0.8s$; (d) $t=1.0s$; (e) $t=1.2s$; (f) $t=1.4s$; (g) $t=1.6s$; (h) $t=1.8s$; (i) $t=2.0s$ and (j) $t=2.2s$.	101
10.15	δ -SPH comparison results for the 2D dam-break flow using a particle refinement technique.	102
10.16	δ -SPH pressure fields for the Lopez <i>et al.</i> [8] refinement pattern for different time instants (blue line represents the begging of the refined area): (a) $t=0.4s$; (b) $t=0.6s$; (c) $t=0.8s$; (d) $t=1.0s$; (e) $t=1.2s$; (f) $t=1.4s$; (g) $t=1.6s$; (h) $t=1.8s$; (i) $t=2.0s$ and (j) $t=2.2s$.	103
11.1	Illustration of some problems related to the particle derefinement procedure: horizontal line shows the limit of the refinement domain and the red and blue particles represent <i>mother</i> and <i>daughter</i> particles respectively.	107
11.2	Constant γ (dark grey) and the transition (light grey) regions on a 2D squared refinement zone.	108

11.3	Wave breaking phenomena: (a) $T^* = 4.8$, (b) $T^* = 5.7$, (c) $T^* = 6.5$, (d) $T^* = 7.3$, (e) $T^* = 8.1$ and (f) $T^* = 8.9$	110
11.4	Pressure signal on $P1$ sensor: comparison between the three simulations and experimental results.	111
11.5	Kinetic and potential energy: comparison between the three simulations.	111
11.6	Sketch of the simulation with an asymmetric mass distribution body	112
11.7	Pressure distribution for three simulations: (a) N200 simulation, (b) N100 to N200 simulations without transition zone and (c) with transition zone	113
11.8	Body motion history for all the simulations considered.	114
11.9	Convergence of the dynamic solution with respect to the size of the refinement domain.	115
11.10	Total mass variation throughout time for the body asymmetric mass distribution test case.	116
11.11	Sketch of the simulation of the impact of a jet on a flat plate with several refinement zones.	117
11.12	(a) Pressure at the stagnation point and (b) four levels of refinement are shown with the value for the γ variable at $t = 3.0s$	117
11.13	Smooth pressure transition between the different refined zones.	118
12.1	Sketch of the Ro-Ro ship flooding simulation	120
12.2	Results' comparison between the constant Δx and dynamic Δx simulations.	121
12.3	Comparison between the experimental results for the water height from [9] and SPH predictions with and without using a dynamic refinement technique.	122
12.4	Wave propagation and breaking test case sketch.	123
12.5	Results obtained for the wave generation and breaking simulations with constant and dynamic Δx	124
15.1	Example of Voronoi and Delaunay tessellations in 2D for a given points distribution. Left: the Voronoi tessellation; middle: the respective Delaunay triangulation and in the right both space sub-divisions are superposed (image from [10]).	133
15.2	Voronoi discretization: interface between volumes i and j	134
16.1	Illustration of the SPH-FVM coupling.	140
16.2	Irregular mesh for a dam-break test case.	142
16.3	Mesh obtained with the use of an arbitrary velocity u_{0i} (a) and without any particular treatment (b).	143
16.4	Mesh distortion between two time steps for the dam-break test case without a mesh control technique.	144
16.5	Hydrostatic case results using the Voronoi-SPH method.	146
16.6	Pressure at probe $p1$ for the two methods considered in this study for $\Delta x = 0.02$ and $\Delta x = 0.01$	148
16.7	Comparison between the results given by the SPH (left) and Voronoi-SPH (right) methods from $t = 1.0s$ to $t = 1.4s$	149

16.8	Comparison between the results given by the SPH (left) and Voronoi-SPH (right) methods from $t = 1.6s$ to $t = 2.0s$	150
16.9	Comparison between the results given by the SPH (left) and Voronoi-SPH (right) methods from $t = 1.0s$ to $t = 1.4s$	151
16.10	Comparison between the results given by the SPH (left) and Voronoi-SPH (right) methods from $t = 1.6s$ to $t = 2.0s$	152
16.11	Plotting of the different regions for the dam-break problem using $\Delta x = 0.01m$: red particles are those that advance in time using an SPH scheme and the rest is evolved using the Voronoi-FVM formalism.	153
16.12	SPH equivalent mesh (red lines) for the Voronoi-SPH dam-break simulation at different time instants.	155
16.13	Situation where the equivalent SPH ‘mesh’ for the Voronoi volume is not totally found.	155
16.14	Decay of the maximum velocity on the domain for the SPH and Voronoi-FVM simulations.	156
16.15	Comparison between the SPH and Voronoi-FVM results for the Gresho vortex problem	157
16.16	Mesh deformation at $t = 1.00s$ for the Voronoi-FVM simulation.	158

List of Tables

6.1	CPU time comparison between semi-implicit ISPH and WCSPH.	39
7.1	Configuration for each method used to simulate the flow around a circular cylinder . . .	49
7.2	Strouhal number comparison.	49
7.3	Convergence study parameters.	57
7.4	CPU time comparison between Explicit ISPH and WCSPH.	60
7.5	Kleefsman's dam-break CPU time comparison	63
7.6	Configuration for each SPH variant used to simulate the impinging jet flow	64
7.7	Configuration for each SPH variant used to simulate the drowning body	66
10.1	Four conditions needed to be ensured during a particle refinement process.	88
11.1	Jet flow simulation set-up.	116
16.1	Numerical values and differences between the results given by the SPH and Voronoi-FVM operators	146

General Introduction

Engineers have always been and will always be confronted to all kinds of challenges in the practice of their profession. Since when engineering served as a mean to protect and to guarantee the survival of our species, when such a profession was not yet called that way, the development of new objects, tools, etc (pulley, lever, wheel, etc) requested a process of “trial and error”, observation and creativity. In other words, since the first inventions of humanity, engineering was only possible if preceded and followed by a process of (or similar to) R&D. Today, research faces more and more difficult challenges. In the field of expertise of the hydrodynamics lab. (LHEEA) of Ecole Centrale de Nantes, Naval and Ocean Engineering the challenges are many: extracting energy has become more difficult (oil drilling is moving to deeper waters); humanity is seeking for cleaner sources of energy (renewable energy from waves, wind, tides, etc); engineers urge to develop more efficient and safer platforms and ships in order to reduce the hazard to the environment; etc (the list may go on for pages). *Ipso facto*, research has, unquestionably, a key role in the better development of the world.

More than two centuries after the appearance of the equations that describe and govern the complexity of fluid dynamics phenomena, their solution remains unknown. With the appearance of the first computers, approaches to solve these equations emerged. Then, as computers evolved and became more and more powerful, the complexity of the phenomena that can be described numerically followed up. Researchers developed more and more refined techniques that served engineers to solve problems that constantly grow in difficulty. Despite the effort, it is not yet possible to solve the totality of the physical aspects that the Navier-Stokes equations describe, and therefore, many questions remain unanswered and many challenges are still there to be taken up.

In the field of hydrodynamics, we have seen the birth and growth of several numerical methods. We may cite several different models based on the potential theory, on finite differences (FDM), on finite volumes (FVM) and finite elements (FEM), where the equations of Navier-Stokes are modeled with different levels of complexity and kinds of simplifying assumptions: taking into account the presence of the free-surface, turbulence effects (Reynolds-Averaged Navier-Stokes, LES, DES, etc), the presence and interaction between different phases (gas, solid, liquid), etc. When a hydrodynamic problem needs to be solved/understood, the choice of the numerical model to be used is not easy. It depends on several aspects: the phenomenon desired to model, the level of accuracy wanted, the amount of details, etc. Every method/technique has its pros and cons, and each one aims a different kind of flow phenomenon. More particularly in the field of naval hydrodynamics, the key problem to be solved is to take the free-surface into account while modeling any flow phenomena. Many industrial scale softwares (based on FVM, FEM, etc) are capable of treating the presence of a free-surface. However,

these codes, that employ the VOF (Volume of Fluid) or Level-set techniques to model this interface, have a lot of difficulties to treat complex free-surface shapes, especially when it suffers of fragmentation and reconnection (wave-breaking phenomena, for example). Moreover, these methods are based on a meshed discretization of the domain and, handling with this mesh in the presence of complex geometries and also with a free-surface, may be very complicated and sometimes impossible.

As an alternative to meshed methods, the Smoothed Particle Hydrodynamics (SPH) appeared in 1977 in the works of Lucy [11] and Gingold & Monaghan [12] which aimed to solve highly complex astrophysical simulations, like the formation of stars, collision between galaxies, etc. Later, and shall we say recently, in 1994, the method was extended to simulate free-surface flows [13]. This method has two major characteristics that make it to be interesting to simulate violent flows (highly convective and with high transfers between inertia and pressure) and/or with interfaces (between two or more fluids, or a free-surface): it is Lagrangian and meshless. Since 1994, the method knew a very rapid development and at the LHEEA laboratory of Ecole Centrale de Nantes and CNRS, since 2001, the Smoothed Particle Hydrodynamics method has been largely studied and improved. Up to the date of this work the total of 8 thesis have been developed and defended in the laboratory: Doring [14], Oger [15], Deuff [16], Guilcher [17], Grenier [18], de Leffe [19], Zhao [20] and Fourey [21]. We could also add the several partnerships with other European institution practicing SPH (INSEAN, Ecole Centrale de Lyon, etc) which increased the knowledge in such a method. Nowadays the LHEEA lab. has a robust code capable of treating complex free-surface flows, with fluid structure interaction (considering a deformable body), multi-fluid flows, shallow water problems, etc (cf. figure 1).

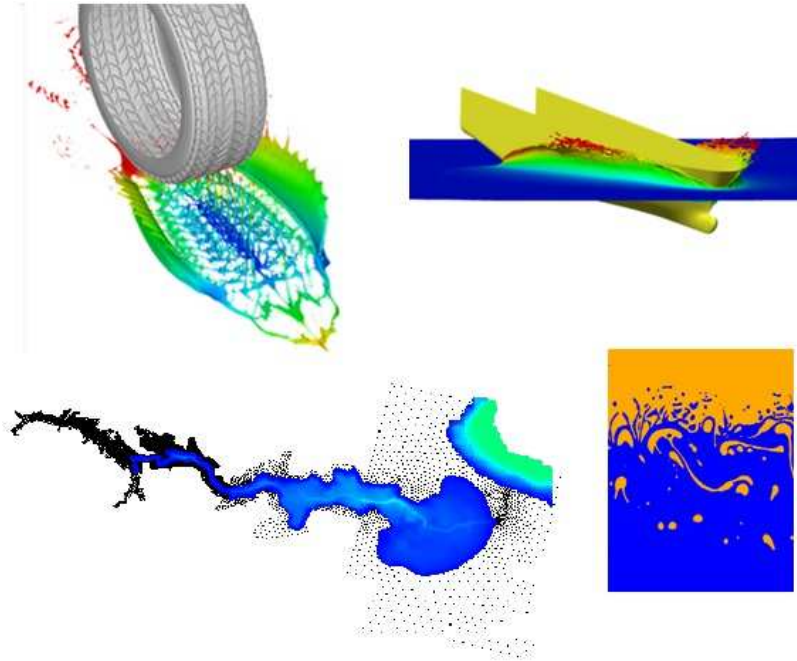


Figure 1: Applications of the SPH method: complex interfaces (top-left), impact phenomena (top-right), shallow water (down-left) and multi-fluid flow (down-right).

The SPH method is now applicable to a large variety of problems and is capable of giving a precise solution for flows that are difficult to be modeled with the traditional mesh-based methods. It has become challenging to apport major improvements to the method because, just like the FVM, FEM or FDM, SPH is getting increasingly mature. Nonetheless, some points regarding the physics considered in the Navier-Stokes equations, the manner in which the fluid domain is discretized and the precision of the discrete operators used in the SPH method, remain to be studied. As the title of this thesis (“Improvement of the precision and the efficiency of the SPH method: theoretical and numerical study”) suggests, we will proceed to a numerical and theoretical study (slightly more numerical) of the method proposing, at the same time, some new techniques towards the continuous improvement of the SPH method.

This manuscript is organized as follows. Firstly, we will establish the basis upon which this thesis is developed, that is, the actual state of the method at the LHEEA and partners laboratories. It will be shown that the works developed so far are based on a weakly-compressible description of the fluid, hence, the second part of this thesis is dedicated to the analysis of the SPH approach based on a truly-incompressible description of the media. This part is followed by the study of a dynamic particle refinement technique and the proposal of a new derefinement method, that allows to dynamically adapt the particle distribution. The last part is a more theoretical study, where we will focus on the possibility creating an Hybrid-SPH method, where the coupling of SPH to a more precise method, like FVM, is analyzed.

Part I

State of the Art

Chapter 1

Introduction

The main goal of this thesis is to apport general improvements to the Smoothed Particle Hydrodynamics method having as starting point the actual state of the method in the literature. Before describing improvements developed in this thesis, it is necessary to give a state of the art of the method. Hence, firstly the general equations that govern our problems are presented and then the method itself is presented together with the variants that are used and were developed both in our (and partners) laboratory(ies) since 2001 and in the general literature. For that reason, focus will be given to general aspects of the method. We shall not, unfortunately, consider the improvements developed for multi-phase, shallow water and fluid-structure interaction problems, even though some of the new techniques/methods that are proposed in this thesis are applicable to these kind of phenomena.

In the end of this first part, a general discussion on the development state of the SPH method in the LHEEA laboratory is performed where its major limitations and problems are briefly discussed. In order to remediate these problems, some possible directions of improvement are presented, directions that will be followed separately throughout this thesis.

Chapter 2

Navier-Stokes Equations

In the domain of computational fluid dynamics there are two main approaches to simulate incompressible flows. The first is to consider the fluid as being weakly-compressible, that is to say that it is possible to retain the compressible Navier-Stokes equations to simulate incompressible flows. In that case the compressibility effects may be neglected. In a Lagrangian formalism the Navier-Stokes equations for a weakly-compressible barotropic fluid are:

$$\left\{ \begin{array}{l} \frac{D\vec{r}}{Dt} = \vec{u} \\ \frac{D\rho}{Dt} + \rho \operatorname{div}(\vec{u}) = 0 \\ \frac{D\vec{u}}{Dt} = \vec{g} + \frac{\nabla \cdot \sigma}{\rho} \\ p = f(\rho) \end{array} \right. \quad (2.1)$$

where \vec{r} is the position vector, \vec{u} is the velocity vector, ρ is the density, p stands for the pressure, \vec{g} represents the external forces vector (gravity acceleration) and σ is the stress tensor of a Newtonian fluid:

$$\sigma = (-p + \lambda d_{kk})\delta_{ij} + 2\mu d_{ij} \quad (2.2)$$

with $d_{ij} = \frac{1}{2} \left(\frac{\partial \vec{u}_i}{\partial r_j} + \frac{\partial \vec{u}_j}{\partial r_i} \right)$.

In the above equation λ is the bulk viscosity and μ the dynamic viscosity. Following the hypothesis that μ is constant and that the fluid is quasi-incompressible, the viscous effects term may be simplified and equations (2.1) may be rewritten in the following manner:

$$\left\{ \begin{array}{l} \frac{D\vec{r}}{Dt} = \vec{u} \\ \frac{D\rho}{Dt} + \rho \operatorname{div}(\vec{u}) = 0 \\ \frac{D\vec{u}}{Dt} = \vec{g} - \frac{\nabla p}{\rho} + \frac{\mu}{\rho} \nabla^2 \vec{u} \\ p = f(\rho) \end{array} \right. \quad (2.3)$$

The use of a compressible approach allows the use of an explicit time integration scheme (for more details on the justification of using the weakly-compressible approach to model incompressible flows, refer to G. Capdeville lecture notes [22]) where the last line in the set of equations (2.3), that is, the

state equation, thermodynamically closes the system of equations. In this work the equation proposed by Tait [23] is considered:

$$p = \frac{1}{\gamma} \rho_0 c_0^2 \left(\left(\frac{\rho}{\rho_0} \right)^\gamma - 1 \right) \quad (2.4)$$

where ρ_0 is the density of the fluid at the free-surface for which the equivalent pressure is $p = 0$, c_0 is the reference speed of sound of the media and γ is the polytropic constant (for water $\gamma = 7$). The set of equations (2.3) can be regrouped in a compact manner considering conservative variables:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot F - \nabla \cdot F^{vis} = S, \quad (2.5)$$

where ϕ is the vector of conservative variables, F and F^{vis} stand for convective and viscous flux tensors respectively and S the source term:

$$\phi = \begin{pmatrix} \rho \\ \rho \vec{u} \end{pmatrix}, \quad F = \begin{pmatrix} \rho \vec{u} \\ \rho \vec{u} \otimes \vec{u} + p \mathbb{I} \end{pmatrix}, \quad F^{vis} = \begin{pmatrix} 0 \\ \sigma \end{pmatrix}, \quad S = \begin{pmatrix} 0 \\ \rho \vec{g} \end{pmatrix}, \quad (2.6)$$

with equation (2.4) completing the system.

The second possible approach to model incompressible flow is to consider the incompressible Navier-Stokes equations. In that case the following assumption must be taken into account in the conservation equation in the set (2.3):

$$\nabla \cdot (\vec{u}) = 0 \quad (2.7)$$

With such an assumption, equations (2.3) are modified to:

$$\begin{cases} \frac{D\vec{r}}{Dt} = \vec{u} \\ \frac{D\vec{u}}{Dt} = \vec{g} - \frac{\nabla p}{\rho} + \frac{\mu}{\rho} \nabla^2 \vec{u} \\ \nabla \cdot \vec{u} = 0. \end{cases} \quad (2.8)$$

With this assumption for the pressure leads to a constant density throughout time and a Poisson equation must be solved to close the system. For that reason, the use of an explicit time integration scheme is no longer the direct choice. A longer discussion on this variant is available in part II. For this last approach the incompressibility is intrinsically respected. As for the first approach, it resides on the choice of the sound speed (c_0). The value of such celerity must be equal or higher than ten times the maximum velocity of the flow being treated (in terms of Mach number $Ma = \frac{|\vec{u}|}{c_0} \leq 0.1$) which ensures that the compressibility effects are purely acoustic (i.e. they can be considered to be superimposed to the main flow with almost no interaction).

2.1 Boundary Conditions

In order to solve the Navier-Stokes equations previously presented, we must consider the boundary conditions as well. These conditions allow to take into account the presence of the free-surface, walls, inflow, outflow, etc and are represented in figure 2.1.

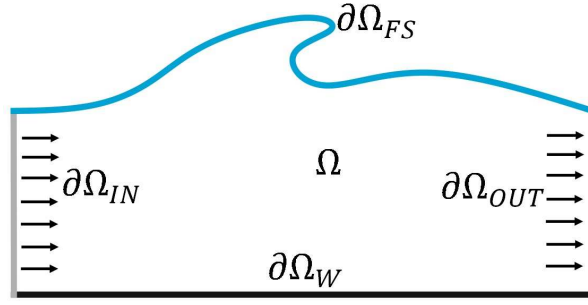


Figure 2.1: Fluid domain and its boundary conditions.

2.1.1 Free-surface

The presence of the free-surface in a flow is ensured by considering two boundary conditions: a kinematic and a dynamic one, being the position of the free-surface itself an unknown of the problem. The first boundary condition (kinematic) is ensured by imposing that the normal component of the fluid velocity is equal to the normal component of the free-surface velocity. The dynamic boundary condition concerns the force balance at the free surface and it ensures the continuity of the stress tensor across it. For the equations (2.3) (and in the absence of any model to consider the air effect and the surface tension) this condition is translated into imposing the atmospheric pressure at this surface.

2.1.2 Walls

Depending on the problem that is being solved, one may chose to consider viscous effects or not. Hence, two possible wall boundary conditions exist: free-slip (inviscid flow) or no-slip (viscous flow) wall. Mathematically, the first condition is represented by

$$\vec{u} \cdot \vec{n} = \vec{u}_W \cdot \vec{n} \quad (2.9)$$

and the no-slip one is imposed by simply forcing the wall velocity to the adjacent fluid ($\vec{u} = \vec{u}_W$). Note that we may have a viscous flow with free-slip boundary conditions, but generally that is not the case.

2.1.3 Inflow/Outflow

Some of the applications considered in this thesis require the use of inflow and outflow boundary conditions. In the formalism considered in this thesis this is translated into imposing the desired velocity at the inlet boundary and the desired pressure at the outlet.

Chapter 3

Smoothed Particle Hydrodynamics

In this chapter the mathematical basis of the Smoothed Particle Hydrodynamics method will be presented along with of the state of development of the method in the LHEEA laboratory at the time when this thesis works started. At the end of this chapter, the possible pathes to improve the method will be briefly discussed.

3.1 SPH integral interpolation

SPH has firstly been introduced by Lucy [11] and Gingold & Monaghan [12] for astrophysical purposes and later, the method was extended by Monaghan [13] to simulate free-surface flows. The SPH method is a particle method, i.e. a Lagrangian mesh-free method where the fluid domain is discretized by a set of particles. Each particle represents a volume of fluid with its own mass, density, pressure and velocity. Based on this set of particles we should be able to compute time derivatives so that the physics of the flow may be integrated on time. The next sections discuss how this interpolation is performed in the SPH formalism.

3.1.1 Continuous interpolation

In the SPH method, a given (scalar) field is interpolated using a convolution integral over the domain based on a weighting function (called kernel or smoothing function). The value of such a variable f at a certain spatial position \vec{r} can be represented as follows:

$$\langle f(\vec{r}) \rangle = \int_D f(\vec{r}^*) W(\vec{r} - \vec{r}^*, h) dV_{\vec{r}^*}, \quad (3.1)$$

where D stands for the compact support of the kernel function (represented in the equation above by W) which converges to the Dirac function as its support size goes to 0. The size of D is measured by h , often referred to as smoothing length. This parameter has a major role on the convergence of the operators as it will be specified later in this chapter and may be seen as a measure of the domain of influence of the point \vec{r} .

This kernel function may assume several shapes, all of them having a compact support and being radial. Recently, a family of smoothing functions proposed by Wendland [24] started to be largely

used in the SPH community and has shown to have very good stability and convergence features. In this thesis, if not noted otherwise, the Wendland C2 kernel is used:

$$W(q, h) = \beta \begin{cases} \frac{1}{8}(2-q)^4(1+2q) & \leftrightarrow 0 \leq q \leq 2 \\ 0 & \leftrightarrow q > 2, \end{cases} \quad (3.2)$$

where $q = \|\vec{r}\|/h$ and β is a normalization constant depending on the dimension of the problem, for 2D $\beta = \frac{7}{64\pi}$. This particular kernel has a support radius of $2h$. The β constant allows to normalize the kernel, in other words, allows the following equality:

$$\int_D W(\vec{r} - \vec{r}^*, h) d\vec{r}^* = 1, \quad (3.3)$$

which is an important condition, necessary to have second order convergence for the approximation (3.1).

The gradient convolution operator must also be defined as it is needed in equations (2.3). The gradient of a scalar f at a spatial point \vec{r} reads:

$$\langle \nabla f(\vec{r}) \rangle = \int_D \nabla f(\vec{r}^*) W(\vec{r} - \vec{r}^*, h) dV_{\vec{r}^*} \quad (3.4)$$

However, one may notice that such an operator depends on the gradient of the function f . Integrating such an equation by parts we get:

$$\int_D \nabla f(\vec{r}^*) W(\vec{r} - \vec{r}^*, h) dV_{\vec{r}^*} = \int_{\delta D} f(\vec{r}^*) W(\vec{r} - \vec{r}^*, h) \vec{n} dS - \int_D f(\vec{r}^*) \nabla W(\vec{r} - \vec{r}^*, h) dV_{\vec{r}^*} \quad (3.5)$$

Being the surface integral in equation (3.5) equal to zero (considering a domain D that does not intersect one of the borders of the global domain), the approximation for $\langle \nabla f(\vec{r}) \rangle$ follows:

$$\langle \nabla f(\vec{r}) \rangle = \int_D f(\vec{r}^*) \nabla W(\vec{r} - \vec{r}^*, h) dV_{\vec{r}^*} \quad (3.6)$$

where another characteristic of the kernel, its anti-symmetry ($\nabla_{\vec{r}^*} W(\vec{r} - \vec{r}^*, h) = -\nabla_{\vec{r}} W(\vec{r} - \vec{r}^*, h)$), is used. Equation (3.6) shows another feature of the convolution operator: the computation of the gradient of a function f depends only on the value of the function itself and of the kernel gradient, which can be computed analytically.

3.1.2 Discrete interpolation

In SPH the continuum is discretized using particles, and consequently a discrete approximation for the convolution operator is needed. Each particle represents a discrete parcel of volume which, in this work, for a certain particle i , is referred to as w_i . Following that, equation (3.1) may be written in a discrete manner:

$$\langle f(\vec{r}_i) \rangle = \sum_{j=1}^N f(\vec{r}_j) W(\vec{r}_i - \vec{r}_j, h_{ij}) w_j, \quad (3.7)$$

where N stands for the number of particles in the domain, w_j represents the volume of particle j and $h_{ij} = \frac{h_i + h_j}{2}$. As an example of the application of equation (3.7), we may cite the evaluation of the particle density given by:

$$\rho_i = \sum_{j=1}^N m_j W(\vec{r}_i - \vec{r}_j, h_{ij}). \quad (3.8)$$

where m_j represents the mass of the particle j . Similarly, equation (3.6) can also be represented in a discrete manner:

$$\langle \nabla f(\vec{r}_i) \rangle = \sum_{j=1}^N f(\vec{r}_j) \nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij}) w_j \quad (3.9)$$

The discrete integrations of equations (3.7) and (3.9) are based on the quadrature formula:

$$\int_D g(\vec{r}) d\vec{r} \approx \sum_{j=1}^N g(\vec{r}_j) w_j. \quad (3.10)$$

The kernel may also be represented in a different way, considering that it may be obtained by a revolution (refer to [25]):

$$\nabla W(\vec{r}_i - \vec{r}_j, h_{ij}) = \|\nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij})\| \vec{n}_{ij}, \quad (3.11)$$

where $\vec{n}_{ij} = \frac{\nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij})}{\|\nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij})\|}$. Hence by using $\alpha_{ij} = \|\nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij})\|$ the discrete operator (3.9) may be rewritten:

$$\langle \nabla f(\vec{r}_i) \rangle = \sum_{j=1}^N f_j \alpha_{ij} \vec{n}_{ij} w_j. \quad (3.12)$$

This equation allows an analogy to the operators used within the Finite Volumes Method (these operators will be described in part IV) where α_{ij} would be equivalent to the area of the face between two neighbor cells. In other words, with such a formulism it is possible to establish a correlation between the operators used in SPH and in FVM which will serve as basis to the developments of part IV.

Nevertheless equations (3.9) and (3.12) do not verify the nullity when applied to constant functions. In order to recover the nullity, equation (3.9) can be modified as follows, where the added discrete integral would be 0 at the continuous due to the odd nature of $\nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij})$:

$$\begin{aligned} \langle \nabla f(\vec{r}_i) \rangle &= \sum_{j=1}^N f(\vec{r}_j) \nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij}) w_j - f_i \sum_{j=1}^N I_d(\vec{r}_j) \nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij}) w_j \\ &= \sum_{j=1}^N (f_j - f_i) \nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij}) w_j, \end{aligned} \quad (3.13)$$

which in turn ensures the nullity of constant fields.

3.1.3 Convergence and Precision

The precision of these discrete operators is directly connected to the particle distribution. As in SPH the particles' positions change in time, it is difficult to ensure a good continuous convergence and precision quality throughout time. This imperfection of the particle distribution leads to errors which can be minimized. As demonstrated in [26, 27, 28], convergence in SPH comes with the following conditions:

$$\frac{h}{\Delta x} \rightarrow \infty \quad (3.14)$$

and

$$h \rightarrow 0. \quad (3.15)$$

In practice only the second condition is often used. A constant $\frac{h}{\Delta x}$ ratio is used and Δx is decreased. Even if the first criteria is not respected, for the most part of the applications, the heuristical order convergence of the whole SPH scheme is of $O(1)$ or $O(2)$ up to saturation due to the other criterion but which occurs at low enough error magnitudes. For more details on these aspects, refer to [15].

There are several methods available in the literature that allow to improve the order of accuracy of the SPH operators. In our research group, these methods have been studied and results are available in [15, 14, 29]. In the current thesis, we will only describe the two main and most used approaches, that is the Shepard and the renormalization corrections.

The first one is used to ensure that the equality

$$\sum_{j=1}^N W(\vec{r}_i - \vec{r}_j, h) w_j = 1 \quad (3.16)$$

is verified. This is done by modifying the kernel function with equation (3.17):

$$W^{Sh}(\vec{r}_i - \vec{r}_j, h_{ij}) = \frac{W(\vec{r}_i - \vec{r}_j, h_{ij})}{\sum_{j=1}^N W(\vec{r}_i - \vec{r}_j, h_{ij}) w_j}. \quad (3.17)$$

The Shepard correction restores the $O(1)$ convergence of the discrete operator (3.7). The second correction, renormalization, was proposed by Randles & Libersky [30] and studied by Oger [15]. This technique increases the order of the discrete operator (3.9) where the objective is to ensure the following equality:

$$\sum_{j=1}^N \nabla W(\vec{r}_i - \vec{r}_j, h) w_j = 0. \quad (3.18)$$

For so the discrete operator is modified:

$$\langle \nabla f(\vec{r}_i) \rangle = L_i \sum_{j=1}^N (f_j - f_i) \nabla_i W(\vec{r}_i - \vec{r}_j, h_{ij}) w_j, \quad (3.19)$$

where L_i is the renormalization matrix given by:

$$L_i = \left[\sum_j \nabla W(\vec{r}_i - \vec{r}_j, h_{ij}) \otimes (r_j - r_i) V_j \right]^{-1} \quad (3.20)$$

For more details on these correction terms, please refer to [17, 15]. From now on in this thesis, we will use simplified notation for the operators, namely:

$$\begin{aligned} W_{ij} &= W_i(\vec{r}_i - \vec{r}_j, h_{ij}) \\ f_{ij} &= f_j - f_i, \end{aligned} \quad (3.21)$$

which may facilitate the reading of this thesis. As different kernel functions may be used in SPH, each one having different support radius ($2h$, $3h$, etc), we may also use the notation R instead of h ($R = 2h$, $R = 3h$, etc).

3.2 Weakly-Compressible SPH

Traditionally the Smoothed Particle Hydrodynamics method considers the weakly-compressible equations presented in chapter 2. This variant is known as Weakly Compressible SPH (WCSPH). Here, the discrete version of these equations are introduced.

There are mainly two manners of discretizing such equations. The first, the classical approach, consists of applying directly the operators (3.7) and (3.13) into the set of equations (2.3) leading to (here without considering the viscous terms, but in this thesis we may consider these terms as well for some specific test cases):

$$\frac{d\vec{r}_i}{dt} = \vec{u}_i \quad (3.22a)$$

$$\frac{d\rho_i}{dt} = -\rho_i \sum_j^N (\vec{u}_i - \vec{u}_j) \cdot \nabla W_{ij} w_j \quad (3.22b)$$

$$\frac{d\vec{u}_i}{dt} = \vec{g} + \frac{1}{\rho_i} \sum_j^N (p_i - p_j) \nabla W_{ij} w_j \quad (3.22c)$$

$$p_i = \frac{1}{\gamma} \rho_0 c_0^2 \left(\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (3.22d)$$

i.e. we use a strong formulation of all differential operators.

However, using the discretization for the pressure gradient in equation (3.22c), the linear and angular momentum are not conserved, leading to a poor quality of results (noise in the pressure field). Hence, it is preferable to use an alternative version for the pressure gradient, as in equation (3.23), that is both conservative and symmetrical (for a particle interaction $i \leftrightarrow j$). Such a formula can be reached both by considering the Principal of Virtual Works (refer to [31]) or the Lagrange equations approach (refer to [32]).

$$\frac{\nabla p}{\rho} = \frac{1}{\rho_i} \sum_j^N (p_i + p_j) \nabla W_{ij} w_j \quad (3.23)$$

Equations (3.22) are rewritten:

$$\frac{d\vec{r}_i}{dt} = \vec{u}_i \quad (3.24a)$$

$$\frac{d\rho_i}{dt} = -\rho_i \sum_j^N (\vec{u}_i - \vec{u}_j) \cdot \nabla W_{ij} w_j \quad (3.24b)$$

$$\frac{d\vec{u}_i}{dt} = \vec{g} - \frac{1}{\rho_i} \sum_j^N (p_i + p_j) \nabla W_{ij} w_j \quad (3.24c)$$

$$p_i = \frac{1}{\gamma} \rho_0 c_0^2 \left(\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (3.24d)$$

Another approach is to follow the works by Vila and his research group ([25, 27]) which results in a SPH discretization for the Navier-Stokes equations similar to the Finite Volumes Method formalism. The result is an Arbitrary Lagrangian-Eulerian set of equations for the conservative version of Euler equations (equation (2.5) without viscous terms) which, if a fully Lagrangian description is chosen, are equivalent to equations (3.24):

$$\frac{d\vec{r}_i}{dt} \Big|_{u_0} = \vec{u}_{0i} \quad (3.25a)$$

$$\frac{dw_i}{dt} \Big|_{u_0} = \sum_j^N w_i w_j (\vec{u}_{0i} - \vec{u}_{0j}) \cdot \vec{n}_{ij} \alpha_{ij} \quad (3.25b)$$

$$\frac{dw_i \vec{\phi}_i}{dt} \Big|_{u_0} = w_i \vec{g} - \sum_j^N w_i w_j (F_i + F_j) \cdot \vec{n}_{ij} \alpha_{ij} \quad (3.25c)$$

$$p_i = \frac{1}{\gamma} \rho_0 c_0^2 \left(\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (3.25d)$$

where u_0 represent the arbitrary velocity.

3.2.1 Stabilizing Techniques

As depicted in the previous sections, the SPH formalism uses centered operators in an explicitly solved system and consequently such a formalism is unconditionally unstable. In order to restore stability it is necessary to add some numerical diffusion, in the literature we often refer to such a technique as upwinding. Each upwinding scheme originates a different WCSPH variant and, in this section, the most used techniques are briefly described.

3.2.1.1 Artificial Viscosity

The classical stabilizing technique is the one introduced by Monaghan [33]. It consists of an artificial viscosity term added to the momentum equation (3.24c) and takes the following form:

$$\begin{cases} \Pi_{ij} = -\alpha_{av} \frac{R_{ij} c_{ij}}{\rho_{ij}} \frac{\vec{u}_{ij} \cdot \vec{r}_{ij}}{\|\vec{r}_{ij}\| + \epsilon R_{ij}} \leftrightarrow \vec{u}_{ij} \cdot \vec{r}_{ij} < 0 \\ \Pi_{ij} = 0 \leftrightarrow \text{else} \end{cases} \quad (3.26)$$

where $R_{ij} = \frac{R_i + R_j}{2}$, $c_{ij} = \frac{c_i + c_j}{2}$, $\rho_{ij} = \frac{\rho_i + \rho_j}{2}$, $\vec{u}_{ij} = \vec{u}_i - \vec{u}_j$ and $\vec{r}_{ij} = \vec{r}_i - \vec{r}_j$. The parameter ϵ is introduced to avoid null denominators when a pair of particles overlaps, however, the stabilizing technique should be sufficient to avoid particle overlapping and, for that reason, this variable is rarely necessary. The coefficient α_{av} dictates how much the system will be decentered and normally varies in $\alpha_{av} = (0.0 : 0.5)$ and is chosen according to the simulated flow. This extra term is added to momentum equation the (3.24c) used in the Monaghan SPH formalism:

$$\frac{d\vec{u}_i}{dt} = \vec{g} - \frac{1}{\rho_i} \sum_j^N (p_i + p_j + \Pi_{ij}) \nabla W_{ij} w_j. \quad (3.27)$$

This solution is very simple and does not add much computational cost to the simulations. However, this technique does not correct the noisiness of the pressure field. Hence, it is very common to add up to such a technique a density filtering or a diffusion term on the mass conservation equation (refer to 3.2.1.3).

3.2.1.2 Riemann-SPH

The same kind of upwinding (artificial viscosity) can be performed within the Vila's formulation, however, this schemes permits to rather use Riemann solvers. The idea of such solvers is to consider the interactions between each pair of particles as a Riemann problem (a shock tube problem). Consequently, at a particle-to-particle interface (which is represented by α_{ij} in equation (3.12)), a Riemann problem is solved in the normal direction (defined as \vec{n}_{ij} in equation (3.12)) where the right and left states are given by ϕ_i and ϕ_j . The solution of such a problem can be computed in an exact or approximative manner. In equation (3.25c) the fluxes $(F_i + F_j) \cdot \vec{n}_{ij}$ are replaced by $G(\phi_i, \phi_j, \vec{n}_{ij}, \vec{u}_{0i}, \vec{u}_{0j})$:

$$\frac{dw_i \vec{\phi}_i}{dt} \Big|_{u_0} = w_i \vec{g} - \sum_j^N w_i w_j 2G(\phi_i, \phi_j, \vec{n}_{ij}, \vec{u}_{0i}, \vec{u}_{0j}) \alpha_{ij} \quad (3.28)$$

where G is the solution of the Riemann problem, \vec{n}_{ij} is the normal vector going from particle i to j . The main asset of this approach with respect to the others is that, here, the physics of the problem is taken into account. In other words, Riemann solver solves accurately the interaction between a pair of particles and at the same time it introduces a numerical diffusion to the system correctly propagating discontinuities in the flow, in other words, the Riemann solver is a numerical upwinding scheme based on a physical assumption whereas the artificial viscosity is a purely numerical upwinding scheme to prevent numerical instabilities. The results and pressure fields obtained with this method are a lot better than for the artificial viscosity solution on some problems, however it is computationally

more intensive than the other approaches. This SPH variant is often referred to in the literature as Riemann-SPH and the same nomenclature is used in this thesis. For broader view on this method refer to [17].

3.2.1.3 δ -SPH

Based on the Monaghan SPH approach and on the previous Artificial Viscosity stabilizing technique, Antuono *et al.* [34] proposed an SPH variant in which, besides the numerical diffusion term of equation (3.26), another term is added to the mass conservation equation (3.24a). The idea is to add a diffusive term that avoids the flickering and the instability of the pressure profiles. A series of works [35, 36, 37, 31] were dedicated to the research and understanding of such a term, and one may refer to these studies for a better description of the method.

The final result is a new SPH variant often referred to in the literature as δ -SPH. It is named after the tuning variable δ that appears on the mass conservation equation below:

$$\frac{d\rho_i}{dt} = -\rho_i \sum_j^N m_j (u_i - u_j) \cdot \nabla W_{ij} w_j + \delta_\delta c_0 h \sum_j^N \psi_{ij} \cdot \nabla W_{ij} w_j, \quad (3.29)$$

where:

$$\psi_{ij} = 2(\rho_i + \rho_j) \frac{\vec{r}_{ji}}{|\vec{r}_{ij}|} - [\langle \nabla \rho \rangle_i + \langle \nabla \rho \rangle_j]. \quad (3.30)$$

This SPH variant has shown to be very robust and is able to give very good results for a large variety of flow problems, all of that without a substantial increase of computational costs with respect to the WCSPH with artificial viscosity.

3.3 Considerations on viscous flows

Viscous flows can also be simulated within the Smoothed Particle Hydrodynamics context. Basically, two manners of discretizing the viscous operator present in the momentum conservation equation (2.3) are often encountered in SPH related literature: the one given by Morris *et al.* [38]

$$\nu \nabla^2 u_i = 2 \sum_j \nu \frac{\vec{r}_{ij} \cdot \nabla W_{ij}}{|\vec{r}_{ij}|^2} \vec{u}_{ij} w_j \quad (3.31)$$

and the one proposed by Monaghan & Gingold [39]

$$\nu \nabla^2 u_i = C \sum_j \nu \frac{\vec{r}_{ij} \cdot \vec{u}_{ij}}{|\vec{r}_{ij}|^2} \nabla W_{ij} w_j, \quad (3.32)$$

where C is equal to 6, 8 and 10 for 1D, 2D and 3D simulations respectively.

However, it has been shown by Colagrossi *et al.* [40] that the viscous operator (3.31) presents some consistency issues when applied to fluid flows involving a free-surface (it does not detect the presence of the free-surface). The second operator (3.32) does not suffer from the same issue, hence, whenever dealing with viscous free-surface flow such an operator is preferable.

A thesis was recently dedicated to this subjected in the laboratory, for a more detailed description of these operators (and others), we strongly refer the reader to [19].

3.4 Boundary Conditions

Imposing the boundary conditions necessary to simulate a flow using SPH has always been and still is one of the main challenges of the method. Considering a particle that is close to one of the borders of the domain (a free-surface, a body, a wall, etc), it is clear that the interpolation domain (kernel) of this particle is incomplete. Consequently, a technique must be employed to make it possible to fill/circumvent this incompleteness of the kernel and take into consideration the physics of the boundary condition that is being treated.

If the boundary that needs to be treated is a free-surface, two boundary conditions must be applied: the kinetic and the dynamic. The first one consists of assuring that particles initially at $\partial\Omega_{FS}$ (see figure 2.1) will remain on the boundary, and such a condition is intrinsically taken into account due to the Lagrangian nature of the SPH method. The second one consists of assuring the continuity of the stress tensor across the free-surface, which, in absence of surface tension, is imposed by applying the atmospheric pressure on the particles belonging to this boundary. It can be demonstrated that such a condition is intrinsically satisfied by the SPH operators, hence, within the weakly-compressible framework (notably because of the use of a state equation) there is no need to model it (refer to [41, 40, 15, 14, 19] for more details). Accordingly, no special treatment is necessary to correctly simulate the free-surface conditions within WCSPH (and its variants). This is one of the main advantages of the SPH method which allows it to robustly treat complex free-surfaces.

Unfortunately, the same cannot be stated for wall boundary conditions to which a treatment is necessary. There are several techniques available in the literature to make it possible to treat these boundaries. The first technique to be proposed was the one presented in [13] based on the use of Lennard-Jones forces between fluid particles and the walls. This kind of technique compensates the lacking part of the supporting kernel by forces. With the same reasoning, Marongiu [42] and Marongiu *et al.* [43] proposed a method based on a partial Riemann solver. Recently, De Leffe [19] and De Leffe *et al.* [44] proposed a similar technique based on the normal flux decomposition method.

Another family of methods consists of completing the kernel support with fictitious particles. Libersky *et al.* [45] and Colagrossi & Landrini [35] proposed the (moving) ghost-particle technique, where at every time instant particles are mirrored on the other side of the wall and particles properties are symmetrized. Another technique is to use fixed ghost particles which are modeled along all the walls from the beginning of the simulation (refer to [38, 46]).

In this thesis the boundary conditions are imposed by using the (moving) ghost-particle technique as it is the technique mainly used in the LHEEA laboratory. Alternatives have been studied and are used in the laboratory, mainly the normal flux method [19, 44]. Details on the ghost-particle treatment are given in part II of this document.

3.5 Time step and integration

The time advance within the SPH method that was presented is done explicitly, as the weakly-compressible Navier-Stokes equations allow. The advantage of an explicit integration is that there is no system to be inverted and thus each time step computation is done faster and with less use of computer memory. Unfortunately, such method is conditionally stable; in other words, in order to

keep the simulations stable, a time step limitation must be respected. The condition that dictates the size of this time step is called Courant Friedrichs Lewy criterion which is translated as:

$$\Delta t = CFL \min(\frac{h_i}{c_i}). \quad (3.33)$$

where CFL is the Courant number, chosen heuristically. This coefficient can vary according to the time integration scheme that is used. More accurate schemes allow bigger Courant numbers. In the laboratory several time integration schemes were studied throughout the past years. In Doring's thesis [14] some results regarding the precision of the different schemes can be found. Historically, in the laboratory, the time advance is done by applying a 4th-order Runge Kutta scheme which allows a $CFL = 0.75$. If a viscosity is taken into account, a viscous CFL condition must also be used as it is shown in section 6.1.

Chapter 4

Possible Improvements

Given the state of the art description of the method, we may draw some remarks concerning the limitations of SPH. First of all, as stated, SPH is suited and designed for violent flows where small time steps are physically necessary and welcome. So far, in the laboratory, the tradition is to consider a weakly-compressible description of the fluid, which implies time step restrictions (CFL) dictated by the stability limits of the method. For that reason, such a weakly-compressible description imposes some limits to the kind of phenomena that can be simulated. One could desire to simulate slow phenomena in parts of the flow, such as wave propagation phenomena, where the compressibility of the flow is almost null, and due to that, the weakly-compressible SPH approach is not the most suited method (even if it is feasible). Generally, slow dynamics simulations last for at least many seconds, and using WCSPH would involve small time steps which would imply high CPU times/costs. For that reason, it is reasonable to consider a fully-incompressible approach in which, theoretically, the time steps are less restricted by stability criteria.

Secondly, meshed Eulerian (or Arbitrary-Lagrangian-Eulerian) methods such as FVM, FEM, FDM, etc, have a very interesting feature: it is possible to adapt the domain discretization (i.e. the mesh) to the problem/flow that is being solved, where a finer mesh will be used in the regions of interest. This is possible because these methods are Eulerian, that is, the mesh does not move (or moves slowly, considering an ALE approach). In SPH, particles move in space with the flow, which makes impossible (for most simulations) to use the same approach as in meshed methods. Actually, such an approach was studied (referred to as ‘variable-h’, which will be addressed later), but has a limited range of application. Therefore, an alternative must be sought to reach the same level (not kind) of adaptivity in SPH as the one found for meshed methods. The main challenge is to deal with the Lagrangian nature of the method, which implies to be able to dynamically refine the particles when needed, which is close to the AMR (Adaptive Mesh Refinement) [47, 48] technique in mesh-based methods.

Lastly, we may cite the lack of convergence of SPH operators. The SPH discrete operators have a limited convergence rate and in a way to restore 0^{th} and 1^{st} order convergence, corrections must be used, cf. Shepard and renormalization in section 3.1.3. However, these corrections are limited and sometimes produce undesired effects in the flow. We may also cite works done by Colagrossi *et al.* [49] and Marrone *et al.* [50] going in the direction of improving SPH interpolators, where a packing algorithm was used in order to get a more accurate particle distribution. However, this solution

does not rectify the fact that SPH operators lack of convergence. In order to improve the order of convergence of SPH operators, a deep change in its structure must be proposed and studied since no simple corrections proved to be both efficient and versatile in all situations. However, that would imply to work on a different method, which shall not result on losing the most interesting property of the method, i.e., the fact that it intrinsically takes into consideration the presence of the free-surface.

In an attempt to solve, to address, to point directions, to understand these main issues, we draw three directions of research that will be explored in this thesis:

- **Fully-incompressible approach**

The first direction is to consider changing equations that are used in the WCSPH approach, i.e., a change in the physical assumptions. The laboratory has a history of using the weakly-compressible approach, but recently a fully incompressible version of the SPH method has appeared [51] (considering the condition of equation (2.7)) and started to be deeply studied.

The literature dedicated to this method states that by using a fully incompressible approach, it is possible to overcome the CFL restriction of WCSPH and increase the time step of a factor of $O(10)$ times the one used in the weakly-compressible approach. Moreover, the quality of results is impressive and comparable to the ones obtained by Riemann-SPH, even though before the beginning of this thesis (2010) such a comparison had not been performed yet. Additionally, by considering a fully incompressible SPH approach we may as well enlarge the scope of applicability of the SPH method and simulate slow dynamics problems, like wave propagation, which is difficult to treat with the weakly-compressible approach.

Part II of this thesis is fully dedicated to the analysis of such an Incompressible SPH approach. It will be shown that the ISPH, as developed in the literature, cannot attend to all of our expectations (which were cited in the last paragraph). As a consequence of this analysis, a new explicit incompressible SPH method is proposed and validated.

- **Adaptivity**

Secondly, we may consider a change in the way that the domain is discretized within SPH. As stated, in SPH, the fluid domain is modeled using particles. Traditionally, SPH simulations consider a constant particle distribution for the entire domain, i.e., a constant Δx . However, alternatives do exist, as, for example, the ‘variable-h’ discretization, where the particle distribution gradually evolves from a fine distribution to a rough one. This approach was studied in the laboratory (see [15, 52, 17]) and has shown to be performing for certain test cases. Nevertheless, this approach is limited by the Lagrangian nature of the method. In this technique, the particle distribution is defined as an initial condition and as particles evolve in time they evolve in space also, which can originate unstable particle distributions (particles with different Δx get mixed).

Recently, a dynamic refinement method has been proposed by Feldman *et al.* [6], which allows to dynamically (depending on time and/or space) refine a particle and increase the precision and efficiency of the SPH temporally and/or locally. Part III of this thesis is dedicated to an analysis of such a technique and its validation in test cases within the scope of application of the SPH method (violent flows). We also verify its applicability to more precise SPH methods such as Riemann-SPH

and δ -SPH. Still in part III, a new technique that allows to derefine particles is presented. The goal of this new technique is to improve the accuracy and efficiency of SPH.

- **Hybrid SPH method**

Despite the good results and the gain of efficiency and accuracy given by the last two directions of improvements, they do not constitute a relevant change of paradigm within the SPH method. Moreover, with these improvements, the classical problems related to the SPH operators are not solved. In the last part of this thesis, we consider a deep change of the SPH formalism. This part may be seen as a consequence of the conclusions of the two first directions pointed out previously.

Taking as basis what was stated in section 3.1 regarding the similarity of SPH discrete operators to the Finite Volume method ones, and considering the recent developments on Voronoi-based Lagrangian FVM [53], part IV treats of a new method that couples a Lagrangian FVM to SPH. The objective of such a coupling is to benefit of the advantages of the Finite Volume method (notably its precision) and the Smoothed Particle Hydrodynamics method (notably the fact that the free-surface conditions are intrinsically taken into account by its operators, which is a hard task in FVM).

We may add to these objectives the desire of better understanding the SPH method itself. Within this method, even though the volume changes in time, it is not explicitly accessible (from the meshless assumption) and researchers have always been concerned by knowing how it evolves and changes during a simulation. The fact of having a mesh (from the Voronoi tessellation) will (maybe) allow us (and SPH practitioners) to have a better view of volume evolution within SPH.

In order to increase the accuracy of the SPH method, other techniques, besides the Voronoi-based Lagrangian FVM, could be used like an Eulerian Cartesian-grid FVM method, a FEM method (e.g. the Particle Finite Element Method [54]) or even an almost meshfree method like the Finite Volume Particle Method [55] to couple to SPH. The Voronoi-based FVM was chosen in this thesis due to the its simplicity and its similarity to the SPH (mainly the algorithm) and its cell-centered feature which will allow us to trace a better parallel to the SPH method.

Part II

Incompressible SPH

Chapter 5

Introduction

As stated in chapter 2, besides of the weakly-compressible approach one may rather model an incompressible flow based on equations (2.8). Within the SPH formalism, the very first fully incompressible approach based on the projection method by Chorin [56] was originally presented by Cummins & Rudman [51] and since it has been used to model free-surface flows [57, 58, 59, 60], wave breaking problems with the inclusion of a turbulence model [61], viscous flows [58, 62], and multi-phase flows [63, 64]. The method can be divided into three main variants: the first, presented in [51], involves a dependency on the velocity divergence in the right hand side (RHS) of the Poisson equation. This variant is referred to as ISPH_DF in this thesis (DF standing for Divergence Free). In the second variant, the RHS of the Poisson equation depends on a density prediction as outlined by Shao & Lo [57], referred to as ISPH_DI (DI standing for Density Invariant). The third variant is a hybridization of ISPH_DF and ISPH_DI, solving two Poisson equations (PPE) as described by Hu & Adams [63], and referred to as ISPH_DFDI (Divergence Free and Density Invariant).

A detailed comparison of these three variants has been performed by Xu *et al.* [62]. They showed that ISPH lacks of accuracy for certain flow problems (free-surface flows mainly), due to errors associated with the truncated kernels. In order to prevent this loss of accuracy, they proposed a FVPM-like (Finite Volume Particle Method) shifting algorithm [55] applied to the divergence free variant. This technique showed to perform very well both on viscous and free-surface flows. The same author proposed a Peclet number-based free-surface stabilization technique [65], where the viscosity of the particles located near the free-surface is artificially increased. This technique has shown to give good results, mainly for wave propagation phenomena. Recently, a new approach has been developed by Lind *et al.* [60] based on Fick's law of diffusion and very good results were obtained for a large variety of fluid flows. In [66] a repulsive component similar to a Lennard-Jones Potential equation was used in the advection to prevent particles fracturing, and therefore stabilized the method.

Other authors proposed to use some alternative corrections to increase the accuracy of the method, involving a greater effort to get in turn a more accurate solution of the pressure Poisson equation (PPE). Khayyer *et al.* [67, 68] proposed to correct the kernel gradient through the use of renormalization. In the larger context of Lagrangian particle methods, the MPS (Moving Particle Semi-implicit) [69, 70] method can be pointed out which is very similar to Incompressible SPH, using the same solving procedure (projection method). Strong efforts were addressed to improving the MPS method by

Khayyer & Gotoh [71], where a high-order Laplacian operator and a renormalization-like operator were used to improve the PPE accuracy.

Another recent approach was proposed by Bøckmann *et al.* [72] where a new time integration scheme that increases the accuracy of the method was proposed, together with a new treatment of the free-surface boundary conditions. In this approach, the Laplacian operator was slightly changed to take into account the presence of the free-surface. Good results were obtained, however the range of problems to which the improvement has been tested was limited.

This Incompressible SPH method is presented and considered as an alternative to the classical weakly-compressible approach and its variants. One of the main advantages of such a technique resides on the fact that the time step is no longer restricted to a sound speed CFL, and therefore higher time steps, with respect to the ones used in WCSPH, may be used (which in turn may lead to lower CPU costs). Also, as a PPE is solved, an incompressible flow (which can be translated into having constant volume or density for all the particles) is ensured throughout the whole simulation, contrary to what may be verified when using the weakly-compressible approach, where density fluctuations are observed. Thus, the range of applicability of the SPH algorithm could be increased by using the incompressible approach.

The combination of these features make the method *a priori* attractive. However, we could not find in the literature an objective analysis of these ISPH algorithms, such as a detailed comparison study between ISPH and WCSPH (and/or its variants) verifying if results can really be improved and if the time step can be substantially increased as theoretically permitted. In other words, in order to better justify the use of such a method it is necessary to perform an analysis to determine if the trade-off between accuracy of the solution and CPU costs is more interesting than the one found for the WCSPH approach.

In this thesis, such an analysis is performed. Also, a series of improvements are proposed to the Incompressible SPH method (that correct some of the problems depicted in the last paragraph). The first involves the enforcement of the boundary conditions at the free-surface which is necessary to solve the PPE: a recently presented accurate particle detection algorithm [1] is used to enforce these boundary conditions which eases the inversion of the linear system needed by such an incompressible approach. Also, the use of a simple stabilization technique for free-surface flow problems is proposed and validated.

Finally, it is shown that a good compromise between accuracy and CPU costs is difficult to achieve within the traditional ISPH method. In order to circumvent this downside, which has been little discussed in the literature, a new incompressible SPH algorithm is proposed, based on an explicit solution of the PPE equation. Such an explicit solution does not require the imposition of free-surface conditions, as for the weakly-compressible formalism.

Summarizing, in this part, firstly the semi-implicit ISPH as presented in the literature is presented along with some corrections to circumvent some well-known problems of method. This presentation is then completed by a performance analysis of the method. Lastly, the explicit ISPH method developed in this thesis is presented where a full validation process is performed. We also show some results for naval and ocean engineering problems.

Chapter 6

Semi-Implicit ISPH

In order to get incompressible pressure and velocity fields, the incompressible Navier-Stokes equations (2.8) are split into two distinct parts following the projection method proposed by Chorin [56]. The first part contains the viscous and body forces, and the second one contains the pressure force. Between these two parts, the pressure Poisson equation is solved, from which the incompressible pressure fields result.

