



UNIVERSITÀ DI PARMA
UNIVERSITÀ DEGLI STUDI DI
PARMA

DOTTORATO DI RICERCA IN INGEGNERIA E ARCHITETTURA
XXXIV CICLO

**Advances in particle distribution for SPH
numerical schemes:
from explicit to implicit shifting techniques**

Coordinatore:

Chiar.mo Prof. Sandro Longo

Tutore:

Chiar.mo Prof. Renato Vacondio

Co-Tutore:

Dr. Jean-Christophe Marongiu

Dottorando: Pietro Rastelli

Anni accademici 2018/2019-2020/2021

Abstract

In meshless numerical methods such as Smoothed Particle Hydrodynamics (SPH), the lack of uniformity in particle distribution, which is manifested as the presence of voids or clusters, negatively affects the accuracy. In these models, the particles follow the Lagrangian trajectories, and for highly distorted flows, their distribution is severely perturbed, generating numerical issues and compromising the quality of the simulations. For this reason, methodologies, called Particle Shifting Technique (PST), have been introduced to reduce these phenomena. The PSTs presented in the literature have an explicit approach meaning that it is not possible to impose a maximum predefined level of perturbation in the particle distribution.

In the present thesis, an explicit shifting technique has been extended and optimized in the framework of Arbitrarily Lagrangian-Eulerian SPH (ALE-SPH) schemes, increasing the accuracy without extra computational overheads. Then, a novel approach for particle shifting, which can be adopted in meshless numerical methods, has been developed and analyzed. The proposed methodology, called Implicit Iterative Particle Shifting (IIPS), uses an iterative procedure to reduce the spatial particle anisotropy, which is associated with the discretisation error. Through the implicit iterative minimization problem, which is based on the particle concentration gradient, the algorithm is able to control the particle spatial distribution and therefore, the anisotropy of the particles. The implicit method has been implemented in the software ASPHODEL of the ANDRITZ group, which adopts an SPH-ALE solver. In order to demonstrate its effectiveness, the IIPS performances have been compared to the explicit shifting technique. Due to the characteristics of ALE-SPH models, in order to keep the scheme consistency, two different methodologies to update the physical quantities, named "Implicit iterative particle shifting with fictitious time step" and "Implicit iterative particle shifting with MLS reconstruction", have been proposed and tested in two-dimensional test cases: the Taylor-Green vortex, the moving box inside a rectangular box and the jet impacting a flat surface. With these applications, it has been shown that for affordable computational overheads, the IIPS maintains isotropic particle distribution, significantly increasing the accuracy, confirming its superiority in comparison to existing explicit shifting approaches.

Acknowledgment

These intense and challenging years have come to the end through a lot of experiences and emotions, there are many people that have supported me to get to this point and they deserve my deepest acknowledgements.

My profound gratitude goes to Prof. Renato Vacondio for the incredible help and support that he has always given to me, he has constantly motivated me through his passion and his knowledge that he has been able to pass to me.

Another special thank goes to Dr Jean-Christophe Marongiu for his mentoring and his constant advice and his patient support he has always provided me.

Thank you both for being excellent examples of passion, commitment and hard-working, I will preciousely keep these teachings with me.

I would like to thank the research group of Hydraulics and Hydraulic Construction at the University of Parma, the Professors and the students for their willingness to support me. A special thought goes to all the colleagues met at the RandD Department of Andritz Hydro in Vevey with whom I shared precious months and the people, spread across the world, who attended the SPH meetings organized by the University of Manchester for the enriching discussions, among them a special thanks to Prof Benedict D. Rogers and Prof Georgeos Fourtakas for their precious help during our project.

A huge thanks to all my friends for the nonstop support in all my decisions, and finally my utmost gratitude for everything goes to my big family for having unconditional trusted me during these years.

To my family and my friends

Contents

ABSTRACT	ii
LIST OF FIGURES	ix
LIST OF TABLES	xiii
LIST OF ABBREVIATIONS	xiv
LIST OF SYMBOLS	xiv
1 Introduction	1
1.1 SPH Grand Challenges	1
1.2 Objective of the project	3
1.3 Outline of the thesis	3
2 Meshless numerical methods	5
2.1 Introduction	5
2.2 Mesh-based and mesh-less approaches	6
2.3 Meshless numerical methods literature review	8
2.4 Introduction to Smoothed Particle Hydrodynamics	9
2.4.1 SPH applications	9
2.4.2 Accuracy of the SPH operator	10
2.4.3 Numerical methods for particle distribution regularization	11
2.5 Summary and conclusions	12
3 Validation tests	13
3.1 Taylor-Green vortex	13
3.2 Moving square inside a rectangular box	16
3.3 Impinging jet on a flat surface	17
3.4 Numerical investigations	18
4 Smoothed Particle Hydrodynamics methodology	21

4.1	SPH theoretical fundamentals	21
4.1.1	SPH continuous interpolation	21
4.1.2	Kernel functions	25
4.1.3	SPH discrete interpolation	28
4.1.4	Convergence analysis of an SPH interpolation	29
4.2	Summary and conclusions	30
5	ALE-SPH model for Weakly-Compressible fluids	31
5.1	Governing equations for fluid dynamics	31
5.2	Arbitrary Lagrangian-Eulerian	34
5.3	Weakly-Compressible SPH model	35
5.4	Weakly-Compressible ALE-SPH model	36
5.5	SPH-ALE for numerical simulations	40
5.5.1	Volume equation	41
5.5.2	Interface reconstruction: MUSCL scheme	41
5.5.3	Time integration	42
6	Correction techniques for particle distribution	43
6.1	SPH schemes with particles reordering techniques	43
6.2	SPH schemes with repulsive forces	44
6.3	Incompressible SPH schemes with particle shifting techniques	45
6.4	Weakly Compressible SPH schemes with Particle Shifting Techniques	49
6.5	ALE-SPH schemes with Particle Shifting Techniques	51
6.5.1	Transport velocity in a quasi-Lagrangian scheme	51
6.5.2	Transport velocity based on Riemann solver	51
6.5.3	δ -ALE-SPH	52
7	Explicit Particle Shifting techniques	54
7.1	Riemann-based particle shifting technique	55
7.1.1	Results solving the kinematic equation	55
7.1.2	Results solving the Navier-Stokes equations	55
7.2	Improvement of a Riemann-based particle shifting technique	58
7.2.1	Particle Shifting coefficient	58
7.2.2	Fictitious Pressure field	59

7.2.3	Results solving the kinematic equation	61
7.2.4	Results solving the Navier-Stokes equations	61
7.3	Summary and conclusions	66
8	Implicit Iterative Particle Shifting Technique	69
8.1	1D formulation	69
8.2	2D formulation	71
8.3	Numerical test cases	77
8.3.1	Static test case	77
8.3.2	Kinematic test case	85
8.4	Summary and conclusions	92
9	Implicit iterative particle shifting in ALE-SPH schemes	93
9.1	IIPS in ALE-SPH schemes	93
9.1.1	IIPS with a fictitious time step	94
9.1.2	IIPS with MLS reconstruction	96
9.2	Evaluation of the IIPS in ALE-SPH	97
9.3	Summary and conclusions	102
10	Applications	103
10.1	Moving box in squared domain	103
10.1.1	Threshold evaluation for IIPS	103
10.1.2	Particle shifting techniques comparison	105
10.2	Impinging jet on a flat surface	112
10.2.1	Threshold evaluation for IIPS	112
10.2.2	Particle shifting techniques comparison	116
10.3	Summary and conclusions	118
11	Conclusions and perspectives	121
11.1	Conclusions	121
11.2	Perspectives	123
	REFERENCES	123

List of Figures

2.1	Fluid flow description of motion in a Cartesian frame of reference, (a) Eulerian: particles pass through the control volume and (b) Lagrangian: particles are followed using a position vector.	6
2.2	Mesh-less discretization, (a) overview and (b) details of the computational domain discretized by particles for a turbine. (Neuhauser 2014)	7
2.3	Mesh-based discretization, (a) overview and (b) details of the computational domain discretized by grid for a turbine. (Hirsch 2007)	8
3.1	Taylor-Green flow. Two-dimensional analytical velocity (a) and pressure (b) field at physical time $t = 0$ s. Flow streamlines (c).	14
3.2	Taylor-Green flow. Two-dimensional analytical velocity components at physical time $t = 0$ s.	15
3.3	Taylor-Green flow. Two-dimensional analytical velocity components at physical time $t = 1$ s.	15
3.4	Taylor-Green flow. Two-dimensional analytical total kinetic energy.	16
3.5	SPHERIC Benchmark Test #6, A. Colagrossi. Initial configuration.	16
3.6	SPHERIC Benchmark Test #6, , A. Colagrossi. Contours of velocity magnitude and pressure around the moving square at physical time $t = 5$ s, using FD solver.	17
3.7	Molteni and Colagrossi (2009), (a) initial configuration and set up, (a) fluid domain and pressure field at $tU/L = 10$	18
3.8	(a) Green et al. (2019) non-dimensional pressure field (b) non-dimensional pressure over the flat surface in steady conditions.	18
3.9	Test function x derivative.	19
4.1	(left) Kernel functions in 1D. (right) Gaussian kernel in 2D	27
4.2	Numerical test. L_2 error norm in SPH estimations of first partial derivative for a test function, (a) perturbed particle spacing $\sigma/\Delta = 0.1$, (b) Cartesian grid.	30
6.1	Oger et al. (2016), purely Lagrangian streamlines past a squared box.	44
7.1	TGV test case. Neuhauser (2014) formulation, imposed analytical solution. Results for $\Delta/L = 0.00625$ at $t = 0.2$ s.	56
7.2	TGV test case. Neuhauser (2014) formulation, solved simulation. Results for $\Delta/L = 0.00625$ at $t = 0.2$ s.	57

7.3	TGV test case. Neuhauser (2014) formulation. Time comparison for $c_0 = 10, 20, \text{ and } 100 \text{ m/s}$	58
7.4	TGV test case. Sensitive analysis for P_S imposed analytical solutions. Results for $\Delta/L = 0.00625$	59
7.5	Sensitive analysis for P_S imposed analytical solutions. Results for $\Delta/L = 0.00625$, a restricted domain is shown for visual evaluation.	60
7.6	TGV test case. Formulations comparison: (b) equation (6.33) Neuhauser (2014) formulation, (c) equation (7.1) shifting technique with particle shifting coefficient P_S , (d) equation (7.4) shifting technique with fictitious pressure field P^F , imposed analytical solution. Results for $\Delta/L = 0.00625$ at $t = 0.2 \text{ s}$	62
7.7	TGV test case. Formulations comparison: (b) equation(6.33) Neuhauser (2014) formulation, (c) equation (7.1) shifting technique with particle shifting coefficient P_S , (d) equation (7.4) shifting technique with fictitious pressure field P^F , solved simulations. Results for $\Delta/L = 0.00625$ at $t = 0.2 \text{ s}$	63
7.8	TGV test case. Time comparison: Neuhauser (2014) formulation, particle shifting coefficient P_S , fictitious pressure field P^F	64
7.9	TGV test case. Convergence analysis $L_2 \mathbf{v}_x - \mathbf{v}_{xA} $, explicit particle shifting comparison at $t = 1 \text{ s}$	65
7.10	TGV test case. $\Delta/L = 0.00625$, Neuhauser (2014) formulation: (a) velocity x component, (b) error on the velocity x component at $t = 1 \text{ s}$	66
7.11	TGV test case. $\Delta/L = 0.00625$, particle shifting coefficient P_S : (a) velocity x component, (b) error on the velocity x component at $t = 1 \text{ s}$	66
7.12	TGV test case. $\Delta/L = 0.00625$, fictitious pressure field P^F : (a) velocity x component, (b) error on the velocity x component at $t = 1 \text{ s}$	67
7.13	TGV test case. Kinematic energy analysis: (a) Kinematic energy error (b) (6.33), Neuhauser (2014) formulation, (c) (7.1), particle shifting coefficient P_S , (d) (7.4), fictitious pressure field P^F at $t = 0.2 \text{ s}$	67
8.1	Static test case. $\Delta/L = 0.05$ and initial perturbation $\sigma/\Delta = 0.10$. Particle distribution at (a) $it = 0$ and (b) $it = 10$	79
8.2	Static test case. $\Delta/L = 0.025$ and initial perturbation $\sigma/\Delta = 0.10$. ∇C magnitude at (a) $it = 0$ and (b) $it = 10$. Note that images have different colorbars.	80
8.3	Static test case. $\Delta/L = 0.25$ and initial perturbation $\sigma/\Delta = 0.10$. $\ \partial_x f^{SPH} - \partial_x f^{an}\ $ at (a) $it = 0$ and (b) $it = 10$.. Note that images have different colorbars.	80

8.4	Static test case. $\Delta/L = 0.025$ and initial perturbation $\sigma/\Delta = 0.10$. $\partial_x f^{SPH}$ at (a) $it = 0$ and (b) $it = 10$. Note that images have different colorbars.	80
8.5	Static test case. Initial perturbation $\sigma/\Delta = 0.10$ (top) and $\sigma/\Delta = 0.25$ (bottom).	81
8.6	Static test case. $\Delta/L = 0.0125$ and initial perturbation $\sigma/\Delta = 0.10$. . .	82
8.7	Static test case. (a) iteration needed to reach the ϵ value by the linear system solver, (b) normalized CPU time, scaled on $\epsilon = 10^{-6}$, using tolerance (normalized residual) value in the linear system solver.	83
8.8	Static test case. Convergence analysis results, $L_2(\partial_x f)$ for different h/Δ values.	84
8.9	Static test case. $\Delta/L = 0.0125$ and initial perturbation $\sigma/\Delta = 0.10$. (a) implicit iterative particle shifting method, (b) explicit iterative shifting method.	85
8.10	Kinematic test case. Procedure comparison, $\Delta/L = 0.025$ and $h/\Delta = 2.0$	87
8.11	Kinematic test case. Procedure comparison, $\Delta/L = 0.025$ and $h/\Delta = 2.0$	88
8.12	Kinematic test case. Method comparison, CPU time. Note different scales.	89
8.13	Kinematic test case. Particle distribution at $t = 0.2$ s.	90
8.14	Kinematic test case. Convergence analysis.	91
9.1	MLS reconstructions, error on a test function. $\Delta/L=0.025$	98
9.2	MLS reconstruction, error on a test function.	98
9.3	TGV test case. $\Delta/L = 0.00625$, (a) Neuhauser (2014) formulation, (b) IIPS with $L_\infty(\nabla C)_{thr} = 0.001/h$	99
9.4	TGV test case. Convergence analysis $L_2 \mathbf{v}_x - \mathbf{v}_{xA} $, implicit particle shifting comparison at $t = 1$ s.	100
9.5	TGV test case. $\Delta/L = 0.00625$, Neuhauser (2014) formulation (a) velocity x component in [m/s], (b) error on the velocity x component at physical time $t = 1$ s.	100
9.6	TGV test case. $\Delta/L = 0.00625$, implicit iterative shifting (a) velocity x component in [m/s], (b) error on the velocity x component at physical time $t = 1$ s.	101
9.7	TGV test case. $\Delta/L = 0.00625$, implicit iterative shifting with NFTS (a) velocity x component in [m/s], (b) error on the velocity x component at physical time $t = 1$ s.	101

9.8	TGV test case. $\Delta/L = 0.00625$, implicit iterative shifting with MLS II° (a) velocity x component in [m/s], (b) error on the velocity x component at physical time $t = 1$ s.	101
9.9	TGV test case. Kinetic energy error.	102
10.1	Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty thr} = 0.08/h$, h/Δ analysis. .	104
10.2	Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty}(\nabla C)_{thr} = 0.08/h$	105
10.3	Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty}(\nabla C)_{thr} = 0.08/h$. Shifting procedure comparison.	106
10.4	Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty}(\nabla C)_{thr} = 0.08/h$. Shifting procedure comparison, gradient of particle concentration at $t=0.75$ s. . .	107
10.5	Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty}(\nabla C)_{thr} = 0.08/h$. Shifting procedure comparison, particle distribution at $t=0.75$ s.	108
10.6	Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty}(\nabla C)_{thr} = 0.08/h$, $Re=100$. Explicit shifting, $P_S=10$, at $t=5$ s.	109
10.7	Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty}(\nabla C)_{thr} = 0.08/h$, $Re=100$. Implicit iterative shifting with NFTS, at $t=5$ s.	110
10.8	Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty}(\nabla C)_{thr} = 0.08/h$, $Re=100$. Implicit iterative shifting with MLS second-order, at $t=5$ s.	111
10.9	Moving Box test case. $\Delta/L = 0.0125$. Shifting procedure comparison, CPU time.	112
10.10	Impinging Jet test case. $H/\Delta = 10$, Shifting procedure comparison. . .	114
10.11	Impinging Jet test case. $H/\Delta = 10$, ∇C , explicit shifting, at $tU/L =$ 0.0525	114
10.12	Impinging Jet test case. $H/\Delta = 10$, ∇C , explicit shifting, $P_S=10$ at $tU/L = 0.0525$	115
10.13	Impinging Jet test case. $H/\Delta = 10$, ∇C , implicit iterative shifting at $tU/L = 0.0525$	115
10.14	Impinging Jet test case. $\Delta/L = 0.0005$, ∇C , shifting procedure com- parison.	116
10.15	Impinging Jet test case. $\Delta/L = 0.002$, $\Delta/L = 0.001$, $\Delta/L = 0.0005$, shifting procedure comparison, normalized pressure at P_0 in the time interval $tU/L = [2, 3]$	118
10.16	Impinging Jet test case. $\Delta/L = 0.0005$, shifting procedure comparison, normalized pressure at P_0 in the time interval $tU/L = [2, 3]$	119
10.17	Impinging Jet test case. $\Delta/L = 0.0005$. Shifting procedure comparison, CPU time.	120

List of Tables

7.1	TGV test case. Convergence analysis results, $L_2 \mathbf{v}_x - \mathbf{v}_{xA} $ and convergence ratio θ , explicit shifting.	64
8.1	Static test case. Convergence analysis results, $L_2(\partial_x f)$ and convergence ratio θ	83
8.2	Static test case. Particle Shifting Techniques comparison	85
8.3	Kinematic test case. Convergence ratio results.	91
9.1	TGV test case. Convergence analysis results, $L_2 \mathbf{v}_x - \mathbf{v}_{xA} $ and convergence ratio θ , implicit iterative shifting.	100
10.1	Impinging Jet test case. $\Delta/L = 0.0005$. Normalized average pressure on the plate and percentage error $E_r\%$	117

List of Abbreviations

ALE	Arbitrarily Lagrangian-Eulerian
APS	Adaptive Particles Refinement
CAD	Computer-aided Design
CFD	Computational Fluid Dynamic
FDM	Finite Difference Method
FEM	Finite Element Method
FPM	Finite Pointset Method
FSI	Fluid-Structure Interaction
FVM	Finite Volume Method
FVPM	Finite Volume Particle Method
GC	Grand Challenges
GUI	Graphic User Interface
HPC	High Performance Computing
IIPS	Implicit Iterative Particle Shifting
ISPH	Incompressible SPH
MPM	Material Point Method
MPS	Moving Particle Semi-implicit
PDE	Partial Differential Equations
PIC	Particle-In-Cell
PPE	Poisson Pressure Equation
PST	Particle Shifting Technique
RBF	Radial Basis Functions
SPH	Smoothed Particle Hydrodynamics
WCSPH	Weakly Compressible SPH

List of Symbol

$\mathbf{x}$	Position vector
$\mathcal{D}$	Computational domain
∇W	kernel function gradient
ν	Kinematic viscosity
ω	Particle volume
ϕ	generic fluid property
ρ	Density
$\mathbf{n}$	Normal vector
$\mathbf{g}$	Gravity force
$\mathbf{v}$	Velocity vector
$\mathbf{v}_0$	Arbitrarily transport velocity vector
d	Number of spatial dimensions
E_k	Kinetic energy
h	Smoothing length
m	Mass
p	Pressure
T	Temperature
t	Time
u	x -component of $\mathbf{v}$
u_0	x -component of $\mathbf{v}_0$
v	y -component of $\mathbf{v}$
v_0	y -component of $\mathbf{v}_0$
W	Kernel function
w	z -component of $\mathbf{v}$
w_0	z -component of $\mathbf{v}_0$
x	Space component
y	Space component
z	Space component

Chapter 1

Introduction

1.1 SPH Grand Challenges

The Smoothed Particle Hydrodynamics (SPH) is a Lagrangian meshless numerical scheme introduced in the late years of 70s by Gingold and Monaghan (1977) and by Lucy (1977). Nowadays, it is increasingly receiving attention from the scientific community as a promising methodology, it is undisputed that it has already made huge signs of progress from its early stages, but its potential is still not fully expressed. Indeed, the SPHERIC (SPH Research and Engineering International Community) scientific organization, whose members are researchers and industrial users, aiming to extend the knowledge on the method, has grouped the areas of research into five categories, named Grand Challenges (GC):

- GC#1 convergence, consistency and stability,
- GC#2 boundary conditions,
- GC#3 adaptivity,
- GC#4 coupling to other models,
- GC#5 applicability to industry.

A review of these challenges has been recently published in Vacondio et al. (2020), in which the progress that have been achieved are presented and the investigations that still require further investigations are disclosed.

The main objective for the first GC is the theoretical formalization of the SPH numerical properties to increase the reliability of the method. The three properties, consistency, stability and convergence, are deeply connected, although, underneath the SPH methods, there is not a specific and equivalent theory, like the Lax Equivalence theorem which is strictly valid for Finite Difference Method (FDM). For this reason, significant mathematical developments are still missing in the literature. On the other hand, the sources of instability that negatively affect the convergence of the method have been

already pointed out, as tensile instability, (Swegle et al. 1995; Monaghan 2000; Morris et al. 1997; Sun et al. 2019) and pairing instability, (Price 2012; Dehnen and Aly 2012).

The particle distribution has a major impact on convergence, which has, consequently, an influence on the accuracy of the SPH interpolation, (Quinlan et al. 2006; Amicarelli et al. 2011). In order to improve the particle distribution, different numerical treatments have been introduced, (Xu et al. 2009; Lind et al. 2012; Vacondio et al. 2013; Oger et al. 2016).

The GC#2 is dedicated to the boundary conditions, which have a key role, especially in simulation with complex geometries. Among the different numerical methodologies that have been proposed throughout the years to treat boundaries, which can be found in different publications, (Monaghan 2005; Gomez-Gesteira et al. 2010; Monaghan 2012; Gotoh and Khayyer 2016; Violeau and Rogers 2016), none of them has gained the supremacy to be broadly valid and effective in every application. In order to fulfill mathematical issues, strictly related even with CG#1, e.g. conservation properties, convergence and consistency, efforts are still required.

The GC#3 takes in consideration algorithms used to discretize the domain with different resolutions, meaning particles with different sizes. The adaptivity is a relevant research field because adopting a uniform resolution in the whole domain can be computationally expensive even with the recently available hardware, whereas a higher level of refinement located exclusively in zones where the flow requires a larger number of interpolation points can potentially drastically reduce the computational costs. Numerical techniques associated with GC#3 have been first seen as remeshing methods, (Koumoutsakos 2005; Børve et al. 2005) and later as non-uniform initial domain discretization (Oger et al. 2006). Nowadays the dynamic refinement methods, (Lastiwka et al. 2005; Feldman and Bonet 2007; Vacondio et al. 2013; Barcarolo et al. 2014; Chiron et al. 2018), are able to dynamically modify the particle size during the simulation, indeed they are applied to a wide range of problems. Nevertheless, parallel implementations and robust theories on adaptivity for multi-phase flows are an open discussion.

Compared to mesh-based methods, SPH has convenient points of strength in simulations with moving objects and complex interfaces, because, in these applications, the mesh generation process is complicated and computationally expensive. In addition, in fields where Finite Volume Method (FVM) or Finite Element Method (FEM) are highly effective, SPH can be coupled with these schemes to simulate specific parts of the problem, aggregating the benefits from the different solvers involved. This foremost topic is explored in CG#4, aiming to smooth the paths toward accurate and fast Fluid-Structure Interaction (FSI) simulations, e.g. FSI with rigid objects, elastic structure, extreme loadings, in which solid components are usually solved with FEM, (Li et al. 2015; Yang et al. 2016; Long et al. 2016; Fourey et al. 2017).

The SPH method is increasingly attracting partners from the industrial world because of its capability to simulate violent flows with fragmented interfaces and/or complex multi-physics problems. Lastly, in this context, the CG#5 encourages the developments of components required among industries and professionals to extensively use the SPH techniques, (Shadloo et al. 2016). To be competitive, SPH needs rigorous and stable solvers as well as user-friendliness in pre-processing, easy integration with Computer-aided Design (CAD) tools, and post-processing, lightweight software capable to handle the visualization of millions of particles simultaneously. For example, to encourage new users, a novel Graphic User Interface (GUI) was introduced in the SPH-based open source code DualSPHysics, (Vieira et al. 2017). In GC#5, the other major task is the mandatory reduction of computational time, throughout High Performance Computing (HPC) and parallelism techniques.

To summarize, the Grand Challenges identified by SPHERIC have to be seen as a motivation for researchers to try to solve some relevant issues of the SPH models and to share knowledge.

1.2 Objective of the project

A brief presentation of the SPHERIC Grand Challenges has been included to better understand the objectives underneath this research, in fact, as previously reported among the studies related to GC#1, the lack of uniformity in particle distribution affects negatively the accuracy of SPH schemes. This phenomenon commonly occurs in Lagrangian models, especially in simulations of violent flows that drastically perturb the particle distribution. As practical examples in engineering, water jets in Pelton turbines produce highly distorted and complex flows while impacting the buckets, (Marongiu et al. 2010). With these motivations, the present thesis aims to investigate solutions to increase the accuracy of SPH schemes improving the particle distribution. This would allow higher quality in modelling the fluid behaviours in the above-mentioned problems.

1.3 Outline of the thesis

In this section is presented the structure of this thesis.

Chapter 2 includes an introduction to the different approaches used in numerical modelling and the description of grid-based and mesh-less models used in fluid dynamics simulations. The review of these schemes is enlightened, among them, the SPH models have a wide applications which are briefly reported.

Chapter 3 introduces the numerical cases used for validation in this thesis, the

Taylor-Green vortex, the moving square inside a rectangular domain and the jet impacting a flat surface for whose analytical or reference solutions are available.

Chapter 4 deals with the theoretical fundamentals of the SPH models, from the continuous to the discrete level. The kernel properties are illustrated with an analysis of the accuracy of the models, referring that to the GC#1.

Chapter 5 reports the SPH formulations for weakly compressible fluids and the steps needed to introduce the Arbitrarily Lagrangian-Eulerian (ALE) SPH models, focusing on the different advantages and drawbacks of these methods. An SPH-ALE solver has been implemented in the software called ASPHODEL, developed in the PhD work of Marongiu (2007) and Leduc (2010), which is briefly presented.

Chapter 6 is dedicated to the literature review of the Particle Shifting Technique (PST), since the original applications to the most recent formulations discussing the main characteristics of these algorithms.

Chapter 7 presents an optimization for the explicit particle shifting technique fitted in ASPHODEL by Neuhauser (2014). Numerical analysis regarding particle distribution quality and computational time has been conducted to assess the behaviour of the original algorithm, then it has been proposed two different upgrades which have been studied and evaluated through the Taylor-Green Vortex (TGV) test case.

Chapter 8 introduces the novel Implicit Iterative Particle Shifting (IIPS) technique, the main relevant objective of this project. Thus, to fully clarify the theoretical aspects, the 1D and 2D formulations are derived, then the method properties are investigated using two purely analytical test cases.

Chapter 9 is used to show the applicability of the new IIPS into the SPH-ALE model, discussing the advantages and the drawbacks in the different strategies available.

The Chapter 10 is reserved to show the new proposed particle shifting methodologies applied to the moving square inside a rectangular box, which is used as a benchmark test case among the SPH international community, and the impinging jet on a flat surface, that can be seen as a close simplified example of a Pelton turbine.

Chapter 11 is left for the conclusions and the general perspectives that this work generates.

Chapter 2

Meshless numerical methods

In this chapter the mesh-based and the mesh-less approaches for numerical modelling are initially introduced, generally, in these techniques, computational nodes are interconnected or disconnected from each other. Then, the main focus is posed on the latter ones, it is preliminary to illustrate the SPH main characteristics, which are later addressed to present the objective of this project.

2.1 Introduction

Numerical simulations are important tools used to analyse complex phenomena. Their versatility has been remarkably incremented due to the increasing computer capabilities. This has allowed obtaining numerical solutions even for extremely complicated engineering applications and natural phenomena. For these reasons numerical modelling drastically changed the way physical problems can be studied and described. Physical and theoretical models have some intrinsic lacks, respectively the needs of calibration through experimental data and the validity for exact solutions only, which limit their reliability as standalone investigation instruments. This has led to a strong interconnection, like comparison and validations, with numerical simulations which are able to translate important aspects of the problem into a discrete form of mathematical laws, using theoretical assumptions and physical principles. The numerical techniques for solving numerical models can be different but all of them have common steps in the solution strategy.

Initially, it is crucial to identify the most relevant aspects of physical phenomena under investigation, for which a simplified mathematical model needs to be derived. The phenomena behaviour is described with the governing equations, together with the initial state conditions and with the boundaries limitations.

Successively, the domain discretization is the first step to numerically solve the above-mentioned set of equations, the process consists of the representation of the continuum by distinct components or by connected elements that are used as computational nodes. The approach utilized in the discretization stage is decisive to define the numerical inherent characteristics of the methodology. As stated in the preamble, there are two

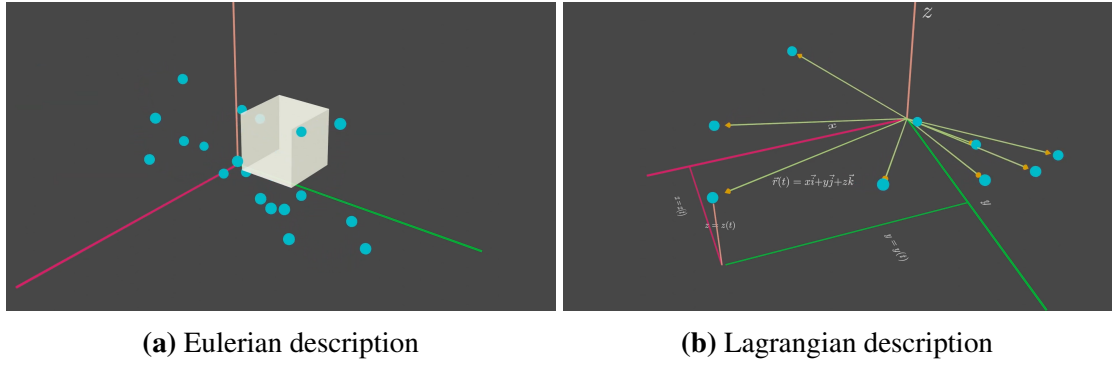


Figure 2.1 Fluid flow description of motion in a Cartesian frame of reference, (a) Eulerian: particles pass through the control volume and (b) Lagrangian: particles are followed using a position vector.

main different domain discretization strategies, mesh-based or mesh-free techniques, both have advantages and disadvantages and it has to be chosen the one better suited to describe the nature of the problem and to achieve accurate numerical solutions, more details later in the chapter.

Following, the mathematical laws that govern the physical problem are derived coherently with the discretization methodology, in the case of transient problems numerical algorithms for time integration has to be introduced. As a final step, these numerical expressions need to be translated in computational codes that can be run in proper infrastructures, the range of options of programming languages and hardware is wide and it constantly increases.

2.2 Mesh-based and mesh-less approaches

Among numerical models, the Computational Fluid Dynamic (CFD) is a branch broadly used to simulate different applications that involve the flows of fluids, characterized by a variety of time and length scales. As previously mentioned, even in fluid dynamics modelling there are approaches better suited to the simulation of specific types of flows.

Mesh-based methods adopt a computational frame that is made up of nodes topologically connected, where the fluid variables are evaluated. Grid-based methods, like finite-difference, or finite-volume are the most widespread schemes, adopted especially in commercial software, utilized by industry, due to their robust theoretical fundamentals and mature computational algorithms.

The accuracy of the numerical approximation is related to the mesh topography, as the size and the shape of the mesh, and to obtain valuable results, the grid preparation and generation need high expertise and it can be very time-demanding. There are different types of mesh with structured or unstructured geometry, (Figure 2.3), in these schemes, mass fluxes between adjacent cells have to be taken into account. These dis-

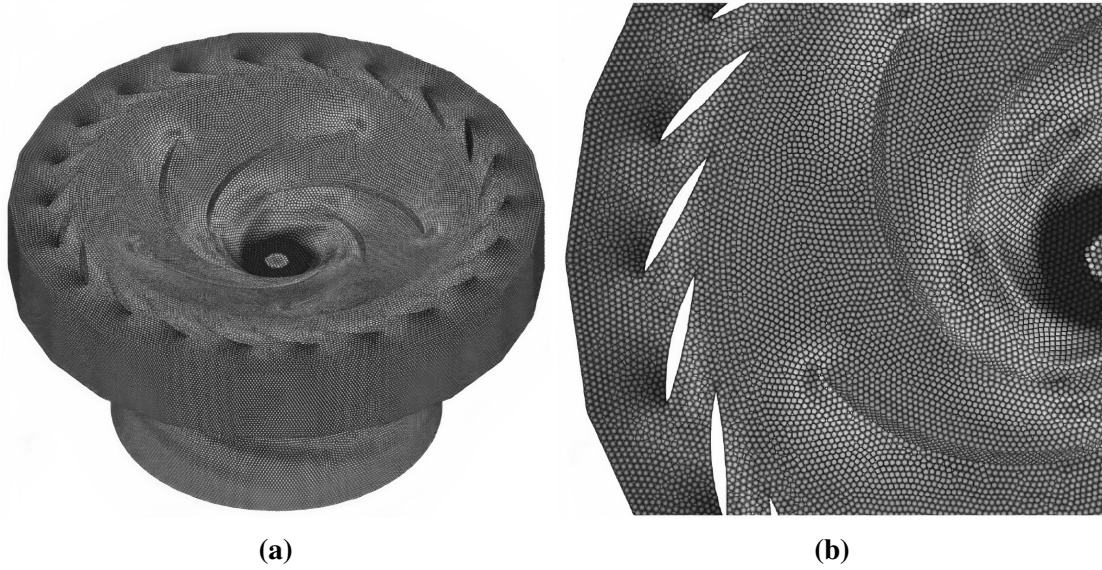


Figure 2.2 Mesh-less discretization, (a) overview and (b) details of the computational domain discretized by particles for a turbine. (Neuhauser 2014)

cretization techniques usually describe the spatial domain with an Eulerian approach in which the computational grid is assumed to be fixed in the space without deformations while the continuum evolves, (Figure 2.1 (a)). As a relevant drawback, the computational grid needs to be large enough to cover even portion of space in which the fluid can move to. These methods provide information on the field quantities globally whereas specific fluid trajectories, streamlines, are difficult to track and follow. In many applications with confined domains or internal stiff boundaries, where no adaptive behaviour is needed, they can produce really accurate simulations, on the contrary, a cumbersome numerical mapping is required to handle complex and deformable geometry. Moreover, these methods can be expensive and slow, especially in presence of moving components, free surfaces and material interfaces in the computational domain.

The need of reliability and efficacy in these latest types of problems led to the introduction and to the development of numerical meshless methods. The basic idea of these approaches is to discretize the continuum through a set of nodes without any topological connections (Figure 2.2), in order to directly follow the deformations experienced by the fluid, avoiding the degradation of the numerical results and maintaining a suitable computational effort. Computational nodes represent fluid material particles carrying physical quantities, moving in space along trajectories accordingly to the dynamics that govern the problem, following their velocities and accelerations. These methods are based, in general, on the Lagrangian approach, (Figure 2.1 (b)), avoiding the computation of convective terms in the governing equations, simplifying the numerical handling. Only the fluid medium is discretized and it is free to move in the whole domain, this allows easily managing of simulations with large deformations.

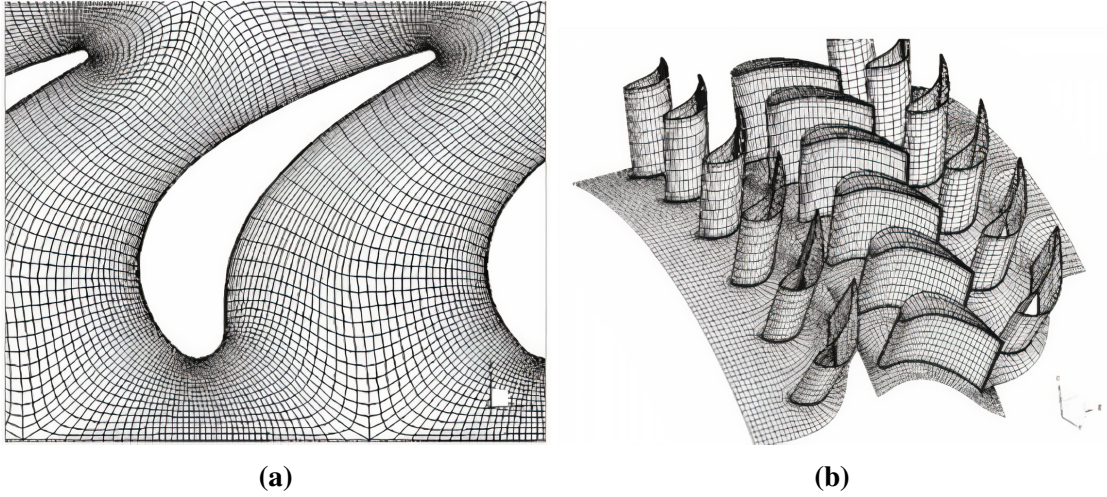


Figure 2.3 Mesh-based discretization, (a) overview and (b) details of the computational domain discretized by grid for a turbine. (Hirsch 2007)

It is worthwhile to introduce, in the context of this thesis, the link between the Lagrangian and the Eulerian description, here valid for fluid dynamics, for a generic fluid property ϕ ,

$$\frac{d\phi}{dt} = \frac{\partial\phi}{\partial t} + \mathbf{v} \cdot \nabla\phi \quad (2.1)$$

$\frac{d\phi}{dt}$ (or an equivalent notation $\frac{D\phi}{Dt}$) is the global derivative that represents the total rate of change of ϕ along the fluid trajectory whereas $\frac{\partial}{\partial t}$ is the local derivative that describes the rate of change of ϕ in a fixed position in space. $\mathbf{v} \cdot \nabla\phi$ expresses the convective derivative.

2.3 Meshless numerical methods literature review

In this section, a review of the main developments and achievements in numerical modelling using meshless methods is reported. Starting from the late years of the 50s, several mesh-free schemes have been adapted to the study of physical or engineering problems whose constitutive equations are described through Partial Differential Equations (PDE).

The first attempt was the Particle-In-Cell (PIC) method by Evans et al. (1957), followed by its extension, the so-called Material Point Method (MPM), presented by Sulsky et al. (1994). In both the PIC and the MPM the motion of the particles is described through the Lagrangian trajectories whereas a background fixed Eulerian mesh is required to compute the field quantities; in these schemes, particles move and exchange information back and forth to the grid.

In 1996, in a similar manner, Oñate et al. (1996) adopted a scattered distribution of points, called Finite Pointset Method (FPM), which is based on a pure Lagrangian approach. A remeshing technique is required in FPM in order to prevent instabilities

triggered by severe mesh deformations.

The Moving Particle Semi-implicit (MPS) method was introduced in the same year by Koshizuka and Oka (1996), to simulate viscous incompressible flows, it has been extensively developed in the following decades and it is frequently utilized, (Tsuruta et al. 2013), in simulations with highly distorted flows. In the MPS schemes, the Lagrangian discrete computational domain is stabilized adopting inter-particle repulsive-attractive forces.

In addition to the mesh-less scheme previously mentioned, the Radial Basis Functions (RBF) have been widely used, as an interpolation technique, for high-order functions reconstruction, using scattered sets of data. In 1990 RBFs were applied by Kansa (1990) to the solution of PDEs, where radial functions are used to determine basis function weights for steady-state or time-dependent problems, (Franke and Schaback 1998).

The differential operators that appear in the PDEs are discretized with a kernel-based spatial interpolation in the Finite Volume Particle Method (FVPM), presented by Hietel et al. (2000) in the early 2000s. This numerical scheme combines consistency and conservation properties typical of classical FVM with the advantages of a Lagrangian description of motion, including accessible implementation of boundary conditions. FVPM has been further developed in Nestor et al. (2009), for viscous flows, in Nestor and Quinlan (2013), for fluid-structure interactions and later in Quinlan (2018), for free-surface applications.

In this section the Smoothed Particle Hydrodynamics schemes have not been mentioned on purpose, a specific introduction, from the initial developments to the newest applications, pointing out some crux aspects of the method is following presented.

2.4 Introduction to Smoothed Particle Hydrodynamics

In recent years, of all the meshless schemes currently available, SPH has received increasing attentions and its range of applications is constantly growing, as presented the reviews of the method (Monaghan 2005; Gomez-Gesteira et al. 2010; Monaghan 2012).

2.4.1 SPH applications

SPH has been firstly developed for astrophysics by Gingold and Monaghan (1977), using a magnetohydrodynamics SPH formalism (SPMHD). Phenomena like the gravitational collapse of interstellar clouds or the interplanetary gases-dust interactions can be simulated to study stars or planets formation processes, (Price and Monaghan 2005). In these problems, the range of length and time scales involved changes by many orders of magnitude, but high-density gradients can be handled directly by SPH models due to the ability to adopt variable smoothing lengths. Additionally, these problems do not

require proper treatments for the boundaries because they simulate phenomena in the infinite space.

Later, the SPH operator has been used to discretize the governing elastic-plastic equations in solid mechanics, (Libersky and Petschek 1991), involving fractures and damage constitutive models, (Gray and Monaghan 2004). In these applications the damage information is directly carried by the SPH particles and the collapse of the modelled medium is directly considered by the meshless nature of the method.

Moreover, SPH has been utilized in geo-mechanic problems, (Zhu et al. 1999), to simulate fluids flowing through a porous matrix, in fact, the versatility of SPH has been extremely efficient in simulations of groundwater flows which are characterized by both macro-scale physics and micro-scale effects, (Masciopinto and Palmiotta 2013).

The first application to fluids simulations has been proposed in Monaghan (1992), reproducing cases with free surface flows impacting solid objects. It was a game-changer achievement due to the multitude of engineering problems in which it has been applied including sloshing tanks, (Delorme et al. 2009), ship-hull design (Veen 2010) and many other applications. These examples are just a limited part of the many fields in which SPH has been applied, presented to demonstrate the advances that the methods have already achieved, even if there are still unsolved issues, as reported in the SPHERIC Grand Challenges; among them, numerical instabilities related to GC#1 are presented in the following section.

2.4.2 Accuracy of the SPH operator

The original version of SPH is based on kernel basis function in which material points follow the fluid trajectories and therefore, the particle distribution can break apart with particles clumping together or separating in nonphysical voids. These behaviours, intrinsically due to the Lagrangian nature of the method, affect negatively the numerical stability and therefore they have been constantly faced over the past years (Morris et al. 1997; Monaghan 2000). Historically, the tensile instability and the pairing instability have been identified as sources of numerical issues.

The first one has been studied by Swegle et al. (1995), this effect tends to cluster or create large holes in the fluid continuum and it is usually triggered by negative pressure, compromising the field variables accuracy. Several techniques have been proposed to fix the tensile instability, introducing a background pressure, (Morris 1996), adding extra terms in the momentum equation, (Monaghan 2000), which produce repulsive forces between particles, or using the non-conservative formulation in the momentum equation, (Morris et al. 1997; Sun et al. 2019); nevertheless these expedients do not completely solve this issue.

Similarly, the second one, the pairing instability, tends to overlap particles in pairs.

It strictly depends on the ratio between smoothing length and particles spacing, and the kernel function used in the simulations, (Price 2012), but it can be prevented using kernels with positive Fourier transforms (e.g. Wendland kernels, Wendland 1995; Wendland 2004; Dehnen and Aly 2012).

These two different types of instability both produce perturbation in the particle distribution reducing the accuracy, but the overall particles spatial positions cannot be controlled due to Lagrangian nature. Nowadays, it is well known that the particle distribution affects the accuracy of the spatial interpolation as illustrated in GC#1, (Vacondio et al. 2020; Colagrossi et al. 2012), and since SPH uses kernel-based operators, the spatial interpolation exhibits numerical instability for disordered particles distributions, (Quinlan et al. 2006; Amicarelli et al. 2011). To regularize the particle distribution numerous approaches have been proposed, mentioned in the following section.

2.4.3 Numerical methods for particle distribution regularization

The particle shifting technique was proposed for the first time by Nestor and Quinlan (2007) in the framework of FVPM, a similar approach, based on particles position, was initially applied to Incompressible SPH (ISPH) scheme by Xu et al. (2009), for interior flows, with the aim of improving the efficiency of the Poisson Pressure Equation (PPE) solver. Later, this method was extended by Lind et al. (2012), to free-surface problems, adopting the so-called Fickian formulation, based on Fick's diffusion law, which adjusts the particle distribution accordingly to the gradient of the particle concentration. This PST was modified and applied to body-water slam simulations by Skillen et al. (2013), for single-phase flows, and later by Mokos et al. (2017), and by Fourtakas and Rogers (2016), for multi-phase flows.