In the first part, considered to be the prediction part, intermediate particle velocities are computed using the following equation:

$$\vec{u}_i^* = \vec{u}_i^n + \left(\nu \nabla^2 \vec{u}_i^n + \vec{g} \right) \Delta t, \quad (6.1)$$

leading to intermediate particle positions, evaluated as:

$$\vec{r}_i^* = \vec{r}_i^n + \Delta t \vec{u}_i^*. \quad (6.2)$$

The final velocity field is obtained by applying the pressure gradient term from the momentum equation in (2.8):

$$\vec{u}_i^{n+1} = \vec{u}_i^* - \frac{\Delta t}{\rho_0} \nabla p_i^{n+1}. \quad (6.3)$$

By applying the divergence operator to (6.3) and knowing that the incompressibility condition (equation (2.7)) must be respected, we get the pressure Poisson equation (PPE) for the divergence free variant of the incompressible SPH algorithm:

$$\nabla \cdot \left(\frac{1}{\rho_0} \nabla p_i^{n+1} \right) = \frac{1}{\Delta t} \nabla \cdot \vec{u}_i^*. \quad (6.4)$$

The pressure field obtained from (6.4) is incompressible, and is used to update particle velocities and positions in equations (6.3) and (6.5), respectively.

$$\vec{r}_i^{n+1} = \vec{r}_i^n + \Delta t \left(\frac{\vec{u}_i^{n+1} + \vec{u}_i^n}{2} \right). \quad (6.5)$$

For the ISPH_DI variant, the PPE is obtained by applying the continuity assumption (equation (6.6)) into equation (6.4).

$$\nabla \cdot \vec{u}_i^* = - \left(\frac{1}{\rho} \frac{d\rho}{dt} \right), \quad (6.6)$$

leading to the following PPE:

$$\nabla \cdot \left(\frac{1}{\rho_0} \nabla p_i^{n+1} \right) = \frac{\rho_0 - \rho_i^*}{\rho_0 \Delta t^2}, \quad (6.7)$$

where ρ_i^* is the intermediate density computed at the intermediate particle positions \vec{r}_i^* (using equation (3.8)). Particle velocities and positions are updated via the same time integration as in the ISPH_DF variant: equations (6.3) and (6.5). A more global version of the PPE was presented by Zhou *et al.* [73], as:

$$\nabla \cdot \left(\frac{1}{\rho_0} \nabla p^{n+1} \right)_i = (1 - \alpha_{ISPH}) \frac{1}{\Delta t} \nabla \cdot \vec{u}_i^* + \alpha_{ISPH} \frac{\rho_0 - \rho_i^*}{\rho_0 \Delta t^2}, \quad (6.8)$$

in which both ISPH variants are included in only one equation with a factor α_{ISPH} controlling whether the divergence free ($\alpha_{ISPH} = 0$) or the density invariant ($\alpha_{ISPH} = 1$) formalism is adopted. In the same article, the authors outlined that one may get best results by using an intermediate value of $\alpha_{ISPH} \sim 0.1$, when simulating violent wave phenomena.

Note that in equation (6.8) a Laplacian operator is needed to completely discretize the pressure Poisson equation. This operator is the same as the one used in equation (3.31) for the viscosity. This operator has already been applied to ISPH by Xu *et al.* [62] and is written as:

$$\nabla^2 p_i = 2 \sum_j \frac{r_{ij} \cdot \nabla W_{ij}}{|r_{ij}|^2} p_{ij} w_j, \quad (6.9)$$

The semi-implicit ISPH algorithm appears as an interesting alternative to the use of WCSPH. As pointed out in the literature, the major advantage of this algorithm is to combine the possibility of using larger time steps together with equivalent (or even better) results (see for instance [58, 62, 59]). However, no comparison was performed to more precise and performing WCSPH algorithms like Riemann-SPH and δ -SPH, which are now frequently used. Indeed, the latter algorithms which are relatively recent, are known to allow larger time steps and much less noisier pressure signals than the very first WCSPH schemes appeared. It was of our understanding that ISPH had to be compared to these more recent methods in order to get fairer and clearer analysis.

6.1 Time step

In WCSPH and in its variants (Riemann-SPH and δ -SPH), the time step has to respect a CFL (Courant-Friedrichs-Lewy) condition depending on the sound speed (c_0), see equation (3.33). In addition, if viscous terms are considered, a viscosity-based CFL condition should be imposed. Unlike WCSPH, the time step in ISPH is not limited by c_0 , but only by the fluid velocity, due to its semi-implicit character:

$$\Delta t_{convective} = CFL \frac{h}{||u_{max}||}, \quad (6.10a)$$

$$\Delta t_{viscous} = CFL \frac{h^2}{\nu_0}, \quad (6.10b)$$

$$\Delta t_{max} = \min(\Delta t_{convective}, \Delta t_{viscous}) \quad (6.10c)$$

where CFL is the Courant number, chosen heuristically. A computational efficiency factor of $O(10)$ is thus expected between the WCSPH and the ISPH methods, since c_0 is usually chosen as $c_0 = 10u_{max}$ in the pseudo-incompressible approach. Note that, when the viscous constraint dominates, both ISPH and WCSPH may use the same time step, if the same CFL coefficient is used.

Literature points out $CFL = 0.25$ as a standard value for semi-implicit ISPH. However, within WCSPH, Riemann-SPH or δ -SPH it is possible to reach at least $CFL = 0.75$ provided that a high-order time-integration scheme is adopted (4^{th} order Runge-Kutta scheme for instance [19]). This leads us to state that in the literature, the ISPH method was validated with a time step $O(3)$ times greater than the latter cited algorithms instead of $O(10)$ as expected. Section 6.5 provides a discussion dedicated to check whether the ISPH time steps can be extended to higher values.

Prior to this analysis, some particularities of the method regarding boundary conditions and stabilization techniques should be discussed.

6.2 Boundary Conditions

We describe here the procedure adopted to impose the different types of boundary conditions that are necessary for the test cases: free-slip, no-slip, inflow and outflow boundary conditions, and also the free-surface treatment.

6.2.1 Free-slip and no-slip walls, inlet and outlet

The ghost-particle method is used in the present study to apply the wall (free-slip or no-slip), inflow and outflow boundary conditions. Within this technique, ghost particles are created outside the wall by mirroring fluid particles as proposed in [45] or [35]. This method has also been widely used in ISPH to enforce boundary conditions [51, 62]. In order to enforce a free-slip boundary condition, velocities and pressures are symmetrized on the ghost particles as on the left part of figure 6.1.

Recently De Leffe *et al.* [74] proposed a modification of the no-slip boundary condition in WCSPH where the no-slip condition is applied only for the viscous part of the Navier-Stokes equations, whereas free-slip condition is applied for the hyperbolic part. Note that this treatment prevents from additional unphysical shear stresses. The same procedure is used in the present ISPH formulation, where no-slip condition is applied for the viscous terms in equation (2.8) while free-slip condition is imposed for the PPE. No-slip boundary conditions is enforced by centrally mirroring the velocities on the ghost particle as it is shown in the right-side of figure 6.1. For the pressure the values of fluid particles are copied at the ghost particles in both situations, where an especial attention must be given for flows under the action of a gravity, i.e., an hydrostatic continuity in the direction of the gravity must be

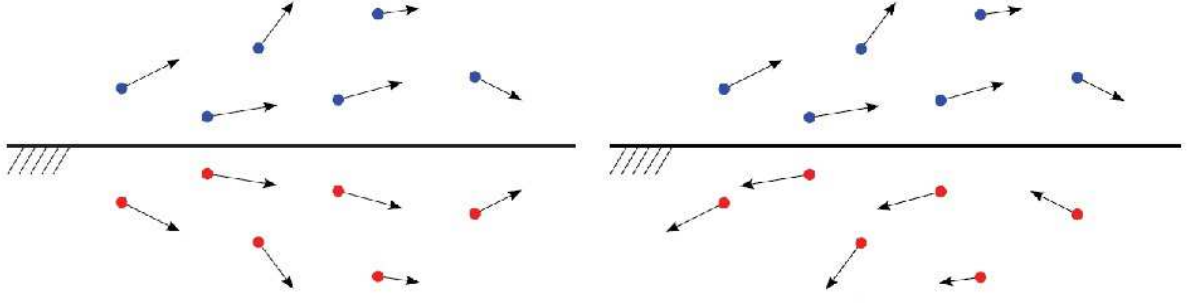


Figure 6.1: Ghost particles (red) are mirrored from the fluid particles (blue) to enforce: on the left a free slip boundary condition and on the right a no-slip boundary condition.

respected when creating the ghost particles. Hence, Neumann boundary conditions needed by the PPE are respected.

Regarding the inflow and outflow boundary conditions we use the scheme from [74] where for the first one ghost particle are used and acoustic waves are filtered out avoiding reflection.

6.2.2 Free-surface

Regarding the special case of the free-surface boundary condition, some issues must be discussed in detail: the first regards its singularity (due to the use of SPH operators), and the second one regards the application of the boundary condition itself (which is necessary for the ISPH approach).

6.2.2.1 Singularity

One may notice that when using the density invariant ISPH (ISPH_DI) method, the RHS of equation (6.7) must take into account the presence of the free-surface. The intermediate density ρ_i^* is evaluated through equation (3.8) and as we get closer to the free-surface, the kernel support starts lacking of supporting particles which implies an underestimation of the density, leading to a higher RHS term for the PPE for these particles as $(\rho_0 - \rho_i^*)$ increases. If nothing is done regarding this issue, unphysical solutions will result from the system solution. In the present study, a simple solution is used to prevent this behavior: before updating particle velocities and positions to the intermediate state with equations (6.1) and (6.2), an initial density is computed using (3.8) and then the PPE, equation (6.8), can be rewritten as:

$$\nabla \cdot \left(\frac{1}{\rho_0} \nabla p^{n+1} \right)_i = (1 - \alpha_{ISPH}) \frac{1}{\Delta t} \nabla \cdot \vec{u}_i^n + \alpha_{ISPH} \frac{\rho_i^n - \rho_i^*}{\rho_i^n \Delta t^2}, \quad (6.11)$$

where ρ_i^n is obtained from the positions \vec{r}_i^n .

6.2.2.2 Detection algorithm

Within the projection method, the PPE must be solved, which requires a well bounded system. In addition to the Neumann boundary conditions at walls, the free-surface dynamic condition must be

imposed (the kinematic one is inherent to the Lagrangian formalism). In the particular case of a monophasic flow, this dynamic condition is applied by imposing $p = 0$ at the particles belonging to the free-surface. Hence such particles should first be detected correctly. This characteristic of semi-implicit ISPH is demanding in terms of computational resources and might be seen as a drawback with respect to WCSPH, where the free-surface dynamic condition is implicitly satisfied, see [41].

Considering the ISPH and MPS [69, 70] methods, two main techniques can be found for imposing such a condition. The traditional way [57, 61, 67, 68, 75, 76] is to detect the free-surface by evaluating the particle intermediate densities and to consider a particle to be at the free-surface if it satisfies $\rho_i^* < 0.99\rho_0$. A similar volume-based technique was proposed in [77]. Later, a more sophisticated method has been proposed in [58] in which the detection depends on the divergence of particle positions, where a threshold value is used to determine whether the particle is located on the free-surface. Ever since, such procedure is the most widespread in the literature (see for instance [60] and a similar particle position divergence-based method [72]).

However, considering a particle that is detected as belonging to the free-surface at a given instant, due to errors related to the SPH operators used for this detection, this particle might be captured in an intermittent way, leading to flip-flop effects between successive time steps. This results into an ill conditioned system and a noisy pressure field solution. Therefore, within this study, a more accurate free-surface detection algorithm based on the particle distribution geometrical features proposed by Marrone *et al.* [1] is used within our ISPH algorithms.

The latter method is divided into two steps: first the eigenvalues for the renormalization matrix are computed for each particle and then, based on the higher eigenvalue (λ), we may split the fluid domain into three subsets of particles: \mathbb{E} represents the particles that belong to jets or drops of water; \mathbb{B} those who are close to the free surface and; \mathbb{I} those who are on the interior of the fluid. The division is done as follows for a cubic spline kernel in 2D. The threshold values may be adapted for other kernels.

$$\begin{aligned} i \in \mathbb{E} &\iff \lambda \leq 0.20 \\ i \in \mathbb{B} &\iff 0.20 < \lambda \leq 0.75 \\ i \in \mathbb{I} &\iff 0.75 < \lambda \end{aligned}$$

This first step being done we may select only the subset of particles \mathbb{B} to be rechecked in the second part of the algorithm as the subset $\mathbb{E} \in \mathbb{F}$, where \mathbb{F} represents the particles on the free-surface. The method is based on the fact that the summation of the kernel gradient, for the particles close to the free surface, give a good approximation of the local normal vector to the free surface. This evaluation can be even more accurate by using the relations that follow:

$$\vec{n}_i = \frac{\vec{v}_i}{|\vec{v}_i|}, \quad \vec{v}_i = -L_i \sum_j \nabla W_{ij} w_j, \quad (6.12)$$

where L_i is the renormalization matrix given by equation (3.20). Based on this normal vector a scan region can be created for each particle (see figure 6.2) and within this region a search will be performed to look for neighbor particles. If no neighbor is found in it, the candidate particle belongs to the free surface, otherwise it belongs to the fluid.

Denoting by $i \in \mathbb{B}$ the particle under investigation and by j the neighbor particle, for each i we also define the point T at a distance h from i in the normal direction and $\vec{\tau}$ a perpendicular vector

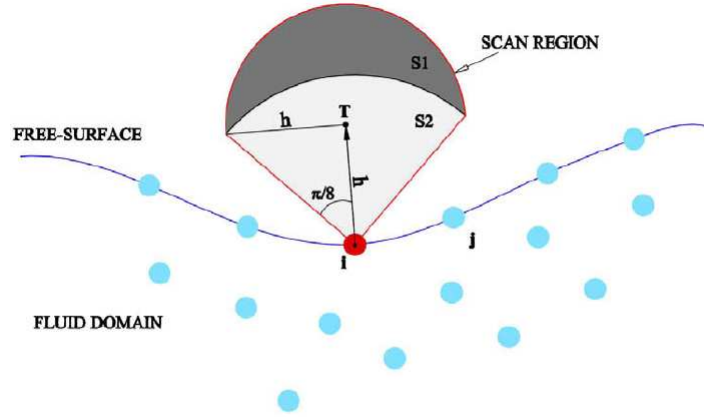
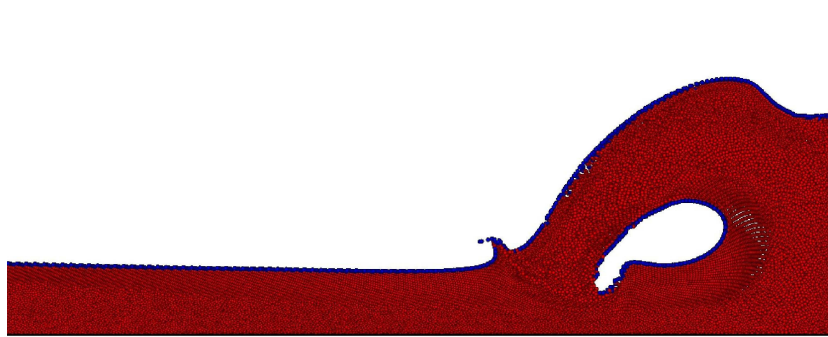


Figure 6.2: Search region for the 2D algorithm [1].

to \vec{n} . Then we can apply the search algorithm to define if the particle belongs to the free surface \mathbb{F} :

$$\begin{aligned}
 j \in \mathbb{N} \left[|\vec{r}_{ij}| \geq \sqrt{2h}, |\vec{r}_{jT}| < h \right] &\Rightarrow i \notin \mathbb{F} \\
 j \in \mathbb{N} \left[|\vec{r}_{ij}| < \sqrt{2h}, |\vec{n} \cdot \vec{r}_{jT}| + |\vec{\tau} \cdot \vec{r}_{jT}| < h \right] &\Rightarrow i \notin \mathbb{F} \\
 \text{otherwise} &\Rightarrow i \in \mathbb{F}
 \end{aligned}$$

The method has its 3D version where the second part of the algorithm is slightly changed. For more details, refer to [1]. Figure 6.3 illustrates the efficiency of this algorithm, through the example of a dam-break.

Figure 6.3: Free-surface particles (blue) detected by Marrone *et al.* [1] algorithm on a dam-break test case

6.3 Stabilizing method

It is stated in the literature that ISPH is unstable in the presence of a free-surface, where particles tend to move more freely. In order to correct this instability, several stabilizing methods have been proposed in the literature (see [78, 62, 65]). All of these techniques seem to be very performing, especially the

FVPM-like technique [62]. However its implementation can be very tricky. Other techniques [65] proposed to artificially increase the viscosity of particles located close to the free-surface, but the physical justification of such a procedure is questionable. Hence, we decided to use the XSPH method [79]. This correction smoothes the particle velocities in a consistent manner and is defined as follows:

$$[\Delta u_i]_{XSPH} = \varepsilon_{XSPH} \sum_j \frac{m_j}{\rho_i + \rho_j} (u_j - u_i) W_{ij}, \quad (6.13)$$

where ε_{XSPH} is usually taken as 0.5, but may vary according to the simulation.

The main advantage of this method is its simplicity compared to the already existing stabilizing techniques. In our experience, it showed to be as efficient as any other technique when applied to Incompressible SPH, and did not interfere significantly with the physics of the flow.

Other authors use more precise operators [67, 71, 77, 72] to stabilize free-surface flows. However, the chosen XSPH correction has been considered to be efficient, versatile and computationally not expensive, enough for not having considered more complex alternatives.

6.4 System solving

In SPH formulation the PPE (equation 6.11) can be written as:

$$\sum_j (p_j - p_i) A_{ij} = B_i, \quad (6.14)$$

where A_{ij} is a matrix-like variable depending on the formulation used to discretize the Laplacian operator, B_i is RHS vector of the PPE. Using equations (3.13), (6.9) and (6.11) one gets:

$$A_{ij} = 2 \frac{r_{ij} \cdot \nabla W_{ij} w_j}{|r_{ij}|^2}, \quad (6.15)$$

$$A_{ii} = - \sum_{j \neq i} A_{ij}, \quad (6.16)$$

$$B_i = (1 - \alpha_{ISPH}) \frac{1}{\Delta t} \sum_j \vec{u}_{ij} \cdot \nabla W_{ij} w_j + \alpha_{ISPH} \frac{\rho_i^n - \rho_i^*}{\rho_i^n \Delta t^2}. \quad (6.17)$$

In order to solve the PPE one has to use an iterative solver like BiCGStab or GMRES. Within this thesis, we use the first algorithm as it is the most used in literature to solve the PPE. The residual limit was chosen to be $\|Res\| = 10e - 6$ for all the simulations that were performed in this thesis and is computed on the following manner:

$$\|Res\| = \sqrt{\sum_{i=1}^N Res_i^2}, \quad (6.18)$$

where $Res_i = B_i - \sum_{j=1}^N (p_j - p_i) A_{ij}$.

6.5 Performance analysis

The main advantage of using fully implicit algorithms to solve Navier-Stokes equations resides in the possibility to overcome the time step limitation of an explicit algorithm, which leads in Eulerian mesh-based methods (like the standard Finite Volume Method - FVM) to time steps that can be, depending on the flow, of several orders of magnitude (100, 1000,...) bigger than the ones used in an explicit solution of the same flow.

However, the solution algorithm of ISPH methods found in the literature is semi-implicit only since particles are evolved in a Lagrangian way directly from the velocity field, and not fully implicit. This makes a major difference. In fact, the same projection method on which ISPH resides has been largely used and studied within the Finite Element Method (FEM) [80, 81, 82, 83]. Gresho *et al.* [82] showed that the semi-implicit projection algorithm must respect a convective stability limit where $CFL \leq O(5 - 10)$ which corresponds to 5 - 10 times the time step given by an explicit FEM solver. Later, Christon & Carroll [83] stated that $CFL \simeq 5$ is a reasonable trade-off between accuracy and computational cost for FEM-based CFD algorithms using Chorin's projection technique [56]. In this section, it will be shown that the same conclusion can be depicted for the ISPH method through a performance analysis: namely, the ISPH time step limitation with respect to WCPSH is similar to the one found within FEM when comparing semi-implicit algorithms to an explicit one.

This analysis is performed using the 2D dam-break test case presented in figure 7.12 (see section 7.4.2), which was chosen as benchmark because of the presence of the free-surface and due to its violent nature which is in the scope of applicability of the SPH algorithm. We performed a total of five simulations, starting with the WCSPH time step (equation (3.33) with $c_0 = 30 \text{ m s}^{-1}$ and being the flow quasi-inviscid, equation (6.10b) does not apply). The CFL number chosen for this first simulation was $CFL_0 = 0.75$, and was linearly increased up to 8 times this value in four other simulations. Note that $CFL_0 = 0.75$ corresponds approximately to $1/3^{rd}$ of the theoretical time step in ISPH (as seen in the literature using $CFL = 0.25$, see section 6.1). Considering the results in the ISPH literature, the ISPH theoretical stability limit is found between $4CFL_0$ and $5CFL_0$. All the simulations were performed with a fixed XSPH correction factor $\varepsilon = 0.2$, with an average of 40 neighbor particles and with a particle space resolution of $\Delta x = 0.01 \text{ m}$.

Figure 6.4-a shows the pressure signal (at probe $p1$) given by semi-implicit ISPH in its divergence-free variant compared to experimental results [84] for the different CFL parameters tested. This figure shows that a high level of noise is present in parts of the pressure signal. To ease the readability, a smoothed version is presented in figure 6.4-b. From the stability point of view, the results tend to get much poorer from $4CFL_0$ ($CFL = 3$) and higher values. Note that all simulations with higher CFL numbers fail to capture the water impact against the wall and the wave breaking phenomenon occurring at non-dimensional instant $T = 6$. Moreover, the noise level increases with the CFL number.

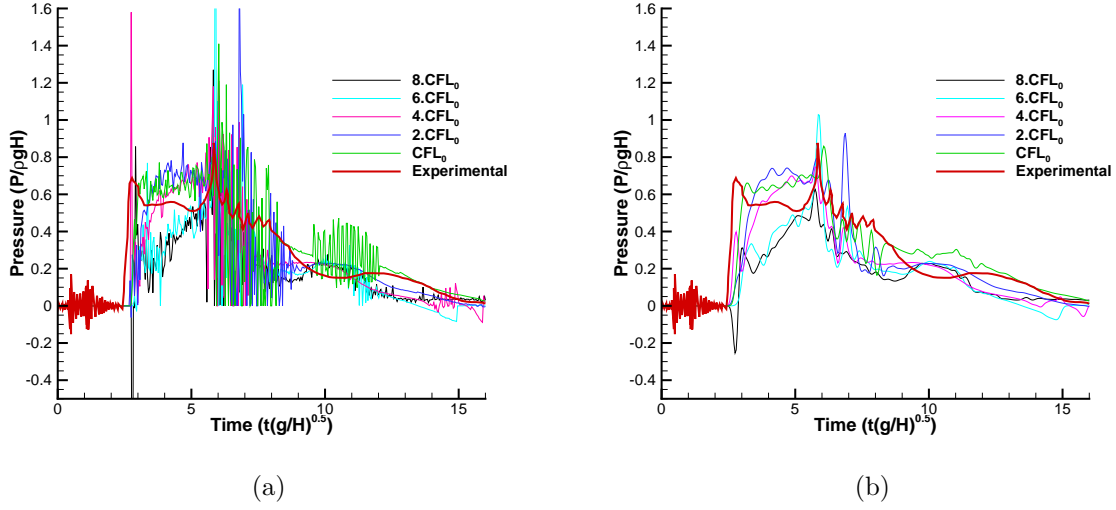


Figure 6.4: Pressure signal comparison between semi-implicit ISPH_DF and experimental results for 5 different CFL numbers: (a) Non smoothed signal and (b) smoothed signal.

Table 6.1 shows CPU costs comparison between semi-implicit ISPH and WCSPH. On this test case, ISPH starts presenting lower CPU costs than WCSPH only for $CFL \geq 4CFL_0$. But the simulation is clearly unstable beyond this limit. The high CPU costs per time step for the ISPH algorithm (with respect to WCSPH) are mainly caused by the need of solving the PPE.

Nevertheless, good results can be obtained by this incompressible approach, but it is only possible by using a time step close to the WCSPH one. For higher time steps, the incompressible approach loses accuracy. The direct conclusion is: the use of higher CFL with the ISPH formulation is possible, but this unfortunately leads to poor-quality or unstable simulations.

Table 6.1: CPU time comparison between semi-implicit ISPH and WCSPH.

Method	CFL	CPU time (seconds)	%WCSPH
WCSPH	CFL_0	200.1	-
Semi-implicit ISPH	CFL_0	762.2	281.0%
Semi-implicit ISPH	$2 \cdot CFL_0$	377.1	88.5%
Semi-implicit ISPH	$4 \cdot CFL_0$	206.0	3.0%
Semi-implicit ISPH	$6 \cdot CFL_0$	135.7	-32.2%
Semi-implicit ISPH	$8 \cdot CFL_0$	100.7	-49.7%

6.6 Discussion

In the literature, ISPH is presented as an alternative to the weakly-compressible algorithm. Theoretically this is justified, since the method is not restrained by an acoustic CFL condition. Actually, the time step is limited by the semi-implicit nature of the algorithm. However, the above study has heuris-

tically demonstrated that the latter semi-implicit stability restriction seems to be close to the acoustic one if good-quality results are desired, which is similar to what was observed for the FEM. Note that a fully-implicit ISPH scheme would not suffer from this issue. However, due to the Lagrangian character of the SPH method, a fully-implicit ISPH algorithm would be very difficult to design efficiently.

Indirectly, this paradigm has already been verified by other authors who improved the precision of the algorithm by using higher order operators for the Laplacian as Khayyer *et al.* [67, 68] who reached better results but still with a certain amount of noise in the pressure signal. The use of preconditioners while solving the PPE as in [62] can be tried out, but this only speeds up the solution of the equation and does not improve the accuracy of the algorithm. Better trade-offs could be reached but not changing much the conclusion drawn here.

This discussion can be extended to the MPS method, which is very similar to ISPH (see [85, 86]). They are both fully Lagrangian and use the same projection method. The main difference resides in the discrete operator. However, modifying the operators does not prevent the occurrence of such instabilities, as already verified by Khayyer *et al.* [75, 71].

So, this finding leads us to the following statement: in order to better justify the use of the projection-based incompressible SPH algorithm rather than enhanced WCSPH variants, two distinct research strategies can be developed, either to reduce the CPU cost per time step of the ISPH solution procedure and/or to overcome the CFL limitation. Note that whatever the solution adopted, it should not compromise the quality of results, neither the scope of applicability of the SPH methods. In the next section, a new formulation reducing the cost of the PPE inversion is presented, together with a validation process.

Chapter 7

Explicit ISPH

In order to better justify the use of an Incompressible SPH algorithm, one must be able to find a manner to reduce the CPU time or a way to overcome the CFL limitation, always in a way to maintain the good quality of results already achieved by the method. In this thesis a solution is proposed to the first restriction (reducing the CPU time). This is done by making explicit the solution of the PPE. That way, the CPU time spent on solving such an equation is reduced and at the same time the quality of results is maintained, as it will be shown later in this chapter.

We shall mention that during the validation process of the method developed in this thesis we became aware that it had already been used to simulate fluid-structure interaction with viscous flows by Rafiee *et al.* [87] and non-Newtonian flows by Hosseini *et al.* [88], but in both articles, focus was given to directly applying the method. More specifically, the two works do not justify such an explicit algorithm and lack of an analysis of the method as well as of a validation process to assess its applicability compared to the existing SPH techniques. Here the Explicit ISPH method is deduced, justified and validated on a series of benchmark test cases.

This chapter is organized in the following way: first the principle of the new explicit ISPH algorithm will be explained, then followed by a validation process using typical and relevant benchmarking test cases. Finally, we present some results extending the method towards naval applications.

7.1 Principle

As stated in the previous chapter, the semi-implicit ISPH algorithm has a major downside: the need to solve a linear system. Being all the other steps of the algorithms explicit, the most expensive part of the method is the solution of such a system. In this chapter, a solution is presented to reduce the CPU costs and achieve more scalability efficiency: solving the PPE in an explicit manner.

All the other steps of the algorithm are kept the same as in chapter 6, only the solution procedure of the PPE must be detailed.

From the last chapter, the PPE equation may be written in a simplified manner:

$$Ax = B. \tag{7.1}$$

Instead of using BiCGStab or GMRES, the Jacobi method can be used to solve equation (7.1). Within this method the A_{ij} matrix is separated into two new ones: one containing the diagonal terms (D) and another containing the terms outside the diagonal (R). This results in the following iterative procedure:

$$x^{k+1} = D^{-1} (B - Rx^k), \quad (7.2)$$

where k refers to the iteration step. Note that this iterative process starts with $x_i^0 = x_i^n$. By substituting equations (6.15), (6.16) and (6.17) into equation (7.2) and by replacing k with n , the following explicit equation for pressure is obtained:

$$p_i^{n+1} = \frac{\left(B_i + \sum_{j \neq i} A_{ij} p_j^n \right)}{\sum_{j \neq i} A_{ij}}. \quad (7.3)$$

Of course, performing only one iteration leads to errors. In Eulerian based methods with large time steps (implicit formulation) it is known that the Jacobi procedure is diffusive and does not converge. But in our formulation, we assume that equation (7.3) can be used to compute the pressure field and that these errors are acceptable if we use small time steps, which will be confirmed heuristically in the next sections. The Explicit Incompressible SPH presented in this thesis raises from this assumption.

We recall that operators A_{ij} and B_i are evaluated at positions \vec{r}_i^* . Equation (7.3) can be split, resulting in equation (7.4): the first term on the RHS is actually an approximation of p_i^n and the second one acts as an extra pressure (Δp_i) which depends on the non-incompressible intermediate state given by B_i . It corresponds to an additional pressure term to be added to the field at instant n in order to ensure incompressibility.

$$p_i^{n+1} = \frac{\sum_{j \neq i} A_{ij} p_j^n}{\sum_{j \neq i} A_{ij}} + \frac{B_i}{\sum_{j \neq i} A_{ij}}. \quad (7.4)$$

As in WCSPH [41, 40], the free-surface is automatically taken into account by the SPH operators in this new Explicit ISPH even though a clean proof of it as in [41, 40] for WCSPH should be brought in the future. This means that the dynamic free-surface condition does not need to be imposed before solving the PPE. This can be seen as another major advantage with respect to the semi-implicit ISPH algorithm, since no free-surface detection procedure is needed in this formulation. A study of this aspect is found in section 7.4.1.

We intend to verify in the present chapter that, with this new method, we are able to keep the quality of results given by the semi-implicit ISPH algorithm and decrease the CPU costs required by the latter. One may notice that this algorithm has some similarities to the ISPH method presented by Ellero *et al.* [89] where a SHAKE procedure is used to reach an incompressible state. This later one is based on the Gauss-Seidel method to solve linear systems which is an iterative method very similar to the Jacobi method. Moreover, the SHAKE algorithm in [89] is based on the fact at each iteration a small increment towards the solution is done ($x^{k+1} \simeq x^k + \delta x$) to compensate the non incompressibility

condition of the intermediate state given by B_i . However, contrary to what is done here, in [89] a full iterative process is considered.

7.2 Solver procedure

The new Explicit Incompressible SPH algorithm here is based on the traditional semi-implicit version and inherits its solver procedure. The main difference relies on the PPE solver. For the sake of clarity, we propose here to summarize the whole algorithm. The Explicit ISPH solver procedure for each time step reads:

Algorithm 1 Explicit ISPH solver.

Require: Factor α_{ISPH} is defined to select divergence free, density invariant or mixed formulation variant.

```

1: while  $t < t_{end}$  do
2:   Compute the particle density at instant  $n$  (section 6.2.2.1).
3:   Compute intermediate particle velocities and positions as in (6.1) and (6.2).
4:   Depending on  $\alpha_{ISPH}$ , compute intermediate density following (3.8).
5:   Compute particle pressure following (7.3).
6:   Update particle velocities using (6.3).
7:   Compute and apply XSPH correction.
8:   Update particle positions at instant  $n + 1$  following (6.5).
9:   Increment time as  $t = t + \Delta t$ .
10: end while
```

For the semi-implicit version of ISPH, step 5 is replaced by a system solver like BiCGStab or GMRES.

7.3 Internal flow validations

Traditionally, the first validating simulations of incompressible solvers are made in cases without the presence of a free-surface. This is typically done using the lid-driven cavity and the flow around a circular cylinder. In this section, we will focus on these two test cases using the divergence free variant. Comparisons are made between the new method and existing weakly-compressible SPH and/or FVM and also against experimental results.

We may add that these two flows are of slow dynamics and for that reason the Incompressible SPH is expected to behave better than the Weakly-Compressible SPH.

7.3.1 Lid-driven cavity flow

The lid-driven cavity flow is a classical benchmark used to validate numerical methods for viscous flows. This problem has been solved by many FVM solvers, see [2], and by particle-based methods. The geometry is a square shaped domain of size $L = 1m$ with three fixed walls and one moving top-wall with a constant imposed velocity $U = 1m/s$, as illustrated in figure 7.1.

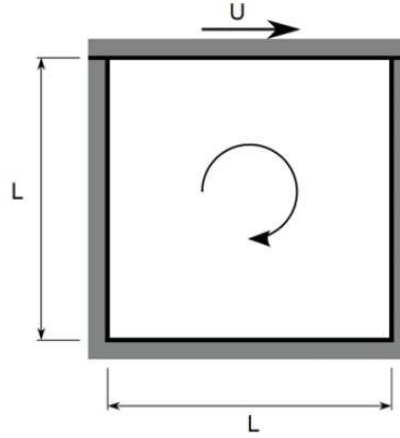


Figure 7.1: Lid-driven cavity flow sketch.

For all these lid-driven cavity flow simulations, an average of 30 neighbor particles per particle is used, and the simulations run until the stationary state is reached (time that may vary according to the Reynolds number). Note that the time step is the same as for the other SPH methods to which ISPH is compared.

- **Results with $Re = 1000$**

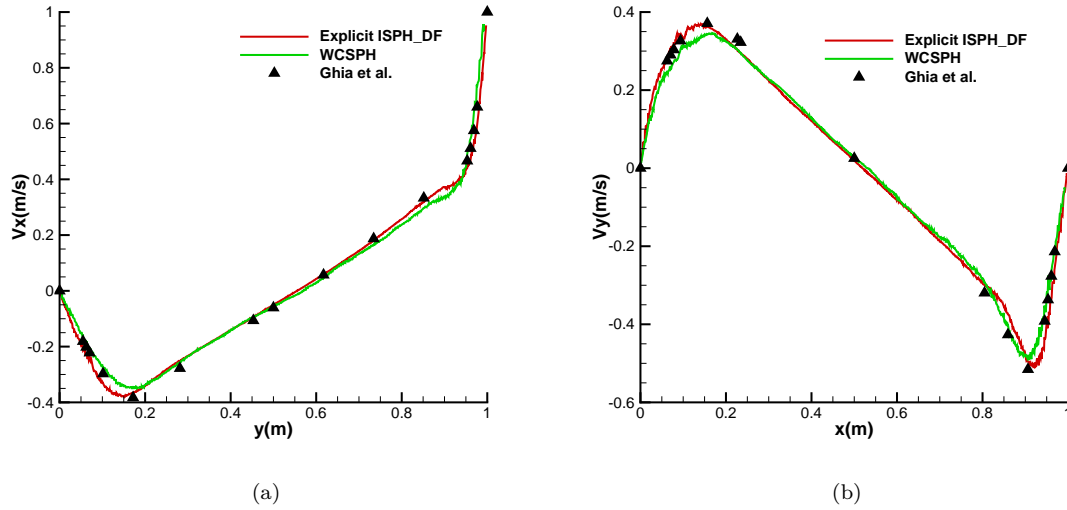


Figure 7.2: Comparison between FVM [2] (black triangles), Explicit ISPH_DF (red line) and WCSPH (green dashed line) with a particle space resolution of $\Delta x = 0.005$ m for: (a) velocity profile in the x-direction and (b) velocity profile in the y-direction.

For the Reynolds number $Re = 1000$, the Explicit ISPH_DF method is compared to WCSPH [74] (without artificial viscosity, being the physical viscosity enough to stabilize the scheme, integrated using a 4th order Runge-Kutta scheme) and to FVM results. The initial inter-particle distance used

is $\Delta x = 0.005\text{ m}$, which results in 40,000 particles (200×200 particles in the squared domain). The velocity profiles are compared to the results from Ghia *et al.* [2]. Figure 7.2 shows a very good agreement between the Explicit ISPH_DF, FVM and WCSPH for the velocity profile.

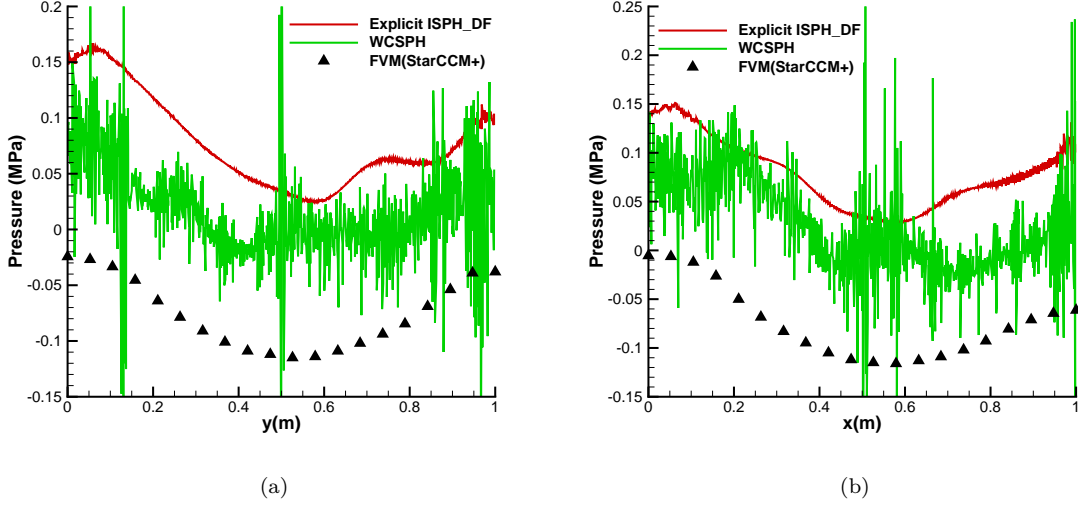


Figure 7.3: Comparison between FVM (STARCCM+, blue dashed line), Explicit ISPH_DF (red line) and WCSPH (green dashed line) with a particle space resolution of $\Delta x = 0.005\text{ m}$ for: (a) pressure profile on the x-direction and (b) pressure profile on the y-direction.

From the figure 7.3 we can see that the new Explicit ISPH method results in a less noisy pressure field than the one given by standard WCSPH. The pressure profile is also quite similar to the one predicted by the Eulerian FVM solver, but for a background pressure. Actually, although in ISPH we no longer have the limitation given by the equation of state we still use the same operators as those used in WCSPH and, as a consequence, the method suffers from tensile instability (TI) when confronted to negative pressures. This explains why there is no negative pressures in figure 7.3: whenever there is a region under negative pressures, particles tend to move in a way to create a void, and in Explicit ISPH no void is actually encountered during the simulations, errors resulted rather in an average positive pressure. However, when a correction is applied to the operators, renormalization [30] for example, one may encounter high levels of tensile instability and in order to avoid this instability, one may use a TI control technique. Here we decide not to apply any TI control neither to use the renormalization as their influence on the ISPH method has not been verified yet, and it may be the subject of a deep and complete separated study. One quick solution would be the use of background pressure in step 6 of algorithm 1, where the value has to be chosen to prevent any negative pressure.

- **Results with $Re = 3200$**

For the Reynolds number $Re = 3200$, the same comparisons as for $Re = 1000$ are presented. Once again, ISPH performs better than WCSPH, by getting closer to the results given by FVM [2]. Note that WCSPH starts lacking of precision at this Reynolds number with respect to the previous one.

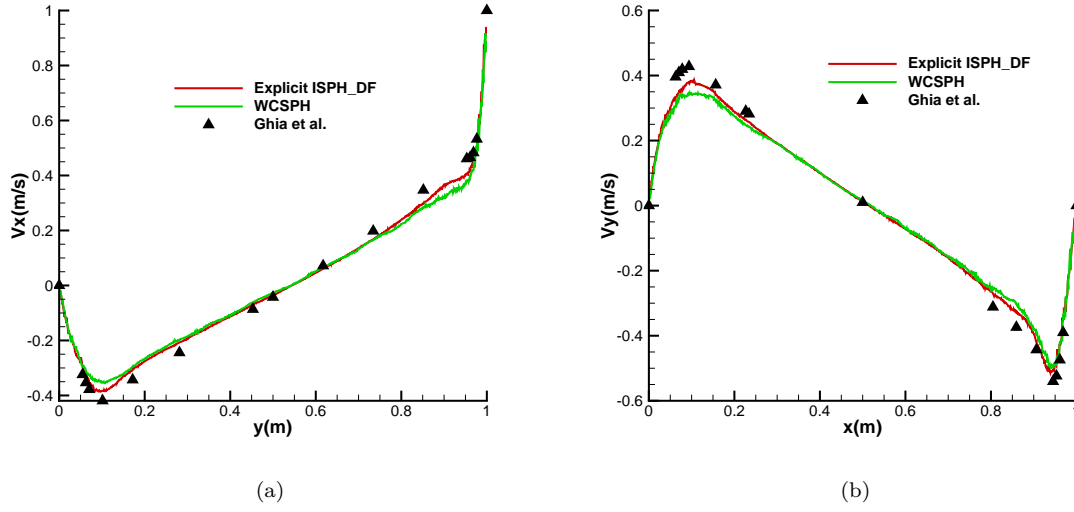


Figure 7.4: Comparison between FVM [2] (black triangles), Explicit ISPH_DF (red line) and WCSPH (green dashed line) with a particle space resolution of $\Delta x = 0.005$ m for: (a) velocity profile on the x-direction and (b) velocity profile on the y-direction.

The same trends are found for higher Reynolds numbers and for that reason we do not present here such results. The Explicit ISPH provides better results than WCSPH and similarly (in terms of quality of results) to FVM (even though the same refinements as the ones used in [2], 128×128 , were not considered here). Other authors [62, 58] validated the semi-implicit ISPH algorithm using this same benchmark for Reynolds number up to $Re = 1000$ and similar conclusions were also traced.

7.3.2 2D flow past a circular cylinder

The objective of this test is to validate the new method for non-flat geometries which are often present in real life problems. Additionally, differently from the last viscous test case (lid-driven cavity flow) here, for a Reynolds number of 100 we encounter an oscillating feature (vortex detaching) on the flow.

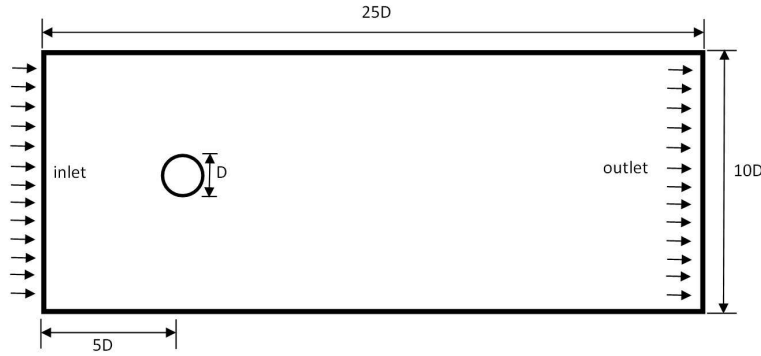


Figure 7.5: 2D flow past a cylinder simulation sketch.

The configuration of the simulations is shown on figure 7.5, where $D = 1$ m. Results obtained by ISPH are compared to several SPH methods (WCSPH and δ -SPH), to FVPM (Finite Volume Particle Method) [55] as well as to experimental data from Tritton [3]. A large domain is used to ensure that the entire wake is simulated (in the cases with no oscillating vortex, $Re < 100$).

In figure 7.6, a steady-state comparison between WCSPH [74] and Explicit ISPH_DF is shown for a Reynolds number $Re = 40$. A very good agreement is obtained between these two methods for the velocity field, although the Explicit ISPH_DF result is slightly more disturbed on the edge of the wake, due to a higher level of reorganization of the particles.

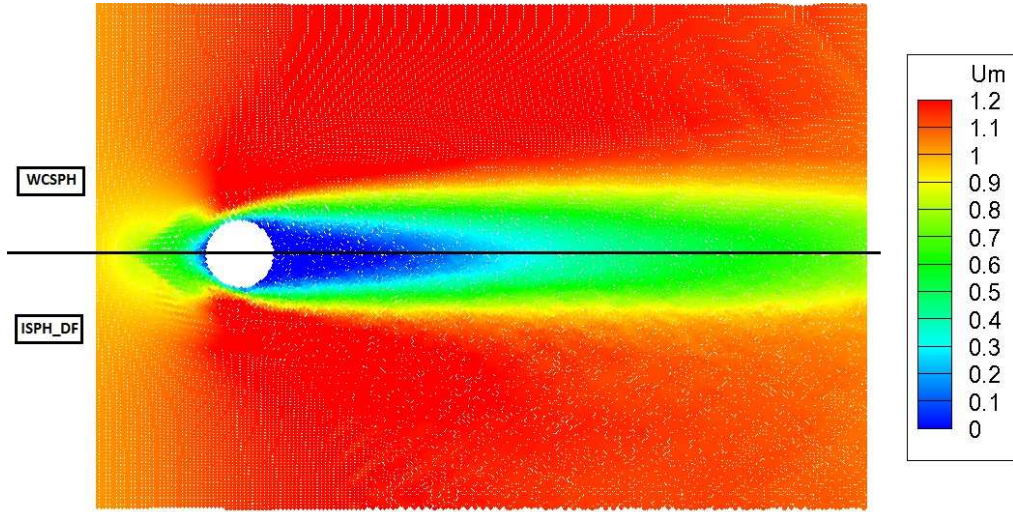


Figure 7.6: Comparison between WCSPH and Explicit ISPH_DF for $Re = 40$ with $\Delta x = 0.005$ m.

Figure 7.7 shows the pressure coefficient along the cylinder for the two methods at this same Reynolds number. The effects of particle reorganization for this Reynolds number are also visible. A pressure elevation occurs at the angle value where the boundary layer detaches ($\theta = 1.7$ for ISPH and $\theta = 1.9$ for WCSPH). The Explicit ISPH algorithm tends to present a higher particle reorganization than WCSPH. This can be considered as a good property of the algorithm, since disordered particle distributions are known to offer lower numerical damping [19]. Note, however that the pressure coefficient at the stagnation point is higher than it should be ($C_p = 1$). Both WCSPH and ISPH methods tend to a more elevated pressure level over the domain, possibly caused by the particle reorganization.

Other simulations have been performed from $Re = 10$ up to $Re = 100$, and a comparison between Explicit ISPH and the experimental results by Tritton [3] is plotted in figure 7.8. The global trend of the drag coefficient evolution obtained with Explicit ISPH is similar to the experimental one, but with slightly higher values which is probably a confinement effect due to the numerical set-up.

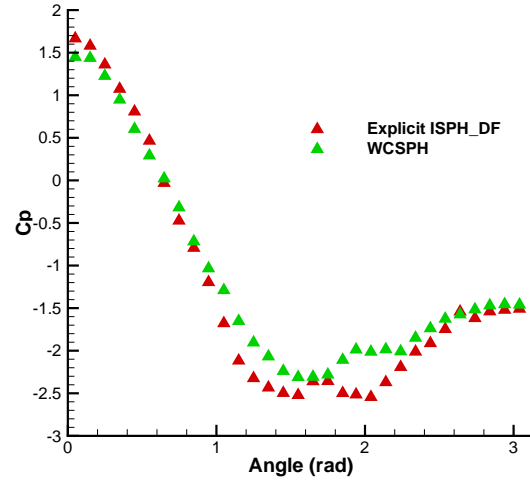


Figure 7.7: Pressure coefficient around the cylinder for $Re = 40$.

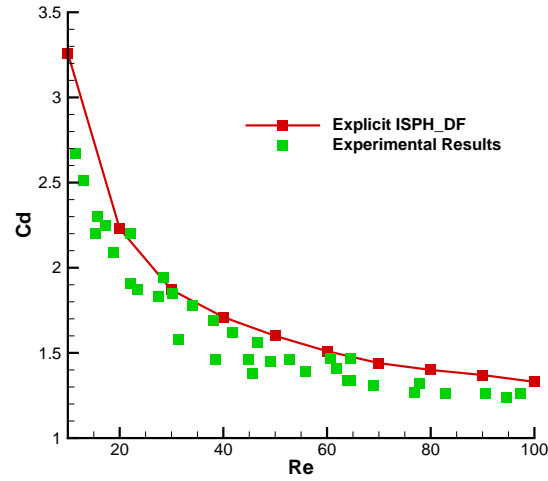


Figure 7.8: Drag coefficient for Reynolds number from 10 to 100. The red line represents the Explicit ISPH_DF solutions, the green points correspond to the experimental results [3].

In the framework of the NextMuSE European project [90], a comparison was performed between the methods developed by each one of the partners using this test case. Knowing that several Reynolds numbers were simulated by each partner and the number of methods considered, we selected only the results obtained at $Re = 100$ to perform a comparison. The frequency of vortex detachment can be analyzed by comparing the Strouhal number. The results given by ISPH, WCSPH, δ -SPH (detailed results can be found in [50]) and FVPM can be confronted to values given by the empirically-deduced formula from Roshko [91]:

$$St = 0.212 - \frac{4.5}{Re}. \quad (7.5)$$

Table 7.1 summarizes the configuration for each one of the methods compared here. All the simulations were performed using the same inter-particle distance of $\Delta x = 0.025$ m except for FVPM where $\Delta x = 0.050$ m was used.

Method	Standard WCSPH	δ -SPH	Explicit-ISPH	FVPM
Kernel	Wendland	Gaussian	Gaussian	Circular support
Neighbor Particles	30	50	50	33
Stabilizing Technique	None	$\alpha_\delta = 0.1$ and $\delta_\delta = 0.1$	$\varepsilon_{XSPH} = 0.01$	Riemann

Table 7.1: Configuration for each method used to simulate the flow around a circular cylinder

Table 7.2 presents the results obtained by each method and the differences with respect to Roshko's formula. All variants are within 10 % regarding this characteristic. Surprisingly, the less accurate variant here is FVPM. This is maybe related to the fact that FVPM did not use a fully Lagrangian description in the present case and also to the lower particle refinement used. The closest method here is δ -SPH. The Explicit ISPH variant is below 5 % which is an acceptable value, from an engineering point of view.

Method	Strouhal Number	%Roshko
Roshko	0.167	-
Standard WCSPH	0.171	+2.3%
Explicit ISPH_DF	0.175	+4.8%
δ -SPH	0.169	+1.2%
FVPM	0.181	+8.3%

Table 7.2: Strouhal number comparison.

At Reynolds number $Re = 100$ it is possible to observe some vortex detachment in the wake of the cylinder. Figure 7.9 compares the wake predicted by each of the method to the experimental results by Wu *et al.* [4]. These figures were obtained by plotting only the particles which were present at the beginning of the simulations, i.e. those not carried away by the incoming flow (colors represent the particle number and are not to be considered). These results show a good qualitative agreement to the experimental result. Few differences can be noted between the different variants, in agreement with the fact that the Lagrangian nature of the method is a predominant asset here, at least in terms of flow kinematics.

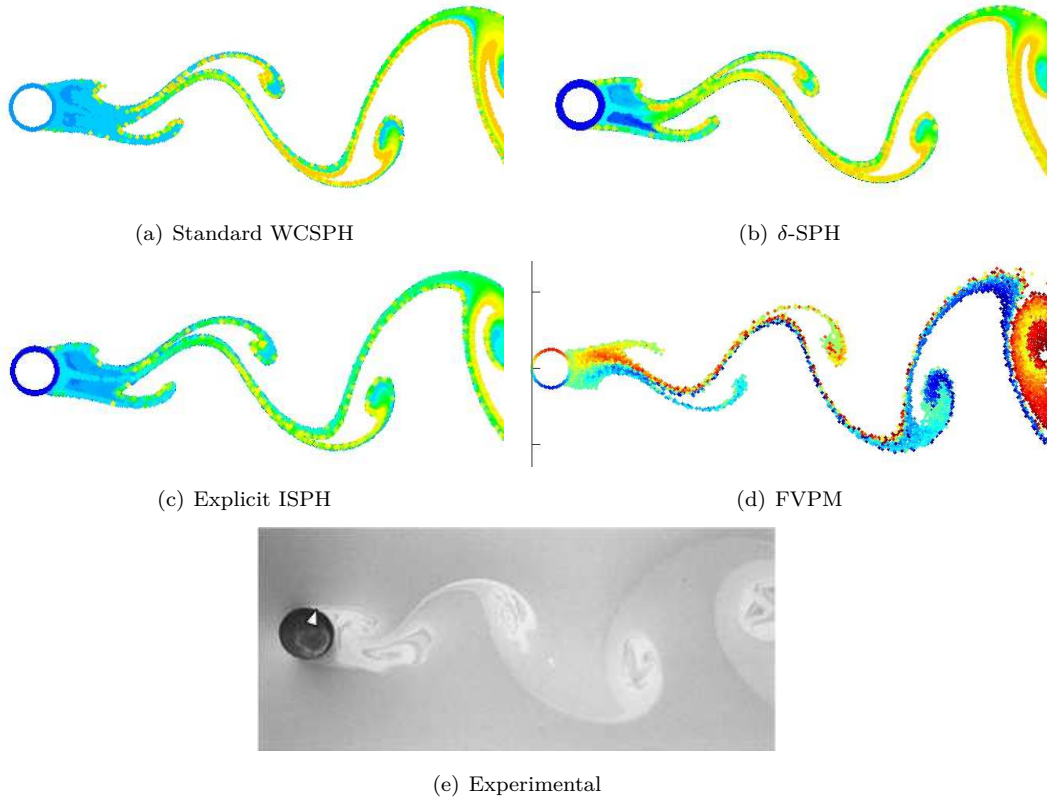


Figure 7.9: Plot of the cylinder wake for each method tested and for an experimental result [4]

7.4 Free-surface flow results

The Smoothed Particle Hydrodynamics method is mainly used to simulate free-surface and violent flows. Hence, it is important to validate every new SPH method for these kind of flows. Several free-surface test cases were simulated using this new fully Explicit Incompressible solver during the period of this thesis. In this section, we focus on the most important, simple and significant simulations: a viscous standing wave test case [40], a 2D dam-break [92], a 3D dam-break [93] and a jet impinging on a flat plate. Together, these four test cases are representative of the scope of application of the SPH method.

The first test case is dedicated to prove that the free-surface conditions are correctly taken into account by the present algorithm. The 2D dam-break case focuses on the validation of three main aspects: a comparison with other SPH methods, a study of the convergence of the method with respect to the kernel support (radius length or number of neighbor particles), and a stability study of the method. The third test case shows a 3D extension of the algorithm and also verifies its applicability to more complex geometry. Moreover, it allows a comparison to Riemann-SPH, δ -SPH and experimental results. Finally, the impinging jet test case is used to show the performance of the new ISPH algorithm in treating violent flow phenomena.

7.4.1 Standing Wave

Colagrossi *et al.* [40] showed that the free-surface conditions are implicitly verified (not exactly but in a consistent way) for viscous flows by using the Monaghan & Gingold [39] viscous operator in WCSPH. In [40] they analyzed the numerical energy decay on a standing wave test case using WCSPH for several particle distributions (other results on energy dissipation of gravity waves are available in [94]. In order to prove that the dynamic free-surface condition is also implicitly verified in the Explicit ISPH scheme, this test case is used here for a Reynolds number $Re = 500$ using the same viscous operator (equation (3.32)) and without imposing the dynamic free-surface boundary conditions.

This benchmark considers a periodic standing wave of wavelength $\lambda = 1$ m and amplitude $A = 0.05$ m, within a numerical domain defined by a width $L = \lambda$ with a water depth $H = L$. Free-slip conditions are imposed on the boundaries. At $t = 0$, the free-surface is at rest and an initial velocity field is applied to the particles according to the inviscid potential theory. For a fluid considered as viscous, an approximate analytical solution was derived by Antuono *et al.* in [95] which takes into account boundary layer effects.

Note that in this benchmark, the gravity g is considered as $g = 1 \text{ m s}^{-2}$. To heuristically verify that the free-surface conditions are implicitly taken into account by the operators in the Explicit ISPH algorithm, this algorithm should predict the same energy decay as the analytical solution from Antuono *et al.* in [95].