In Weakly Compressible SPH (WCSPH) schemes, an artificial particle displacement algorithm was initially introduced by Shadloo et al. (2012). Since then, Vacondio et al. (2013), and Vacondio et al. (2016) combined a Fickian-based shifting technique with δ -SPH proposed by Marrone et al. (2011) and the Adaptive Particles Refinement (APS) method, later repeated in the δ^+ -SPH formulation by Sun et al. (2017). A more general PST for free-surface or multi-phase flows has been recently presented by Khayyer et al. (2017), without tuning parameters, later implemented and optimized with an iterative procedure in Khayyer et al. (2019).

In addition, Oger et al. (2016) introduced a methodology with limitations for the maximum shifting displacements, in the context of Arbitrarily Lagrangian-Eulerian SPH, Vila (1999). In the same framework, a shifting procedure without limitations in the shifting distance has been introduced by Neuhauser (2014).

Moreover, Antuono et al. (2021), extended the δ^+ SPH formulation to an ALE-SPH framework. In this scheme, the explicit diffusive terms, in the equations of mass

and momentum, and the PST, directly introduced in the transport velocity formulation, guarantee consistency and increase the accuracy.

Recently, after having pointed out the main theoretical aspects of Fick’s law-based shifting methods, Michel et al. (2021) introduced a more general formulation that can be valid in many SPH schemes.

2.5 Summary and conclusions

In this chapter, starting from the global overview regarding numerical modelling, passing rapidly through the different domains’ discretization techniques and then through the two approaches to fluid simulations, the Particle Shifting Techniques (PSTs) has been introduced. PSTs represent the solution adopted to prevent anisotropic particle distributions which are a critical numerical issue in SPH schemes. Additional details about the algorithms utilized in these techniques are later reported in Chapter 6 and new methodologies are presented in Chapter 7 and Chapter 8 as the main investigations objectives.

Chapter 3

Validation tests

In this chapter the validation cases used to analyse the novel particle shifting methodologies, proposed later in this work, are described.

The unbounded Taylor-Green vortex, for which an analytical solution is available, is principally adopted to evaluate the numerical fundamental of the new shifting algorithms.

Then, two different tests, that can be representatives for more complex problems, have been chosen to verify the applicability of the novel methodologies in real applications: the first one with a bounded domain and a moving solid object and the second one with free surface. The results for these two applications can be compared with accurate reference solutions.

Additionally, some basic but important investigations tools, widely utilized in this document, are reported to avoid non-necessary repetitions.

3.1 Taylor-Green vortex

The Taylor-Green vortex is a two-dimensional case that has an analytical solution for the Navier-Stokes equations. The case has a bi-periodic squared domain and for each point, the velocity components on x and y directions, respectively, u and v , and the pressure p are computed as,

$$\begin{cases} u = Ue^{-8\pi^2 t/Re} \cos(2\pi x/L) \sin(2\pi y/L) \\ v = Ue^{-8\pi^2 t/Re} \sin(2\pi x/L) \cos(2\pi y/L) \\ p = \frac{\rho}{4} e^{-16\pi^2 t/Re} [\cos(4\pi x/L) + \cos(4\pi y/L)] \end{cases} \quad (3.1)$$

where ρ , is the density and $Re = UL/\nu$ is the dimensionless Reynolds number computed with ν , the kinematic viscosity, U the initial reference velocity and L the characteristic length of the problem which coincides with the size of the domain. The Taylor-Green vortex represents a counter-rotating decaying flow in which the streamlines are highly distorted as it can be seen in the velocity field, Figure 3.1 (c). Following in this work, in purely kinematic investigations focus has been posed on the motion of

the particles only, therefore, it has been simulated without any physical viscosity and the velocity and pressure initial fields have been reimposed at each time steps to kept them constant throughout the entire simulation, the flow is thus considered in its steady condition.

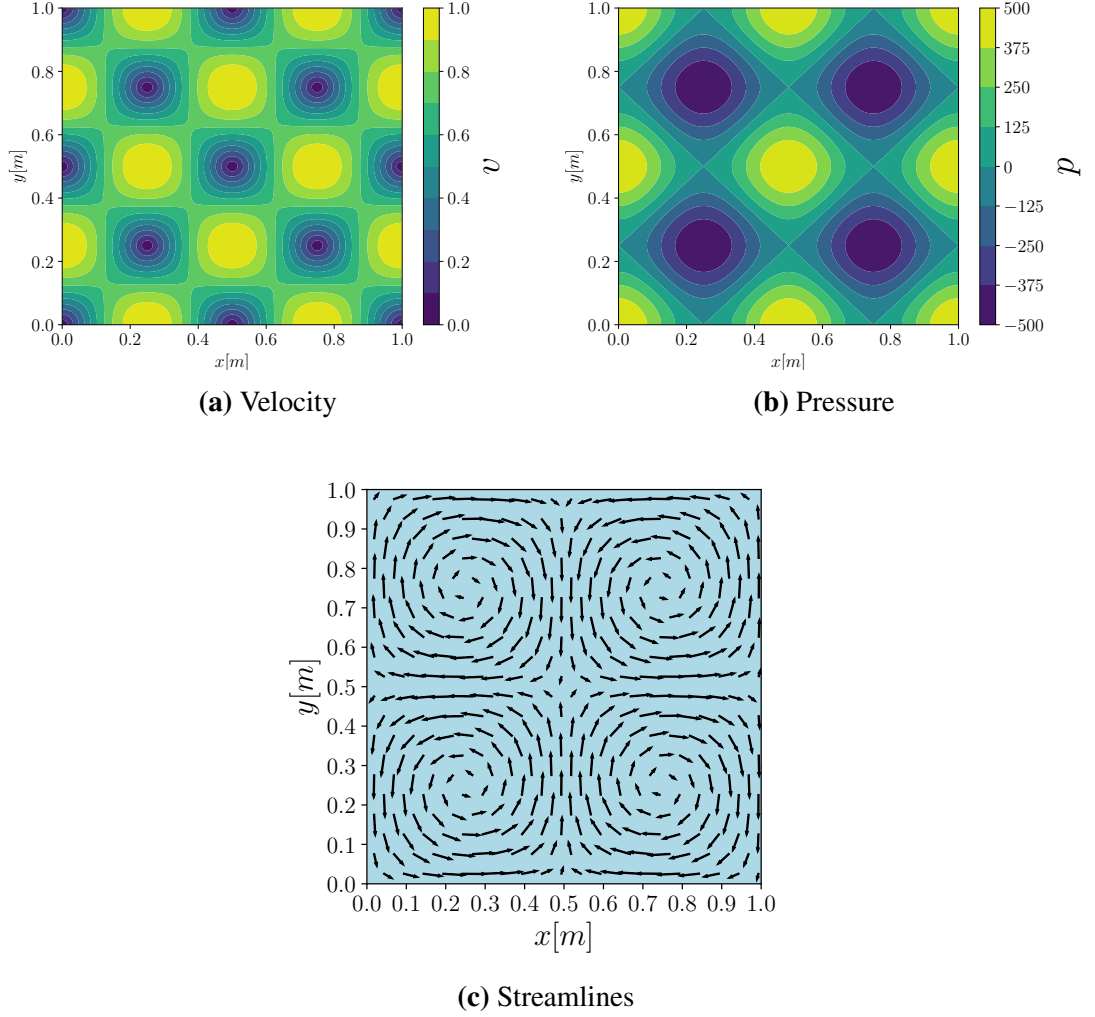


Figure 3.1 Taylor-Green flow. Two-dimensional analytical velocity (a) and pressure (b) field at physical time $t = 0$ s. Flow streamlines (c).

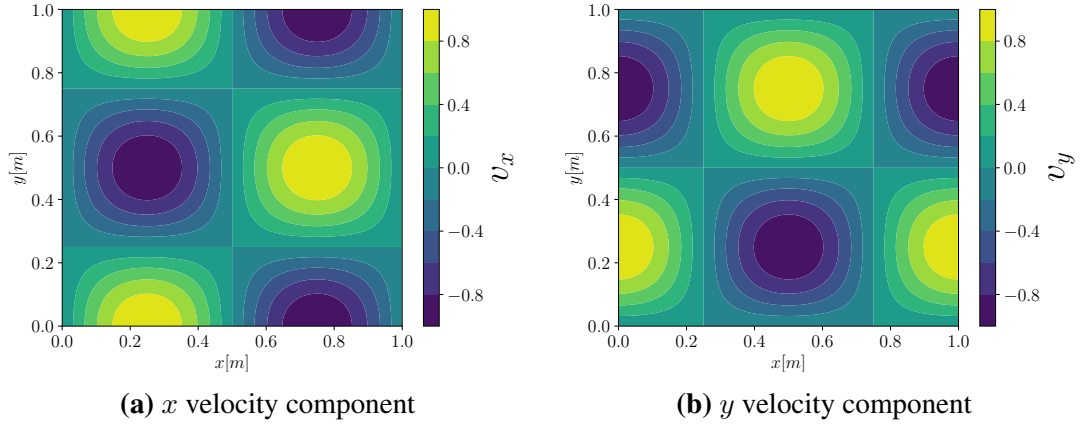


Figure 3.2 Taylor-Green flow. Two-dimensional analytical velocity components at physical time $t = 0$ s.

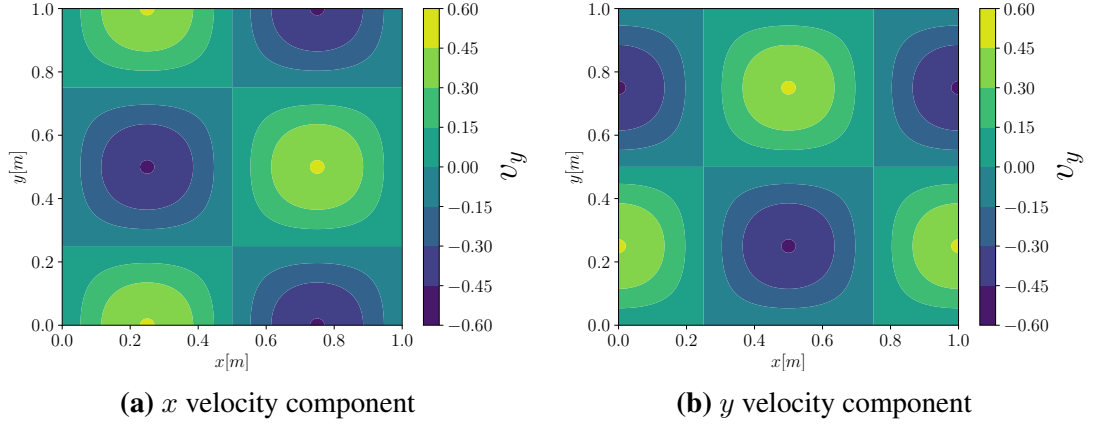


Figure 3.3 Taylor-Green flow. Two-dimensional analytical velocity components at physical time $t = 1$ s.

The global decaying of the velocity field can be assessed using the theoretical decay of the kinetic energy, E_{kA} , which is computed integrating in space equation (3.1),

$$E_{kA} = E_{k0} e^{-16\pi^2 \nu t}, \quad (3.2)$$

Figure 3.4 shows the analytical decay of the kinetic energy E_{kA} , normalized on the initial value E_{k0} , considering two different Reynolds numbers.

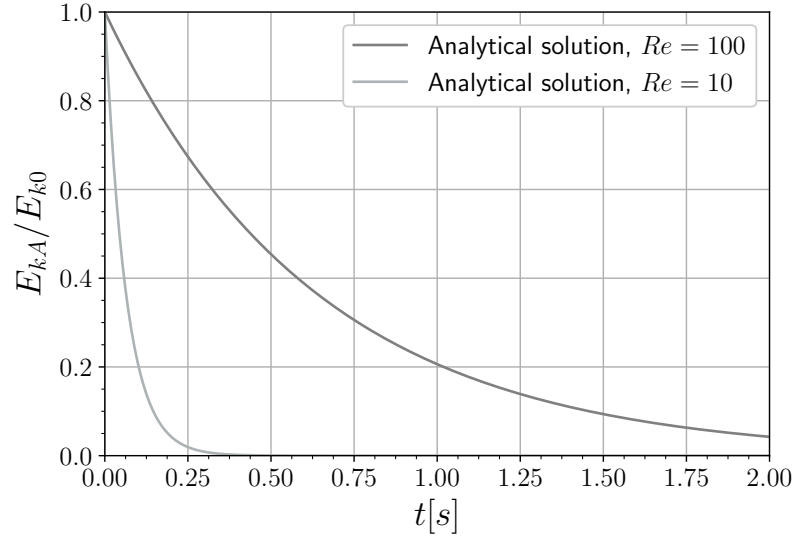


Figure 3.4 Taylor-Green flow. Two-dimensional analytical total kinetic energy.

3.2 Moving square inside a rectangular box

The first application is the 2D incompressible flow around a moving square inside a rectangular box which is a benchmark problem in the SPH community (Colicchio et al. 2006).

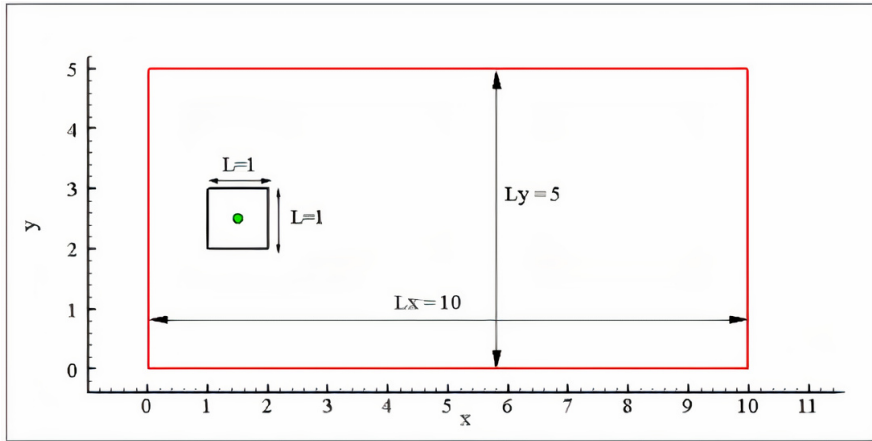


Figure 3.5 SPHERIC Benchmark Test #6, A. Colagrossi. Initial configuration.

The test case configuration is shown in Figure 3.5, the inner square is $1 \text{ m} \times 1 \text{ m}$ and its centre of mass has coordinate $x = 1.5 \text{ m}$ and $y = 2.5 \text{ m}$ while the external domain is 10 m wide and 5 m high. The fluid is initially at rest, then the object is accelerated for 1 s until it reaches the final velocity $U_{obj} = 1 \text{ m/s}$, which is maintained constant during the entire simulation. In the analysis shown later in this thesis, the reference density ρ_0 is set equal to 1000 kg/m^3 and the kinematic viscosity ν is set equal to $0.01 \text{ N/m}^2\text{s}$.

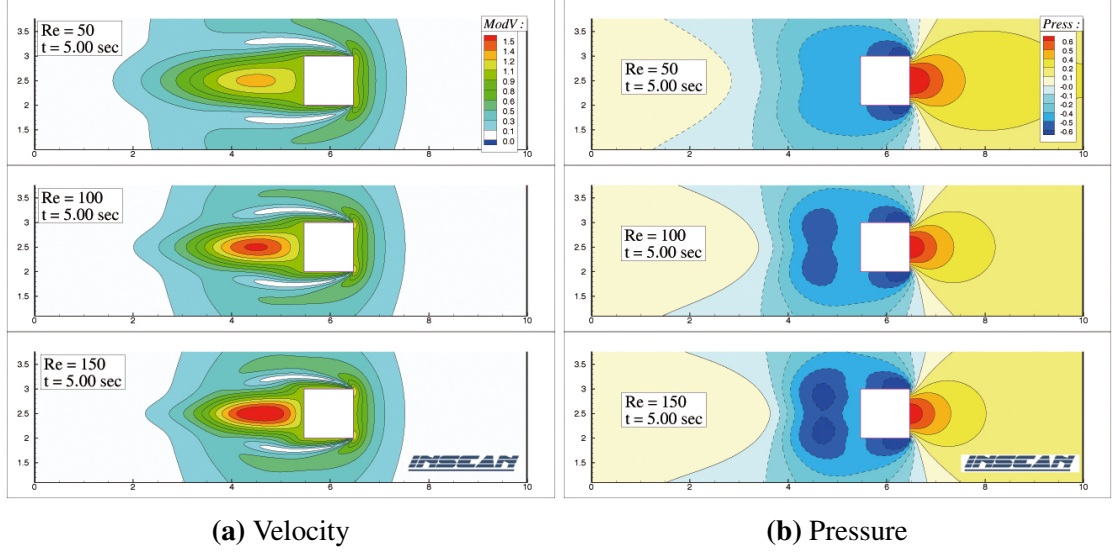


Figure 3.6 SPHERIC Benchmark Test #6, , A. Colagrossi. Contours of velocity magnitude and pressure around the moving square at physical time $t = 5$ s, using FD solver.

In the present test, the sharp edges of the moving object develop quite an intense vorticity and one of the main issues is related to the fact that the Lagrangian motion of the particles can generate voids.

3.3 Impinging jet on a flat surface

The second application is the jet impinging on a flat surface, this case has a complex analytical solution, firstly presented by Michell (1890), in addition Taylor (1966) derived an implicit solution for velocity and pressure at the wall, with an arbitrary degree of impact, which has the following formulation,

$$\begin{cases} \frac{x}{H} = \frac{1}{2}(1 + \cos \varphi) \ln \left(\frac{1+q}{1-q} \right) + \sin \varphi \sin q^{-1} \\ u = \frac{\sqrt{(1-q^2)} \sin \varphi - 1 - \cos \varphi}{q - \cos \varphi} \\ p = \frac{1}{2} \rho U^2 (1 - u^2) \end{cases} \quad (3.3)$$

where q is a free parameter in the solution, φ is the angle measured from the vertical, x is the horizontal distance, H is the characteristic length which is the half-width of the inflow section (Figure 3.7 (a)) and U is the jet velocity.

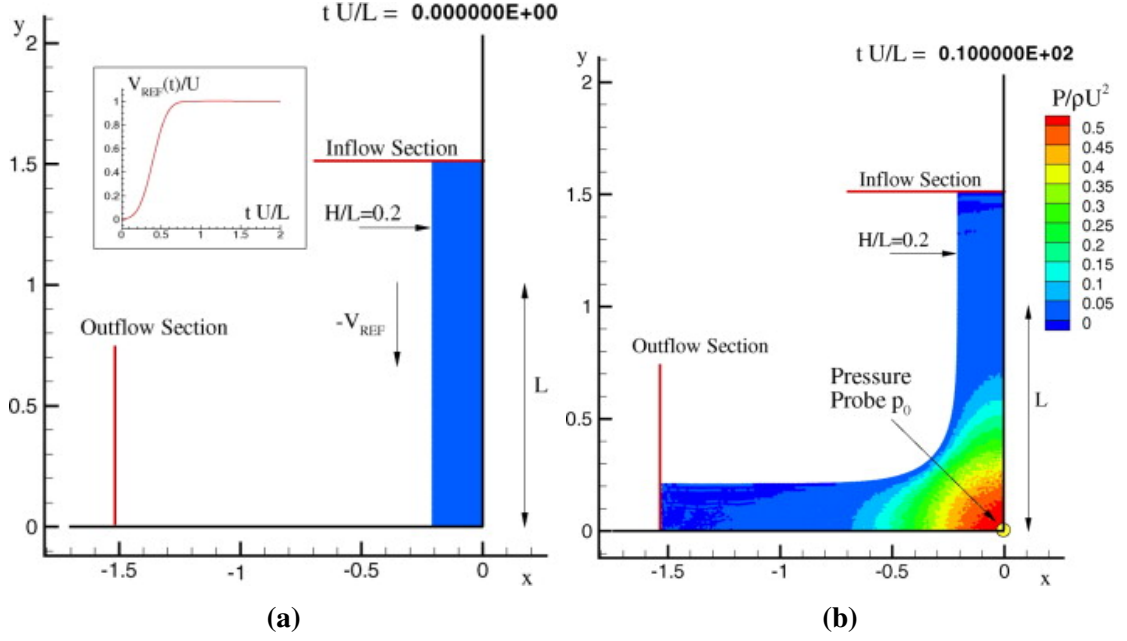


Figure 3.7 Molteni and Colagrossi (2009), (a) initial configuration and set up, (a) fluid domain and pressure field at $tU/L = 10$.

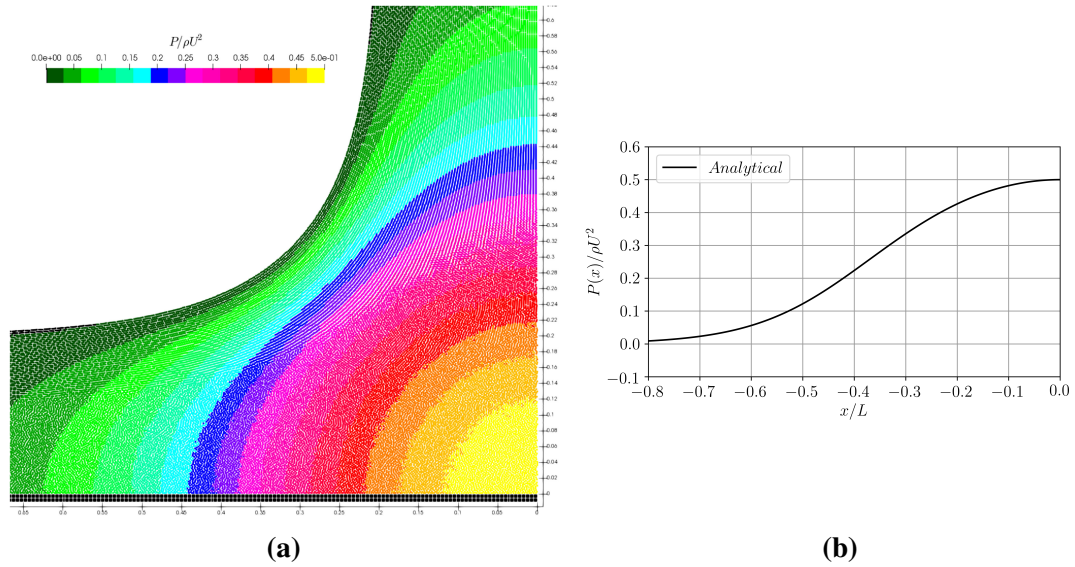


Figure 3.8 (a) Green et al. (2019) non-dimensional pressure field (b) non-dimensional pressure over the flat surface in steady conditions.

This test case is a free surface problem that can be adopted as a first attempt to simulate flow impacting rigid structures such as the ones that develop in different types of hydraulic turbines.

3.4 Numerical investigations

In this work, results of particles distribution analysis are usually presented in terms of particle concentration gradient, ∇C , detailed in Section 6.3, and its non-dimensional L_2

and L_∞ norms,

$$L_2(\nabla C) = h \sqrt{\sum_{i=1}^n \frac{\|\nabla C_i\|^2}{n}}, \quad (3.4)$$

and

$$L_\infty(\nabla C) = h(\max_i |\nabla C_i|). \quad (3.5)$$

where h is the smoothing length of the kernel function, and n is the number of particles in the computational domain. As explained later, ∇C can be seen as a measure of the disorder in particle distribution, therefore analyzing dimensionless norms allows comparing the level of perturbation even in particle distributions with different resolutions.

In numerical experiments, a periodic test function $f(x, y)$, defined in a two-dimensional squared domain, has been used to evaluate the accuracy of the spatial SPH interpolation,

$$f(x, y) = \sin\left(\frac{\pi x}{\lambda}\right) + \cos\left(\frac{\pi y}{\lambda}\right), \quad (3.6)$$

where x and y are the coordinates of a two-dimensional domain and λ can be changed to modify the steepness.

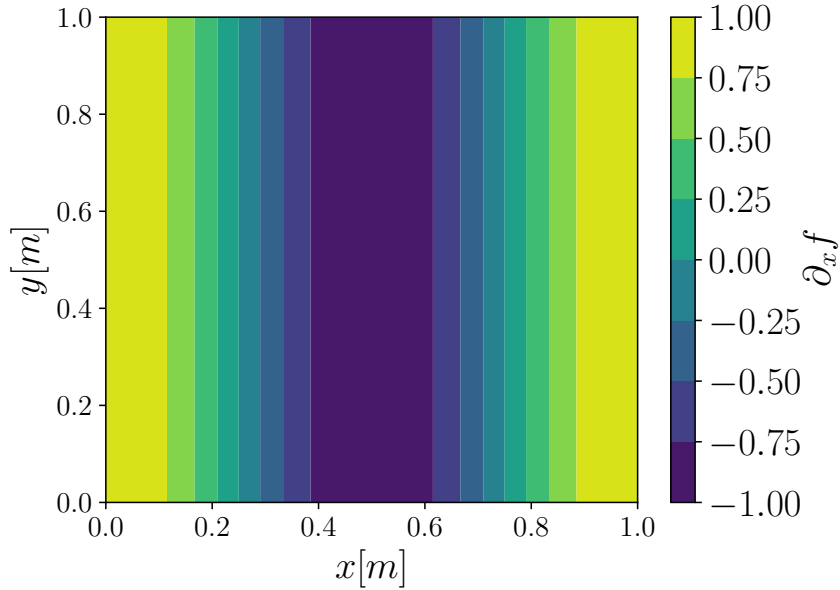


Figure 3.9 Test function x derivative.

The SPH spatial interpolation error has been assessed computing the norm,

$$L_2(\partial_x f) = \sqrt{\sum_{i=1}^n \frac{\|\partial_x f_i^{SPH} - \partial_x f_i^{an}\|^2}{n}}, \quad (3.7)$$

while in the Taylor-Green vortex the accuracy is evaluated through the error on the

velocity field,

$$L_2|\mathbf{v}_x - \mathbf{v}_{xA}| = \sqrt{\sum_{i=1}^n \frac{\|\mathbf{v}_x - \mathbf{v}_{xA}\|^2}{n}}, \quad (3.8)$$

where v_x and v_{xA} are, respectively the x components of the interpolated and the analytical velocity, Figure 3.2 (a), and Figure 3.3 (a).

Chapter 4

Smoothed Particle Hydrodynamics methodology

In the following chapter, an overview of the smoothed particle hydrodynamics models is presented. The introduction to the theoretical fundamentals is illustrated through the main steps that allow the method to be applied in fluid dynamics, extended contents are published in Liu and Liu (2003) and in Violeau (2012).

4.1 SPH theoretical fundamentals

The SPH method is a spatial interpolation technique that can be adopted to reconstruct field quantities defined over a domain, $\mathcal{D}$, using a convolution integral. The process needed to derive the basics of SPH starts by defining a scalar or vector field function $f(\mathbf{x}): \mathbb{R}^d \rightarrow \mathbb{R}$, where $\mathbf{x}$ is the position vector and d is the number of spatial components. Using the Dirac delta function, $\delta(\mathbf{x})$, the following identity can be written,

$$f(\mathbf{x}) = \int_{\Omega} f(\mathbf{x}') \delta(\mathbf{x} - \mathbf{x}') d\Omega, \quad (4.1)$$

where Ω defines the integration domain and the position vectors $\mathbf{x}$ and $\mathbf{x}'$ respectively define the interpolation and the surrounded points.

4.1.1 SPH continuous interpolation

The function expressed in equation (4.1) cannot be estimated by any numerical scheme due to the infinitesimally narrow domain of the Dirac delta function which has a zero value everywhere except at location $\mathbf{x}$,

$$\delta(\mathbf{x} - \mathbf{x}') = \begin{cases} \infty, & \mathbf{x} = \mathbf{x}' \\ 0, & \mathbf{x} \neq \mathbf{x}' \end{cases} \quad (4.2)$$

therefore equation (4.1) is useless for practical computations. It is convenient to introduce an approximation, at the continuous level, substituting $\delta(\mathbf{x})$ with the weighted function $W(\mathbf{x} - \mathbf{x}', h)$ which is called the smoothing kernel and depends on the interpolation distance between $\mathbf{x}$ and $\mathbf{x}'$ and on the smoothing (or characteristic) length

h .

The convolution integral in equation (4.1) is thus rewritten introducing the continuous approximation,

$$\langle f(\mathbf{x}) \rangle = \int_{\Omega} f(\mathbf{x}') W(\mathbf{x} - \mathbf{x}', h) d\Omega, \quad (4.3)$$

where $\langle f(\mathbf{x}) \rangle$ is the SPH approximated value. The kernel function has high relevance in the SPH models and it has to possess some important properties, described in detail in Liu and Liu (2003), and summarized below.

- The kernel function is normalized:

$$\alpha \int_{\Omega} W(\mathbf{x} - \mathbf{x}', h) d\Omega = 1, \quad (4.4)$$

a normalization coefficient α is used to ensure that the integral of the kernel function over the domain is equal to unity.

- The kernel function has a compact support:

$$W(\mathbf{x} - \mathbf{x}', h) = 0 \quad \text{if} \quad |\mathbf{x} - \mathbf{x}'| > \kappa h, \quad (4.5)$$

where κh represents the kernel radius of influence, and κ is often twice the smoothing length.

- The kernel function is symmetric:

$$W(\mathbf{x} - \mathbf{x}', h) = W(\mathbf{x}' - \mathbf{x}, h). \quad (4.6)$$

- The kernel function tends to Dirac delta function if h tends to zero:

$$\lim_{h \rightarrow 0} W(\mathbf{x} - \mathbf{x}', h) = \delta(\mathbf{x} - \mathbf{x}'). \quad (4.7)$$

- The kernel function is positive over the entire support:

$$W(\mathbf{x} - \mathbf{x}', h) > 0 \quad \text{if} \quad |\mathbf{x} - \mathbf{x}'| < \kappa h, \quad (4.8)$$

- The kernel function is k -times differentiable and the derivative is continuous:

$$W(\mathbf{x} - \mathbf{x}', h) \in \mathcal{C}_0^k. \quad (4.9)$$

In the applications presented in this thesis and in many other SPH frameworks, the above-mentioned kernel function characteristics are commonly satisfied, nevertheless,

non all of them are strictly required, e.g. Gaussian kernel (Monaghan 1992) has no compact support (equation (4.5)), high-order schemes (Nasar et al. 2021) require not-positive function (equation (4.8)).

The consistency of equation (4.3) can be estimated using a Taylor Series expansion for the generic field function $f(\mathbf{x})$ around $\mathbf{x}'$,

$$f(\mathbf{x}') = f(\mathbf{x}) + \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}(\mathbf{x} - \mathbf{x}') + \frac{1}{2} \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x}^2}(\mathbf{x} - \mathbf{x}')^2 + \mathcal{O}(\mathbf{x} - \mathbf{x}')^3. \quad (4.10)$$

It is substituted in convolution integral, equation (4.3), leading to

$$\begin{aligned} \langle f(\mathbf{x}) \rangle &= f(\mathbf{x}) \int_{\Omega} W(\mathbf{x} - \mathbf{x}', h) d\Omega + \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \int_{\Omega} W(\mathbf{x} - \mathbf{x}', h)(\mathbf{x} - \mathbf{x}') d\Omega + \\ &+ \frac{1}{2} \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x}^2} \int_{\Omega} W(\mathbf{x} - \mathbf{x}', h)(\mathbf{x} - \mathbf{x}')^2 d\Omega + \mathcal{O}(\mathbf{x} - \mathbf{x}')^3. \end{aligned} \quad (4.11)$$

Then, the generic k -moment for the kernel function is be defined as,

$$\mathcal{M}_{W,k} = \int_{\Omega} W(\mathbf{x} - \mathbf{x}', h)(\mathbf{x} - \mathbf{x}')^k d\Omega \quad (4.12)$$

due to the properties stated in equations (4.4) and (4.6), the zero-order moment $\mathcal{M}_{W,0}$ is equal to 1 and first-order moment $\mathcal{M}_{W,1}$ is equal to 0. These important assumptions allow to simplify equation (4.11) as,

$$\langle f(\mathbf{x}) \rangle = f(\mathbf{x}) + \mathcal{O}(h)^2 \quad (4.13)$$

A kernel function with the above-mentioned properties grants that the SPH approximation, (equation (4.3)), is second-order accurate in space at continuous level, (Monaghan 1992), (Amicarelli et al. 2011), meaning that the method is able to reproduce exactly zeroth and first-order polynomial functions. Nevertheless, the order of accuracy of the SPH interpolation can be moved up to higher orders adopting modified kernel functions.

Several authors, Monaghan (1985), Chaniotis and Poulikakos (2004), Lind and Stansby (2016), have introduced kernel functions that can be used to achieve higher orders of consistency. Equation (4.11) is extended and rewritten substituting equation (4.12) as

$$\begin{aligned} \langle f(\mathbf{x}) \rangle &= f(\mathbf{x})\mathcal{M}_{W,0} + \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\mathcal{M}_{W,1} + \frac{1}{2!} \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x}^2}\mathcal{M}_{W,2} + \\ &+ \frac{1}{3!} \frac{\partial^3 f(\mathbf{x})}{\partial \mathbf{x}^3}\mathcal{M}_{W,3} + \mathcal{O}(\mathbf{x} - \mathbf{x}')^4. \end{aligned} \quad (4.14)$$

In order to remove the error term greater than second-order, the property in equation (4.4) can not be satisfied, meaning that the kernel function needs to assume negative

values. For $\mathcal{M}_{W,2}$ guaranteed equals to zero, SPH reaches higher accuracy order,

$$\langle f(\mathbf{x}) \rangle = f(\mathbf{x}) + \mathcal{O}(h)^4 \quad (4.15)$$

The same procedure could be applied to higher order, improving arbitrarily the SPH consistency. However, introducing not positive defined kernel functions, which are not monotonically decreasing, can produce numerical issues in purely Lagrangian SPH schemes, in particular, in the case of free-surface flow, (Nasar et al. 2021).

Moreover, the major point of interest in the SPH method is its capability to reproduce the gradient of field quantities using exclusively the field values and the gradient of the kernel function. Indeed, the function gradient SPH approximation, equation (4.3), is rearranged as

$$\langle \nabla f(\mathbf{x}) \rangle = \int_{\Omega} \nabla f(\mathbf{x}') W(\mathbf{x} - \mathbf{x}', h) d\Omega. \quad (4.16)$$

Then, recalling the product rule for derivatives

$$\nabla[f(\mathbf{x})W(\mathbf{x} - \mathbf{x}', h)] = \nabla f(\mathbf{x})W(\mathbf{x} - \mathbf{x}', h) + f(\mathbf{x})\nabla W(\mathbf{x} - \mathbf{x}', h), \quad (4.17)$$

it can be used to substitute equation (4.17) in (4.16) obtaining

$$\langle \nabla f(\mathbf{x}) \rangle = \int_{\Omega} \nabla[f(\mathbf{x}')W(\mathbf{x} - \mathbf{x}', h)]d\Omega - \int_{\Omega} f(\mathbf{x}')\nabla W(\mathbf{x} - \mathbf{x}', h) d\Omega. \quad (4.18)$$

The Gauss theorem is applied to the first term of equation (4.18),

$$\langle \nabla f(\mathbf{x}) \rangle = \int_S [f(\mathbf{x}')W(\mathbf{x} - \mathbf{x}', h)]\mathbf{n} dS - \int_{\Omega} f(\mathbf{x}')\nabla W(\mathbf{x} - \mathbf{x}', h) d\Omega \quad (4.19)$$

where $\mathbf{n}$ is the vector normal to the integration surface S . Thus, the gradient of the field quantity is computed by two terms associated with the values of the field itself. Under the hypothesis that the kernel support is fully extended inside the computational domain, meaning that the integration region is not intersected by boundaries, the first term is neglected because the kernel function is null at the extreme of its support, heading to

$$\langle \nabla f(\mathbf{x}) \rangle = - \int_{\Omega} f(\mathbf{x}')\nabla W(\mathbf{x} - \mathbf{x}', h)d\Omega \quad (4.20)$$

The symmetry condition allows to change sign, leading to the SPH continuous approximation for the gradient of a function,

$$\langle \nabla f(\mathbf{x}) \rangle = \int_{\Omega} f(\mathbf{x}')\nabla W(\mathbf{x} - \mathbf{x}', h)d\Omega \quad (4.21)$$

This formulation represents the core of the SPH methods, the computation of the field

derivatives can be carried out using the convolution integral between the field quantity and the kernel function derivative.

Under the same conditions, the accuracy for the SPH gradient approximation can be evaluated, as previously, substituting equation (4.21) in Taylor Series expansion, similarly to equation (4.10), defining the generic moment for the gradient function as

$$\mathcal{M}_{\nabla W, k} = \int_{\Omega} \nabla W(\mathbf{x} - \mathbf{x}', h) (\mathbf{x} - \mathbf{x}')^k d\Omega. \quad (4.22)$$

It is possible to show, (Marongiu et al. 2010), that the error of the gradient field at the continuous level is first-order consistent and it converges as $\mathcal{O}(h)^2$.

4.1.2 Kernel functions

As previously highlighted, the choice of the kernel function has a relevant impact on the SPH interpolation and, therefore, on the reconstruction of the field quantities (and their gradient). In Section 4.1 the main properties required to guarantee first-order consistency have been presented, different kernel functions that satisfy these characteristics have been proposed in SPH, formulated with the generic expression,

$$W(x, h) = \frac{c}{h^d} f\left(\frac{x}{h}\right) \quad (4.23)$$

where $q = x/h$ is the ratio between the distance from the origin and smoothing length h ; as previously defined, d is the number of spatial dimensions and c is a constant that allows to specify the coefficient $\alpha = c/h^d$, which verifies the normalization condition, equation (4.4).

In their first publication, Gingold and Monaghan (1977) adopted a Gaussian kernel defined as

$$W(q) = \alpha e^{(-q^2)}, \quad (4.24)$$

where $\alpha = 1/(h\sqrt{\pi}), 1/(h^2\pi), 1/(h^3\pi^{3/2})$ is the normalization factor in one, two, three dimensions. The Gaussian kernel is able to closely approximate the Dirac delta function, meaning that the smoothing approximation is minimized, but its support is infinitely extended and, for this reason, its usage is computationally expensive.

The third-order Cubic spline has been later introduced by Monaghan and Lattanzio (1985),

$$W(q) = \alpha \begin{cases} 1 - \frac{3}{2}q^2 + \frac{3}{4}q^3 & 0 \leq q \leq 1 \\ \frac{1}{4}(2 - q)^3 & 1 \leq q \leq 2 \\ 0 & \text{otherwise} \end{cases} \quad (4.25)$$

where $\alpha = 2/(3h), 10/(7h^2\pi), 1/(h^3\pi)$ is the normalization factor in one, two, three

dimensions. It accurately reproduces the Gaussian kernel even though it has a compact support and it is, therefore, computationally cheaper. The third-order Cubic spline is piecewise-defined and it is differentiable two times, Quinlan et al. (2006) and Amicarelli et al. (2011) have demonstrated that the smoothness of the kernel function affects the SPH interpolation errors. Differently, the Wendland kernel, (Schaback and Wendland 2006), which are able to prevent the pairing instabilities (Dehnen and Aly 2012), see Section 2.4.2, are usually preferred in fluid simulations, (it is the case for this project). These functions, generally defined as

$$W(x, h) = \frac{c}{h^d} f\left(\frac{x}{2h}\right), \quad (4.26)$$

where $q = x/2h$, have been constructed using high-order polynomials and, therefore, they are able to capture higher-order effects improving the accuracy.

The Wendland kernel C2 is

$$W(q) = \alpha \begin{cases} (1 - q)^4 (4q + 1) & q \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad (4.27)$$

with $\alpha = 5/8, 7/(4\pi), 21/(16\pi)$ in 1D, 2D and 3D.

The Wendland kernel C4 is

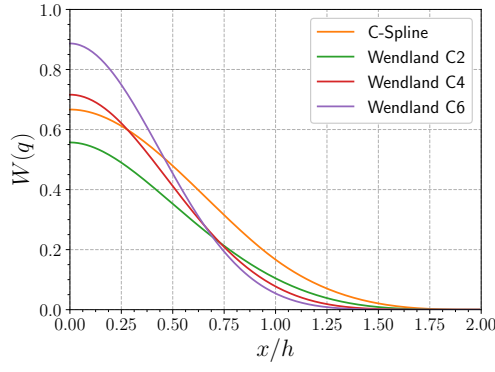
$$W(q) = \alpha \begin{cases} (1 - q)^6 (35q^2 + 18q + 3) & q \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad (4.28)$$

with $\alpha = 3/4, 3/(4\pi), 165/(256\pi)$ in 1D, 2D and 3D.

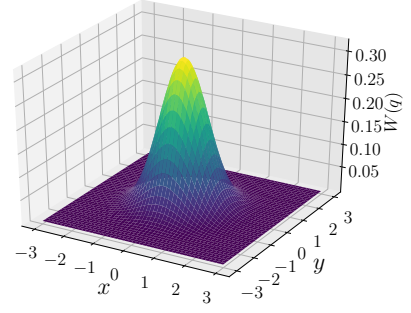
The Wendland kernel C6 is

$$W(q) = \alpha \begin{cases} (1 - q)^8 (35q^3 + 25q^2 + 8q + 1) & q \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad (4.29)$$

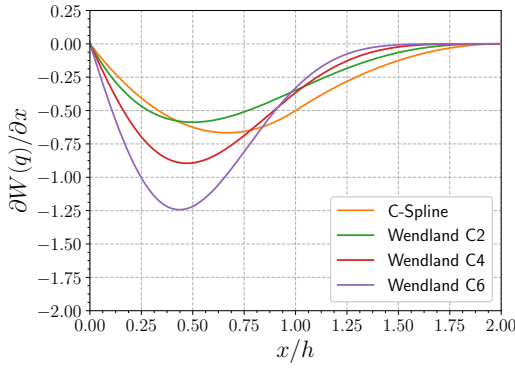
with $\alpha = 55/64, 38/(14\pi), 1365/(512\pi)$ in 1D, 2D and 3D.



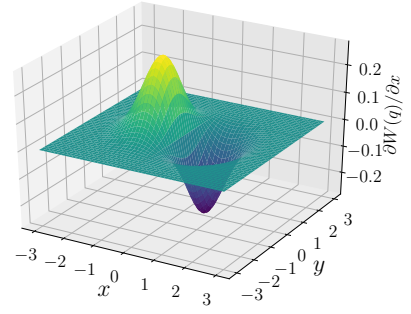
(a) Kernel functions 1D



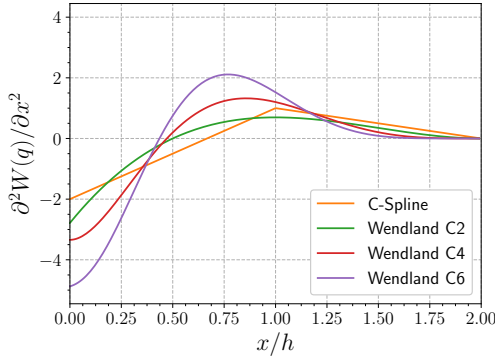
(b) Gaussian kernel 2D



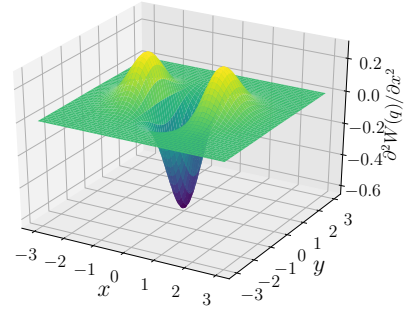
(c) First derivative kernel functions 1D



(d) First derivative Gaussian kernel 2D



(e) Second derivative kernel functions 1D



(f) Second derivative Gaussian kernel 2D

Figure 4.1 (left) Kernel functions in 1D. (right) Gaussian kernel in 2D

In Figure 4.1 can be seen the shape of the kernel functions and it can be noted that the second derivatives of the Wendland kernels remain derivable, however, despite the fact that all these kernel functions are compactly supported, the value of W varies significantly among them. For example, the Wendland C6, (equation (4.29)), gives greater weight to points close to the origin, while the Cubic spline, (equation (4.25)) assigns more weight to distant particles compared to other functions. For this reason, different results for the interpolations are expected and the kernel function has to be

carefully chosen.

4.1.3 SPH discrete interpolation

The second main step to derive the SPH methodology is the discrete approximation of the convolution integral previously presented in equation (4.3). In the interior of the domain, the equations (4.3) and (4.21) are discretized as

$$\begin{aligned}\langle f(\mathbf{x}_i) \rangle &= \int_{\Omega} f(\mathbf{x}') W(\mathbf{x} - \mathbf{x}', h) d\Omega \\ &\approx \sum_{j=1}^J f(\mathbf{x}_j) W(\mathbf{x}_i - \mathbf{x}_j, h) \omega_j\end{aligned}\tag{4.30}$$

$$\begin{aligned}\langle \nabla f(\mathbf{x}_i) \rangle &= \int_{\Omega} f(\mathbf{x}') \nabla W(\mathbf{x} - \mathbf{x}', h) d\Omega \\ &\approx \sum_{j=1}^J f(\mathbf{x}_j) \nabla W(\mathbf{x}_i - \mathbf{x}_j, h) \omega_j\end{aligned}\tag{4.31}$$

where i represents the calculation point, j defines the generic interpolation point with the associated discrete volume ω_j , which for a uniform particle size is defined as $\omega_j = \Delta_i^d$, and J is the total number of particles within the support of particle i .