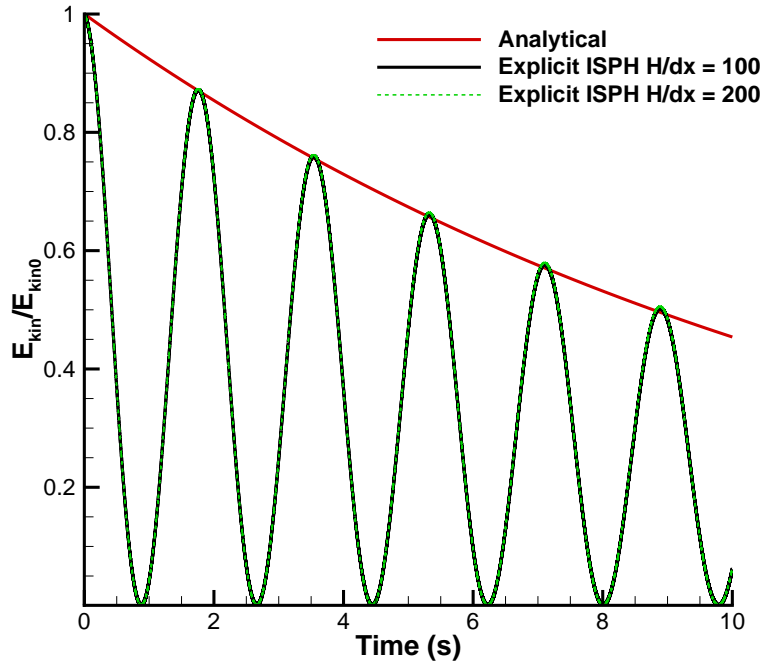


Figure 7.10: Comparison of the analytical and explicit ISPH energy decay obtained for the standing wave test case.

Two particle distributions resolutions are used for the ISPH simulations: $H/\Delta x = 100$ and $H/\Delta x = 200$. Here we only consider the divergence-free variant. Similar results would be obtained for the density invariant version, since only the source term of the PPE would change. Figure 7.10 shows that the procedure proposed to solve the incompressible Navier-Stokes equations correctly predicts the viscous damping rate. Therefore, there is no need to use a specific free surface detection algorithm. Surely, this conclusion may be considered as a little bit inappropriate since the classical ISPH approach demands the free-surface dynamic boundary conditions to be imposed, however, the analysis presented here may be better analyzed in the future following the works presented in [94].

The same simulation was performed by imposing the dynamic free-surface conditions to the particles belonging to the free-surface (to detect these particles, the technique by Marrone *et al.* [1] was used). Figure 7.11 shows the pressure distribution and the velocity field (on the y-direction) at $t = 7.0s$. Whenever the dynamic free-surface condition is imposed, spurious velocities appear close to the free-surface, leading to the creation of vortices which, in turn, lead to a disturbed flow. This happens due to the fact that the algorithm naturally takes into account the presence of the free-surface and by imposing the dynamic free-surface condition we doubly constraint our flow, leading to these spurious velocities. Nevertheless, the pressure field remains smooth.

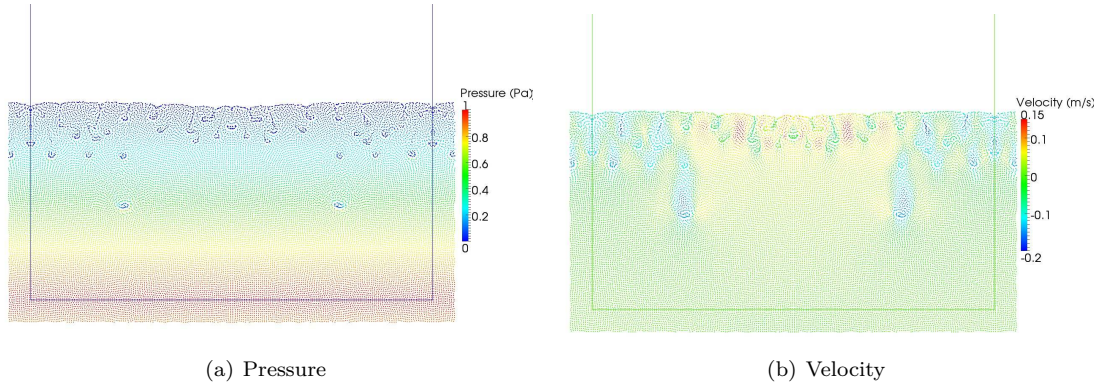


Figure 7.11: Pressure distribution and velocity field at $t = 7.0s$ with dynamic free-surface condition imposed

7.4.2 2D dam-break flow against a wall

In this first dam-break test case, a column of water initially at rest collapses and impacts a vertical wall. The sketch of the problem is depicted in figure 7.12 where $p1$ and $p2$ indicate two pressure probes located at a distance of 0.16 m and 0.584 m, respectively, from the bottom, according to the experiment performed by [92]. The total physical time simulated is of 4.5 seconds.

As stated before, this test case is emblematic of the flows simulated using the Smoothed Particle Hydrodynamics method, that is, violent free-surface flows. It has been largely used to validated all the existing variants of SPH: WCSPH, Riemann-SPH, δ -SPH and semi-implicit ISPH. The large experience and wide literature for this test case serves as a good basis to better examine the results and hence perform a better analysis of the method being validated.

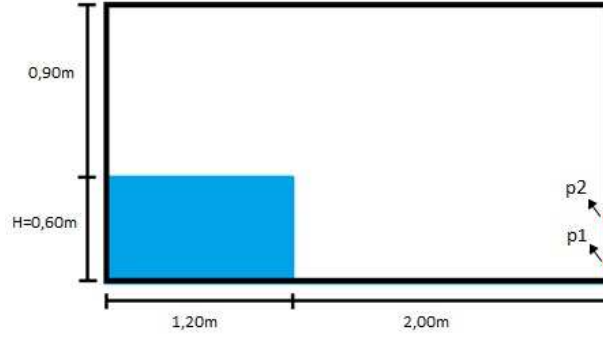
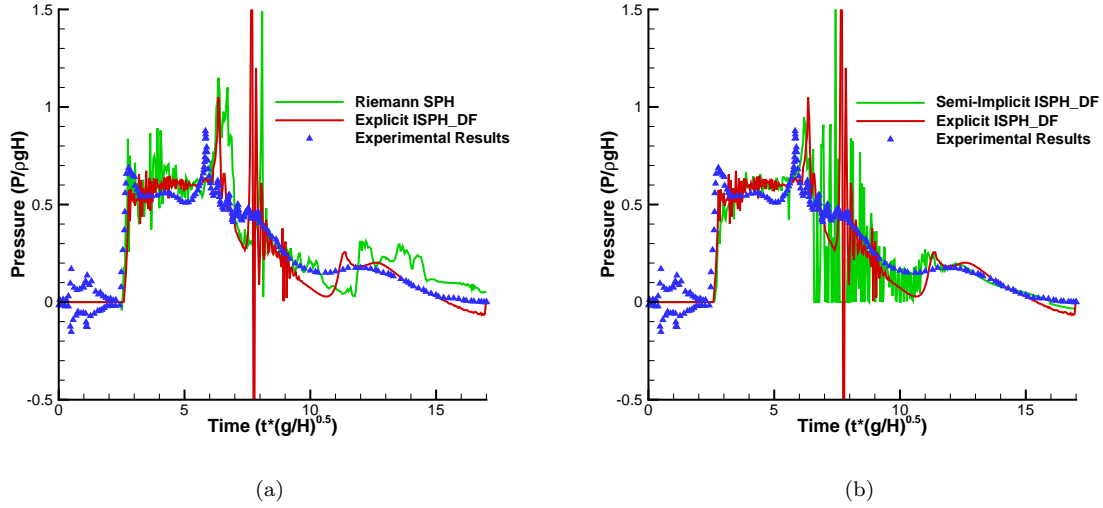


Figure 7.12: 2D dam-break simulation sketch.

Figure 7.13: Comparison of the pressure signal at $p1$ for the coarse particle distribution: (a) Riemann SPH, Explicit ISPH and Experimental Results; (b) Semi-Implicit ISPH, Explicit ISPH and Experimental Results

Firstly the validation results are presented, that is, the pressure evolution given by the new explicit ISPH algorithm is compared to experimental results and available results for other SPH variants, such as Riemann-SPH and semi-implicit ISPH. This is followed by a convergence and energy conservation analysis and finally, we perform a simple, but conclusive, CFL stability analysis.

7.4.2.1 Validation results

Two sets of results obtained with Explicit ISPH are shown in this section: a coarse resolution with $\Delta x = 0.01$ m (corresponding to $H/\Delta x = 60$) and a fine one using $\Delta x = 0.005$ m (corresponding to $H/\Delta x = 120$), resulting in 7,200 and 28,800 particles respectively. The results given by the new ISPH method are compared to experimental results from Buchner [92], to a semi-implicit ISPH solver (cf. chapter 6) and to a Riemann-based SPH algorithm [25] integrated in time using a 4th order Runge

Kutta scheme. In order to better compare the accuracy of the different solvers, all these simulations are performed with the same time step, with $c_0 = 30 \text{ m s}^{-1}$ in equation (3.33), and with $CFL = 0.75$ (which is a high value, if compared to standard values used within the SPH literature). As mentioned previously, the Explicit ISPH simulations are performed without imposing the free-surface condition, contrary to the semi-implicit variant.

In figure 7.13(a), the pressure profile at $p1$ for both Riemann-SPH and Explicit ISPH are compared to experimental results. The Explicit ISPH presents a similar behavior as the already validated Riemann-SPH solver, but with a less noisy signal. The Explicit ISPH approach seems globally closer to the experimental signal, while Riemann-SPH better captures the first peak at $T \approx 2.9$. The Riemann-SPH variant is better suited for these violent flow due to the fact that it better captures and solves the small variations in the flow which are high during impact phenomena. Nevertheless, the Explicit ISPH solver better captures the pressure peak at $T \approx 6.4$. Figure 7.13(b) shows that the explicit and semi-implicit ISPH formulations lead to similar results, although the latter gives a more noisy pressure profile.

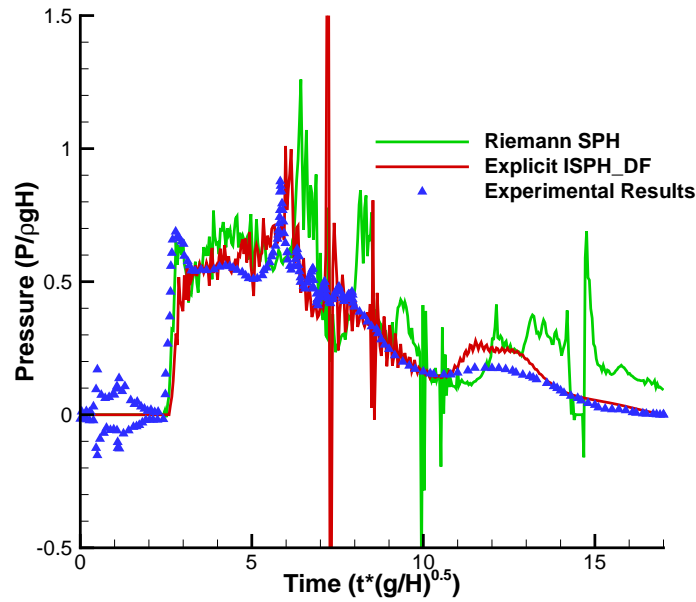


Figure 7.14: Comparison of the pressure signal at $p1$ for the fine particle distribution.

Figure 7.14 shows a comparison between Explicit ISPH, Riemann-SPH and experiments for the pressure signal at $p1$ using the finer particle distribution ($\Delta x = 0.005m$). Apart from the first peak, the Explicit ISPH scheme seems closer to the experiments than the Riemann-SPH solution. The bubble collapse peak at $T \approx 8$ is well detected and the pressure decay is in very good agreement with the experimental signal.

Figure 7.15 shows the pressure field at different instants for the finer resolution. As a qualitative observation, note that the pressure field obtained with the Explicit ISPH formulation is smoother than

what could be expected using other SPH formulations with similar flow evolution. Note that the cost of the semi-implicit variant is three times as big as for the Explicit variant for the coarser mesh, and this factor increases when using a finer particle resolution. A more detailed CPU costs comparison is performed in section 7.4.3 for the 3D dam-break test case.

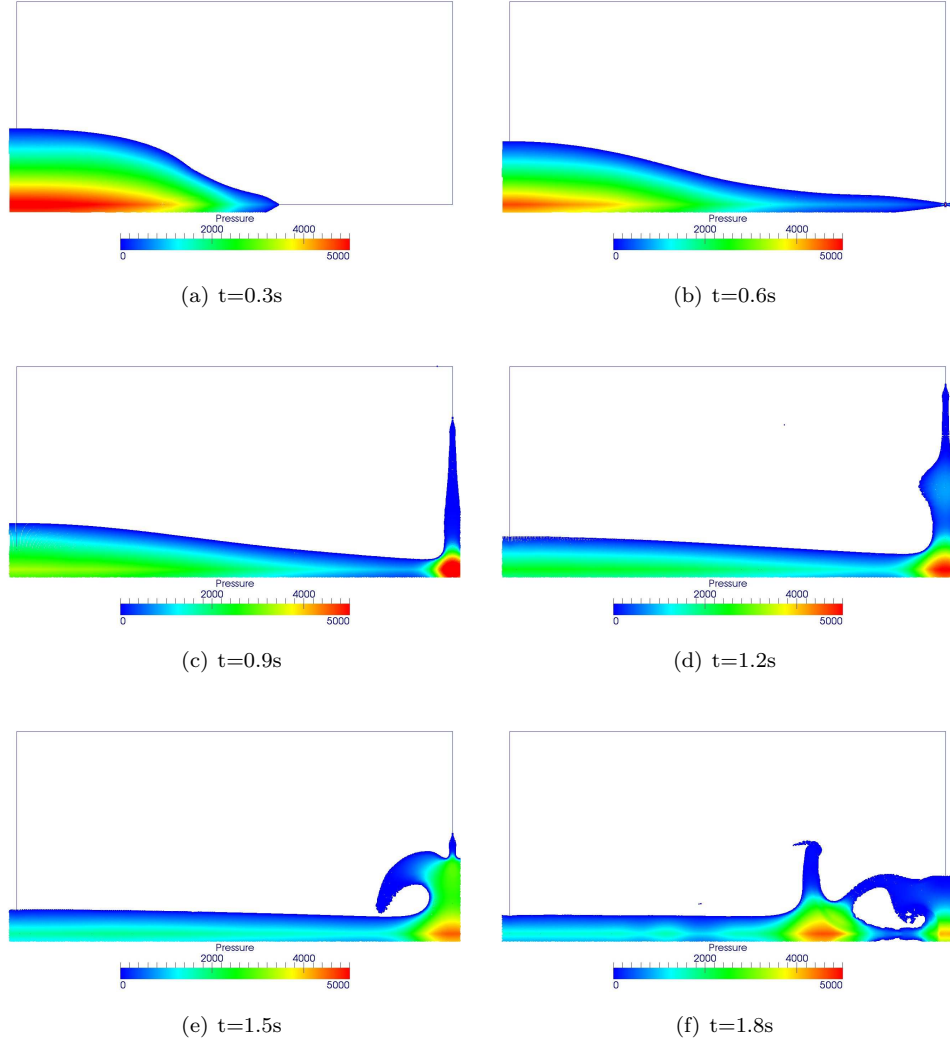


Figure 7.15: Pressure field obtained with the Explicit ISPH formulation.

Another important issue concerning the validation of such an explicit ISPH algorithm concerns the verification of the incompressibility of the final velocity field. Due to errors related to the assumptions made to justify the use of such an explicit expression, at the end of a time step a non-incompressible state may be reached. It is known, as reported by Colin *et al.* [96], that the semi-implicit ISPH algorithm, due to error accumulation and to the projection method itself, no completely null divergence velocity field is reached (the levels are usually higher than the tolerance imposed for solving the PPE) even if it can be minimized by imposing some special techniques (cf. [96]). We may add to that the fact that the evaluation of the pressure gradient presents some errors and hence the final velocity

which is computed based on this latter variable will inherit these errors. Moreover, another source of imprecision comes from the computation of the velocity divergence field itself. We use this present test case to analyze the incompressibility of the final velocity field comparing the semi-implicit approach to the explicit one. Figure 7.16 shows the maximum value for the velocity divergence in the flow throughout time for both the semi-implicit and explicit ISPH approaches for $\frac{H}{\Delta x} = 60$. At the first time instants semi-implicit ISPH presents lower levels of velocity divergence than the explicit variant even after the flow impacts the wall, but this trend is inverted after the bubble collapses ($T \approx 6.5$ where a peak is observed in the divergence behavior for the semi-implicit ISPH) when explicit ISPH starts to present better results than the semi-implicit variant which was much unexpected. Note that after the waterfront impact against the wall the level divergence of the velocity field increases rapidly and this is precisely the time instant in which the particle tend to more ‘random’ distribution inducing to error on the SPH operators evaluation. These results are similar to the ones presented in [96] and show that, globally, explicit ISPH behaves better than the semi-implicit. This difference may be explained by the fact that in this latter one a system is solved to a given small residual but the system itself is ill-conditioned (due to the imprecision of the SPH operators and the Lagrangian feature of the method) and it results sometimes in poor solutions. In other words, the semi-implicit ISPH is much more sensible than the explicit variant where the SPH imprecisions are filtered out (one iteration of the Jacobi solver is often used as a filter in mesh-based methods).

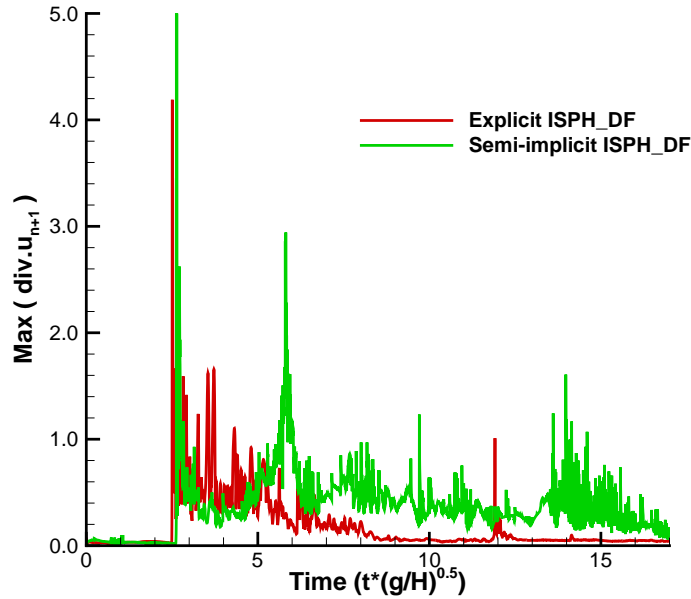


Figure 7.16: Comparison of the maximum value found in the flow for the velocity divergence between semi-implicit and explicit ISPH

7.4.2.2 Convergence and energy conservation analysis

The convergence with respect to physical results and to energy conservation is an important aspect of the method and needs to be studied. This study is performed here on the 2D dam-break test case with five distinct inter-particle distances Δx and interpolation kernel radii R . Table 7.3 summarizes the parameters used for this convergence study.

$H/\Delta x$	$R/\Delta x$
40	2.00
60	3.00
80	4.00
100	5.00
120	6.00

Table 7.3: Convergence study parameters.

Indeed, two different convergence criteria have to be respected in the SPH method [27]. The first one states that Δx has to tend to zero (as for any other numerical method) which is responsible for the total number of particles involved. The second one states that $\frac{\Delta x}{R}$ also has to tend to zero which is linked to the use of a kernel-based interpolation for which the number of neighbor particles per particle rules the convergence of the SPH operators themselves. In that way, decreasing both Δx and $\frac{\Delta x}{R}$ leads to convergence, and a faster convergence in practice than when reducing Δx only, which leads to saturation eventually. A similar study is available in [21].

Figures 7.17 and 7.18 show the pressure signal convergence at sensor $p2$ as well as the mechanical energy evolution for the divergence free (DF) and the density invariant variants (DI), respectively. The mechanical energy loss is computed using expression (7.6)

$$E = \frac{E_c + E_p - E_p^1}{E_p^1 - E_p^2}, \quad (7.6)$$

where E_c is the kinetic energy, E_p the potential energy and the superscripts 1 and 2 stand for the initial and the last time instants. For all these simulations, no free-surface condition is imposed and the XSPH correction constant is set to $\epsilon_{XSPH} = 0.05$. This value is chosen so as to minimize the influence of this correction on the energy conservation results.

Figures 7.17 and 7.18 show that the ISPH_DI variant is slightly more conservative than the ISPH_DF. Nevertheless, both give the same rate of convergence (of the order of the WCSPH), while ISPH_DF gets closer to the experimental results. These results are reassuring as they present the same trends as the ones obtained for the WCSPH, Riemann-SPH and δ -SPH which are available in [21].

The same simulations were performed using the same parameters but by imposing the free-surface dynamic condition, that is $p = 0$ (similarly to what was done for the standing wave problem, section 7.4.1) to particles belonging to the free-surface, for both ISPH_DF and ISPH_DI. Results are shown in figures 7.19 and 7.20. Whenever the free-surface condition is applied, oscillations appear in the energy conservation behaviour and the results given for the pressure probe are much poorer than for

the simulations where this conditions is no imposed. This confirms what has been verified on section 7.4.1.

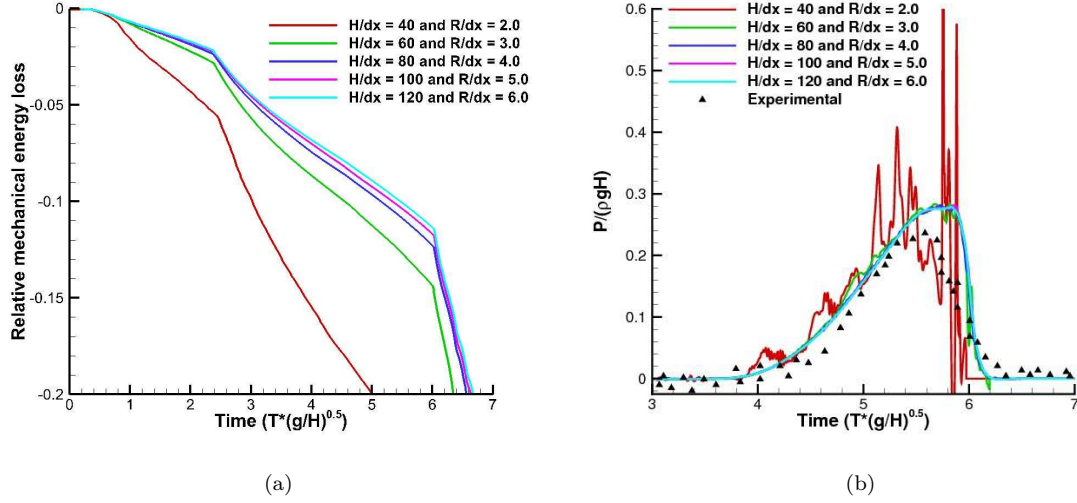


Figure 7.17: Mechanical energy and pressure signal evolution for the five different simulations of the ISPH_DF variant without free-surface detection procedure.

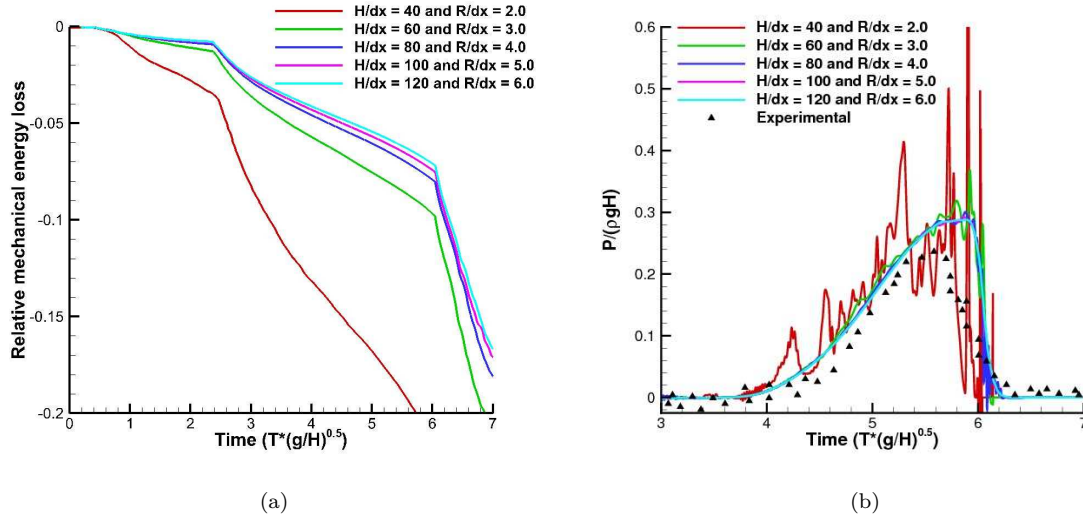


Figure 7.18: Mechanical energy and pressure signal evolution for the five different simulations of the ISPH_DI variant without free-surface detection procedure.

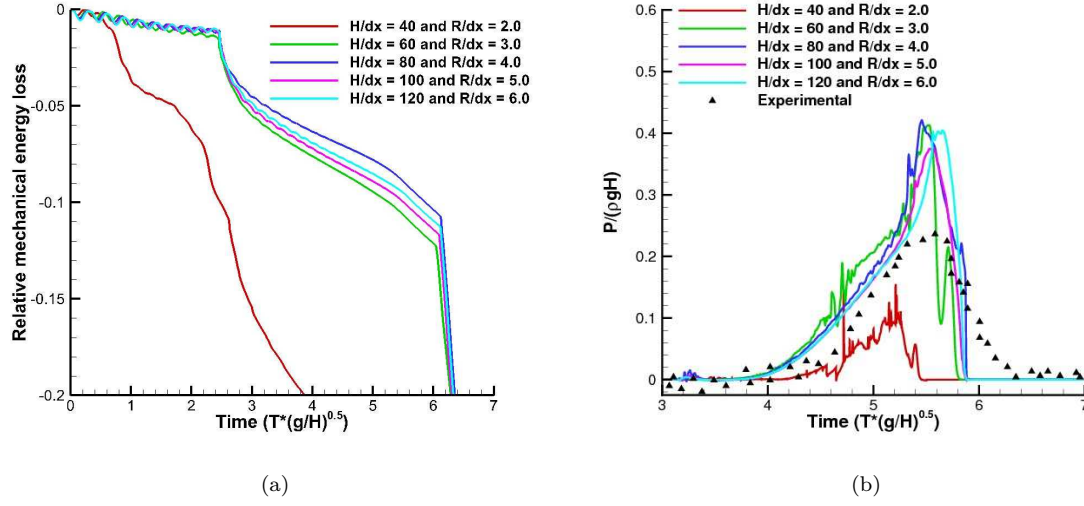


Figure 7.19: Mechanical energy and pressure signal evolution for the five different simulations of the ISPH_DF variant with free-surface detection procedure.

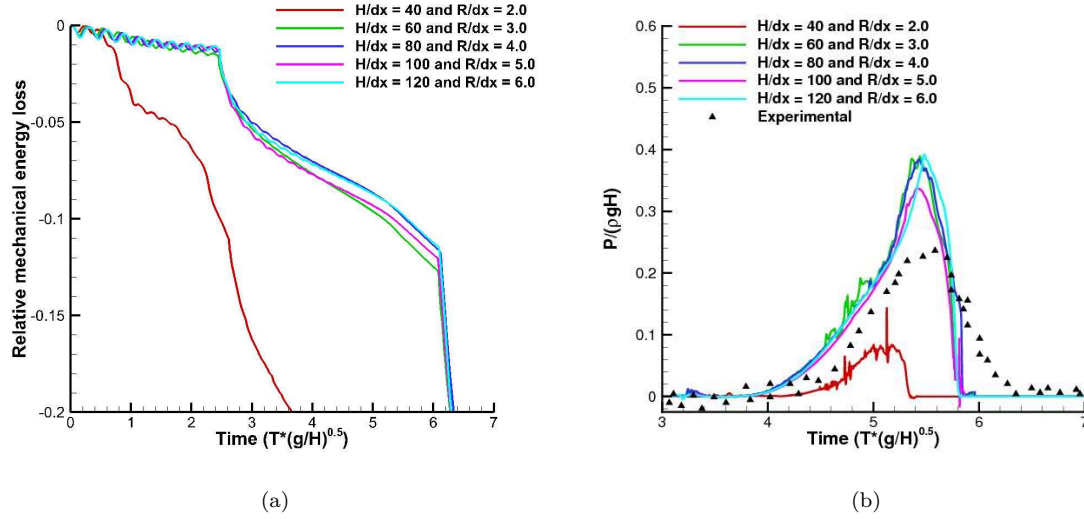


Figure 7.20: Mechanical energy and pressure signal evolution for the five different simulations of the ISPH_DI variant with free-surface detection procedure.

7.4.2.3 CFL study within the Explicit ISPH scheme

As performed previously for the semi-implicit ISPH formulation, we check here the possibility of increasing the time step over the limits usually stated in the literature. The resolution adopted in this study is $\Delta x = 0.005$ m ($H/\Delta x = 120$), with an average of 50 neighbor particles per particle. The same CFL numbers as those used in section 6.5 are employed.

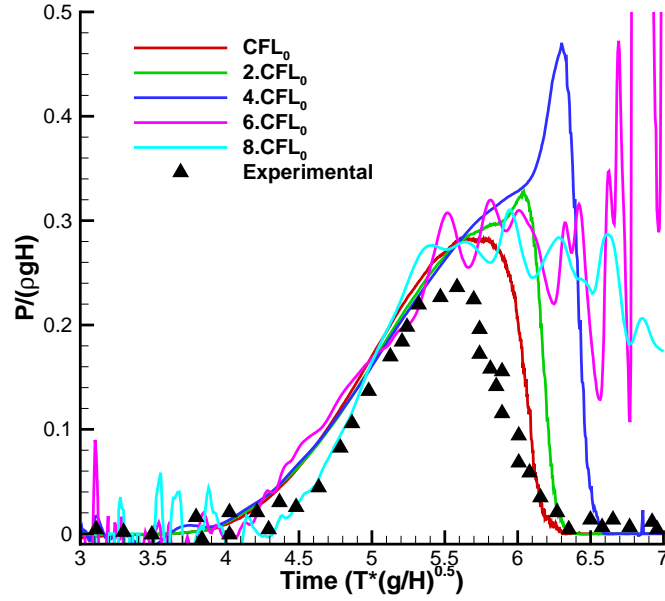


Figure 7.21: Pressure signal comparison between Explicit ISPH_DF and experimental results for five different CFL conditions.

Figure 7.21 shows the pressure signal obtained with the Explicit ISPH_DF formulation at pressure probe $p2$ for various Courant number (CFL) values. For $CFL \leq 4CFL_0$, a smooth pressure signal is obtained but with a lower accuracy as the Courant number increases. For higher values of CFL , some perturbations start to appear due to instability effects, as observed in the semi-implicit ISPH method. While this latter method starts lacking of accuracy on predicting the impact pressure (lower pressures), the opposite behavior is observed for the explicit ISPH solver: pressure are overestimated when the CFL is increased.

Table 7.4 shows the CPU costs of the Explicit ISPH compared to WCSPH (Riemann-SPH integrated in time using a 4th order Runge Kutta scheme). Explicit ISPH is already more efficient than WCSPH at the lowest Courant number tested.

Method	CFL	CPU time (seconds)	%WCSPH
WCSPH	CFL_0	3359.2	-
Explicit ISPH	CFL_0	2025.6	-39.7%
Explicit ISPH	$2CFL_0$	1141.2	-66.0%
Explicit ISPH	$4CFL_0$	861.3	-74.4%
Explicit ISPH	$6CFL_0$	676.6	-79.9%
Explicit ISPH	$8CFL_0$	454.5	-86.5%

Table 7.4: CPU time comparison between Explicit ISPH and WCSPH.

As a conclusion of these results, it can be said that the Explicit and semi-implicit ISPH methods have similar CFL limitations. However, the CFL must be kept at low values, in order to preserve accuracy and to ensure that the assumptions made to establish this new Explicit formulation are still justified. It is verified again that the Explicit-ISPH is about three times as fast as semi-implicit ISPH, and even faster than the traditional WCSPH method.

7.4.3 3D dam-break flow against a rectangular step

This section treats of the 3D dam-break occurring within a prismatic tank fitted in with a rectangular step. Figure 7.22 illustrates the geometrical setup of this benchmark. Due to the rectangular step presence in the middle of the domain, 3D effects appear during the flow, and if these effects are not well simulated, the pressure profile cannot be well predicted, mainly after the flow impact against the obstacle.

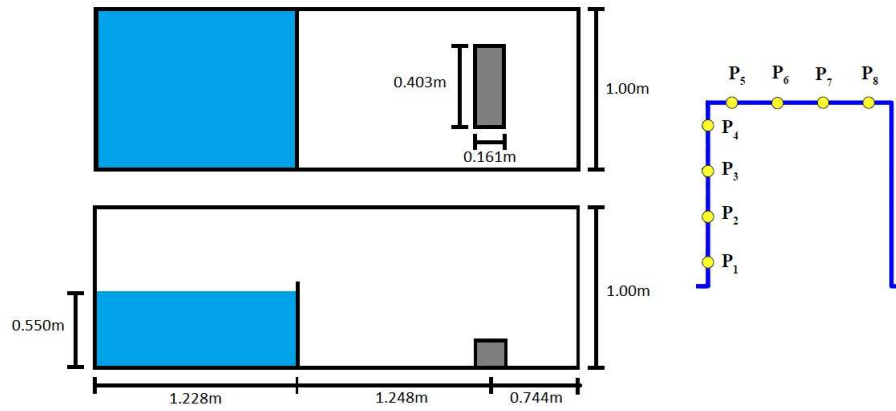


Figure 7.22: 3D dam-break simulation sketch with the pressure sensors on the right.

The results given by the explicit Incompressible SPH are compared here to the experiments performed by Kleefsman *et al.* [93], as well as to two different SPH formulations available in the SPH-Flow code developed by Ecole Centrale de Nantes and Hydrocean: Riemann SPH and δ -SPH formulations. This last method has already been validated on this test case by Marrone *et al.* [97]. These last two WCSPH formulations use a 4th order Runge Kutta time integration scheme, and all the simulations are performed using an average of 80 neighbor particles together with an inter-particle distance $\Delta x = 0.01$ m, thus involving 676,500 particles.

In the experiments of Kleefsman *et al.*, eight pressure sensors were located along the step on a constant y section ($y = 0.471$ m - see figure 7.22). Figure 7.23 shows a comparison of the pressure given by the Explicit ISPH with the experimental results and the WCSPH solutions. Figures 7.23(a) to 7.23(c) show the smoothed pressure signal (for the sake of clarity) and 7.23(d) shows a focus at the second pressure peak for the non-smoothed pressure signal. These figures show that the Explicit ISPH formulation behaves similarly to δ -SPH regarding the fluid dynamics. Both predict the pressure peak occurrence at the same instant for pressures probes 1 and 3. An interesting behavior can be observed at pressure probe 6, which is located at the top of the rectangular step: all of the three methods predict

very well the instant at which the wave front breaks over the step. Note, finally, that the pressure signal obtained with the Explicit IPSH formulation is smoother than for the WCSPH, as seen in figure 7.23(d).

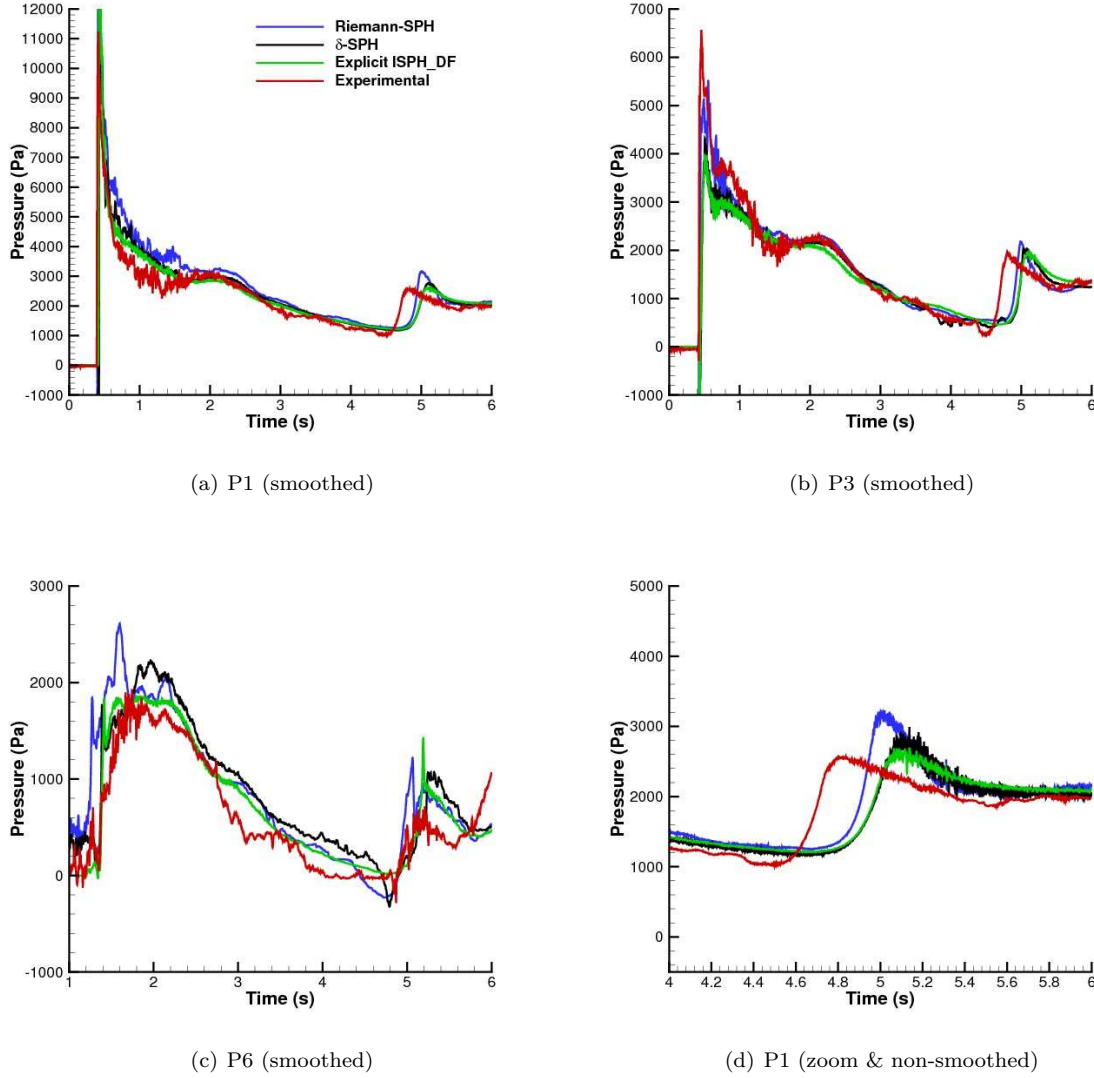


Figure 7.23: Pressure signal comparison for the three pressure sensors.

As all the variants compared in this study were coded in the same architecture, the CPU costs can be compared. Table 7.5 compares the performances of each SPH method tested here with the same time step for a total physical time of 6 seconds. Explicit ISPH outperforms the other methods in terms of CPU cost, with a similar quality in the results. This lower cost can be explained by the smaller number of loops over all the particles necessary in the scheme presented in this article, in comparison to the usual enhanced WCSPH schemes. In Explicit-ISPH there are only 3 major particle interactions loops (prediction, correction, and pressure computation), 3 particles updating and 3 ghost particles

updating per time step, contrary to WCSPH using a 4th order Runge Kutta scheme where 4 of each are present. Moreover, if we consider the Riemann-SPH algorithm we may add to that the solution of the Riemann problem and if δ -SPH is considered we may add 1 particle interactions loop which is necessary in order to compute one extra variable needed by the scheme. It is possible to use less accurate time integration schemes for the latter methods (modified Euler, for example) but that would result on poorer results and it would require for lower values for CFL parameter than $CFL = 0.375$ used here.

Method	T(s)	% Riemann SPH	% δ -SPH
Riemann SPH	273712,21	-	-
δ -SPH	147518,50	-46,10%	-
Explicit ISPH	97020,16	-64,55%	-34,26%

Table 7.5: Kleefsman's dam-break CPU time comparison

7.4.4 Impinging jet on a flat plate

This test case is used to check the method capability to simulate impact flows and to precisely predict the impact pressure in these kind of flows. Figure 7.24 shows the sketch of the simulation where $\theta = 30^\circ$ and $H = 0.4$ m. Similarly to the previous test case, comparisons are done between Explicit ISPH, δ -SPH, Riemann-SPH and a reference analytical solution for the pressure at the impact point and for the shape of the free-surface. For all the methods tested here the viscosity is not considered. The inlet velocity is chosen as equal to 1 m s^{-1} in the normal direction to the inflow section and a fluid density of 1000 kg m^{-3} is considered. Table 7.6 summarizes the configuration used for each of the SPH variants tested.

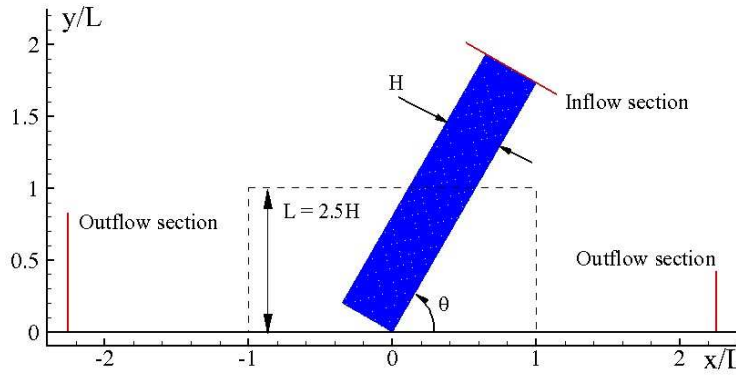


Figure 7.24: Sketch of the impinging jet flow test case

The first comparisons are done for the pressure field at a certain time instant (where a steady state is already reached). Figure 7.26 shows the results obtained for each SPH variant considered here, where the red lines represent the steady state analytical solution for the shape of the free-surface. The level of noise is practically the same for all the methods, being the less noisier the Riemann-SPH results.

Method	Riemann-SPH	δ -SPH	Explicit-ISPH
Kernel	Cubic Spline Kernel	Gaussian Kernel	Gaussian Kernel
Neighbor Particles	30	50	50
Stabilizing Technique	Riemann	$\alpha_\delta = 0.01$ and $\delta_\delta = 0.1$	$\varepsilon_{XSPH} = 0.01$

Table 7.6: Configuration for each SPH variant used to simulate the impinging jet flow

Figure 7.25 shows the pressure at the impact point ($x/L = 0.2$). The time shift in the pressure increase for the δ -SPH result is not to be considered (due to differences on the construction of the problem). All the variants are quite close to the target result. Contrary to what is observed in figure 7.26(a), when observing the temporal behavior of the pressure signal, the Riemann solution is noisier than the incompressible solution, and also predicts a smaller pressure coefficient. This may be explained by the different boundary treatment considered for the Riemann-SPH, i.e., normal flux method [19] was used here instead of the ghost particle technique for the others. As for the Explicit ISPH, it presents the same level of noise as δ -SPH. Note that this test case was also used within the NextMuSE project to perform comparisons between the different SPH variants present in the project.

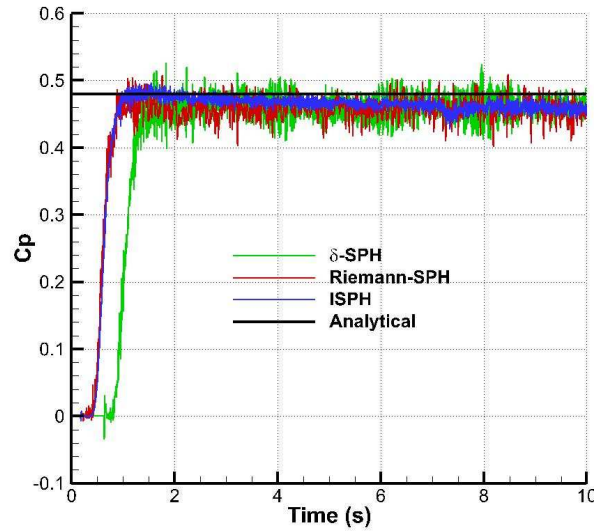


Figure 7.25: Pressure coefficient at $x/L=0.2$: the red line shows the Riemann-SPH result, the green one the SPH result, the blue dashed line the Explicit-ISPH result and finally the pink line is the analytical reference value

The Explicit ISPH scheme presented in this thesis is thus validated to a wide range of flow phenomena, including violent flows, for which good results were achieved, comparable to the best other SPH method variants, at a lower CPU cost.

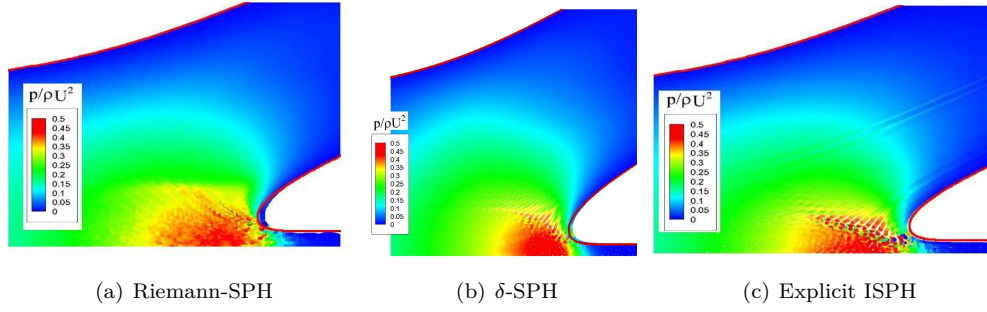


Figure 7.26: Pressure field for each SPH variant tested

7.5 Towards Naval Applications

The last sections were dedicated to the benchmarking/validation of Explicit ISPH. The LHEEA Lab. and partners (Hydrocean, INSEAN, etc) focus on naval/offshore and ocean flow problems. Therefore, in this section we will consider some problems that get closer to real world problems.

Two test cases are considered. The first treats a drowning body in a viscous flow. We intend then to validate the new Explicit Incompressible SPH algorithm in a flow where the computation of the fluid forces acting on the body are crucial, and may serve as reference for future fluid-structure interaction problems. The second considers the simulation of sloshing phenomena, a recurring problem in Naval/Offshore Engineering: a laboratory experiment will be reproduced.

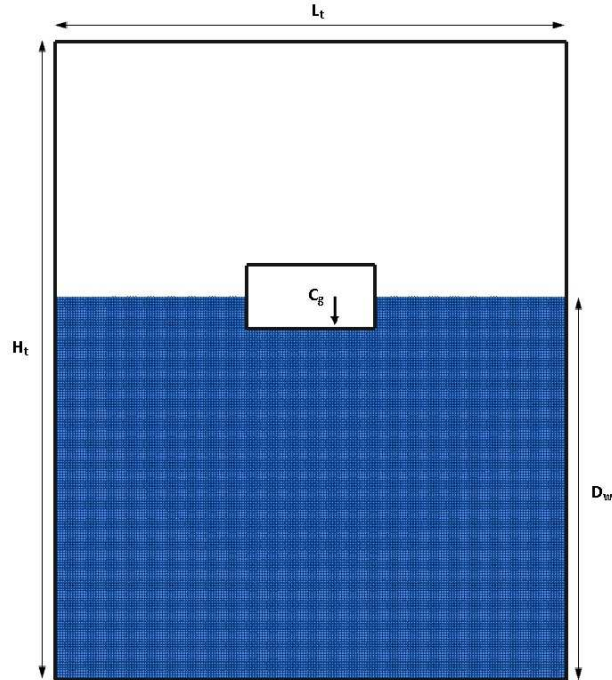


Figure 7.27: Sketch of the simulation with an asymmetric mass distribution body

Both flow phenomena are (and have been) often simulated using our in-house code, SPH-Flow [98], and several studies have been performed using the different SPH variants coded in it. We intend to verify if, by using this new explicit ISPH, we are still able to reproduce these flows and if we can improve the results achieved so far. We extend our comparisons, for the first test case, not only to SPH methods but also to the Finite Volume Particle Method (FVPM).

7.5.1 Flow interaction with a rigid body dynamics

This test case was studied within the NextMuSE project which financed part of this thesis. We considered a body with an asymmetric mass distribution immersed in a viscous flow. Figure 7.27 shows the sketch of the simulation where the black arrow represents the gravity action on the body's center of mass (C_m) located at $(0.2, 0.0)m$. At $t = 0s$ the body is in an unstable position and being twice as heavy as the fluid into which it is immersed, it will sink and move in a way to find its stable position (towards a vertical position). The special feature of this test case is that it considers the interaction of a viscous flow with a rigid body and hence it allows to verify if all the forces (body force, viscous forces, etc) are correctly computed and if the Explicit ISPH gives the same trends as the Riemann-SPH, δ -SPH and FVPM solvers. This test case is a simple version of complex real world problems in the field of naval and offshore engineering, e.g. the launching of lifeboats, mooring buoys dynamics, immersing submarines etc.

Simulation	Method	Kernel	Reffinement	Correction Factor	Neighbor Particles
1	Riemann-SPH	Wendland	$\frac{L_b}{\Delta x} = 25$	-	50-60
2	Riemann-SPH	Wendland	$\frac{L_b}{\Delta x} = 50$	-	50-60
3	Riemann-SPH	Wendland	$\frac{L_b}{\Delta x} = 100$	-	50-60
4	Exp. ISPH	Wendland	$\frac{L_b}{\Delta x} = 25$	-	50-60
5	Exp. ISPH	Wendland	$\frac{L_b}{\Delta x} = 50$	-	50-60
6	Exp. ISPH	Wendland	$\frac{L_b}{\Delta x} = 100$	-	50-60
7	δ -SPH	Gaussian	$\frac{L_b}{\Delta x} = 25$	$\alpha_\delta = 0.01 ; \delta_\delta = 0.02$	50
8	δ -SPH	Gaussian	$\frac{L_b}{\Delta x} = 50$	$\alpha_\delta = 0.01 ; \delta_\delta = 0.02$	50
9	δ -SPH	Gaussian	$\frac{L_b}{\Delta x} = 100$	$\alpha_\delta = 0.01 ; \delta_\delta = 0.02$	50
10	FVPM	Circular	$\frac{L_b}{\Delta x} = 24$	-	37
11	FVPM	Circular	$\frac{L_b}{\Delta x} = 50$	-	37
12	FVPM	Circular	$\frac{L_b}{\Delta x} = 100$	-	37

Table 7.7: Configuration for each SPH variant used to simulate the drowning body

The fluid density is $\rho_0 = 1.0 \text{ kg/m}^3$ and gravity is set to $g_0 = 1.0 \text{ m/s}^2$. The tank dimensions are $L_t = 4.0 \text{ m}$ and $H_t = 5.0 \text{ m}$. Two simulations were performed for each method tested: one with a free surface and another without a free-surface. For the first one a water depth of $D_w = 3.0 \text{ m}$ is considered. The body dimensions are $L_b = 1.0 \text{ m}$ and $H_b = 0.5 \text{ m}$, its mass is $m = 1.0 \text{ kg}$ which makes it twice as heavy as the surrounding fluid and its inertia momentum is $I = 0.083 \text{ kg m}^2$ (in order to compute this inertia, the mass was considered as being homogenously distributed). The simulation

was performed in 2D where all the degrees of freedom of the body were free. For both simulations a viscosity of $\mu = 1/500$ Pa s was used.

For each set-up (with and without free-surface) a total of 12 simulations were performed, to which the parameters are shown in table 7.7. Comparisons will focus on the center of gravity motion and the body's roll angle throughout the time simulation. We will confront the methods for each particle distribution and each simulation configuration.

•With free-surface

Figures 7.28, 7.29 and 7.30 show, respectively, the time history of the center of gravity position in the x-direction and y-direction, and the body's roll angle.