In grid-based schemes, the numerical solution tends toward the exact solution while the mesh size tends to zero. In meshless methods, the inter-particle distance Δ defines the resolution of the discretized domain. Additionally, in SPH schemes, the smoothing length h determines the number of particles that fall into the kernel radius of influence, which corresponds to the number of interpolation points. These parameters have a key role in the method spatial convergence and both have to tend to zero. The ratio between h and Δ rules the number of interpolation points in the kernel support; Raviart (1985) showed that it has to increase while the discretization parameters tend to zero. The set of conditions involved in the convergence of SPH interpolation is,

$$\begin{cases} \Delta \rightarrow 0 \\ h \rightarrow 0 \\ \frac{h}{\Delta} \rightarrow \infty \end{cases}\tag{4.32}$$

In practice, the ratio h/Δ has a finite value and the choice has to be balanced between numerical error and computational costs. The smoothing error is reduced while h tends to zero, equation (4.7), whereas the discretization error is lessened while h/Δ increases, gaining accuracy but raising the computational time.

4.1.4 Convergence analysis of an SPH interpolation

At discrete level, the SPH interpolations, equations (4.30) and (4.31), can reach the theoretical convergence rate if the smoothing error is much larger than the discretisation error, conversely, if the discretization error is greater than the smoothing error, the second-order convergence rate is no longer achieved. To restore first-order consistency the following equations need to be fulfilled,

$$\sum_{j=1}^J W(\mathbf{x}_i - \mathbf{x}_j, h) \omega_j = 1 \quad (4.33)$$

$$\sum_{j=1}^J \nabla W(\mathbf{x}_i - \mathbf{x}_j, h) \omega_j = 0 \quad (4.34)$$

but for significant disorder in particle distribution, these conditions are not satisfied. In discrete SPH approximation, the evaluation of the accuracy is a really complex topic, because, as stated previously, all the conditions presented in equations (4.32) are involved in the convergence.

The accuracy for the SPH interpolation at the discrete level depends on the particle distribution, Quinlan et al. (2006) demonstrated that in one dimension, for particles uniformly spaced, the difference between the analytical and the approximated values can be estimated as

$$\frac{\partial f(\mathbf{x}_i)}{\partial x} - \langle \frac{f(\mathbf{x}_i)}{\partial x} \rangle = h^2 \frac{\partial^3 f(\mathbf{x}_i)}{\partial x^3} \int_{\Omega} q^2 W(q) dq + C \left(\frac{\Delta}{h} \right)^{\beta+2}. \quad (4.35)$$

The SPH error is the sum of a second-order error in h (smoothing error) and a $\beta+2$ order error in Δ/h (discretization error), where β represents the smoothness of the kernel at the boundaries.

To analyse these aspects a convergence analysis of the test function derivative has been performed using equation (4.31) and adopting the Wendland C6 kernel, (equation (4.29)).

The accuracy of the spatial SPH interpolation in a two-dimensional domain, see Section 3.4, has been evaluated using the test function (3.6) and the test function gradient error norm (3.7). The analysis has been conducted using a Cartesian particle distribution and a perturbed particle distribution, obtained applying a normalized random perturbation, σ/Δ , to each particle, originally placed in a Cartesian grid, whereas three values of h/Δ have been tested. The results of these numerical experiments are shown in Figure 4.2 through the L_2 norm of the derivative error against the particle spacing Δ . In experiments with a perturbation, $\sigma/\Delta = 0.1$, the error remains constant and does not decrease with Δ , whereas, in tests with a Cartesian distribution the theoretical

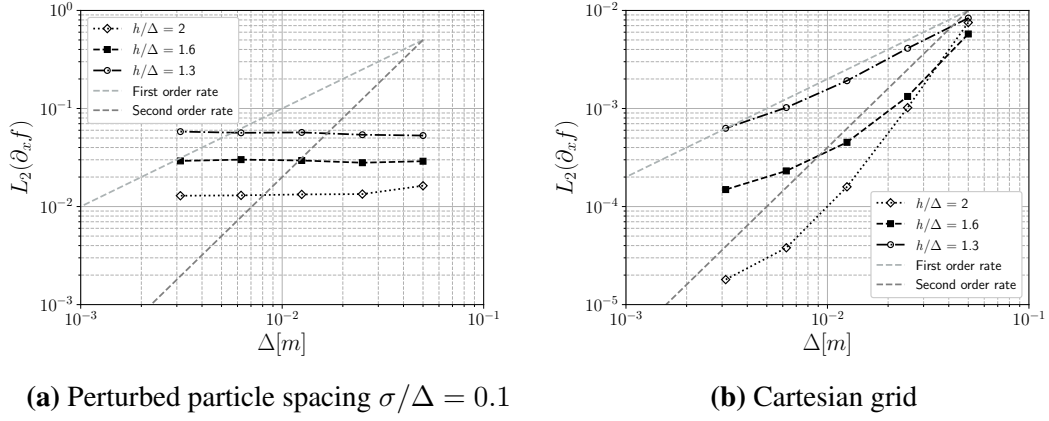


Figure 4.2 Numerical test. L_2 error norm in SPH estimations of first partial derivative for a test function, (a) perturbed particle spacing $\sigma/\Delta = 0.1$, (b) Cartesian grid.

second-order convergence rate is restored.

4.2 Summary and conclusions

In this chapter the SPH fundamentals have been illustrated, then the method accuracy has been presented from a theoretical point of view and through numerical evaluations. At the discrete level, the global scheme precision is strictly related to the particle distribution, the interpolation points.

Since in SPH, (Monaghan 1994), the particles move accordingly to the Lagrangian trajectories, the particle distribution cannot be controlled during the simulation leading to poor interpolation accuracy. In conclusion, it has been once more confirmed that proper techniques that aim to minimize the particle spatial perturbation are crucial to maintaining the SPH accuracy.

Chapter 5

ALE-SPH model for Weakly-Compressible fluids

In this chapter, the review of the main steps used to discretize the set of equations that governs fluid dynamics within the SPH formalism is presented. Different SPH schemes can be adopted, the original formalism introduced by Monaghan (1992) is presented, then the focus is posed on the process needed to derive the SPH equations in the Weakly-Compressible Arbitrarily Eulerian-Lagrangian framework. This latest model has been implemented in the code ASPHODEL, (Marongiu 2007; Leduc 2010; Neuhauser 2014; Pineda Rondon 2017), whose structure is presented later in the chapter because it has been widely utilized in this project.

5.1 Governing equations for fluid dynamics

The motion of fluids is a physical phenomenon that has always attracted interest and it has been widely investigated for many centuries. Nowadays, the flow of Newtonian fluids is described with a macroscopic approach using the Navier-Stokes equations, presented as a system of partial differential equations, whose solution is able to predict the evolution of density, velocity, pressure and temperature in fluids.

In thermo-fluid dynamics the equations for continuity, momentum and energy conservation, expressed with the conservative differential form, are:

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \\ \frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \nabla \cdot \boldsymbol{\tau} + \rho \mathbf{f}_e \\ \frac{\partial \rho E}{\partial t} + \nabla \cdot (\rho \mathbf{v} E) = -p \nabla \cdot \mathbf{v} + \boldsymbol{\tau} \nabla \cdot \mathbf{v} + \nabla \cdot (k \nabla T) + \mathbf{q}_c \end{cases} \quad (5.1)$$

where ρ and $\mathbf{v}$ are respectively the density and the velocity of the fluid, p is the pressure field, e is the internal energy and T is the temperature, the external forces and heat sources are $\mathbf{f}_e$ and $\mathbf{q}_c$. The total energy E is defined as the sum of internal energy e and kinetic energy,

$$E = e + \frac{1}{2} \mathbf{v}^2. \quad (5.2)$$

The intrinsic fluid properties are described by μ , the dynamic viscosity or ν the kinematic viscosity (with the relation $\mu = \rho\nu$), while k is the thermal conductivity and τ , describes the viscous stress tensor.

The system of equations represents a fully coupled and non-linear problem, a smooth analytical solution is not currently available, excluding some simplified cases in which are specified peculiar initial and boundary conditions (an exact analytical solution is known for the two-dimensional Taylor Green vortex for incompressible flow, as seen in Section 3.1).

Nevertheless, it can be approximated using numerical modelling, (Section 2.1), to compute the physical quantities, $\rho, p, \mathbf{v}$ and T in a finite number of computational points, but, even using highly sophisticated numeral models, obtaining the approximated solution can be extremely difficult and computationally expensive.

For this reason, the system (5.1) can be rewritten under certain simplifying hypotheses to reduce the complexity of the problem, without drastically tightening the range of applications, as introducing a closure model for fluid to link the internal energy with pressure, density, generally stated as $e = e(p, \rho)$. As an additional simplification, the temperature can be assumed constant to remove the equation of energy conservation from the system: in the problems which are investigated in this project the thermal behaviour is not taken into account because it does not play any significant role.

Moreover, in studies involving water flows, the fluid can be simulated with different assumptions regarding compressibility behaviour. For incompressible models, the continuity equations can be rewritten assuming that density remains constant,

$$\nabla \cdot \mathbf{v} = 0. \quad (5.3)$$

Differently, in Monaghan (1994) water is modelled as weakly-compressible fluid in which the range of density variation is kept close to 1%, an equation of state is needed to couple the equations of mass and momentum conservation, defined as $p = p(\rho)$, and specifically using the barotropic Tait's equation

$$p = \frac{c_0^2 \rho_0}{\gamma} \left[\left(\frac{\rho}{\rho_0} \right)^\gamma - 1 \right] + p_0, \quad (5.4)$$

where $\gamma = 7$ is the polytropic fluid coefficient and c_0 , ρ_0 and p_0 are respectively the reference speed of sound, the reference density and the reference pressure (which is usually set equal to zero). This equation is highly representative of water compressibility in the case in which the reference speed of sound is defined as the physical speed of sound ($c_0 \approx 1480 \text{ m/s}$), however, this assumption requires to adopt a time step that

fulfills the Courant-Friedrichs-Lewy condition written as:

$$\frac{(c_0 + \|\mathbf{v}\|)\Delta t}{\Delta} \leq CFL \quad (5.5)$$

where the coefficient CFL in explicit time integration schemes is set less than one, to capture any acoustic wave that passes through the discrete particle. To avoid the computational overhead, in most of the SPH practical applications is usually imposed $c_0 \approx 10\|\mathbf{v}\|$, maintaining the fluid model still in the range of weakly-compressibility, but allowing a greater time step, under this condition the Mach number Ma , defined as $Ma = \|\mathbf{v}\|/c_0$ remains around 0.1.

Important assumptions on the physical fluid viscosity need to be specified, in Newtonian fluid, the viscous stresses are proportional to the fluid velocity rate of change, many common fluids, like water, can fit in this viscosity model and if the dynamic viscosity μ is constant the viscous stress is defined by

$$\boldsymbol{\tau} = \mu \nabla \cdot \mathbf{v}. \quad (5.6)$$

As previously mentioned, these considerations are taken into account to reduce the complexity of the problem described in equations (5.1), which are rewritten in the following simplified form:

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \\ \frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \nabla \cdot \boldsymbol{\tau} + \rho \mathbf{f}_e \end{cases} \quad (5.7)$$

while the vector form for equations (5.7) is

$$\frac{\partial \Phi}{\partial t} + \nabla \cdot (\mathbf{F}_c + \mathbf{F}_\nu) = \mathbf{Q}. \quad (5.8)$$

defining

$$\Phi = \begin{pmatrix} \rho \\ \rho \mathbf{v} \end{pmatrix} \quad \mathbf{F}_c = \begin{pmatrix} \rho \mathbf{v} \\ \rho \mathbf{v} \otimes \mathbf{v} + p \mathbf{I} \end{pmatrix} \quad \mathbf{F}_\nu = \begin{pmatrix} 0 \\ \boldsymbol{\tau} \end{pmatrix} \quad \mathbf{Q} = \begin{pmatrix} 0 \\ \rho \mathbf{f}_e \end{pmatrix} \quad (5.9)$$

where Φ represents the conservatives field variables vector. $\mathbf{F}_c$ and $\mathbf{F}_\nu$ are respectively the convective and the viscous stress tensors, $\mathbf{I}$ the identity matrix and $\mathbf{Q}$ are the source terms which contain the external forces that may act on the fluid, for real applications it includes the gravity force $\mathbf{g}$.

Whenever fluids are considered inviscid, excluding viscosity forces, equations (5.7)

is rewritten as

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \\ \frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \mathbf{f}_e \end{cases} \quad (5.10)$$

called Euler equations, or, similarly to equation (5.8), in vector form,

$$\frac{\partial \Phi}{\partial t} + \nabla \cdot (\mathbf{F}_c) = \mathbf{Q} \quad (5.11)$$

Therefore, neglecting the effects of the viscosity, the numerical model is reduced both in terms of physics and in terms of mathematics.

As a recap, the natural phenomena of fluids flow can be described in different ways and it can be translated into different numerical schemes, in Section 5.3 the above-mentioned models are discretized using the SPH schemes.

5.2 Arbitrary Lagrangian-Eulerian

An initial overview of the different approaches that can be used in fluid dynamics has been presented in Section 2.2. The Eulerian approach describes the fluid flow in a fixed frame of reference, single particles of fluid are not identified, instead, a control volume is defined and all other flow properties are computed within this volume. Conversely, the Lagrangian approach describes the fluid flow in a frame of reference attached to the particles that carry the flow properties while moving in time and space.

A mixed approach that is able to generalize both the Lagrangian and the Eulerian description has been presented by Hirt et al. (1974). In this technique, called Arbitrary Lagrangian-Eulerian, the velocity of the frame of reference $\mathbf{v}_0$ is additionally introduced and it can be arbitrarily defined. The Eulerian or Lagrangian description can be recovered by setting $\mathbf{v}_0$ respectively equal to zero or to the fluid velocity $\mathbf{v}$. The arbitrary or transport velocity adds flexibility to the numerical schemes as a degree of freedom. In this thesis different options and strategies to define $\mathbf{v}_0$ are analyzed, (Neuhauser 2014).

The Euler equations are presented in the ALE conservative form using the flux vector notation:

$$L_{\mathbf{v}_0}(\Phi) + \nabla \cdot (\mathbf{F}_c - \mathbf{v}_0 \otimes \Phi) = \mathbf{Q}, \quad (5.12)$$

where the transport velocity operator $L_{\mathbf{v}_0}$ is defined as

$$L_{\mathbf{v}_0}(\Phi) = \frac{\partial \Phi}{\partial t} + \nabla \cdot (\mathbf{v}_0 \otimes \Phi), \quad (5.13)$$

and it can be replaced in equation (5.12),

$$\frac{\partial \Phi}{\partial t} + \nabla \cdot (\mathbf{v}_0 \otimes \Phi) + \nabla \cdot (\mathbf{F}_c - \mathbf{v}_0 \otimes \Phi) = \mathbf{Q}. \quad (5.14)$$

To avoid a recursive formulation a compact notation for the flux tensor $\mathbf{F}$ is introduced,

$$\mathbf{F}(\Phi, \mathbf{v}_0) = \mathbf{F}(\Phi) - (\mathbf{v}_0 \otimes \Phi). \quad (5.15)$$

In equation (5.14), the Leibniz-Reynolds transport theorem is applied on the convective term whereas the Gauss divergence theorem is imposed on the flux term passing from the volume integral over Ω , to the surface integral on $\partial\Omega$,

$$\frac{d}{dt} \int_{\Omega} \Phi \, d\Omega + \int_{\partial\Omega} \mathbf{F}_c(\Phi, \mathbf{v}_0) \cdot \mathbf{n} \, dS = \int_{\Omega} \mathbf{Q} \, d\Omega. \quad (5.16)$$

In the ALE framework the integration domain moves accordingly to the transport velocity $\mathbf{v}_0$, this additional degree of freedom can increase the accuracy of the simulation. In Section 5.4, this approach has been used to fully derive the governing equation for fluid motion in the ALE formalism.

5.3 Weakly-Compressible SPH model

The SPH numerical model has been firstly applied for fluid dynamic by Monaghan (1992), as a Lagrangian scheme, the particle trajectory is defined by the field velocity:

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i. \quad (5.17)$$

The Euler equations, defined in the system (5.10), are discretized in the classical SPH formulation as,

$$\begin{cases} \frac{d\rho_i}{dt} = \sum_{j=1}^J (\mathbf{v}_i - \mathbf{v}_j) \cdot \nabla W_{ij} m_j \\ \frac{d\mathbf{v}_i}{dt} = - \sum_{j=1}^J \left(\frac{p_j}{\rho_j^2} + \frac{p_i}{\rho_i^2} \right) \nabla W_{ij} m_j + \mathbf{g} \end{cases} \quad (5.18)$$

where m_i is the mass of the particle i , which can be used to compute the volume, $\omega_i = m_i/\rho_i$. Additionally, following the Weakly-Compressible approach, the equation of state (5.4) closes the system.

The set of equations (5.17) and (5.18) present some numerical physical issues, to prevent particles mixing and penetration, the XSPH scheme, see Section 6.1, has been proposed by Monaghan (1994), introducing, in the continuity equation, a factor that averages the velocity field. For stability reasons, numerical dissipation is introduced in the momentum equation as artificial viscosity Π^{ij} , (Monaghan 1992). Although the classical formulation of equations (5.17) and (5.18) has been widely used, and several advancements have been introduced to improve the quality of the pressure field, (Marone et al. 2011; Lind et al. 2012; Oger et al. 2016; Green et al. 2019), in this work a

different SPH formalism, called SPH-ALE, is followed.

5.4 Weakly-Compressible ALE-SPH model

The Arbitrary Lagrangian-Eulerian model has been adopted for the first time in the WCSPH framework by Vila (1999). This section reviews the formulation presented by Vila (1999) and then taken up in the PhD works of Marongiu (2007), Leduc (2010), Neuhauser (2014) and Pineda Rondon (2017), which are outlined to introduce the ALE-SPH solver used in the framework of this project. It is worth underling that in classical SPH the domain is represented by particles that move according to their kinematics, equation (5.17), and carry their own mass, while their volumes can change even though specific information on their topological deformations are not known. Conversely, in ALE-SPH schemes particle volumes move accordingly to their arbitrary velocity,

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}_0. \quad (5.19)$$

The calculation points do not correspond anymore to the actual material points, for this reason, physical fluxes between neighbours particles need to be taken into account. Following, the governing equations are rewritten with a different formulation coherently with the ALE description. Neglecting the source term, the Euler equations (5.10), written in conservative form, are a hyperbolic system that can be integrated over a defined volume Ω , which is free to move and deforms,

$$\int_{\Omega} \left[\frac{\partial \Phi}{\partial t} + \nabla \cdot \mathbf{F}_c(\Phi) \right] d\Omega = 0. \quad (5.20)$$

The vectors (5.9) of conserved variables and physical fluxes are expanded for the three dimensions:

$$\Phi = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \end{bmatrix}, \quad \mathbf{F}_c(\Phi) = \begin{bmatrix} \rho u & \rho v & \rho w \\ \rho u^2 + p & \rho uv & \rho uw \\ \rho uv & \rho v^2 + p & \rho vw \\ \rho uw & \rho vw & \rho w^2 + p \end{bmatrix}. \quad (5.21)$$

Using the Leibniz-Reynolds transport theorem, through the transport velocity $\mathbf{v}_0$,

$$\frac{d}{dt} \int_{\Omega} \Phi d\Omega = \int_{\Omega} \frac{\partial \Phi}{\partial t} d\Omega + \int_{\partial\Omega} \Phi (\mathbf{v}_0 \cdot \mathbf{n}) dS, \quad (5.22)$$

and the divergence theorem,

$$\int_{\Omega} \nabla \cdot (\Phi \otimes \mathbf{v}_0) d\Omega = \int_{\partial\Omega} \Phi (\mathbf{v}_0 \cdot \mathbf{n}) dS, \quad (5.23)$$

the equation (5.20) is rewritten as

$$\frac{d}{dt} \int_{\Omega} \Phi d\Omega + \int_{\Omega} \nabla [\mathbf{F}_c(\Phi) - \mathbf{v}_0 \otimes \Phi] d\Omega = 0, \quad (5.24)$$

or, adopting the ALE formalism introduced in equation (5.15), as

$$\frac{d}{dt} \int_{\Omega} \Phi d\Omega + \int_{\Omega} \nabla [\mathbf{F}_c(\Phi, \mathbf{v}_0)] d\Omega = 0. \quad (5.25)$$

This latest equation is defined at the continuous level for a generic control volume; at this point the continuum is partitioned in a set of distinct volumes, called particles, then equation (5.25) is specified for particle i ,

$$\frac{d}{dt} \int_{\Omega_i} \Phi d\Omega + \int_{\Omega_i} \nabla [\mathbf{F}_c(\Phi, \mathbf{v}_0)] d\Omega = 0. \quad (5.26)$$

In the discretized domain, the volumes of particle i is defined,

$$\omega_i = \int_{\Omega_i} d\Omega, \quad (5.27)$$

and it is thought to be centered on the material point described by i . Therefore, any discrete quantities Φ_i are computed as the average over the particle volume ω_i , without any information about the shape,

$$\Phi_i = \frac{1}{\omega_i} \int_{\Omega_i} \Phi d\Omega \quad \rightarrow \quad \omega_i \Phi_i = \int_{\Omega_i} \Phi d\Omega \quad (5.28)$$

Taking into account these considerations, the ALE form of the Euler equation (5.26) is rewritten using equation (5.28) in the vector notation,

$$\frac{d(\omega_i \Phi_i)}{dt} + \omega_i \nabla \cdot [\mathbf{F}_c(\Phi_i, \mathbf{v}_{0i})] = 0. \quad (5.29)$$

or fully expanded,

$$\frac{d(\omega_i \Phi_i)}{dt} + \omega_i \nabla \cdot [\mathbf{F}_c(\Phi_i) - \mathbf{v}_{0i} \otimes \Phi_i] = 0. \quad (5.30)$$

These steps are preliminary to introduce, in equation (5.30), a discrete approximation using ALE-SPH formalism for,

$$\nabla \cdot [\mathbf{F}_c(\Phi_i) - \mathbf{v}_{0i} \otimes \Phi_i]. \quad (5.31)$$

Applying the classical SPH interpolation is not recommended,

$$\omega_i \nabla \cdot [\mathbf{F}_c(\Phi_i, \mathbf{v}_{0i})] = \omega_i \sum_{j=1}^J \mathbf{F}_c(\Phi_j, \mathbf{v}_{0j}) \nabla W_{ij} \omega_j, \quad (5.32)$$

it can be seen that equation (5.32) loses conservative property, mutual fluxes between particle i and particle j do not cancel out,

$$\omega_i \omega_j \mathbf{F}_c(\Phi_i, \mathbf{v}_{0i}) \nabla W_{ij} \neq \omega_j \omega_i \mathbf{F}_c(\Phi_j, \mathbf{v}_{0j}) \nabla W_{ji}. \quad (5.33)$$

For this reason, the identity,

$$\nabla \cdot \mathbf{v}_{0i} = \nabla \cdot \mathbf{v}_{0i} - \mathbf{v}_{0i} \nabla 1, \quad (5.34)$$

is used to rewrite equation (5.31) as

$$\nabla [\mathbf{F}(\Phi_i) - \mathbf{v}_{0i} \otimes \Phi_i] = \nabla [\mathbf{F}(\Phi_i) - \mathbf{v}_{0i} \otimes \Phi_i] + [\mathbf{F}(\Phi_i) - \mathbf{v}_{0i} \otimes \Phi_i] \nabla 1. \quad (5.35)$$

Following this approach, a symmetric conservative form for the Euler equations is obtained as seen in the following formulation:

$$\frac{d(\omega_i \Phi_i)}{dt} + \omega_i \sum_{j=1}^J [\mathbf{F}(\Phi_i) - \mathbf{v}_{0i} \otimes \Phi_i + \mathbf{F}(\Phi_j) - \mathbf{v}_{0j} \otimes \Phi_j] \nabla W_{ij} \omega_j = 0. \quad (5.36)$$

Moreover, through the Leibniz–Reynolds transport theorem the volume variation

$$\frac{d\omega_i}{dt} = \omega_i \nabla \cdot \mathbf{v}_{0i} \quad (5.37)$$

is approximated as

$$\frac{d\omega_i}{dt} = \omega_i \sum_{j=1}^J (\mathbf{v}_{0j} - \mathbf{v}_{0i}) \nabla W_{ij} \omega_j \quad (5.38)$$

Grouping together equation (5.38) and equation (5.36), the full set of ALE-SPH equations for fluid motion are:

$$\left\{ \begin{array}{l} \frac{d\mathbf{x}_i}{dt} = \mathbf{v}_{0i} \\ \frac{d\omega_i}{dt} = \omega_i \sum_{j=1}^J (\mathbf{v}_{0j} - \mathbf{v}_{0i}) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i}{dt} = -\omega_i \sum_{j=1}^J (\rho_i (\mathbf{v}_i - \mathbf{v}_{0i}) + \rho_j (\mathbf{v}_j - \mathbf{v}_{0j})) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i \mathbf{v}_i}{dt} = -\omega_i \sum_{j=1}^J (\rho_i \mathbf{v}_i \otimes (\mathbf{v}_i - \mathbf{v}_{0i}) + \\ \quad + \rho_j \mathbf{v}_j \otimes (\mathbf{v}_j - \mathbf{v}_{0j}) + p_i + p_j) \nabla W_{ij} \omega_j + \omega_i \rho_i \mathbf{g} \end{array} \right. \quad (5.39)$$

Under a mathematical perspective, the ALE formulation (5.39) is a centred spatial discretization, and, as a hyperbolic system of equations, it is numerically unstable.

In order to add numerical stabilization diffusion terms can be introduced similarly to classical SPH, by introducing artificial viscosity, (Monaghan 1994; Molteni and Colagrossi 2009; Antuono et al. 2012):

$$\begin{aligned} \frac{d(\omega_i \Phi_i)}{dt} + \omega_i \sum_{j=1}^J (\mathbf{F}_i(\Phi_i) - \mathbf{v}_{0i} \otimes \Phi_i + \\ \mathbf{F}_j(\Phi_j) - \mathbf{v}_{0j} \otimes \Phi_j + \Pi^{ij}) \nabla W_{ij} \omega_j = 0 \end{aligned} \quad (5.40)$$

or shortened through equation (5.15),

$$\frac{d(\omega_i \Phi_i)}{dt} + \omega_i \sum_{j=1}^J (\mathbf{F}_i(\Phi_i, \mathbf{v}_{0i}) + \mathbf{F}_j(\Phi_j, \mathbf{v}_{0j}) + \Pi^{ij}) \nabla W_{ij} \omega_j = 0 \quad (5.41)$$

A different methodology to add numerical diffusion in the scheme, has been proposed by Vila (1999), in which the flux terms $\mathbf{F}_i$ and $\mathbf{F}_j$ are replaced by a decentred flux, $\mathbf{F}_{ij}$:

$$\mathbf{F}_{ij}(\Phi_{ij}, \mathbf{v}_0(\mathbf{x}_{ij})) \approx \frac{\mathbf{F}_i(\Phi_i, \mathbf{v}_{0i}) + \mathbf{F}_j(\Phi_j, \mathbf{v}_{0j})}{2}, \quad (5.42)$$

located at the interface between particle i and particle j , in $\mathbf{x}_{ij} = \frac{\mathbf{x}_i + \mathbf{x}_j}{2}$, which moves with the interface transport velocity defined by

$$\mathbf{v}_0(\mathbf{x}_{ij}) = \frac{\mathbf{v}_{0i} + \mathbf{v}_{0j}}{2}. \quad (5.43)$$

It is computed solving a Riemann problem defined by the left and the right state of Φ_i

and Φ_j , then it is substituted in equation (5.36),

$$\frac{d(\omega_i \Phi_i)}{dt} + \omega_i \sum_{j=1}^J 2\mathbf{F}_{ij}(\Phi_{ij}, \mathbf{v}_0(\mathbf{x}_{ij})) \nabla W_{ij} \omega_j = 0 \quad (5.44)$$

In the continuity and in the momentum equations, the solution of the Riemann problem $\Phi_{ij}^E = (\rho_{ij}^E, \rho_{ij}^E, \mathbf{v}_{ij}^E)$ at the i - j interface replaces the centred field quantities $\rho_i, \rho_j, p_i, p_j, \mathbf{v}_i, \mathbf{v}_j$ whereas $\mathbf{v}_0(\mathbf{x}_{ij})$ substitutes $\mathbf{v}_{0i}$ and $\mathbf{v}_{0j}$. Gathering together the motion of the particle (5.19) and the volume evolution (5.38), the ALE-SPH Euler equations are obtained as presented in Oger et al. (2016),

$$\left\{ \begin{array}{l} \frac{d\mathbf{x}_i}{dt} = \mathbf{v}_{0i} \\ \frac{d\omega_i}{dt} = \omega_i \sum_{j=1}^J (\mathbf{v}_{0j} - \mathbf{v}_{0i}) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i}{dt} = -\omega_i \sum_{j=1}^J 2\rho_{ij}^E (\mathbf{v}_{ij}^E - \mathbf{v}_0(\mathbf{x}_{ij})) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i \mathbf{v}_i}{dt} = -\omega_i \sum_{j=1}^J 2 [\rho_{ij}^E \mathbf{v}_{ij}^E \otimes (\mathbf{v}_{ij}^E - \mathbf{v}_0(\mathbf{x}_{ij})) + p_{ij}^E] \nabla W_{ij} \omega_j + \omega_i \rho_i \mathbf{g} \end{array} \right. \quad (5.45)$$

This scheme has the capability to recover both the Eulerian and the Lagrangian description, imposing respectively $\mathbf{v}_0 = 0$ or $\mathbf{v}_0 = \mathbf{v}$.

5.5 SPH-ALE for numerical simulations

In this section are specified complementary information regarding the methodologies that have been implemented in the SPH-ALE based software ASPHODEL, (Marongiu 2007; Leduc 2010; Neuhauser 2014; Renaut 2015; Pineda Rondon 2017), which has been developed in the latest years especially for simulating water flows in turbines, (Rentschler et al. 2018; Pineda et al. 2019).

In particular, in order to solve the fluid governing equations expressed in (5.45), the reconstruction of field quantities has to be computed at the interfaces and a time integration scheme is needed to update position, volume, density and velocity. Additionally, solid objects in the computational domain are treated following the techniques introduced in Marongiu (2007) and in Li (2013), in which a partial Riemann problem is solved at the interfaces between fluid and solid particles; these approaches are well suited to deal with complex geometries with thin components. Moreover, the open boundary conditions have to be carefully imposed; the Lagrangian nature of SPH makes it troublesome to manage particles that are created in inlet sections or removed in outlet sections, (Marongiu 2007; Neuhauser 2014; Pineda Rondon 2017).

5.5.1 Volume equation

Pineda Rondon (2017) introduced an additional assumption regarding the transport velocities that are used in equation (5.39), to express the equation for volume variation using a consistent approach. In the mass and in the momentum equations, the average transport velocity, defined by equation (5.43), is adopted, while in the volume equation the difference between $\mathbf{v}_{0i}$ and $\mathbf{v}_{0j}$ is taken into account. In order to be consistent, it is rewritten as,

$$\mathbf{v}_{0j} = 2\mathbf{v}_0(\mathbf{x}_{ij}) - \mathbf{v}_{0i}, \quad (5.46)$$

then, subtracting $\mathbf{v}_{0i}$ on both sides, it leads to

$$\mathbf{v}_{0j} - \mathbf{v}_{0i} = 2(\mathbf{v}_0(\mathbf{x}_{ij}) - \mathbf{v}_{0i}). \quad (5.47)$$

The right-hand side term is replaced in the volume equation, the final set of governing equations for the SPH-ALE scheme, which is solved in the code ASPHODEL, is presented:

$$\begin{cases} \frac{d\mathbf{x}_i}{dt} = \mathbf{v}_{0i} \\ \frac{d\omega_i}{dt} = \omega_i \sum_{j=1}^J 2(\mathbf{v}_0(\mathbf{x}_{ij}) - \mathbf{v}_{0i}) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i}{dt} = -\omega_i \sum_{j=1}^J 2\rho_{ij}^E (\mathbf{v}_{ij}^E - \mathbf{v}_0(\mathbf{x}_{ij})) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i \mathbf{v}_i}{dt} = -\omega_i \sum_{j=1}^J 2[\rho_{ij}^E \mathbf{v}_{ij}^E \otimes (\mathbf{v}_{ij}^E - \mathbf{v}_0(\mathbf{x}_{ij})) + p_{ij}^E] \nabla W_{ij} \omega_j + \omega_i \rho_i \mathbf{g} \end{cases} \quad (5.48)$$

5.5.2 Interface reconstruction: MUSCL scheme

In the continuity and in the momentum equations the fluxes between neighbours particles are computed at the interfaces, solving a Riemann problem. The method used to define the initial states Φ_L and Φ_R has not been explained yet. The Godunov's first-order conservative upwind scheme for FVM has been adapted for ALE formalism in Harten and Hyman (1983), but it usually generates too numerical dissipation, therefore, to overcome this issue, a second-order method, called Monotone Upstream-centred Scheme for Conservation Laws (MUSCL) is adopted to reconstruct the Left and Right state, Φ_L and Φ_R , of the Reimann problem, Van Leer (1979). In this method, the values of the field quantities are approximated with a linear interpolation between i and j . To adapt the MUSCL scheme to the SPH-ALE method, (Marongiu et al. 2010; Vila 1999), the particle interface position, which is not known in SPH is estimated at

the midpoint, and the reconstruction scheme assumes the following formulation:

$$\begin{cases} \Phi_L = \Phi_i + \alpha(\Phi_i, \Phi_j, \nabla\Phi_i) \nabla\Phi_i \cdot \left(\frac{\mathbf{x}_j - \mathbf{x}_i}{2} \right) \\ \Phi_R = \Phi_j - \alpha(\Phi_i, \Phi_j, \nabla\Phi_j) \nabla\Phi_j \cdot \left(\frac{\mathbf{x}_j - \mathbf{x}_i}{2} \right) \end{cases} \quad (5.49)$$

where $\nabla\Phi$ are the gradient computed using the SPH formulation and $\alpha \in [0, 1]$ is the limiter *minmod* function, (LeVeque 1992).

5.5.3 Time integration

In this project, the integration scheme used to update in time the equations (5.48), which are generally expressed as,

$$\frac{d\Psi}{dt} = \mathcal{H}(\Psi, \mathbf{x}, t), \quad (5.50)$$

is an explicit Runge-Kutta (RK) third-order, (Hirsch 2007). In explicit methods, the state variables Φ^n , at $t = t^n$, are incremented to Φ^{n+1} , at $t = t^{n+1}$, through the known values of the function $\mathcal{H}^n = \mathcal{H}(\Psi^n, \mathbf{x}^n, t^n)$. The RK third-order scheme includes three intermediate sub-steps defined by

$$\begin{cases} \Psi^{(1)} = \Psi^n \\ \Psi^{(2)} = \Psi^n + \frac{1}{2}\Delta t \mathcal{H}^{(1)} \\ \Psi^{(3)} = \Psi^n - \Delta t \mathcal{H}^{(1)} + 2\Delta t \mathcal{H}^{(2)} \\ \Psi^{(n+1)} = \Psi^n + \frac{1}{6}\Delta t \mathcal{H}^{(1)} + \frac{2}{3}\Delta t \mathcal{H}^{(2)} + \frac{1}{6}\Delta t \mathcal{H}^{(3)} \end{cases} \quad (5.51)$$

where the $\mathcal{H}^{(k)} = \mathcal{H}(\Psi^{(k)})$ with $k = 1, 2, 3$.

The choice of time step is imposed with the same criteria as in the FVM, the Courant-Friedrichs-Lewy condition,

$$\Delta t = CFL \min_{i \in \Omega} \left(\frac{h_i}{c_{0i} + \|\mathbf{v}_{0i}\|} \right) \quad (5.52)$$

where c_{0i} is the local speed of sound and, in this work, $CFL = 0.2$.

Chapter 6

Correction techniques for particle distribution

As previously illustrated in Chapter 4, in meshless methods the quality of the particle distribution has a significant influence on the space interpolation accuracy. In this chapter, a literature review of the most relevant particle distribution correction techniques, applied in SPH schemes, is reported. These corrections aim to avoid the formation of Lagrangian structures of particles along flow trajectories, maintaining homogeneous particle distribution. The disordering of particles position, which are the calculation points for field variables in the SPH methods, increases the numerical error and reduces the accuracy in the simulations. Throughout the years, several studies have been conducted to analyze the performance of different algorithms. As a global view, Particle Shifting Techniques (PSTs) are formulated with two different approaches that can be adapted based on the specif solver for the governing equations in which they are implemented. In practice, the correction term can be introduced as a displacement directly applied to the particle position,

$$\bar{\mathbf{x}}_i = \mathbf{x}_i + \delta\mathbf{x}_i, \quad (6.1)$$

where $\mathbf{x}_i$ is the original position, $\bar{\mathbf{x}}_i$ is the adjusted or shifted position and $\delta\mathbf{x}_i$ is the shifting vector of the i -th particle, or, in other approaches, as a modification to the particle velocity,

$$\bar{\mathbf{v}}_i = \mathbf{v}_i + \delta\mathbf{v}_i, \quad (6.2)$$

where $\mathbf{v}_i$ is the field velocity, $\bar{\mathbf{v}}_i$ is the modified velocity and $\delta\mathbf{v}_i$ is the correction term.

In these techniques, the Lagrangian nature of SPH, in which particles move with their velocity, want to be maintained and not drastically altered, therefore, the shifting velocity, $\delta\mathbf{v}_i$, (or the shifting displacement $\delta\mathbf{x}_i$), is usually limited to a factor of the maximum physical velocity (or displacement).

6.1 SPH schemes with particles reordering techniques

In the early stages of the development for applying SPH to the study of fluids, Monaghan (1989) introduced the so-called XSPH method, which is a correction to the Lagrangian particle velocity that prevents particles penetration. This formulation has been

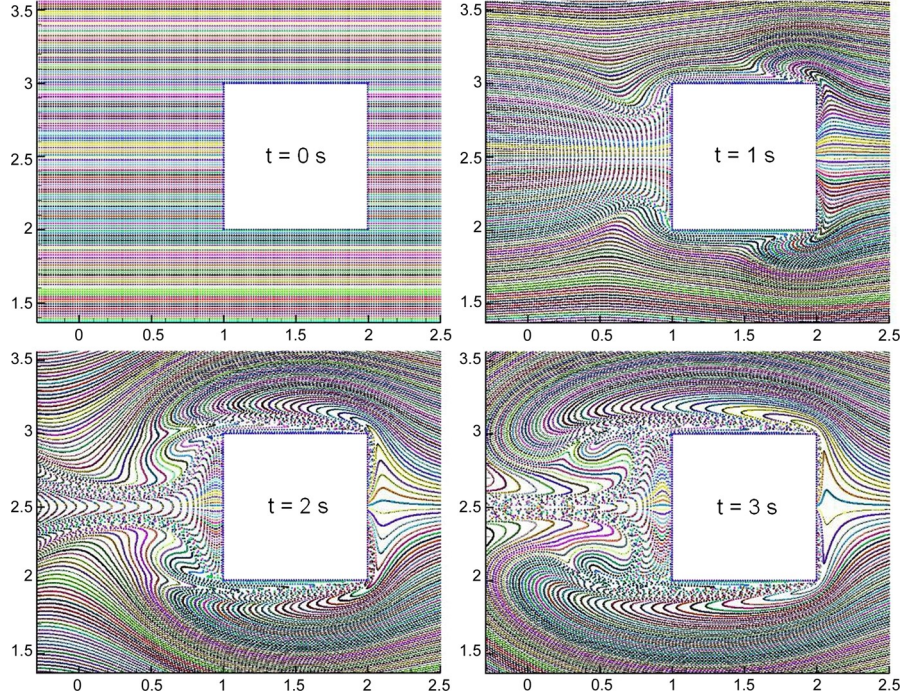


Figure 6.1 Oger et al. (2016), purely Lagrangian streamlines past a squared box.

defined as follows, (Monaghan 1994),

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i + \epsilon \sum \left(\frac{\mathbf{v}_i - \mathbf{v}_j}{\rho_{ij}} \right) m_j W_{ij}, \quad (6.3)$$

where m_j is the mass and $\rho_{ij} = (\rho_i + \rho_j) / 2$ is the average density, while ϵ is the tuning parameter that can range between 0 and 1, usually set equal to 0.5. This formulation was proposed in the context of WCSPH scheme, the additional term in the kinematic equation smooths the particles' velocity with the neighbours average velocities. The motion of particles does not exactly follow any longer the Lagrangian trajectories, but the conservation in linear and angular momentum is preserved.

A different approach, in which the particles' position is not modified, but completely reinitialized, has been proposed by Chaniotis et al. (2002), this methodology is substantially equivalent to a remeshing process, in which particles are periodically relocated on a regular grid, then, the physical quantities are re-interpolated.

6.2 SPH schemes with repulsive forces

To remove tensile instabilities, Monaghan (2000) introduced some repulsive terms in the form of Lennard-Jones forces, modifying the formulation using the kernel function W through a function that is inversely proportional to the distance between particles,

$$f_{ij} = \frac{W_{ij}}{W_{ij}}, \quad (6.4)$$

where $\overline{W_{ij}}$ is the kernel value computed for the average spacing between the neighbours of particle i . Using this approach in the momentum equation, written in the canonical form, (equation (5.18)), the repulsive force is added,

$$\frac{d\mathbf{v}_i}{dt} = - \sum_{j=1}^J \left(\frac{p_j}{\rho_j^2} + \frac{p_i}{\rho_i^2} + \Pi_{ij} + R f_{ij}^n \right) \nabla W_{ij} m_j + \mathbf{g}, \quad (6.5)$$

where n is positive and usually set equal to 4, R depends on the pressure and on the density, it is basically an artificial pressure field, defined as,

$$R = \epsilon \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right), \quad (6.6)$$

where ϵ assumes the values previously defined for equation (6.3). Repulsive forces have been introduced to stabilize the particle distribution, removing tensile instability that generates clustering effects. This methodology has been reposed by several authors in the following years, for example, Tsuruta et al. (2013) introduced the dynamic stabilization method in the context of incompressible smoothed particle hydrodynamics,

$$\left(\frac{1}{\rho} \nabla p \right) = \sum_{j=1}^J \frac{m_j}{\rho_i \rho_j} (p_j - p_i) \nabla W_{ij} + \mathcal{A}_{DS}, \quad (6.7)$$

where $\mathcal{A}_{DS}$ is a stabilization term for the pressure projected equation that keeps particles uniformly spaced.

6.3 Incompressible SPH schemes with particle shifting techniques

In the context of ISPH, the initial milestone for particle shifting techniques, properly intended as methodologies to move particles in order to obtain and maintain spatial uniformity, has been presented by Xu et al. (2009) through a formulation which was inspired by the work of Nestor and Quinlan (2007), developed within Finite Volume Particle Methods (FVPM).

The particle shifting algorithm has the formulation reported in equation (6.1) in which the correction, explicitly designed to regularize the particle distribution, is applied to the particle position. Xu et al. (2009) implemented the shifting correction term in a projection-based SPH solver as,

$$\delta \mathbf{x}_i = \mathcal{C} \alpha \mathbf{R}_i, \quad (6.8)$$

where $\mathcal{C}$ is a tuner coefficient and α is a factor determined as a multiplication between the maximum expected field velocity U_{max} and the time step Δt . The shifting adjust-

ment $\mathbf{R}_i$ is defined as a weighted distance,

$$\mathbf{R}_i = \sum_{j=1}^J \frac{\tilde{\mathbf{x}}_i^2}{\mathbf{x}_{ij}^2} \mathbf{n}_{ij}, \quad (6.9)$$

where $\tilde{\mathbf{x}}_i$ the average particle spacing between the neighbours of particle i , $\mathbf{x}_{ij}$ and $\mathbf{n}_{ij}$ are respectively the distance and the unit distance vector between particle i and particle j . The shifting procedure requires to correct the generic hydrodynamic variable ϕ using a Taylor series expansion,

$$\phi_{i'} = \phi_i \delta \mathbf{x}_{ii'} \cdot (\nabla \phi)_i + \mathcal{O}(\delta \mathbf{x}_{ii'}^2), \quad (6.10)$$

where $\delta \mathbf{x}_{ii'}$ is the distance vector between the particle's old position i , defined by $\mathbf{x}_i$ and its new position i' , defined by $\bar{\mathbf{x}}_i$. This correction allows particles to redistribute themselves in a more regular spacing, it is based on a weighting approach and it moves particles outward areas of aggregation. This method has been tested for a range of internal flows, showing an increase in accuracy. The results presented by Xu et al. (2009) encouraged further investigations on particle shifting algorithms in order to enlarge the cases in which these methods can be applied.

Still in the framework of Incompressible SPH, Lind et al. (2012) proposed an implementation following the key idea of redistributing the particles from packed areas to lacking areas. The author introduced a particle shifting method based on Fick's law of diffusion, in which particles move from the regions of high concentration to regions of low concentration and the particles concentration C is computed as the summation of the kernel function,

$$C_i = \sum_{j=1}^J \omega_j W_{ij}. \quad (6.11)$$

The gradient of particles concentration, ∇C_i , can be physically seen as a measure of the non-uniformity of particle distribution at particle i , (Colagrossi et al. 2012). ∇C is the gradient operator of SPH and it shows the accuracy of the SPH interpolation, as viewed in Section 4.1. Large variations of the particle concentration, which can be due to not homogeneous particle distribution, presence of clusters or voids, are numerically translated into high values of ∇C , which is calculated as,

$$\nabla C_i = \sum_{j=1}^J \omega_j \nabla W_{ij}. \quad (6.12)$$

In a physical simulation, the particles shifting vector, defined in equation (6.1), has been

rewritten by Lind et al. (2012),

$$\delta \mathbf{x}_i = -D \nabla C_i. \quad (6.13)$$

where D is a diffusion coefficient. A correction function, f_{ij} , that act like the artificial pressure, similarly to equation (6.4), is defined,

$$f_{ij} = b \left(\frac{W_{ij}}{W(\Delta)} \right)^n, \quad (6.14)$$

where Δ is the initial particle size. The function f_{ij} is used to minimize the instability due to zero kernel gradient at the origin and to remove the interactions when the particles tend to overlap, (b and n are suggested in Monaghan (2000)). $\delta \mathbf{x}_i$ is adjusted introducing the following corrected concentration gradient, reformulating from equation (6.12) using equation (6.14),

$$\widehat{\nabla C}_i = \sum_{j=1}^J \omega_j (1 + f_{ij}) \nabla W_{ij}. \quad (6.15)$$

The particle shifting technique improves the robustness of the ISPH scheme, particular attention has been posed on the shifting maximum values; in areas of strong variation of particles concentration equation 6.15 can take to relatively high displacements. It has been introduced a limitation in terms of velocity in order to maintain the shifting correction terms coherent with the physical velocities imposed by the Lagrangian motion:

$$\delta \mathbf{x}_i = \begin{cases} -0.5h \frac{h}{\Delta t} \widehat{\nabla C}_i, & \text{if } \|0.5h \frac{h}{\Delta t} \widehat{\nabla C}_i\| < 0.2 \frac{h}{\Delta t} \\ -0.2 \frac{h}{\Delta t} \frac{\widehat{\nabla C}_i}{\|\widehat{\nabla C}_i\|}, & \text{otherwise} \end{cases} \quad (6.16)$$

Moreover, Lind et al. (2012) introduced an implementation to the shifting algorithm in order to treat simulations with free surface flow; equation (6.13) is rewritten, using a correction and a free surface detector algorithm, which stabilizes problems that occur during the simulation due to anisotropic particle distribution at the interface,

$$\delta \mathbf{x}_i = -D \left(\frac{\partial C_i}{\partial s} \mathbf{s}_i + \alpha \left(\frac{\partial C_i}{\partial n} - \beta \right) \mathbf{n}_i \right), \quad (6.17)$$

where $\mathbf{s}_i$ and $\mathbf{n}_i$ are respectively the tangent and the normal vectors to the free surface of particle i ; β and α are parameters that prevent nonphysical shifting toward the free surface. Adopting this algorithm the accuracy has been improved even in presence of free surface.

To avoid instabilities Skillen et al. (2013) limited the diffusion coefficient thought

out a Von Neumann stability analysis, linking the parameter D of equation (6.13), with the smoothing length h , the fluid velocity $\mathbf{v}$ and the time step Δt . The proposed formulation is,

$$\delta \mathbf{x}_i = -Ah \|\mathbf{v}_i\| \Delta t \nabla C_i, \quad (6.18)$$

where A is a dimensionless constant problem invariant.

To determine the free surface particles Lee et al. (2008) proposed a threshold value for the particles position divergence $\nabla \mathbf{r}$, defined as,

$$\nabla \mathbf{r}_i = \sum_{j=1}^J \frac{m_j}{\rho_j} \mathbf{r}_{ij} \nabla W_{ij}, \quad (6.19)$$

where $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ is the distance vector. The value obtained from equation (6.19) has been used to compute the free surface correction coefficient, A_{FSC} ,

$$A_{FSC} = \frac{\nabla \mathbf{r}_i - A_{FST}}{A_{FSM} - A_{FST}}, \quad (6.20)$$

where A_{FSM} is 2 in 2D and 3 in 3D, and it represents the maximum value of $\nabla \mathbf{r}$ for particles in the inner domain, (Mokos et al. 2017). A_{FST} is the threshold value for particles to the free surface, (1.5 in 2D and 2.5 in 3D) then, equation (6.18) is multiplied by the coefficient A_{FSC} to obtain the value of the particle shifting vector,

$$\delta \mathbf{x}_i = \begin{cases} -A_{FSC} Ah \|\mathbf{v}_i\| dt \nabla C_i, & \text{if } (\nabla \mathbf{r}_i - A_{FST}) < 0 \\ -Ah \|\mathbf{v}_i\| dt \nabla C_i, & \text{if } (\nabla \mathbf{r}_i - A_{FST}) = 0 \end{cases} \quad (6.21)$$

The Fickian formulation, (Lind et al. 2012), in equation (6.17), is considered as the real breakthrough in the PSTs, it has been seen a starting point for improvements in more specific applications. Many authors proposed enhancement to better suit real problems, among them Khayyer et al. (2017) proposed an optimization to minimize some nonphysical gaps in fluids in proximity to interfaces. The correction has been carried on introducing a modified unit normal vector $\tilde{\mathbf{n}}_i$,

$$\tilde{\mathbf{n}}_i = -\frac{\mathbf{B}_i \nabla C_i}{|\mathbf{B}_i \nabla C_i|}, \quad (6.22)$$

where $\mathbf{B}_i$ is defined by,

$$\mathbf{B}_i = \left(\sum_{j=1}^J \mathbf{r}_{ij} \otimes \nabla W_{ij} \omega_j \right)^{-1}. \quad (6.23)$$

Introducing this assumption, equation (6.17) is rewritten using the modified normal, $\tilde{\mathbf{n}}$,

and tangential, $\tilde{\mathbf{s}}$ vectors as,

$$\delta \mathbf{x}_i = -D \left(\frac{\partial C_i}{\partial \tilde{\mathbf{s}}} \tilde{\mathbf{s}}_i + \frac{\partial C_i}{\partial \tilde{\mathbf{n}}} \tilde{\mathbf{n}}_i \right). \quad (6.24)$$

It can be seen that the tuning parameters α and β of equation (6.17) are no longer needed and the particle shifting in the normal direction are automatically cancelled out. This methodology has been extended to multi-phase flows by Khayyer et al. (2019).

6.4 Weakly Compressible SPH schemes with Particle Shifting Techniques

The particle shifting techniques, presented for ISPH, have been reformulated in the context of Weakly-Compressible SPH by Shadloo et al. (2012) showing improvements in the accuracy compared to pure Lagrangian SPH schemes.

Immediately after, Vacondio et al. (2013), applied a modification on the particle's velocity instead that on the particles position,

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i + \delta \mathbf{x}_i, \quad (6.25)$$

where, in this case, $\delta \mathbf{x}_i$ has to be seen as a velocity shifting vector. In order to manage simulations with different particle sizes, due to splitting or coalescing processes, an adjusted scheme has been introduced,

$$\delta \mathbf{x}_i = \frac{\beta}{m_T} \sum_{j=1}^J m_j \frac{\mathbf{x}_{ij}}{\mathbf{r}_{ij}^3} r_0^2 \mathbf{v}_{max}, \quad (6.26)$$

where $r_0 = \sum_j r_{ij}/N$ is average inter-particles distance, $m_T = \sum_j m_j$, is the total mass, β is a non-dimensional parameter, and $\mathbf{v}_{max}$ is the maximum fluid velocity.

The natural development of the δ -SPH scheme was proposed by Sun et al. (2017), named δ^+ -SPH, aiming to improve the model with a PST that specifically treats and re-arranges free surface particles. Using a Fickian-based approach, the position is modified introducing a dependency on the Mach number, Ma , through the following equation, suitable for multi-resolution simulations,

$$\delta \mathbf{x}_i = -CFL Ma (2h_{ij})^2 \widehat{\nabla C_i} \phi_{ij} \frac{m_j}{(\rho_i + \rho_j)}, \quad (6.27)$$

where h_{ij} is average smoothing length, R and n are constant respectively set to 0.2 and 4, (to prevent tensile instability as proposed in XSPH, equation (6.3)), and ϕ is a function used for solve multi-resolution problems. The equation (6.27) proposed some main differences in comparison with equation (6.13), specifically, the WCSPH approach limits the physical time step and, consequently, the particle shifting displacement is bounded

as a fraction of the particle spacing, which is related to the Mach number and the CFL number. For this reason, the field quantities do not require to be re-interpolated after the shifting. Moreover, in order to conserve total linear and angular momentum for particles in interior regions, the volume is substituted with the ratio between mass and average density. The δ^+ -SPH scheme has been implemented with an algorithm that turns off the shifting in the direction normal to the free surface, based on the distance from the interface. A parameter λ is designed to obtain information about the location of the particles and their normal vectors. It has to be calculated for each particle as the minimum eigenvalue of the tensor,

$$L(\mathbf{x}_i) = \left[\sum_{j=1}^J (\mathbf{x}_j - \mathbf{x}_i) \otimes \nabla W_{ij} \omega_i \right]^{-1}. \quad (6.28)$$

Particles with λ_i smaller than 0.2 belong to the free surface, particles with λ_i greater than 0.75 belong to the inner domain, then the particle correction has to be moderated as follow:

$$\delta \mathbf{x}_i = \begin{cases} 0 & \text{if } \lambda_i < 0.4 \quad \text{and} \quad i \in \text{free-surface region} \\ (\mathbb{I} - \mathbf{n}_i \otimes \mathbf{n}_i) \delta \mathbf{x}_i & \text{if } \lambda_i > 0.4 \quad \text{and} \quad i \in \text{free-surface region} \\ \delta \mathbf{x}_i & i \notin \text{free-surface region} \end{cases} \quad (6.29)$$

where $\mathbb{I}$ is the identity tensor and $\mathbf{n}_i$ is referred to particle i .

Still in the framework of WCSPH, a slightly different algorithm has been recently proposed by Sun et al. (2019), introducing a particle shifting technique in a quasi-Lagrangian scheme with a consistent approach. The correction, defined similarly to equation (6.27),

$$\delta \mathbf{v}_i = \begin{cases} -Mac_0(2h_{ij}) \widehat{\nabla C}_i & \text{if } \|Mac_0(2h_{ij}) \widehat{\nabla C}_i\| < 0.5U_{max} \\ -0.5U_{max} \frac{\widehat{\nabla C}_i}{\|\widehat{\nabla C}_i\|}, & \text{otherwise} \end{cases} \quad (6.30)$$

is applied directly to the particle velocity; then $\bar{\mathbf{v}}_i$ is substituted in the continuity and in the momentum equations maintaining values close to the physical velocity, without altering significantly the Lagrangian nature of the method.

In these schemes the particles' position is usually adjusted at the end of each time step, however, the explicit nature of the WCSPH models does not allow to ensure whether the updated particle configuration satisfies any quantitative requirements in terms of isotropy of the particle distribution.

For this reason, Vacondio and Rogers (2017) proposed to integrate the shifting tech-

nique, equation (6.18), into an iterative procedure. The aim is to fulfill a quantitative condition on the level of particle distribution disorder, evaluated by means of the maximum ∇C value. The iterative algorithm is activated whether ∇C rises above a predefined threshold and, at the end of the iterative shifting procedure the reinterpolation of the physical quantities is required. This method produces results more accurate in comparison to non iterative shifting methods but the number of iterations needed to reach the predefined threshold increases with the resolution, augmenting significantly the computational cost.

6.5 ALE-SPH schemes with Particle Shifting Techniques

6.5.1 Transport velocity in a quasi-Lagrangian scheme

The Fick's law of diffusion has been applied even in the context of ALE-SPH initially by Oger et al. (2016). The nature of the scheme, presented in Section 5.4, avoids the reinterpolation of the physical quantities. The shifting velocities is directly taken into account, in a consistent way, by the transport velocity. Oger et al. (2016) proposed the following law for the arbitrary velocity,

$$\bar{\mathbf{v}}_{0i} = \mathbf{v}_i + \delta \mathbf{v}_i, \quad (6.31)$$

based, as previously mentioned, on the field velocity adjusted with a velocity shifting term $\delta \mathbf{v}_i$ which value is far less than the fluid velocity $\mathbf{v}_i$,

$$\delta \mathbf{v}_i = \begin{cases} -U^{char} 2\Delta_i \nabla C_i & \text{if } U^{char} 2\Delta_i \|\nabla C_i\| < 0.25 \|\mathbf{v}_i\| \\ -0.25 \|\mathbf{v}_i\| \frac{\nabla C_i}{\|\nabla C_i\|} & \text{otherwise} \end{cases} \quad (6.32)$$

where $U^{char} = Mac_0$ is the characteristic velocity. The formulation limits the shifting to a percentage of the fluid velocity, maintaining the quasi-Lagrangian approach. For particles close to the interface, detected using specific algorithms, the shifting is disabled and the motion is purely Lagrangian.

6.5.2 Transport velocity based on Riemann solver

Another approach to shift the particles, reducing the spatial disorder with a different transport velocity formulation, has been proposed in the PhD project of Neuhauser

(2014), defined as acceleration correction method,

$$\begin{aligned} \left(\frac{d\mathbf{v}_{0i}}{dt} \right)_{corr} &= \sum_{j=1}^J \left[\frac{p_{ref}}{\rho_0} + \frac{c_0}{2} (\mathbf{v}_{0i} - \mathbf{v}_{0j}) \mathbf{n}_{ij} \right] \nabla W_{ij} \omega_j \\ &+ \sum_{j=1}^{J^\partial} \left[\frac{p_{ref}}{\rho_0} + \frac{c_0}{2} (\mathbf{v}_{0i} - \mathbf{v}_{0j}) \mathbf{n}_{ij} \right] W_{ij} \mathbf{n}_j \omega_j^\partial, \end{aligned} \quad (6.33)$$

where $p_{ref} = \rho_0 c_0^2 / \gamma$ is the uniform reference pressure. The transport velocity correction is computed in equation (6.33) with an upwind treatment needed to stabilize the hyperbolic system of particle position, particle velocity and background pressure. An additional term is added to the formulation introducing diffusion in order to smooth the arbitrary velocity,

$$\left(\frac{d\mathbf{v}_{0i}}{dt} \right)_{smooth} = \left(\frac{d\mathbf{v}_{0i}}{dt} \right)_{corr} - \alpha \sum_{j=1}^J \frac{c_0}{h_i} (\mathbf{v}_{0i} - \mathbf{v}_{0j}) W_{ij} \omega_j, \quad (6.34)$$

where α is a numerical parameter usually chosen equal to 0.1. The smoothed transport acceleration is scaled with the physical time step, which implicitly limits its magnitude, and it is introduced in the transport velocity formulation as,

$$\bar{\mathbf{v}}_0 = \mathbf{v}_i + \Delta t \left(\frac{d\mathbf{v}_{0i}}{dt} \right)_{smooth}. \quad (6.35)$$

This correction is implemented in the code ASPHODEL, previously mentioned in Section 5.5, its behaviour will be deeply investigated in Chapter 7.

6.5.3 δ -ALE-SPH

Recently, a new particle shifting methodology has been proposed by Antuono et al. (2021) in the contest of ALE-SPH with some different treatments,

$$\delta \mathbf{v}_i = \min \left(\left\| -Mac_0(2h_i) \widehat{\nabla C_i} \right\|, \frac{U_{max}}{2} \right). \quad (6.36)$$

In the work of Antuono et al. (2021) the Navier-Stokes equations have been rewritten in terms of density and velocity, introducing numerical diffusive terms.

An extensive recap of the state-of-the-art of PSTs is reported in Michel et al. (2021), in which is firstly derived a general expression that can summarize equations (6.16),

(6.30) and (6.32) in the following form,

$$\delta \mathbf{v}_i = \begin{cases} U^{char} d^{char} \widehat{\nabla C_i}, & \text{if } \|U^{char} d^{char} \widehat{\nabla C_i}\| < U^{lim} \\ U^{lim} \frac{\widehat{\nabla C_i}}{\|\widehat{\nabla C_i}\|} & \text{otherwise} \end{cases} \quad (6.37)$$

This formulation depends on U^{char} , the characteristic velocity, d^{char} , characteristic length, and it is bounded by the limit U^{lim} . Michel (2020) and Vergnaud (2020) proposed to define the characteristic velocity as follow:

$$U^{char} = \max_{j \in J} \left(\left| (\mathbf{v}_j - \mathbf{v}_i) \frac{\mathbf{x}_j - \mathbf{x}_i}{\|\mathbf{x}_j - \mathbf{x}_i\|} \right| \right). \quad (6.38)$$

Using relative velocities between neighbouring particles the shifting magnitude is influenced only by the local kinematics of the flow. Then, the formulation for the PST is expanded as,

$$\delta \mathbf{v}_i = 0.5 \begin{cases} -U^{char} \left(\frac{R}{\Delta} \right)^3 R \widehat{\nabla C_i}, & \text{if } \left\| \left(\frac{R}{\Delta} \right)^3 R \widehat{\nabla C_i} \right\| < \frac{1}{2} \frac{R}{\Delta} \\ -U^{char} \frac{1}{2} \frac{R}{\Delta} \frac{\widehat{\nabla C_i}}{\|\widehat{\nabla C_i}\|} & \text{otherwise} \end{cases} \quad (6.39)$$

Michel (2020) and Vergnaud (2020) illustrated solid points regarding the approach to define a novel PST but still some limitations are not debated.

Chapter 7

Explicit Particle Shifting techniques

This chapter presents an investigation on the particle shifting technique developed in the PhD project of Neuhauser (2014), which has been implemented in the code AS-PHODEL, for the SPH-ALE numerical scheme described in Section 5.5. In ALE-SPH schemes the particle position is controlled by the transport velocity $\boldsymbol{v}_0$. As demonstrated by Oger et al. (2016), adopting a quasi-Lagrangian approach, in which the transport velocity is defined applying a correction term to the fluid velocity, equation (6.31), prevents the formation of anisotropic particle distribution. Different analyses have been conducted aiming to optimize the explicit algorithm, equation (6.33), increasing the effectiveness without introducing computational overhead.

The Taylor-Green vortex test case, see Section 3.1, has been used to study the shifting techniques. Initially, the shifting formulations have been evaluated under a purely kinematic point of view, only the governing equation for the particle position is solved (first equation in the system (5.39)). In these analyses, the velocity and the pressure fields are reinitialized through the exact solution, equation (3.1) for $t = 0$ s, at each physical time step, to assess the role of the PSTs on the particle distribution quality. In this kinematic analysis, the variation of physical quantities is not considered, the particle shifting quality is assessed through the evaluation of ∇C and its L_∞ norm. Later, the explicit shifting algorithms have been studied solving the full set of the Navier-Stokes equations (system (5.39)), to analyze how the shifting methodologies affect the accuracy of the results.

The results reported in the following sections have a fixed resolution, $\Delta/L = 0.00625$, however, simulations with greater and smaller particle sizes have been tested producing similar outcomes; the numerical tests start from an initial Cartesian particle distribution. In the TGV the maximum fluid velocity, (U in equation (3.1)), has been set equal to 1 m/s; in Weakly-Compressible SPH, the Mach number is usually assumed in the order of 0.1, ($\text{Ma} = \|\boldsymbol{v}\|/c_0 < 0.1$), and the minimum value of c_0 is 10 m/s.

7.1 Riemann-based particle shifting technique

Preliminary analysis has been conducted to better understand the formulation behaviour presented in Section 6.5.2 and the role of the parameters in the correction term, defined in equation (6.33). In this formulation, the reference speed of sound c_0 is the leading term, which affects both the correction term and the time step size.

Under a theoretical point of view, the formulation (6.33) uses the relative velocities, it is invariant for uniform translations and because it is locally defined and scaled by Δt , which depends on the particle size, it results resolution-independent. Analytically, it is expected that higher values of c_0 increase the correction term, producing an increment in the deviation of particles from the original Lagrangian streamlines, therefore, the role of c_0 is deeply analyzed in the following section.

7.1.1 Results solving the kinematic equation

As previously mentioned, initially the focus has been posed on the kinematic only, to assess the capability of equation (6.33) in maintaining low level in particles disorder, therefore, at each physical time step the velocity and the pressure fields have been reinitialized at $t = 0$ s, following equation (3.1), to force a continuous perturbation in the particle positions. The value of c_0 has been gradually incremented, and results for $c_0 = 10, 20, 100$ m/s are shown in Figure 7.1. It can be noted in Figure 7.1(a) that the maximum value of ∇C is reduced by one order of magnitude increasing the reference speed of sound from 10 m/s to 100 m/s, this behaviour can be observed even in the ∇C field, Figure 7.1(b), (c) and (d).

7.1.2 Results solving the Navier-Stokes equations

Later, the experiments have been repeated using the above-mentioned values of c_0 , while solving the full set of Navier-Stokes equations (5.48). In these tests the kinematic viscosity ν has been set equal to $0.01 \text{ m}^2/\text{s}$, meaning a Reynolds number equal to 100. Similar results have been founded, in Figure 7.2 the decreasing trends in the L_∞ norm of ∇C is due to the decay of the maximum velocity.

Computational times for the simulations presented in Figure 7.2 are reported in Figure 7.3 normalized with the maximum computational time T_{Max} for c_0 equal to 10 m/s. As expected, increasing the reference speed of sound augments the magnitude of the shifting velocity, improving the strength of the methodology, but, as relevant drawback, it reduces the size of the time step raising significantly the *CPU* time due to the CFL condition.

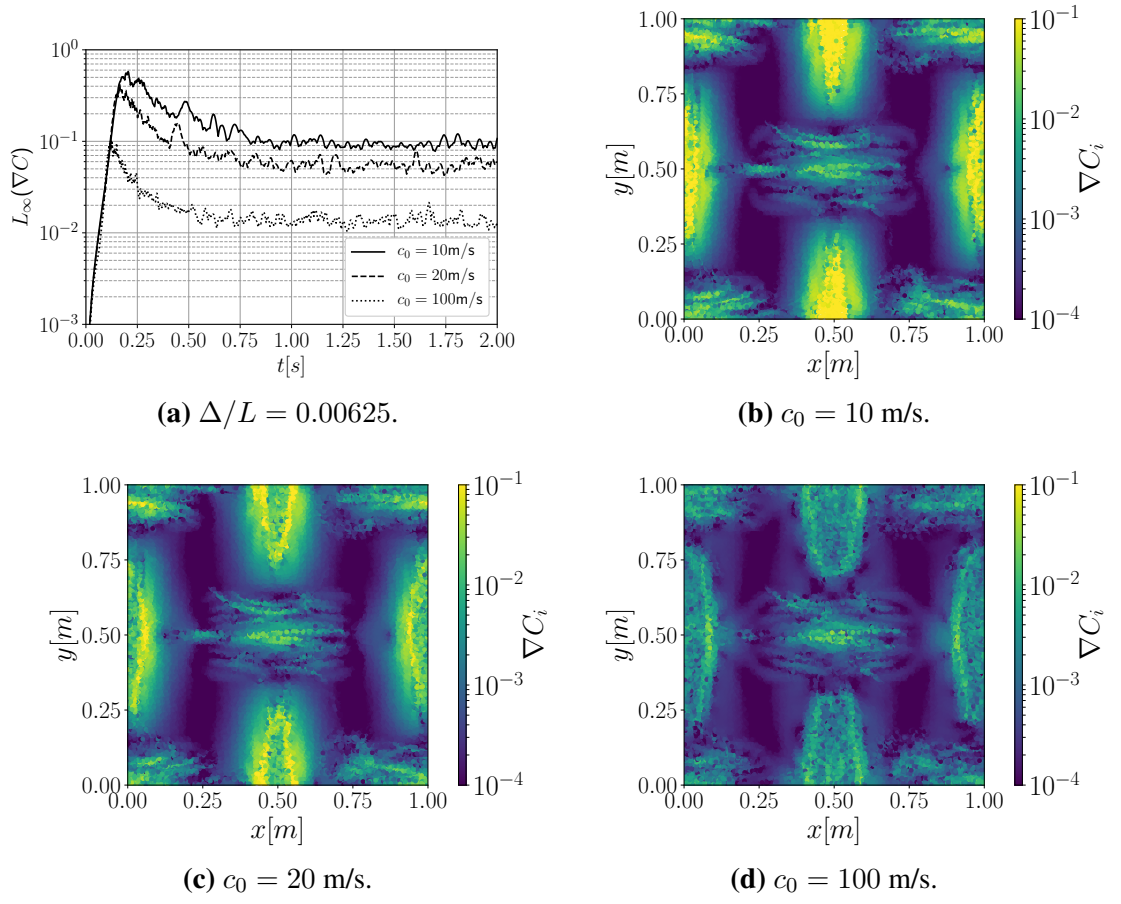


Figure 7.1 TGV test case. Neuhauser (2014) formulation, imposed analytical solution. Results for $\Delta/L = 0.00625$ at $t = 0.2$ s.

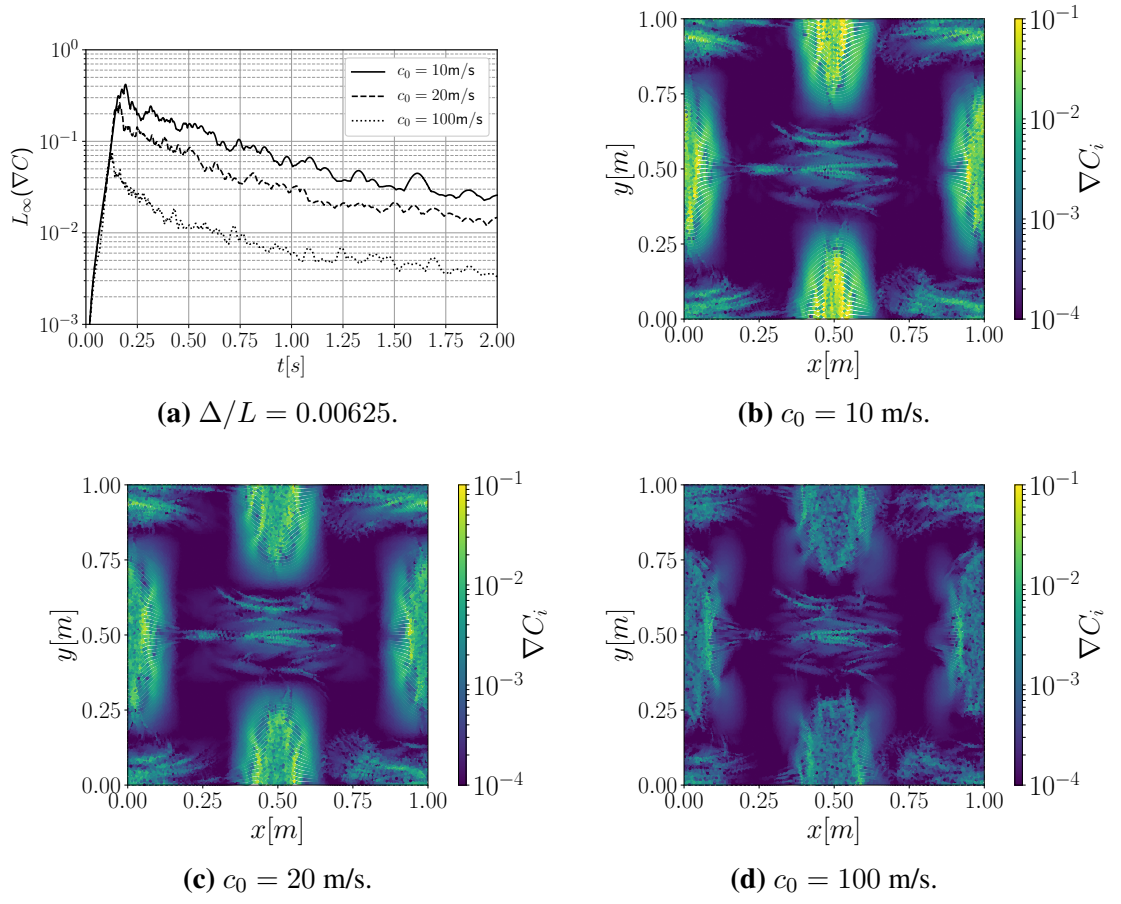


Figure 7.2 TGV test case. Neuhauser (2014) formulation, solved simulation. Results for $\Delta/L = 0.00625$ at $t = 0.2$ s.

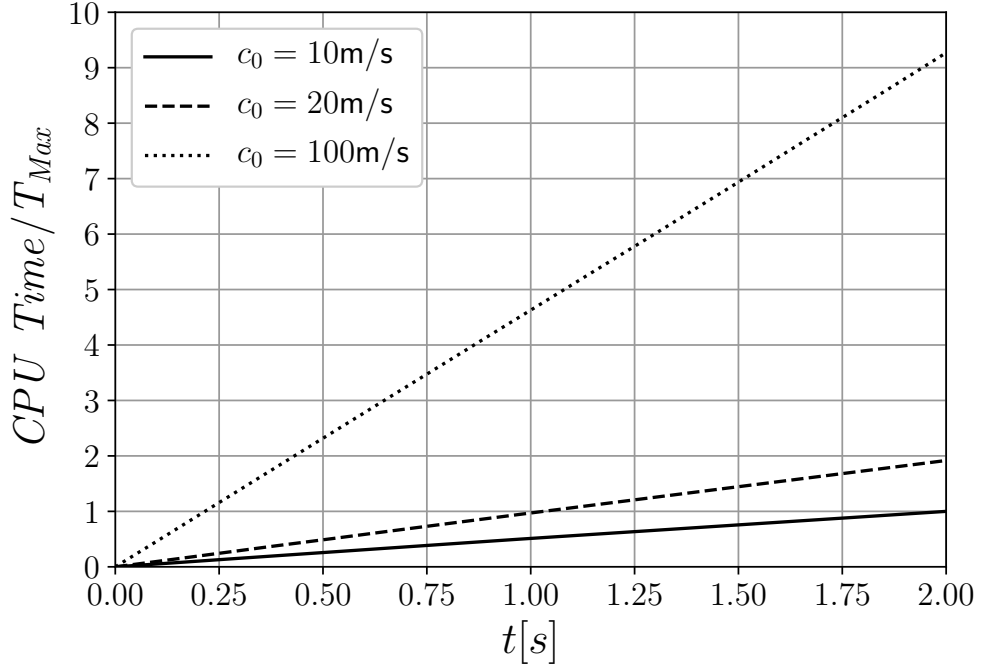


Figure 7.3 TGV test case. Neuhauser (2014) formulation. Time comparison for $c_0 = 10, 20$, and 100 m/s.

7.2 Improvement of a Riemann-based particle shifting technique

Following, using the information obtained in the preliminary studies, presented in Section 7.1, two different formulations are proposed to enhance the equation (6.33).

7.2.1 Particle Shifting coefficient

In the first optimization proposed, an empirical particle shifting coefficient, P_S , has been introduced to directly increment the reference pressure, keeping the particle shifting term elevate even in case of low kinematics; the formulation for the correction, defined in equation (6.33), is upgraded as follow,

$$\begin{aligned} \left(\frac{d\mathbf{v}_{0i}}{dt} \right)_{corr} = & \sum_{j=1}^J \left[P_S \frac{p_{ref}}{\rho_0} + \frac{c_0}{2} (\mathbf{v}_{0i} - \mathbf{v}_{0j}) \mathbf{n}_{ij} \right] \nabla W_{ij} \omega_j \\ & + \sum_{j=1}^{J^\partial} \left[\frac{p_{ref}}{\rho_0} + \frac{c_0}{2} (\mathbf{v}_{0i} - \mathbf{v}_{0j}) \mathbf{n}_{ij} \right] W_{ij} \mathbf{n}_j \omega_j^\partial, \end{aligned} \quad (7.1)$$

A sensitivity analysis on the particle shifting coefficient has been conducted to determine its optimal value. The L_∞ and the L_2 norms for the particle concentration gradient are shown in Figure 7.4, for $P_S = 2, 5, 10, 20$, these results have been obtained imposing the initial analytical solution, at $t = 0$ s, during the entire simulation. Note that the

original formulation of Neuhauser (2014) is recalled using P_S equal to 1.

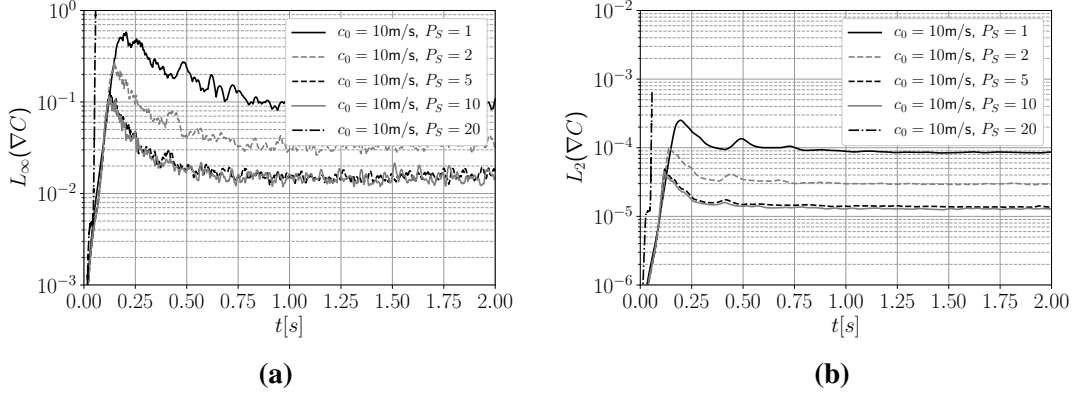


Figure 7.4 TGV test case. Sensitive analysis for P_S imposed analytical solutions. Results for $\Delta/L = 0.00625$

Increasing the P_S value improves the particle distribution quality until it is reached an optimal value. The P_S value cannot be augmented indefinitely, the Fick's based techniques move the particles towards areas of lower concentration, imposing the P_S value higher than 10, Figure 7.5 (d), leads to numerical instabilities, as seen in Figure 7.4, the magnitude of the shifting correction tends to displace the particles beyond their optimal location, affecting negatively the particle distribution, as shown in Figure 7.5. In the following simulations, the equation (7.1) is tuned with P_S equal to 10.

7.2.2 Fictitious Pressure field

In the second optimization proposed, a fictitious pressure field, P^F , is introduced to indirectly connect the shifting correction with the governing equations. In fact, in ALE-SPH scheme, there is no explicit theoretical connection between the transport velocity correction term, equation (6.33) and the other equations of the system (5.39), in particular the volume equation (second equation in the system (5.39)), which is computed with the divergence of the transport velocity itself.

To overcome this lack, a formalism, that recalls the Euler equations (5.10), is used to introduce a dependency between the changes in the particle volume, equation (7.2a), and the changes in the Lagrangian trajectory, equation (7.2b), with the following formulation,

$$\frac{d\omega}{dt} = -\omega \nabla \mathbf{v}_0 \quad (7.2a)$$

$$\left(\frac{d\mathbf{v}_0}{dt} \right)_{corr} = -\frac{1}{\rho} \nabla P^F \quad (7.2b)$$

These equations represent respectively the volume variation and the transport velocity variation with respect to the fluid velocity. P^F is a fictitious pressure field and, to create

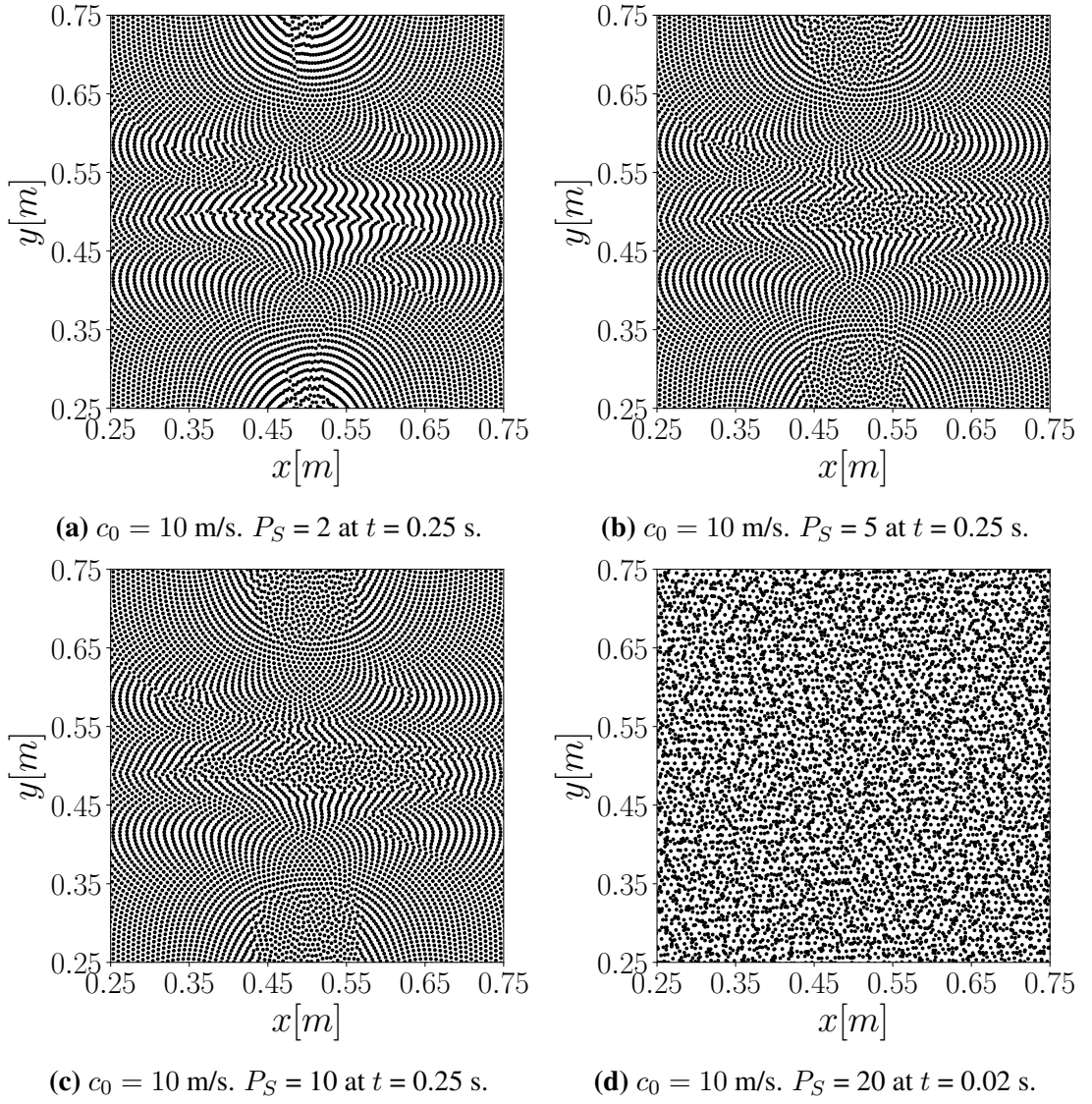


Figure 7.5 Sensitive analysis for P_S imposed analytical solutions. Results for $\Delta/L = 0.00625$, a restricted domain is shown for visual evaluation.

the sought connection, it is defined relying on the particle volume ω and the its original value ω_0 ,

$$P^F = \frac{\rho_0 c_0^2}{\gamma} + \frac{\rho_0 c_0^2}{\gamma} \left[\left(\frac{\omega_0}{\omega} \right)^\gamma - 1 \right] \quad (7.3)$$

To introduce the link between the divergence of the transport velocity and the variation of volume, equation (6.33) has been rewritten,

$$\begin{aligned} \left(\frac{d\mathbf{v}_{0i}}{dt} \right)_{corr} = & \sum_{j=1}^J \left[-\frac{1}{\rho_0} \frac{P_i^F + P_j^F}{2} - \frac{c_0}{2} (\mathbf{v}_{0i} - \mathbf{v}_{0j}) \mathbf{n}_{ij} \right] \nabla W_{ij} \omega_j \\ & + \sum_{j=1}^{J^\partial} \left[\frac{p_{ref}}{\rho_0} + \frac{c_0}{2} (\mathbf{v}_{0i} - \mathbf{v}_{0j}) \mathbf{n}_{ij} \right] W_{ij} \mathbf{n}_j \omega_j^\partial, \end{aligned} \quad (7.4)$$

The fictitious pressure field P^F does not act on the physics of the problem, it is only related to the definition of the transport velocity correction. It is thought to be a smoothing term and it is able to counterbalance volume changes.

To evaluate whether the novel particle shifting techniques, equations (7.1) and (7.4), have been optimized, compared to the original formulation (6.33), the analyses described in Section 7.1 have been repeated, using the reference speed of sound equal to 10 m/s, which corresponds to its minimal value.

7.2.3 Results solving the kinematic equation

As a first investigation step, the initial analytical TGV solution has been imposed to evaluate exclusively how the new formulations are reflected on the particles kinematic and on the particle distribution quality.

The L_∞ norm and the ∇C fields are shown in Figure 7.6. Compared to the original formulation, the particle shifting coefficient and the fictitious pressure field are able to maintain a lower particle concentration gradient in the entire domain during the simulation. Moreover, the new results are similar to the ones obtained with the original formulation but using $c_0 = 100$ m/s instead, Figure 7.1 (d), which is an order of magnitude higher than the one adopted in these tests.

7.2.4 Results solving the Navier-Stokes equations

As a second investigation step, the simulations have been repeated solving the full set of governing equations to include the accuracy evaluation for the proposed particle shifting techniques. Again, the L_∞ norm and the ∇C fields are shown in Figure 7.7, these results confirm the effectiveness of equations (7.1) and (7.4), demonstrating that the particle concentration uniformity is enhanced.

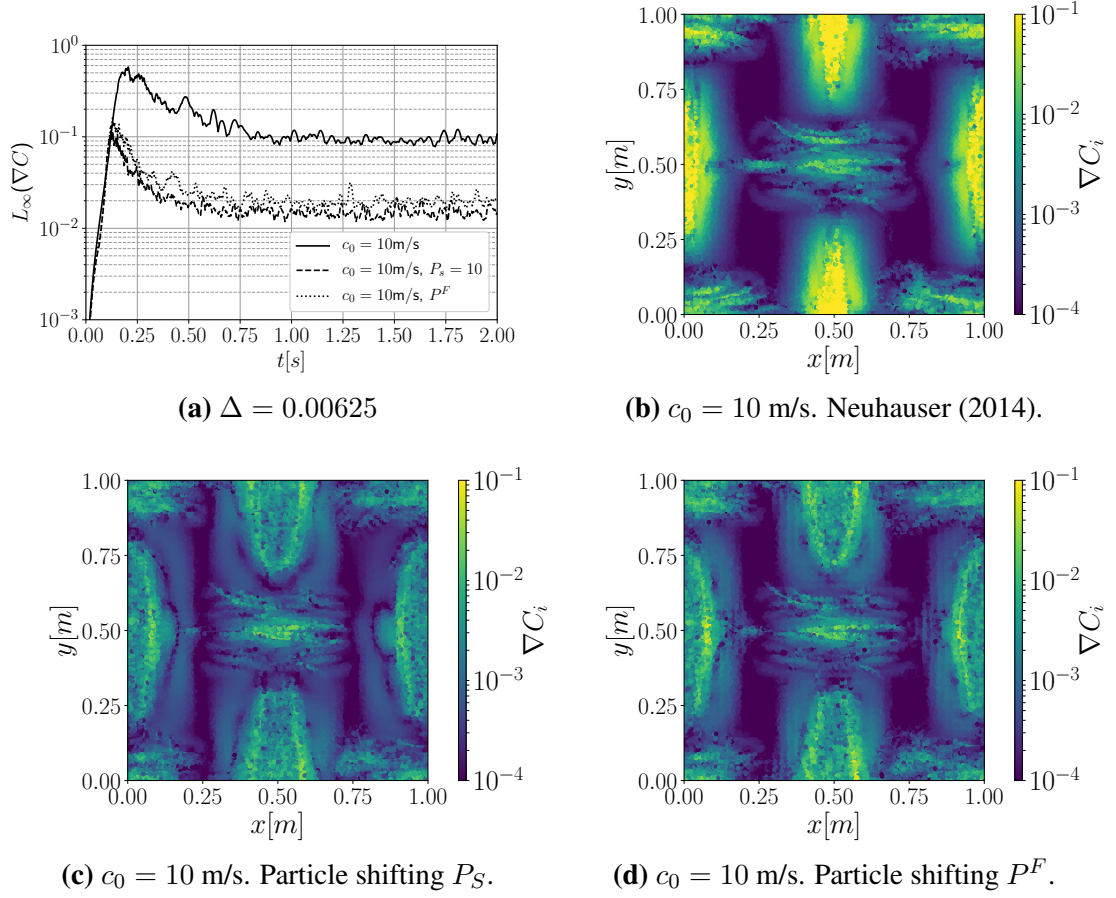


Figure 7.6 TGV test case. Formulations comparison: (b) equation (6.33) Neuhauser (2014) formulation, (c) equation (7.1) shifting technique with particle shifting coefficient P_S , (d) equation (7.4) shifting technique with fictitious pressure field P^F , imposed analytical solution. Results for $\Delta/L = 0.00625$ at $t = 0.2 \text{ s}$.