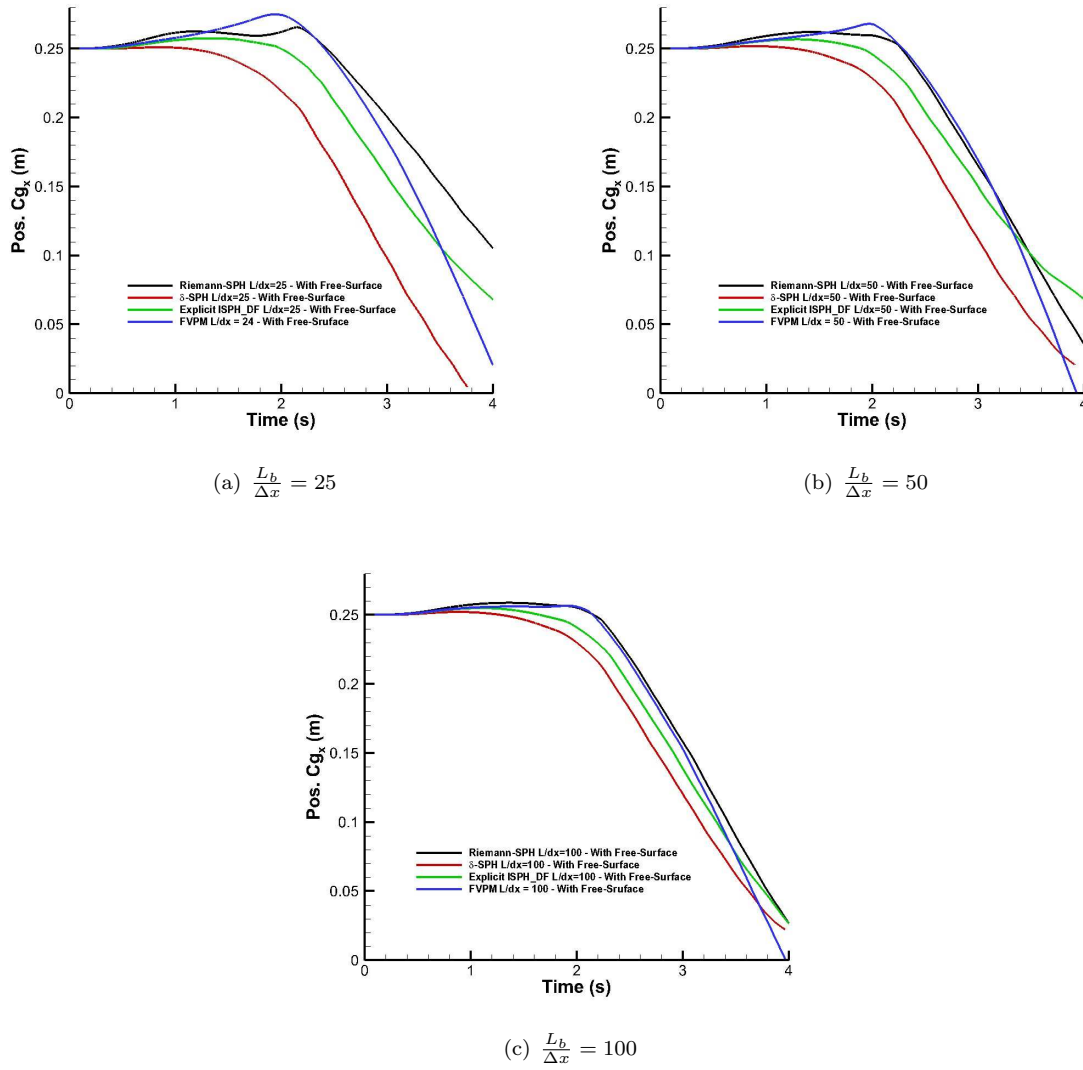


Figure 7.28: Center of Gravity position (x-direction) comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 with free-surface

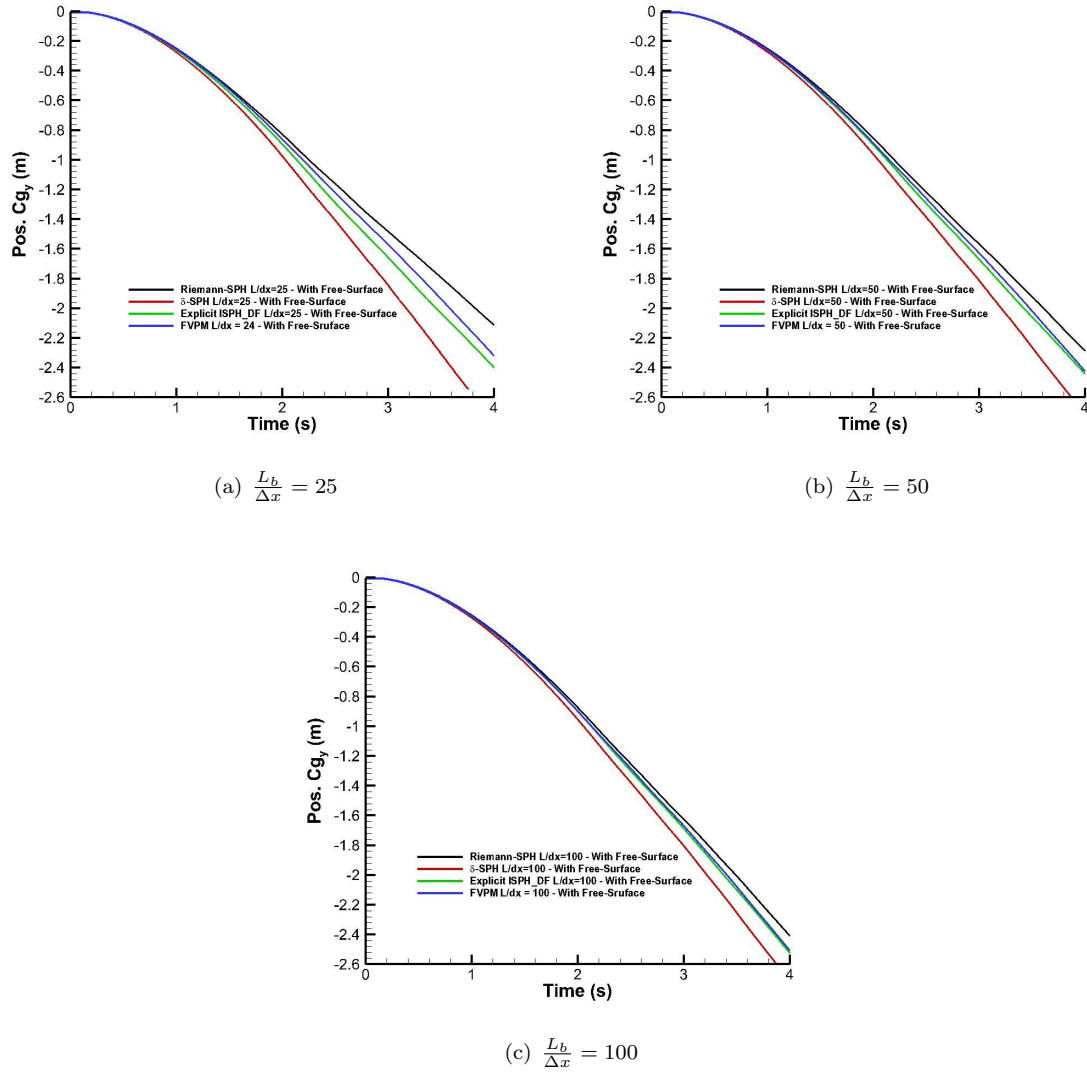


Figure 7.29: Center of Gravity position (y-direction) comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 with free-surface

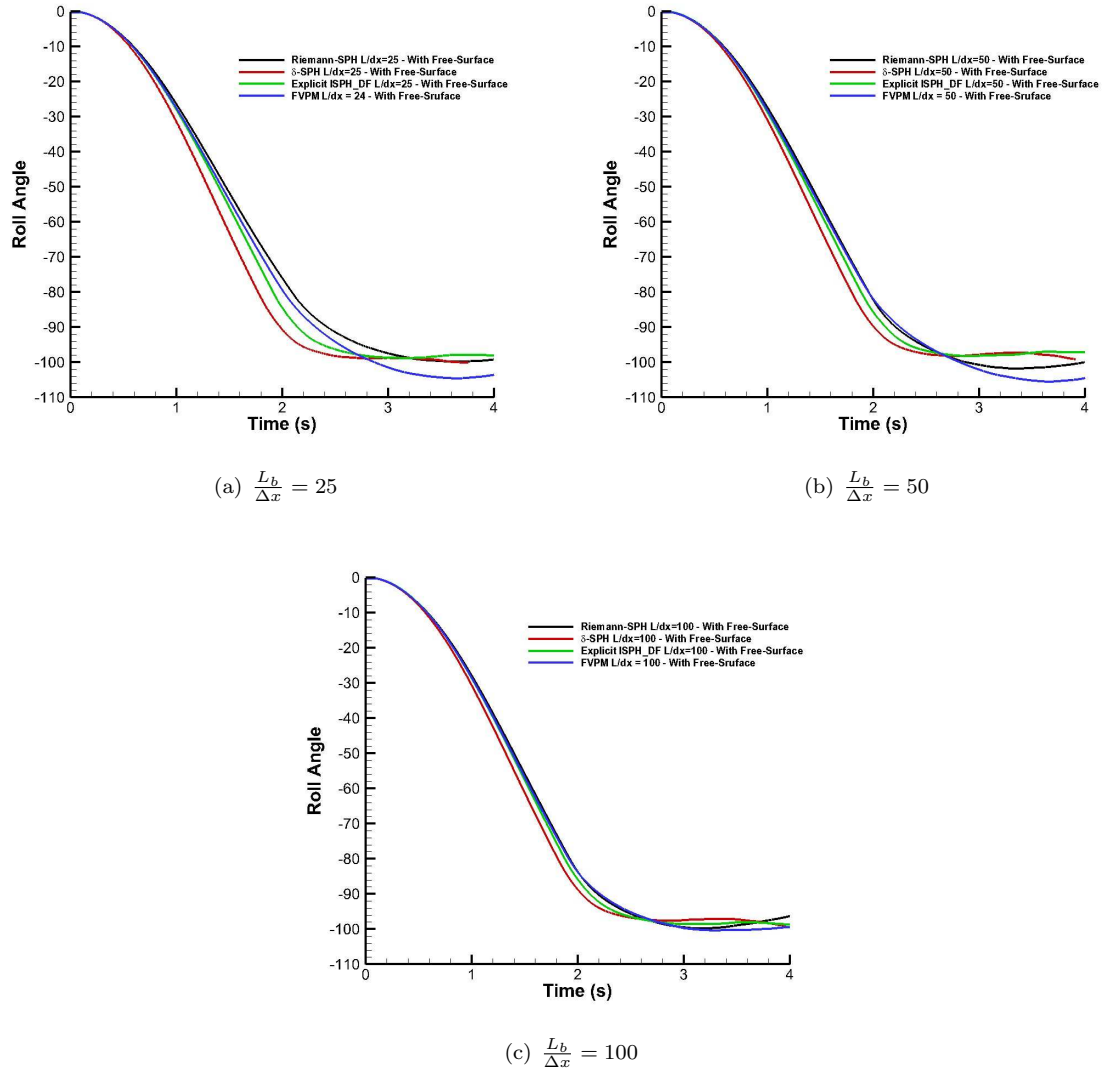
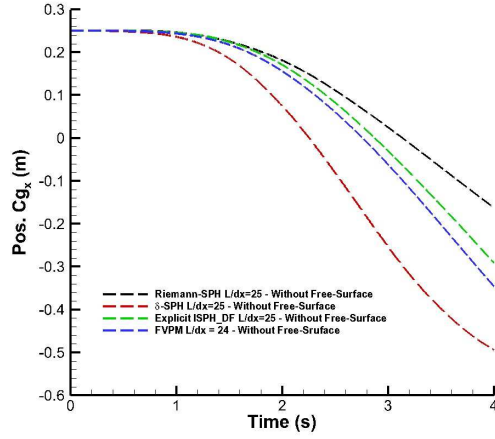


Figure 7.30: Body's roll angle comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 with free-surface

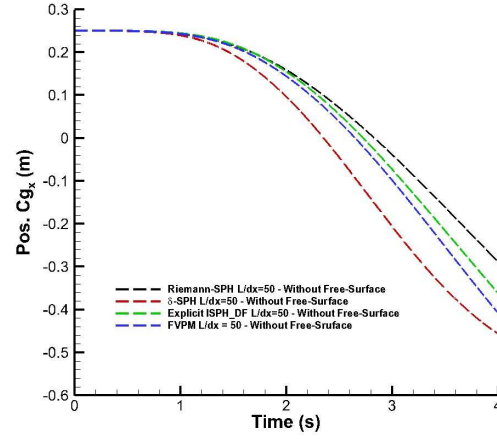
Looking at these results, we can see how FVPM and Riemann-SPH give very similar results, especially if we consider the body's roll angle (figure 7.30). As for Incompressible SPH and δ -SPH, the less intensive algorithms in terms of CPU costs (cf. comparison below), they behave similarly. However, when analyzing the vertical center of gravity history, one may notice that, since from the simulations with less particles, Incompressible SPH and FVPM have almost the same behavior.

•Without free-surface

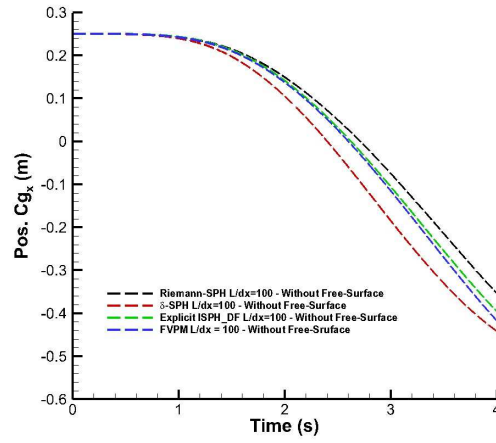
Figures 7.31, 7.32 and 7.33 show, respectively, the time history for the center of gravity position on the x-direction and y-direction, and the body's roll angle.



(a) $\frac{L_b}{\Delta x} = 25$



(b) $\frac{L_b}{\Delta x} = 50$



(c) $\frac{L_b}{\Delta x} = 100$

Figure 7.31: Center of Gravity position (x-direction) comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 without free-surface

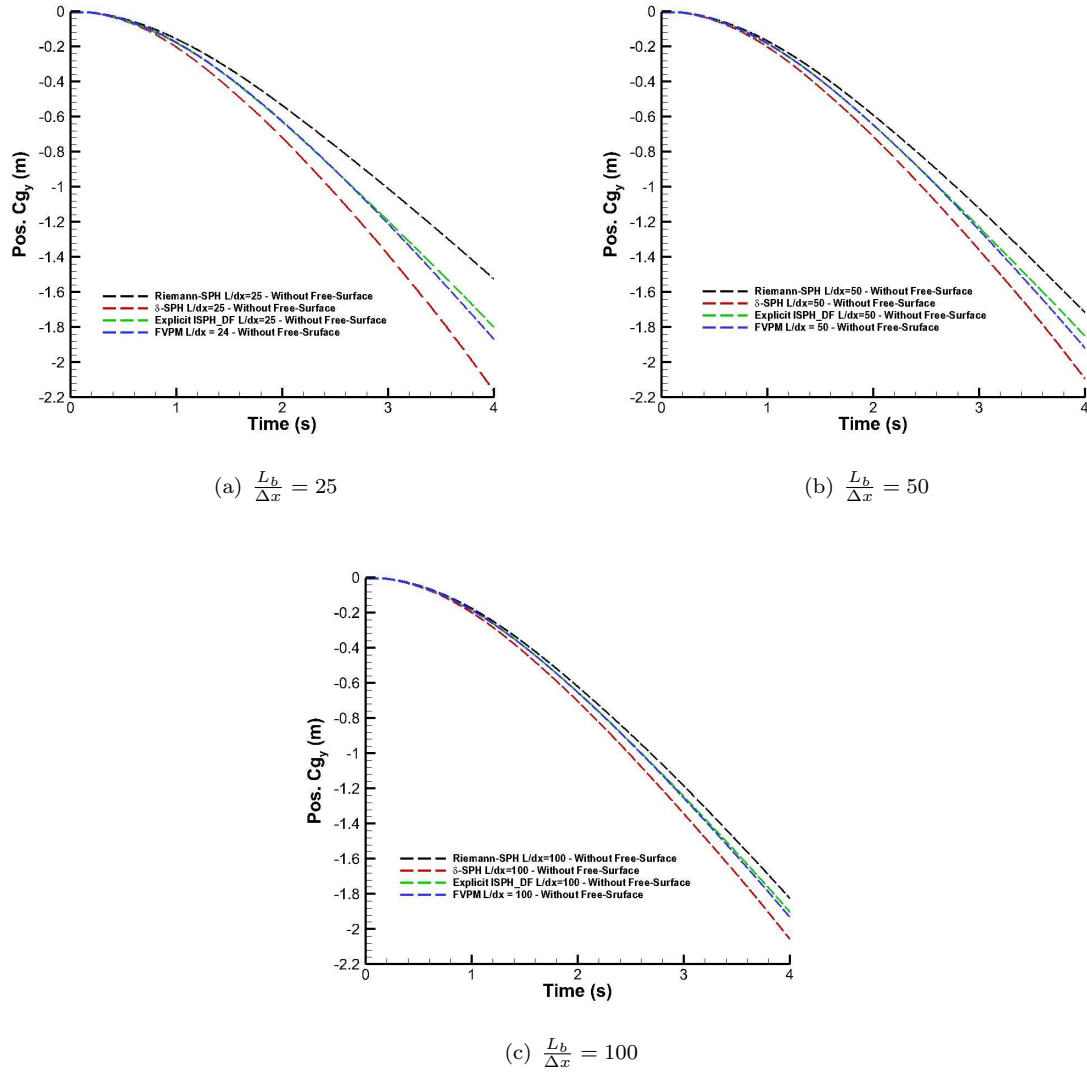


Figure 7.32: Center of Gravity position (y-direction) comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 without free-surface

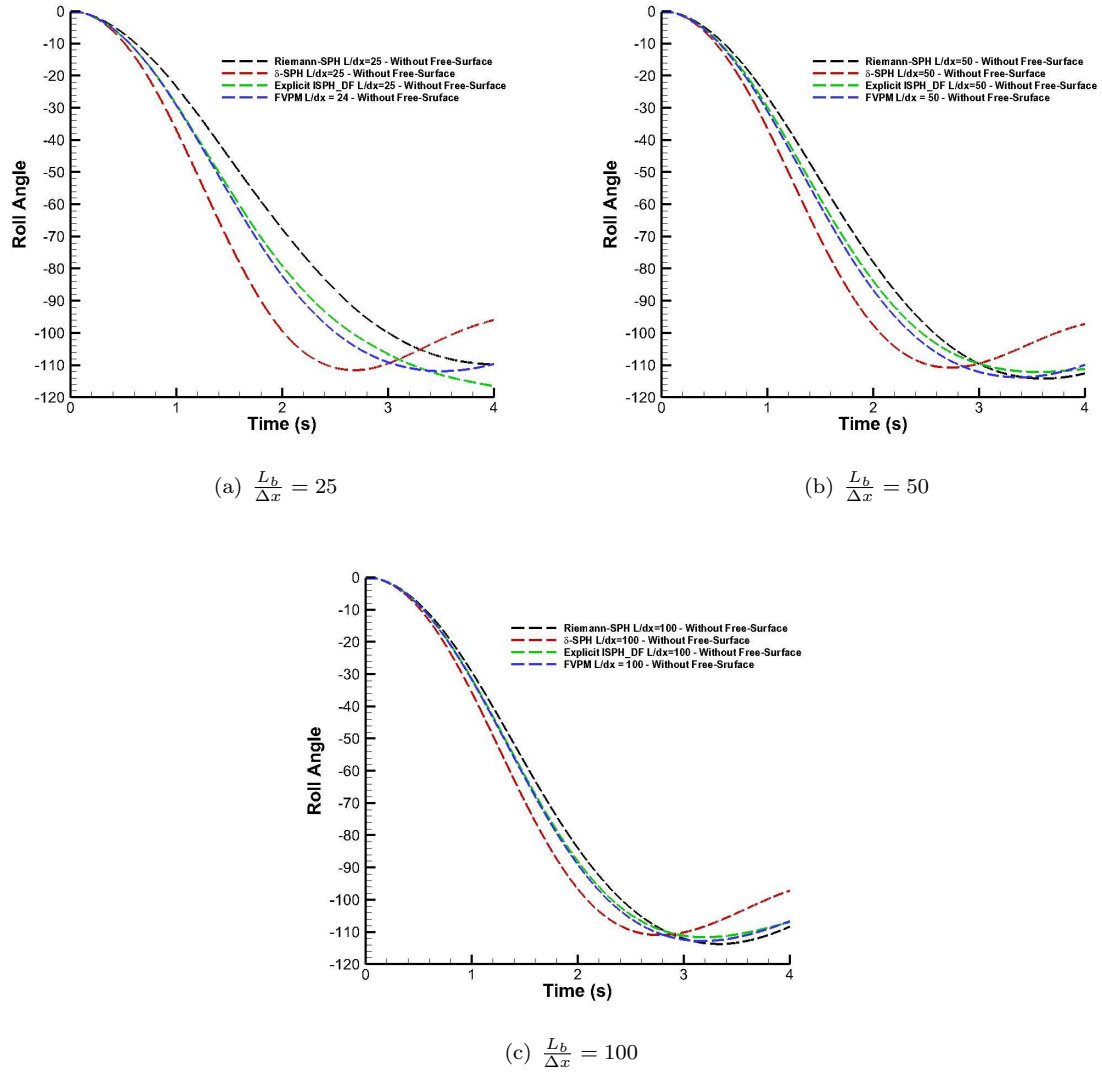


Figure 7.33: Body's roll angle comparison for: simulations (a) 1, 4, 7, 10; (b) 2, 5, 8, 11; and (c) 3, 6, 9, 12 without free-surface

For the simulations without the free-surface, we see that Incompressible SPH, Riemann-SPH and FVPM behave similarly. δ -SPH, as the number of particles is increased, tends to the same results. If we consider FVPM to be the most precise algorithm (given the similarity of its operators to the Finite Volume method ones), we can state that Explicit ISPH is able to give very good results with less CPU resources (cf. below).

In order to have a better view on the compromise "quality of results - CPU costs" of the methods used in this comparison, we draw a raw comparison (the difficulty relying on the fact that each method has its own architecture and was simulated on different hardwares in the framework of the NextMuSE project). Considering the $\frac{L_b}{\Delta x} = 100$ with free-surface simulation (which totalizes 120.000 particles approximately) we have:

- δ -SPH simulation took 1 hour and 6 minutes on a 7 cores Xeon E5410 2.33 Ghz.
- FVPM simulation took 91 hours on a 1 core Xeon W3520 2.67Ghz.
- Riemann-SPH simulation took 3 hours and 1 minute on a 8 cores Intel Nehalem EP 2.8GHz.
- Explicit ISPH simulation took 59 minutes on a 8 cores Intel Nehalem EP 2.8GHz.

It can be roughly stated that Incompressible SPH and δ -SPH have almost the same CPU costs for this particular test case, being faster then Riemann-SPH and FVPM by factors of 3 and 10 respectively. However, globally, the new explicit Incompressible SPH presents the best compromise “CPU costs - quality of results” since it seemed to behave better than δ -SPH on this test case.

7.5.2 Sloshing phenomena

Sloshing is a recurring problem in Naval Engineering. During the conception phase of naval and offshore structures, it is crucial to correctly compute the forces and tensions caused by the sloshing inside a tank, as it directly affects the stability and the structural dimensioning. Sloshing is characterized by its violent fluid impacts against the tank’s side structure, hence SPH is suited to simulate and analyze this problem as it has been done by several authors [99, 100, 101, 102, 103].

On the SPHERIC (SPH European Research Interest Community) website [90] the sloshing experiments (with result data) are available. Here we will use one of them [5] where tank’s movement is imposed and two fluids are considered: water and oil. For each fluid two levels of filling are considered. Here we will focus only on the lower level ($H_f = 0.093m$). Figure 7.34 shows the sketch of the problem with the pressure sensors present in the experiment. Here only the results for the pressure sensor 1 are considered.

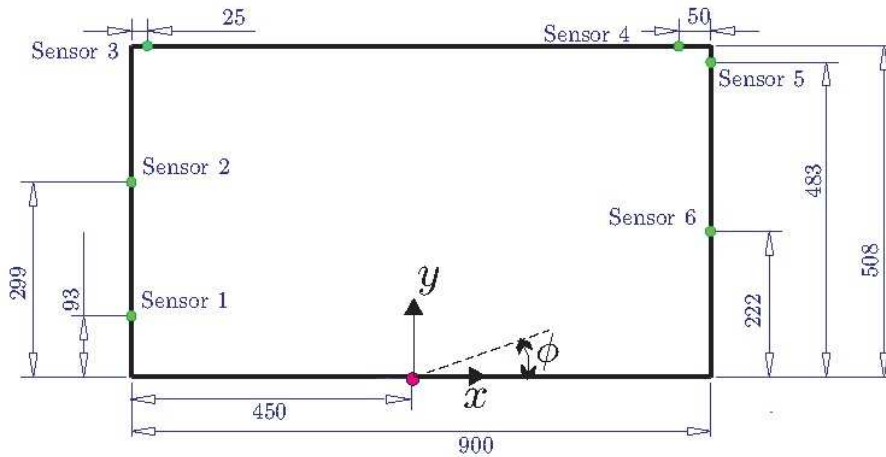


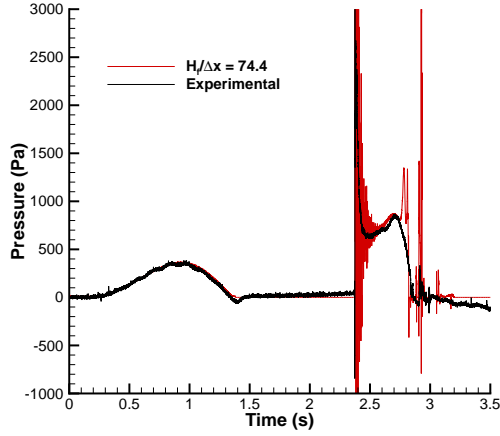
Figure 7.34: Sloshing problem sketch (figure from [5]).

The results obtained considering a water and an oil filling are presented below. Four particle distributions were used but here, we will only show the results for the most refined particle distribution:

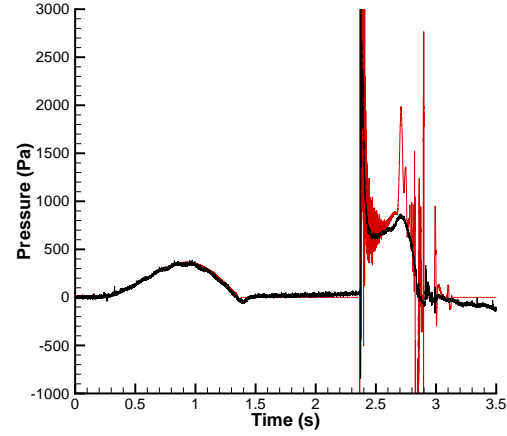
$$\frac{H_f}{\Delta x} = 74.4.$$

•Water filling

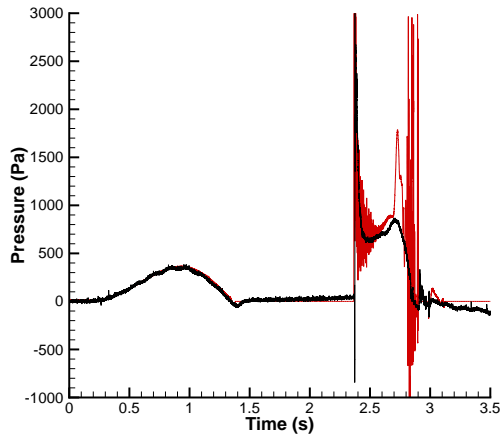
At $t = 0.0s$ the free-surface is flat and the tank is in an horizontal position. The water is considered to have a density equal to $\rho_0 = 998kg/m^3$ and to be inviscid. The gravity is equal to $g_0 = 9.81m/s$. For Explicit ISPH we used an XSPH correction factor $\varepsilon_{XSPH} = 0.02$ and δ -SPH was tuned with the following values: $\alpha_\delta = 0.05$ $\delta_\delta = 0.1$.



(a) ISPH



(b) Riemann-SPH (Godunov)



(c) Riemann-SPH (Acoustic)

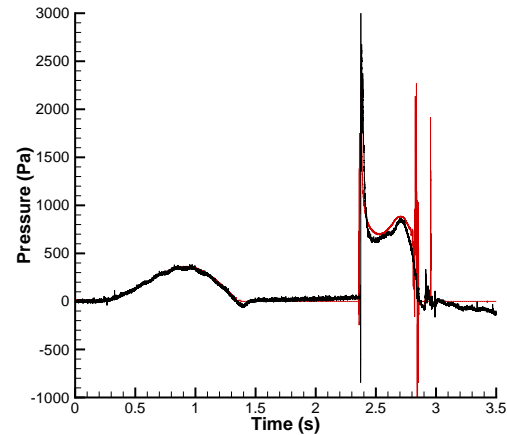
(d) δ -SPH

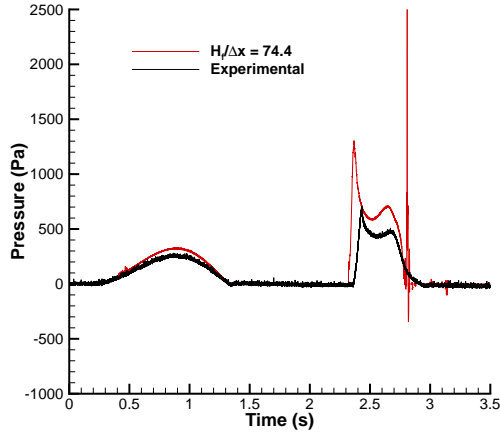
Figure 7.35: Results obtained for different SPH formulations for the sloshing test case with water filling.

The new explicit Incompressible SPH approach presents the same level of noise than SPH with exact and approximate Riemann-solvers. δ -SPH presents the lowest level of noise for the pressure during the first impact against the left wall, but we remember that the amount of dissipation in the

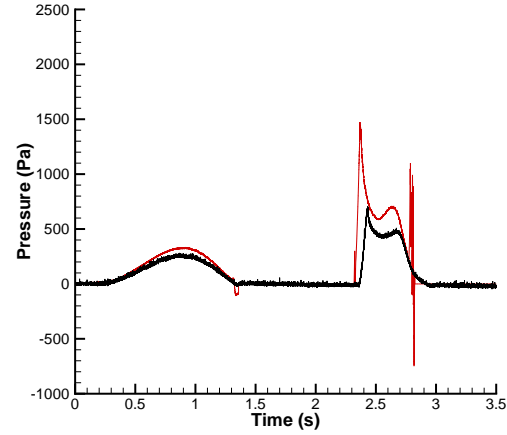
mass conservation equation can be tuned, which is not the case of the other methods. Globally, ISPH behaves similarly to the WCSPH variants but it does not seem to be the best choice to simulate this kind of problem, however, due to its lower CPU costs, it may be used to have a first estimation of the results.

•Oil filling

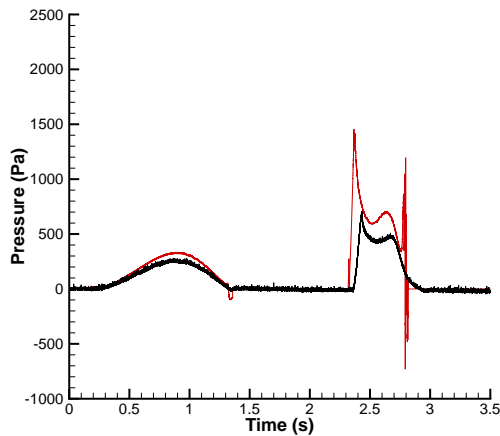
At $t = 0.0s$ the free-surface is flat and the tank is in an horizontal position. The oil considered is a sunflower oil with density equal to $\rho_0 = 990kg/m^3$ and a dynamic viscosity $\mu_0 = 0.045Pa.s$. The gravity is equal to $g_0 = 9.81m/s$. For the Explicit ISPH no XSPH correction was used since the fluid is viscous enough and for the δ -SPH was tuned with the following values: $\alpha_\delta = 0.0$ $\delta_\delta = 0.1$.



(a) ISPH



(b) Riemann-SPH (Godunov)



(c) Riemann-SPH (Acoustic)

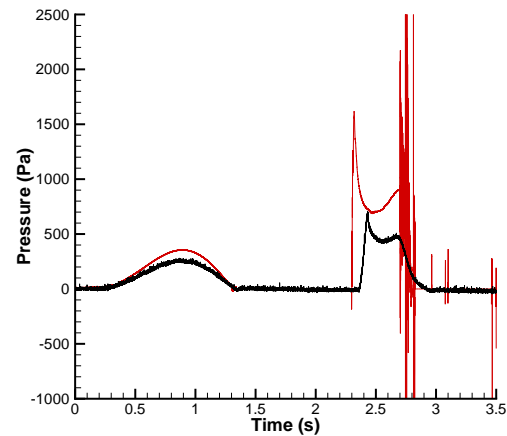
(d) δ -SPH

Figure 7.36: Results obtained for different SPH formulations for the sloshing test case with oil filling.

Contrary to what was observed for the water-filling, δ -SPH presents the most noisy results. This may be a consequence of the fact that the diffusion in the momentum conservation equation was set to zero. We considered the fluid viscosity to be enough to stabilize the flow. On the other hand, when comparing explicit ISPH to Riemann-SPH results, the same trends as for the water filling are observed even though a large overestimation of the pressure is found with respect to the experiments, maybe due to a bad estimation of the oil viscosity value. Looking only at the results given for the oil filling test case, we may conclude that ISPH can be used as an alternative to WCSPH.

Chapter 8

Discussion

In this part, a new fully Explicit Incompressible SPH method has been presented. It is based on the semi-implicit incompressible approach which has first been analyzed in this part. We showed that such a semi-implicit formulism has some problems regarding the compromise “CPU costs-quality of results”, in other words, it is not possible to have lower CPU costs in comparison to WCSPH without increasing the time steps, which in turn compromises the quality of results. These conclusions are similar to what had been already verified in the FEM context. In order to better justify the use of an Incompressible SPH approach we developed and proposed a new fully explicit variant by which we avoid the computational costs connected to the solution of the Pressure Poisson Equation. Besides, the new proposed method restores one of the main advantages of the (Weakly-Compressible) SPH approaches, namely, the fact that the free surface dynamic condition is intrinsically taken into account by the chosen operators, which was not the case of semi-implicit ISPH. As a result, there is no need to perform any particle detection procedure, which allows to keep a simpler algorithm and to reduce the computational costs with respect to semi-implicit ISPH. Moreover, we avoid any problem connected to the need of imposing the dynamic free-surface condition, e.g. instabilities and the ill conditioned system of equations (PPE).

The new method was submitted to a thorough validation process. Looking at the literature, one may find validation results, but often considering only classical benchmark test cases and, sometimes, too simple to assess the true capability of the incompressible SPH method. The explicit version introduced and presented in this thesis was applied/validated both to free-surface flows (standing wave, dam-break problems, impinging jet flow, etc) and to viscous flows (lid-driven cavity and flow around a cylinder) without forgetting to mention flows towards Naval/Offshore Engineering applications, like the interaction with a rigid body and sloshing phenomena. For all these test cases, the novel ISPH algorithm showed similar and, in some cases, better results than the existing SPH variants (including the semi-implicit ISPH formulation). For all the test cases presented in this thesis, a lower CPU cost has been observed with this new Explicit ISPH method than with any other variant (semi-implicit ISPH or improved WCSPH). We may foresee the extension of this new ISPH method to a wider range of fluid flows: wave propagation, multi-phase flows and fluid-structure interaction.

We may also point out some weak points of this novel SPH method. The first one regards the explicit solution itself. As one may notice, the solution proposed does not ensure that the incompressibility

condition is achieved at every time step as it is the case of the semi-implicit approach (given that, for this last one, we ensure that the iterative solver converges). This may be seen contradictory as we still refer to it as being an Incompressible variant, but we remind that we use small time steps, and as shown in the previous section, the results given by the explicit approach are equivalent to the semi-implicit ones, both regarding the quality of results and stability issues. This leads us to state that our assumption is correct, that is, the solution of the PPE may be done explicitly but the time step must be kept within a limiting range of stability. However, this statement still lacks a strong theoretical (mathematical) support, which may be brought by a future Ph.D. or M.Sc. research project.

The second point that we consider important to highlight is the one regarding the stabilization of free-surface flows. Unfortunately, the novel ISPH approach inherits instability issues from the semi-implicit approach, mainly due to the imprecision of the Laplacian operator. This corroborates what had already been found out in the literature, i.e. that this problem is related to kernel truncation errors. As a consequence, the new explicit approach needs an stabilizing technique when applied to violent free-surface flows (for the viscous flow problems such stabilization technique was not necessary). However, we found out that only a small level of correction is necessary ($\varepsilon_{XSPH} = 0.01$), similarly to what is observed for δ -SPH, where only a small amount of diffusion is needed in the mass conservation equation to achieve good results.

Another important point is the tensile instability issue. During our validation process, more specifically for the lid-driven cavity flow, we were confronted to some tensile instability (TI) issues as pointed out briefly in section 7.3.1. This TI leads to voids in the flow which compromise the simulation. These instabilities were present when using correction terms as the renormalization. This leads us to state that the method is sensible to more precise SPH operators and, therefore, we leave as a suggestion for the researchers that, in the future, propose or intend to use such corrections, to verify if these instabilities increase as operators precision does. As an example of these instabilities, we show in figure 8.1 a result obtained for the lid-driven cavity flow at Reynolds number $Re = 3200$ using a renormalization correction.

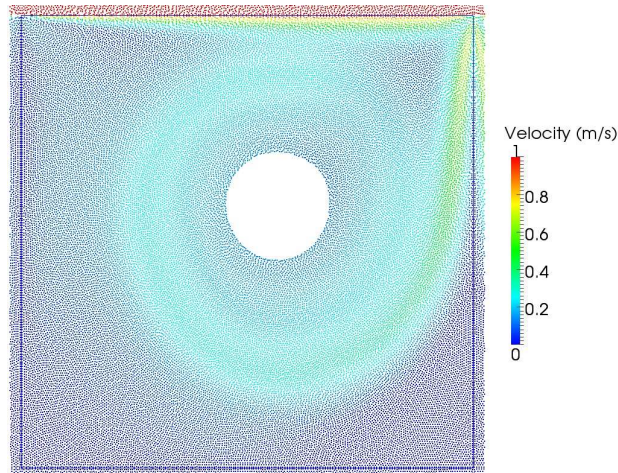


Figure 8.1: Void formation due to tensile instability for $Re = 3200$ with renormalization.

Lastly, even if with this new explicit ISPH approach a slightly better compromise between CPU

costs and quality of results is reached (comparing to Riemann-SPH and δ -SPH), the same problems found for any SPH method are still present. In order to get very good results in SPH, avoiding any numerical noise or numerical viscosity, it is necessary to increase the convergence ratios conditions which implies on higher CPU costs. That is to say that the convergence issues are still encountered for this new approach as SPH operators did not change.

As general conclusion we may state that the proposed method is a good candidate in the continuous improvement of the Smoothed Particle Hydrodynamics method.

Part III

Adaptivity in SPH

Chapter 9

Introduction

Traditionally SPH simulations are performed with a uniform particle distribution. In some cases the size of the fluid domain is very large and using a constant distribution of particles in the entire domain may become very consuming in terms of CPU and storage. In order to prevent this problem, some authors proposed the use of variable size particle distribution [104, 105, 53, 106, 107, 108, 109, 110, 111, 52, 15, 17]. One of these alternatives is to use a ‘variable-h’ distribution where the particle interspace is smoothly increased from the region of interest up to the borders of the domain. In these simulations, similarly to traditional meshed-based method, the zones of interest are defined with a more concentrated particle distribution. However, due to the Lagrangian nature of SPH, particles can be mixed-up together, so that such an approach may lead to strong instabilities. Additionally, this technique implies the use of a special particle ‘meshers’ in order to take into account different regions of interest. With this approach one of the qualities of the SPH algorithm is lost: facility to generate the initial particle distribution. We could add the fact that with a ‘variable-h’ approach the particles growth rate is very low due to stability issues, which limits its efficiency and its range of applicability.

In order to address this problem, it is more reasonable to define a dynamic refinement criterion where any particle reaching such criterion is projected onto a finer particle distribution. In the SPH community this is called adaptivity. The first efforts towards adaptive particle refinement in SPH were done for astrophysical purposes using a density-based criterion to change the particle resolution in regions of interest [112, 113, 114, 115, 116]. We may also make reference to the works by Bouscasse *et al.* [117] where a multi-purpose interface technique was presented which allows to couple SPH with another method/solution, which, for instance, can be another SPH solution. They presented results on a coupling of two SPH solutions: one with a finer particle distribution and another courser one. With such a coupling, they achieved to increase the precision of the SPH solution locally. Recently, Feldman [7] and Feldman *et al.* [6] proposed a particle splitting technique where one bigger (*mother*) particle is projected onto several smaller (*daughter*) particles. Among the available techniques the latter seems to be the most promising, and for that reason, in this thesis we decided to study its application to flow phenomena within the field of violent and free-surface flows. However, besides refining, it would be interesting to be able to derefine particles. Aiming at the improvement of adaptive techniques in SPH, we will propose a new particle derefinement method that complements the existing particle refinement technique.

Besides, the techniques studied and presented in this thesis present similarities is found in mesh-based methods to adapt the spatial/temporal discretization during a simulation: Adaptive Mesh Refinement (AMR). Hence, throughout this part, whenever possible, a comparison/parallel between what is called here Adaptive Particle Refinement (APR) and AMR will be made.

Chapter 10

Particle Refinement

As stated, in this thesis the particle refinement approach presented in [7, 6] is studied. In such a method, the *daughter's* properties such as mass, volume, density, velocity and pressure are chosen in a way that energy is conserved and density continuity is preserved during the refinement process. This technique was a major step towards automatic adaptivity in SPH as it permits to have a finer distribution wherever and whenever needed. This technique has already been applied to many problems: from viscous flows [6] to shallow water problems [118] (for which some slight changes were needed). Later, Lopez *et al.* [8] improved Feldman *et al.* [6] technique by considering as variable to be conserved the density rate of change, which implied the use of derivative operators and thus a higher level of accuracy was achieved. A similar but much simpler method was proposed by Omidvar *et al.* [119] permitting to obtain good results for free-surface flows and fluid-structure interaction (rigid body).

However, such a refinement technique [7, 6, 8] has not been validated using test cases that are typically treated using SPH, an inviscid dam-break simulation for instance. Moreover, it has only been used in the framework of WCSPH with artificial viscosity [120]. We decided, in this thesis, to proceed to a validation of such an adaptive refinement technique using Riemann-SPH [25] and δ -SPH [34] which are more precise variants of SPH and hence, the effect (discontinuities, instabilities, etc) of using a dynamic refinement is expected to be better seen/measured. This is done using the test cases used in [7, 6], a simple hydrostatic benchmark and also an inviscid 2D dam-break which is a representative violent flow often simulated with SPH.

The following sections will focus on the theory and validation of such a dynamic refinement technique.

10.1 Theory

In the algorithm proposed by Feldman [7] and Feldman *et al.* [6], when a particle needs to be refined (called *mother* particle), it is divided into a finite number of smaller particles (called *daughter* particles) following a pre-defined pattern. This refinement is done in such a way that local density errors are minimized.

First of all we must define two refinement variables: the separation parameter $\epsilon_{PR} \in [0..1]$ and the

radius ratio parameter $\alpha_{PR} \in [0..1]$. The first determines the distance between *daughter* particles and the second defines the radius length of the *daughter* particles with respect to the *mother* particle one. With d standing for *daughter* particle and n for the *mother* particles, the equations

$$\Delta x_d = \epsilon_{PR} \Delta x_n \text{ and } R_d = \alpha_{PR} R_n \quad (10.1)$$

can be written. We may also define a mass ratio $\lambda_d \in [0..1]$ which will define the mass of each *daughter* particle:

$$m_d = \lambda_d m_n, \quad (10.2)$$

knowing that in order to conserve mass the following restriction must be respected:

$$\sum_j^D \lambda_j = 1. \quad (10.3)$$

where D is the number of *daughter* particles originated one *mother*. When a particle is refined into smaller ones, the local properties of the fluid are modified, leading to an error on the estimation of a function and its gradient. Figure 10.1 illustrates the splitting process, going from a coarse particle distribution to a refined one: *mothers* (red) are split into a finite number of *daughter* particles. Within the method proposed by Feldman *et al.* [6] the refinement parameters are chosen in a way that the *local refinement error* on the density estimation is minimized.

Before the refinement, the density at a given location is defined by:

$$\rho(\vec{r}) = \sum_j^M m_j W_{\vec{r}j}, \quad (10.4)$$

where M is the number of *mother* particles present on the yet not refined domain. Now we assume that one of the *mother* particles is refined (split) into D *daughter* particles resulting on a new density estimation:

$$\rho^*(\vec{r}) = \sum_j^{M-1} m_j W_{\vec{r}j} + \sum_j^D m_j^* W_{\vec{r}j}. \quad (10.5)$$

Based on equations (10.4) and (10.5) we can define the *local density refinement error* as:

$$e^\rho(\vec{r}) = \rho(\vec{r}) - \rho^*(\vec{r}) = m_M W_{\vec{r}M} - \sum_j^D m_j^* W_{\vec{r}j}. \quad (10.6)$$

Finally, based on this *local error* the *global refinement error* reads:

$$E(\vec{r}) = \int_{\Omega} e d\vec{r}. \quad (10.7)$$

When expression (10.7) is developed we find that the *global refinement error* depends on the refinement parameters: α_{PR} , λ and ϵ_{PR} . In other words, the radius length, mass and the positions of the *daughter* particles must be chosen in a way that the *global refinement error* on the density estimation is minimized. For more details, refer to [7, 6].

Lopez *et al.* [8] proposed an improvement of the Feldman *et al.* procedure. In their algorithm they minimize the error on the gradient of a function f . The value of such gradient after a *mother* particle has been refined is given by:

$$\nabla f(\vec{r})^* = \nabla f(\vec{r}) - \frac{m_M}{\rho_M} f(\vec{r}_M) \nabla W_{\vec{r}M} + \sum_j^D \frac{m_j^*}{\rho_j^*} f(\vec{r}_j) \nabla W_{\vec{r}j}. \quad (10.8)$$

Based on this equation, the *local refinement error* expression for the gradient of a function can be defined:

$$e^{\nabla f}(\vec{r}) = \nabla f(\vec{r}) - \nabla f(\vec{r})^*. \quad (10.9)$$

In order to define the values for α_{PR} , λ and ϵ_{PR} a variable must be chosen for f . Lopez *et al.* [8] chose the density, as in [6], with the difference that here the density material derivative is considered. Hence, equation (10.9) writes:

$$e^{\frac{D\rho}{Dt}}(\vec{r}) = \left\langle \frac{D\rho(\vec{r})}{Dt} \right\rangle - \left\langle \frac{D\rho(\vec{r})}{Dt} \right\rangle^*. \quad (10.10)$$

Applying the SPH operators, the expression for the *global refinement error* for the gradient of the density can be defined:

$$E_{\frac{D\rho}{Dt}}(\vec{r}) = m_M^2 \int_{\Omega} \left[(u(\vec{r}) - u(M)) \cdot \left(\nabla W_{ij} - \sum_j^D \lambda_j \nabla W_{\vec{r}j} \right) \right]. \quad (10.11)$$

Equation (10.11) depends on particle velocities and therefore such an equation would have to be minimized at each time step for every particle. Obviously, that is very costly and, for that reason, it is considered that particles velocities do not change during the refinement process. Consequently, equation (10.11) is rewritten as:

$$E_{\nabla W}(\vec{r}) = \int_{\Omega} \left[\frac{\partial W_{\vec{r}M}}{\partial \vec{r}} - \sum_j^D \lambda_j \frac{\partial W_{\vec{r}j}}{\partial \vec{r}} \right] d\vec{r}^*. \quad (10.12)$$

Equation (10.12) gives the *global kernel gradient refinement error*. As all of the SPH derivatives are based on the evaluation of the kernel gradient, Lopez *et al.* [8] variant is expected to be more precise than Feldman *et al.* [6].

Within these two particle refinement methods, the values for α_{PR} , λ and ϵ_{PR} can be chosen so that equations (10.7) or (10.12) are minimized. As the choice of these parameters' values is independent of the flow characteristics, these values are defined prior to the simulation. Each author proposed optimum values for these variables according to the refinement pattern that is used. In this chapter, these values are respected to ensure fairer comparisons and conclusions.

For both techniques, the energy, mass and momentum conservation during the refinement process must be ensured. This is done simply by respecting the conditions presented in Table 10.1.

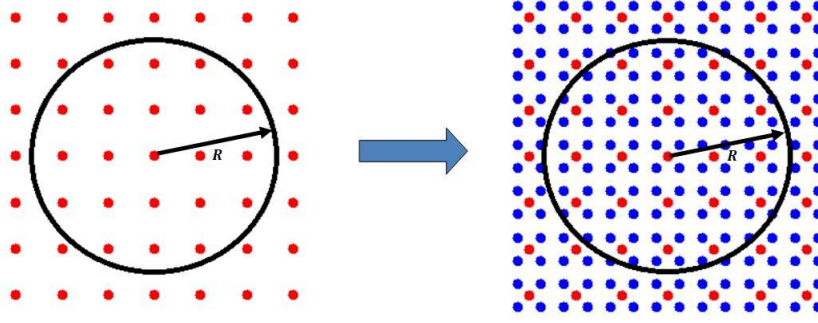


Figure 10.1: Illustration of the refinement process and on the change of the particles inside the kernel: *mother* (red) particles are split into a finite number of *daughter* (blue) particles.

Table 10.1: Four conditions needed to be ensured during a particle refinement process.

Quantity	Before Refinement	After Refinement
(1) Mass	m_n	$\sum_j^D m_j$
(2) Kinetic Energy	$\frac{1}{2} m_n \vec{u}_n \cdot \vec{u}_n$	$\frac{1}{2} \sum_j^D m_j \vec{u}_j \cdot \vec{u}_j$
(3) Linear Momentum	$m_n \vec{u}_n$	$\sum_j^D m_j \vec{u}_j$
(4) Angular Momentum	$\vec{r}_n \times m_n \vec{u}_n$	$\sum_j^D \vec{r}_j \times m_j \vec{u}_j$

The first condition is respected by equation (10.2). The other conditions are ensured by copying the *mother* particle velocities on the *daughters*. For more detail on these refinement techniques, we refer the reader to [7, 6, 8].

10.2 Refinement Criteria

In order to apply this splitting technique, a refinement criterion must be defined. Basically, there are two possible criteria: physical and spatial. The first relies on observing a certain physical quantity and whenever a threshold value is exceeded for a particle, it is split. We can choose as variable to be observed the pressure, density, velocity or another quantity like pressure gradient, velocity divergence, pressure Hessian, etc. However, the physical criterion can only be successfully applied to smooth flow phenomena and by using a stable version of the SPH algorithm. That way the well-known fluctuation that can be observed during an SPH simulation can be considered as minimum and a physical criteria could be used. However, a threshold value must be defined which can be tricky for certain flows. Even for Riemann-SPH or δ -SPH, the smoothness of the simulation is fragile (oscillations in pressure and velocities may occur). For that reason, the use of a physical criterion may not be adequate for SPH.

The range of application of SPH in our laboratory covers mainly violent flows (wave impacts, sloshing, green-water, etc) where abrupt changes in the pressure and velocity are often encountered. Based on this analysis, the spatial criterion seems to be the most suitable to be applied to SPH algorithms as it relies only on the particle positions. Also, it is a criterion that can be easily defined and that allows having several refinement regions with different characteristics. Besides, it seems to be the criterion that suits the best to the Lagrangian nature of SPH methods. Therefore, in this thesis,

we use this criterion to define the regions of the domain where a more refined particle distribution is needed.

10.3 Refinement Patterns

When a *mother* particle is split into its *daughter* particles, a refinement pattern must be followed to place particles with respect to the origin (i.e. where the *mother* lies). In [7] two 2D and one 3D pattern were proposed and in [8] only one 2D pattern was presented. For all these patterns, standard values for α_{PR} , λ and ϵ_{PR} are proposed in [7, 6, 8].

Feldman *et al.* proposed two particle patterns for 2D simulations:

- First pattern: a four particle distribution centered at the position of the *mother* particle: in figure 10.2 the blue particle is the *mother* one (that is kept after refinement) and the red are the *daughter* particles. This pattern will not be used as it is known not to perform as well as the second pattern that follows.

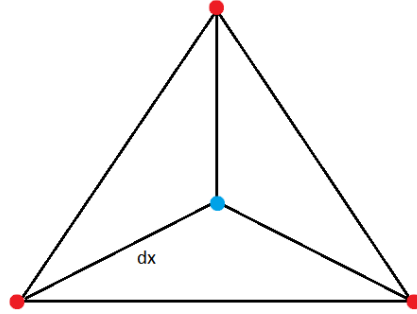
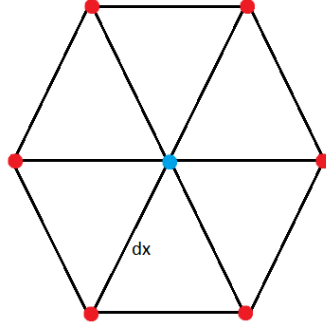
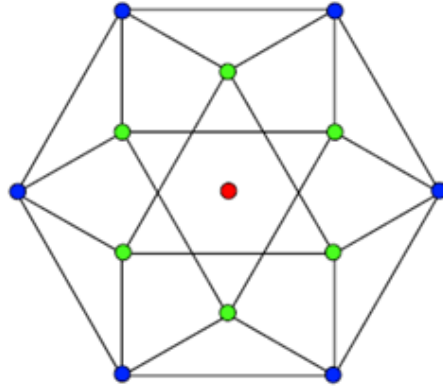


Figure 10.2: Feldman *et al.* [6] first 2D distribution pattern.

- Second pattern: a seven particle distribution centered at the position of the *mother* particle: in figure 10.3 the blue particle is the *mother* one (that is kept after refinement) and the red are the *daughter* particles.

For both the last 2D patterns, Feldman [7] proposed to use $\alpha_{PR} = 0.6$ and $\epsilon_{PR} = 0.6$ in order to have a minimized error on the density estimation. The values for λ vary according to the refinement pattern and to the *daughter* particle position (farther it is from the *mother* particle less mass it will inherit from the latter). For more detail refer to [7]. These values are respected for the simulations presented in section 10.6. Note that the *mother* particle is replaced by a *daughter* in the previous refinement patterns.

The 3D refinement pattern presented by Feldman is an extension of the last 2D pattern presented in figure 10.3. It has the total of 13 particles as presented in figure 10.4 and the same values for α_{PR} and ϵ_{PR} as those used for the 2D patterns are advised.

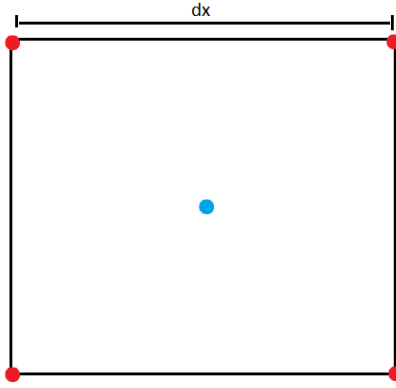
Figure 10.3: Feldman *et al.* [6] second 2D distribution pattern.Figure 10.4: Feldman *et al.* [6] 3D distribution pattern (figure from [7]).

Lopez *et al.*, on the other hand, presented only one particle pattern for 2D simulations that is shown below:

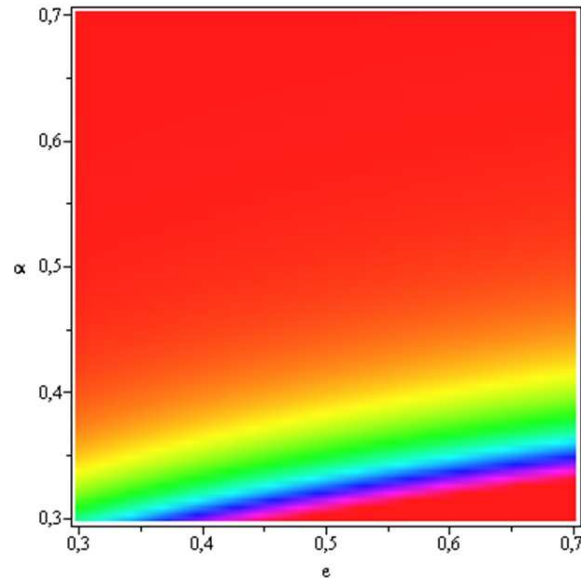
- Lopez pattern: a squared four particle distribution centered at the position of the *mother* particle: in figure 10.5 the blue particle is the *mother* one and the red particles the *daughter* ones.

For this latter pattern $\lambda = 0.25$ is used for all the *daughter* particles and in [8] it is stated that $0.60 \leq \alpha_{PR} \leq 0.65$ and $0.50 \leq \epsilon_{PR} \leq 0.55$ ensure a good trade-off between CPU costs and minimal refinement error for the kernel gradient estimation. Note that if $\alpha_{PR} = 1.0$ and $\epsilon_{PR} = 0.5$ are used together with this refinement pattern, the solution proposed by Omidvar *et al.* [119] is found.

Using this refinement pattern, we can trace the error following the two main parameters α_{PR} and ϵ_{PR} : we vary these parameters from 0.3 up to 0.7, which ensures that there will be less neighbor particles for the *daughter* particles than for the *mother* and avoids clumped configurations. Figure 10.6 shows the kernel gradient error using the Wendland kernel: the region in red (from $\alpha_{PR} > 0.45$ on) represents a relative error less than 1%. This confirms the results obtained by Lopez *et al.* Moreover,

Figure 10.5: Lopez *et al.* [8] 2D distribution pattern.

the trends observed in figure 10.6 go on the direction of the two parameters that dictate the convergence in SPH, that is $\Delta x \rightarrow 0$ and $\frac{\Delta x}{R} \rightarrow 0$ (lower Δx values at constant $\frac{R}{\Delta x}$) which may lead us to conclude that similar values for α_{PR} , λ and ϵ_{PR} parameters in [7, 6, 8] may be found for any other kernel function.

Figure 10.6: Graphic for the kernel gradient error following Lopez *et al.* (read α_{PR} and ϵ_{PR}).

We propose a 3D extension of this pattern by simply copying twice the 2D one in two sections equidistant from the *mother* particle resulting on an eight particle distribution. This refinement pattern is used for one test case in this thesis and we assume that the conclusions depicted for the 2D simulations can be extended to 3D.

Note also that an orientation may be used to place the *daughter* particles. It may be defined as being the local direction of the flow (based on the velocity field). However, for the current study the

orientation is fixed and equal to the global axis.

10.4 General Remarks on the Particle Refinement Technique

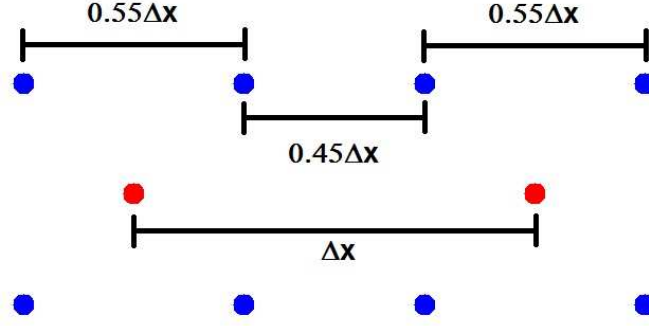
We may, however, draw some remarks/comments on the state of the art of particle refinement applied to SPH, mainly by comparing it to the AMR methods. In [47] (we only cite one article, but a very extensive bibliography may be found on the matter), for instance, the time step varies following the mesh discretization, i.e., the regions with the most refined mesh have smaller time steps than the coarse-mesh regions, and as a consequence these first zones are integrated in time using more time-steps than the last ones. That is to say that the AMR technique is adapted both to the spatial and temporal discrimination of the physical system. This is not the case, yet, of the Adaptive Particle Refinement applied to SPH: the smaller time step is used for all the particles regardless of their size.

Also, in AMR, during a refinement process, the gradient of the physical quantities are respected. In other words, when a cell is split, the value of the pressure, velocities, etc respect the local gradient evolution in order to avoid discontinuities and consequently a stable simulation is achieved. In the particle refinement procedure presented in [7, 6, 8] this is not done and that leads, as it will be shown in section 10.5, to small instabilities.

Moreover, the technique developed in [7, 6, 8] was analyzed using the standard artificial viscosity WSPH method [120] which is known for its great amount of noise in the pressure field. For that reason, it is not so clear in these articles if the proposed particle refinement technique is really well performing regarding quality of results, especially the pressure field. In this part such a technique is validated using more precise SPH techniques (Riemann-SPH and δ -SPH) and results are shown in sections 10.5 and 10.6.

Some other aspects regarding convergence issues can be commented. If $\epsilon_{PR} = 0.55$ is used together with the refinement pattern of Lopez *et al.*, a particle configuration as depicted in figure 10.7 is achieved, where two *mother* particles are shown in red together with their *daughter* particles in blue where Δx stands for the initial inter-particle space. Each *mother* particle has four daughters and the figure shows that these *daughter* particles are spaced of $0.55\Delta x$ one from another but they are spaced of $0.45\Delta x$ from the *daughter* particles ‘belonging’ to the other *mother* particle. Consequently the average particle space is of $0.5\Delta x$ instead of $0.55\Delta x$, i.e., when Lopez *et al.* refinement pattern is used the final average value for ϵ_{PR} is to be considered always being equal to 0.5. This ‘hidden’ feature of the particle refinement has as collateral effect the fact that the *daughter* particle volumes (which is an abstract quantity in SPH) overlap and even if contradictory, this overlapping makes the simulation more stable. Note that if the refinement patterns presented in [7, 6] are considered the same conclusions are less evident. In these works $\epsilon_{PR} = 0.6$ is considered with a triangular 4 particle and a hexagonal 7 particle refinements and in these cases the volume overlapping is much higher and may cause undesired effects on the stability of the simulation.