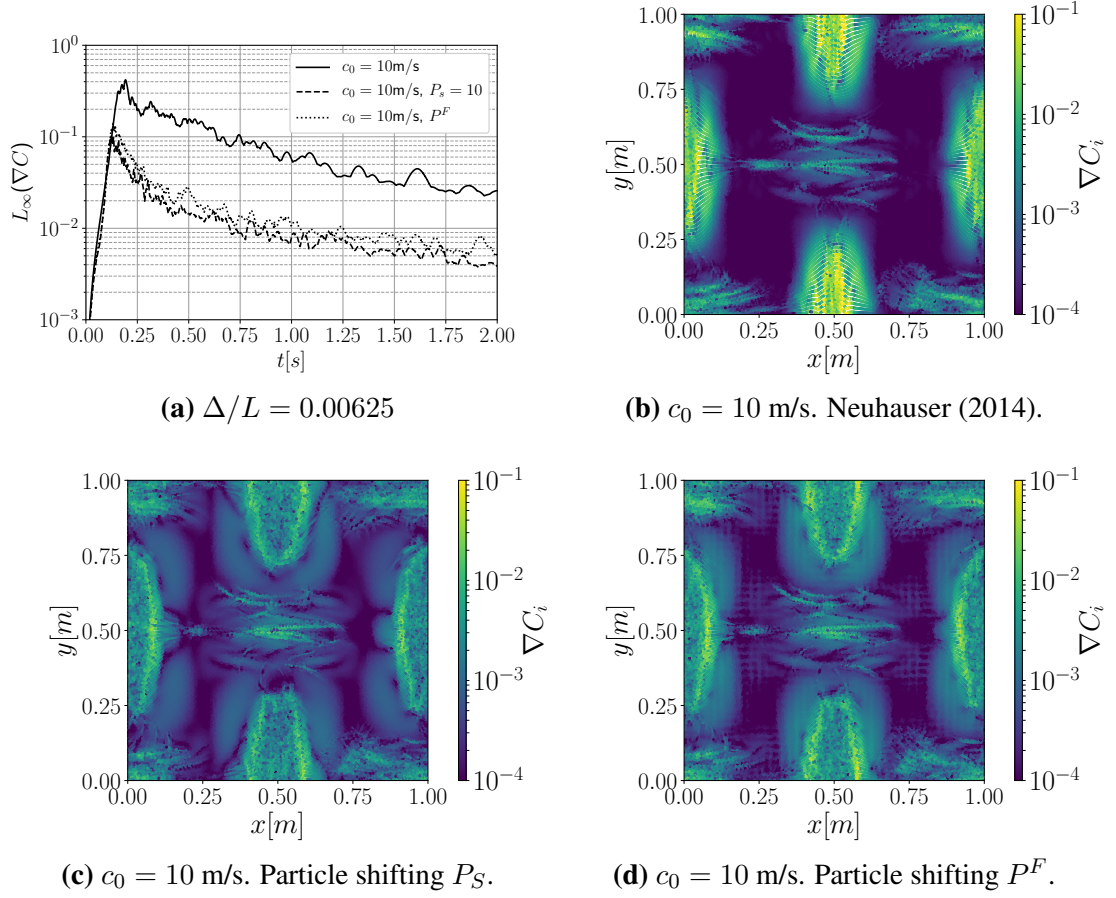


Figure 7.7 TGV test case. Formulations comparison: (b) equation(6.33) Neuhauser (2014) formulation, (c) equation (7.1) shifting technique with particle shifting coefficient P_S , (d) equation (7.4) shifting technique with fictitious pressure field P^F , solved simulations. Results for $\Delta/L = 0.00625$ at $t = 0.2$ s.

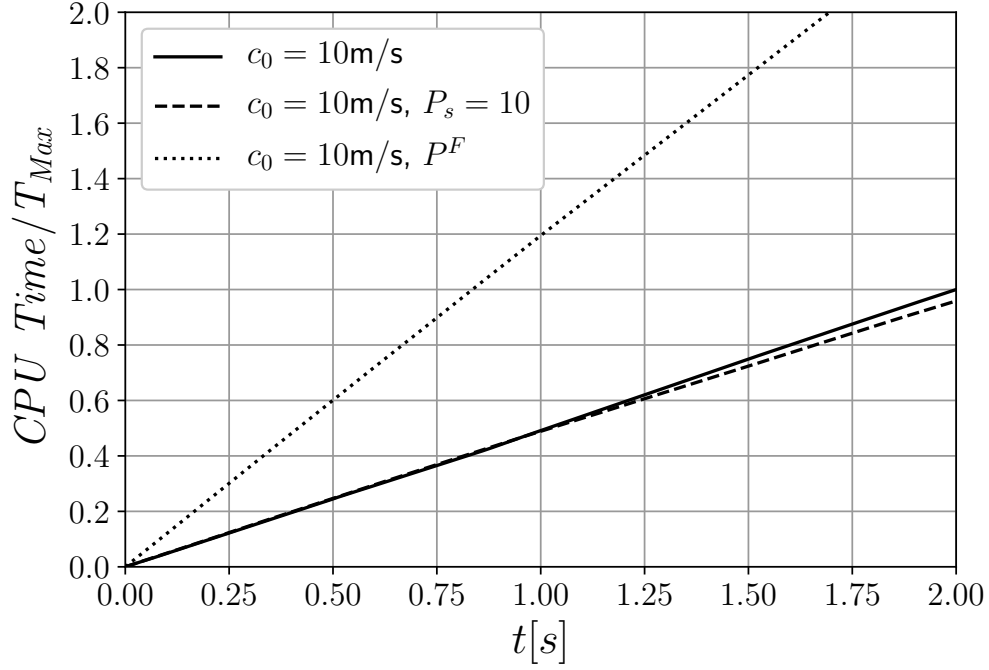


Figure 7.8 TGV test case. Time comparison: Neuhauser (2014) formulation, particle shifting coefficient P_S , fictitious pressure field P^F .

In Figure 7.3 the computational times are shown, as expected introducing a particle shifting coefficient does not add any computational time while an additional fictitious pressure field produces an affordable overhead.

The error between the analytical velocity and the computed velocity has been evaluated; a convergence analysis has been carried on, using $\Delta/L = 0.05, 0.025, 0.0125, 0.00625$ and h/Δ equal to 2, showing the results in Figure 7.9, while the numerical values for the convergence rates θ and the norms of the velocity errors are reported in Table 7.1. The theoretical second-order convergence rate is restored, before reaching saturation, using the particle shifting coefficient and the fictitious pressure field formulations. The velocity field and the error on the velocity field are shown in Figures 7.10,

Δ/L	Neuhauser (2014)		P_S		P^F	
	$L_2(\mathbf{v}_x)$	θ	$L_2(\mathbf{v}_x)$	θ	$L_2(\mathbf{v}_x)$	θ
0.05	0.16288		0.13321		0.12889	
0.025	0.05559	1.55	0.02594	2.36	0.02846	2.17
0.0125	0.03373	0.72	0.00667	1.95	0.00760	1.90
0.00625	0.01761	0.93	0.00238	0.68	0.00542	0.48

Table 7.1 TGV test case. Convergence analysis results, $L_2|\mathbf{v}_x - \mathbf{v}_{xA}|$ and convergence ratio θ , explicit shifting.

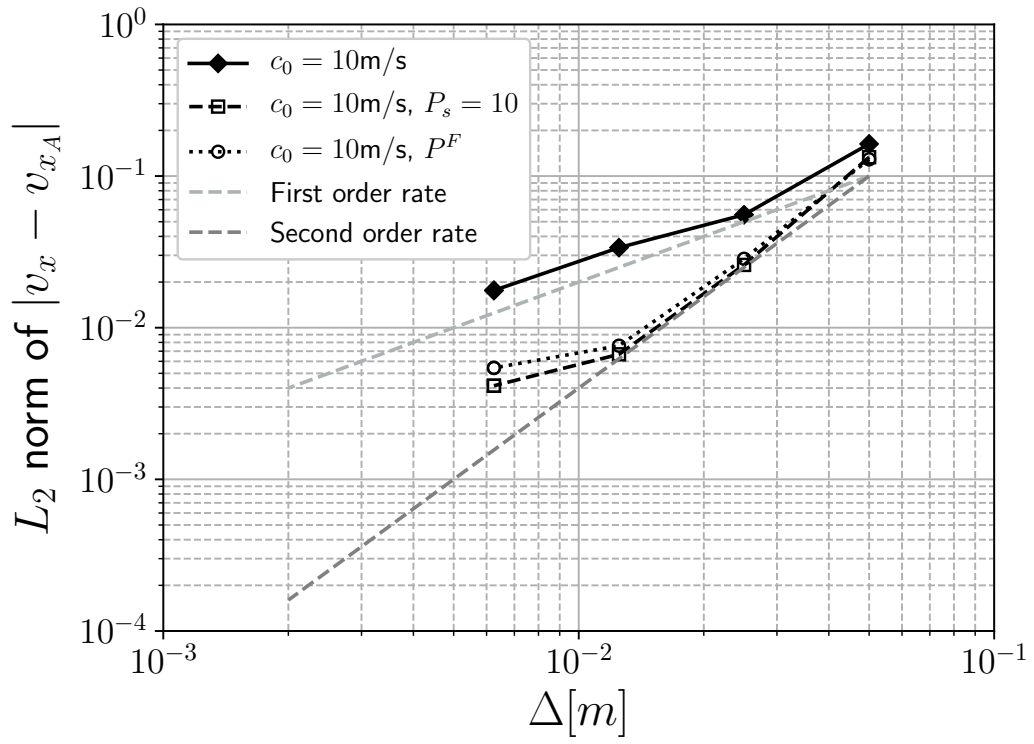


Figure 7.9 TGV test case. Convergence analysis $L_2|v_x - v_{x_A}|$, explicit particle shifting comparison at $t = 1s$.

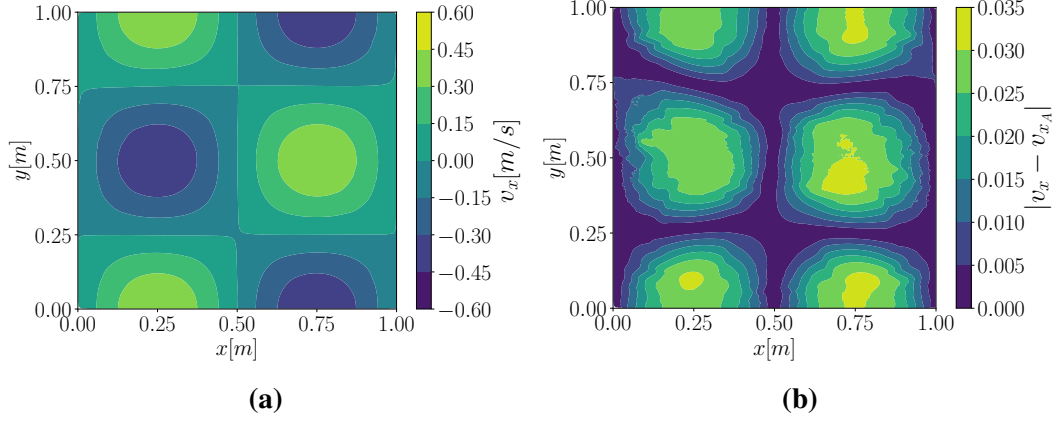


Figure 7.10 TGV test case. $\Delta/L = 0.00625$, Neuhauser (2014) formulation: (a) velocity x component, (b) error on the velocity x component at $t = 1$ s.

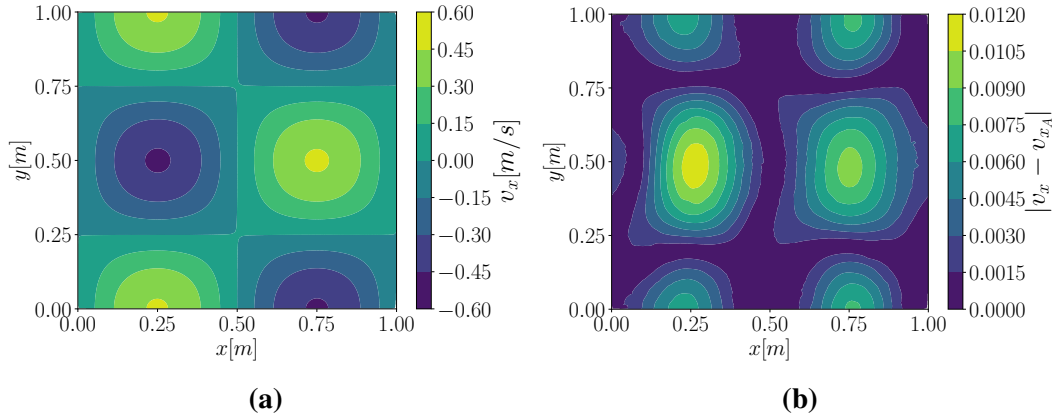


Figure 7.11 TGV test case. $\Delta/L = 0.00625$, particle shifting coefficient P_S : (a) velocity x component, (b) error on the velocity x component at $t = 1$ s.

7.11 and 7.12 for a visual evaluation.

The analysis of the kinetic energy shows that the new proposed formulations closely match the analytical solution, (Figure 7.13 (c) and Figure 7.13 (d) with curves basically overlapped), while the original formulation (Figure 7.13 (b)) is less accurate; these trends are better seen looking at the errors in Figure 7.13 (a), which demonstrate that the novel particle shifting technique reduce the errors by almost an order of magnitude compared to the original one.

7.3 Summary and conclusions

In these studies, the explicit particle shifting technique presented in Neuhauser (2014) has been optimized, showing relevant improvements. Two different formulations have been proposed, in the first one, the magnitude of the shifting has been augmented through a coefficient, while in the second one, a fictitious pressure field has been introduced. Both significantly improve the quality of the particle distribution as seen in

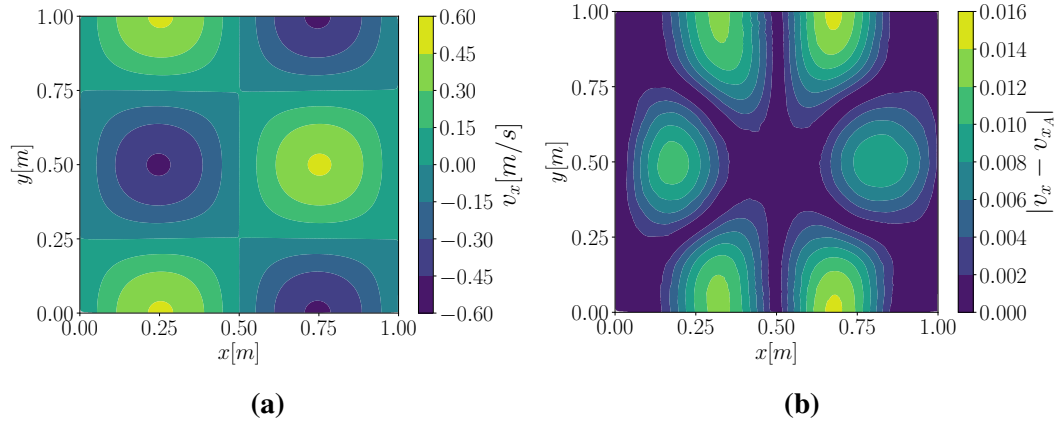


Figure 7.12 TGV test case. $\Delta/L = 0.00625$, fictitious pressure field P^F : (a) velocity x component, (b) error on the velocity x component at $t = 1$ s.

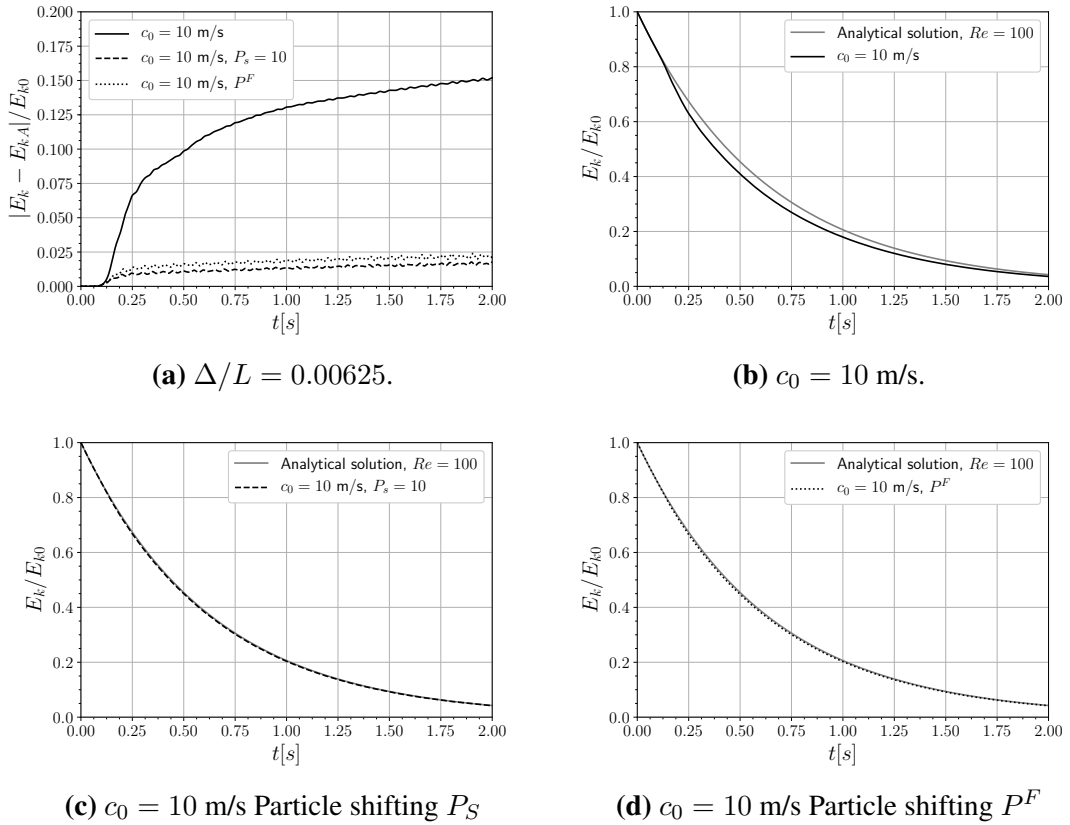


Figure 7.13 TGV test case. Kinematic energy analysis: (a) Kinematic energy error (6.33), Neuhauser (2014) formulation, (c) (7.1), particle shifting coefficient P_S , (d) (7.4), fictitious pressure field P^F at $t = 0.2$ s.

Figure 7.7, gaining almost an order of magnitude in accuracy, the convergence rate has been restored to the theoretical one, higher than 2, before reaching saturation. Adopting the same value of reference speed of sound keeps the computational times similar for the different shifting formulation, as reported in Figure 7.8; the fictitious pressure formulation has slightly increased the computational overheads but it does not produce the expected global improvements, nevertheless deeper investigation could try to better catch these aspects. In this thesis, due to its simplicity and effectiveness, the particle shifting technique presented in equation (7.1) has been used in other validation tests as optimized explicit formulation.

Chapter 8

Implicit Iterative Particle Shifting Technique

In this work a novel shifting method is presented aiming to solve the issues of the classical explicit shifting techniques. The schemes, described in Chapter 6, assume that particles concentration is a valid indicator for the discretization error of the SPH interpolation accounting for particles disorder and thus, the particle distribution is adjusted accordingly, (Colagrossi et al. 2012). However, the explicit nature of these techniques implies that they cannot guarantee a predefined and spatially homogeneous SPH discretization error, nor do they ensure that the discretization error is minimised, either locally or globally. For these reasons, none of the methods reported in Chapter 6 is identified as broadly and generally accepted as superior, indeed particle shifting techniques are still a field under investigation.

In the next sections, a new implicit particle shifting method is presented, in which the corrected particle positions are computed adopting an iterative scheme to obtain a spatially homogeneous particle distribution, imposing a predefined level of particle concentration. For clarity, the novel shifting method is derived in 1D and 2D.

8.1 1D formulation

The technique can be described starting by defining a generic function in 1D $f(\mathbf{X}): \mathbb{R}^n \rightarrow \mathbb{R}$ where $\mathbf{X} = (x_1, \dots, x_i, \dots, x_n)$ represents the vector of particles position and n the number of particles in the domain. The objective is to identify the new particle's position array $\bar{\mathbf{X}} = (\bar{x}_1, \dots, \bar{x}_i, \dots, \bar{x}_n)$ in which the following condition holds,

$$f_i(\bar{\mathbf{X}}) = 0, \quad i = 1, \dots, n, \quad (8.1)$$

where $f_i(\bar{\mathbf{X}})$ is the value of the scalar function f at particle i .

To find the roots of equation (8.1) a Newton-Raphson procedure is adopted, therefore the Taylor expansion truncated at the first-order of function f at position $\bar{x}_i$ has been

considered,

$$f_i(\bar{\mathbf{X}}) = f_i(\mathbf{X}) + \sum_{j=1}^J \frac{\partial f_i(\mathbf{X})}{\partial x_j} (\bar{x}_j - x_j) + \mathcal{O}(\bar{x}_i - x_i)^2. \quad (8.2)$$

Here it has been assumed that the function f corresponds to the derivative of the particle concentration,

$$f(\mathbf{X}) = \frac{\partial C(\mathbf{X})}{\partial x}, \quad (8.3)$$

therefore,

$$\frac{\partial f(\mathbf{X})}{\partial x_j} = \frac{\partial}{\partial x_j} \left(\frac{\partial C(\mathbf{X})}{\partial x} \right), \quad (8.4)$$

substituting equations (8.3) and (8.4) in equation (8.2) leads to

$$\frac{\partial C_i(\bar{\mathbf{X}})}{\partial x} = \frac{\partial C_i(\mathbf{X})}{\partial x} + \sum_{j=1}^J \frac{\partial}{\partial x_j} \left(\frac{\partial C_i(\mathbf{X})}{\partial x} \right) (\bar{x}_j - x_j) + \mathcal{O}(\bar{x}_i - x_i)^2. \quad (8.5)$$

The particle concentration SPH derivative approximation is,

$$\frac{\partial C_i(\mathbf{X})}{\partial x} = \sum_{k=1}^K \frac{\partial W(x_i - x_k)}{\partial x_k} \omega_k, \quad (8.6)$$

where K is the number of neighbouring particles inside the kernel support of particle i . Then, the summation in equation (8.5) is reformulated

$$\frac{\partial}{\partial x_j} \left(\frac{\partial C(\mathbf{X})}{\partial x} \right) = \sum_{k=1}^K \frac{\partial}{\partial x_j} \left(\frac{\partial W(x_i - x_k)}{\partial x_k} \right) \omega_k. \quad (8.7)$$

The only term in equation (8.7) which is non-null is the one in which $j = k$, therefore, it is rewritten as,

$$\frac{\partial}{\partial x_j} \left(\frac{\partial C_i(\mathbf{X})}{\partial x} \right) = \frac{\partial^2 W(x_i - x_j)}{\partial x_j^2} \omega_j. \quad (8.8)$$

Substituting equation (8.8) in equation (8.5) leads to

$$\frac{\partial C_i(\bar{\mathbf{X}})}{\partial x} = \frac{\partial C_i(\mathbf{X})}{\partial x} + \sum_{j=1}^J \frac{\partial^2 W(x_i - x_j)}{\partial x_j^2} \omega_j (\bar{x}_j - x_j) + \mathcal{O}(\bar{x}_i - x_i)^2. \quad (8.9)$$

As explained before, in order to improve the accuracy of the SPH operator at the discrete level, the particle concentration C has to be uniform and thus its derivative should be equal to zero. Imposing this constraint in equation (8.9), and neglecting non-linear

terms, leads to the following equation for the generic particle i ,

$$\sum_{j=1}^J \frac{\partial^2 W(x_i - x_j)}{\partial x_j^2} \omega_j (\bar{x}_j - x_j) = \frac{\partial C_i(\mathbf{X})}{\partial x}. \quad (8.10)$$

In equation (8.10) the terms on the left-hand side $(\bar{x}_j - x_j)$ are the unknowns which correspond to the particle shifting $\delta \mathbf{x}_j$ for $j = 1, \dots, n$, which have to be applied to all particles respectively to obtain a uniform particle concentration. This leads to a linear system of equations which can be expressed in matrix form (where $W(x_i - x_j)$ is replaced with W_{ij} for brevity) as follows:

$$\underbrace{\begin{bmatrix} \frac{\partial^2 W_{11}}{\partial x_1^2} \omega_1 & \dots & \frac{\partial^2 W_{1i}}{\partial x_i^2} \omega_i & \dots & \frac{\partial^2 W_{1n}}{\partial x_n^2} \omega_n \\ \vdots & & \vdots & & \vdots \\ \frac{\partial^2 W_{i1}}{\partial x_1^2} \omega_1 & \dots & \frac{\partial^2 W_{ii}}{\partial x_i^2} \omega_i & \dots & \frac{\partial^2 W_{in}}{\partial x_n^2} \omega_n \\ \vdots & & \vdots & & \vdots \\ \frac{\partial^2 W_{n1}}{\partial x_1^2} \omega_1 & \dots & \frac{\partial^2 W_{ni}}{\partial x_i^2} \omega_i & \dots & \frac{\partial^2 W_{nn}}{\partial x_n^2} \omega_n \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} (\bar{x}_1 - x_1) \\ \vdots \\ (\bar{x}_i - x_i) \\ \vdots \\ (\bar{x}_n - x_n) \end{bmatrix}}_{\mathbf{X}} \approx \underbrace{\begin{bmatrix} \frac{\partial C(x_1)}{\partial x} \\ \vdots \\ \frac{\partial C(x_i)}{\partial x} \\ \vdots \\ \frac{\partial C(x_n)}{\partial x} \end{bmatrix}}_{\mathbf{B}} \quad (8.11)$$

By solving the linear system of equation (8.11), the new particle positions $\bar{\mathbf{X}} = (\bar{x}_1, \dots, \bar{x}_i, \dots, \bar{x}_n)$ are found.

Note that in equation (8.2) the problem has been linearised by neglecting high-order term in the Taylor series, this effectively corresponds to a Newton-Raphson algorithm to find the solution of equation (8.9) and thus, an iterative approach is necessary in order to obtain the particle distribution which fulfil the condition $\partial C_i / \partial x = 0$ for $i = 1, \dots, n$.

8.2 2D formulation

Similarly, a generic function is defined in 2D, $f(\mathbf{X}, \mathbf{Y}): \mathbb{R}^{2n} \rightarrow \mathbb{R}$ with $\mathbf{X} = (x_1, \dots, x_i, \dots, x_n)$ and $\mathbf{Y} = (y_1, \dots, y_i, \dots, y_n)$. As previously explained for the 1D case, the objective is to obtain the updated particle distribution represented by $\bar{\mathbf{X}} = (\bar{x}_1, \dots, \bar{x}_i, \dots, \bar{x}_n)$ and $\bar{\mathbf{Y}} = (\bar{y}_1, \dots, \bar{y}_i, \dots, \bar{y}_n)$ in which

$$f_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}}) = 0, i = 1, \dots, n. \quad (8.12)$$

Recalling equation (8.2), the Taylor series expansion truncated at the first order of function f at position $\bar{x}_i, \bar{y}_i$ is

$$\begin{aligned}
f_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}}) = & f_i(\mathbf{X}, \mathbf{Y}) + \sum_{j=1}^J \frac{\partial f_i(\mathbf{X}, \mathbf{Y})}{\partial x_j} (\bar{x}_j - x_j) + \\
& + \sum_{j=1}^J \frac{\partial f_i(\mathbf{X}, \mathbf{Y})}{\partial y_j} (\bar{y}_j - y_j) + \mathcal{O}((\bar{x}_i - x_i)(\bar{y}_i - y_i))^2.
\end{aligned} \tag{8.13}$$

For the two-dimensional case, two different functions express the gradient of particle concentration components along the two axes,

$$f^{(1)}(\mathbf{X}, \mathbf{Y}) = \frac{\partial C(\mathbf{X}, \mathbf{Y})}{\partial x}, \tag{8.14}$$

$$f^{(2)}(\mathbf{X}, \mathbf{Y}) = \frac{\partial C(\mathbf{X}, \mathbf{Y})}{\partial y}, \tag{8.15}$$

therefore, equations (8.14) and (8.15) are substituted separately in equation (8.13) leading to separate equations for each spatial component,

$$\begin{aligned}
\frac{\partial C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}})}{\partial x} = & \frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x} + \underbrace{\sum_{j=1}^J \frac{\partial}{\partial x_j} \left[\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x} \right]}_{\text{second derivative}} (\bar{x}_i - x_i) + \\
& + \underbrace{\sum_{j=1}^J \frac{\partial}{\partial y_j} \left[\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x} \right]}_{\text{cross derivative}} (\bar{y}_i - y_i) + \mathcal{O}((\bar{x}_i - x_i)(\bar{y}_i - y_i))^2,
\end{aligned} \tag{8.16}$$

along x axis,

$$\begin{aligned}
\frac{\partial C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}})}{\partial y} = & \frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y} + \underbrace{\sum_{j=1}^J \frac{\partial}{\partial x_j} \left[\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y} \right]}_{\text{cross derivative}} (\bar{x}_i - x_i) + \\
& + \underbrace{\sum_{j=1}^J \frac{\partial}{\partial y_j} \left[\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y} \right]}_{\text{second derivative}} (\bar{y}_i - y_i) + \mathcal{O}((\bar{x}_i - x_i)(\bar{y}_i - y_i))^2,
\end{aligned} \tag{8.17}$$

along y axis.

The SPH approximation for ∇C components in 2D are

$$\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x} = \sum_{k=1}^K \frac{\partial W((x_i - x_k), (y_i - y_k))}{\partial x_k} \omega_k, \tag{8.18}$$

$$\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y} = \sum_{k=1}^K \frac{\partial W((x_i - x_k), (y_i - y_k))}{\partial y_k} \omega_k. \quad (8.19)$$

The second and the cross derivatives of equations (8.16) and (8.17) are manipulated as previously illustrated in equation (8.7).

For the x axis, introducing equation (8.18) in the first-order term of equation (8.16),

$$\frac{\partial}{\partial x_j} \left(\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x} \right) = \sum_{k=1}^K \frac{\partial}{\partial x_j} \left(\frac{\partial W((x_i - x_k), (y_i - y_k))}{\partial x_k} \right) \omega_k, \quad (8.20)$$

$$\frac{\partial}{\partial y_j} \left(\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x} \right) = \sum_{k=1}^K \frac{\partial}{\partial y_j} \left(\frac{\partial W((x_i - x_k), (y_i - y_k))}{\partial x_k} \right) \omega_k, \quad (8.21)$$

and for y axis, introducing equation (8.19) in the first-order term of equation (8.17),

$$\frac{\partial}{\partial x_j} \left(\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y} \right) = \sum_{k=1}^K \frac{\partial}{\partial x_j} \left(\frac{\partial W((x_i - x_k), (y_i - y_k))}{\partial y_k} \right) \omega_k, \quad (8.22)$$

$$\frac{\partial}{\partial y_j} \left(\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y} \right) = \sum_{k=1}^K \frac{\partial}{\partial y_j} \left(\frac{\partial W((x_i - x_k), (y_i - y_k))}{\partial y_k} \right) \omega_k. \quad (8.23)$$

On the RHS of equations (8.20) and (8.23) the only non-null terms of the sums are the ones where $j = k$. Therefore equations (8.20) and (8.23) are written as,

$$\frac{\partial}{\partial x_j} \left(\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x} \right) = \frac{\partial^2 W((x_i - x_k), (y_i - y_k))}{\partial x_j^2} \omega_j, \quad (8.24)$$

$$\frac{\partial}{\partial y_j} \left(\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x} \right) = \frac{\partial^2 W((x_i - x_k), (y_i - y_k))}{\partial y_j \partial x_j} \omega_j, \quad (8.25)$$

along x axis,

$$\frac{\partial}{\partial x_j} \left(\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y} \right) = \frac{\partial^2 W((x_i - x_k), (y_i - y_k))}{\partial x_j \partial y_j} \omega_j, \quad (8.26)$$

$$\frac{\partial}{\partial y_j} \left(\frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y} \right) = \frac{\partial^2 W((x_i - x_k), (y_i - y_k))}{\partial y_j^2} \omega_j, \quad (8.27)$$

along y axis.

Substituting equations (8.24) and (8.27) in (8.16) and (8.17) leads to

$$\begin{aligned}
\frac{\partial C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}})}{\partial x} &= \frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x} + \\
&+ \sum_{j=1}^J \frac{\partial^2 W((x_i - x_k), (y_i - y_k))}{\partial x_j^2} \omega_j(\bar{x}_j - x_j) + \\
&+ \sum_{j=1}^J \frac{\partial^2 W((x_i - x_k), (y_i - y_k))}{\partial x_j \partial y_j} \omega_j(\bar{y}_j - y_j) + \\
&+ \mathcal{O}((\bar{x}_i - x_i)(\bar{y}_i - y_i))^2,
\end{aligned} \tag{8.28}$$

for the x axis,

$$\begin{aligned}
\frac{\partial C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}})}{\partial y} &= \frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y} + \\
&+ \sum_{j=1}^J \frac{\partial^2 W((x_i - x_k), (y_i - y_k))}{\partial y_j \partial x_j} \omega_j(\bar{x}_j - x_j) + \\
&+ \sum_{j=1}^J \frac{\partial^2 W((x_i - x_k), (y_i - y_k))}{\partial y_j^2} \omega_j(\bar{y}_j - y_j) + \\
&+ \mathcal{O}((\bar{x}_i - x_i)(\bar{y}_i - y_i))^2,
\end{aligned} \tag{8.29}$$

for y axis.

As explained previously, in order to improve the accuracy of the SPH operator at the discrete level, the particle concentration C has to be uniform and therefore each component of its derivatives should be equal to zero. In 2D, in order to satisfy equation (8.12) and by neglecting non-linear terms, two linear equations have to be solved to for the generic particle i ,

$$\begin{aligned}
&\sum_{j=1}^J \frac{\partial^2 W((x_i - x_j), (y_i - y_k))}{\partial x_j^2} \omega_j(\bar{x}_j - x_j) + \\
&+ \sum_{j=1}^J \frac{\partial^2 W((x_i - x_j), (y_i - y_k))}{\partial x_j \partial y_j} \omega_j(\bar{y}_j - y_j) = \frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial x},
\end{aligned} \tag{8.30}$$

and

$$\begin{aligned}
&\sum_{j=1}^J \frac{\partial^2 W((x_i - x_j), (y_i - y_k))}{\partial y_j \partial x_j} \omega_j(\bar{x}_j - x_j) + \\
&+ \sum_{j=1}^J \frac{\partial^2 W((x_i - x_j), (y_i - y_k))}{\partial y_j^2} \omega_j(\bar{y}_j - y_j) = \frac{\partial C_i(\mathbf{X}, \mathbf{Y})}{\partial y}.
\end{aligned} \tag{8.31}$$

The notation $W((x_i - x_j), (y_i - y_j))$ is shorten to W_{ij} , the 2D system of linear equations

in a matrix form is:

$$\begin{aligned}
& \underbrace{\left[\begin{array}{cccccc} \frac{\partial^2 W_{11}}{\partial x_1^2} \omega_1 & \frac{\partial^2 W_{11}}{\partial x_1 \partial y_1} \omega_1 & \dots & \frac{\partial^2 W_{1i}}{\partial x_i^2} \omega_i & \frac{\partial^2 W_{1i}}{\partial x_i \partial y_i} \omega_i & \frac{\partial^2 W_{1n}}{\partial x_n \partial y_n} \omega_n \\ \frac{\partial^2 W_{11}}{\partial y_1 \partial x_1} \omega_1 & \frac{\partial^2 W_{11}}{\partial y_1^2} \omega_1 & \dots & \frac{\partial^2 W_{1i}}{\partial y_i \partial x_i} \omega_i & \frac{\partial^2 W_{1i}}{\partial y_i \partial y_i} \omega_i & \frac{\partial^2 W_{1n}}{\partial y_n \partial y_n} \omega_n \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ \frac{\partial^2 W_{i1}}{\partial x_1^2} \omega_1 & \frac{\partial^2 W_{i1}}{\partial x_1 \partial y_1} \omega_1 & \dots & \frac{\partial^2 W_{ii}}{\partial x_i^2} \omega_i & \frac{\partial^2 W_{ii}}{\partial x_i \partial y_i} \omega_i & \frac{\partial^2 W_{in}}{\partial x_n \partial y_n} \omega_n \\ \frac{\partial^2 W_{i1}}{\partial y_1 \partial x_1} \omega_1 & \frac{\partial^2 W_{i1}}{\partial y_1^2} \omega_1 & \dots & \frac{\partial^2 W_{ii}}{\partial y_i \partial x_i} \omega_i & \frac{\partial^2 W_{ii}}{\partial y_i \partial y_i} \omega_i & \frac{\partial^2 W_{in}}{\partial y_n \partial y_n} \omega_n \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ \frac{\partial^2 W_{n1}}{\partial x_1^2} \omega_1 & \frac{\partial^2 W_{n1}}{\partial x_1 \partial y_1} \omega_1 & \dots & \frac{\partial^2 W_{ni}}{\partial x_i^2} \omega_i & \frac{\partial^2 W_{ni}}{\partial x_i \partial y_i} \omega_i & \frac{\partial^2 W_{nn}}{\partial x_n \partial y_n} \omega_n \\ \frac{\partial^2 W_{n1}}{\partial y_1 \partial x_1} \omega_1 & \frac{\partial^2 W_{n1}}{\partial y_1^2} \omega_1 & \dots & \frac{\partial^2 W_{ni}}{\partial y_i \partial x_i} \omega_i & \frac{\partial^2 W_{ni}}{\partial y_i \partial y_i} \omega_i & \frac{\partial^2 W_{nn}}{\partial y_n \partial y_n} \omega_n \end{array} \right]}_{\mathbf{A}} \underbrace{\left[\begin{array}{c} (\bar{x}_1 - x_1) \\ (\bar{y}_1 - y_1) \\ \vdots \\ (\bar{x}_i - x_i) \\ (\bar{y}_i - y_i) \\ \vdots \\ (\bar{x}_n - x_n) \\ (\bar{y}_n - y_n) \end{array} \right]}_{\mathbf{X}} \approx \underbrace{\left[\begin{array}{c} \frac{\partial C(x_1, y_1)}{\partial x} \\ \frac{\partial C(x_1, y_1)}{\partial y} \\ \vdots \\ \frac{\partial C(x_i, y_i)}{\partial x} \\ \frac{\partial C(x_i, y_i)}{\partial y} \\ \vdots \\ \frac{\partial C(x_n, y_n)}{\partial x} \\ \frac{\partial C(x_n, y_n)}{\partial y} \end{array} \right]}_{\mathbf{B}}
\end{aligned}
\tag{8.32}$$

The extension in 3D is analogous, in general there are $d \cdot n$ linear equations in the system, where d is number of spatial dimensions and, the size of $\mathbf{A}$ is $d \cdot n \times d \cdot n$.

The matrix $\mathbf{A}$, defined in equations (8.11) and (8.32), is sparse and it is convenient to adopt an iterative solver for the linear system, similar to the one used for the Poisson equation of pressure in incompressible SPH schemes, (Xu et al. 2009; Lind et al. 2012; Chow et al. 2018). In the present work the Jacobi preconditioner and the BiCGStab have been utilized and the Wendland C6 kernel, (equation (4.29)), has been used in all the SPH interpolations. In the implicit formulation, two different levels of iterations are present, the external one is related to the Newton-Raphson algorithm adopted to solve the non-linear equations (8.2) and (8.13), whereas internal iterations are referred to the linear system solver, equations (8.11) and (8.32) in 1D and 2D.

8.3 Numerical test cases

In order to validate the proposed method two different numerical experiments are initially presented in this section. A static case is used to analyze the robustness of the IIPS algorithm for different SPH parameters (such as the smoothing length h and the particle size Δ). Later, the algorithm has been evaluated in a test that generates a continuous distortion of the particle distribution. In this second numerical assessment, particles are kept in motion by imposing the kinematic that reproduces counter-rotating vortices in the domain, through the entire simulation. This flow (that recall the TGV motion, see Section 3.1) has been chosen as a test case because it is known to be very demanding in maintaining uniformity in the distribution.

In both test cases, no physical quantities are attached to the particles (velocity or pressure), only the accuracy of the SPH approximation on the test function is evaluated. In this way, since the main area of interest is the accuracy of the discrete operators, the effectiveness of the proposed iterative shifting formulation is verified regardless of the properties of the specific meshless solver adopted.

The non-dimensional L_2 and L_∞ norms of the ∇C error, defined in Section 3.4, have been employed as a measure of particle disorder. A comparison (in terms of efficiency and accuracy) with the explicit shifting formulation proposed by Vacondio and Rogers (2017) has also been conducted. The accuracy of the spatial SPH interpolation has been evaluated using the test function (3.6) and its derivative norm (3.7).

8.3.1 Static test case

The aim of this numerical experiment is to assess how effective and efficient the IIPS formulation is in restoring a uniform particle distribution starting from a pseudo-random one. The particles distribution perturbation is introduced exclusively at the beginning of the numerical experiment, thus, this test is referred as "static". For this purpose a bi-

periodic squared domain has been initialized with particles placed on a Cartesian grid, then a pseudo-random normalized perturbation with standard deviation σ has been assigned to the particles position and the IIPS algorithm has been activated. With the aim of investigating the performance of the proposed formulation, the IIPS procedure has been run for 100 Newton-Raphson (NR) iterations, it_{Max} , as described in the pseudo-code (Algorithm 1). At the end of each iteration, the particle positions are updated using the results obtained solving the linear system of equations (8.32).

Algorithm 1 Static case

```

1: procedure IMPLICIT ITERATIVE SHIFTING ▷
2:   CartesianGridDistribution()
3:   RandomPerturbation  $\sigma$ 
4:   Compute  $\nabla C_i, L_2(\nabla C), L_2(\partial_x f)$ 
5:   while  $it \leq it_{Max}$  do
6:     AssembleMatrix( $\nabla C_i$ )
7:     LinearSystemSolver(AssembleMatrix)
8:     UpdateParticlePosition(Implicit PST)
9:     Compute  $L_2(\nabla C), L_2(\partial_x f)$ 
10:  end while
11: end procedure

```

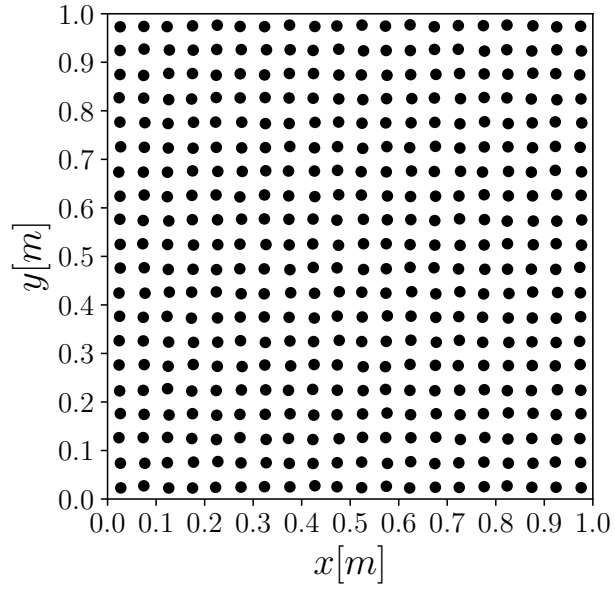
Figure 8.1 shows the particle distribution before and after the IIPS procedure for a resolution $\Delta/L = 0.05$. It is immediately noted how the particle stencil changes, particles arrange themselves into a triangular-staggered configuration. This distribution is reached for the different resolutions ($\Delta/L = 0.025, 0.0125$) presented. It is notable that the final particle configuration is similar to a hexagonal-centred distribution which represents a minimum extreme for ∇C in the particle system configuration, as demonstrated in Colagrossi et al. (2012).

Figure 8.2 shows the maps of ∇C magnitude at the beginning, and at the end of the simulation, it can be seen that the magnitude is reduced in the whole domain by more than two orders of magnitude.

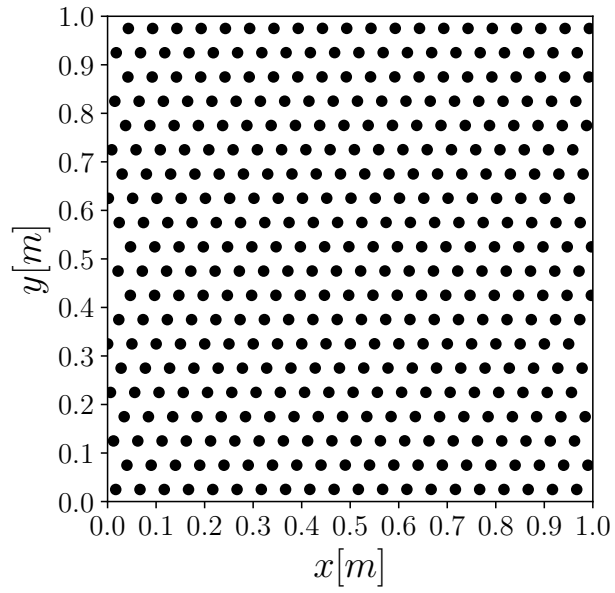
The error on the test function gradient and the test function itself are shown in Figure 8.3 and in Figure 8.3. The error on the test function gradient is reduced and the quality of the SPH approximation for the test function itself increases, distortions, due to the anisotropy of particle distribution, are corrected.

As previously mentioned, different resolutions have been analysed with an increasingly initial disorder, and it has been verified that the IIPS algorithm is able to generate a particle distribution that globally minimizes the $L_2(\nabla C)$. Herein, two different values of σ have been considered, as the initial maximum value of perturbation, and results are reported in Figure 8.5.

In Figure 8.5 (b) and (d) the $L_2(\nabla C)$ for three different particle size ($\Delta/L = 0.05, 0.025, 0.0125$) is shown against the Newton-Raphson (NR) iterations of the IIPS.



(a) $it = 1$



(b) $it = 10$

Figure 8.1 Static test case. $\Delta/L = 0.05$ and initial perturbation $\sigma/\Delta = 0.10$. Particle distribution at (a) $it = 0$ and (b) $it = 10$.

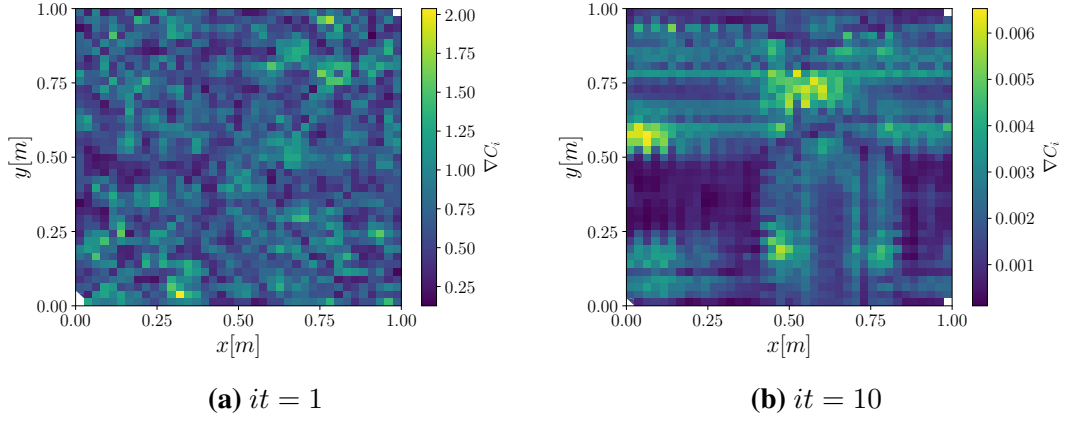


Figure 8.2 Static test case. $\Delta/L = 0.025$ and initial perturbation $\sigma/\Delta = 0.10$. ∇C magnitude at (a) $it = 0$ and (b) $it = 10$. Note that images have different colorbars.

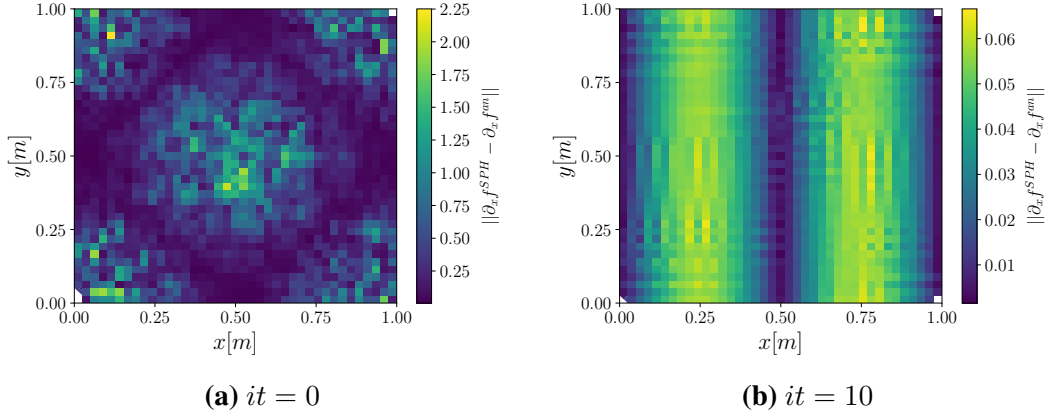


Figure 8.3 Static test case. $\Delta/L = 0.25$ and initial perturbation $\sigma/\Delta = 0.10$. $\|\partial_x f^{SPH} - \partial_x f^{an}\|$ at (a) $it = 0$ and (b) $it = 10$. Note that images have different colorbars.

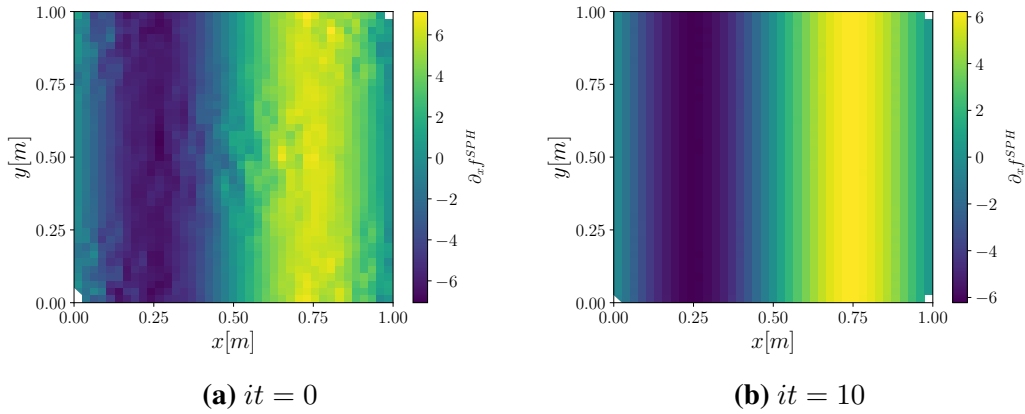
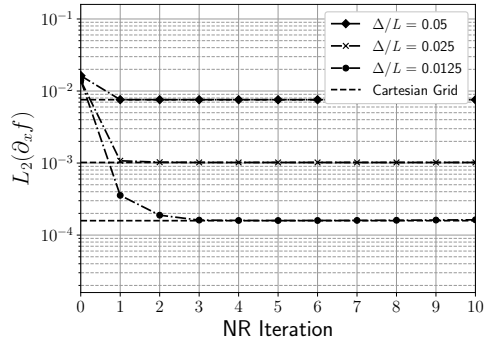
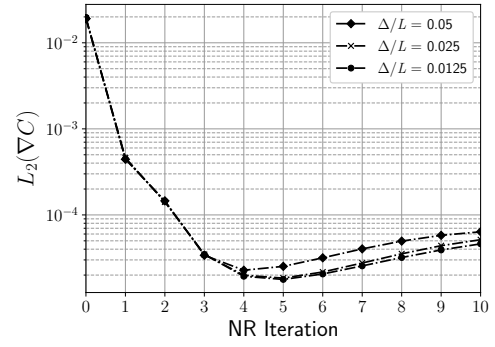


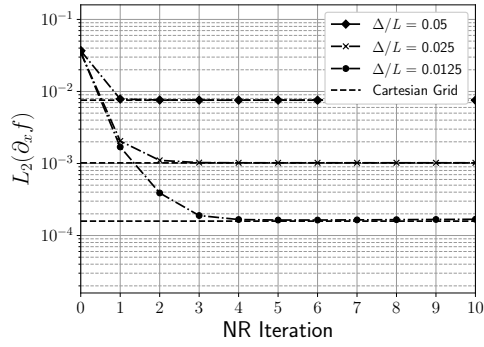
Figure 8.4 Static test case. $\Delta/L = 0.025$ and initial perturbation $\sigma/\Delta = 0.10$. $\partial_x f^{SPH}$ at (a) $it = 0$ and (b) $it = 10$. Note that images have different colorbars.



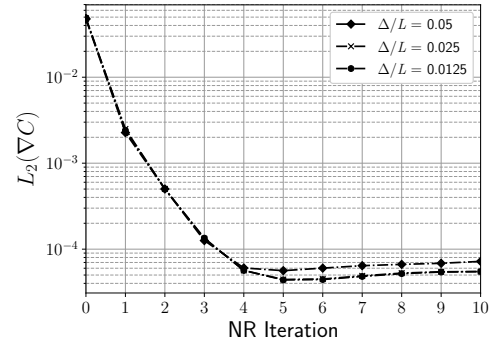
(a)



(b)



(c)



(d)

Figure 8.5 Static test case. Initial perturbation $\sigma/\Delta = 0.10$ (top) and $\sigma/\Delta = 0.25$ (bottom).

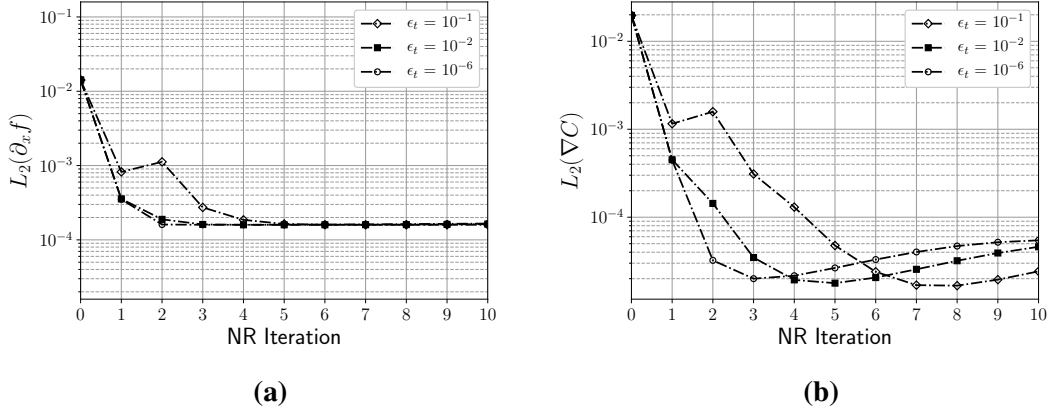


Figure 8.6 Static test case. $\Delta/L = 0.0125$ and initial perturbation $\sigma/\Delta = 0.10$.

It is observed that the $L_2(\nabla C)$ is reduced by almost three orders of magnitude and that the minimum value is reached after approximately 5 Newton-Raphson iterations, regardless of the resolution adopted and the initial level of particle disorder. This demonstrates that the IIPS algorithm is robust and generates a uniform particle distribution even starting from a significant initial disorder in a few Newton-Raphson iterations.

In order to assess the quality of the particle distribution in terms of SPH accuracy, the $L_2(\partial_x f)$ norm is shown in Figure 8.5 (a) and (c). As a reference, in the same figure the same norms obtained adopting no initial disorder (which corresponds to an initial Cartesian particle distribution) are plotted with dashed lines, regardless of the initial values of σ/Δ , an accuracy similar to the Cartesian one is achieved with less than 4 iterations, demonstrating that the IIPS algorithm is able to remove the effects of particle anisotropy from the SPH operators.

These initial tests, described above, have been conducted using a tolerance ϵ_t equal to 10^{-6} for the linear system iterative solver. However, since its value clearly affects the efficiency, additional analysis have been carried out to evaluate how its value affect the overall accuracy; therefore $\epsilon_t = 0.1$, 0.01 , 10^{-6} have been tested, for $\Delta/L = 0.0125$ and $\sigma/\Delta = 0.10$.

Figure 8.6 shows the norms $L_2(\nabla C)$ and $L_2(\partial_x f)$ against the number of Newton-Raphson iterations, while using $\epsilon_t = 0.01$, $\epsilon_t = 10^{-6}$ results are similar, adopting $\epsilon_t = 0.1$ both norms are not monotonically decreasing, suggesting that this value is not adequate to obtain a robust convergence of the IIPS algorithm.

The number of linear solver iterations and the cumulative computational time at each Newton-Raphson iteration are shown in Figure 8.7. Choosing $\epsilon_t = 0.01$ is a good compromise between computational cost and accuracy, this value will be used in the following numerical tests.

A convergence analysis, for the static test case, has been performed, following the procedure reported in the pseudo-code (Algorithm 1), with $it_{Max} = 10$. In these eval-

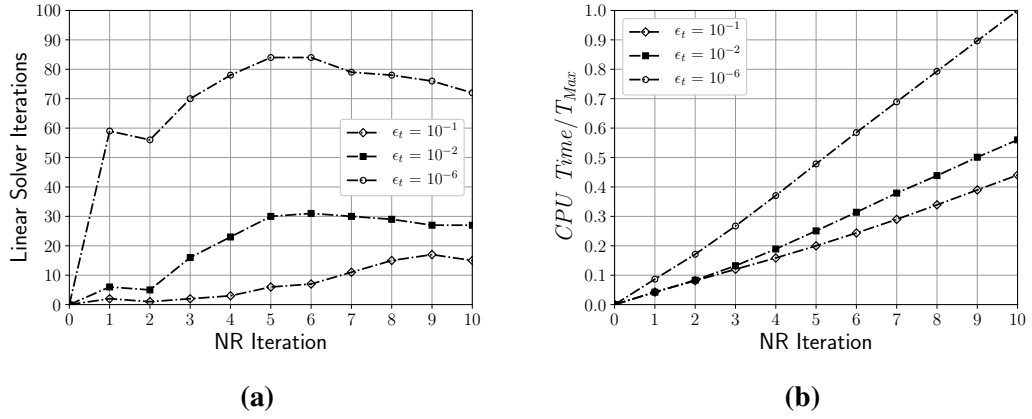


Figure 8.7 Static test case. (a) iteration needed to reach the ϵ value by the linear system solver, (b) normalized CPU time, scaled on $\epsilon = 10^{-6}$, using tolerance (normalized residual) value in the linear system solver.

	$\frac{h}{\Delta} = 2.0$		$\frac{h}{\Delta} = 1.6$		$\frac{h}{\Delta} = 1.3$	
Δ/L	$L_2(\partial_x f)$	θ	$L_2(\partial_x f)$	θ	$L_2(\partial_x f)$	θ
0.05	0.007532		0.005754		0.008354	
0.025	0.001020	2.89	0.001327	2.12	0.004109	1.19
0.0125	0.000159	2.68	0.000452	1.55	0.001919	1.10
0.01	0.000094	2.35	0.000352	1.11	0.001557	0.92
0.00625	0.000038	1.93	0.000231	0.90	0.001022	0.89
0.003125	0.000018	1.11	0.000149	0.64	0.000628	0.70
0.0015625	0.000013	0.43	0.000127	0.22	0.000484	0.38

Table 8.1 Static test case. Convergence analysis results, $L_2(\partial_x f)$ and convergence ratio θ .

uations, as previously illustrated, an initial particle distribution with $\sigma/\Delta = 0.10$ is considered, then, at each Newton-Raphson iteration, the implicit procedure is called, particles are shifted exclusively using the implicit displacements obtained through the formulations expressed in equation (8.32) and the $L_2(\partial_x f)$ is computed. In Figure 8.8 the values of $L_2(\partial_x f)$ obtained at the end of the NR iterations are shown using different kernel size and in Table 8.1 these values are reported together with the order of convergence θ .

The SPH spatial interpolation has two main different sources of error, the smoothing and the discretization error, generated respectively by the approximation of equation (4.3) and by the quadrature equation (4.31). For each h/Δ adopted in the convergence analysis, the $L_2(\partial_x f)$ reduces when the smoothing error is much larger than the discretization error, (Quinlan et al. 2006). Smaller h/Δ corresponds to larger discretization error, which therefore becomes dominant for a larger Δ causing saturation for the

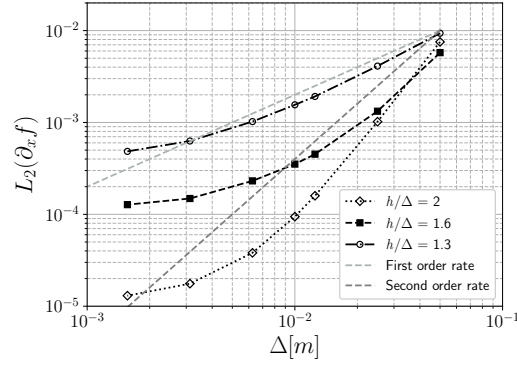


Figure 8.8 Static test case. Convergence analysis results, $L_2(\partial_x f)$ for different h/Δ values.

overall convergence.

Looking at the convergence rate θ in Table 8.1, for $h/\Delta = 2.0$, greater than theoretical order of convergence (equal to 2 for the adopted kernel) are obtained for $\Delta/L > 0.0125$ m, possibly this is due to the fact that the IIPS scheme reduces also the discretization error when the resolution is increased. The capability of the IIPS procedure to restore the SPH spatial convergence becomes apparent comparing Figure 8.8 to Figure 4.2 (b), obtained with Cartesian distributions, implying that the IIPS is able to completely remove the SPH discretization error.

Having established that the IIPS is able to minimize the discretization error, its performances against a different explicit shifting methodology are evaluated. Through the static test case, the implicit iterative method has been compared with a methodology that recalls the explicit iterative method proposed by Vacondio and Rogers (2017). At each explicit iteration the gradient of particles concentration, ∇C , is computed and particles are moved as described in equation (6.13), (adopting $D = 0.5$, as suggested in Lind et al. (2012)), with the aim to analyse the formulation effectiveness in term of the total number of iterations. The comparison between the implicit iterative and the explicit iterative methodologies has been conducted starting from the same perturbed particle distribution, with $\sigma/\Delta = 0.10$, using $h/\Delta = 2.0$ and $\Delta/L = 0.0125$, to assess the number of iterations and the computational time needed to reach a predefined level of the $L_2(\partial_x f)$, using different steepness of the test function, considering $\lambda/L = 1, 0.5, 0.25$ which correspond respectively to a 40, 20, 10 particles in wavelength.

Figure 8.9 shows the $L_2(\partial_x f)$ against the number of iterations needed to reach the Cartesian grid accuracy (with a 5% tolerance), for the implicit iterative (Figure 8.9 (a)) and explicit iterative (Figure 8.9 (b)) shifting procedures, whereas Table 8.2 reports the computational time. While the implicit method reaches the Cartesian accuracy in maximum of 3 Newton-Raphson iterations, the explicit method requires 68, 380 and 2408 iterations to achieve similar accuracy.

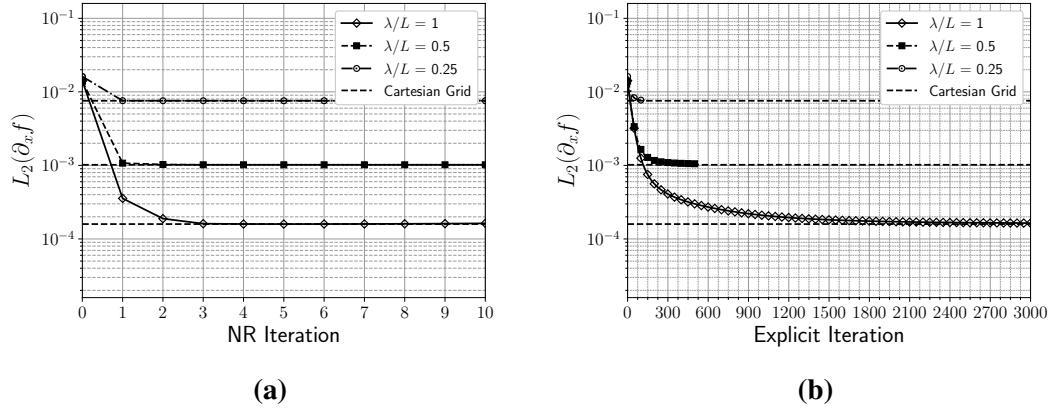


Figure 8.9 Static test case. $\Delta/L = 0.0125$ and initial perturbation $\sigma/\Delta = 0.10$. (a) implicit iterative particle shifting method, (b) explicit iterative shifting method.

	Implicit		Explicit	
	Iteration	Time [<i>ms</i>]	Iteration	Time [<i>ms</i>]
$\lambda/L = 0.25$	1	66	68	2619
$\lambda/L = 0.5$	2	126	380	13964
$\lambda/L = 1.0$	3	207	2408	98302

Table 8.2 Static test case. Particle Shifting Techniques comparison

Moreover, for the explicit algorithm, the total number of iterations strongly depends on λ . Therefore, despite the fact that each iteration of the IIPS has a larger computational cost (due to the fact that a linear system with $d \times n$ unknowns has to be solved), it is always more efficient, with speedups varying from 40 for $\lambda/L = 0.25$ to 475 for $\lambda/L = 1$.

The estimated extra memory required to the implicit procedure is proportional to $d \cdot n \cdot n_b$, the number of non-null elements on the matrix, where n_b is the number of neighbours for each particle.

8.3.2 Kinematic test case

The second validation test is a purely kinematic case, the aim is to test the capability of IIPS in maintaining low discretization error, while an external motion constantly perturbs the distribution. For this reason, this test is referred as "kinematic" because the perturbation in the particles distribution is introduced during the entire simulation. Specifically, particles move following the Lagrangian trajectories defined by the integration in time of the analytical TGV initial velocity field, at $t = 0$ s:

$$\begin{aligned}\Delta x_i &= -U \cos(2\pi x_i/L) \sin(2\pi y_i/L) \Delta t \\ \Delta y_i &= U \sin(2\pi x_i/L) \cos(2\pi y_i/L) \Delta t\end{aligned}\tag{8.33}$$

where Δx and Δy are the particles' displacements in the vortices on the x and y axis, respectively, U is equal to 1 m/s and L is equal to 1 m. The reference speed of sound $c_0 = 10U$ and the CFL = 0.2 have been set to compute an equivalent time step size, $\Delta t = 0.2(\Delta/c_0)$, to obtain the particle displacement. In order to assess which one guarantees the lower error in the SPH interpolation and optimizes the computational costs, two methodologies, named *Procedure A*, and *Procedure B* have been investigated and presented respectively in Algorithm 2 and Algorithm 3. The only difference among them is that in the *Procedure A* at least one Newton-Raphson iteration is activated at each time step, regardless of the other condition.

The initial particle distribution has been obtained by introducing an iterative pre-procedure (line 2 of Algorithm 2 and 3), which corresponds to run the static test case for 100 Newton-Raphson iterations. Using the pre-procedure the initial minimal value, $L_\infty(\nabla C)_{Init}$, is obtained and it is utilized to set the threshold used to trigger the IIPS, $L_\infty(\nabla C)_{Thr}$, (line 3 of Algorithm 2 and 3), which is set equal to:

$$L_\infty(\nabla C)_{Thr} = \beta L_\infty(\nabla C)_{Init}, \quad (8.34)$$

where β is an arbitrary coefficient. At each time step the particle distribution has to fulfil this condition, (line 5 of Algorithm 2 and 3):

$$L_\infty(\nabla C) \leq L_\infty(\nabla C)_{Thr}. \quad (8.35)$$

Therefore, the maximum value of $h|\nabla C_i|$ has to be always below the fixed threshold otherwise the IIPS procedure is activated. This condition is valid in both procedures, the only difference is the extra condition added in Algorithm 2, which requires to run at least one Newton-Raphson iteration at each time step even if the condition defined in equation (8.35) is satisfied.

The IIPS procedure runs for an unlimited number of Newton-Raphson iterations, until each particle has $h|\nabla C_i|$ that meet the condition.

In *Procedure A* and in *Procedure B* two different factors have been tested, $\beta=5$ and $\beta=10$, but because at least one Newton-Raphson iteration is required in *Procedure A*, $L_\infty(\nabla C)$ never reaches the imposed threshold and the results, as presented later, do not depend on β .

To illustrate this, results for $\Delta/L = 0.025$, $h/\Delta = 2.0$, and $\lambda/L = 0.5$ are presented, showing the difference between the procedures. Figure 8.10 and Figure 8.11 report the comparison between the different implicit iterative procedures and the explicit method. The explicit shifting technique is called in line 8 of Algorithm 4, at each

Algorithm 2 Kinematic case

```
1: procedure A IMPLICIT ITERATIVE ▷
2:   Static test case (Algorithm 1)
3:   Set  $L_\infty(\nabla C)_{Thr} = \beta L_\infty(\nabla C)_{Init}$ 
4:   Start Simulation
5:   while ( $L_\infty(\nabla C) \leq L_\infty(\nabla C)_{Thr}$  and  $NR_{it} \geq 1$ ) do
6:     AssembleMatrix( $\nabla C_i$ )
7:     LinearSystemSolver(AssembleMatrix)
8:     UpdateParticlePosition(Implicit PST)
9:     Compute  $L_\infty(\nabla C)$ ,  $L_2(\partial_x f)$ 
10:  end while
11:  ParticleDisplacement (Eq. (8.33))
12: end procedure
```

Algorithm 3 Kinematic case

```
1: procedure B IMPLICIT ITERATIVE ▷
2:   Static test case (Algorithm 1)
3:   Set  $L_\infty(\nabla C)_{Thr} = \beta L_\infty(\nabla C)_{Init}$ 
4:   Start Simulation
5:   while ( $L_\infty(\nabla C) \leq L_\infty(\nabla C)_{Thr}$ ) do
6:     AssembleMatrix( $\nabla C_i$ )
7:     LinearSystemSolver(AssembleMatrix)
8:     UpdateParticlePosition(Implicit PST)
9:     Compute  $L_\infty(\nabla C)$ ,  $L_2(\partial_x f)$ 
10:  end while
11:  ParticleDisplacement (Eq. (8.33))
12: end procedure
```

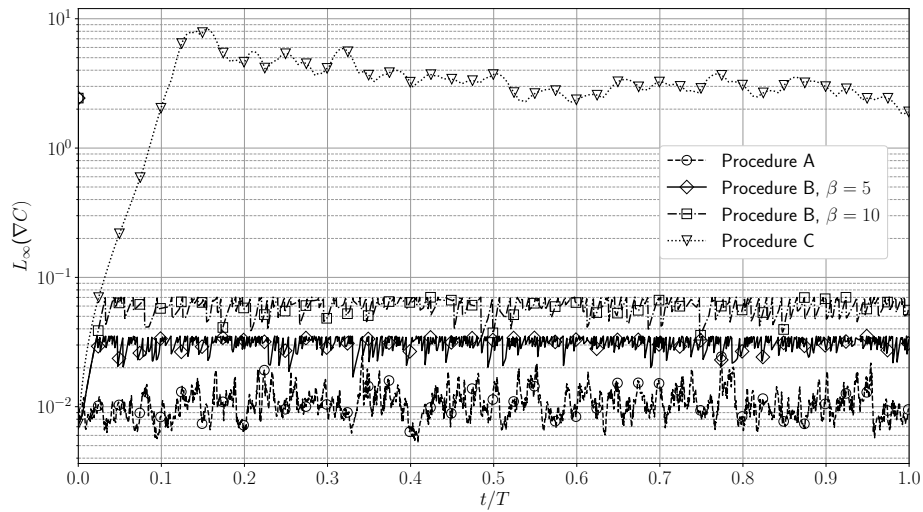
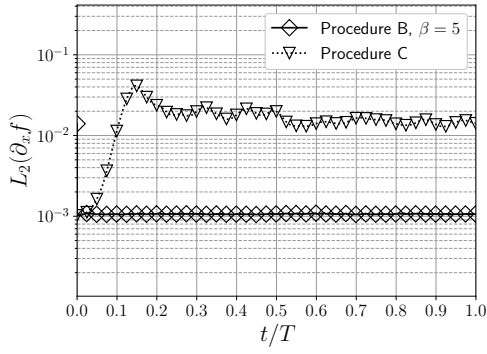
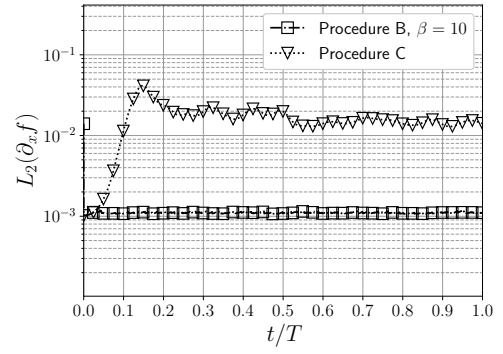


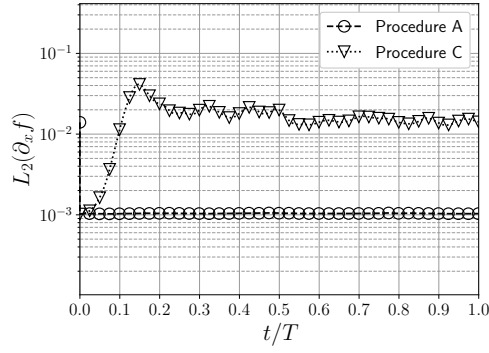
Figure 8.10 Kinematic test case. Procedure comparison, $\Delta/L = 0.025$ and $h/\Delta = 2.0$.



(a)



(b)



(c)

Figure 8.11 Kinematic test case. Procedure comparison, $\Delta/L = 0.025$ and $h/\Delta = 2.0$.

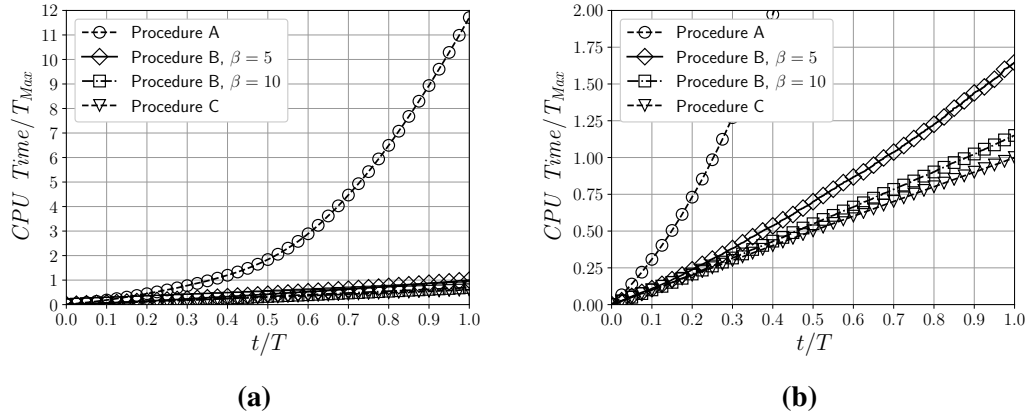


Figure 8.12 Kinematic test case. Method comparison, CPU time. Note different scales.

time step the particle position is updated using equation (6.1) and equation (6.13),

$$\delta \mathbf{x}_i = \Delta t^2 c_0^2 \sum_{j=1}^J \nabla W_{ij} \omega_j. \quad (8.36)$$

Algorithm 4 Kinematic case.

- 1: **procedure** C EXPLICIT ▷
 - 2: Static test case (Algorithm 1)
 - 3: Start Simulation
 - 4: UpdateParticlePosition(Explicit PST)
 - 5: Compute $L_\infty(\nabla C)$, $L_2(\partial_x f)$
 - 6: ParticleDisplacement (Eq. (8.33))
 - 7: **end procedure**
-

The implicit procedures show differences evaluating $L_\infty(\nabla C)$, Figure 8.10, and similarities analysing $L_2(\partial_x f)$, Figure 8.11. In all the implicit procedures, a gain of more than an order of magnitude in the SPH accuracy is obtained and measuring ∇C the particle distribution quality is improved by almost two order of magnitude.

The computational costs for Algorithm 2, Algorithm 3 and Algorithm 4) are presented in Figure 8.12, the results are re-scaled with explicit method computational time.

As seen in Figure 8.9 and reported in Table 8.2, the explicit iterative procedure is not always sustainable in term of computational efforts and for this reason the implicit iterative procedures have been compared with a non-iterative explicit method.

Procedure A calls the IIPS at each time step, significantly increasing the computational costs, the non linear trend is due to the higher number of iterations that the linear solver requires to converge, explained by the fact that the IIPS procedure is activated even if the $L_\infty(\nabla C)$ is already close to the minimal value, therefore it needs a larger number of iterations to reduce the normalized residuals. The *Procedure B* calls the IIPS

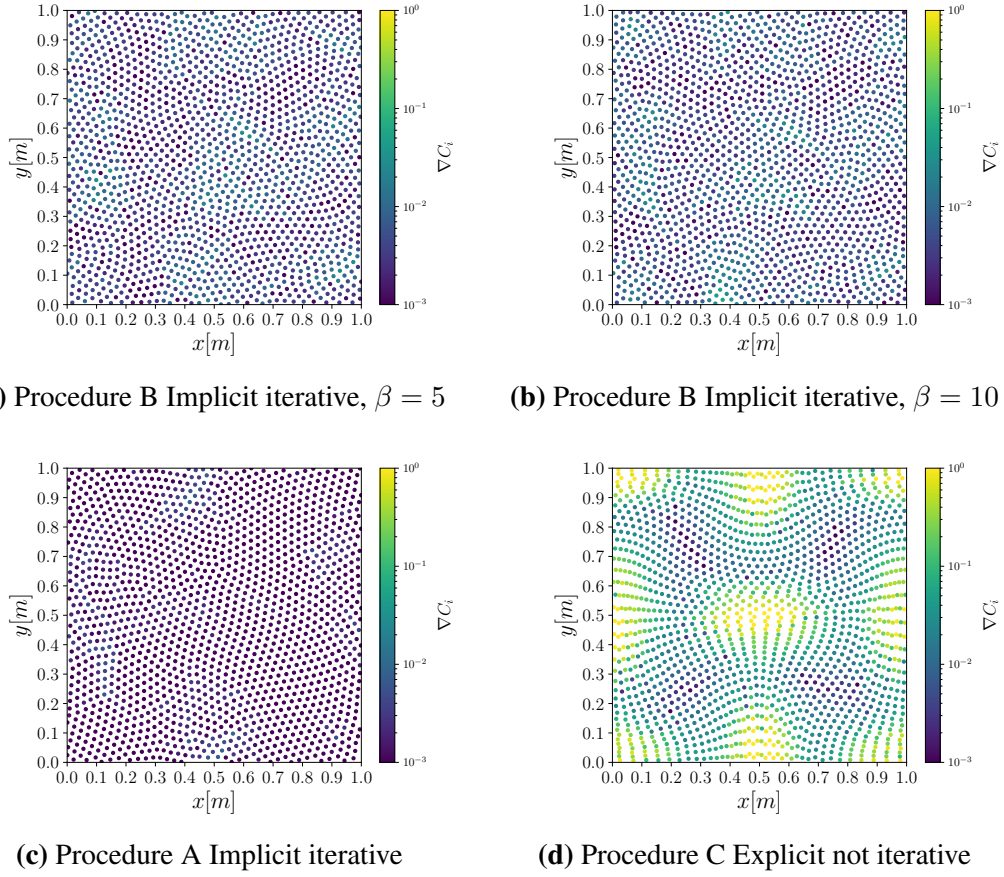
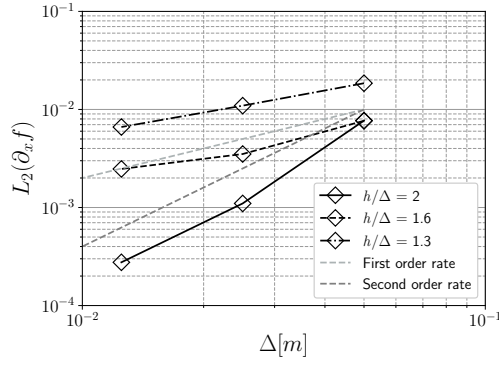


Figure 8.13 Kinematic test case. Particle distribution at $t = 0.2$ s.

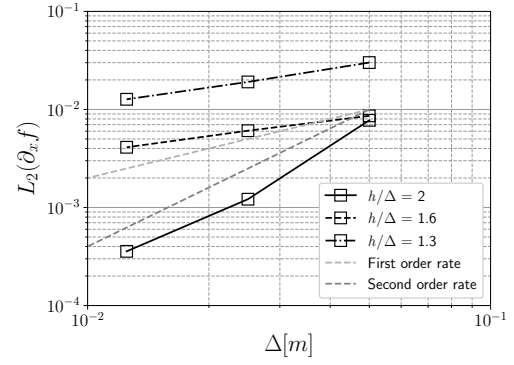
whenever the $L_\infty(\nabla C)$ value rises above the set threshold and it generates an extra computational cost between 10 to 70 percent based on β ; the *CPU* times are shown in Figure 8.12.

Looking at the particle distributions in Figure 8.13, it can be immediately noted that the implicit procedures maintain the particle uniformly distributed avoiding the creation of distinct streamlines in the domain.

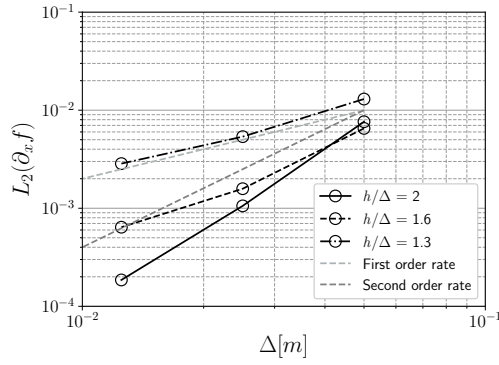
Results presented in Figure 8.14 shows the convergence analysis for all the different procedures, using $\Delta/L = 0.05, 0.025, 0.0125$ and $h/\Delta = 1.3, 1.6, 2.0$. The *Procedure A* shows same results regardless β , as it was already seen for just a single resolution and kernel size in Figure 8.11. The convergence rates for results presented in Figure 8.14, are reported in Table 8.3; these are close to the ones obtained in the static test case (Table 8.1), meaning that even with a continuous sources of perturbation the IIPS is able to prevent anisotropy in particle distribution, differently to the explicit procedure which convergence rate is lower.



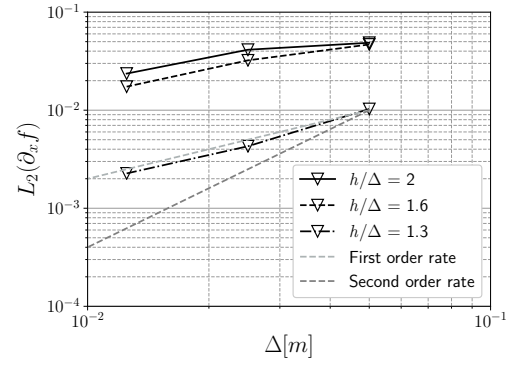
(a) Procedure B Implicit iterative, $\beta = 5$



(b) Procedure B Implicit iterative, $\beta = 10$



(c) Procedure A Implicit iterative



(d) Procedure C Explicit not iterative

Figure 8.14 Kinematic test case. Convergence analysis.

	Implicit iterative all		Implicit iterative $\beta=5$		Implicit iterative $\beta=10$		Explicit not iterative	
	θ_1	θ_2	θ_1	θ_2	θ_1	θ_2	θ_1	θ_2
$\frac{h}{\Delta} = 2.0$	2.85	2.51	2.81	1.99	2.67	1.76	0.23	0.81
$\frac{h}{\Delta} = 1.6$	2.04	1.30	1.12	0.51	0.51	0.56	0.53	0.89
$\frac{h}{\Delta} = 1.3$	1.27	0.91	0.76	0.72	0.65	0.58	1.26	0.92

Table 8.3 Kinematic test case. Convergence ratio results.

8.4 Summary and conclusions

A novel particle shifting technique has been presented in this chapter, the implicit iterative methodology is purely based on geometric evaluations for the particle distribution. The algorithm has the capability to directly tackle the formation of areas in the domain with non-uniform particle concentration using a minimization procedure. It has been evaluated the efficiency in restoring the accuracy and of the SPH interpolation but it is applicable in different meshless numerical scheme method based on kernel basis functions. Specific applications to a fluid dynamic SPH solver are illustrated in the following chapter.

Chapter 9

Implicit iterative particle shifting in ALE-SPH schemes

In Chapter 8 has been proposed a new formulation called Implicit Iterative Particle Shifting (IIPS) to improve the particle position in meshless schemes based on kernel spatial interpolators. It is relevant to point out that the IIPS can be embedded and applied for meshless methods in which isotropy in computational points is required to achieve a high level of accuracy; in the following, specific additional details on the methodologies adopted to embed the IIPS method in an SPH-ALE solver, presented in Section 5.5, are clarified and numerical validations are shown. In particular, two different approaches are proposed to update the physical quantities at the new particles' positions after the IIPS is completed.

9.1 IIPS in ALE-SPH schemes

During the physical simulations, at the end of each time step, the particle positions and the attached physical quantities are computed using equations (5.48), the particles move following a pure Lagrangian motion, in which $\mathbf{v}_0 = \mathbf{v}$. Based on the defined condition on the level of particle distribution disorder, the IIPS procedure is activated, then those positions are further updated using the formulation described in Section 8.3. Therefore, there are two different sets of particles: the initial one, computed at the end of the physical time step, and the updated one, obtained through the IIPS procedure.

As an initial attempt, the IIPS has been directly implemented in the SPH-ALE solver without any modification to the physical quantities, after the particles' positions are corrected. This is a basic approach that introduces inconsistency, nevertheless, it is an option to speed up the simulations.

Update the physical quantities after the shifting procedure guarantees the scheme consistency and reduces numerical instabilities. The ALE-SPH model, proposed in Oger et al. (2016), coherently adjusts these quantities, while using a shifting technique; the scheme itself allows to arbitrarily define the transport velocity $\mathbf{v}_0$, which directly embed the explicit shifting correction. In this method, the arbitrary velocity field is the summation of two terms, equation (6.31), which are both smooth, in particular, the correction term, equation (6.32), is scaled by the fluid velocity and it is limited to

maintain the quasi-Lagrangian nature, therefore the transport velocity field results to be smooth too. Following this approach, introducing directly the particle displacements obtained with the implicit procedure in the transport velocity formulation is not reasonable because it is not guaranteed the smoothness of implicit shifting correction, on the contrary, in numerical experiments as the ones presented in Section 8.3, it has been noted that even adjacent particles can be displaced in opposite directions. The shifting generated by IIPS is based on pure geometric evaluations, for this reason, it is aimed to maintain separations between the correction to particle distribution and the flow characteristics, i.e. fluid velocity. Following these considerations, different methodologies, to update the physical quantities, are presented.

9.1.1 IIPS with a fictitious time step

An innovative approach is adopted, following the explicit nature of the ALE-SPH scheme, the displacement obtained from the IIPS is used to compute the transport velocity in a fictitious time step. The equation (6.1), is manipulated as,

$$\bar{\mathbf{x}} = \mathbf{x} + \delta\mathbf{x} \quad \rightarrow \quad \delta\mathbf{x} = \bar{\mathbf{x}} - \mathbf{x} \quad (9.1)$$

and, isolating the implicit shifting term, a fictitious time step, Δt^* , is applied,

$$\frac{\delta\mathbf{x}}{\Delta t^*} = \frac{\bar{\mathbf{x}} - \mathbf{x}}{\Delta t^*} = \mathbf{v}_0, \quad (9.2)$$

defining the transport velocity $\mathbf{v}_0$ for that fictitious time step.