Lopez *et al* [8] stated that the upper limit of α_{PR} must be respected in order to avoid clumped conditions but this was not be observed during the development of this thesis. Actually, if we recall the convergence criteria already mentioned in this thesis:

Figure 10.7: Uneven particle inter-particle space when using $\epsilon_{PR} = 0.55$.

$$\frac{R}{\Delta x} \rightarrow \infty \quad (10.13)$$

and

$$R \rightarrow 0. \quad (10.14)$$

and if we analyze the values suggested in [8] for α_{PR} and ϵ_{PR} it can be verified that during the refining procedure both criteria are respected at the same time. The second one is directly given by the value of α and the first one by the ratio $\frac{\alpha_{PR}}{\epsilon_{PR}}$ (which is equal to 1.2 for $\alpha_{PR} = 0.6$ and $\epsilon_{PR} = 0.5$). However, if the value of α_{PR} is increased, the second convergence criterion starts to be lost but this is automatically compensated because the first one is increased, which is translated into a gain in the quality of the SPH interpolation. Classically, in SPH, when the quality of the SPH interpolation is increased, particles tend to in-line formations along the trajectories, especially when starting with regular arrangements (e.g. Cartesian lattice). These in-line formations must not be confused with clumped configurations. In this thesis $\alpha_{PR} = 0.7$ is considered whenever not mentioned otherwise, which allows a good rate of decrease of R per refinement level and also a higher increase of $\frac{R}{\Delta x}$.

Typically, when an experienced SPH user needs to perform a convergence study, $\frac{R}{\Delta x}$ is kept constant and Δx is decreased, which is numerically translated into respecting equation (10.14). On the other hand, when refinement takes place the ratio $\frac{R}{\Delta x}$ increases as R decreases and consequently, from the CPU costs point of view, convergence when using the particle refinement procedure may be faster but leads to higher CPU costs per particle interpolation. This point was not treated in the referenced articles on the subject and we consider it to be important to have better conclusions on the efficiency of the particle refinement procedure.

10.5 Static Refinement Simulations

This first test case aims at showing the effect of using Lopez *et al.* [8] algorithm in a simple (but often difficult to SPH) fluid problem. A rectangular fluid domain with a free-surface is considered, as illustrated in figure 10.8. The central domain (region A) highlighted with dark lines will have a

different particle refinement than the rest of the domain (region B). Two simulations were performed using the following configurations:

- “Normal” simulation (1): the central domain has a particle refinement of $\Delta x_A = \frac{\Delta x_B}{2}$ and the ratio $\frac{R}{\Delta x}$ in each region follows the values obtained by the “dynamic” simulation (cf. below);
- “Dynamic” simulation (2): at $t = 0$, before computing the time derivatives, the particles inside the dark box are refined using the Lopez *et al.* procedure with $\alpha_{PR} = 0.7$ and $\epsilon_{PR} = 0.55$.

Note that the name “dynamic” is used but the simulation is quasi-static. This is just to emphasize on the fact that a dynamic procedure is used in this simulation. Both simulations were performed for 5 seconds using a density equal to $\rho_0 = 1000 \text{ kg/m}^3$, speed of sound equal to $c_0 = 10 \text{ m/s}$ and the gravity is equal to $g_0 = 9.81 \text{ m/s}^2$.

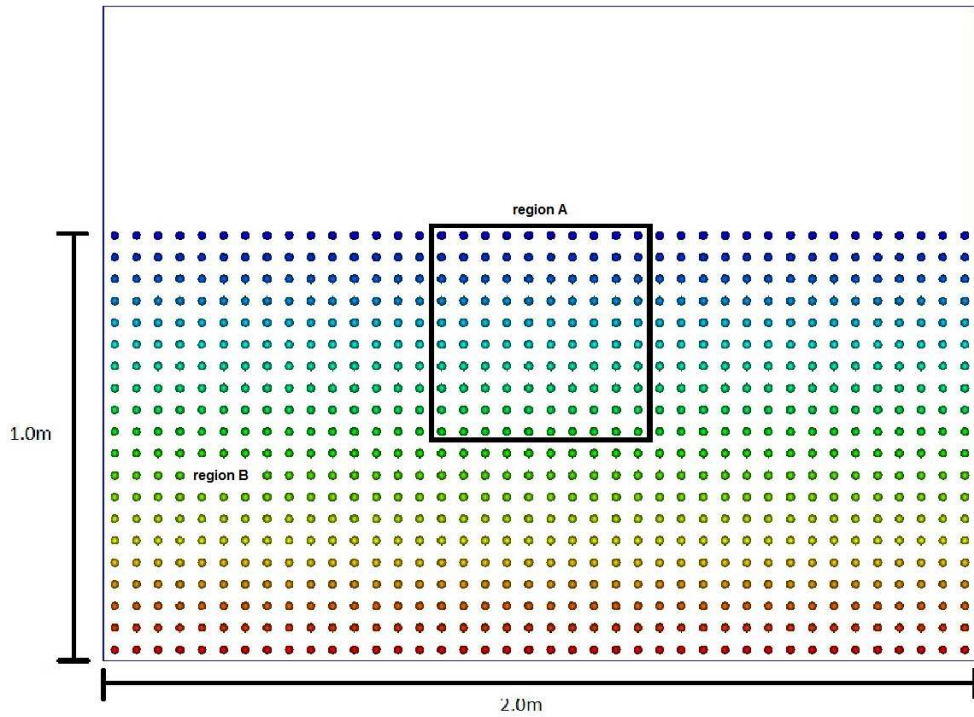


Figure 10.8: Hydrostatic test case: the central domain highlighted with dark lines will have a different particle refinement.

At $t = 0 \text{ s}$ the fluid is in equilibrium and therefore particles shall not move at any instant. In figure 10.9 the pressure in the domain is plotted for some time instants. From the beginning of the simulation ($t = 1 \text{ s}$) the “normal” simulation (left column) starts showing perturbations on the interface between zones A and B while for the stabilized simulation this interface remains almost intact and when the Lopez *et al.* technique is not used, large and asymmetric position perturbations are observed whereas particle displacements are close to zero when such a technique is used. Despite the simplicity of the test case, it shows the robustness of the refinement technique. The fact of using a slightly decentralized particle distribution (with respect to the central *mother* particles) produces

a much more stable simulation. Nevertheless, small deformations are still observed in the “dynamic” simulation and this is related to the fact that during the refinement procedure the pressure gradient (which is an hydrostatic gradient) is not exactly respected and therefore the particle may move a little bit to compensate this discontinuity.

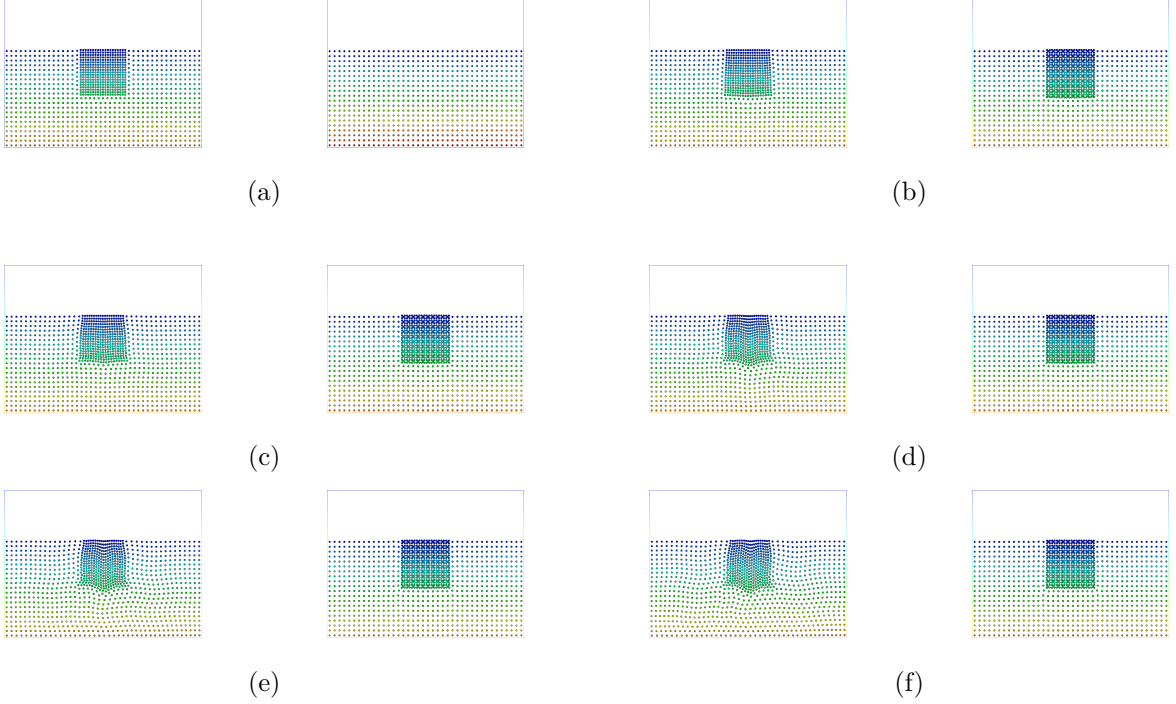


Figure 10.9: Evolution of particle distribution and pressure contours for simulation without any particular treatment (left columns) and with the dynamic refinement treatment by Lopez *et al.* (right columns) at: (a) $t=0.0s$; (b) $t=1.0s$; (c) $t=2.0s$; (d) $t=3.0s$; (e) $t=4.0s$ and (f) $t=5.0s$.

A similar test case was already used by Omidvar *et al.* [119] to assess the performance of their particle refinement technique already mentioned earlier in this thesis. Comparable results were obtained, however the Lopez *et al.* method seems to be better performing. The bigger difference between both techniques relies on the position of the *daughter* particle with respect to the *mother* particle: in [8] particles are $0.05\Delta x$ farther than in [119] which seems to be necessary to have more stable solutions.

10.6 Dynamic refinement simulations

In the previous section, the refinement method presented earlier in this thesis was validated using a test case with a static nature. Here we will consider a series of dynamic test cases which are more relevant for the SPH community. So far, this technique has been validated using the Weakly-Compressible SPH approach with artificial viscosity [120] and here we intend to apply the technique to more precise and reliable WSPH variants, like Riemann-SPH [25] and δ -SPH [34]. Moreover, we expect the effect of using a refinement technique to be more visible when applied to these last two methods than to the

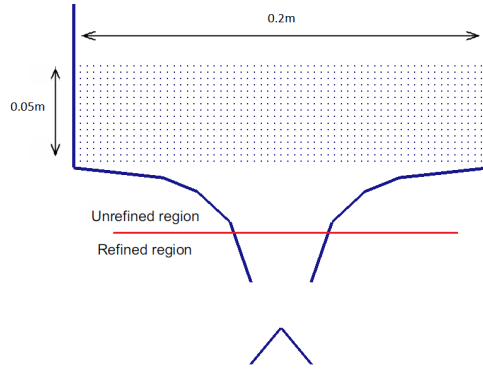


Figure 10.10: Flow separation through a funnel simulation sketch.

classical WCSPH, for which the pressure field is very noisy.

Firstly, we will repeat a test case already used in [6, 7] to validate and show the interest of a particle refinement technique within the framework of Riemann-SPH and δ -SPH methods. Then, a more complex flow phenomenon will be studied: a 2D dam-break problem, which is at the center of the range of application of SPH methods.

10.6.1 Flow separation through funnel

This test case was used by Feldman [6, 7] to validate and to show the advantages of using dynamic refinement within a SPH simulation. In this test case, particles will pass through a funnel and then fall into a separator. While passing through the funnel particles are refined. Due to the local refinement, the fluid separation are expected to be better detected and simulated. The sketch of the problem is shown in figure 10.10. Feldman considered only the WCPSH with artificial viscosity in his study. Here, we decided to validate as well the use of such refinement technique using the Riemann-SPH and the δ -SPH methods.

The initial inter-particle distance is set to $\Delta x = 0.0033$ m, resulting in 915 particles. The total of three simulations were performed:

- Riemann-SPH;
- δ -SPH: where the dissipative parameters are set to $\delta_\delta=0.05$ and $\alpha_\delta=0.1$;
- Artificial Viscosity SPH: where the dissipative parameter is set to $\alpha_{AV}=0.1$;

Figure 10.11 and 10.12 confront Feldman's results (left column) to the ones given by the other variants of the SPH method at $t = 0.13$ s and $t = 0.21$ s. All the simulations performed for this particular test case use the Lopez *et al* [8] refinement pattern whereas Feldman's results were performed with his first 2D particle refinement pattern.

The validation of this test case is done qualitatively, comparing the results given by the different SPH variants to the ones presented in [6, 7]. The results from Feldman seem to be slightly more viscous than the ones from this thesis. This may be due to a higher level of artificial viscosity used by

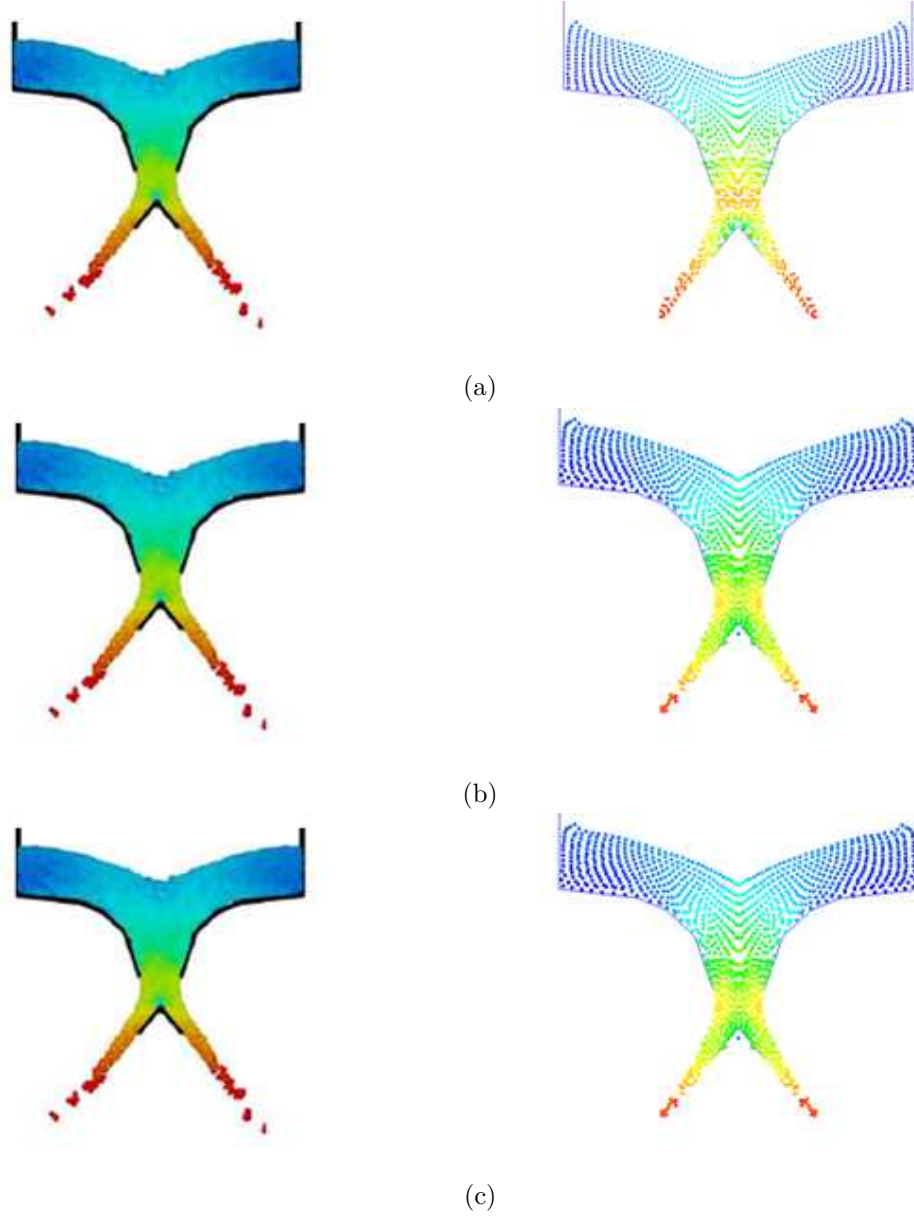


Figure 10.11: Particles passing through the funnel at $t=0.13s$: right column shows (a) Riemann-SPH, (b) δ -SPH and (c) AV-SPH present results and the left column shows Feldman's results (AV-WCSPH only).

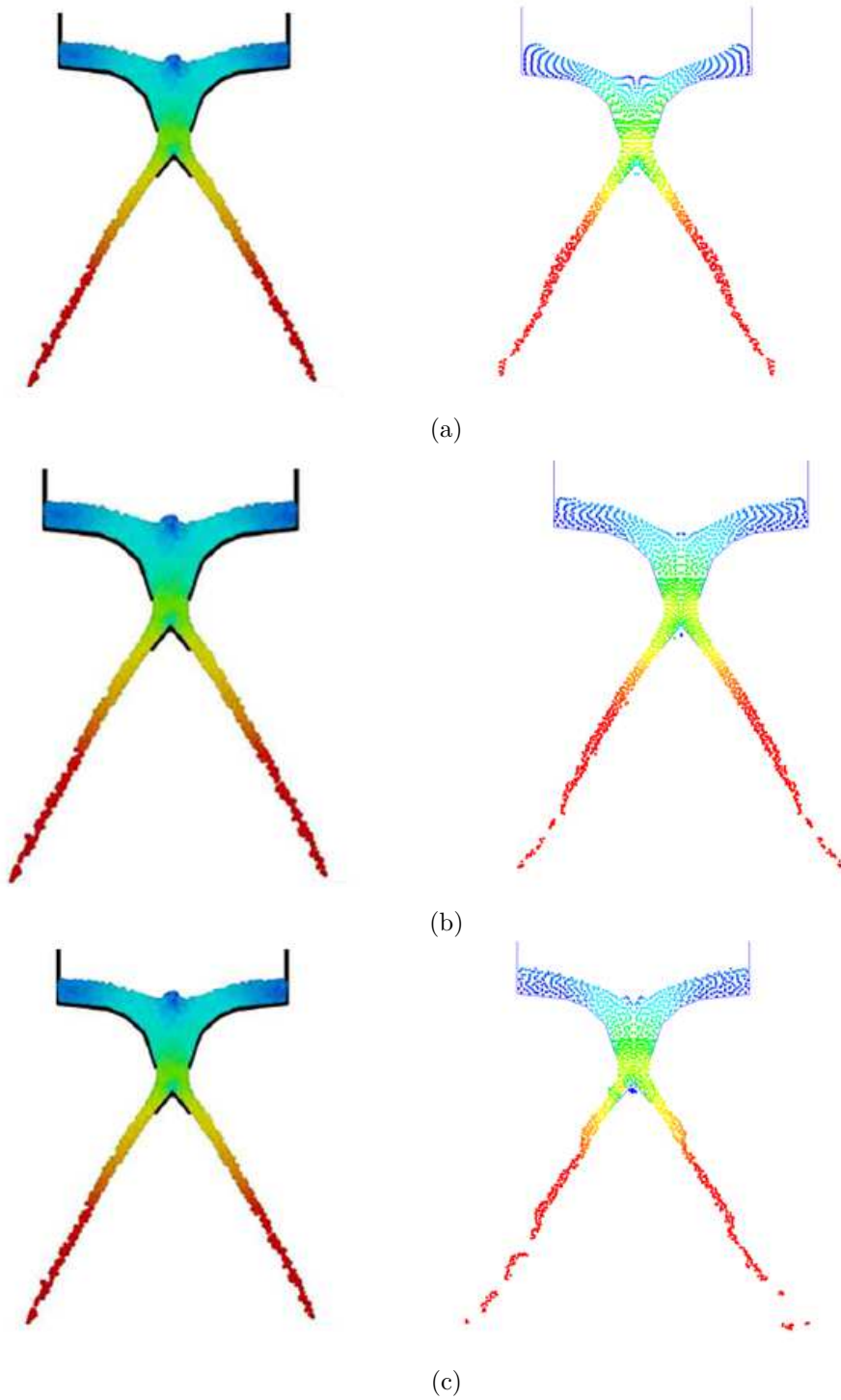


Figure 10.12: Particles passing through the funnel at $t=0.21s$: right column shows (a) Riemann-SPH, (b) δ -SPH and (c) AV-SPH present results and the left column shows Feldman's results (AV-SPH only).

Feldman for his simulations. Nevertheless, we may consider Riemann-SPH, δ -SPH and WCSPH (with artificial viscosity) to be validated as the results are quite similar to Feldman's. Note that, as more precise versions of WCSPH are used, the filaments have a more continuous shape and the particles tend to a more organized distribution.

10.6.2 2D dam-break flow against a wall

A more interesting test case is the dam-break flow, which is a well-known and tested benchmark within the SPH community. The sketch for this test case is presented in figure 7.12, where it was used to validate the ISPH method. We consider a refinement zone close to the pressure sensors with the same height as the domain and with a width of 0.4 m (distance to be measured from the right wall). The pressure measured at $p1$ and the energy behavior throughout the simulation time will be used to compare the different dynamic refinement techniques presented previously.

The intention here is not to validate the SPH method itself, and for that reason, we shall not compare the obtained results to available experimental data. If one wishes to verify the behavior of the SPH methods considered in this study against experimental results, one may refer to the SPH literature (some results are even available in this thesis). Here, we will focus on the use of the dynamic refinement procedure within Riemann-SPH and δ -SPH. The total of four simulations were performed for each method:

- Unrefined particle distribution: no dynamic refinement is used and we start the simulation with an inter-particle distance is $\Delta x = 0.02$ m;
- Refined particle distribution: no dynamic refinement is used and we start the simulation with a Δx similar to the one obtained after the application of the refinement procedure, that is $\Delta x = 0.011$ m;
- Dynamic refinement Lopez: where the initial inter-particle distance is $\Delta x = 0.02$ m and the Lopez refinement [8] pattern is used;
- Dynamic refinement Feldman: where the initial inter-particle distance is $\Delta x = 0.02$ m and the second (2D) Feldman's refinement pattern is used;

The advantage of using a dynamic refinement procedure is to have more particles in a certain zone of interest and that way increase the quality of the results locally. Therefore, we decided to compare the results provided by the different algorithms (Lopez *et al.* [8] and Feldman *et al.* [6, 7]) to a simulation where particle distribution is already the one desired in the region of interest. We performed a simulation with the unrefined particle distribution to check whether a convergence trend can be observed.

• Riemann-SPH

Here we verify the performance of the algorithm using a Riemann-solver based SPH method. Figure 10.13 shows the pressure at pressure probe $p1$ and the (kinetic, potential and internal) energy behavior throughout the simulation time: black line represents the unrefined mesh simulation, green line the refined mesh simulation, blue line the results obtained by Feldman *et al.* algorithm (with the second refined pattern) and red line the results by the refinement pattern by Lopez *et al.*.

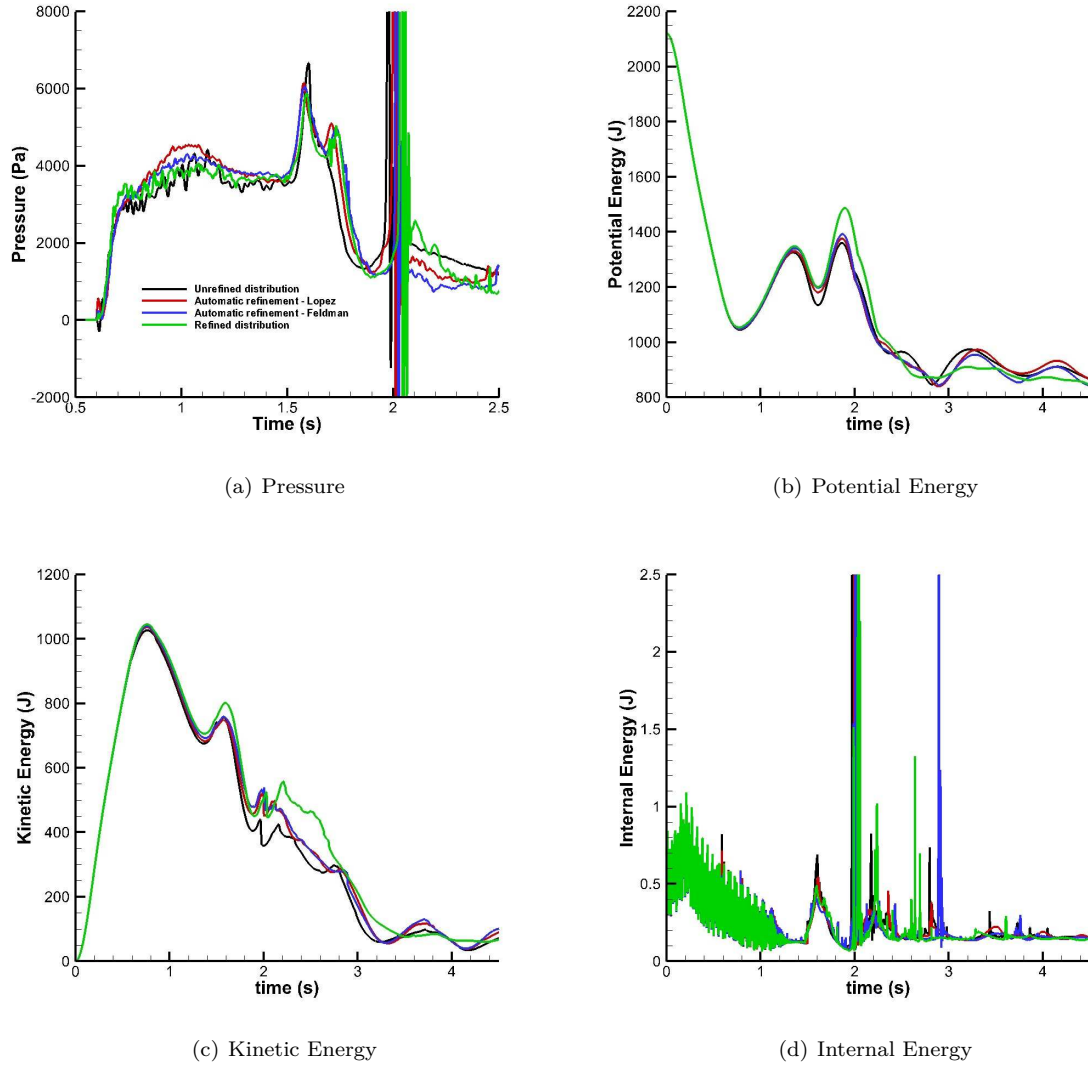


Figure 10.13: Riemann-SPH comparison results for the 2D dam-break flow using a particle refinement technique.

We observe that Feldman *et al.* and Lopez *et al.* techniques behave similarly in terms of energy conservation and quality of the pressure signal. However, Lopez *et al.* particle distribution pattern gives slightly smoother pressure results. Concerning the energy time history, it is interesting to observe that by refining the particles, the energy time history gets closer to the results given by the fully refined simulation, e.g. $t = 2.0$ s for the kinetic energy in figure 10.13-(c). Nevertheless, the energy behavior is much similar to the unrefined simulation as most of the particles are still belonging to the coarser resolution for the dynamic simulations.

In figure 10.14, we plot the pressure fields obtained by Lopez *et al.* algorithm at different time instants.

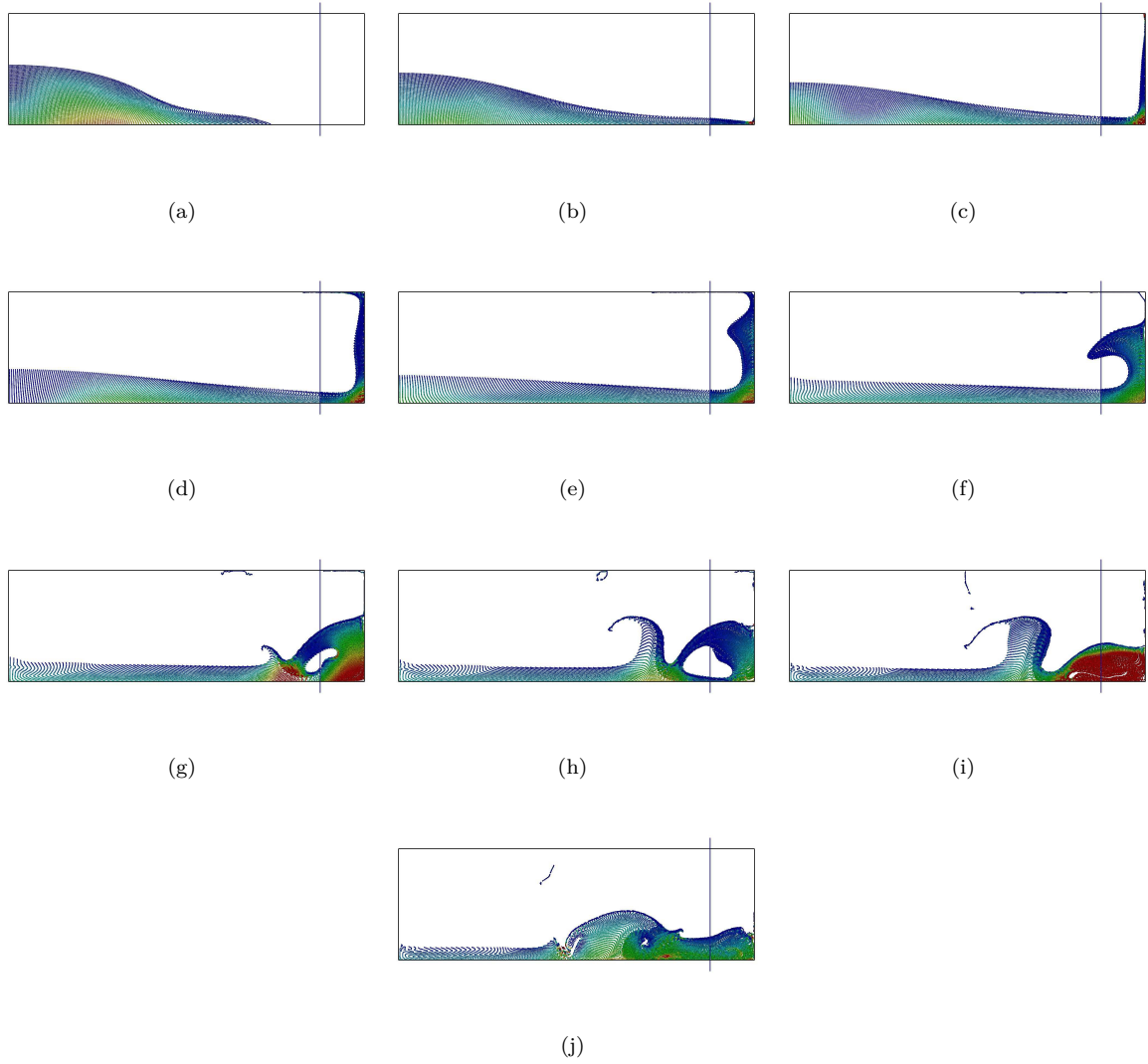


Figure 10.14: Riemann-SPH pressure fields for the Lopez *et al.* [8] refinement pattern for different time instants (blue line represents the begging of the refined area): (a) $t=0.4s$; (b) $t=0.6s$; (c) $t=0.8s$; (d) $t=1.0s$; (e) $t=1.2s$; (f) $t=1.4s$; (g) $t=1.6s$; (h) $t=1.8s$; (i) $t=2.0s$ and (j) $t=2.2s$.

It is interesting to notice that there is almost no discontinuity at the interfaces between the zones with different particle refinements. Especially after the water front impacts the wall, it comes back creating a wave breaking phenomenon, and in the special case of this simulation, the breaking wave is composed by refined particles while the rest of the fluid domain is still with a coarse particle distribution, cf. figure 10.14-(g), and despite that, no pressure discontinuity is created. We may thus consider that the particle refinement technique as presented in [6, 7, 8] may be used with the Riemann-SPH method, with preference given to the Lopez *et al.* technique as it is capable of giving slightly better results.

• δ -SPH

Differently to what was done for Riemann-SPH, we test only the Lopez *et al.* [8] technique with the δ -SPH method. Figure 10.15 shows the pressure at the pressure probe: black line represents the unrefined mesh simulation, the red line the pressure obtained by the Lopez *et al.* algorithm and the green line the refined results. For all the simulations we used $\alpha_\delta = 0.01$ and $\delta_\delta = 0.1$.

Figure 10.15-(a) shows that, contrary to what was observed for the Riemann-SPH method, using a dynamic refinement does not have much effect on the pressure prediction with δ -SPH. The effect is slightly clearer for the energy behavior, but still not considerable.

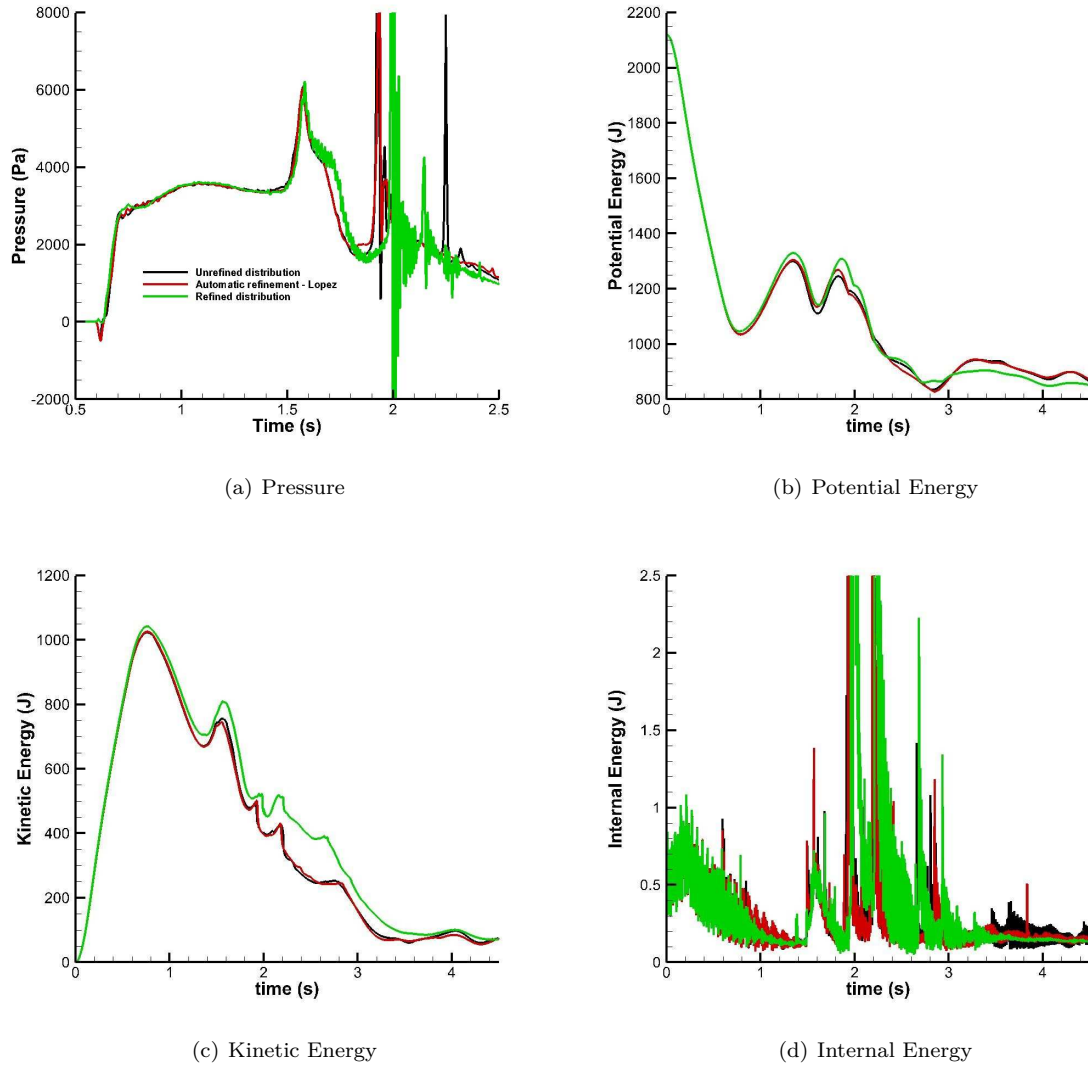


Figure 10.15: δ -SPH comparison results for the 2D dam-break flow using a particle refinement technique.

In figure 10.16, we plot the pressure fields obtained by the Lopez *et al.* algorithm at different time

instants. Despite the results from figure 10.15, the same trends found for the Riemann-SPH method are observed here, i.e., there is almost no pressure instability at the interface between the unrefined and refined zones. We may conclude that the refinement algorithm presented in this chapter may as well be applied to the δ -SPH, but not with a considerable gain in quality of results. We shall add that the particle discretization chosen as coarse and fine are already able to give a good quality of results for this test case.

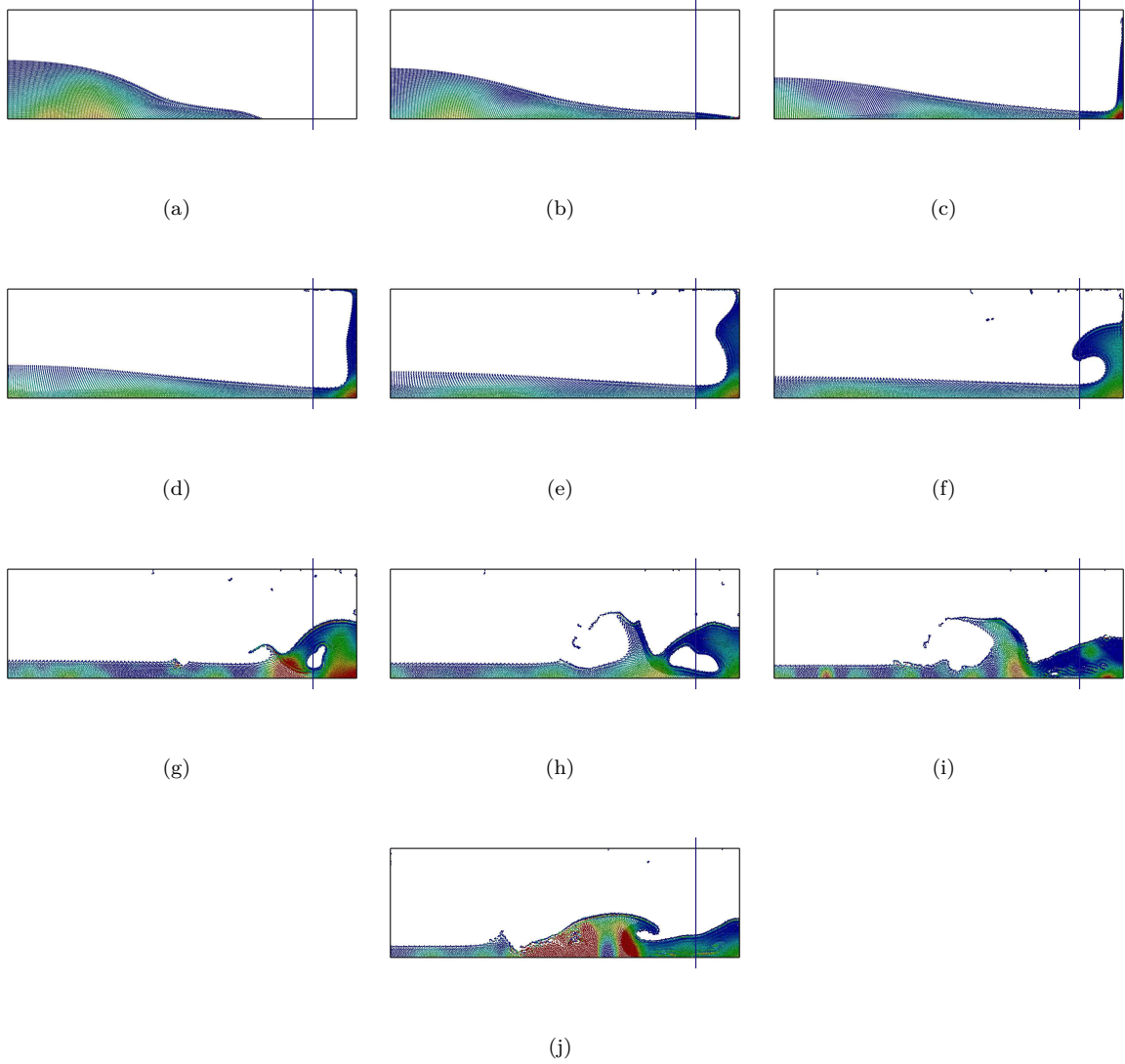


Figure 10.16: δ -SPH pressure fields for the Lopez *et al.* [8] refinement pattern for different time instants (blue line represents the begging of the refined area): (a) $t=0.4s$; (b) $t=0.6s$; (c) $t=0.8s$; (d) $t=1.0s$; (e) $t=1.2s$; (f) $t=1.4s$; (g) $t=1.6s$; (h) $t=1.8s$; (i) $t=2.0s$ and (j) $t=2.2s$.

10.7 Discussion

In this chapter, we presented the particle refinement techniques developed and proposed by Feldman [7] and Feldman *et al.* [6] with the later improvements by Lopez *et al.* [8]. A brief validation and verification process was also developed in this chapter, with the intent of verifying the applicability and effects of these technique when applied to flow problems that are usually simulated using the SPH method. These results may be useful for the SPH community for future developments.

Within the test cases considered in this chapter, the refinement method has proven to be robust for both slow and violent flows. By applying such a technique, it is possible to locally improve the results given by an unrefined particle distribution as it was shown on the 2D dam-break test case. Recently, the use of the δ -SPH method has become more common among the research community and, for that reason, we verified the applicability of Lopez *et al.* [8] technique to such a method. Results are promising but show that improvements are necessary before successfully applying these techniques to δ -SPH.

We can also state that by considering the improvements proposed in [8], better results are achieved with respect to the original technique [7, 6]. From now on, in this thesis, we will focus on the use of the particle refinement technique as presented by Lopez *et al.* [8]. In the next chapter, the dynamic refinement technique depicted here will be improved by coupling it to a new particle derefinement technique. By doing that, we intend to improve the efficiency of particle distribution adaptivity within SPH.

Chapter 11

Particle Derefinement

The efficiency of the dynamic refinement algorithms could be enhanced by coupling them with a derefinement technique. That is, besides the capability of splitting bigger particles into smaller ones, it should be possible to erase them or clump them whenever they are no longer needed. One could point out the remeshing technique [121] as a solution to perform such derefinement, where every interval of time we would run a check procedure on the fluid domain to search for particles that are out of the ‘fine’ domain and remesh them to an unrefined Δx . The problem is that we can, following De Leffe’s thesis [19], predict some problem when dealing with free-surface flows like losing the free-surface condition and also having a more viscous flow.

Recently, a coalescing technique was proposed by Vacondio *et al.* [122]. Smaller particles, by pairs, are coalesced into one larger particle by conserving linear momentum and mass (minimizing the error on density during the process, similarly to the splitting technique by Feldman [7] and Feldman *et al.* [6]). The drawback of this procedure is that it cannot be performed at every time step as it implies high computational costs as pointed out in [122]. We could add the fact that the particle coalescing is less effective than the refinement procedure, i.e., coalescing is performed for each pair of particles while for the refinement procedure one particle can be transformed into three, four or more particles. Consequently, the rate of particle creation is not the same as the coalescing one. In order to be able to return to the initial particle distribution, several coalescing procedures must be performed at each time step. Moreover, in order to apply such a procedure, it is necessary to have a stabilized flow as the particles that will be merged need to have similar properties (velocity, density, pressure, etc), otherwise errors are introduced. That said, very interesting results were obtained for a series of viscous flow phenomena with such a coalescing scheme. This method was extended to shallow water problems in [123].

In this thesis, we propose a new particle derefinement technique that is coupled to the previously presented particle refinement technique. This new technique raises from the assumption that a *mother* particle when split can be kept on the simulation instead of being erased and can be advanced in time with the flow in a passive way. That way, it is possible to switch back to the unrefined particle distribution whenever needed. The advantage of this new technique is its independence of the flow characteristics and the fact that the rate of derefinement is naturally the same as the refinement one. All the details of this new method and a validation process are explained on the following sections.

11.1 Theory

The derefinement procedure developed in this thesis allows to erase the particles that were created on the regions of interest once they left this region and switch from a refined distribution to an unrefined one. To better explain the technique, we start from the principle that there are two subsets of particles: *ON* particles and *OFF* particles. In order to distinguish the two sets of particles a new variable is introduced:

- $\gamma = 1 \rightarrow$ particle is *ON*,
- $\gamma = 0 \rightarrow$ particle is *OFF*.

This γ variable works as an weighting function for each particle and is introduced in the SPH operators as follows:

$$f_i = \sum_j f_j W_{ij} w_j \gamma_j, \quad (11.1)$$

$$\nabla f_i = \sum_j f_j \nabla W_{ij} w_j \gamma_j, \quad (11.2)$$

in which the sum takes into account all the particles, including the *mother* and the *daughter* ones. As a direct consequence of inserting γ in the SPH operators, only the *ON* particles are considered during the computation of the SPH operators. Note that the *OFF* particles time derivatives are still computed using the *ON* particles.

The refinement techniques available in the literature usually erase the *mother* particles during the refinement procedures (e.g. [7, 6, 8]). In our new approach, the *mother* particles are not erased but turned *OFF*, i.e. these particles are no longer considered to compute the SPH operators, but still exist. Concerning the *daughter* particles, they are turned *ON* as soon as created. Consequently, in such a scheme, a turned *OFF mother* particle still exists and follows the flow dynamics since its SPH time derivatives are computed using interpolations based on all *daughter* particles that are *ON* and inside its kernel support. As a result, this *mother* particle can be then turned *ON* when leaving the refinement area. Note that on the contrary, *daughter* particles are simply erased when leaving the refinement zone.

Besides allowing spatial and temporal particle refinement adaptivity, this procedure offers the possibility of having several levels of refinements where a *daughter* particle may have its own *daughter* particles and be turned *OFF* and *ON* again in a very straightforward and simple manner (an example of application of this technique to multiple particle refinement zones is shown in section 11.2.3).

However, some limitations and problems related to this new procedure should be highlighted. Consider a stable and uniform flow that has been subjected to a refinement procedure where both *ON* and *OFF* particles coexist. When a derefinement occurs, a situation as depicted in figure 11.1 may happen. In this figure, the dark horizontal line corresponds to a boundary of the refinement domain prescribed, and the red and blue particles correspond to the *mother* and *daughter* particles respectively. This figure shows a situation where a *mother* particle is turned *ON* while two of its *daughter* particles are erased (crossed particles) and two others remain active (still turned *ON*). This

situation should be absolutely avoided since it creates an instantaneous increase of the total mass of the system to solve.

Though small and very local, this excess of mass is responsible for an additional error in the global solution which may lead to instabilities. However, the main difficulty in treating this problem resides in developing a simple and robust technique that can be applied to any flow phenomenon. The derefinement technique presented here relies on the assumption that a *OFF* particle will have its time derivatives computed using the particles that are *ON*, and because of that, a situation may appear where a *mother* particle will follow a different Lagrangian path than the daughters that it originated. This situation is very unlikely to happen in a flow using the technique presented in this thesis because the *mother* particle is ‘numerically’ attached (through the kernel interpolation) to the *daughter* particles and for that reason, the *mother* and *daughter* particles follow similar pathes and tend to move to the same regions (behavior that is more present in a simulation when considering similar radius lengths for *mother* and *daughter* particles).

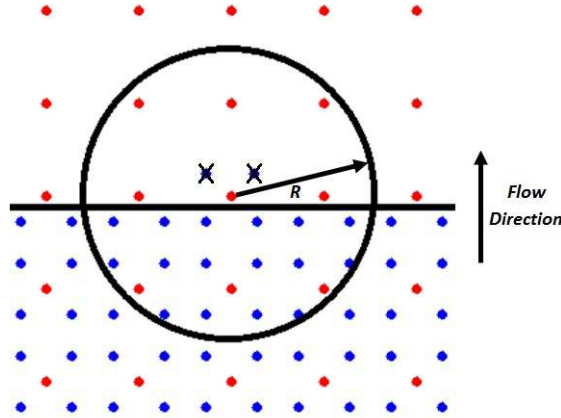


Figure 11.1: Illustration of some problems related to the particle derefinement procedure: horizontal line shows the limit of the refinement domain and the red and blue particles represent *mother* and *daughter* particles respectively.

We may also highlight the fact that some pressure discontinuities may occur when a *mother* particle enters (or leaves) a refined domain. This is caused by the instantaneous increase (or decrease) of the number of neighbor particles per particle when passing from one region to another (which is also shown in figure 11.1). This “refinement shock” causes an instantaneous variation leading to spurious pressure peaks.

In order to deal with these problems, a transition region is defined between the unrefined and refined zones. In this transition zone, the *mother* and *daughter* particles are smoothly turned *OFF* and *ON* respectively. For both *mother* and *daughter* particles $0 < \gamma < 1$ in this zone and γ increases for *mother* particles and decreases for *daughter* particles (or vice-versa depending on the transition region). Figure 11.2 shows the transition zone marked with light grey, contrasting with the region with constant γ marked with dark grey. While the flow gets through this refinement zone, the interpolation is smoothly transferred from the *mother* to the *daughter* particles (or vice-versa) and the excess of

mass previously mentioned is smoothed out.

With this approach any particle in the fluid domain respects

$$\sum_j W_{ij} \gamma_j w_j = 1, \quad (11.3)$$

where j refers to the particles present within the compact support of the kernel function, which in turn ensures the continuity of equations (11.1) and (11.2) throughout this region. In order to complete the description of this new derefinement technique, the value of γ needs to be defined. *Mother* and *daughter* particles respect

$$\gamma_i = \frac{dB_i}{dB} \quad (11.4)$$

and

$$\gamma_i = 1 - \frac{dB_i}{dB} \quad (11.5)$$

respectively, where dB is the size of the transition region (taken as a measure of R in this paper, namely $dB = 2R$) and dB_i is the distance of the considered particle to the region with constant γ . Note that a special treatment must be performed in the corner regions of the refinement domain by using a rounded shape for γ as illustrated in figure 11.2. For the *mother* particles we have

$$\gamma_i = \sqrt{\gamma_{i_x}^2 + \gamma_{i_y}^2}, \quad (11.6)$$

and for the *daughter* particles

$$\gamma_i = 1 - \sqrt{\gamma_{i_x}^2 + \gamma_{i_y}^2}, \quad (11.7)$$

where γ_{i_x} and γ_{i_y} are computed using equation (11.4) with respect to the x and y directions.

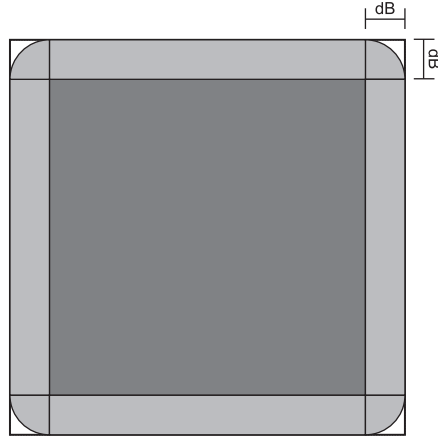


Figure 11.2: Constant γ (dark grey) and the transition (light grey) regions on a 2D squared refinement zone.

The extension to 3D problems is easily obtained by extruding the figure 11.2. The result is a cube with a central constant γ region and six regions similar to the 2D refinement region in figure 11.2. The

darker central part of this region follow the law of evolution given by equations (11.4) and (11.5), the rectangular not-rounded regions in lighter grey follow the treatment given by equations (11.6) (11.7) and finally the rounded regions on the corner which in 3D form $1/8^{th}$ of a sphere follow a special 3D treatment where γ is defined for the *mother* particles as

$$\gamma_i = \sqrt{\gamma_{i_x}^2 + \gamma_{i_y}^2 + \gamma_{i_z}^2}, \quad (11.8)$$

and for the *daughter* particles

$$\gamma_i = 1 - \sqrt{\gamma_{i_x}^2 + \gamma_{i_y}^2 + \gamma_{i_z}^2}, \quad (11.9)$$

Note that here we used a linear function for the value of γ but other functions could be used and tested. Also, we opted for this transition (or buffer) zone, however, other techniques could be used as the one in Bouscasse *et al.* [117]. Also, the simulations presented in the previous chapter did not consider the use of this smoothing zone and there were no major pressure discontinuities. These discontinuities are more present for slow motion simulations and some results with and without using this smoothing zone are available in section 11.2.2.

Moreover, the derefinement technique presented in this section (especially when coupled with the smoothing zone), does not suffer from the problem of having discontinuous physical fields when applying the refinement process. As the *OFF mother* particles time derivatives are computed using the *daughter* particles, when the latter are erased and the first are turned *ON* the continuity of the physical fields is automatic.

11.2 Validation results

In order to validate the particle derefinement technique, we have selected some classical test cases that are often used to assess the capabilities of new developments within the SPH method. We will follow an increasing level of difficulty for the validation process. The first test case is the classical 2D dam-break that has already been used twice in this thesis, from which some conclusions were depicted in the previous chapter where only the splitting technique was applied. Secondly, the symmetric impinging jet on a flat plate is analyzed, the intention being to check the applicability of the derefinement process for strong impact phenomena.

11.2.1 2D dam-break flow against a wall

Here, the new particle derefinement technique is validated using the 2D dam-break already used in this thesis, cf. 7.4.2. In the right side of the domain, close to the pressure sensors, a refinement zone is defined with the same height as the domain and with $0.4m$ width (distance to be measured from the right wall), similar to what was done in the previous chapter with the difference that, in this zone, Lopez *et al.* [8] refinement procedure is now coupled with the new derefinement technique. This way, the number of particles needed to get converged results is lower than for a fully refined particle distribution simulation.

The goal is to show that such a refinement/derefinement procedure allows increasing the quality of results and getting similar results to those given by a fully-refined particle distribution. For that purpose, three simulations are considered:

1. Coarse simulation (1): constant $\Delta x = 0.01m$ inter-particle space;
2. Fine simulation (2): constant $\Delta x = 0.005m$ inter-particle space;
3. Dynamic refinement/derefinement simulation (3): a refinement zone is considered using $\alpha_{PR} = 0.7$ and $\epsilon_{PR} = 0.55$ for the refinement/derefinement procedure (with the transition zone).

In this test case, the first point that is checked is the pressure field close to the interface between the refined and derefined zones, as seen in figure 11.3 for different time instants (in dimensionless notation). It can be observed that the pressure is smooth in such a zone, hence we conclude that the refinement/derefinement technique coupled with the transition zone does not introduce any discontinuity in the flow. These results are very similar to the ones presented in figure 10.14, obtained by applying only the particle splitting technique, especially for $t > 1.6s$.

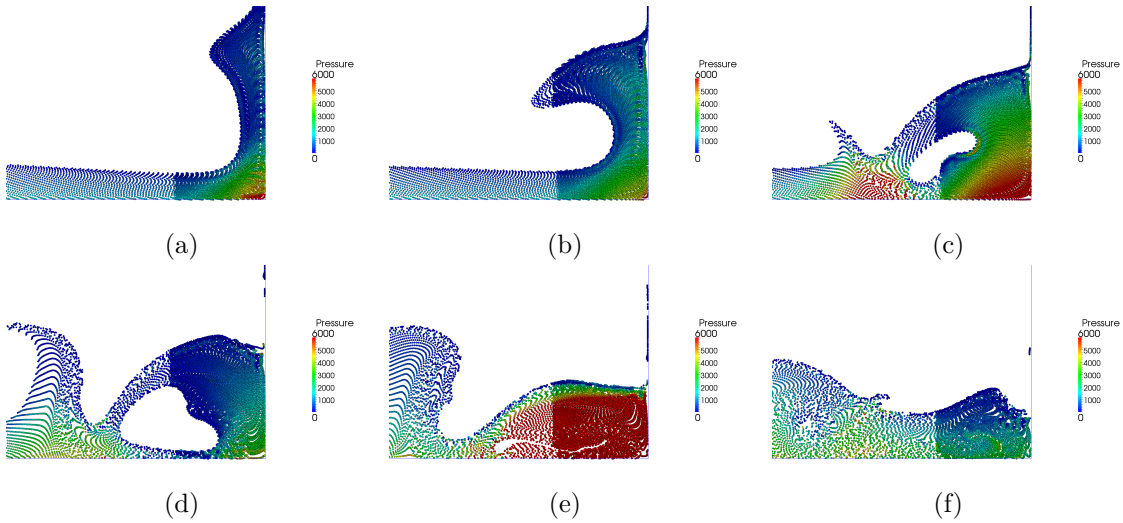


Figure 11.3: Wave breaking phenomena: (a) $T^* = 4.8$, (b) $T^* = 5.7$, (c) $T^* = 6.5$, (d) $T^* = 7.3$, (e) $T^* = 8.1$ and (f) $T^* = 8.9$.