Due to the ALE-SPH nature, the interpolation points displacements, obtained through the implicit iterative shifting procedure, generate convective fluxes that need to be taken into account. To compute the variation of the physical quantities, the above-mentioned defined $\mathbf{v}_0$ is introduced in equation (5.39), and, neglecting internal or external forces leads to the set of equations that need to be solved in the fictitious time step,

$$\left\{ \begin{array}{l} \frac{\delta\mathbf{x}_i}{\Delta t^*} = \mathbf{v}_{0i} \\ \frac{d\omega_i}{\Delta t^*} = \omega_i \sum_{j=1}^J (\mathbf{v}_{0j} - \mathbf{v}_{0i}) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i}{\Delta t^*} = -\omega_i \sum_{j=1}^J (\rho_i (\mathbf{v}_{0i}) + \rho_j (\mathbf{v}_{0j})) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i \mathbf{v}_i}{\Delta t^*} = -\omega_i \sum_{j=1}^J (\rho_i \mathbf{v}_i \otimes (\mathbf{v}_{0i}) + \rho_j \mathbf{v}_j \otimes (\mathbf{v}_{0j})) \nabla W_{ij} \omega_j \end{array} \right. \quad (9.3)$$

The equations in system 9.3 need to be solved exclusively after the IIPS procedure, it

has no relation to the physical time step. Then, substituting equation (9.2), the ALE-SPH system is rewritten,

$$\left\{ \begin{array}{l} \frac{\delta \mathbf{x}_i}{\Delta t^*} = \mathbf{v}_{0i} \\ \frac{d\omega_i}{\Delta t^*} = \omega_i \sum_{j=1}^J \left(\frac{\delta \mathbf{x}_j}{\Delta t^*} - \frac{\delta \mathbf{x}_i}{\Delta t^*} \right) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i}{\Delta t^*} = -\omega_i \sum_{j=1}^J \left(\rho_i \left(\frac{\delta \mathbf{x}_i}{\Delta t^*} \right) + \rho_j \left(\frac{\delta \mathbf{x}_j}{\Delta t^*} \right) \right) \nabla W_{ij} \omega_j \\ \frac{d\omega_i \rho_i \mathbf{v}_i}{\Delta t^*} = -\omega_i \sum_{j=1}^J \left(\rho_i \mathbf{v}_i \otimes \left(\frac{\delta \mathbf{x}_i}{\Delta t^*} \right) + \rho_j \mathbf{v}_j \otimes \left(\frac{\delta \mathbf{x}_j}{\Delta t^*} \right) \right) \nabla W_{ij} \omega_j \end{array} \right. \quad (9.4)$$

The time step Δt^* , as expected, has non influence and it is canceled out,

$$\left\{ \begin{array}{l} \bar{\mathbf{x}}_i - \mathbf{x}_i = \delta \mathbf{x}_i \\ \bar{\omega}_i - \omega_i = \omega_i \sum_{j=1}^J (\delta \mathbf{x}_j - \delta \mathbf{x}_i) \nabla W_{ij} \omega_j \\ \bar{\omega}_i \bar{\rho}_i - \omega_i \rho_i = -\omega_i \sum_{j=1}^J (\rho_i (\delta \mathbf{x}_i) + \rho_j (\delta \mathbf{x}_j)) \nabla W_{ij} \omega_j \\ \bar{\omega}_i \bar{\rho}_i \bar{\mathbf{v}}_i - \omega_i \rho_i \mathbf{v}_i = -\omega_i \sum_{j=1}^J (\rho_i \mathbf{v}_i \otimes (\delta \mathbf{x}_i) + \rho_j \mathbf{v}_j \otimes (\delta \mathbf{x}_j)) \nabla W_{ij} \omega_j \end{array} \right. \quad (9.5)$$

In this system, $\bar{\omega}_i$, $\bar{\rho}_i$, and $\bar{\mathbf{v}}_i$ are the updated quantities at the end of the fictitious time step. A preliminary consideration regarding the maximum implicit displacement has to be pointed out; the IIPS methodology has no numerical constrains on the maximum allowed shifting, even if the conducted numerical experiences have shown that for low perturbed distributions the values are usually less than 0.2Δ . Nevertheless, with the respect to the physics, it has been proposed to indirectly bound the displacements, reducing the fictitious time step with a factor related to the CFL condition. It results that the field quantities are updated with implicit displacements that do not exceed the maximum allowed physical displacement obtained in the time integration. A numerical coefficient, N , is computed at the end of each implicit procedure as the nearest greater integer number of the ratio between the maximum shifting displacement and the particle size, inversely scale by CFL,

$$N = \left\lceil \frac{(\max_i |\delta \mathbf{x}_i|)}{CFL \Delta} \right\rceil. \quad (9.6)$$

where the symbol $\lceil \cdot \rceil$ stands for the ceiling function. It is used to reduce the shifting

displacement, therefore, the final version of equations (9.3) is rewritten as,

$$\begin{cases} \bar{\mathbf{x}}_i - \mathbf{x}_i = \frac{\delta \mathbf{x}_i}{N} \\ \bar{\omega}_i - \omega_i = \omega_i \sum_{j=1}^J \left(\frac{\delta \mathbf{x}_j}{N} - \frac{\delta \mathbf{x}_i}{N} \right) \nabla W_{ij} \omega_j \\ \bar{\omega}_i \bar{\rho}_i - \omega_i \rho_i = -\omega_i \sum_{j=1}^J \left(\rho_i \left(\frac{\delta \mathbf{x}_i}{N} \right) + \rho_j \left(\frac{\delta \mathbf{x}_j}{N} \right) \right) \nabla W_{ij} \omega_j \\ \bar{\omega}_i \bar{\rho}_i \bar{\mathbf{v}}_i - \omega_i \rho_i \mathbf{v}_i = -\omega_i \sum_{j=1}^J \left(\rho_i \mathbf{v}_i \otimes \left(\frac{\delta \mathbf{x}_i}{N} \right) + \rho_j \mathbf{v}_j \otimes \left(\frac{\delta \mathbf{x}_j}{N} \right) \right) \nabla W_{ij} \omega_j \end{cases} \quad (9.7)$$

These equations are integrated for N times to obtain the final value of the field quantities in the position obtained using the implicit iterative algorithm.

The set of equations proposed is consistent with the ALE-SPH formalism and it conserves mass and momentum for particles in the interior region of fluid.

9.1.2 IIPS with MLS reconstruction

A second approach is proposed to update the physical quantities, $\bar{\omega}_i$, $\bar{\rho}_i$, and $\bar{\mathbf{v}}_i$, at the final particle position, obtained using the IIPS procedure, by means of spatial interpolation, conducted using the information available at the particle positions computed at the end of the physical time step. This approach has been utilized in explicit iterative particle shifting procedure, (Vacondio and Rogers 2017), adjusting the quantities by using the corrected SPH interpolation proposed in (Liu and Liu 2006). The spatial interpolation procedure needs to be chosen in such a way that it minimizes the unavoidable numerical diffusivity of the reinterpolation. The first-order and second-order Moving Least Square (MLS) reconstruction, presented in Renaut (2015), have been evaluated against the test function, equation (3.6). For each particle in the fluid domain to reconstruct field quantities a linear system is assembled and solved.

The linear reconstruction system is,

$$\underbrace{\sum_{j=1}^J W_{ij} \omega_j \begin{pmatrix} 1 & x_{ij} & z_{ij} \\ - & x_{ij}^2 & x_{ij} z_{ij} \\ - & - & z_{ij}^2 \end{pmatrix}}_A \cdot \underbrace{\begin{pmatrix} f_i \\ \partial_x f_i \\ \partial_z f_i \end{pmatrix}}_X = \underbrace{\sum_{j=1}^J \phi_j W_{ij} \omega_j \begin{pmatrix} 1 \\ x_{ij} \\ z_{ij} \end{pmatrix}}_B \quad (9.8)$$

and quadratic reconstructions system is,

$$\underbrace{\sum_{j=1}^J W_{ij} \omega_j \begin{pmatrix} 1 & x_{ij} & z_{ij} & x_{ij}^2 & x_{ij} z_{ij} & z_{ij}^2 \\ - & x_{ij}^2 & x_{ij} z_{ij} & x_{ij}^3 & x_{ij}^2 z_{ij} & x_{ij} z_{ij}^2 \\ - & - & z_{ij}^2 & x_{ij}^2 z_{ij} & x_{ij} z_{ij}^2 & z_{ij}^3 \\ - & - & - & x_{ij}^4 & x_{ij}^3 z_{ij} & x_{ij}^2 z_{ij}^2 \\ - & - & - & - & x_{ij}^2 z_{ij}^2 & x_{ij} z_{ij}^3 \\ - & - & - & - & - & z_{ij}^4 \end{pmatrix}}_A \cdot \underbrace{\begin{pmatrix} f_i \\ \partial_x f_i \\ \partial_z f_i \\ \partial_x^2 f_i \\ \partial_{xz} f_i \\ \partial_z^2 f_i \end{pmatrix}}_X = \underbrace{\sum_{j=1}^J \phi_j W_{ij} \omega_j \begin{pmatrix} 1 \\ x_{ij} \\ z_{ij} \\ x_{ij}^2 \\ x_{ij} z_{ij} \\ z_{ij}^2 \end{pmatrix}}_B \quad (9.9)$$

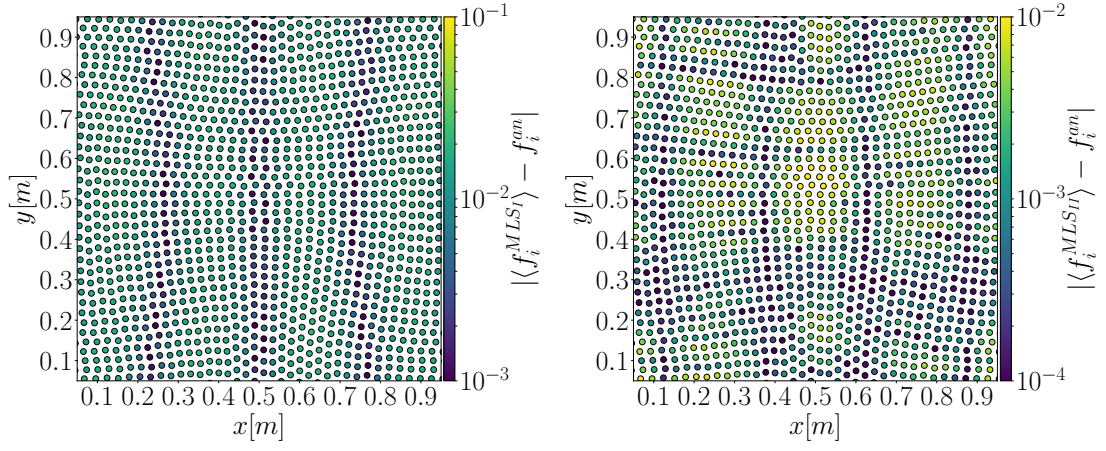
where $x_{ij} = \frac{x_i - x_j}{h_i}$ and $z_{ij} = \frac{z_i - z_j}{h_i}$, and i is the particle in the updated configuration and j are the particle inside the kernel support of i in the initial configuration.

Numerical tests have been performed using the test function defined in equation (3.6); starting from the same particle distribution, with the same amount of disorder, the implicit procedure has been applied and the new configuration has been obtained. Then, the first and the second-order MLS reconstructions have been computed, to interpolate the function f on the updated particle distribution, through the values of the original particle position. The errors fields are reported in Figure 9.1, for the $\Delta/L = 0.025$; as expected the second-order MLS reconstruction increases the level of accuracy; in Figure 9.2, the L_2 norm for different resolutions is shown, confirming the results. For this reason, it has been chosen the second-order MLS to reconstruct the physical quantities.

9.2 Evaluation of the IIPS in ALE-SPH

The Taylor-Green vortex test case, described in Section 3.1, has been used to analyze the methodologies, previously introduced; in this section the notation "Implicit iterative shifting" stands for no correction of physical quantities (particle positions are moved using the IIPS procedure, but the physical quantities are not updated), "Implicit iterative shifting NFTS" stands for equations (9.7) and "Implicit iterative shifting MLS II" stands for equation (9.9), while "Explicit shifting" is referred to the methodology described in Neuhauser (2014).

The IIPS methods has been implemented in a numerical SPH-ALE solver, (equa-



(a) MLS I^o order reconstruction.

(b) MLS II^o order reconstruction.

Figure 9.1 MLS reconstructions, error on a test function. $\Delta/L=0.025$

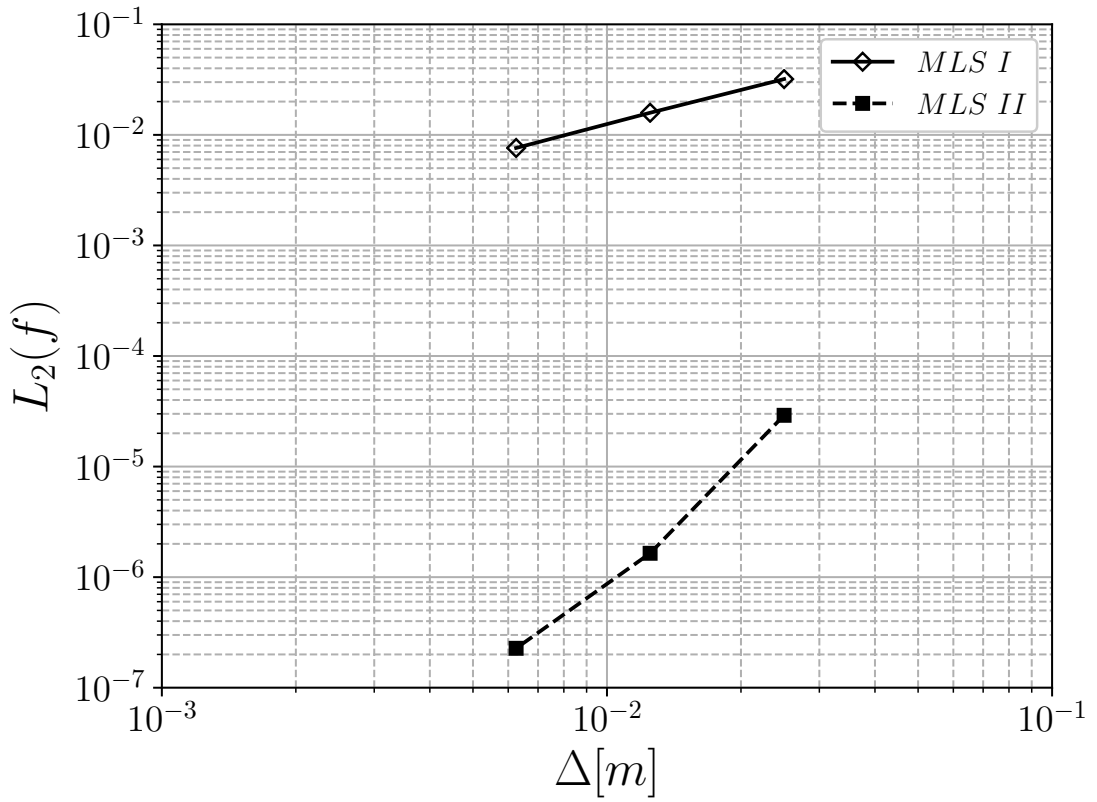


Figure 9.2 MLS reconstruction, error on a test function.

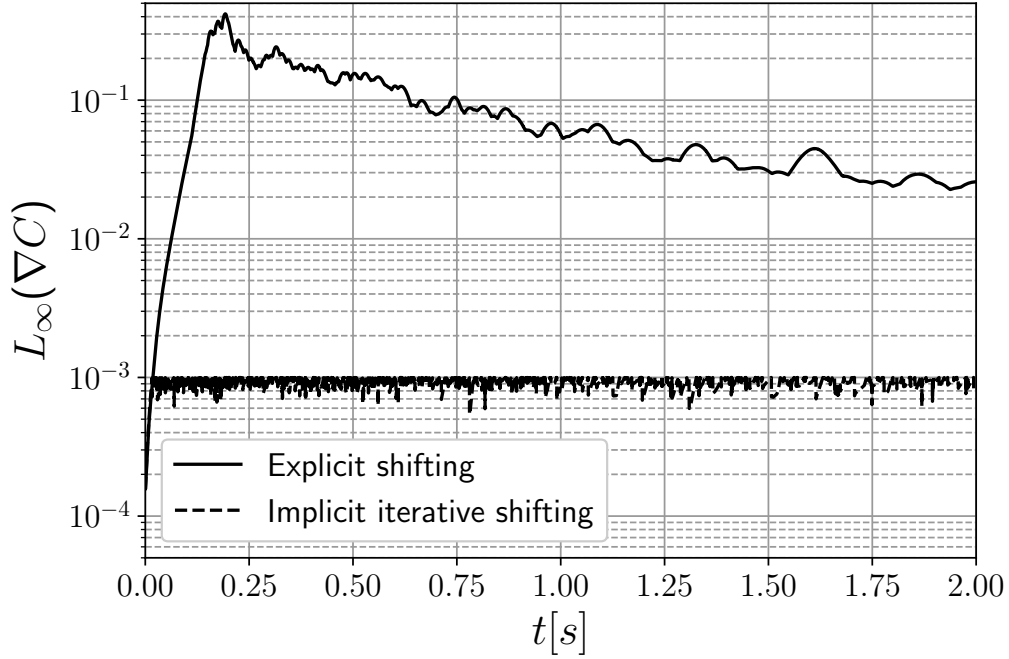


Figure 9.3 TGV test case. $\Delta/L = 0.00625$, (a) Neuhauser (2014) formulation, (b) IIPS with $L_\infty(\nabla C)_{thr} = 0.001/h$.

tions (5.39), following the *Proceduce B* reported in Algorithm 3, (Section 8.3.2), with $L_\infty(\nabla C)_{thr} = 0.001/h$. Initially, the quality of the particle distribution has been evaluated through the L_∞ norm of the particles concentration gradient. The implicit iterative shifting has been compared with the explicit shifting introduced in equation (6.33). The methodologies illustrated in Section 9.1.1 and in Section 9.1.2 aim is to increase the accuracy for the field variables; they do not improve the quality of particle distribution obtained using the IIPS, the L_∞ trends are coincident and, for this reason, not reported in Figure 9.3.

To evaluate the effectiveness of these methodologies the convergence analysis has been done using $\Delta/L = 0.05, 0.025, 0.0125, 0.00625$, and h/Δ equal to 2, showing the results in Figure 9.4, while the numerical values for the convergence rates θ and the norms of the velocity errors are reported in Table 9.1. These results are compared with the original explicit shifting formulation, (equation (6.33)). The explicit methodology is able to guarantee a first-order of convergence only, while, as already experienced in the kinematic test cases presented in Section 8.3, the IIPS technique increases the convergence rates to values higher than 2 reducing the discretization error. Additionally, updating the physical quantities with the proposed methodologies enhances the convergence rates while the resolution increases.

The velocity field and the error on the velocity field are shown in Figure 9.5, 9.6, 9.7 and 9.8, while the analysis of the error on the total kinetic energy is reported in Figure

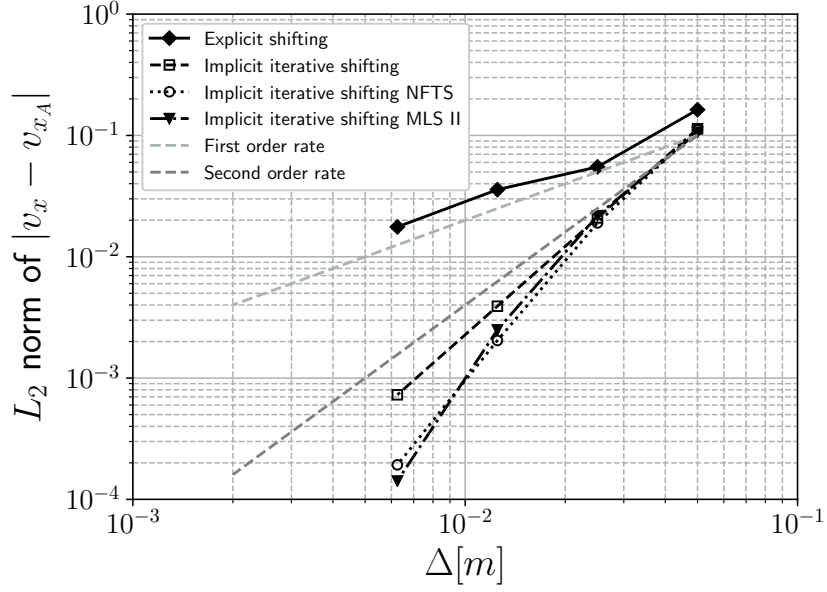


Figure 9.4 TGV test case. Convergence analysis $L_2|\mathbf{v}_x - \mathbf{v}_{x_A}|$, implicit particle shifting comparison at $t = 1$ s.

Δ/L	Neuhauser (2014)		IIPS		IIPS - NFTS		IIPS - MLS II	
	$L_2(\mathbf{v}_x)$	θ	$L_2(\mathbf{v}_x)$	θ	$L_2(\mathbf{v}_x)$	θ	$L_2(\mathbf{v}_x)$	θ
0.05	0.1628		0.1139		0.1111		0.1062	
0.025	0.0556	1.55	0.0208	2.44	0.0189	2.55	0.0218	2.28
0.0125	0.0337	0.72	0.0039	2.42	0.0020	3.22	0.0025	3.12
0.00625	0.0176	0.93	0.0007	2.41	0.00019	3.41	0.00014	4.13

Table 9.1 TGV test case. Convergence analysis results, $L_2|\mathbf{v}_x - \mathbf{v}_{x_A}|$ and convergence ratio θ , implicit iterative shifting.

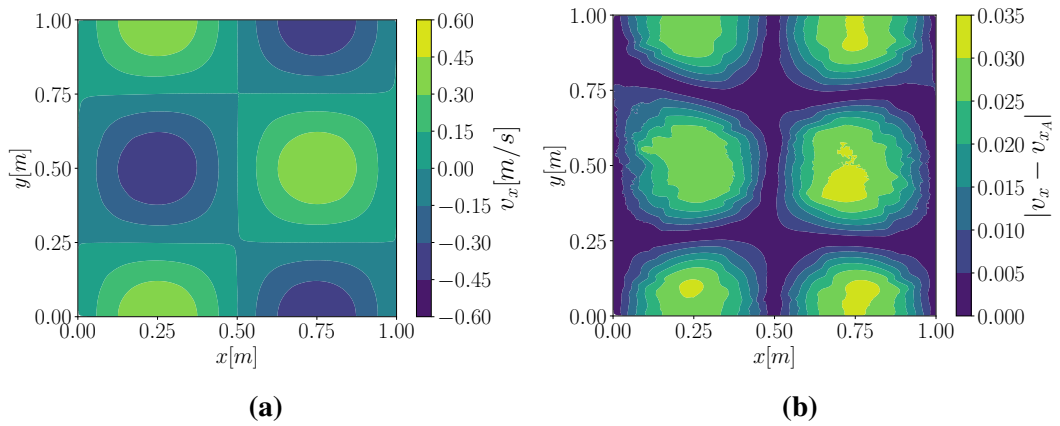


Figure 9.5 TGV test case. $\Delta/L = 0.00625$, Neuhauser (2014) formulation (a) velocity x component in [m/s], (b) error on the velocity x component at physical time $t = 1$ s.

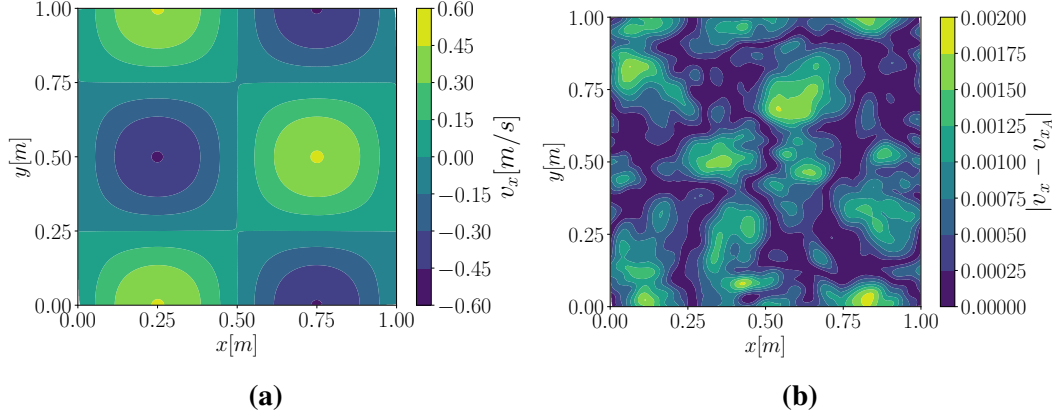


Figure 9.6 TGV test case. $\Delta/L = 0.00625$, implicit iterative shifting (a) velocity x component in [m/s], (b) error on the velocity x component at physical time $t = 1$ s.

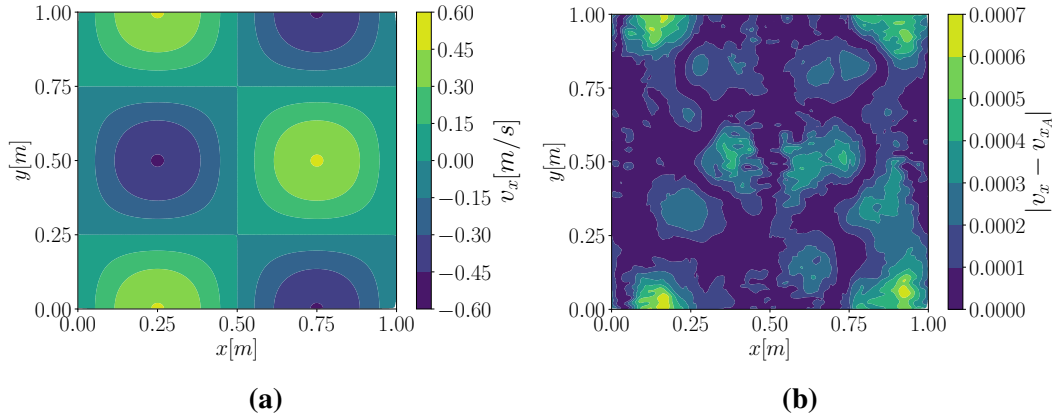


Figure 9.7 TGV test case. $\Delta/L = 0.00625$, implicit iterative shifting with NFTS (a) velocity x component in [m/s], (b) error on the velocity x component at physical time $t = 1$ s.

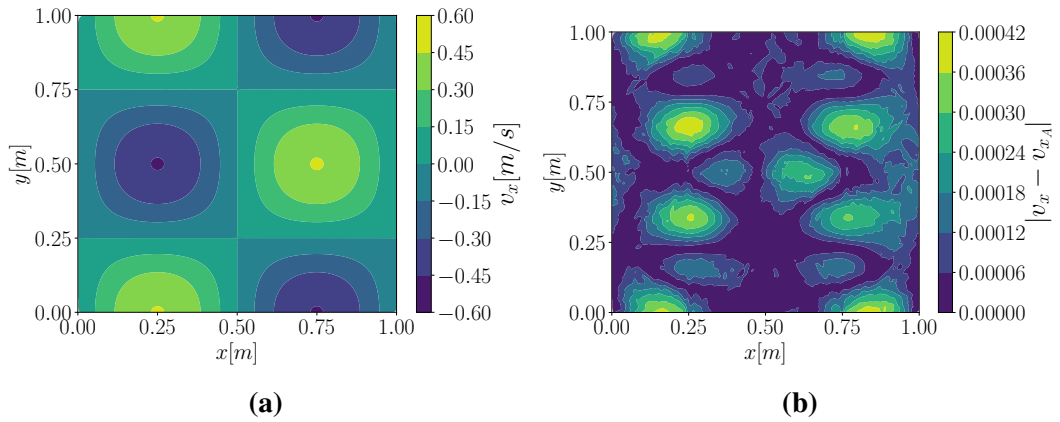


Figure 9.8 TGV test case. $\Delta/L = 0.00625$, implicit iterative shifting with MLS II° (a) velocity x component in [m/s], (b) error on the velocity x component at physical time $t = 1$ s.

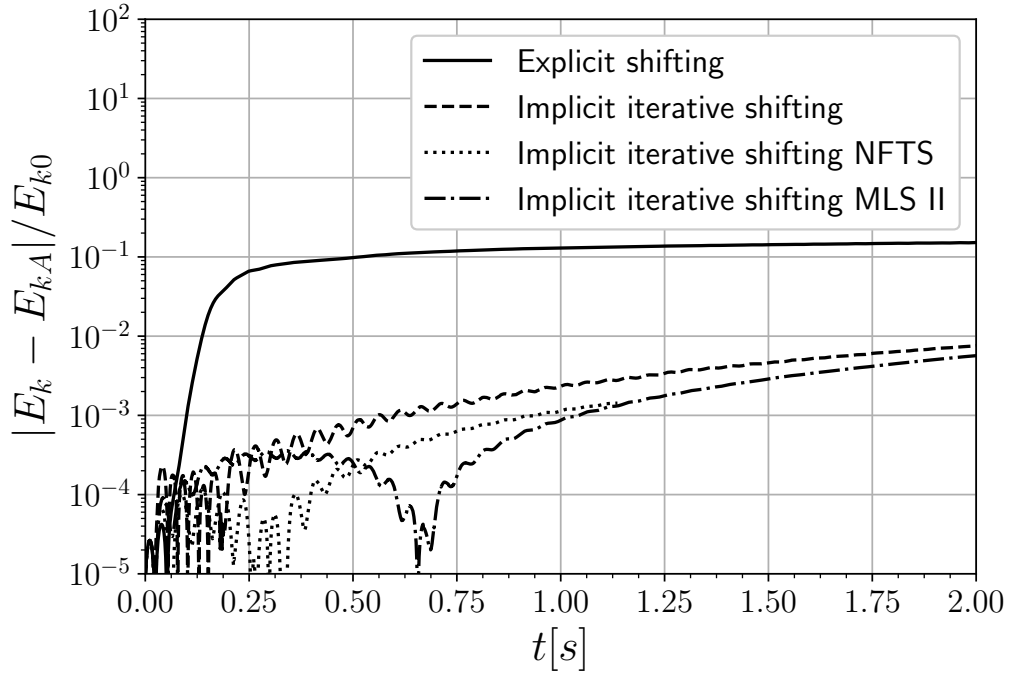


Figure 9.9 TGV test case. Kinetic energy error.

9.9; those results are referred to the higher resolution, $\Delta/L = 0.00626$. Comparing Figure 9.6 (b) with Figure 9.7 (b) and Figure 9.8 (b) confirms that the velocity field is more accurate and less noisy, this trends is verified during the entire simulation, where the error on the kinematic energy is reduced by more than one order of magnitude.

9.3 Summary and conclusions

In this chapter two methodologies have been proposed to update the physical quantities after the implicit iterative procedure in an SPH-ALE scheme for fluid dynamics. Firstly, it has been proposed to introduce a fictitious time step in which, taking advantage of the transport velocity, the field quantities are updated computing the convective fluxes generated moving the particles from the initial position to the position obtained using the implicit procedure. Secondly, a an MLS second-order reconstruction has been adopted to reinterpolated the particle volume, mass and momentum on the adjusted particle distribution. Overall, these methodologies keep the consistency in the scheme, therefore, it has been demonstrated that updating the physical quantities after the implicit iterative procedure is fundamental to increase the accuracy, in particular for high resolutions, in which the discretization error is minimized.

Chapter 10

Applications

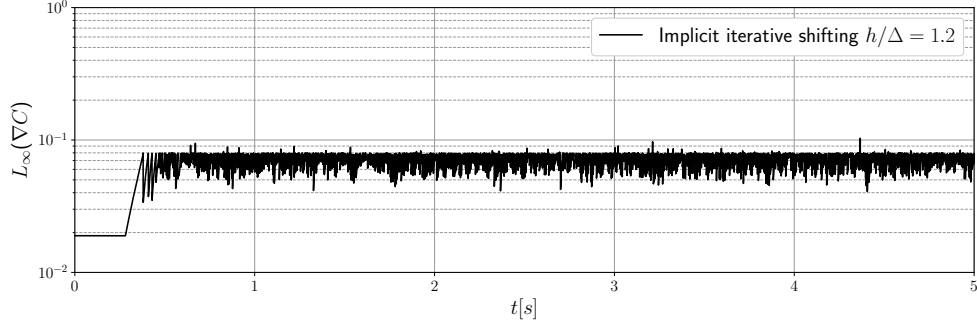
Chapter 7 presented an optimized formulation for an explicit PST while Chapter 8 proposed a completely new formulation for an implicit iterative PST. These methodologies have been tested against the Taylor Green vortex case for global evaluations, validating their overall applicability. In this chapter, these algorithms are utilized to simulate more complex applications, in order to compare them in different scenarios.

10.1 Moving box in squared domain

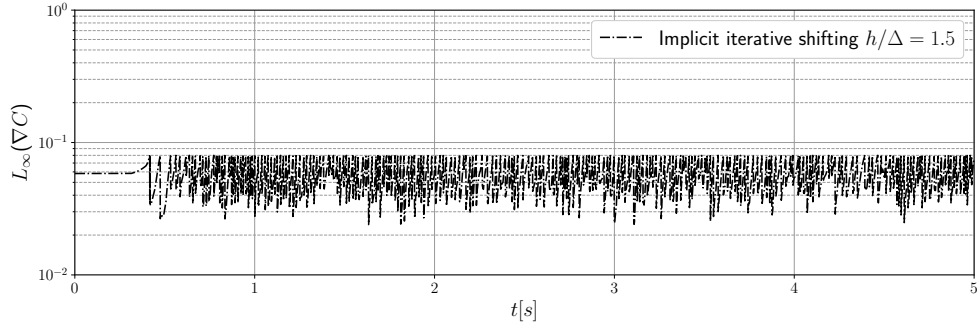
The first application considered is the moving square inside a rectangular box which has been described in Section 3.2. The results shown in the following section are referred to the resolution $\Delta/L = 0.0125$ and the Reynolds number $Re=100$.

10.1.1 Threshold evaluation for IIPS

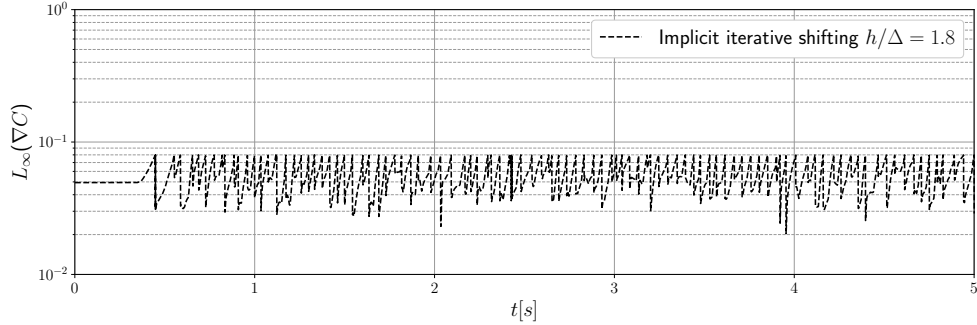
Initially, the moving box has been simulated focusing mainly on the kinematics, without any quantities updating after the IIPS, (Section 9.1.1 and Section 9.1.2). In order to assess which value of $L_\infty(\nabla C)$ has to be imposed as a threshold, during the physical simulation. A preliminary evaluation of the implicit procedure performances, regarding the quality of particle distribution and the computational time, has been conducted. The initial particle distribution has been obtained using an iterative pre-procedure, which, as illustrated in Section 8.3.2, is equivalent to applying the implicit shifting procedure on a static configuration, for several iterations. The value obtained, $L_\infty(\nabla C)_{Init}$, is larger than the one obtained for the TGV, due to the presence of solid boundaries, which has an influence on the minimum value of the gradient of particle concentration. Finally, it has been chosen $L_\infty(\nabla C)_{Thr} = 0.08/h$.



(a) $h/\Delta=1.2$



(b) $h/\Delta=1.5$



(c) $h/\Delta=1.8$

Figure 10.1 Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty thr} = 0.08/h$, h/Δ analysis.

An initial investigation, on the size of the kernel support, has been conducted and the test case has been performed using $h/\Delta = 1.2, 1.5, 1.8$; for these simulations, the results of the particle distribution analyses, in terms of the maximum value of particle concentration gradient, are reported in Figure 10.1. As shown in Figure 10.1 (a), in which $h/\Delta = 1.2$, the IIPS methodology is not able to guarantee the predefined threshold at all the physical time steps; this is due to the fact that the max number of Newton–Raphson iterations, which has been set equal to 20, is reached before the predefined level of error, in the particle distribution, is achieved; this additional condition has been introduced for practical reasons, to reduce computational costs. Conversely, increasing the size of the kernel support, $h/\Delta = 1.5$ or $h/\Delta = 1.8$, is sufficient to overcome the

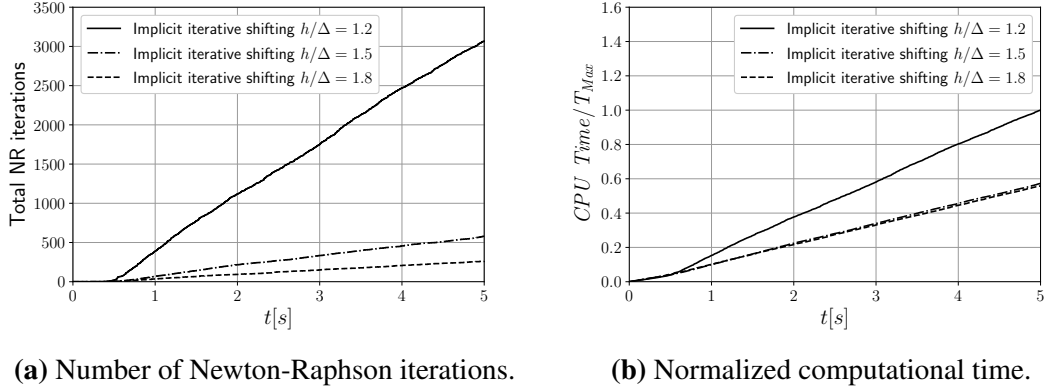


Figure 10.2 Moving Box test case. $\Delta/L = 0.0125$, $L_\infty(\nabla C)_{thr} = 0.08/h$.

above-mentioned numerical instability, as clarified in Figure 10.1 (b) and (c), in which the imposed $L_\infty(\nabla C)$ is always guaranteed.

The effects of varying the kernel size, on the total number of Newton-Raphson iterations and on the total computational time, are presented in Figure 10.2. In particular, as observed in Figure 10.2 (a), the total cumulative implicit iterations are almost 6 or 12 times, comparing the results for $h/\Delta=1.2$ respectively with the results for $h/\Delta=1.5$ and $h/\Delta=1.8$. Nevertheless, these differences are reduced in the total computational time, as confirmed by the analyses presented in Figure 10.2 (b), in which the total computational time of $h/\Delta=1.2$ are increased by around 60%, with respect to $h/\Delta=1.5$ and $h/\Delta=1.8$.

Reducing h/Δ decreases the effectiveness of the implicit iterative methodology, this is due to the fact that the system of linear equations (8.32) is assembled with a lower number of particles, reducing the implicit interactions among them. For this reason, the updated configuration is less stable, and small perturbations are sufficient to increase the $L_\infty(\nabla C)$ above the threshold. Therefore the IIPS has to be called a higher number of times, Figure 10.2 (a), on the contrary, a smaller kernel support reduces the computational time during the physical simulation, Figure 10.2 (b), partially counterbalancing the computational overheads.

For this reason, it has been chosen $h/\Delta = 1.5$, which allows keeping the predefined level of error in the particle distribution while does not increase the overall computational time. Another option to reduce the CPU time, maintaining the IIPS effectiveness, is to adopt different kernel sizes in the physical simulation and in the implicit procedure; these considerations are left for future investigations.

10.1.2 Particle shifting techniques comparison

Having set the threshold for the implicit procedure, $L_\infty(\nabla C)_{thr} = 0.08/h$, the size of the kernel support, $h/\Delta = 1.5$, and the reference speed of sound $c_0 = 12$ m/s, the

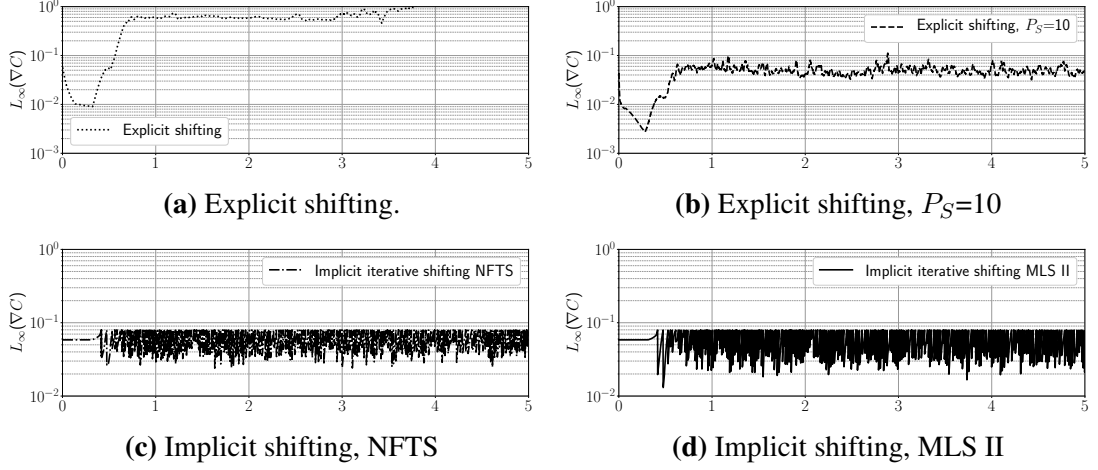


Figure 10.3 Moving Box test case. $\Delta/L = 0.0125$, $L_\infty(\nabla C)_{thr} = 0.08/h$. Shifting procedure comparison.

investigations on the different particle shifting techniques have been conducted; the implicit iterative shifting procedures are compared to the explicit shifting methods.

Focusing on the gradient of particle concentration, the results of $L_\infty(\nabla C)$, obtained using the original and the optimized explicit shifting, Section 7.1 and Section 7.2, and the implicit iterative procedures, Section 9.1.1 and Section 9.1.2, are shown in Figure 10.3. As demonstrated by Lee (2007), the case presents some numerical issues, due to the highly perturbed particle distribution, introduced by the motion of the squared box. As expected, the simulation in which the original explicit shifting is adopted, Figure 10.3 (a), manifests numerical instabilities and it does not reach the end. On the other hand, it has to be pointed out that the enhanced explicit shifting presents the same order of magnitude of $L_\infty(\nabla C)$, compared to the implicit iterative shifting procedures, as confirmed observing Figure 10.3 (b), and Figures 10.3 (c) and (d).

In Figure 10.4, the ∇C fields are shown at $t = 0.75$ s, moreover, in Figure 10.5, the particles distributions near the moving square are reported at the same physical time. These results are referred to a physical instant in which the object is still in an acceleration phase.

It is clear that the proposed shifting methods, both the optimized explicit shifting, with $P_S = 10$, and the implicit iterative procedures avoid the formation of areas with lack of particles on the tail of the moving object. Furthermore, it is notable that there are no relevant differences in the particles distributions reported in Figure 10.4 (c) 10.4 (d). Updating physical quantities, even with different methodologies, as illustrated in Chapter 9, do not strongly affect the particles configurations.

The non-dimensional velocity and the non-dimensional pressure fields are shown in Figures 10.6, 10.7 and 10.8, while the reference solution obtained with FD solver has been reported in Section 3.2.

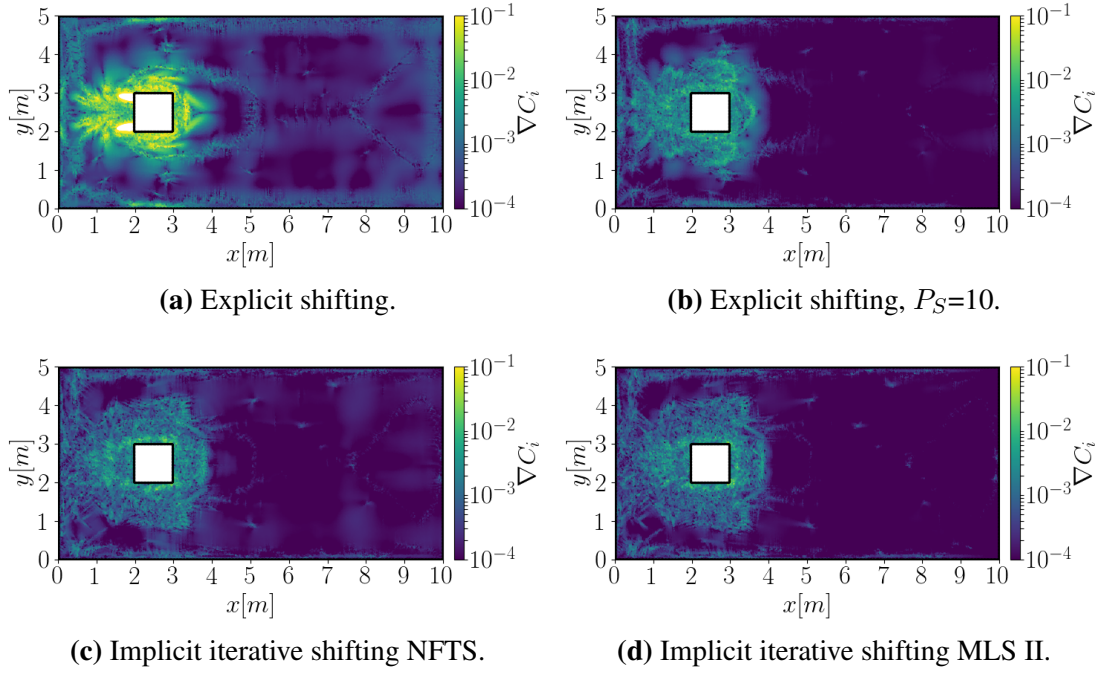


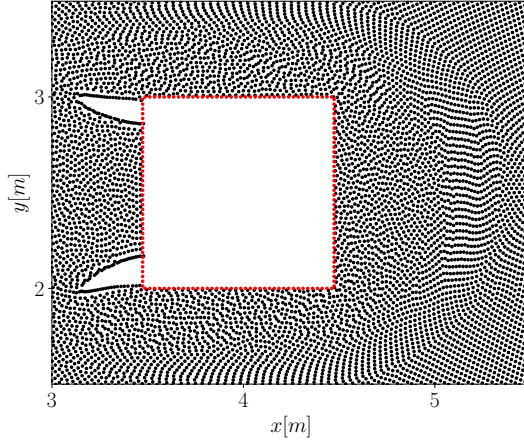
Figure 10.4 Moving Box test case. $\Delta/L = 0.0125$, $L_\infty(\nabla C)_{thr} = 0.08/h$. Shifting procedure comparison, gradient of particle concentration at $t=0.75$ s.

The results obtained using the improved explicit method, Figure 10.6, comparable with the reference solution regarding the velocity field, while the pressure field exhibits some noise, as expected, adopting a weakly-compressible approach.

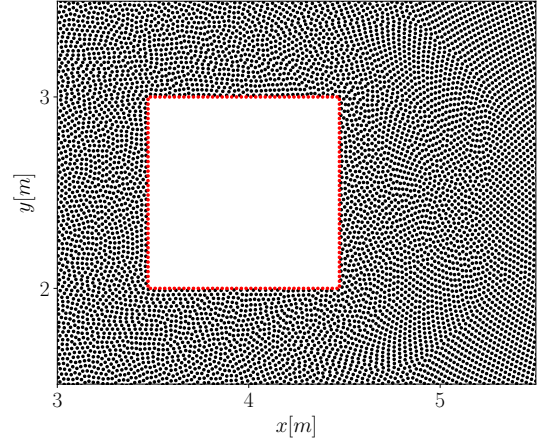
Introducing the procedures to update the physical quantities, after the IIPS, shows different outcomes, especially in the pressure field as shown in Figure 10.7 (b) and in Figure 10.8 (b). The fictitious time step, Section 9.1.1, produces a velocity field with great agreements while the pressure field results are quite noisy, Figure 10.7, this is due to the fact that the fluxes, in the artificial time step, are generated from a not smoothed field of implicit displacements. On the contrary, the second-order MLS reconstruction, Section 9.1.2, adds numerical dissipation to the model, and, therefore, the velocity field results are less accurate, while the pressure field is smoothed, even if a background pressure is generated, Figure 10.8.