Figure 11.4 compares the pressure signal at $P1$ sensor (located $0.160m$ from the bottom of the domain) for all the simulations (1 to 3) against the experimental results found in [92]. The flow impact against the wall (the first pressure peak) and the wave-braking phenomena happening at $T^* = 6$ (dimensionless) are well predicted by the dynamic simulation as well as by the other simulations, being the best result given by the simulation (2) (the most refined). Globally, the dynamic simulation result is better than the one given by the unrefined particle distribution and has the same trends as the fully refined simulation. Note that the fact of dynamically refining and derefining the particles does not introduce noise in the simulation and does not result in an unstable simulation.

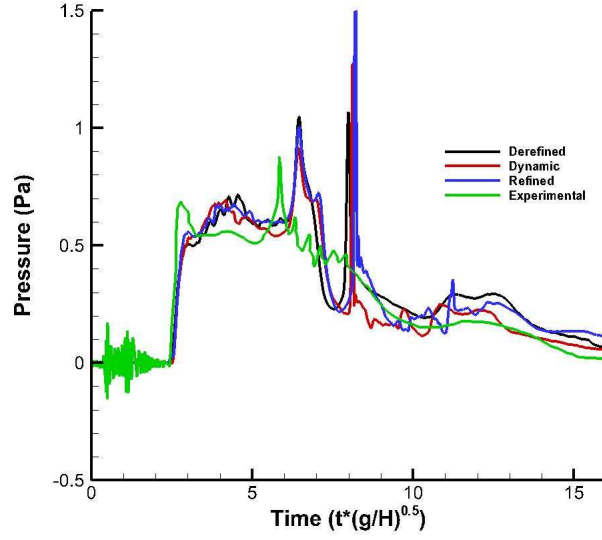


Figure 11.4: Pressure signal on $P1$ sensor: comparison between the three simulations and experimental results.

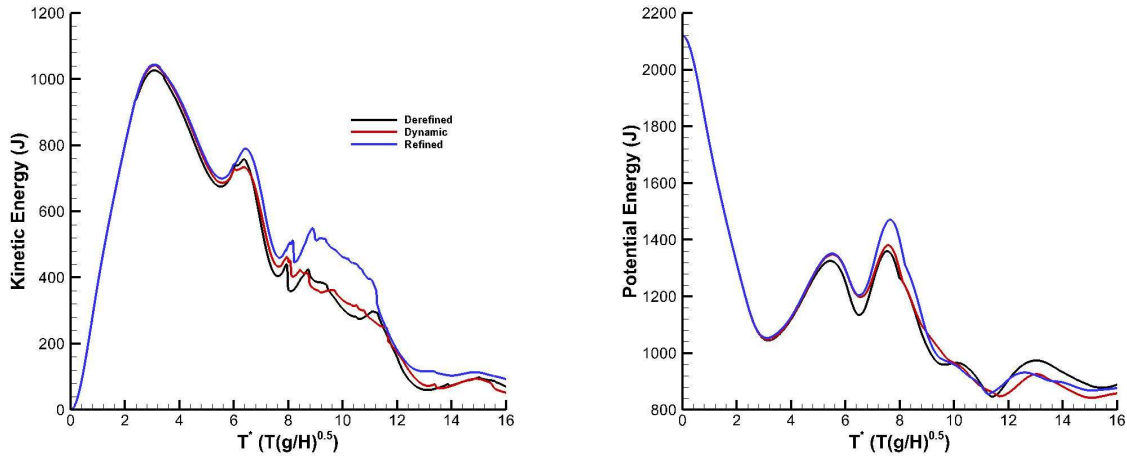


Figure 11.5: Kinetic and potential energy: comparison between the three simulations.

It is important to check whether the same trends observed for the pressure profile are found for the kinetic and potential energy behavior. Figure 11.5 shows that the refinement/derefinement procedure does not create any temporal discontinuity in the energy, as expected (cf. table 10.1). For the simulation 3 (dynamic), most particles are inside the unrefined zone, and because of that, its energy curves get closer to the unrefined results. At $T^* = 2.8$, the particles enter the refinement zone. At this instant, the energy evolution predicted by the dynamic simulations tend to the refined simulation results. When the particles leave this zone, at $T^* = 6.5$, the results diverge.

Globally, the previous results are in a good agreement with the results obtained when only the splitting technique was considered. This leads us to state that the derefinement process does not affect the quality of the results. We remind that, in order to improve the quality of the results given by the dynamic simulation, the α_{PR} factor must be increased but that would imply higher CPU costs. A good trade-off between accuracy and CPU costs is found by keeping $\alpha_{PR} = 0.7$. For the time being of this thesis we could not perform CPU time comparisons between the different simulation because the particle refinement/derefinement algorithm was not coded in the most efficient manner.

11.2.2 Asymmetric mass distribution body

In this second example we consider a modified version (deeper tank) of the test case in section 7.5.1 where a body with an asymmetric mass distribution sinks in a viscous fluid. The sketch of the problem is found on figure 11.6 where the arrow points out the gravity action on the body's center of mass (located $0.25m$ of the geometrical center of the body). As stated before, the body is initially in an unstable position and it will sink and move in a way to reach its stable position. The tank's dimensions are $H_t = 10.0m$ and $L_t = 4.0m$ and the body's dimensions are $H_b = 0.5m$ and $L_b = 1.0m$. The fluid density is $\rho_f = 1.0kg/m^3$ and the gravity is $g_0 = 1.0m/s^2$. Here, a viscosity of $\mu = 1/500 Pa \cdot s$ was used.

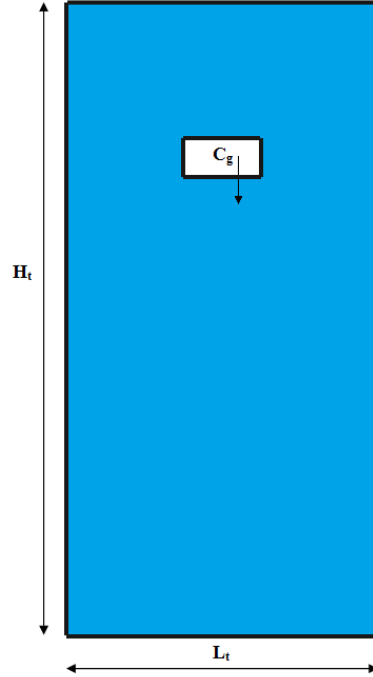


Figure 11.6: Sketch of the simulation with an asymmetric mass distribution body

For this test case the refinement domain follows the body movement, i.e., the domain's velocity is equal to the body's. The body's roll angle and center of gravity position histories are analyzed. Initially, five simulations were performed:

1. *N50* : constant $\frac{L_b}{\Delta x} = 50$ particle distribution;
2. *N100*: constant $\frac{L_b}{\Delta x} = 100$ particle distribution;
3. *N200*: constant $\frac{L_b}{\Delta x} = 200$ particle distribution;
4. *N50 to N100*: a refinement zone is considered surrounding the body and the value of $\alpha_{PR} = 0.7$ and $\epsilon_{PR} = 0.55$ are used for the refinement procedure. Unrefined particles have $\frac{L_b}{\Delta x} = 50$ and refined ones $\frac{L_b}{\Delta x} = 100$.
5. *N100 to N200*: a refinement zone is considered surrounding the body and the value of $\alpha_{PR} = 0.7$ and $\epsilon_{PR} = 0.55$ are used for the refinement procedure. Unrefined particles have $\frac{L_b}{\Delta x} = 100$ and refined ones $\frac{L_b}{\Delta x} = 200$.

Performing the total of 3 simulations with distinct particle distributions allows to define a convergence pattern that is used to better assess the results reached by the dynamic simulations. Also, the two different dynamic simulations may allow us to verify if the particle refinement/derefinement is sensitive to the initial particle distribution. Note that, for the dynamic simulations we consider a refinement domain surrounding the body's center of mass measuring $1.8m$ on both directions (x and y).

The body motion is very sensible to the fluid viscosity as it acts as a damping factor, i.e., changing the viscosity of the flow impacts directly the body's motion. The results given by each of the above simulations will be compared here and, by observing the body center of gravity position and roll angle time history, it is possible to determine whether the dynamic refinement/derefinement technique interferes or not on viscous flow simulations.

As stated before, the pressure discontinuity is more important for slow flows, as the one considered in this test case. Hence, the effect of using a transition zone (presented in section 11) is verified (cf. figure 11.7). Pressure peaks appear on the limit between the unrefined and refined zones when the transition zone is not used, as seen in figure 11.7-(b) (simulation *N100 to N200* without transition zone). These perturbations are much less important when the zone is used, as seen in figure 11.7-(c) (simulation *N100 to N200* with transition zone) resulting in a pressure distribution close to what is observed with a constant particle distribution (figure 11.7-(a) for the simulation *N200*). These perturbations are not completely canceled because the discontinuity between the two zones is always present. With this transition zone, such a discontinuity is however much smoothed out.

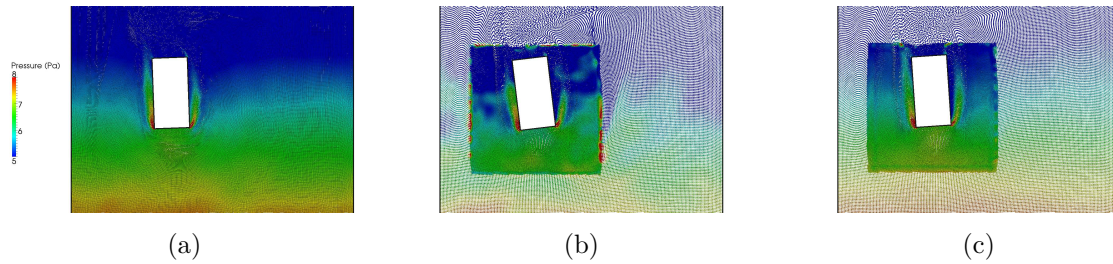


Figure 11.7: Pressure distribution for three simulations: (a) *N200* simulation, (b) *N100 to N200* simulations without transition zone and (c) with transition zone

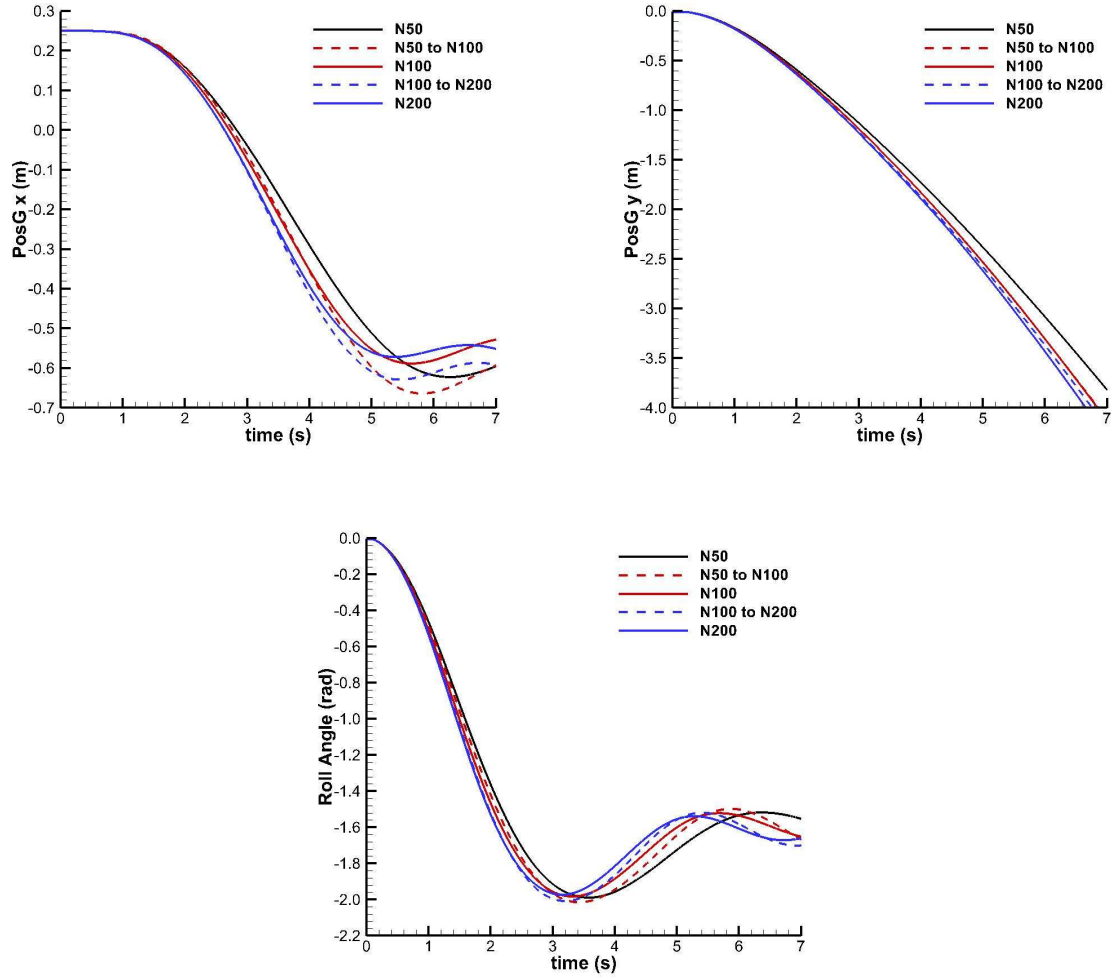


Figure 11.8: Body motion history for all the simulations considered.

The previous test cases were concentrated on the flow characteristics such as pressure field and energy evolution. Here, this validation procedure is extended to the prediction of the body dynamics. Figure 11.8 shows the roll angle and the position of the center of gravity histories for the whole simulation (from $t = 0.0s$ up to $t = 7.0s$). The interest of this test case can be seen by the discrepancy of results between the fully unrefined ($N50$) and the fully refined ($N200$) simulations. Refining the domain rapidly changes the shape of the results even though convergence is observed since $N200$ results are much closer to $N100$ than the latter to $N50$.

Using a dynamic refinement for this kind of problem can be very interesting because it allows to have a coarse particle distribution on the domain and concentrate the particles around the body, where the physical effects of viscosity are more important. The results given by using this approach have the same trends (with some minor discrepancies) as the results given by a fully refined simulation, as verified in figure 11.8, for the roll angle history where it is possible to see that the dynamic refine-

ment/derefinement results tend to the ones obtained with more particles. The same conclusion may be drawn by investigating the motion history of the center of gravity in the x-direction, even though the amplitude for the dynamic simulation results are higher. Note that for the history of the y-direction position of the center of gravity, simulations *N50 to N100* and *N100* give the same result.

Being the flow viscous, a vortex detachment phenomenon happens during the sinking. This vortex formation affects the body's behavior and the more precisely they are modeled, the more accurate will be the solution. As this phenomenon is predominant in the y-direction (the sinking direction), we decided to perform two new dynamic simulations with a bigger domain in this direction ($3.2m$ instead of $1.8m$). Figure 11.9 shows the roll angles obtained for such simulation in addition to the previous ones, and it is possible to verify that, by using a bigger refinement domain, the solution converges more rapidly to the constant particle distribution result (*N200*).

Another important point to be checked is the mass conservation. As stated before, there may be small fluctuations caused by the refinement/derefinement process. Figure 11.10 shows the mass variation with respect to the initial total mass of the system (m_0) for the first dynamic simulation (*N50 to N100* and *N100 to N200*). The fluctuations remain in an acceptable range of less than 0.30% for *N50 to N100* simulation and below 0.02% for the *N100 to N200* (which shows convergence), and for that reason, may not affect the physics of the flow.

Globally, all these results show that the new derefinement procedure can be easily adapted to the particularity of the flow that is simulated, and that it is possible to increase the quality of the results only by a locally refining the particles. This same conclusion was already traced in the framework of 'variable-h' resolution, however, here we extend it to the framework of dynamic refinement with a moving body. By extension, we can conclude that the new particle refinement/derefinement procedure presented in this thesis can be successfully applied to viscous flows around bluff bodies.

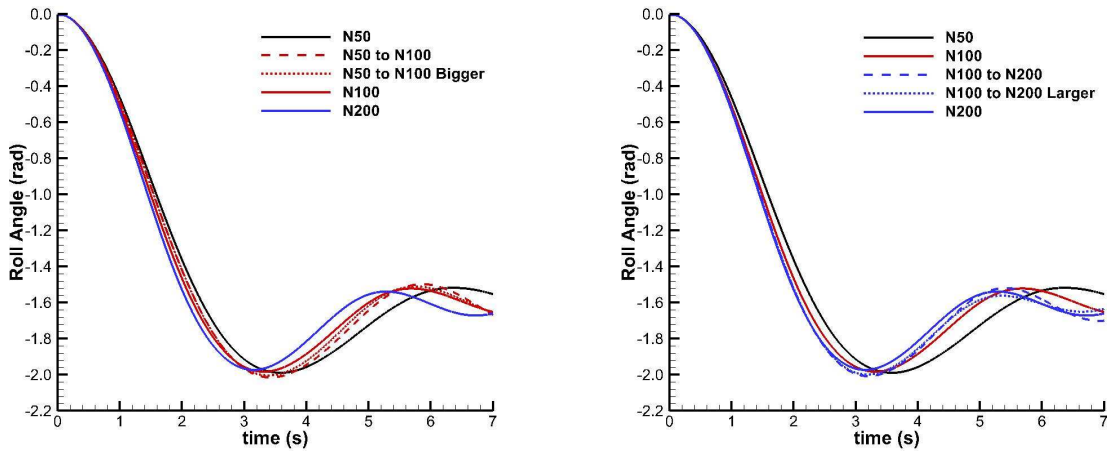


Figure 11.9: Convergence of the dynamic solution with respect to the size of the refinement domain.

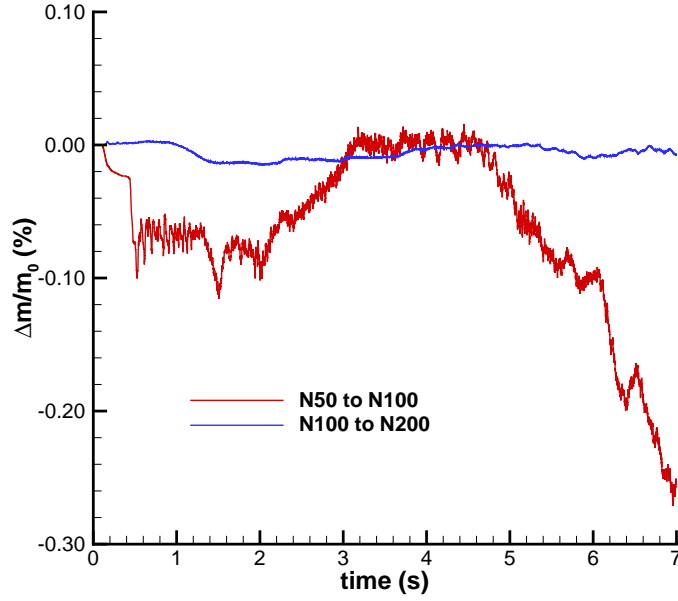


Figure 11.10: Total mass variation throughout time for the body asymmetric mass distribution test case.

11.2.3 Impinging jet on a flat plate

This section targets the capability of using multiple refinement zones within the same simulation. The test case retained here is the simulation of a jet impacting a flat plate, which is often used in the literature for validating the SPH [37] and MPS [124] methods. The goal here is to reach 4 refinement levels, as shown in figure 11.11, with an inter-particle distance in the range of $\Delta x_{max} = 0.04m$ to $\Delta x_{min} = 0.005m$. In table 11.1, the different refinement regions are presented. At each level the inter-particle distance is divided by two, and one particle (*mother*) originates four *daughters*.

Table 11.1: Jet flow simulation set-up.

Level	Δx	$X_{min}:X_{max}$	$Y_{min}:Y_{max}$
0	0.04m	Domain limits	Domain limits
1	0.02m	-3.0:3.0m	0.0:2.8m
2	0.01m	-2.0:2.0m	0.0:2.0m
3	0.005m	-1.0:1.0m	0.0:1.0m

As the flow considered here is very violent, in order to be able to accurately predict the pressure at the impact point ($r = (0, 0)$) a fine particle distribution must be used. By using several refinement levels, it is possible to start with a very small number of particles and to reach the desired refinement using less particles, and consequently, less memory.

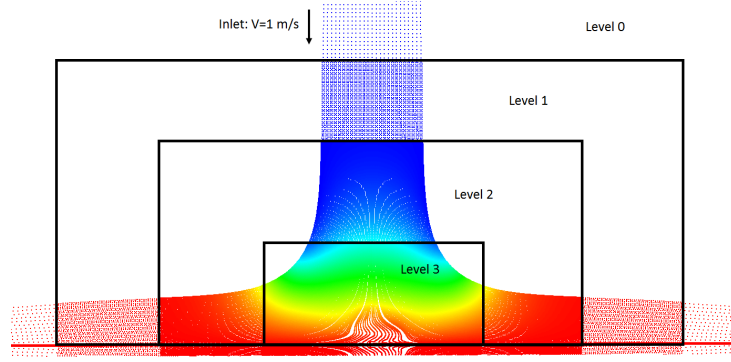


Figure 11.11: Sketch of the simulation of the impact of a jet on a flat plate with several refinement zones.

Several simulations are performed here. Two simulations involve the refinement/derefinement technique where the α_{PR} parameter is changed ($\alpha_{PR} = 0.8, 0.9$), keeping $\epsilon_{PR} = 0.5$ constant using the Lopez *et al.* refinement pattern. Being this flow phenomena very violent, in order to ensure stable simulations the values for α_{PR} had to be higher than those used in the previous simulations. Two other simulations are performed with a constant $\Delta x = 0.005m$ but with varying $R/\Delta x$, matching to the final ratio achieved in the dynamic simulations at the fine zone. These simulations form a basis permitting a better analysis of the results given by the dynamically refined simulations.

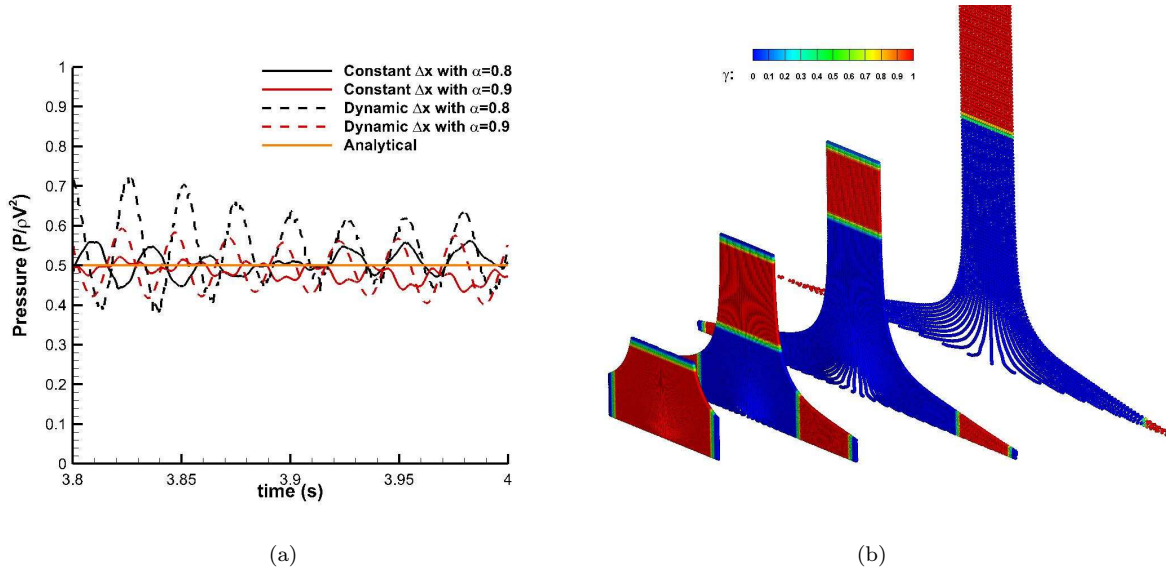


Figure 11.12: (a) Pressure at the stagnation point and (b) four levels of refinement are shown with the value for the γ variable at $t = 3.0s$

Figure 11.12-(a) shows the pressure at the stagnation point predicted by all the simulations listed above. Focus is given to the last instants of the simulations, which allows a clearer view of the different curves. As for the previous case, the dynamic simulations have the same convergence behavior as for

the simulations with constant Δx , i.e., the amplitude of the oscillations decrease and results tend to the analytical solution as the number of neighboring particles increase. These oscillations are also observed in [37, 124]. In figure 11.12-(b), particles with different Δx are shown separately at $t = 3.0s$, where the colors represent the γ variable values for the simulation set with $(\alpha_{PR}, \epsilon_{PR}) = (0.8, 0.5)$. The red zones are the fully active zones and the blue zones represent the particles that are fully *OFF*. It is possible to observe in the same figure how one zone overlaps with the other in the transition zones.

Figure 11.13 shows the pressure contours at $t = 3.0s$, where the three last refinement zones are shown. The new derefinement procedure presented and developed in this article allows predicting a smooth pressure distribution close to the wall. The smooth transition between two distinct refinement zones is only assured by using the transition (buffer) zone presented earlier. If one desires to obtain the same quality of results but without such a transition zone, it is necessary to use $\alpha_{PR} = 1.0$ but that would imply higher CPU costs. In the same figure one can observe the in-line formation of particles in the region close to the stagnation point.

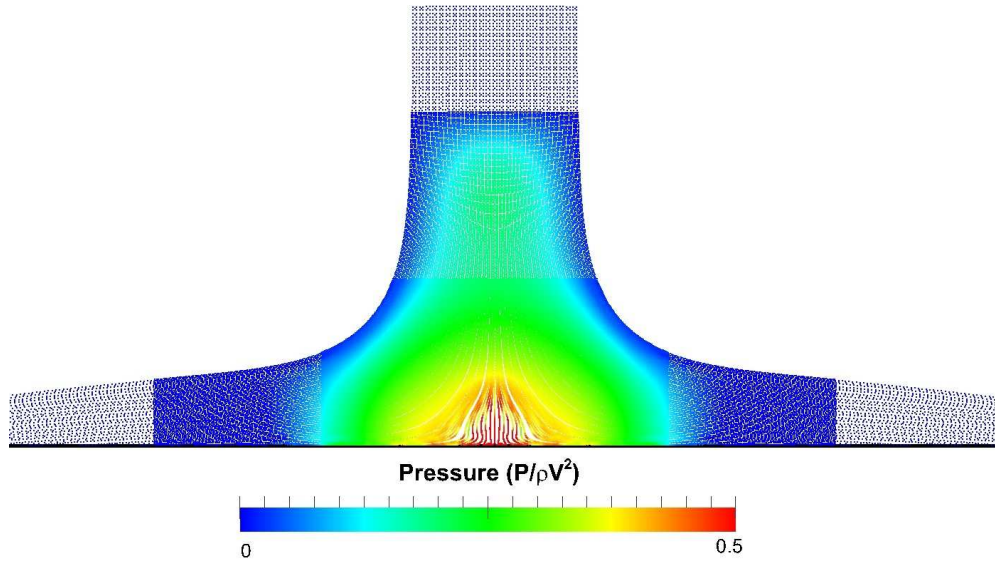


Figure 11.13: Smooth pressure transition between the different refined zones.

We conclude that the refinement/derefinement technique presented and developed in this thesis can be applied to simulations using several levels of refinement without affecting the smoothness of the pressure profile and the stability of the simulations.

Chapter 12

Towards Naval and Offshore Problems

We proceeded to the application of such a particle refinement and derefinement technique closer to real world problems. We intend to show that we can successfully apply these new techniques to simulate problems that were already treated previously in the laboratory with very large CPU times, problems where the interest of using a local refinement procedure may be useful.

Firstly, the case of a flooding of a Roll-on Roll-off ship section will be treated and then the test case of a wave propagation coupled with a wave breaking phenomenon will be quickly/simply analyzed.

12.1 Flooding of a Roll-on Roll-off ship section

Here we consider a flooding simulation of the mid-section of a Ro-Ro ship, which has already been simulated using the SPH method by Marsh *et al.* [111]. The results were in good agreement with the experimental data, experience that was performed in the LHEEA laboratory in the framework of Khaddaj-Mallat's Ph.D. thesis [9] and other works [125, 126]. Figure 12.1 shows the experiment set-up. The internal configuration of the mid-section consists of five compartments: the double bottom forward (DBF) and double bottom aft (DBA), above which we find the generator room (GR) and the engine room (ER). In the experiments the ship engines and generators are not considered. Figure 12.1 shows also a storage room (SR) located above the GR. For more details on the experimental set-up and model, refer to [111, 9, 125, 126].

In the experiments, a rectangular section of the ship hull is removed on one side of the model, to represent an external damage caused by an eventual impact. Internally, a symmetrical V-shape is cut from the decks at the position of impact, which reproduces a damage caused by the collision of another vessel's bow. A water height sensor is placed inside the tank (marked by the blue circle in figure 12.1) and these experimental results are compared to the numerical ones. A door is placed in front of the damaged area. When the simulation is launched, the door is subjected to a vertical opening velocity of $0.467m/s$ which remains constant for the entire simulation.

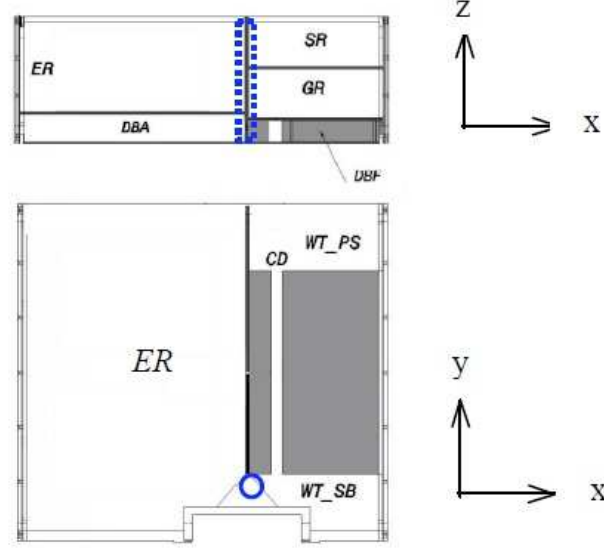


Figure 12.1: Sketch of the Ro-Ro ship flooding simulation

The major difficulty of the SPH simulation performed in [111] concerned the particle distribution. In order to correctly predict the flooding process, a very fine particle distribution (a small Δx) must be used. At the same time, the fluid domain surrounding the ship must be big enough to avoid confinement effects. Note that the Ro-Ro model length, width, height and draught are equal to $L_{Ro-Ro} = 0.698m$, $B_{Ro-Ro} = 0.653m$, $D_{Ro-Ro} = 0.250m$ and $T_{Ro-Ro} = 0.140m$ respectively and that the tank dimensions are $1.75m$ wide, $2.0m$ long and $0.28m$ deep. Using a constant particle distribution on the entire domain would be too costly; consequently, Marsh *et al.* [111] used a ‘variable-h’ resolution. However, a zone with constant Δx surrounding the ship section must be used in order to ensure that all the particles that will go inside the section have the same and a fine resolution. This implied, even within a ‘variable-h’ context, the use of many particles. In [111] the total of 1,020,107 ($\Delta x = 0.01m$) particles were used and the simulation lasted 21 days for the total of 3 seconds of physical time (running on 8 processors).

In this thesis we will reproduce this experiment but by imposing a dynamic refinement. As soon as particles approach the door, they are split achieving the target refinement of $\Delta x = 0.005m$. That way we double the local particle refinement without increasing the total number of particles in the domain. With such an approach, we intend to increase the quality of SPH results with respect to the ones presented in [111]. By applying the dynamic refinement to this problem, we avoid having to use 4.8 million particles, which would be the number of particles needed by a ‘variable-h’ mesh having as target $\Delta x = 0.005m$. Note that for the simulations using the dynamic refinement the values $\alpha_{PR} = 0.8$ and $\epsilon_{PR} = 0.55$ were used for the splitting controls using the 3D extension of the Lopez *et al.* [8] refinement pattern. By the end of this dynamic simulation there was an increase of 199,744 particles in the domain which resulted in 50% extra CPU time.

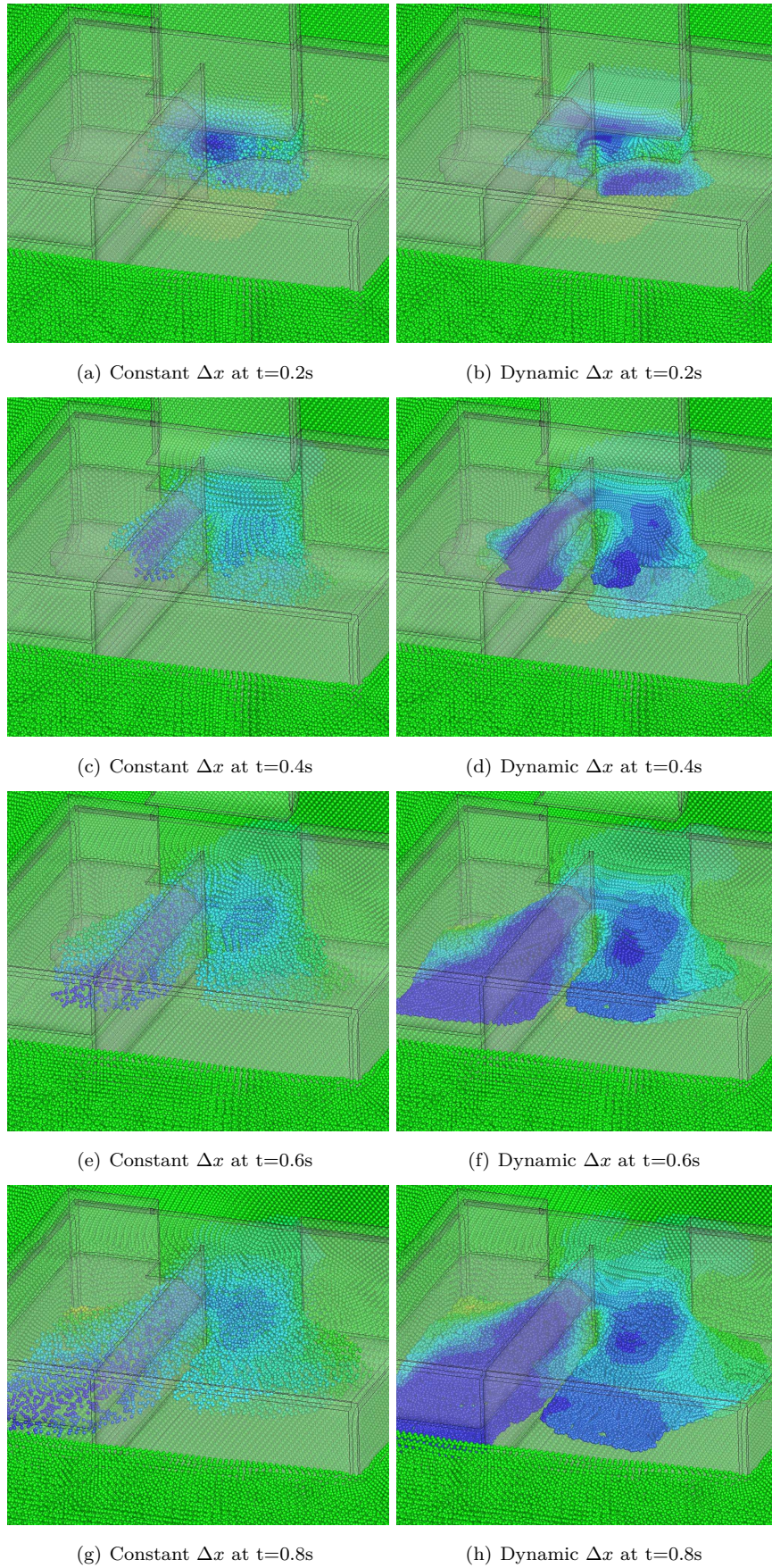
Figure 12.2: Results' comparison between the constant Δx and dynamic Δx simulations.

Figure 12.2 shows the results obtained by both simulations. By applying the dynamic refinement process to this simulation a finer particle distribution is reached and a better description of the free-surface shape is predicted. Note that the wave fronts are much better defined by using more particles. For the unrefined particle distribution simulation, the particles tend to a dispersed formation at the initial instants, forming some jet-like formations, which is not the case for the refined simulation. From $t \geq 0.6s$, it becomes clear that the dynamic simulation is in advance with respect to the constant particle distribution simulation, possibly a result of convergence. With a finer distribution, particle spread more easily inside the compartments than with a coarse Δx . Looking at the results for the wave height sensor place right in front of the flooding entrance presented in figure 12.3, we see that SPH is capable to provide acceptable predictions of the floodwater time of arrival, along with the initial steep rate of increase of the water height. However, as the rate of flooding diminishes, SPH predicts a water height of an average of $30mm$ which corresponds to 3 and 6 particles for the particle refinements of $\Delta x = (0.01, 0.005)m$ respectively. Considering the final water height, it is possible to verify a certain convergence towards the experimental results when the dynamic refinement procedure is used which shows the interest of using a dynamic particle refinement technique. It is expected that better numerical results can be achieved by increasing even more the number of particles used for the SPH simulations.

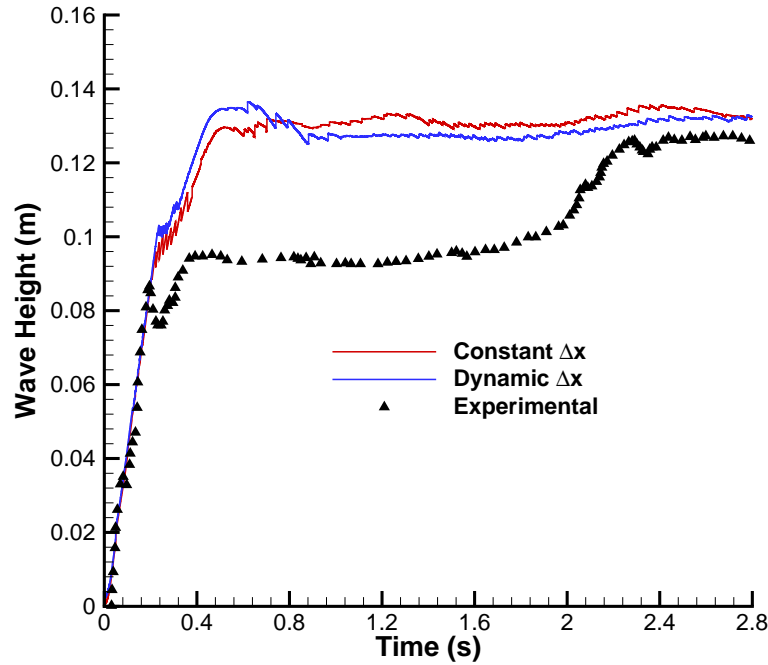


Figure 12.3: Comparison between the experimental results for the water height from [9] and SPH predictions with and without using a dynamic refinement technique.

12.2 Wave Breaking

Here we intend to show the interest of dynamic adaptivity within SPH for oceanic flow problems. For that matter, we consider a simple test case that couples wave propagation to wave breaking phenomena. A small basin is considered, as show in figure 12.4, where a wave is generated on the left side of the domain by a moving flap and propagates towards the right side. At this part of the domain, a step is modeled with a height enough to have the wave break.

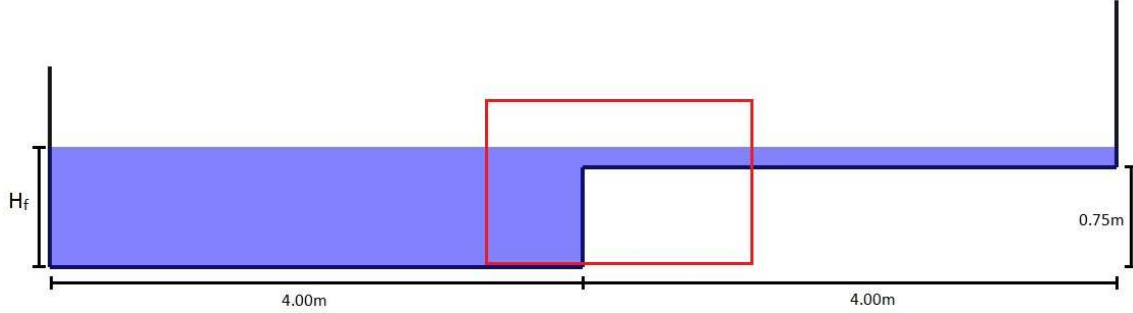


Figure 12.4: Wave propagation and breaking test case sketch.

The generated wave has an amplitude of $0.15m$ and a period of 3.5 seconds. Several simulations were performed where the particle distribution was increased, up to a level where the breaking phenomenon could be captured, i.e., $\Delta x = 0.0125m$. Then, we performed a simulation using as initial particle refinement $\Delta x = 0.025m$, and inside the red squared domain in figure 12.4 particles are refined achieving the distribution that allows to capture the wave breaking phenomenon (using $\alpha_{PR} = 0.7$ and $\epsilon_{PR} = 0.55$). Note that with a constant particle distribution of $\Delta x = 0.025m$, the breaking phenomenon is not captured. This refined domain goes from $x = 3.2m$ to $x = 5.5m$ and vertically it covers all the water depth.

The goal is to verify if by refining locally only, close to where the breaking phenomenon happens, it is possible to correctly predict the breaking wave as for a simulation with constant particle distribution. Figure 12.5 shows the results obtained for the simulations with constant and variable particle distributions. At $t = 5.8s$ the wave is starting to break and at $t = 6.0s$ the wave is already breaking. The dynamic simulation is capable of predicting the beginning of the wave breaking, even though the breaking phenomenon itself happens later for the dynamic simulation, at $t = 6.1s$. Note, however, that the fact of using the new particle refinement and derefinement technique does not interfere much with the dynamics of the wave breaking phenomenon. Finally, it can be observed that the pressure fields obtained by the two simulations are in very good agreement. The fact of using a smoothing transition zone cancelled the discontinuity caused by the abrupt change in the number of neighboring particles between zones.

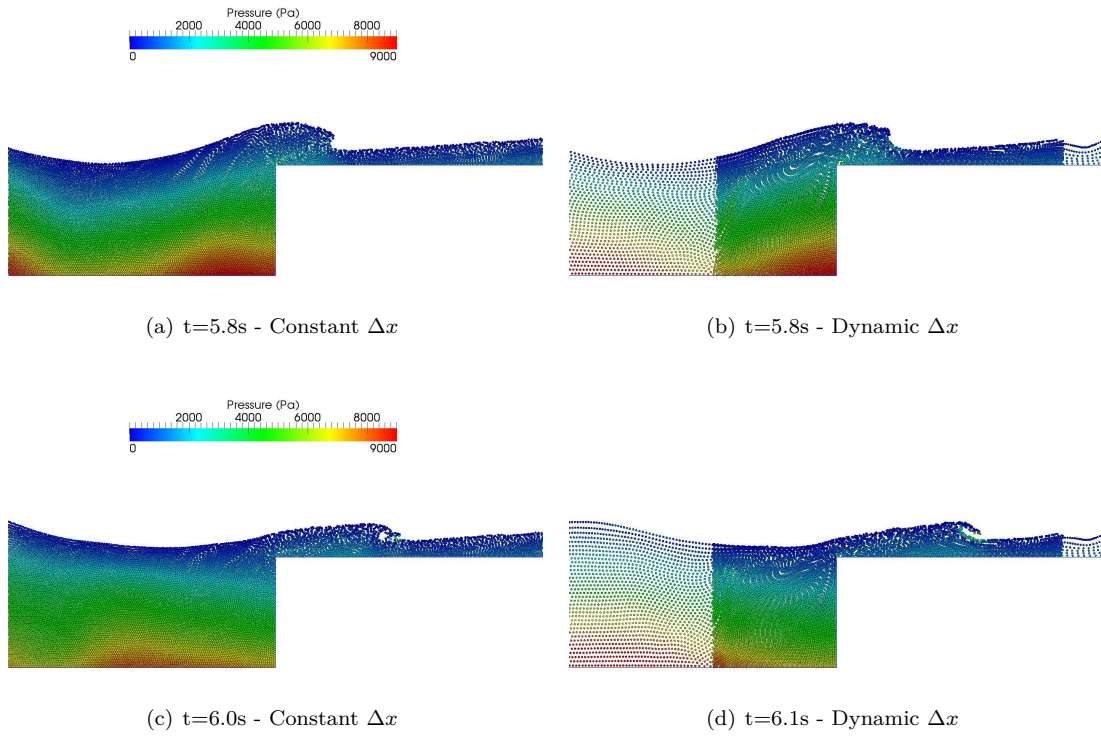


Figure 12.5: Results obtained for the wave generation and breaking simulations with constant and dynamic Δx .

Chapter 13

Discussion

In this part of this thesis, the main particle refinement procedure existing in the SPH literature was coupled with a new derefinement procedure. This new technique allows to dynamically adapt the particle distribution in regions of interest in a straightforward way. Also, in order to avoid undesired pressure peaks or flow perturbations and to ensure stable simulations, a new treatment has been proposed to smooth the transition from the unrefined region to the refined one. This new technique was tested and validated against typical (viscous and inviscid) flows to which the SPH method is applied: dam-break, impinging jet on a flat plate and also the sinking of a floating body. We also verified its application towards real-world Naval and Offshore problems, where the fact of being able to locally and dynamically refine and derefine particles showed to be very interesting and promising.

The efficiency of the Smoothed Particle Hydrodynamics method was improved with this new technique, as it allows to locally and dynamically adapt the particle distribution gaining in precision and without loosing in stability during the refinement/derefinement process. It is possible to use such a method with multiple levels of refinement, which allows to start the simulation with a very coarse particle distribution and reach a much finer one dynamically when and where needed. Moreover, the new refinement/derefinement technique was mainly validated for the Riemann-SPH method. However, it can be easily extended to other variants such as δ -SPH [34]. We remind that, in this thesis, some results using δ -SPH were presented, but for the refinement technique only.

Nonetheless, regarding the adaptivity in SPH, some problems remain yet to be solved. The present particle derefinement procedure can only be successfully applied to simulations where the spatial criterion of refinement is used. If one wishes to be able to derefine (or even refine) particles by using a physical criterion, a different method may be developed, used or improved. Moreover, the SPH is known for its instability issues (including Riemann-SPH and δ -SPH), i.e., even if the pressure field given by the SPH method can be very smooth, its operators are not precise enough to allow a precise computation of some derivatives that are often used to assess if a refinement is necessary (pressure gradient, pressure Hessian, etc). A certain amount of noise would always be present in the evaluation of these derivatives, which could compromise the results and the feasibility of a simulation. Also, the pressure peaks that appear during the simulations using this new technique need to be better studied by continuing the works started in this part.

Moreover the present particle refinement/derefinement technique present some problems regarding

CPU costs. When a refinement takes place the number of neighbor particle per particles increases which is translated in higher CPU costs. This is actually a limitation of the method and it is much more evident when trying to use multiple levels of refinement (having in mind the standard values for α_{PR} and ϵ_{PR} , see section 10.4).

It is not senseless to say that the future of adaptivity in SPH relies on coupling the method to another one that is more accurate than SPH. When more precision is needed, the solver would switch from SPH to another method with better interpolation properties. Somehow, the same conclusions drawn for the new Incompressible SPH developments can be made again here, that is, the fact of dynamically refining and derefining the particles in a way to suit the purposes of the simulation has its limitations (regarding the convergence and precision of operators). A true breakthrough would be to perform a drastic change in the method towards precision, stability and convergence.

Part IV

Hybrid-SPH: Proof of Concept

Chapter 14

Introduction

The Smoothed Particle Hydrodynamics method has shown, for the past years, to be very robust and applicable to a large variety of flow phenomena. As stated in part I of this thesis, SPH has reached a certain maturity and hence relevant improvements are more difficult to be developed. Nevertheless, in this thesis, we have developed two new improvements: one that relies on changing the equations that are modeled and another that improves the manner by which the domain is discretized. Both were proven to be performing and able to achieve the purposes to which they were developed. However, these developments do not change the main drawback of the method which relies on imprecise operators: its poor CPU costs - precision ratio. Because of that, the same issues encountered for the classical weakly-compressible SPH variant were found within the new explicit ISPH method, that is, the incapability of increasing significantly the time-step, the lack of convergence of operators (they are the same as in WCSPH), the well-known tensile instability issues, etc. Moreover, the second development proposed in this thesis, tried only to improve the efficiency of the SPH method by changing the way of discretizing the domain, getting closer to what can be done for Eulerian mesh-based methods, that is, adaptable particle distribution (similar to AMR). Again, improvement was obtained by the effective techniques developed, but involving no drastic change with respect to this main drawback.

SPH is known for performing well for some kind of flow phenomena, violent flows of small time duration for example (some of which were studied in this thesis). However, when SPH is applied to slow flow phenomena of long time duration (wave propagation for example) it quickly shows its limitations: CPU costs become prohibitive and dissipation becomes too important. In this part we intend to study a possibility of improving the operators that are used within SPH. We propose to couple the SPH method to a Lagrangian Finite Volumes method (which, theoretically, has more precise operators) with the intention of taking the advantages of SPH into FVM and vice-versa. In other words, we want to show that it is possible to have an Finite Volumes (or more generally, a mesh-based) method that does not need any special treatment to take into account the free-surface or, seen in the opposite way that it is possible to have an SPH method with the precision of a FVM. The choice of a Lagrangian FVM as mesh-based method was made here only because it initially appeared to be the easiest choice for having a coupling.

It is necessary to state that in this part we aim only at a “proof of concept”, i.e., we want to take the first steps (and justify them) towards a more precise Smoothed Particle Hydrodynamics algorithm

which can also be seen as a more flexible Finite Volume method. We could not anyhow go further than the “proof of concept” in this thesis, both because we lacked of time, three years is not enough to perform a complete study on this matter (knowing that other directions of improvement were also followed), and because we encountered some problems on the way, which will be left to be answered in future works. Also, we intend, with such a coupling to better understand some of the properties of SPH, that are difficult to analyze due to its meshless nature.

First of all, we will present the Voronoi Finite Volume Method upon which the developments of this part reside. Afterwards, we will develop the justification that permits to couple SPH to FVM, and finally, in the last chapter, some initial results are presented.

Chapter 15

Voronoi Finite Volumes Method

The finite volumes method (FVM) is a numerical technique where the fluid domain is discretized by a finite number of cells that together compose a "mesh". Each one of these cells represents a volume of control by which the rate of change of physical fluxes are computed and physical quantities are advanced in time. Traditionally, FVM is an Eulerian method and in some cases (generally when applied to Fluid Structure Interaction problems where the mesh is deformed only locally) Arbitrary Lagrangian Eulerian (ALE). Due to the fact that the fluid domain is discretized using a mesh, when an ALE approach is used, it becomes computationally demanding to move (or reconstruct) a mesh. Nevertheless, this approach was followed by several researchers and the decades of development and experience developing these methods resulted in very accurate and reliable algorithms.

However, some authors [127, 128, 129, 130, 10, 131, 132] started to develop fully Lagrangian (actually ALE, detail that is addressed later) FVM. Due to the increase of CPU capabilities of today's computers, to the maturity achieved by the traditional FVM algorithms and the need of simulating complex flows which are not possible to be simulated or not precisely enough with an Eulerian (or ALE with small displacements) Finite Volume approach, Lagrangian approaches started to be studied. The need for such kind of algorithm appeared firstly in the field of astrophysics where the simulations have the characteristic of being very violent. There are several manners of better reproducing and simulating violent flows, and one of these is to let the mesh move freely with the flow, however, such technique is not popular in the research community due to the complexity (mainly linked to the algorithm) of moving a mesh. Meshless methods have then been developed, among them the Smoothed Particle Hydrodynamics, which was firstly developed for astrophysical purposes as well, as a solution to circumvent any problems related to moving a mesh. Since then, SPH knew a rapid development and reached a maturity stage, issue that was addressed earlier in this thesis.

In the family of meshed methods, Whitehurst [127] has presented a hydrodynamics code (FLAME) based on Delaunay and Voronoi tessellations which performed very well for a selection of academic test cases (no practical problems were treated). Gnedin [128] and Pen [129] have also proposed moving mesh hydrodynamic algorithms based on the continuous deformation of Cartesian meshes, which limited the level of distortion to which cells could be submitted. In some regions the Lagrangian solver failed due to high mesh distortion and Gnedin [128], for example, circumvented the problem by using an Eulerian solver whenever the distorted grid failed. Other developments with focus on cosmological simulations

can be mentioned, cf. Xu [130]. Moreover, very recently Dumbser *et al.* [132] proposed a new high order Lagrangian finite volume scheme using a nonlinear WENO reconstruction.

More recently, Springel [10] proposed a Voronoi moving mesh FVM method based on Voronoi volumes directed to cosmological hydrodynamic problems where the compressible Navier-Stokes equations are solved with the help of a Riemann solver. At every time step the mesh is rebuilt and it can automatically adapt to volume expansion, similarly to SPH. A very close approach was proposed by Heß [133] and Heß *et al.* [134] where the difference with respect to Springel's approach is the use of an artificial viscosity [120] to stabilize the scheme. With these two approaches, good results have been achieved both for hydrodynamic (cf. additionally to the previous citations [135]) and cosmological problems (see also [136, 137]).

A very good and complete bibliographical review was done by Loubère *et al.* [131]. In this article, a new Voronoi-based Arbitrary-Lagrangian-Eulerian method is presented (ReALE) which is divided into three steps: first a Lagrangian phase where the mesh is updated in time, followed up by a rezoning phase where the Voronoi mesh is rebuilt and lastly a remapping procedure is performed to transfer the Lagrangian solution into the new mesh. The Lagrangian step is based on a formulation that respects the Geometric Conservation Law, which states that the mesh volume change must be equal to the numerical computation of volume variation. We refer the reader to [131, 138] for a broader view of the Lagrangian mesh-based methods.

We may also cite the work done by Español & Serrano [139] where a fluid particle model is described, very close to the one presented in [10], based on the Voronoi volumes. In this article, they draw some comparisons to SPH and to a Delaunay triangulation based discretization.

In this thesis, we propose a coupled FVM-SPH method. The FVM in question is very similar to the ones presented in [10] and [139], and it was chosen due to the similarity to the Riemann-based SPH solvers, which may ease the coupling. It is also much simpler than the ReALE method [131] which comprises two additional steps, even though the latter method is more consistent and accurate (we recall that we only seek a “proof of concept” here). The following section are dedicated to the presentation of the Voronoi volume entity, to the description of the derived FVM discrete operators and to the description of the discrete Navier-Stokes equations that are solved in such a scheme.

15.1 The Voronoi volume

The Voronoi tessellation consists of a mathematical formulation to subdivide a given space into a finite number of closed and non overlapping regions. These volumes are generated (polygons in 2D and polyhedra in 3D) having as basis a given distribution of points in space. The dual of the Voronoi tessellation is the Delaunay tessellation which in 2D is a triangulation of the plane having as vertices the points that generate the Voronoi volumes. We give as example the tessellation seen in figure 15.1 (gently borrowed from [10]).

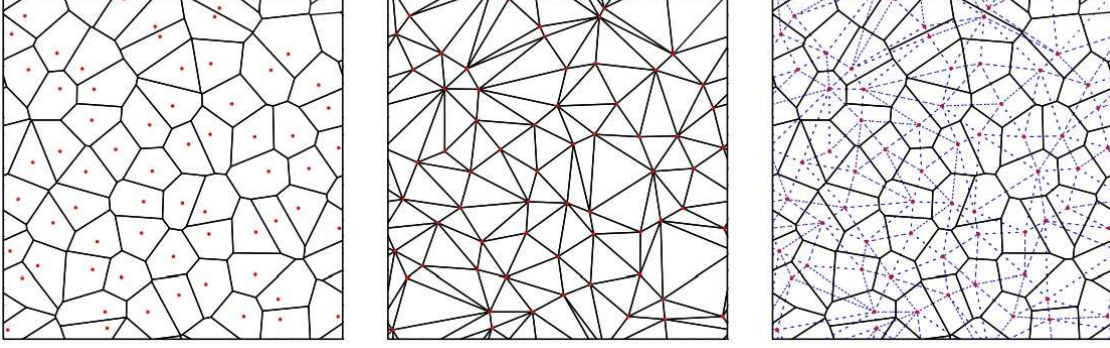


Figure 15.1: Example of Voronoi and Delaunay tessellations in 2D for a given points distribution. Left: the Voronoi tessellation; middle: the respective Delaunay triangulation and in the right both space sub-divisions are superposed (image from [10]).

The correlation between the Voronoi and the Delaunay tessellation is that the vertices of the Voronoi volumes are given by the midpoints of the circumcircles that surround each Delaunay triangle. Besides, we may add that what distinguishes the Delaunay triangulation of any other is that, for the first one, a circumcircle of a certain triangle may not contain any other generating point (that is, a vertex of another triangle).

The Voronoi tessellation seems to be a correct choice to represent the volume surrounding a given particle of fluid. The question is to know whether the Voronoi volumes have the desired properties. Español & Serrano [139] showed that the Voronoi volumes respect part of the desired features. To make it simple, we have linear consistency for the ∇P and $\nabla \cdot \sigma$ operators but not for $\nabla \cdot \vec{u}$ and $\nabla \vec{u}$ operators. However, even if the discrete divergence operator given by the Voronoi volume is not linearly consistent, it is exact for linear velocity fields (refer to [139]).

In the framework of this thesis, a Voronoi tessellation code was not developed. We preferred to use the Qhull library [140] which was very well suited to the purposes of this thesis. In order to correctly mesh the borders of the domain, we used the ghost particles technique, just like it is done for the SPH method, i.e, part of the fluid domain is symmetrized beyond the wall and the mesh is also generated for these ghost particles. This symmetrization (in position) allows to correctly model the wall-interfaces in the simulation domain.

15.1.1 Voronoi derivatives

The equations that we consider in our Finite Volume formalism are the same as the ones used for the weakly-compressible approach. Hence, Voronoi-based Finite Volume operators must be deduced and analyzed. As for SPH, the gradient and the divergence have to be defined.

By applying the Gauss theorem we may find the cell averaged gradient of any quantity f :

$$\langle \nabla f \rangle = \frac{1}{V} \int_V \nabla f dV = \frac{1}{V} \int_{\partial V} f d\vec{S} \quad (15.1)$$

where S is the contour surface of the volume V . This theorem implies that the value of ∇f at each

cell is approximated by averaging its own value with its neighbors. However, we may also apply the Gauss theorem on $(1 \cdot \vec{r})\nabla f$, yielding, for a giving volume:

$$\int_V \nabla f dV = \int_{\partial V} \vec{r} (\nabla f \cdot d\vec{S}) - \int_V \vec{r} \Delta f dV. \quad (15.2)$$

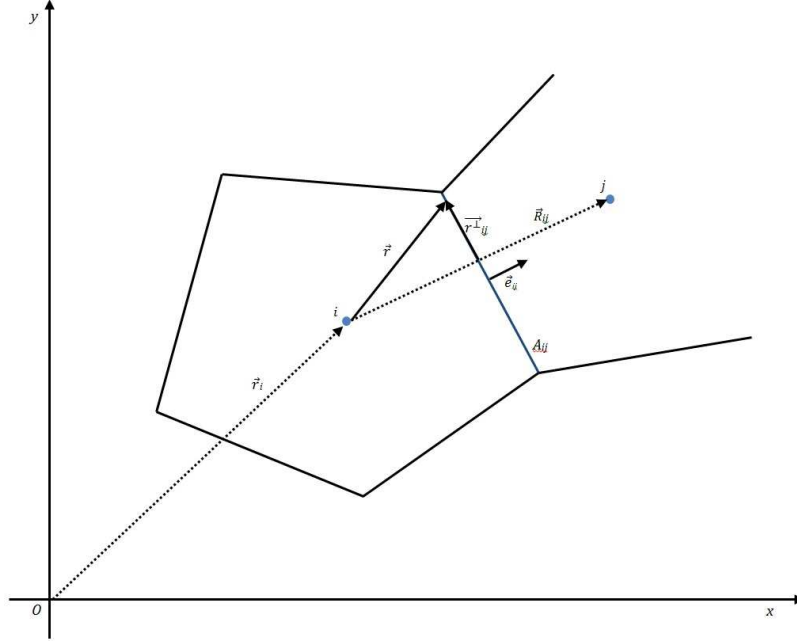


Figure 15.2: Voronoi discretization: interface between volumes i and j .

If we consider a Voronoi volume discretization, as in figure 15.2, the right-hand side of equation (15.2) for a given volume i reads:

$$\int_{\partial V} \vec{r} (\nabla f \cdot d\vec{S}) - \int_V \vec{r} \Delta f dV = \sum_{j \neq i} \int_{A_{ij}} \vec{r} (\nabla f \cdot d\vec{S}) - \int_V \vec{r} \Delta f dV. \quad (15.3)$$

From figure 15.2, we can write $\vec{r} = \vec{r}^\perp + \frac{(\vec{R}_{ij})}{2} + \vec{r}_i$ (which is based on the assumption that the surface separating two volumes is equidistant from both volumes centroid), hence equation (15.3) can be rewritten as:

$$= \sum_{j \neq i} \int_{A_{ij}} \left(\vec{r}^\perp + \frac{\vec{R}_{ij}}{2} + \vec{r}_i \right) (\nabla f \cdot d\vec{S}) - \int_V \vec{r} \Delta f dV \quad (15.4a)$$

$$= \sum_{j \neq i} \int_{A_{ij}} \left(\vec{r}^\perp + \frac{\vec{R}_{ij}}{2} \right) (\nabla f \cdot \vec{e}_{ij} d\vec{S}) + \vec{r}_i \sum_{j \neq i} \int_{A_{ij}} \nabla f \cdot \vec{e}_{ij} d\vec{S} - \int_V \vec{r} \Delta f dV \quad (15.4b)$$

$$= \sum_{j \neq i} (\nabla f \cdot \vec{R}_{ij}) A_{ij} \left(\frac{1}{2} \vec{e}_{ij} + \frac{1}{\|\vec{R}_{ij}\| A_{ij}} \int_{A_{ij}} \vec{r}^\perp d\vec{S} \right) + \vec{r}_i \int_V \Delta f dV - \int_V \vec{r} \Delta f dV. \quad (15.4c)$$

where \vec{e}_{ij} is used to represent the normal vector to the facet separating the volumes i and j , which is different from the notation \vec{n}_{ij} used in this thesis (mainly for the SPH formulation) to denote the normal vector that connects points i and j . Using the following geometrical equality $\vec{c}_{ij} = \frac{1}{A_{ij}} \int_{A_{ij}} \vec{r} d\vec{S}$, which is a vector pointing to center of the face between i and j from the mid point between positions \vec{r}_i and \vec{r}_j , we may write equation (15.4c) as:

$$\langle \nabla f \rangle_i = \frac{1}{V_i} \left[\sum_{j \neq i} (\nabla f \cdot \vec{R}_{ij}) A_{ij} \left(\frac{1}{2} \vec{e}_{ij} + \frac{\vec{c}_{ij}}{\|R_{ij}\|} \right) - \int_V (\vec{r} - \vec{r}_i) \Delta f dV \right]. \quad (15.5)$$

The second term in the equation above is equal to zero for linear scalar fields. If we use the simplification $\nabla f \cdot \vec{R}_{ij} = (f_j - f_i)$, we obtain the gradient estimation of f for a given point i :

$$\langle \nabla f \rangle_i = \frac{1}{V_i} \sum_{j \neq i} A_{ij} (f_j - f_i) \left(\frac{1}{2} \vec{e}_{ij} + \frac{\vec{c}_{ij}}{\|R_{ij}\|} \right) \quad (15.6a)$$

$$= \frac{1}{V_i} \sum_{j \neq i} A_{ij} \left[(f_j - f_i) \frac{\vec{c}_{ij}}{\|R_{ij}\|} + (f_j + f_i) \frac{\vec{e}_{ij}}{2} \right] \quad (15.6b)$$

where $\sum_j A_{ij} \vec{e}_{ij}$ is used to get a symmetrical expression for $\langle \nabla f \rangle_i$ that ensures conservation of momentum just like in SPH (refer to section 3.2). The same procedure may be followed to derive the divergence operator of a vector field \vec{f} for a given particle i , which reads:

$$\langle \nabla \cdot \vec{f} \rangle_i = -\frac{1}{V_i} \sum_{j \neq i} A_{ij} (\vec{f}_j - \vec{f}_i) \cdot \left(\frac{1}{2} \vec{e}_{ij} + \frac{\vec{c}_{ij}}{\|R_{ij}\|} \right) \quad (15.7)$$

Note that these equations are obtained considering a cell centered finite volume formalism, i.e., the physical quantities are known at the center of these cells and not at its vertices or facets centers. However, we remind that if the variables at each facet or vertex of the volume are known, it is possible to use the gradient operator as:

$$\langle \nabla f \rangle_i = \frac{1}{V_i} \sum_{j \neq i} f_{ij} \vec{e}_{ij} A_{ij} \quad (15.8)$$

and the divergence as:

$$\langle \nabla \cdot \vec{f} \rangle_i = \frac{1}{V_i} \sum_{j \neq i} \vec{f}_{ij} \cdot \vec{e}_{ij} A_{ij} \quad (15.9)$$

where \vec{f}_{ij} is the value of \vec{f} on the middle of the facet between volumes i and j . Note that, in order to achieve a good order of convergence with operators (15.8) and (15.9), one needs to be able to precisely predict the values of f at this location. If the variables are cell-centered, which is the case of the FVM studied in this thesis, this implies using an appropriate extrapolation scheme (eg. MUSCL).

15.1.2 Convergence of the Voronoi discrete operators

The convergence of these Voronoi operators was studied by Español & Serrano [139]. In this study, they verified the order of convergence of the gradient and divergence operators with respect to the grid

refinement. They studied two grid topologies, one where the points were randomly distributed and another where a triangular distribution was followed (regular grid).

For the gradient operator they showed that with a random grid, a $O(1)$ convergence order is achieved while with a regular grid $O(2)$ is found. For the divergence operator, the order of convergence with a random grid is $O(0)$ and with a regular grid the same order achieved for the gradient operator is achieved, that is $O(2)$. It is no surprise that the Voronoi-based operators deduced in section 15.1.1 do not perform well because they are based on the assumption that the two neighbor volumes centroid are equidistant from the facet separating them which is true for regular grids but not applicable to irregular ones.

Hence, in order to keep a good order of convergence, it is necessary to have a regular point distribution. In this thesis we focus on Lagrangian methods, which in most cases (mainly considering the experience from the SPH method) on irregular/random points distributions result in a bad order of convergence. Having an order of convergence of $O(0)$ may be an impediment to achieve a good quality of results, which has already been showed in [139]. Consequently, as it will be shown later in this thesis, a manner to keep/achieve a ‘regular’ mesh distribution must be used in order to keep simulations stable.

15.2 Discrete Euler equations in the Voronoi-FVM context

The weakly-compressible approach is used for the proof of concept in this section, hence, equations (2.5) are discretized. Here, we use a Riemann-based FVM approach just as the one used for Riemann-SPH, and consequently, following the notation in equations (3.25) and (3.28) (concerning the ALE formalism, \vec{u}_0 , and Riemann problem, $G(\phi_i, \phi_j, \vec{e}_{ij}, \vec{u}_{0i}, \vec{u}_{0j})$), we have:

$$\frac{d\vec{r}_i}{dt}\bigg|_{\vec{u}_0} = \vec{u}_{0i} \quad (15.10a)$$

$$\frac{dw_i}{dt}\bigg|_{\vec{u}_0} = \sum_{j \neq i} A_{ij} (\vec{u}_{0i} - \vec{u}_{0j}) \cdot \left(\frac{1}{2} \vec{e}_{ij} + \frac{\vec{c}_{ij}}{\|R_{ij}\|} \right) \quad (15.10b)$$

$$\frac{dw_i \vec{\phi}_i}{dt}\bigg|_{\vec{u}_0} = w_i \vec{g} - \sum_{j \neq i} A_{ij} G(\phi_i, \phi_j, \vec{e}_{ij}, \vec{u}_{0i}, \vec{u}_{0j}) \quad (15.10c)$$

$$p_i = \frac{1}{\gamma} \rho_0 c_0^2 \left(\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (15.10d)$$

The attentive reader may notice that the volume w_i evolution is performed numerically, when it could be done by computing it directly from the Voronoi tessellation. The choice of using the numerical integration rather than the direct one will be addressed in the next chapter.

15.2.1 MUSCL Scheme

In order to solve the Riemann problem at each interface between two Voronoi cells, it is necessary to perform a gradient estimation and linear reconstruction. This is done for the basic primitive variables

ρ and u using the gradient operator of equation (15.6). A generalized unstructured grid MUSCL approach has been proposed in [10] for the specific case of Voronoi meshes using the slope limiter by Barth & Jespersen [141]. In this thesis we use the generalized minmod limiter due to its known efficiency when applied to SPH and to a Cartesian-grid Finite Volume method [142].

Chapter 16

Voronoi-SPH coupling

In this chapter the new Voronoi-based ALE Finite Volume method coupling with the Smoothed Particle Hydrodynamics method is described and justified. After the theoretical description, some important aspects of the coupling are discussed. Some initial and conclusive results are presented and conclusion are drawn.

16.1 Description

The idea of coupling the SPH method to a Voronoi Finite Volume approach came from the fact that the two methods are equivalent (from the point of view of discrete operators) and similar (centered operators with centered variables and the possibility of using a Riemann-solver in both methods). Moreover, one method can be used to overcome the drawbacks of the other. For example, the first one lacks of precision for its operators as for the latter one it is difficult to treat interfaces. Therefore, they can be seen and coupled in a complimentary way.

Considering the SPH discrete operator in equation (3.12), the following analogy to the Finite Volume method can be done:

$$\sum_{j=1}^N f_j \alpha_{ij} \vec{n}_{ij} w_j \equiv \frac{1}{w_i} \sum_{i \neq j} f_{ij} A_{ij} \vec{e}_{ij}, \quad (16.1)$$

where $\alpha_{ij} w_j$ represents a fictitious boarder with the neighbor particles, with the summation forming a fictitious volume of control. Note that, even if the SPH and the FVM operators are equivalent, they have different orders of convergence and precision (the exception is when handling with random-distributed meshes where they have similar behaviors).

As stated in the introduction chapter of this part, the idea is to take advantage of the fact that SPH takes into account the presence of the free-surface implicitly, and consequently, in this thesis we will focus on free-surface flows (the method proposed here may be adapted to multi-phase flows). Therefore, considering a fluid domain as in figure 2.1, the region close to the free-surface is modeled in a SPH manner and the rest with Finite Volume formalism. The main problem with such a coupling resides in the treatment given to the SPH-FVM interface.

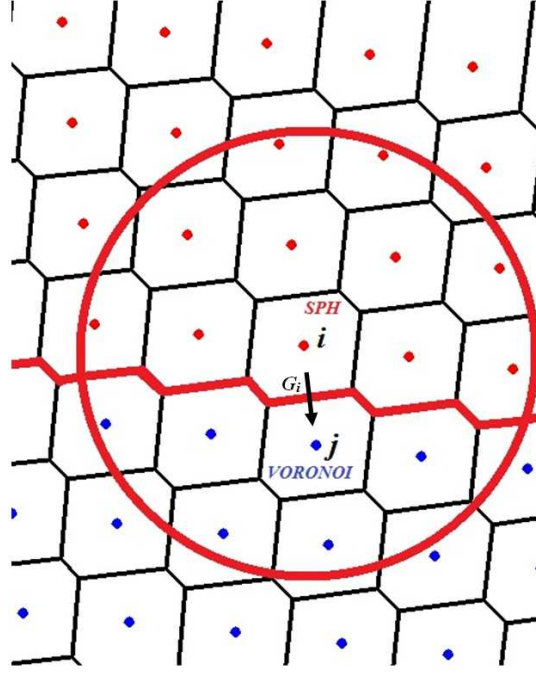


Figure 16.1: Illustration of the SPH-FVM coupling.

Figure 16.1 shows a typical particle distribution where the red particles represent SPH particles and blue ones the center of the Voronoi volumes. The red circle represents the SPH kernel support centered at particle i . The interface between the two zones, also in red, is known from the Voronoi tessellation and is available. The first question to be raised is: how the Voronoi volume j is taken into account into the SPH operators of particle i and the other way around, how is the SPH particle i taken into account in the Finite Volume Voronoi discrete operators for volume j ?

Considering that we are in the context of weakly-compressible Navier-Stokes equations and that we are treating the interaction between two particles or two volumes as a Riemann problem, the solution G_i of this problem can be precisely computed for the Voronoi volume j and hence we could obtain the Riemann fluxes at every face of the SPH-FVM interface. After obtaining these fluxes, it is necessary to find the equivalent of these fluxes to be inserted into the SPH operators in a conservative manner. Amongst all the thought possibilities to include these fluxes into the SPH operators, an adaptation of the Normal Flux method proposed by De Leffe *et al.* [44] and De Leffe [19] would seem as the most appropriate. However, a special attention would be necessary to not add up too much flux into the SPH sub-domain which would result in instabilities.

A simpler way to treat this interface exists, but less precise than the last one, that is to do nothing. In other words, as the tessellation is done through the whole domain and thus performed also for the SPH particles of the fluid domain, the Voronoi volumes would take into account the SPH particles as if they were also Voronoi Finite Volumes, and the SPH particles would consider the Voronoi volumes whose centroids are inside this kernel as if they were also SPH particles. Despite being simple, this approach is non-conservative due to the different level of accuracy of the SPH and FVM operators (numerical results are presented later in this thesis). However, such a coupling is a priori consistent, i.e.

it is expected that non conservative errors are less important as the mesh discretization (or particles distribution) increases and as the number of neighboring particles per particle for the SPH operator increases. This is the approach used in the proposed mixed method.

In order to separate the fluid domain into the SPH and FVM sub-domains, it is necessary to perform a free-surface detection. This is done by applying the method proposed by Marrone *et al.* [1] already used in this thesis for the semi-implicit ISPH algorithm. The thickness of the SPH sub-domain must be chosen as well and may vary following the test case considered. For each test case considered here, a different thickness for the SPH sub-domain was considered. As far as we could verify, increasing the thickness of the SPH zone only make the tessellation more robust, but does not interfere with the quality of the solution. Note that one SPH particle may become a Voronoi volume and vice-versa by transiting from one zone to another. When a SPH particle becomes a Voronoi volume, its location becomes the cell centroid that gives origin to the tessellation, and on the other way around, when a Voronoi volume becomes a SPH particle, its centroid becomes the SPH particle. In both directions, the physical quantities are kept the same, without any particular treatment.

From now on we refer to this news coupled method as Voronoi-SPH.

16.2 Questions on the Voronoi-SPH coupling

From a first analysis of this new Voronoi-SPH coupled method some questions raised. The first one concerns the volume evolution and the second the grid regularity (intrinsically connected to the ALE feature of the algorithm) which affects directly the method order of convergence, precision and robustness. These questions are addressed in the following sections.

16.2.1 Volume time integration

In the SPH context, the particle volume evolution is computed numerically (actually the volume itself is ‘imaginary’) and the same could be done for the FVM part of the domain. Instead, considering the fact that at every time step the tessellation is performed, it is possible to compute a volume evolution based on the mesh deformation. However, due to the Lagrangian nature of the method, particles may transit from one region to another and consequently, using this last approach could lead to instabilities and discontinuities when a Voronoi volume passes to the SPH domain and becomes a particle. For robustness and stability reasons, a continuity in the volume computation must be kept and consequently we preferred to proceed with a numerical volume evolution for both SPH and FVM. Moreover, choosing to compute the volume evolution through the mesh deformation would lead to a dephased computation with respect to the other derivatives. It is only possible to compute the volume evolution through the mesh from time instant $n - 1$ to n and not from n to $n + 1$ as it is done for the other variables. Nonetheless, letting the volume evolve numerically leads, as stated in [131], to the non respect of the Geometric Conservation Law.

One alternative would be to use the approach from [131], where a rezoning and a remapping procedure are performed to ensure that the mesh deformation and the volume time derivatives are in phase and in agreement. As pointed out in [131], it is not possible to have a true Lagrangian formalism using a Voronoi-cell as this entity is not a Lagrangian object. Moreover, a fully-Lagrangian approach

using Voronoi cells does not respect the Geometric Conservation Law, which implies that small changes in the particle positions can produce huge volume variations leading to instabilities. It is, therefore, preferable to control the velocity with which the points originating the Voronoi-cells move, which leads us to the next section.

16.2.2 Grid regularity

As seen in section 15.1.2, as the grid goes irregular, the convergence properties of the Voronoi Finite Volume gets poorer. Hence, it is important to keep the grid regularity throughout the simulation. However, due to the Lagrangian nature of the flow and also to the non trully-Lagrangian nature of the Voronoi object, volumes tend to an irregular distribution. Figure 16.2 shows a typical irregular mesh for the test case of a dam-break (see 7.4.2, the volumes that are close to the top of the figure belong to the free-surface zone). Being the Voronoi-based FVM presented in chapter 15 cell-centered, a desired feature of the mesh is to have the center-of-mass of each cell to coincide with the point originating the cell. As mesh goes irregular, cells tend to lose their isotropic shape leading to a non-centroidal Voronoi tessellation.

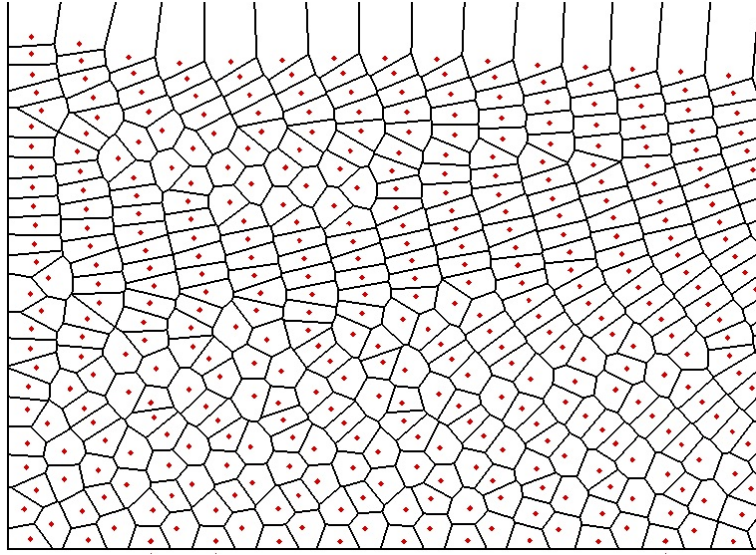


Figure 16.2: Irregular mesh for a dam-break test case.

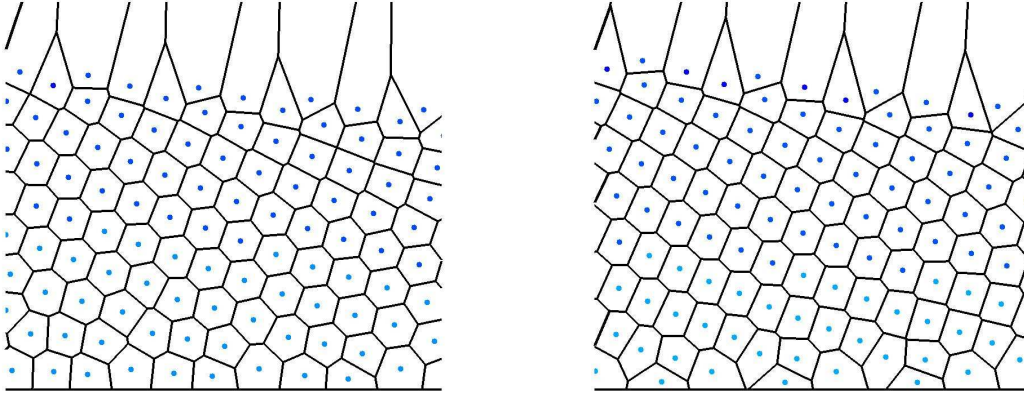
Springel [10] proposed two different techniques to preserve a regular grid, one acting on the isotropy of each cell and another that regulates the mesh motion (techniques also used in [133]). The first one is an extra velocity added to equation (15.10a) and the second one requires the solution of a Poisson-like equation, which results into displacement vectors used, then, to modify once again equation (15.10a). One could also opt for a special type of artificial viscosity or special artificial forces to regularize the particle positions.

In this thesis, we propose to future researchers a simpler method inherited from the SPH method to control both the isotropicity of cells and the grid regularity. Sometimes in SPH it is necessary to correct particle velocities or positions to prevent particle of following their Lagrangian path which

sometimes lead to particle distributions that are not good for the SPH interpolators (typically, in-line formation of particle with some voids that appear between those lines during an accurate SPH simulation). Recently, in the LHEEA laboratory, we started using a method that corrects the velocity by which particles are moved (u_{0i}) and therefore profiting of the ALE nature of the Riemann-SPH scheme. This technique is similar to the XSPH correction [79], but differently from this correction, it does not smooth the physical velocities of particles (u_i) which remain unchanged. The following equation can be used to achieve a regular particle distribution:

$$u_{0i} = u_i - \frac{F_{ALE}|u_i|R_i}{w_i} \sum_{j=1}^N \nabla W_{ij} w_i w_j, \quad (16.2)$$

where F_{ALE} stands for a non-dimensional controlling factor which has to be set depending on the simulation. This term depending on the kernel gradient summation gives an equivalent of a normal internal vector pointing towards regions of void or with less particles. Note that, in order to use this technique, the free-surface thickness (SPH zone) must be sufficiently thick so that the calculation of this term within the Voronoi volumes is not influenced by its presence. As a result we show in figure 16.3 the meshes obtained in two different simulations, one with an arbitrary velocity (where $F_{ALE} = 2$) and another without, for the 2D dam-break test case (presented in section 7.4.2) at $t = 0.88s$. We can verify that by using an arbitrary velocity u_{0i} , it is possible to achieve a more uniform mesh (in terms of connectivities and volumes) and consequently a better convergence for the Voronoi FVM operators can be reached. This is especially true close to the solid boundaries. Note that the volume shape in the upper part (SPH zone) do not matter since they are not used.



(a) with arbitrary velocity

(b) without arbitrary velocity

Figure 16.3: Mesh obtained with the use of an arbitrary velocity u_{0i} (a) and without any particular treatment (b).

Such a technique prevents as well the large instantaneous distortions that may occur during a simulation based on the Voronoi tessellation, which can cause instabilities (mainly caused by the non

respect of the Geometric Conservation Law). Figure 16.4 shows the large volume and connectivity changes that may occur between two time instants without using any mesh control technique (in this example we also consider a dam-break test case, see section 16.4.2).

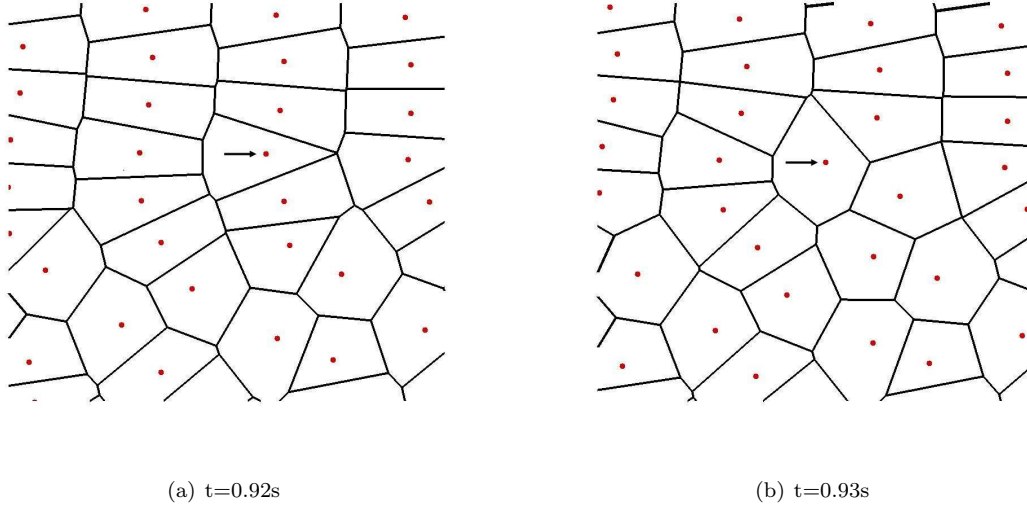


Figure 16.4: Mesh distortion between two time steps for the dam-break test case without a mesh control technique.

16.3 Algorithm

The algorithm of the coupled Voronoi-SPH method presented in the previous sections is shown below. Note that it is very similar to a SPH algorithm and that a simple Euler time-advance scheme is used here. In order to use a more precise time integration scheme, like a 4th order Runge Kutta, it would be necessary to perform more than once the tessellation which would compromise the CPU costs feasibility of the method. Moreover, using a different integration scheme than the Euler one would introduce an extra variable in the study proposed in this thesis, that is to assess whether if the fact of performing more than one tessellation per time step affects the results obtained.

Algorithm 2 Voronoi-SPH solver procedure

- 1: **while** $t < t_{end}$ **do**
 - 2: Perform the Voronoi tessellation based on \vec{r}_i
 - 3: Perform the free-surface detection algorithm and define the SPH and Voronoi sub-domains
 - 4: Compute the arbitrary velocities for the Voronoi volumes u_{0i} (if desired)
 - 5: Compute the gradients to be used for the MUSCL scheme for the SPH and Voronoi sub-domains
 - 6: Compute the SPH and the Voronoi-FVM time-derivatives
 - 7: Advance particles and cells in time
 - 8: **end while**
-

The time step used in this algorithm is the same as presented in section 3.5.

16.4 Results

Due to the difficulties related to the execution of a test case using a Voronoi-based FVM (mainly regarding the meshing procedure), we did not proceed to a validation of the method itself (simulation without a free-surface or an interface) even though we obviously verified its implementation. We left the process of validation of such a method to be done directly by testing the coupling SPH-FVM. To do so, we performed two test cases. The first one, which is very simple, intends to verify that such a coupling is capable of maintaining a hydrostatic pressure profile. The second one has the objective of verifying the applicability of this new Hybrid-SPH algorithm to typical violent flow phenomenon: 2D dam-break. The validation process is then concluded by the Gresho vortex problem where the intention is to verify the dissipation differences between the Voronoi-FVM and SPH methods.

16.4.1 Hydrostatic test case

This first test case is similar to the test case used in section 10.5, but with a smaller domain. Instead of a 2.0 meter length, here we use a 1.0 meter length tank with the same depth. The density is taken as $\rho_0 = 1 \text{ kg/m}^3$, the reference speed of sound equal to $c_0 = 10 \text{ m/s}$ and the gravity equal to $g_0 = 9.81 \text{ m/s}^2$. The simulation was performed for the total time of 5 seconds and a mesh refinement of $\Delta x = 0.02 \text{ m}$ was considered. Note that no arbitrary velocity was used ($u_{0i} = u_i$).

Figure 16.5 shows the particle positions at $t = 1.0 \text{ s}$. The two first lines of particles in this simulation are treated as SPH particles and the rest as Voronoi volumes. On the top right plot of figure 16.5 a plot with the velocity vectors is shown where the particles and volume centers are colored with the velocity in the y-direction. The difference between the values given by the SPH and FVM interpolators results in a small local discontinuity that propagates across the Voronoi domain bouncing on the bottom, which makes volumes to deform as shown in the down right plot of the same figure. For this special test case, the fluid must remain static, therefore, any small perturbation results in an unstable simulation (this also happens with the SPH method when applied to this same test case, but at a lower level).

In order to better understand the discontinuity at the SPH-FVM interface, this test case was used to analyze the differences between the Finite Volume and Smoothed Particle Hydrodynamics operators. Being the initial particle distribution and mesh for the Voronoi-FVM solver Cartesian, we can easily analyze these operators. At $t = 0.0 \text{ s}$, for a particle located exactly in the middle of the domain ($x = 0.5 \text{ m}, y = 0.5 \text{ m}$), we computed the Riemann-fluxes summation for this volume/particle following the FVM and the SPH approaches (equations (15.10c) and (3.28) respectively, without considering the gravity acceleration) and separated it into the two different directions and into right and left (positive/negative) for the x-direction and into up and down (positive/negative) for the y-direction. Table 16.1 shows these different values. The final result of the operator summation is the same for both methods, apart from the operator precision difference (here, 0.095%), however the contributions in each direction are not the same. Despite the correlation of equation (16.1) and the works by Vila [25] that showed the equivalence of SPH operators to Finite Volume operators, these results recall that SPH uses convolution operators and for that reason, the numerical correlation with other methods is not simple to be done. This makes complex the derivation any kind of flux exchange between SPH and FVM at the interface, if not considering the simple non-conservative way used here.

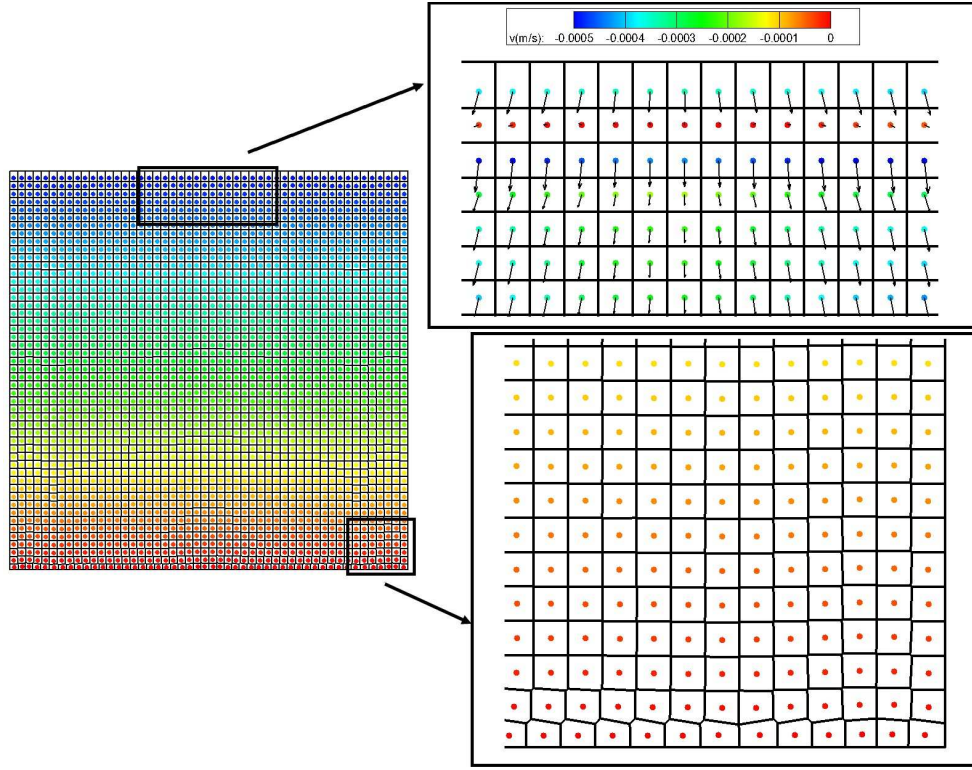


Figure 16.5: Hydrostatic case results using the Voronoi-SPH method.

Table 16.1: Numerical values and differences between the results given by the SPH and Voronoi-FVM operators

Method	Direction	Positive	Negative	Summation
SPH	x	-21.457	21.457	0.0
Voronoi-FVM	x	-26.090	26.090	0.0
SPH	y	-20.936	21.980	1.044
Voronoi-FVM	y	-25.568	26.613	1.045

16.4.2 2D dam-break flow

This test case is the same that was treated in the two previous parts of this thesis. Two different particle distributions (or mesh sizes) were tested: $\Delta x = (0.02m, 0.01m)$ resulting in 1800 and 7200 particles/meshes respectively. We compare the pressure given at the pressure probe $p1$ between Voronoi-SPH, SPH and experimental results. For the SPH simulation and the SPH part of the Voronoi-SPH coupling we used the Wendland kernel with an average of 30 neighbor particles per particle.

Differently from the last test case (hydrostatic tank), the dam-break is a dynamic problem. Therefore, the instabilities created in the interface Voronoi-SPH may be filtered by the mesh deformation and particle displacements. The gravity is taken as $g_0 = 9.81 \text{ m/s}^2$ and the reference speed of sound

is equal to $c_0 = 30 \text{ m/s}$. The Courant-Friedrichs-Lewy factor that controls the time step is set to $CFL = 0.1$ which is $O(3 - 6)$ times smaller than the typical value used in SPH. In the Voronoi-SPH simulation the mesh deforms, and consequently the characteristic length for each cell changes, and as stated, in such a formalism this length dictates the time step. As we do not have a good understanding of how the Voronoi volumes deform in time, the CFL constant is kept very small which ensures that the acoustic time step limit is respected. Also, the thickness of the free-surface in this test case is considered to be equal to three SPH particles, or $3\Delta x$, i.e. a bit more than R . Note as well that for the SPH simulation a 4th order Runge Kutta scheme is used and for the Voronoi-SPH a simple Euler scheme is used.

Figures 16.7 and 16.8 show the particle distributions for the SPH simulation and the mesh distortion together with the center of each mesh for the Voronoi-SPH simulations with an inter-particle initial distance of $\Delta x = 0.02m$. Figures 16.9 and 16.10 show the same results for $\Delta x = 0.01m$. Focus is given to time instants during the impact against the right-wall and the wave breaking phenomenon formed after the impact.

Globally, if we take as basis the results given by SPH (having in mind that the SPH method was already validated for this flow), Voronoi-SPH can predict well the dynamics of the flow. Note that the shapes of the free-surface have a very good resemblance for both particle distributions. The level of noise in the pressure field is quite the same for both methods considering coarser particle refinement, however, the returning wave seems to be a bit more viscous just before breaking in the Voronoi-SPH simulation, cf. $t = 1.4s$ in figure 16.7. After this time, a bubble is formed and the shape of this bubble is quite well predicted (always compared to the result given by SPH), cf. figure 16.8. As mesh is refined, the results given by Voronoi-SPH and SPH show a closer agreement. The pressure becomes quite smooth and comparable to the results given the SPH simulation. Also, the shapes of the wave predicted by both methods, right before breaking, are quite similar. Even the small deformations (ripples) in the free-surface at $t = 1.4s$ in figure 16.9 are captured by both methods. For the finer mesh, at $t = 1.8s$ (cf. figure 16.10), both methods are capable of predicting a small wave that comes back towards the wall during the wave breaking process. This phenomenon is not predicted with the coarser particle distribution, leading us to state that, just like for the SPH method, the Voronoi-SPH converges to a better description of the free-surface flow when a finer mesh/particle distribution is used.

The attentive reader will notice that the SPH results in figures 16.7, 16.8, 16.9 and 16.10, particles tend to keep in-line formations, feature of SPH that has already been addressed by several authors (mainly when the precision of the algorithm is increased by using a Riemann-based SPH formulation or renormalization). The Voronoi-SPH coupling, on the other hand, does not present the same feature, that is, the lines are broken during the simulation. Actually, without any extra correction (not even an arbitrary velocity u_0), the volumes tend to take an isotropic shape (even if it results in cells with different volumes), breaking the initial Cartesian formation, leading to quite regular unstructured distribution. This feature is sought in SPH, as an unstructured distribution induces better accuracy, stability and convergence [49, 50]. Note that the thin water filaments close to the wall, cf. $t = 1.4s$ in figure 16.9, have a structured formation for the SPH simulation and an unstructured one for the Voronoi-SPH result, even if these particles, for the latter method are considered as being SPH. While

particles are considered as finite volumes, they acquire an unstructured formation and when they become SPH particles this formation is kept. The noise originated by the lack of conservation in the SPH-Voronoi interface may be a source of increase of the speed of the mesh “unstructuring” process.

A small amount of particle clumping may also be seen in the SPH results, especially close to the free-surface and right after the bubble explosion at $t = 2.0s$ (this phenomenon is more important in the coarser simulation, cf. figure 16.8). For the Voronoi-SPH coupling results the same issue is not seen: the volumes tend to prevent the formation of clumped configurations. Due to the fact that SPH is a meshless method, the “fictitious” volumes of particles may get distorted in a different manner than for the Voronoi volumes. They can deform keeping the structured in-line formation, or originating clumping, which are not stable and precise and therefore leading to undesirable configurations. In a way, the fact of having a mesh in the Voronoi-SPH algorithm introduces a certain stiffness that prevents clumped configurations.

We may also compare the pressure at probe $p1$ given by the two compared methods against the experimental results. This is done in figure 16.6 for the two particle refinements $\Delta x = (0.02m, 0.01m)$. The solution given by Voronoi-SPH is always noisier than the SPH one. However, despite the higher level of noise, the same trends and behavior as in experiments are observed. Voronoi-SPH predicts at the same instant the impact against the wall ($T = 2.5$), the wave-breaking phenomenon ($T = 6.5$) and the bubble collapse ($T = 7.5$). Contrary to what can be observed in figures 16.7, 16.8, 16.9 and 16.10, the pressure profile measured at $p1$ is noisier for the finer resolution. It is interesting to observe that the pressure average level for the Voronoi-SPH follows the SPH one and that the same convergence tendencies are observed, notably the pressure peak elevation during the water front impact against the right wall.

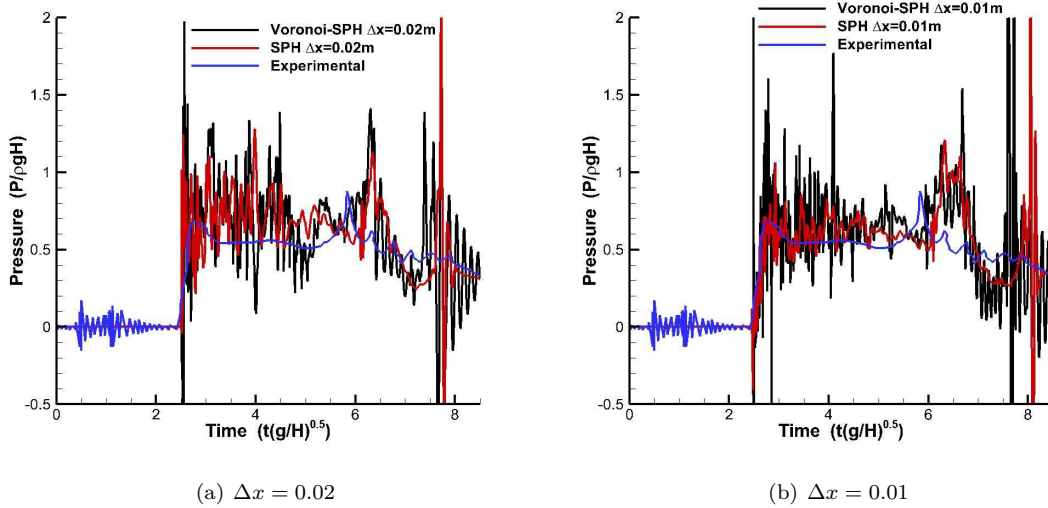


Figure 16.6: Pressure at probe $p1$ for the two methods considered in this study for $\Delta x = 0.02$ and $\Delta x = 0.01$.

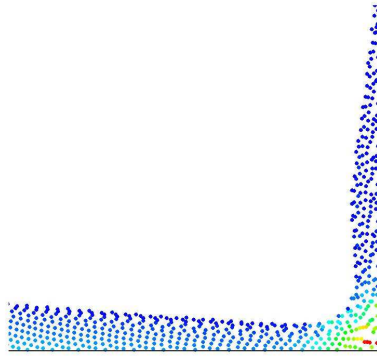
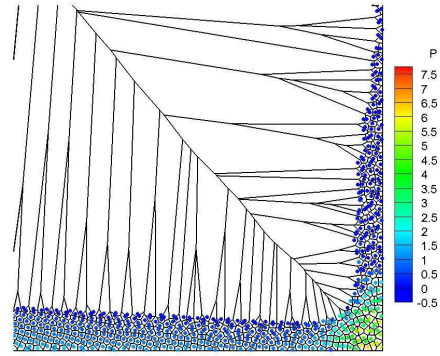
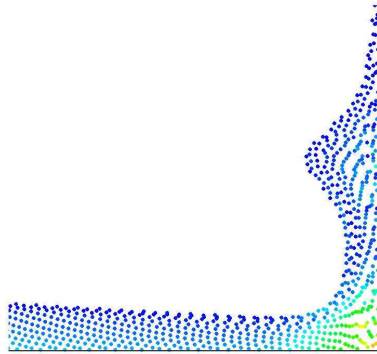
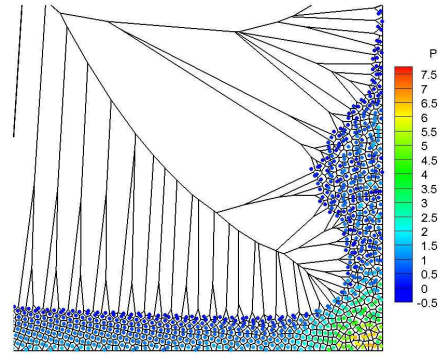
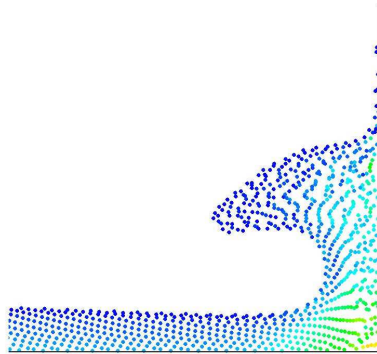
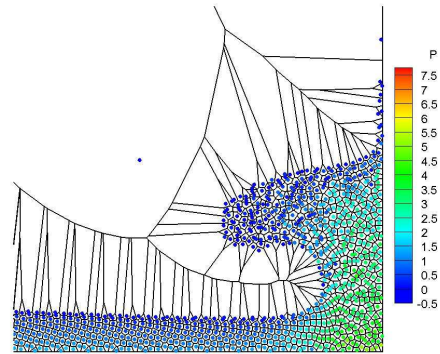
(a) SPH $t = 1.0s$ (b) Voronoi-SPH $t = 1.0s$ (c) SPH $t = 1.2s$ (d) Voronoi-SPH $t = 1.2s$ (e) SPH $t = 1.4s$ (f) Voronoi-SPH $t = 1.4s$

Figure 16.7: Comparison between the results given by the SPH (left) and Voronoi-SPH (right) methods from $t = 1.0s$ to $t = 1.4s$.

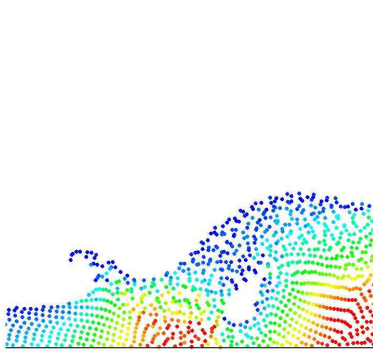
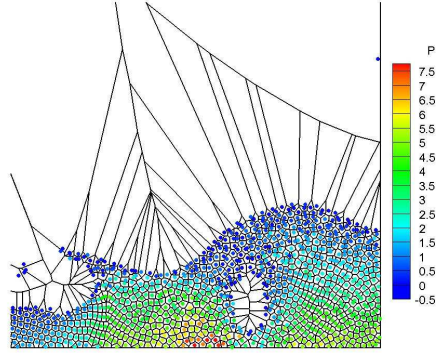
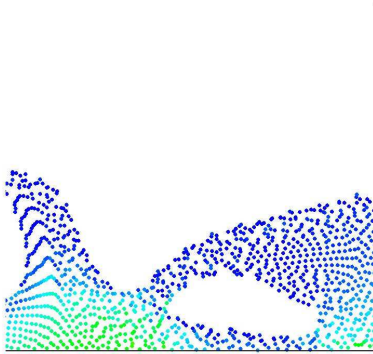
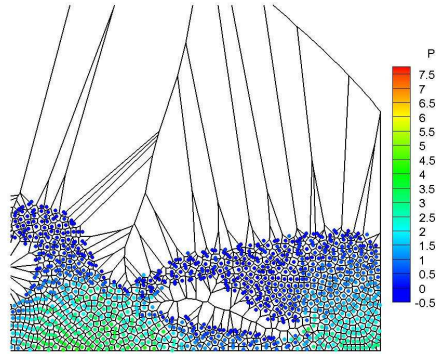
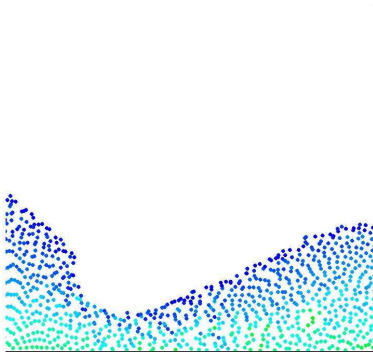
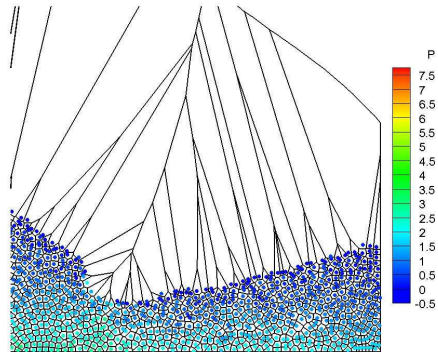
(a) SPH $t = 1.6s$ (b) Voronoi-SPH $t = 1.6s$ (c) SPH $t = 1.8s$ (d) Voronoi-SPH $t = 1.8s$ (e) SPH $t = 2.0s$ (f) Voronoi-SPH $t = 2.0s$

Figure 16.8: Comparison between the results given by the SPH (left) and Voronoi-SPH (right) methods from $t = 1.6s$ to $t = 2.0s$.

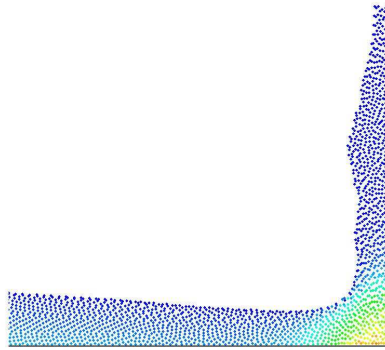
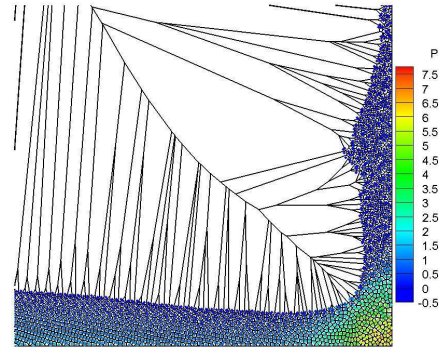
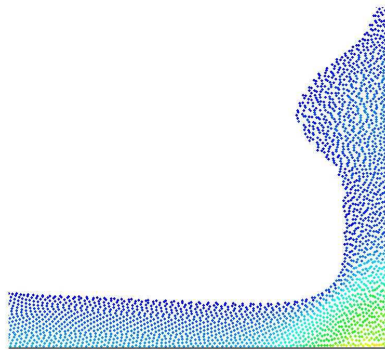
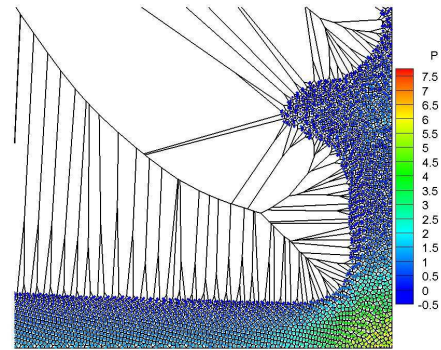
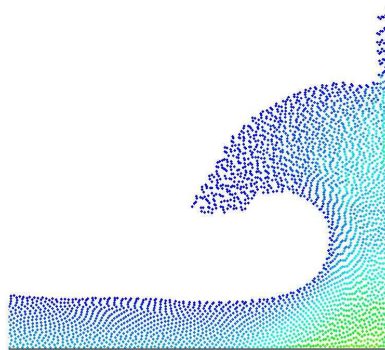
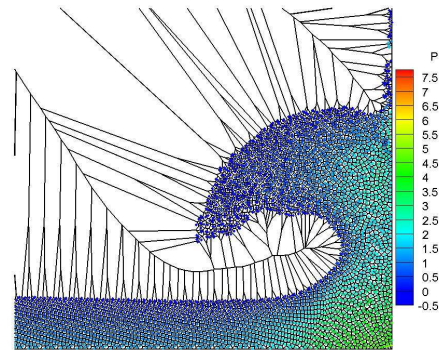
(a) SPH $t = 1.0s$ (b) Voronoi-SPH $t = 1.0s$ (c) SPH $t = 1.2s$ (d) Voronoi-SPH $t = 1.2s$ (e) SPH $t = 1.4s$ (f) Voronoi-SPH $t = 1.4s$

Figure 16.9: Comparison between the results given by the SPH (left) and Voronoi-SPH (right) methods from $t = 1.0s$ to $t = 1.4s$.

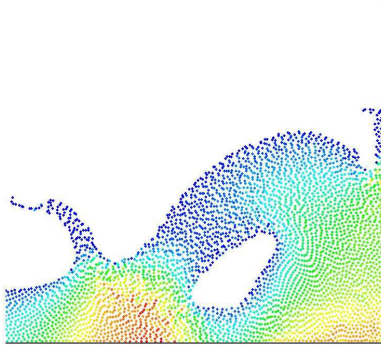
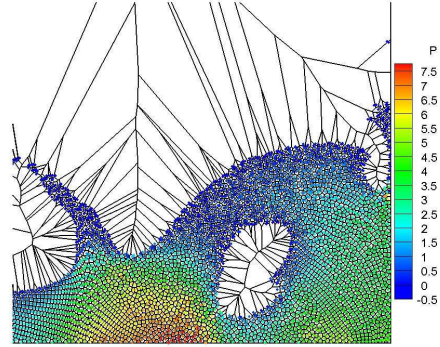
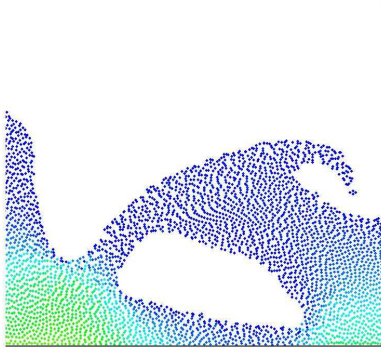
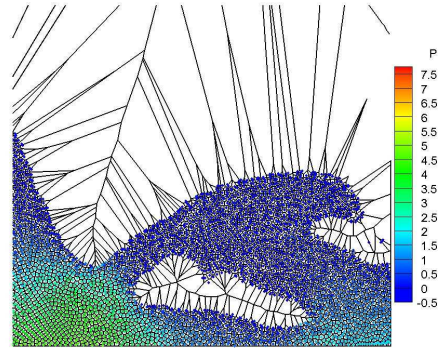
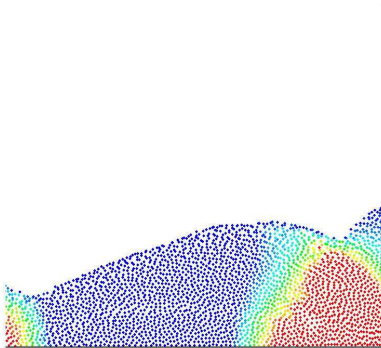
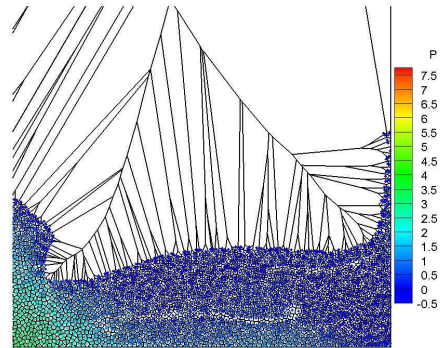
(a) SPH $t = 1.6s$ (b) Voronoi-SPH $t = 1.6s$ (c) SPH $t = 1.8s$ (d) Voronoi-SPH $t = 1.8s$ (e) SPH $t = 2.0s$ (f) Voronoi-SPH $t = 2.0s$

Figure 16.10: Comparison between the results given by the SPH (left) and Voronoi-SPH (right) methods from $t = 1.6s$ to $t = 2.0s$.

During the impact, the Voronoi volumes suffer from instantaneous compressions and they react to this compression by having their pressures elevated. Even if the Voronoi-FVM operators are more precise than the SPH ones, they are much more sensible to compression. These compressions induce volume changes and abrupt adjustments in the volume connectivities, some facets may disappear and other may gain in importance in the total sum of the discrete operator. All these factors result in a higher level of noise for Voronoi-SPH. SPH is better suited to predict these impacts as its operators, due to an elevated number of neighboring particles per particle and to the absence of a mesh, describe more continuously these compressions phases. Figure 16.11 shows for some time instants the SPH and the FVM regions of the fluid domain for $\Delta x = 0.01m$. At $t = 1.2s$ and $t = 1.8s$ we can see that some regions far from the free-surface are considered as being SPH particles when they should not. This issue is due to some difficulties when calibrating the free-surface detection algorithm (cf. 6.2.2.2) and can be considered as the source of some (only small amount) of the pressure noises verified in figure 16.6.

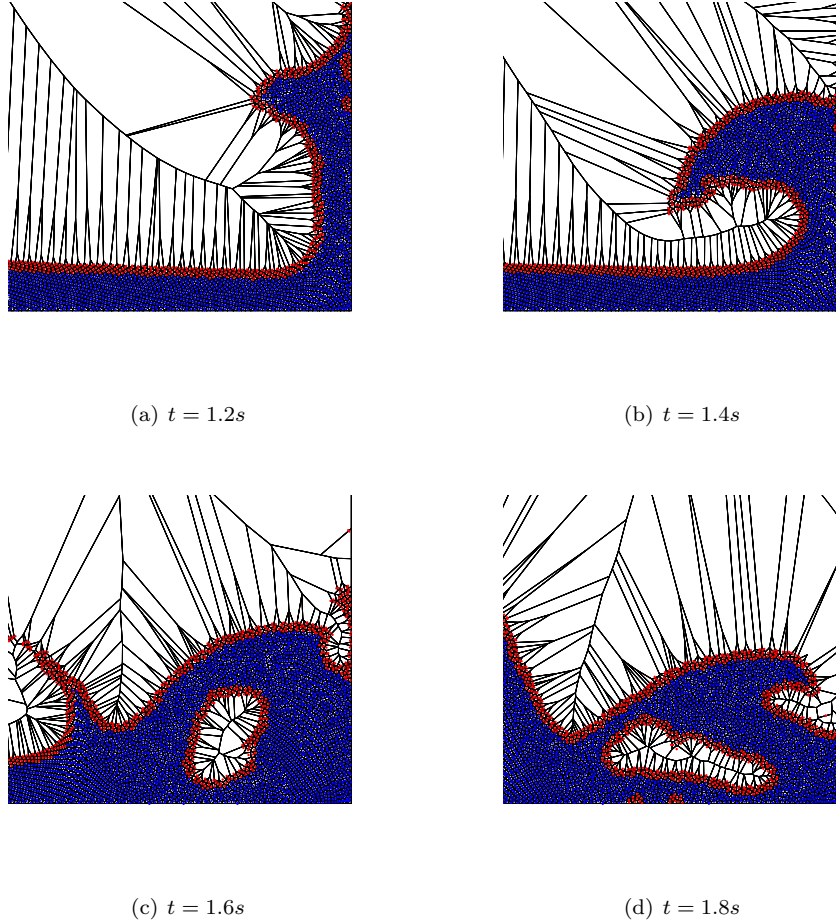


Figure 16.11: Plotting of the different regions for the dam-break problem using $\Delta x = 0.01m$: red particles are those that advance in time using an SPH scheme and the rest is evolved using the Voronoi-FVM formalism.

We have also proceeded to another study using this test case. As stated in section 15.1.1 we have two options for the discrete finite volume operators: either we consider the Voronoi-based differential operators (equations (15.6) and (15.7)) or the classical ones (equations (15.8) and (15.9)). These operators are used for computing the volume time derivative and the gradients to be used in the MUSCL scheme. Note, also, that it is possible to use the SPH typical operators (and operators only but keeping the Finite Volumes integral for the flux exchange) to compute these derivatives. Based on this idea, we proceeded to a study where we gradually migrated from an almost full-SPH operators (for the Voronoi sub-domain) simulation to a full-Voronoi operators one (passing by using the classical FVM operators). Several combinations were tested and for most of them, at a certain time instant, the solution of the Riemann problem on a volume-to-volume facet failed to converge due to instabilities created in the flow. The simulations that lasted longer were the ones closest to a full-Voronoi description of the flow (results that were analyzed in the previous paragraphs). The simulations that mixed SPH and FVM (Voronoi-based or not) operators for the Voronoi sub-domain failed very quickly, even before the impact against the wall for some configurations. As general remark for these results, we may say that a consistent formulation must be used for the Voronoi sub-domains of the Voronoi-SPH scheme, where the operators that are used must follow the assumption done to discretize the media.

Taking as basis the equality of equation (16.1) it is possible to express equivalent SPH interaction interfaces which together form an equivalent SPH ‘cell’ for each particle. Using the dam-break test case, we plotted this equivalent SPH mesh for a certain particle i in order to have a better vision on the correlation stated in equation (16.1). Figure 16.12 shows both the Voronoi mesh and the equivalent SPH ‘mesh’ in red at $t = 0.0s$ and $t = 0.7s$ for a particle i far enough from the free-surface and from the domain borders. When the particle distribution follows a Cartesian lattice the Voronoi mesh is also Cartesian (see $t = 0.0s$ in figure 16.12). After a certain time, particles tend to a disordered distribution which results in a deformed Voronoi mesh, as shown in figure 16.12 at $t = 0.7s$. It is interesting to see that the SPH equivalent mesh follows, in a certain manner, the Voronoi one, i.e., there is always a red segment (or a SPH-SPH interaction) coincident with a Voronoi facet (at least those that represent the interactions with the closest particles). Moreover, it is possible to see that the SPH segment sizes grow in similar way as the Voronoi facets grow, that is to say that if a certain Voronoi facet becomes bigger the equivalent SPH segment also grows.

In figure 16.12 a region of the fluid domain where the particles follow a rather organized distribution pattern was analyzed. If we consider a more ‘perturbed’ region, where particles tend to a very non uniform distribution, a different behavior is observed. In figure 16.13 it is possible to verify that for certain situations there may not be an equivalent coincident SPH segment for every Voronoi facet. This is a typical situation that is complicated to be treated by the Voronoi-FVM method, where it starts lacking accuracy (cf. section 15.1.2) and which SPH, being a more flexible method, is able to handle more easily.

Lastly, as stated in section 16.2.2, in order to ensure a good rate of convergence for the discrete operators in equations (15.6) and (15.7), it is necessary to ensure a certain mesh regularity throughout the simulation. We proposed the use of an arbitrary velocity (ALE formulation) based on the SPH method which, as showed, results in a much more regular grid. However, its effect in the Voronoi-SPH coupling could not be measured in the context of this thesis. Hence, the results presented previously

do not take into account the use of such an arbitrary velocity.

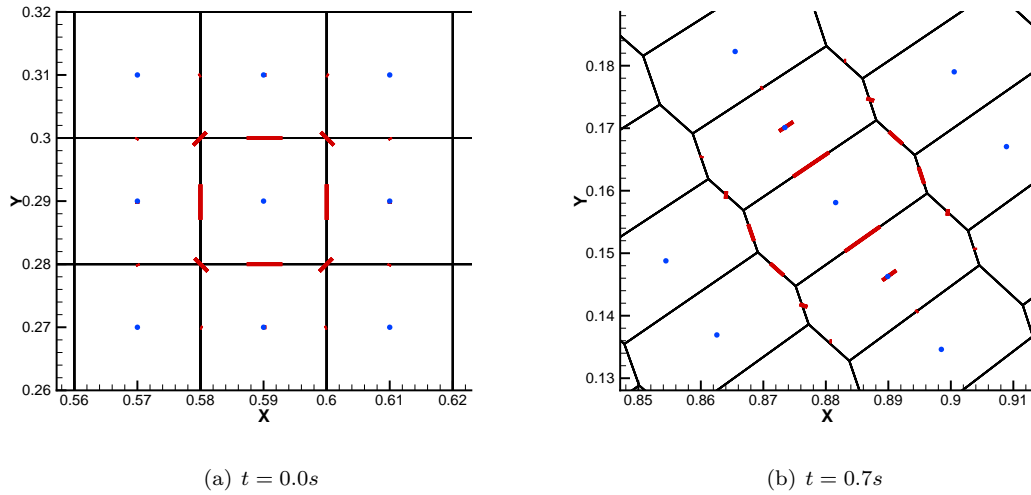


Figure 16.12: SPH equivalent mesh (red lines) for the Voronoi-SPH dam-break simulation at different time instants.

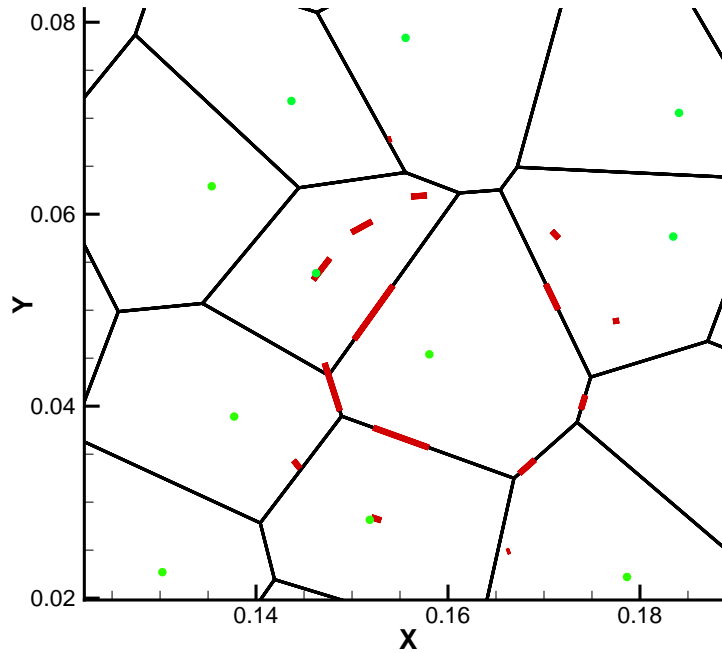


Figure 16.13: Situation where the equivalent SPH 'mesh' for the Voronoi volume is not totally found.

16.4.3 Gresho's Vortex

Given the problems evidenced by the last test case, the Gresho's vortex problem (found in [81]) used in [10] is used to compare SPH and Voronoi-FVM methods. The goal is to verify the differences given by the methods, numerical dissipation for instance, for the same test case without the presence of any coupling between the methods. In this test case a circular region on a fluid domain is initialized as a vortex where the initial (azimuthal) velocity profile is given by:

$$u_\phi(r) = \begin{cases} 5r & \text{for } 0 < r < 0.2 \\ 2 - 5r & \text{for } 0.2 < r < 0.4 \\ 0 & \text{for } r \geq 0.4 \end{cases} \quad (16.3)$$

where r is the radial distance to the center of the vortex. The same initial particle/mesh resolution was considered for both the SPH and Voronoi-FVM simulations, that is $\Delta x = 0.02m$ and the sound speed taken as $c_0 = 10 \text{ m/s}$ for both methods. Figure 16.15 shows the velocity magnitude both for SPH and Voronoi-FVM at different time instants (up to $t = 1.00s$) and shows that the latter one is slightly less dissipative than the first, which was an expected result due to the higher order interpolation of FVM with respect to SPH. This is confirmed by figure 16.14 which shows the maximum velocity decay on the entire domain for the whole simulation where it is possible to see that SPH is more dissipative than Voronoi-FVM.

We also show in figure 16.16 the mesh deformation for the Voronoi-FVM solver at $t = 1.00s$. It is possible to see that in the region with the highest velocities the mesh deformation tends to a more random formation which may lead to lower order operators. However, even when the mesh is already well distorted and disorganized ($t \geq 0.7$) the Voronoi-FVM is less dissipative than SPH. These results lead us to state that the small dissipation verified for the Hybrid-SPH method in the dam-break simulation when comparing a full-SPH simulation is (as expected) not caused by the Voronoi-FVM itself but rather by the coupling procedure.

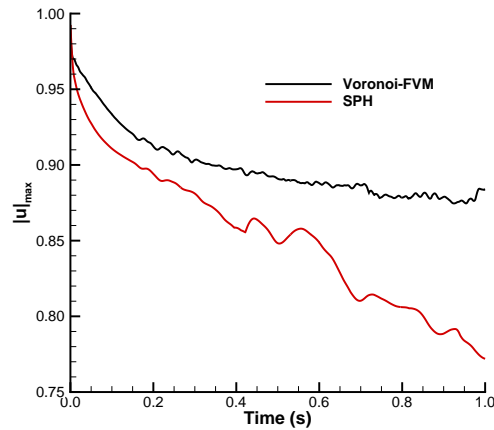


Figure 16.14: Decay of the maximum velocity on the domain for the SPH and Voronoi-FVM simulations.

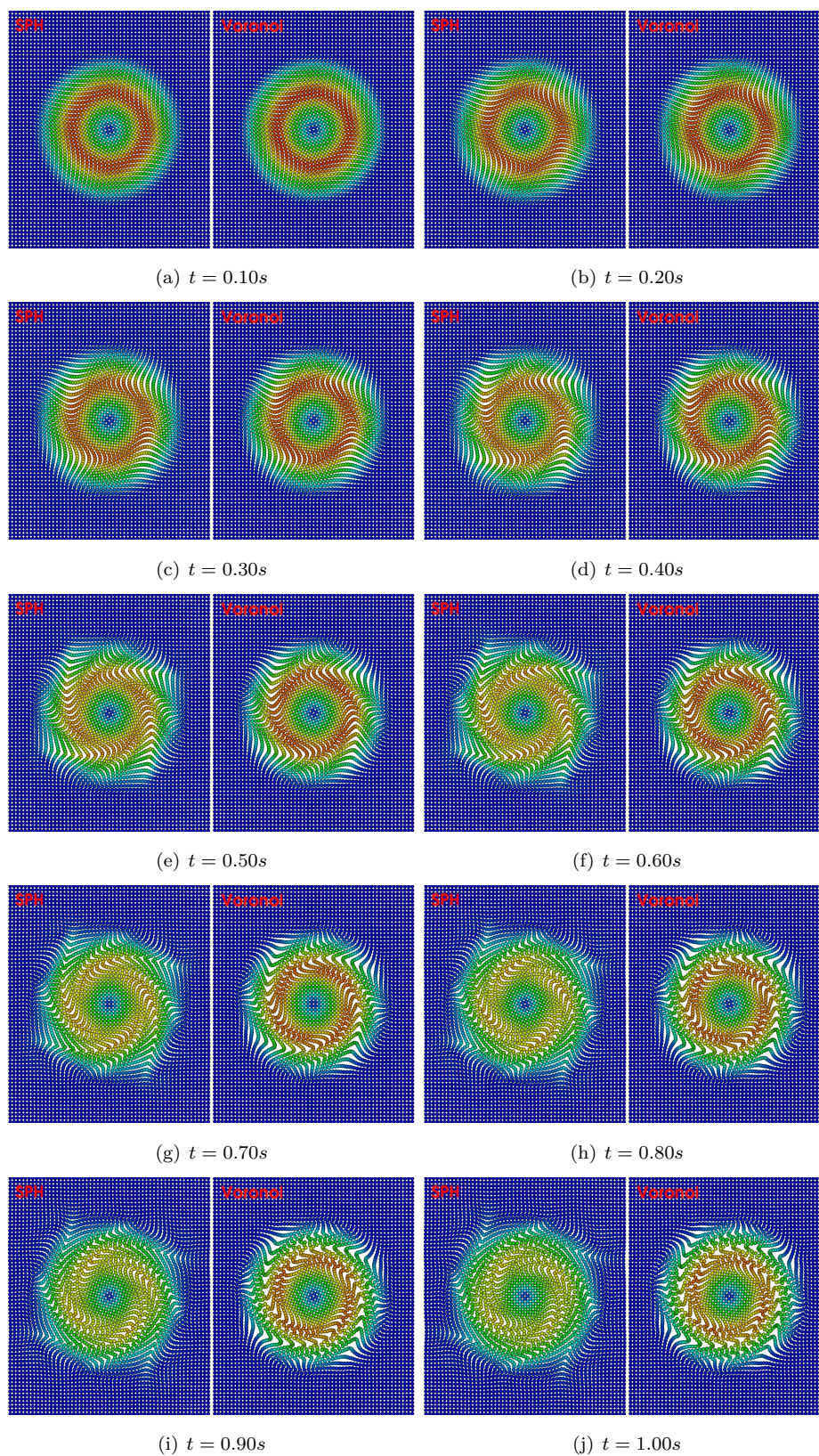


Figure 16.15: Comparison between the SPH and Voronoi-FVM results for the Gresho vortex problem

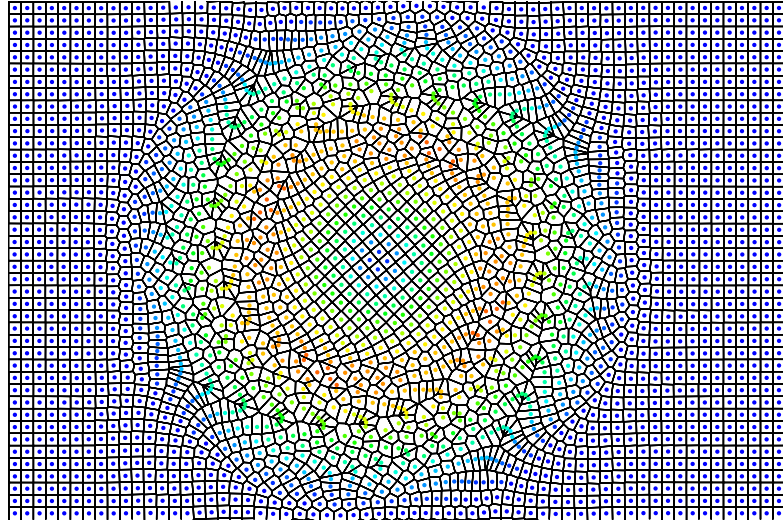


Figure 16.16: Mesh deformation at $t = 1.00$ s for the Voronoi-FVM simulation.

Chapter 17

Discussion

A FVM-SPH coupled method has been presented where the main goal was to verify the possibility of coupling a more precise numerical method to SPH, in order to have a method that benefits from the advantages of both techniques. A hybrid method raised from this coupling: Voronoi-SPH. Within this method, close to the free-surface a SPH formalism is used and for the rest of the fluid domain FVM is used. Considering the centered feature of SPH, we decided to use a Finite Volume algorithm based on the domain discretization using Voronoi geometric entities. This tessellation can easily adapt itself to a random distribution of points which facilitated the coupling, and particles can easily transit from SPH to FVM formalism without any particular treatment.

So far, the Voronoi-SPH method introduced in this thesis was tested for two test cases: the first considered a tank with a flat horizontal free-surface under the action of the gravity, and the second was the classical 2D dam-break benchmark. Considering the first test case, results show that, for a short period of time, the coupling is capable of maintaining a hydrostatic pressure profile. However, due to the fact that the SPH-FVM coupling interface is non-conservative, some instabilities appear in the free-surface and propagate towards the bottom of the domain resulting in an unstable simulation. As for the dam-break simulation, the results are more encouraging. A fair agreement is verified between the new Voronoi-SPH and the full-SPH results, both for the shape of the free-surface and for the pressure probe signals, even if for the Voronoi-SPH presents a higher level of noise. In order to better understand the coupling, we have also performed a comparison between the SPH and the Voronoi-FVM methods without the presence of the free-surface and without coupling them: Gresho's vortex. It showed that Voronoi-FVM is less dissipative than SPH even when its operators start lacking of precision.

A more thorough analysis is still necessary to have a better appreciation of the method limitations and capabilities. However, with these two simple test cases and arising analysis, it was possible to verify that such a coupling is possible. Some directions of research and improvements can be drawn:

- Regarding the coupling between the SPH and the FVM methods, a conservative technique may be sought: as it is, the coupling does not ensure that the flux going into the SPH part of the fluid is the opposite to the one going into the FVM. An alternative has already been mentioned earlier in this manuscript, that is, to adapt the Normal Flux method proposed by De Leffe *et al.*

[44] and De Leffe [19] in order to correctly and conservatively insert into the SPH part the flux computed by the Voronoi-based discrete operators. All the key points for such an adaptation are ready, the FVM and the SPH solvers and the surface separating both regions of the domain are available at each time step through the Voronoi tessellation. We may also cite the coupling developed by Bouscasse *et al.* [117] as a possible candidate to perform a conservative coupling between SPH and FVM. What lacked in this thesis was time to develop and validate such an approach to treat the FVM-SPH interface.

- The second concerns the homogenization of the Voronoi volumes. As stated previously, to have a regular volume distribution increases the quality of the FVM interpolators and if nothing is done, volumes tend to irregular forms. We proposed to use an arbitrary velocity field to ensure the regularity of the mesh and also to prevent the abrupt changes in the volume connectivities during a simulation. Such a technique was tested and it results in a better mesh distribution, however it did not lead to more stable simulations. In future works, we recommend this aspect of the Voronoi-based Finite Volume formulation to be addressed as it is a key feature to ensure the stability, convergence and precision of the method.
- We may also mention the fact the the Voronoi-based Finite Volume operators used in this thesis have low order of convergence for unstructured grids. We may consider in the future the use of higher order FVM schemes as presented in [132].

More globally we can state that proof of concept is made of the idea of coupling SPH to treat the free-surface to a mesh-based method to bring accuracy elsewhere. We now understand that the Lagrangian Voronoi-FVM was maybe in the end not the simplest candidate to make this proof of concept since Voronoi-FVM is tricky. Besides, test cases where both slow dynamics and a free-surface are present will have to be tested to better assess the interest of such a coupling.

Part V

Final Considerations and Perspectives

This manuscript gathers the different improvements to the Smoothed Particle Hydrodynamics method developed during the three years of a Ph.D. thesis performed in the LHEEA (Laboratoire de recherche en Hydrodynamique, Énergétique et Environnement Atmosphérique) laboratory. The proposed improvements enhanced the efficiency and the precision of SPH, which was the main goal of this thesis. During these three years we could also gather more information concerning the performance and behavior of the method, which may help the research community for future works.

After establishing the state of the art of the Smoothed Particle Hydrodynamics method in the LHEEA laboratory and in the literature, three main directions of improvement were traced to be discussed in this thesis. The first one considered trying a different approach for the Navier-Stokes equations than the traditional weakly-compressible assumption. For that matter, we studied the Incompressible Smoothed Particle Hydrodynamics method as presented in the literature. This method was seen as an alternative to the Weakly-Compressible SPH, capable of giving the same quality of results as the precise Riemann-SPH variant, and of overcoming the CFL limitation imposed by the acoustic part of the compressible Navier-Stokes equations. This method was analyzed and it was concluded that, ISPH, as it is presented in the literature, is capable of giving good-quality results comparable to Riemann-SPH, but time steps cannot be increased to higher levels than the ones used in WCSPH and thus resulted often in worse CPU cost to precision ratio. Nonetheless, the quality of results achieved by ISPH encouraged the search of a way to overcome the CPU cost limitation related to inverting a matrix (required by the PPE). In this thesis, we proposed a new fully-explicit approach to solve such an equation which circumvents this problem and ensures a good quality of results with the tiny time steps used in SPH. Moreover, contrary to the semi-implicit ISPH, the Explicit variant does not require the dynamic free-surface condition to be imposed prior to solve of the PPE, which makes the algorithm to be much simpler. This method was validated against a series of academic and industrial test cases. For all these test cases, the new Explicit ISPH provided good-quality results (comparable to Riemann-SPH and δ -SPH) in ensuring a lower computational cost than any other physically-reliable SPH algorithm. This new method improved substantially the ‘quality of results - CPU costs’ compromise of the Smoothed Particle Hydrodynamics method.

The second line of research developed in this thesis aimed to study the way by which the fluid domain is discretized using SPH. The particle splitting technique proposed and studied by some authors in the literature was studied in this thesis. It allows to adapt the particle distribution to the flow characteristics wherever and whenever desired. Up to the beginning of this thesis, this refinement technique had not been applied to Riemann-SPH or δ -SPH, nor had it been applied to flow phenomena typically treated by SPH, namely violent flows. In a first time, such a validation process was performed where good results were achieved. In a second time, a new, versatile and easy to implement particle derefinement technique was proposed, which has shown to be efficient and well suited to the Lagrangian nature of SPH. It permitted to treat a wide range of flow phenomena: dam-break, impact flows (jet on a flat plate), flow around bodies, etc. It is now possible, with this new particle refinement/derefinement technique, to adapt the particle distribution not only by increasing the number of particles but also by decreasing it which improves the efficiency of SPH simulations. It brings the SPH method close to mesh-based methods which have the capability of using adaptive mesh refinement (AMR) to adapt on

the fly the mesh to the needs of the simulation.

Despite the good results and advances in efficiency and quality of results achieved by these two first directions of improvement, they did not consider any change in the SPH discrete operators, and because of that, the well-known problems related to them as the lack of convergence and precision, numerical dissipation, etc, are still present. Several improvements (corrective terms, new kernel functions) have been proposed trying to mend these problems but they were not totally successful. Somehow, the development evolution of the SPH method will pass by a severe change in its operators, possibly by completely changing them. In this thesis, we studied such an approach by creating a Hybrid-SPH method that couples the Smoothed Particle Hydrodynamics to a Finite Volume solver. The SPH description of the media is used where it performs better, close to the free-surface and in the rest of the domain SPH is replaced by a more precise numerical method, here a Finite Volume approach. This coupling raised from the equivalence between the operators of both methods traced by Vila [25]. The results were rather encouraging and we could conclude that such a coupling is possible. Some problems on this coupling, however, still reside, for instance the treatment given to the FVM-SPH interface which is not numerically conservative and the FVM method used in this thesis presents some difficulties regarding the homogeneity of its cell volumes. Possible solutions for these problems were briefly described (and partially tested) in this thesis.

All these works allowed a better understanding of the Smoothed Particle Hydrodynamics method and opened the path for future developments. The new Explicit ISPH method development can be pursued by extending its application to wave propagation phenomena, and other slow dynamics flows, allowing that way to verify its limitations. It can also be advised to verify with more precision (possibly through a mathematical analysis) the CFL limitation of such a technique. Moreover, one could apply some developments proposed for the semi-implicit ISPH to the explicit variant aiming to improve even more the efficiency and precision of the latter one. Concerning the particle refinement/derefinement technique, several outcomes can be foreseen. Firstly, it could be adapted to different (spatial) refinement criteria and therefore be extended to more flow phenomena, multi-fluid simulations for example where refinement is necessary close to the interface between fluids. Secondly, the pressure peaks observed in some simulations could be studied in detail, giving continuity to the analysis started in this thesis (with the proposal of a transition zone). A third line of research was opened in this thesis proposing the development of a Hybrid-SPH algorithm. The particular method developed in this thesis (Voronoi-SPH) can be continued or may serve as basis to the creation of other algorithms that propose to mix SPH with a mesh-based method. Parallel to this thesis, some authors have started the development of similar techniques [117, 143], showing that such Hybrid-SPH algorithms start to become a new trend in the SPH community.

Finally, we cite the works produced during this thesis that are listed (in chronological order) in the appendix of this thesis, after the bibliography under the “Associated publications” section: [1, 2, 3, 4, 5, 6, 7, 8].

Bibliography

- [1] S. Marrone, A. Colagrossi, D. L. Touzé, G. Graziani, Fast free-surface detection and level-set function definition in SPH solvers, *Journal of Computational Physics* 229 (2010) 3652–3663.
- [2] U. Ghia, N. Ghia, C. T. Shin, High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method, *Journal of Computational Physics* 48 (1982) 387–411.
- [3] D. J. Tritton, Experiments on the flow past a circular cylinder at low Reynolds, *Journal of Fluids Mechanics* 6 (1959) 547.
- [4] M. Wu, C. Wen, R. Yen, M. Weng, A. Wang, Experimental and numerical study of the separation angle for flow around a circular cylinder at low Reynolds number, *Journal of Fluid Mechanics* 515 (2004) 233–260.
- [5] A. Souto-Iglesias, E. Botia-Vera, Sloshing wave impact problem, SPHERIC - SPH European Research Interest Community Website.
- [6] J. Feldman, J. Bonet, Dynamic refinement and boundary contact forces in SPH with applications in fluid flow problems, *Int. J. Numer. Meth. Engng* 72 (2007) 295–324.
- [7] J. Feldman, Dynamic refinement and boundary contact forces in smoothed particle hydrodynamics with applications in fluid flow problems, Ph.D. thesis, University of Wales, School of Engineering, Swansea, 2006.
- [8] Y. R. Lopez, D. Roose, C. R. Morfa, Dynamic particle refinement in SPH: application to free surface flow and non-cohesive soil simulations, *Comput Mech* 51 (2013) 731–741.
- [9] C. Khaddaj-Mallat, Design of experiments approach to the flooding of damaged ships, Ph.D. thesis, Ecole Centrale de Nantes, 2010.
- [10] V. Springel, E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh, *Monthly Notices of the Royal Astronomical Society* 401 (2009) 791–851.
- [11] L. Lucy, Numerical approach to the testing of the fission hypothesis, *Astron. J.* 82 (1977) 1013–1024.
- [12] R. Gingold, J. Monaghan, Smoothed particle hydrodynamics - theory and application to non-spherical stars, *Monthly Notices of the Royal Astronomical Society* 181 (1977) 375–389.

- [13] J. J. Monaghan, Simulating free surface flows with SPH, *Journal of Computational Physics* 110 (1994) 399–406.
- [14] M. Doring, Developpement d’une methode SPH pour les applications à surface libre en hydrodynamique, Ph.D. thesis, Ecole Centrale de Nantes, 2005.
- [15] G. Oger, Aspects theoriques de la methode SPH et applications a l’hydrodynamique a surface libre, Ph.D. thesis, Ecole Centrale de Nantes, 2006.
- [16] J.-B. Deuff, Extrapolation au réel des mesures de pressions obtenues sur des cuves modèle réduit, Ph.D. thesis, Ecole Centrale de Nantes, 2007.
- [17] P.-M. Guilcher, Contribution au développement d’une méthode SPH pour la simulation numérique des interactions houle-structure, Ph.D. thesis, Ecole Centrale de Nantes, 2008.
- [18] N. Grenier, Modélisation numérique par la méthode SPH de la separation eau-huile dans les séparateurs gravitaires, Ph.D. thesis, Ecole Centrale de Nantes, 2009.
- [19] M. de Leffe, Modelisation d’écoulements visqueux par methode SPH en vue d’application a l’hydrodynamique navale, Ph.D. Thesis, Ecole Centrale de Nantes, France, 2011.
- [20] J. Zhao, Development of a fast SPH model for non linear shallow water flows: application to coastal flooding and dam breaking, Ph.D. thesis, Ecole Centrale de Nantes, 2012.
- [21] G. Fourey, Développement d’une méthode de couplage fluid-structure SPH-Eléments Finis en vue de son application à l’hydrodynamique navale, Ph.D. thesis, Ecole Centrale de Nantes, 2012.
- [22] G. Capdeville, Modelisation numerique d ecoulements compressibles, parties I et II, Lecture Notes on Ecole Centrale de Nantes, 1990.
- [23] P. Tait, Report on some of the physical properties of fresh water and sea water, *Physical Chemistry* 2 (1888) 1–76.
- [24] H. Wendland, Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, *Advances in Computational Mathematics* 4 (1995) 389–396.
- [25] J. P. Vila, On particle weighted methods and SPH, *Mathematical Models and Methods in Applied Sciences* 9 (1999) 161–210.
- [26] B. B. Moussa, Analyse numérique de méthodes particulières régularisées de type SPH pour les lois de conservation, Ph.D. thesis, Université Paul Sabatier, Toulouse, 1998.
- [27] S. Mas-Gallic, P. Raviart, A particle method for first-order symmetric systems, *Numer. Math.* 51 (1987) 323–352.
- [28] J.-L. Lacome, Analyse de la méthode particulière SPH, Applications à la détonique, Ph.D. thesis, INSA Toulouse, 1998.
- [29] D. le Touzé, Developpement d’un modele SPH tri-dimensionnel applique aux interactions fluide-structure en hydrodynamique navale, Activity Report DGA, INSEAN and ECN, 2005.

- [30] P. W. Randle, L. D. Libersky, Smoothed particle hydrodynamics: Some recent improvements and applications, *Comput. Methods Appl. Mech. Engrg.* 139 (1996) 375–408.
- [31] S. Marrone, Enhanced SPH modeling of free-surface flows with large deformations, Ph.D. thesis, Università di Roma Sapienza, 2011.
- [32] J. J. Monaghan, D. J. Price, Variational principles for relativistic smoothed particle hydrodynamics, *Monthly Notices of the Royal Astronomical Society* 328 (2001) 381–392.
- [33] J. J. Monaghan, Smoothed particle hydrodynamics, *Rep. Prog. Phys.* 68 (2005) 1703–1759.
- [34] M. Antuono, A. Colagrossi, S. Marrone, D. Molteni, Free-surface flows solved by means of SPH schemes with numerical diffusive terms, *Computer Physics Communications* 181 (2010) 532–549.
- [35] A. Colagrossi, M. Landrini, Numerical simulation of interfacial flows by smoothed particle hydrodynamics, *Journal of Computational Physics* 191 (2003) 448–475.
- [36] A. Colagrossi, A Meshless Lagrangian Method for Free Surface and Interface Flows with Fragmentation, Ph.D. thesis, Università di Roma Sapienza, 2005.
- [37] D. Molteni, A. Colagrossi, A simple procedure to improve the pressure evaluation in hydrodynamic context using the SPH, *Comp. Phys. Comm.* 180 (2009) 861–872.
- [38] J. Morris, P. Fox, Y. Zhu, Modeling low reynolds number incompressible flows using SPH, *Journal of Computational Physics* 136 (1997) 214–226.
- [39] J. J. Monaghan, R. A. Gingold, Shock simulation by the particle method SPH, *Journal of Computational Physics* 52 (1983) 374–389.
- [40] A. Colagrossi, M. Antuono, A. Souto-Iglesias, D. L. Touzé, Theoretical analysis and numerical verification of the consistency of viscous smoothed-particle-hydrodynamics formulations in simulating free-surface flows, *Phys. Rev. E* 84 (2011) 026705.
- [41] A. Colagrossi, M. Antuono, D. L. Touzé, Theoretical considerations on the free-surface role in the smoothed-particle-hydrodynamics model, *Phys. Rev. E* 79 (2009) 056701.
- [42] J. Marongiu, Méthode numérique lagrangienne pour la simulation d’écoulements à surface libre. Application aux turbines Pelton, Ph.D. thesis, Ecole Centrale de Lyon, 2007.
- [43] J. Marongiu, Riemann solvers and efficient boundary treatments: an hybrid SPH-finite volume numerical method, in: *Proceedings of the 3rd International SPHERIC workshop*, 2008.
- [44] M. de Lefte, D. le Touzé, B. Alessandrini, Normal flux method at the boundary for SPH, in: *Proceedings of the 4th International SPHERIC workshop*, 2008.
- [45] L. Libersky, A. Petschek, T. Carney, J. Hipp, F. Allahdadi, High Strain Lagrangian Hydrodynamics: A Three-dimensional SPH Code for Dynamic Material Response, *Journal of Computational Physics* 109 (1993) 67–75.

- [46] A. Colagrossi, M. Antuono, S. Marrone, A 2D+t SPH model with enhanced solid boundary treatment, in: Proceedings of the 4th International SPHERIC workshop, 2009.
- [47] M. J. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, *Journal of Computational Physics* 82 (1998) 64–84.
- [48] T. Plewa, T. J. Linde, V. G. Weirs, *Adaptive Mesh Refinement, Theory and Applications*, Springer, 2005.
- [49] A. Colagrossi, B. Bouscasse, M. Antuono, S. Marrone, Particle packing algorithm for SPH schemes, *Computer Physics Communications* 183 (2012) 1641–1653.
- [50] S. Marrone, A. Colagrossi, M. Antuono, G. Colicchio, G. Graziani, An accurate SPH modeling of viscous flows around bodies at low and moderate reynolds numbers, *Journal of Computational Physics* In Press.
- [51] S. Cummins, M. Rudman, An SPH projection method, *Journal of Computational Physics* 152 (1999) 584–607.
- [52] A. Marsh, G. Oger, D. le Touzé, D. Guibert, Validation of a conservative variable-resolution SPH scheme including gradient-h terms, in: Proceedings of the 6th International SPHERIC workshop, 2011.
- [53] V. Springel, L. Hernquist, Comological smoothed particle hydrodynamics simulations: the entropy equation, *Monthly Notices of the Royal Astronomical Society* 333 (2002) 649–664.
- [54] E. Oñate, S. R. Idelsohn, F. del Pin, R. Aubry, The particle finite element method. an overview, *International Journal Computational Method* 1(2) (2004) 267–307.
- [55] R. Nestor, M. Basa, N. Quinlan, B. Iwanowski, B. Buchner, Moving boundary problems in the finite volume particle method, in: Proceedings of the 3rd International SPHERIC workshop, 2008, pp. 109–114.
- [56] A. J. Chorin, Numerical solution of the Navier-Stokes equations, *Mathematics of Computation* 22 (1968) 745–762.
- [57] S. Shao, E. Y. M. Lo, Incompressible SPH method for simulating newtonian and non-newtonian flows with a free surface, *Advances in Water Resources* 26 (2003) 787–800.
- [58] E.-S. Lee, C. Moulinec, R. Xu, D. Violeau, D. Laurence, P. Stansby, Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method, *Journal of Computational Physics* 227 (2008) 8417–8436.
- [59] E. S. Lee, D. Violeau, R. Issa, S. Ploix, Application of weakly compressible and truly incompressible SPH to 3-D water collapse in waterworks, *Journal of Hydraulic Research* 48 (2010) 50–60.

- [60] S. J. Lind, R. Xu, P. K. Stansby, B. D. Rogers, Incompressible smoothed particle hydrodynamics for free-surface flows: a generalised diffusion-based algorithm for stability and validations for impulsive flows and propagating waves, *Journal of Computational Physics* 231 (2012) 1499–1523.
- [61] S. Shao, Incompressible SPH simulation of wave breaking and over toppling with turbulence modelling, *International Journal for Numerical Methods in Fluids* 50 (2006) 597–621.
- [62] R. Xu, P. K. Stansby, D. Laurence, Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach, *Journal of Computational Physics* 228 (2009) 6703–6725.
- [63] X. Hu, N. Adams, An incompressible multi-phase SPH method, *Journal of Computational Physics* 227 (2007) 264–278.
- [64] X. Hu, N. Adams, A constant-density approach for incompressible multi-phase SPH, *Journal of Computational Physics* 228 (2009) 2082–2091.
- [65] R. Xu, P. K. Stansby, The influence of the truncated kernel to free-surface predictions with ISPH and a new solution, in: *Proceedings of the 5th International SPHERIC workshop, 2010*, pp. 130–137.
- [66] M. S. Shadloo, A. Zainali, S. H. Sadek, M. Yildiz, Improved Incompressible Smoothed Particle Hydrodynamics method for simulating flow around bluff bodies, *Comput. Methods Appl. Mech. Engrg.* 200 (2011) 1008–1020.
- [67] A. Khayyer, H. Gotoh, S. Shao, Corrected incompressible SPH method for accurate water-surface tracking in breaking wave, *Coastal Engineering* 55 (2008) 236–250.
- [68] A. Khayyer, H. Gotoh, S. Shao, Enhanced predictions of wave impact pressure by improved incompressible SPH methods, *Applied Ocean Research* 31 (2009) 111–131.
- [69] S. Koshizuka, Y. Oka, H. Tamako, A particle method for calculating splashing of incompressible viscous fluid, *International Conference, Mathematics and Computations, Reactor Physics, and Environmental Analyses* 2 (1995) 1514–1521.
- [70] S. Koshizuka, Y. Oka, Moving-particle semi-implicit method for fragmentation of incompressible fluid, *Nuclear Science and Engineering* 123 (3) (1996) 421–434.
- [71] A. Khayyer, H. Gotoh, A higher order laplacian model for enhancement and stabilization of pressure calculation by the MPS method, *Applied Ocean Research* 32 (2010) 124–131.
- [72] A. Bømann, O. Shipilova, G. Skeie, Incompressible SPH for free surface flows, *Computer and Fluids* 67 (2012) 138–151.
- [73] J. Zhou, Q. Ma, L. Z. S. Yan, Numerical Investigation of Violent Wave Impact on Offshore Wind Energy Structures using MLPG R Method, in: *Proceedings of the Nineteenth (2009) International Offshore and Polar Engineering Conference, 2009*.

- [74] M. de Leffe, D. le Touzé, B. Alessandrini, A modified no-slip condition in weakly-compressible SPH, in: Proceedings of the 6th International SPHERIC workshop, 2011, pp. 291–297.
- [75] A. Khayyer, H. Gotoh, Modified moving particle semi-implicit methods for the prediction of 2D wave impact pressure, *Coastal Engineering* 56 (2009) 419–440.
- [76] B. Ataie-Ashtiani, G. Shobeyri, Numerical simulation of landslide impulsive waves by incompressible smoothed particle hydrodynamics, *International Journal for Numerical Methods in Fluids* 56 (2008) 209–232.
- [77] H. F. Schwaiger, An implicit corrected SPH formulation for thermal diffusion with linear free surface boundary conditions, *International Journal for Numerical Methods in Engineering* 75 (2008) 647–671.
- [78] S. J. Lind, R. Xu, P. K. Stansby, B. D. Rogers, A Stabilising Diffusion-Based Shifting Algorithm for Incompressible Smoothed Particle Hydrodynamics, in: Proceedings of the 6th International SPHERIC workshop, 2011, pp. 14–20.
- [79] J. J. Monaghan, On the problem of penetration in particle methods, *Journal of Computational Physics* 82 (1989) 1–15.
- [80] P. M. Gresho, On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. part 1: Theory, *International Journal for Numerical Methods in Fluids* 11 (1990) 587–620.
- [81] P. M. Gresho, S. T. Chan, On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. part 2: Implementation, *International Journal for Numerical Methods in Fluids* 11 (1990) 621–659.
- [82] P. M. Gresho, S. T. Chan, R. L. Lee, C. D. Upson, A modified finite element method for solving the time-dependent, incompressible navier-stokes equations. part 1: Theory, *International Journal for Numerical Methods in Fluids* 4 (1984) 557–598.
- [83] M. A. Christon, D. E. Carroll, Unstructured-grid, parallel, projection solver for computing low-speed flows, Technical Report.
- [84] J. G. Zhou, D. M. Causon, C. G. Mingham, D. M. Ingram, Numerical Prediction of Dam-Break Flows in General Geometries with Complex Bed Topography, *Journal of Hydraulic Engineering* 130 (2004) 332–340.
- [85] A. Souto-Iglesias, F. Macia, L. M. Gonzalez, J. L. Cercos-Pita, On the consistency of MPS, *Computer Physics Communications* 184 (2013) 732–745.
- [86] F. Macia, A. Souto-Iglesias, L. M. Gonzalez, J. Cercos-Pita, $\langle \text{MPS} \rangle \equiv \langle \text{SPH} \rangle$, in: Proceedings of the 8th International SPHERIC workshop, 2013.

- [87] A. Rafiee, K. P. Thiagarajan, An SPH projection method for simulating fluid-hypoelastic structure interaction, *Computer Methods in Applied Mechanics and Engineering* 198 (2009) 2785–2795.
- [88] S. Hosseini, M. Manzari, S. Hannani, A fully explicit three-step SPH algorithm for simulation of non-newtonian fluid flow, *International Journal of Numerical Methods for Heat & Fluid Flow* 17 (2007) 715 – 735.
- [89] M. Ellero, M. Serrano, P. Espanol, Incompressible smoothed particle hydrodynamics, *Journal of Computational Physics* 226 (2007) 1731–1752.
- [90] <http://nextmuse.cscs.ch>.
- [91] A. Roshko, On the development of turbelent wakes from vortex streets, NACA Rep. 1191, 1954.
- [92] B. Buchner, Green Water on Ship-type Offshore Structures, Ph.D. thesis Delft University of Technology, 2002.
- [93] K. M. T. Kleefsman, G. Fekken, A. E. P. Veldman, B. Iwanowski, B. Buchner, A Volume-of-Fluid based simulation method for wave impact problems, *Journal of Computational Physics* 206 (2005) 363–393.
- [94] A. Colagrossi, A. Souto-Iglesias, M. Antuono, S. Marrone, Smoothed-particle-hydrodynamics modeling of dissipation mechanisms in gravity waves, *Phys. Rev. E* 87 (2013) 023302.
- [95] M. Antuono, A. Colagrossi, The damping of viscous gravity waves, *Wave Motion* 50 (2013) 197–209.
- [96] F. Colin, R. Egli, F. Y. Lin, Computing a null divergence velocity field using smoothed particle hydrodynamics, *Journal of Computational Physics* 217 (2006) 680–692.
- [97] S. Marrone, M. Antuono, A. Colagrossi, G. Colicchio, D. le Touzé, G. Graziani, Delta-SPH model for simulating violent impact flows, *Computer Methods in Applied Mechanics and Engineering* 200 (2011) 1526–1542.
- [98] <http://www.sph-flow.com>.
- [99] G. Oger, E. Jacquin, J.-F. Sigrist, A validation test-case for SPH-based simulations of sloshing flows in cylindrical confinement, in: *Proceedings of the ASME Pressure Vessel and Piping division conference*, 2009.
- [100] P. M. Guilcher, L. Brosset, N. Couty, D. le Touzé, Simulations of breaking wave impacts on a rigid wall at two different scales with a two phase fluid compressible SPH mode, in: *Proceedings of The Twenty-second (2012) International Offshore and Polar Engineering Conference (ISOPE)*, 2012.
- [101] M. Antuono, B. Bouscasse, A. Colagrossi, C. Lugni, Two-dimensional modal method for shallow-water sloshing in rectangular basins, *Journal of Fluid Mechanics* 700 (2012) 419–440.

- [102] L. Delorme, A. S. Iglesias, S. A. Perez, Sloshing loads simulation in LNG tankers with SPH, in: International Conference on Computational Methods in Marine Engineering, MARINE 2005, 2005.
- [103] A. Colagrossi, G. Colicchio, C. Lugni, M. Brocchini, A study of violent sloshing wave impacts using an improved SPH method, *Journal of Hydraulic Research* 48 (2010) 94–2010.
- [104] R. P. Nelson, J. C. B. Papaloizou, Variable smoothing length and energy conservation in smoothed particle hydrodynamics, *Monthly Notices of the Royal Astronomical Society* 270 (1994) 1–29.
- [105] L. Hernquist, Some cautionary remarks on smoothed particle hydrodynamics, *The Astrophysical Journal* 404 (1993) 717–722.
- [106] R. A. Gingold, J. J. Monaghan, Kernel estimates as a basis for general particle methods in hydrodynamics, *Journal of Computational Physics* 46 (1982) 429–453.
- [107] J. M. Alimi, A. Serna, C. Pastor, G. Bernabeu, Smoothed particle hydrodynamics: importance of correction terms in adaptative resolution algorithms, *Journal of Computational Physics* 192 (2003) 157–174.
- [108] S. Rosswog, Astrophysical smoothed particle hydrodynamics, *New Astronomy Reviews* 53 (2009) 78–104.
- [109] J. M. Owen, J. V. Villumsen, P. R. Shapiro, H. Martel, Adaptative smoothed particle hydrodynamics: Methodology II, *Astrophys. J. Suppl. Ser.* 116 (1998) 155–209.
- [110] L. Hernquist, N. Katz, Treesph: A unification of SPH with the hierachical tree method, *Astrophys. J. Suppl. Ser.* 70 (1989) 419–446.
- [111] A. Marsh, G. Oger, C. Khaddaj-Mallat, D. le Touzé, Numerical predictions of ship flooding scenatios using SPH, in: Proceedings of the 5th International SPHERIC workshop, 2010.
- [112] M. Bate, I. Bonnell, N. Price, Modelling accretion in protobinary systems, *Monthly Notices of the Royal Astronomical Society* 277 (1995) 362–376.
- [113] S. Kitsionas, A. Whitworth, Smoothed particle hydrodynamics with particle splitting, applied to self-gravitating collapse, *Monthly Notices of the Royal Astronomical Society* 330 (2002) 129–136.
- [114] S. Kitsionas, A. Whitworth, High-resolution simulations of clump-clump collisions using sph with particle splitting, *Monthly Notices of the Royal Astronomical Society* 378 (2007) 507–524.
- [115] Z. Meglicki, D. Wickramasinghe, G. Bicknell, 3D structure of truncated accretion disks in close binaries, *Monthly Notices of the Royal Astronomical Society* 264 (1993) 691–704.
- [116] J. J. Monaghan, S. Varnas, The dynamics of interstellar cloud complexes, *Monthly Notices of the Royal Astronomical Society* 231 (1988) 515–534.

- [117] B. Bouscasse, S. Marrone, A. Colagrossi, A. D. Mascio, Multi-purpose interfaces for coupling SPH with other solvers, in: *Proceedings of the 8th International SPHERIC workshop*, 2013.
- [118] R. Vacondio, B. D. Rogers, P. K. Stansby, Accurate particle splitting for smoothed particle hydrodynamics in shallow water with shock capturing, *International Journal for Numerical Methods in Fluids* 69 (2012) 1377–1410.
- [119] P. Omidvar, P. K. Stansby, B. D. Rogers, Wave body interaction in 2D using smoothed particle hydrodynamics (SPH) with variable particle mass, *Numerical Methods in Fluids* 68 (2011) 686–705.
- [120] J. J. Monaghan, SPH and riemann solvers, *Journal of Computational Physics* 136 (1997) 298–307.
- [121] S. E. Hieber, P. Koumoutsakos, An immersed boundary method for smoothed particle hydrodynamics of self-propelled swimmers, *Journal of Computational Physics* 224 (2008) 8636–8654.
- [122] R. Vacondio, B. D. Rogers, P. K. Stansby, P. Mignosa, J. Feldman, Development of SPH variable resolution using dynamic particle coalescing and splitting, in: *Proceedings of the 7th International SPHERIC workshop*, 2012, pp. 347–354.
- [123] R. Vacondio, B. D. Rogers, P. K. Stansby, P. Mignosa, Shallow water SPH for flooding with dynamic particle coalescing and splitting, *Advances in Water Resources* 58 (2013) 10–23.
- [124] A. Khayyer, H. Gotoh, Enhancement of stability and accuracy of the moving particle semi-implicit method, *Journal of Computational Physics* 230 (2011) 3093–3118.
- [125] C. Khaddaj-Mallat, J.-L. Toularastel, J.-M. Rousset, P. Ferrant, Investigating the transient flooding and sloshing in internal compartments of an ITTC damaged Ro-Ro passenger ferry part I: experimental setup, *Proceedings of the ASME 2010 29th International Conference on Ocean, Offshore and Arctic Engineering*, OMAE2010.
- [126] C. Khaddaj-Mallat, F. Vadeboin, L. Davoust, B. Pettinotti, B. Alessandrini, J.-M. Rousset, P. Ferrant, Investigating the transient flooding and sloshing in internal compartments of an ITTC damaged Ro-Ro passenger ferry part II: experimental analysis, *Proceedings of the ASME 2010 29th International Conference on Ocean, Offshore and Arctic Engineering*, OMAE2010.
- [127] R. Whitehurst, A free lagrange method for gas dynamics, *Monthly Notices of the Royal Astronomical Society* 227 (1995) 655–680.
- [128] N. Y. Gnedin, Softened lagrangian hydrodynamics for cosmology, *Astrophysical Journal Supplement Series* 97 (1995) 231–257.
- [129] U.-L. Pen, Reconstructing nonlinear stochastic bias from velocity space distortions, *The Astrophysical Journal* 504 (1998) 1–15.
- [130] G. Xu, Hydrodynamic and n-body schemes on an unstructured, adaptive mesh with applications to cosmological simulations, *Monthly Notices of the Royal Astronomical Society* 288 (1997) 903–919.

- [131] R. Lobere, P.-H. Maire, M. Shashkov, J. Breil, S. Galera, ReALE: a reconnection-based arbitrary-Lagrangian-Eulerian method, *Journal of Computational Physics* 229 (2010) 4724–4761.
- [132] M. Dumbser, W. Boscheri, High-order unstructured lagrangian one-step weno finite volume schemes for non-conservative hyperbolic systems: Applications to compressible multi-phase flows (2013). [arXiv:arXiv:1304.4816](#).
- [133] S. Heß, Particle hydrodynamics with tessellation techniques, Ph.D. thesis, Ludwig Maximilian University of Munich, 2011.
- [134] S. Heß, V. Springel, E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh, *Monthly Notices of the Royal Astronomical Society* 406 (2010) 2289–2311.
- [135] V. Springel, Hydrodynamic simulations on a moving voronoi mesh (2011). [arXiv:arXiv:1109.2218](#).
- [136] M. Vogelsberger, D. Sijacki, D. Keres, V. Springel, L. Hernquist, Moving mesh cosmology: numerical techniques and global statistics, *Monthly Notices of the Royal Astronomical Society* 425 (2012) 3024–3057.
- [137] M. Vogelsberger, D. Sijacki, D. Keres, V. Springel, L. Hernquist, Simulations on a moving mesh: the clustered formation of population III protostars, *The Astrophysical Journal* 737:75.
- [138] R. Lobere, P.-H. Maire, M. Shashkov, ReALE: a reconnection-based arbitrary-Lagrangian-Eulerian method in cylindrical geometry, *Computers and Fluids* 46 (2011) 59–69.
- [139] P. Espanol, M. Serrano, Voronoi fluid particles & tessellation fluid dynamics, *Tessellations in the Sciences Virtues, Techniques and Applications of Geometric Tilings* 3(9) (2009) 42.
- [140] C. Barber, T. G. Center, Qhull, University of Minnesota, 1993-2012.
- [141] T. Barth, D. Jespersen, The design and application of upwind schemes on unstructured meshes, *AIAA* 89-0366.
- [142] C. Leroy, Developpement d’une methode volumes finis à flux caracteristiques pour la simulation d’écoulements en hydrodynamique, Ph.D. thesis, Ecole Centrale de Nantes, 2012.
- [143] M. Neuhauser, F. Leboeuf, J.-C. Marongiu, M. Rentschler, E. Parkinson, Coupling of a SPH-ALE and a Finite Volume Method, in: *Proceedings of the 8th International SPHERIC workshop*, 2013.

Associated publications

- [1] D. A. Barcarolo, D. L. Touzé, F. de Vuyst, Incompressible Smoother Particle Hydrodynamics: proposition and validation of a fully-explicit algorithm, in: Proceedings of the 7th International SPHERIC workshop, 2012.
- [2] D. L. Touzé, D. A. Barcarolo, M. Kerhuel, N. Quinlan, L. Lobovsky, M. Basa, J.-C. Marongiu, F. Leboeuf, J. Caro, A. Colagrossi, S. Marrone, M. de Leffe, P.-M. Guilcher, SPH benchmarking: a comparison of SPH variants on selected test cases within the NextMuSE initiative, in: Proceedings of the 7th International SPHERIC workshop, 2012.
- [3] D. A. Barcarolo, D. L. Touzé, F. de Vuyst, Validation of a new fully-explicit incompressible Smoothed Particle Hydrodynamics method, in: Proceedings of the 10th World Congress on Computational Mechanics, 2012.
- [4] D. A. Barcarolo, D. L. Touzé, F. de Vuyst, Presentation of an Explicit Incompressible Smoothed Particle Hydrodynamics algorithm and its validation to violent flows, in: Proceedings of the 2nd International Conference on Violent Flows, 2012.
- [5] D. L. Touzé, D. A. Barcarolo, M. Kerhuel, N. Quinlan, L. Lobovsky, M. Basa, J.-C. Marongiu, F. Leboeuf, J. Caro, A. Colagrossi, S. Marrone, M. de Leffe, P.-M. Guilcher, Investigation of SPH variants for violent flows simulations within the NextMuSE initiative, in: Proceedings of the 2nd International Conference on Violent Flows, 2012.
- [6] D. L. Touzé, D. A. Barcarolo, M. Kerhuel, G. Oger, N. Grenier, N. Quinlan, L. Lobovsky, M. Basa, J.-C. Marongiu, F. Leboeuf, J. Caro, A. Colagrossi, S. Marrone, M. de Leffe, P.-M. Guilcher, Smoothed Particle Hydrodynamics: benchmarking on selected test cases within the NextMuSE initiative, in: Proceedings of the ASME 2013 32th International Conference on Ocean, Offshore and Arctic Engineering, OMAE2013, 2013.
- [7] D. A. Barcarolo, D. L. Touzé, G. Oger, F. de Vuyst, Particle refinement and derefinement procedure applied to the Smoothed Particle Hydrodynamics method, in: Proceedings of the 8th International SPHERIC workshop, 2013.
- [8] D. A. Barcarolo, D. L. Touzé, G. Oger, F. de Vuyst, Assessment of SPH method variants for sloshing problems: comparison to experimental results, in: Proceedings of the 5th International Conference on Computational Method in Marine Engineering, MARINE 2013, 2013.