The total computational times have been monitored and the results are shown in Figure 10.9. Due to the fact that the explicit shifting simulation does not reach the final physical time, $t = 5$ s, the CPU times are normalized using the explicit shifting with $P_S = 10$, which, as reported in Section 7.2, do not increase the computational cost. The ranges of computational overhead are similar to the ones obtained in the theoretical test case, Section 8.3.2, the fictitious time step and the MLS second-order reconstruction increase the CPU time respectively by 33 and by 54 percent.

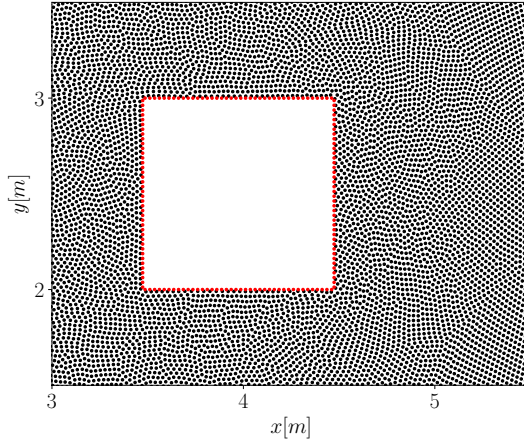
In conclusion, in this application, the presence of a moving square and external walls interfere with the shifting displacements generated by the implicit procedure; the



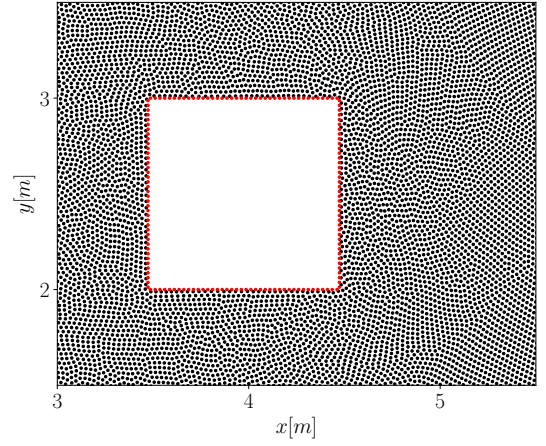
(a) Explicit shifting.



(b) Explicit shifting, $P_S=10$

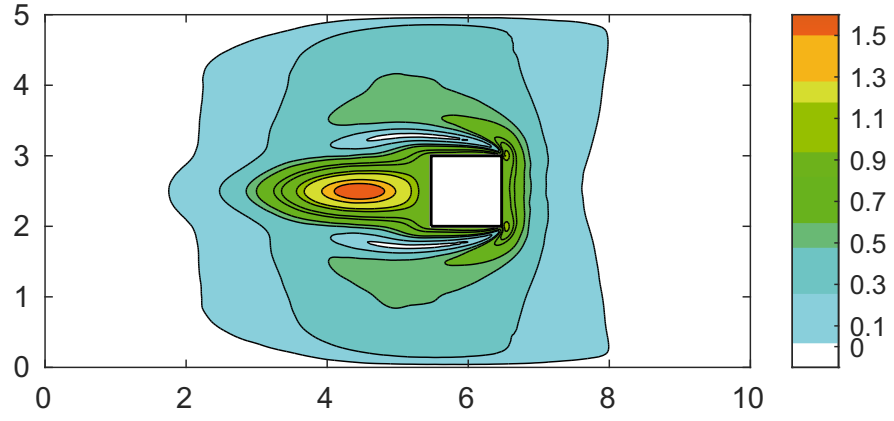


(c) Implicit iterative shifting NFTS.

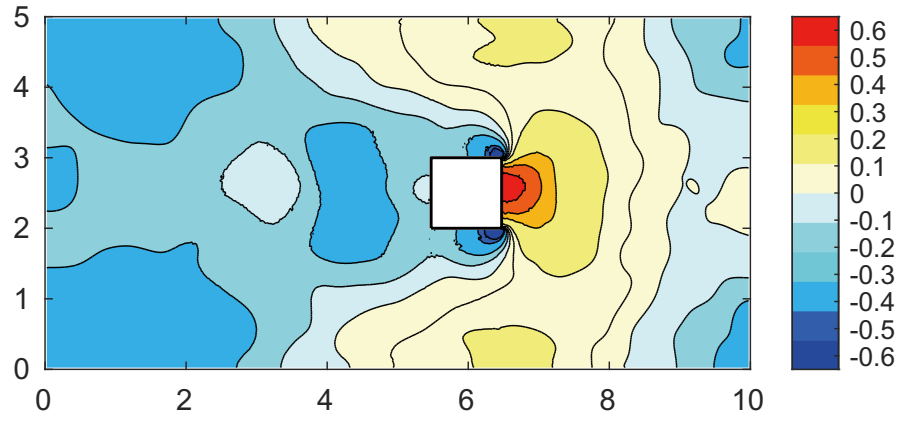


(d) Implicit iterative shifting MLS II.

Figure 10.5 Moving Box test case. $\Delta/L = 0.0125$, $L_\infty(\nabla C)_{thr} = 0.08/h$. Shifting procedure comparison, particle distribution at $t=0.75$ s.

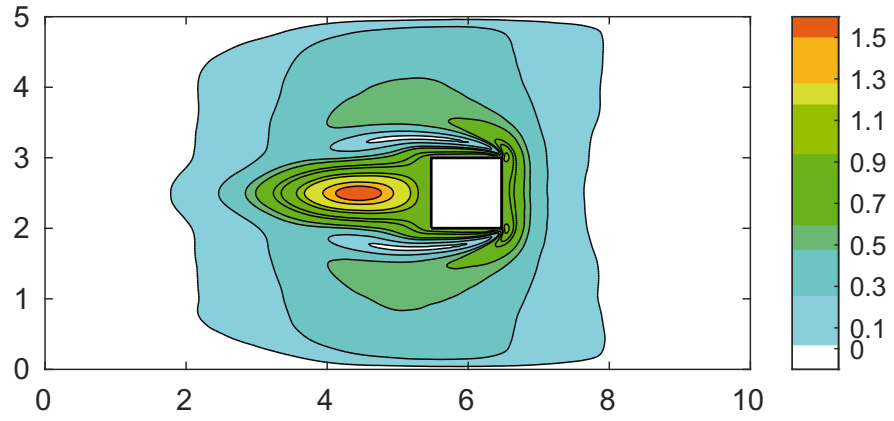


(a) Non-dimensional velocity field $\mathbf{v}/U_{ref}$.

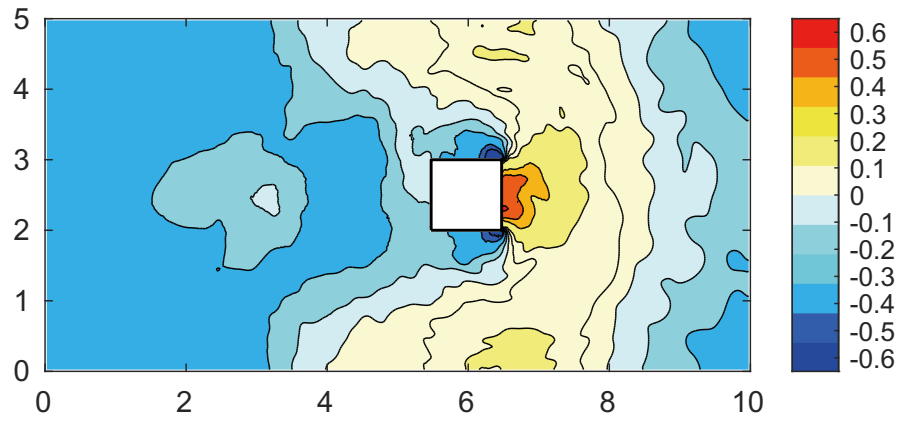


(b) Non-dimensional pressure field $p/(\rho U_{ref}^2)$.

Figure 10.6 Moving Box test case. $\Delta/L = 0.0125$, $L_\infty(\nabla C)_{thr} = 0.08/h$, $Re=100$. Explicit shifting, $P_S=10$, at $t=5$ s.

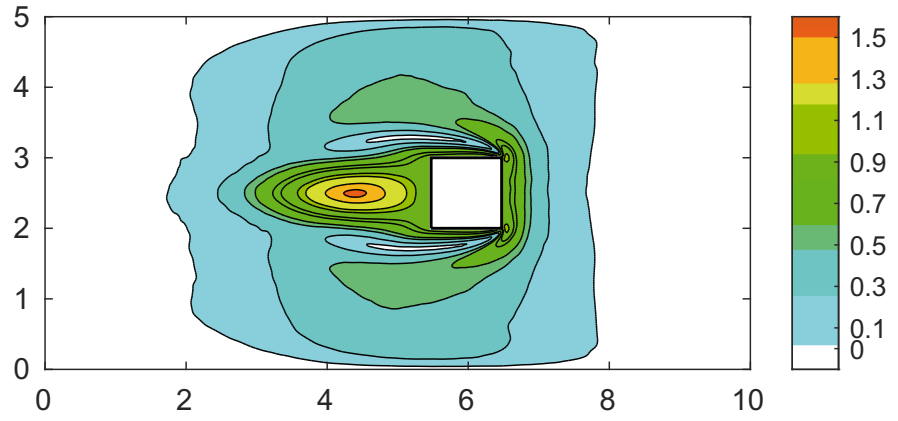


(a) Non-dimensional velocity field v/U_{ref} .

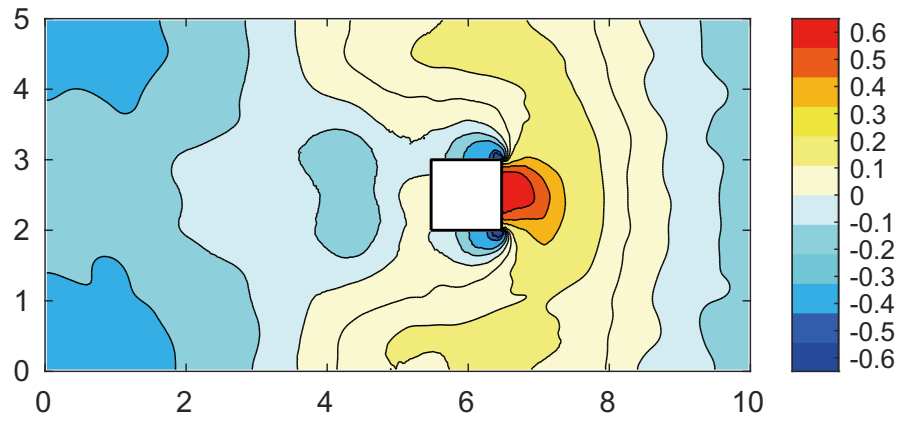


(b) Non-dimensional pressure field $p/(\rho U_{ref}^2)$.

Figure 10.7 Moving Box test case. $\Delta/L = 0.0125$, $L_{\infty}(\nabla C)_{thr} = 0.08/h$, $Re=100$. Implicit iterative shifting with NFTS, at $t=5$ s.



(a) on-dimensional velocity field v/U_{ref} .



(b) Non-dimensional pressure field $p/(\rho U_{ref}^2) - 0.6$.

Figure 10.8 Moving Box test case. $\Delta/L = 0.0125$, $L_\infty(\nabla C)_{thr} = 0.08/h$, $Re=100$. Implicit iterative shifting with MLS second-order, at $t=5$ s.

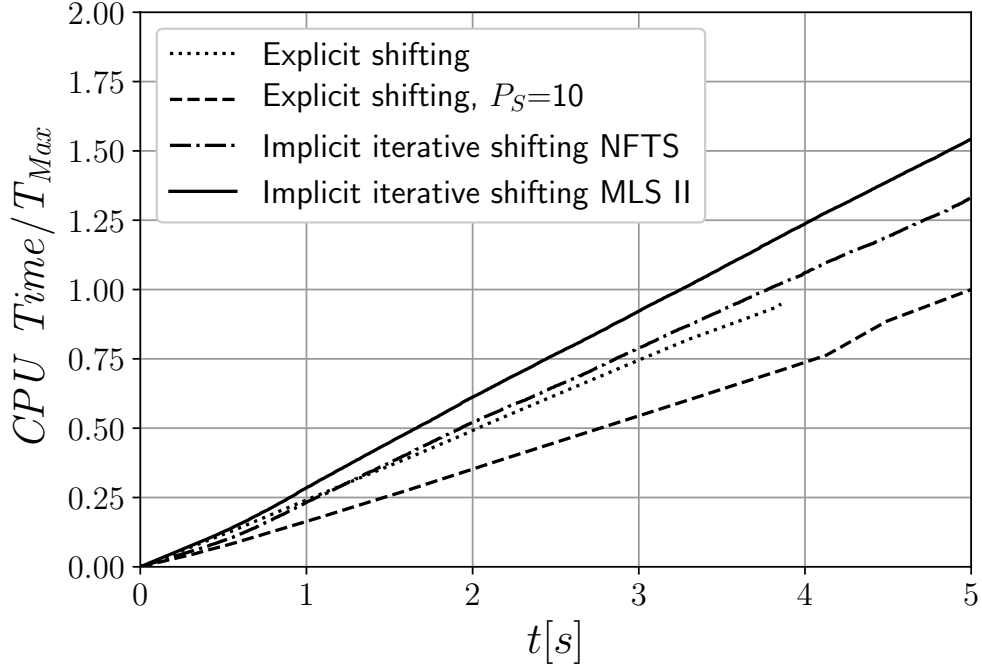


Figure 10.9 Moving Box test case. $\Delta/L = 0.0125$. Shifting procedure comparison, CPU time.

solid objects act as constraints on the rearrangement of particles, therefore the IIPS is not able to guarantee the same level of discretization error, as the one obtained in the TGV test case, with an unbounded domain. However, its performances are overall superior with respect to the explicit shifting methodology. Moreover, even introducing the procedures to update the physical quantities, which keep the consistency in the scheme, the computational times are not drastically increased and remain affordable.

10.2 Impinging jet on a flat surface

The second application is the impinging jet on a flat surface, which has been described in Section 3.3. It has been tested to establish the performance of the IIPS method in free surface problems. It has been initially simulated with a resolution $H/\Delta = 10$ for a preliminary studies, and $H/\Delta = 40$ for global assessments. The gradient of particle concentration, in a restricted area of the domain, a representative pressure field along the fluid-wall interface and the pressure at the stagnation point $P_0 = [0,0]$, during the entire simulations, are reported in the following analyses.

10.2.1 Threshold evaluation for IIPS

In this free surface test case, the implicit iterative method needs some modifications to deal with particles located at the free surface, at the inlet or at the outlets. The

detection location algorithm is able to find the particles near the interfaces; it is based on a reference value of ∇C , which is computed considering the kernel support half empty. In this way, particles with a lower value of ∇C are identified as belonging to interfaces, these particles keep moving through their Lagrangian trajectories, without any particle position correction. Therefore, the free surface particles are removed from the system of linear equations (8.32), which is solved in the iterative implicit shifting procedure, and, during the IIPS, are considered as fixed wall particles. The size of the adjusted linear system downgrades to $d \cdot (n - n_{int})$, where n_{int} is the number of the interface particles. Using this expedient allows to correct the particle position for the interior fluid particles only. The presence of a wall boundary, as seen in Section 10.1, do not add any complexities to the problem.

In this test case, at the beginning of the simulation the domain is empty, therefore, the preliminary evaluations on $L_\infty(\nabla C)$ through a static pre-procedure, illustrated in Section 10.1.1, cannot be adopted. To assess the value of $L_\infty(\nabla C)_{thr}$, which has to be imposed during the physical simulations, an initial investigation on the particle distribution has been conducted, using $H/\Delta = 10$. In this assessment, it has been studied the trends of the gradient of particle concentration at the first steps of the physical simulation, without investigations on the physical quantities. As specified in the previous application, the use of the fictitious time step or the MLS reinterpolation has no significant interference with the particles distributions. In Figure 10.10 are reported the $L_\infty(\nabla C)$, computed in a restricted area of the domain, where $x/L = [-0.3, 0.3]$ and $z/L = [0, 0.6]$, for the explicit and the implicit procedures; the latter one is called at each physical iteration, regardless to the maximum ∇C value. The particle distributions and the gradients of particle concentration, for the simulation presented in Figure 10.10, are shown in Figure 10.11, 10.12 and 10.13, at the normalized time $tU/L = 0.0525$. In the impinging jet, the Lagrangian trajectories are particularly distorted in the areas around the stagnation point and its axis, where the water jet splits, thus the gradient of particle concentration has been evaluated in this area, (red box in Figures). It is clear that the IIPS acts to redistribute the particle even in the zone close to the stagnation point in which the physical kinematics are low.

Through these investigations, it has been set $L_\infty(\nabla C)_{thr} = 0.02/h$, as a condition on the particle distribution disorder, that is used in the simulations presented in the following section. Moreover, it has been immediately observed that the IIPS is effective since the first steps of the physical simulation and it reduces the norm of ∇C by almost an order of magnitude with respect to the explicit procedures, Figure 10.10.

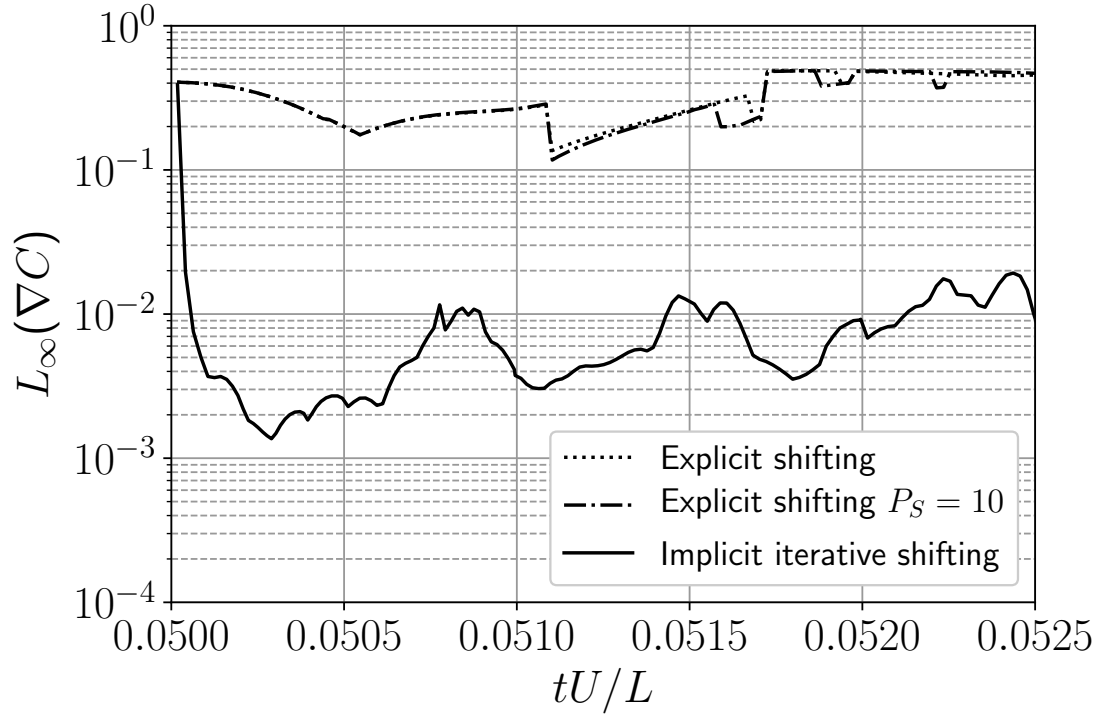


Figure 10.10 Impinging Jet test case. $H/\Delta = 10$, Shifting procedure comparison.

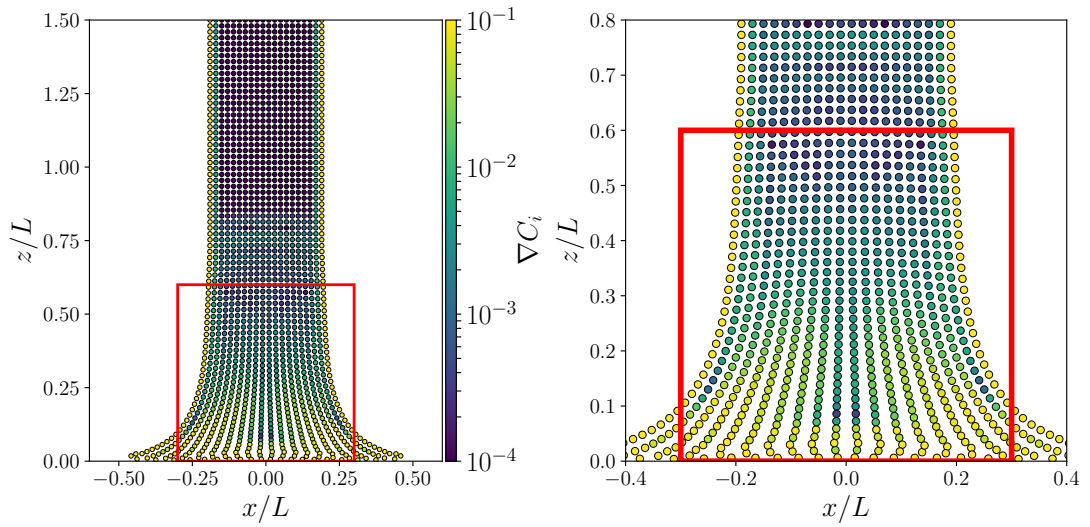


Figure 10.11 Impinging Jet test case. $H/\Delta = 10$, ∇C , explicit shifting, at $tU/L = 0.0525$.

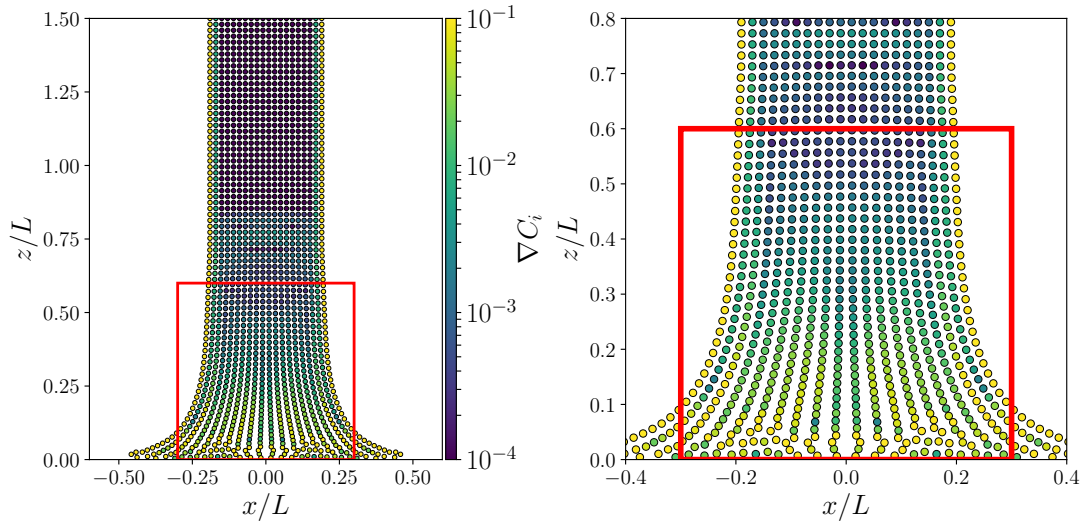


Figure 10.12 Impinging Jet test case. $H/\Delta = 10$, ∇C , explicit shifting, $P_S=10$ at $tU/L = 0.0525$.

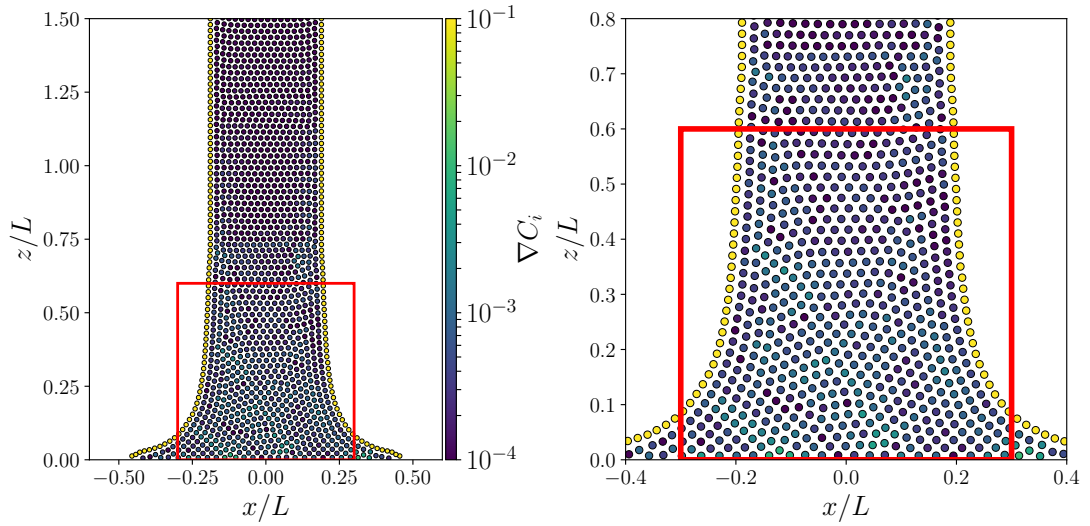


Figure 10.13 Impinging Jet test case. $H/\Delta = 10$, ∇C , implicit iterative shifting at $tU/L = 0.0525$.

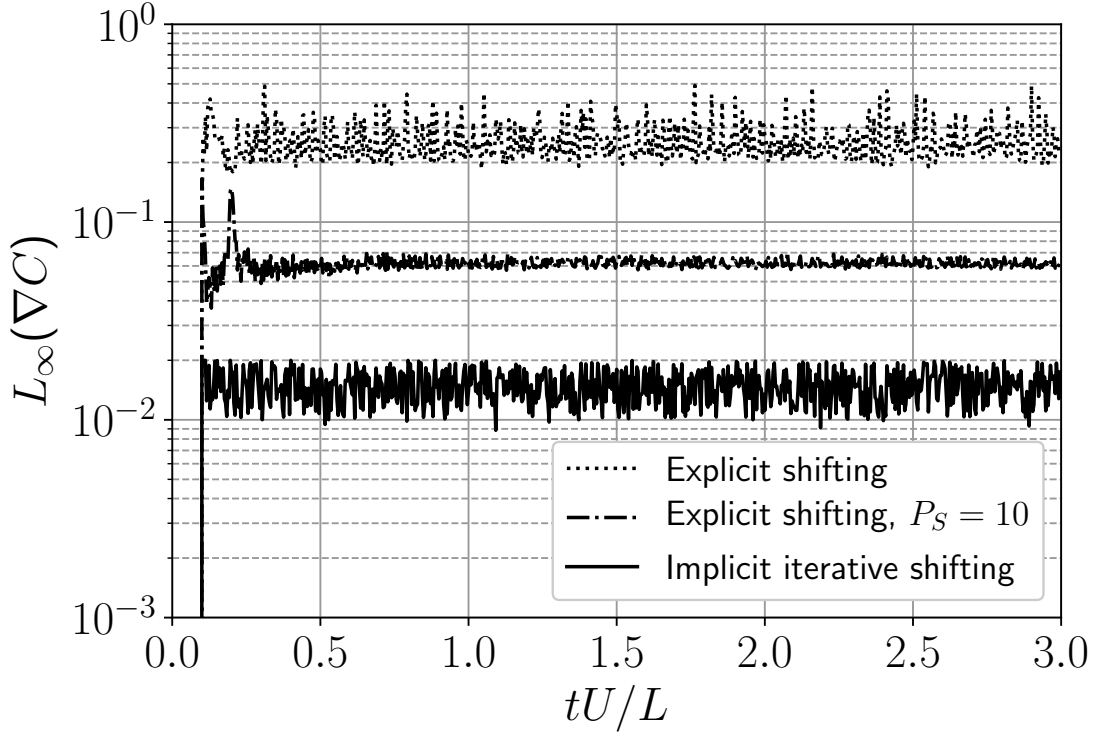


Figure 10.14 Impinging Jet test case. $\Delta/L = 0.0005$, ∇C , shifting procedure comparison.

10.2.2 Particle shifting techniques comparison

Having set the threshold for the implicit procedure, $L_\infty(\nabla C)_{thr} = 0.02/h$, and the size of the kernel support, $h/\Delta = 1.4$, the comparison between the shifting techniques has been carried on. The simulations reach the steady-state at $tU/L = 2$ and it has been considered the time interval $tU/L = [2, 3]$ for the pressure analyses. The inflow velocity is $U = 100$ m/s, the reference speed of sound is set $c_0 = 1000$ m/s, which is equal to 10 times the maximum velocity, with respect to the consideration on the Mach number, for weakly compressible models. As previously mentioned, it has been monitored the pressure at the stagnation point and at the interface with the flat wall in the interval $x/L = [-0.8, 0]$, in a defined interval of time. These results are compared to the analytical reference solution reported in equation (3.3).

In Figure 10.14 is shown the particle distribution quality, measured as $L_\infty(\nabla C)$ in the above-mentioned area of interest, in long run simulations, these results are reported from the beginning of the simulation. It has been confirmed that since the moment in which the jet impacts the plate the implicit shifting formulation is more effective compared to the explicit ones. Moreover, even in presence of a free surface, the implicit iterative procedure is able to guarantee the predefined level of particle disorder, in terms of $L_\infty(\nabla C)$.

The pressure at the interface with the plate, averaged in the normalized time inter-

x/L	Explicit		Explicit $P_S = 10$		IIPS - NFTS		IIPS - MLS II	
	$P/\rho U^2$	$E_r\%$	$P/\rho U^2$	$E_r\%$	$P/\rho U^2$	$E_r\%$	$P/\rho U^2$	$E_r\%$
0.0	0.448	10.48	0.488	2.49	0.486	2.85	0.493	1.45
-0.05	0.444	10.38	0.483	2.47	0.481	2.88	0.487	1.62
-0.10	0.432	10.37	0.468	2.81	0.466	3.18	0.473	1.84
-0.15	0.412	10.25	0.444	3.23	0.441	3.73	0.448	2.24
-0.20	0.384	9.80	0.410	3.86	0.408	4.16	0.414	2.74
-0.25	0.351	8.82	0.368	4.23	0.368	4.36	0.373	3.13
-0.30	0.309	7.84	0.320	4.48	0.320	4.49	0.324	3.19
-0.35	0.263	6.17	0.269	3.91	0.269	3.81	0.273	2.69
-0.40	0.216	3.43	0.218	2.47	0.218	2.26	0.221	1.16

Table 10.1 Impinging Jet test case. $\Delta/L = 0.0005$. Normalized average pressure on the plate and percentage error $E_r\%$.

val $tU/L = [2, 3]$, is shown in Figure 10.15, the numerical values for probe points and the percentage errors are reported in Table 10.1. In these results, it can be seen that the original explicit shifting formulation has a percentage error higher than 10%, close to the stagnation point. Differently, introducing the improved explicit shifting method, with $P_S = 10$, or the implicit formulation, through both the procedures to update the physical quantities, drastically reduce the pressure error along the plate. Moreover, in Figure 10.15 is observed that the explicit shifting with P_0 is quite effective for high resolution, $H/\Delta = 40$, while for lower resolutions, $H/\Delta = 20$ and $H/\Delta = 10$ the pressure profiles have more discrepancy compared to the analytical solution. The percentage error of $P(0)/\rho U^2$, which corresponds to the point of higher perturbation of the Lagrangian streamlines, using the fictitious time step and the MLS interpolation are respectively 2.85% and 1.45% in respect to the analytical value.

The trends of the normalized pressure at the stagnation point are presented in Figure 10.16, in the time interval $tU/L = [2, 3]$. From the comparison between Figure 10.16 (a) and Figure 10.16 (b) can be noted that the improved explicit shifting increases the quality of the particles distribution, as can be seen in Figure 10.14, improving the overall accuracy, but it introduces more fluctuation in the pressure field. Regarding the methodologies proposed to updated the physical quantities after the implicit iterative procedure, the fictitious time step, Figure 10.16 (c), produces a quite noisy pressure at the stagnation point, whereas the MLS interpolation, Figure 10.16 (d), generates smoother results. In Figure 10.17 the computational times are reported and the results are similar to the ones obtained in the theoretical test cases, Section 8.3.2.

Through the impinging jet impacting on a flat, it has been confirmed that even in presence of a free surface, the implicit methodology is feasible in these applications.

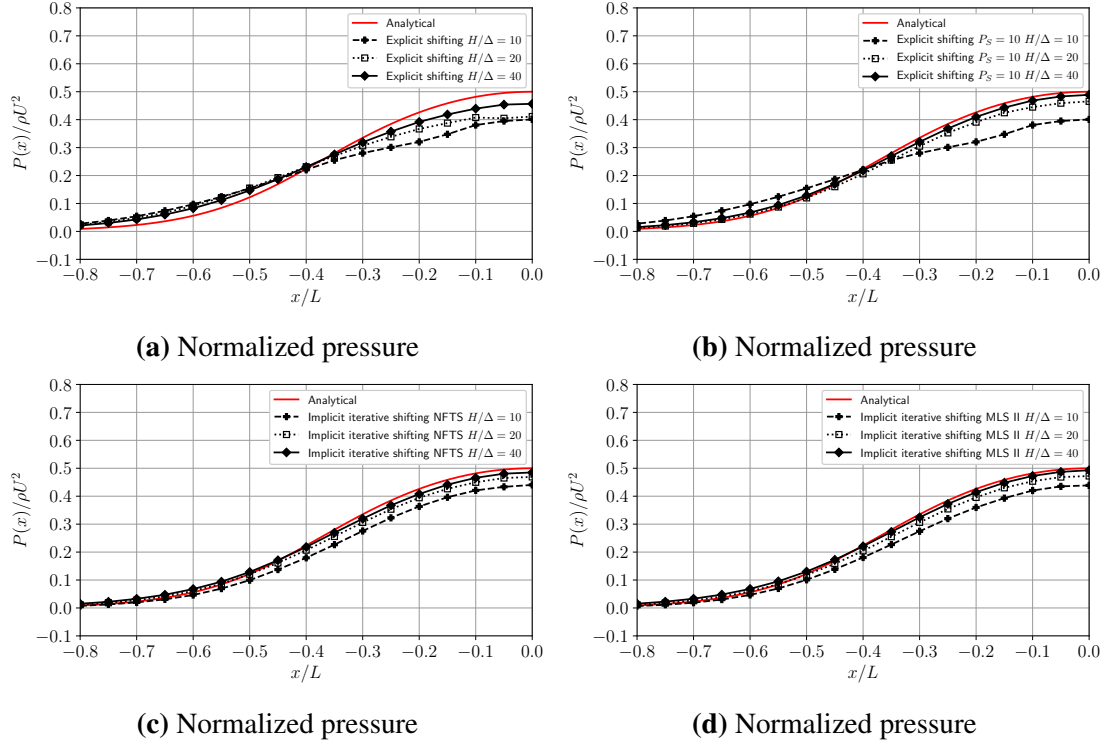
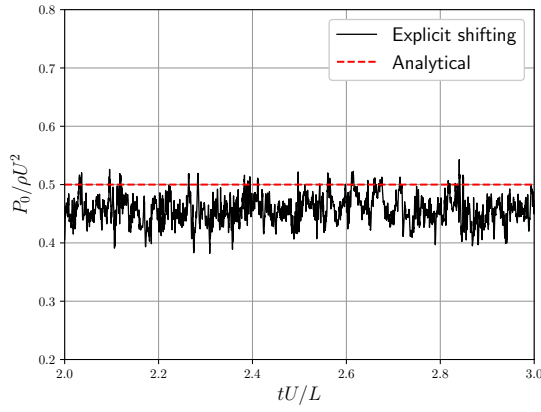


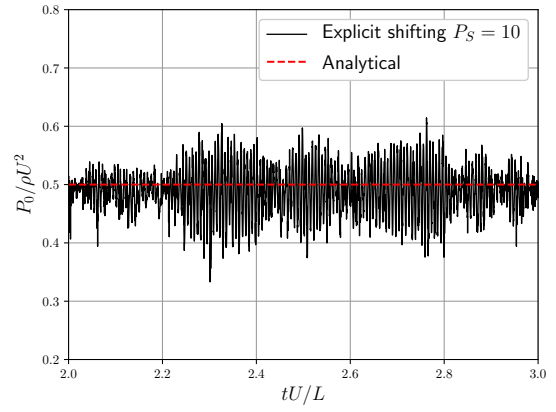
Figure 10.15 Impinging Jet test case. $\Delta/L = 0.002$, $\Delta/L = 0.001$, $\Delta/L = 0.0005$, shifting procedure comparison, normalized pressure at P_0 in the time interval $tU/L = [2, 3]$.

10.3 Summary and conclusions

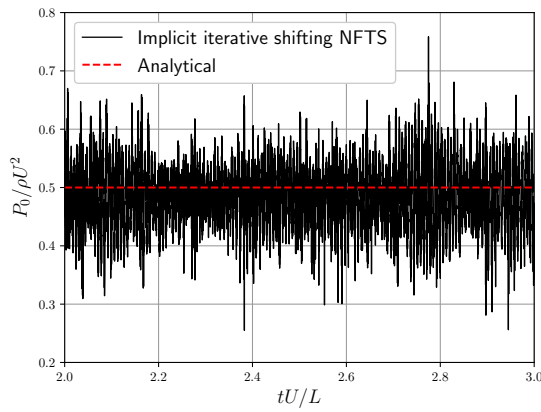
In this chapter two different applications, which are representative of a wide range of practical problems, have been simulated using the shifting methodologies introduced in this project. The optimized explicit shifting has shown good results in the application with bounded domain, while in presence of free surfaces the effectiveness is slightly reduced for low resolutions. Moreover, it does not increase the computational time, compared to the original formulation, therefore, it is still considered a valid methodology for particle shifting. In addition, the implicit iterative methodology has demonstrated its superior performance in controlling the particle distribution quality and in maintaining a predefined level of discretization error in both the moving squared box and in the impinging jet test cases. The methodologies introduced in Section 9.1.1 and in Section 9.1.2 are different under a mathematical point of view and the outcomes that these provided, in the cases presented, are not coincident. Nevertheless, it has been demonstrated that both the fictitious time step and the MLS interpolation are valid formulations, thus based on the aims of the application under investigation, it can be adopted the more suitable methodology.



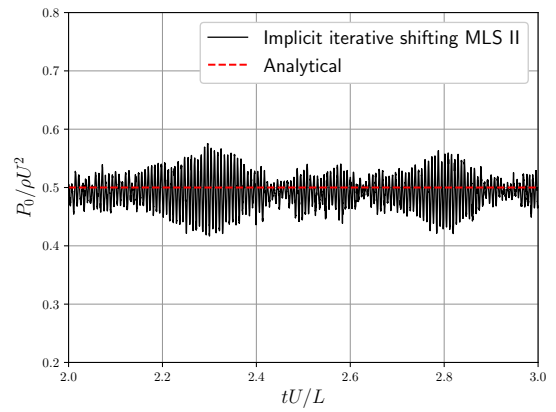
(a) Normalized pressure



(b) Normalized pressure



(c) Normalized pressure



(d) Normalized pressure

Figure 10.16 Impinging Jet test case. $\Delta/L = 0.0005$, shifting procedure comparison, normalized pressure at P_0 in the time interval $tU/L = [2, 3]$.

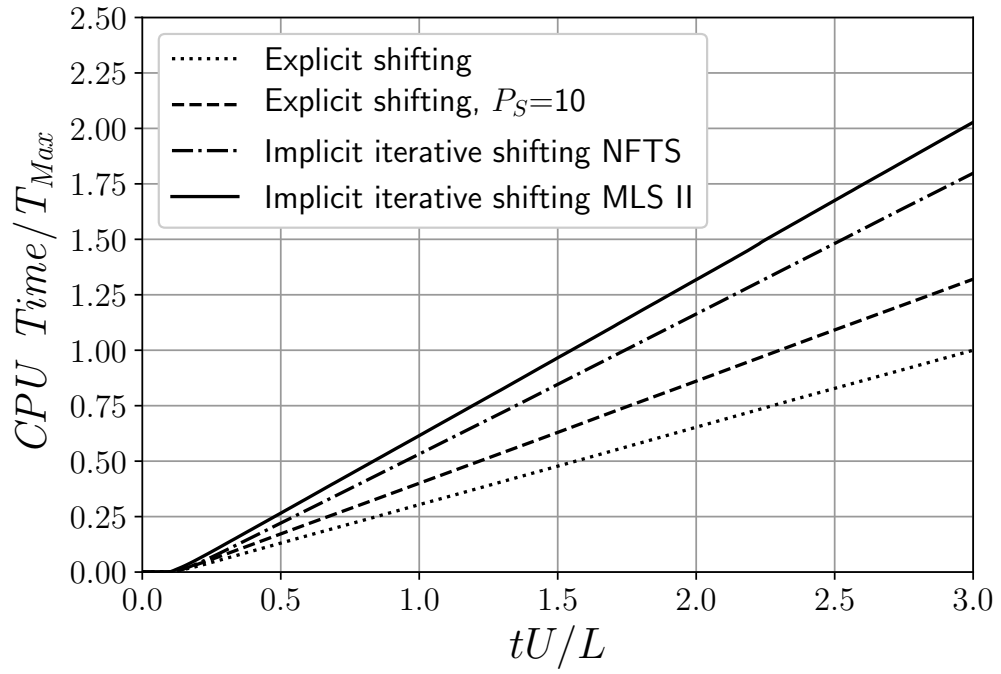


Figure 10.17 Impinging Jet test case. $\Delta/L = 0.0005$. Shifting procedure comparison, CPU time.

Chapter 11

Conclusions and perspectives

11.1 Conclusions

The main objective of the present thesis is to analyze the link between particle distribution and accuracy in the Smoothed Particle Hydrodynamics schemes. In the SPH schemes, Particle Shifting Techniques have been introduced in the past years as methodologies with the aim of reducing the discretization error.

In this work, an optimization for the existing explicit particle shifting technique based on Fick's law of diffusion has been initially presented; this shifting algorithm has been implemented in the code ASPHODEL, which is an in house-software developed by the ANDRITZ group, used to model hydraulic turbines. This solver adopts a specific version of the ALE-SPH numeric models, which differ from the classical SPH in the way the particle motion is described. In the ALE-SPH schemes, the position of the interpolation points is defined by the transport velocity, which is not equal to the fluid velocity; this parameter can be arbitrarily formulated, allowing to correct the particle distribution without introducing any numerical inconsistency and to grant mass and momentum conservation. The improved shifting algorithm is able to maintain a homogeneous particle distribution while keeping the Mach number around 0.1, as common in Weakly-Compressible SPH models, without adding extra computational overhead. Explicit shifting techniques have been widely applied and their effectiveness, in improving the particle distribution quality, has been empirically proved, however, the results obtained are very case-dependent, due to the local and explicit characteristics of this type of algorithms, which are not able to guarantee that the discretization error is maintained under a certain threshold in every part of the domain during the entire simulation.

Having this latest issue in mind, to overcome this limitation, an innovative formulation has been developed based on the global assessment of the particle distribution aiming at guaranteeing that discretization error is reduced all over in the domain. The particle concentration gradient is still the key component underneath this idea, but it

is adopted to assemble a system of linear equations that are derived minimizing the particle concentration gradient for all the particles in the domain. In this way, all the interpolation points are taken into account at the same time, creating implicit connections between the shifting displacements. This novel methodology has been called Implicit Iterative Particle Shifting (IIPS) because it uses an iterative procedure to minimize those functions.

This IIPS takes on a numerical issue of mesh-less methods based on kernel basis functions, it has not been thought specifically for SPH and neither for fluid dynamic simulations. The IIPS directly tackles the discretization error in the meshless methods. For this reason, in contrast to other shifting algorithms, properly developed for SPH simulations, the methodology does not require any empirical coefficient or parameter related with the fluid characteristics, for example, the reference speed of sound or the fluid velocity. Moreover, from the numerical experiences conducted and presented in this thesis, it has been demonstrated that the IIPS is capable of minimizing the discretization error a few iterations, without any external constraints on the maximum allowed shifting. The IIPS requires to define only a threshold value for the particle concentration gradient, which defines the predefined level of disorder, this value is resolution independent, but it needs to be adjusted based on the applications: walls and free-surface affect the minimum value that can be reached.

The ways in which the IIPS can be embedded in an SPH solver, for fluids dynamics, are different and, in this project, two strategies have been herein proposed for an SPH-ALE solver. These methodologies have been called "Implicit iterative shifting with fictitious time step" and "Implicit iterative shifting with MLS interpolation", both of them have been introduced to update the physical quantities at the end of the implicit procedure allowing to maintain consistency in the numerical scheme. To adjust volume, mass and momentum on the particle distribution obtained using the IIPS, the first one adopts the ALE formalism to compute the convective fluxes generated by the implicit shifting displacements, while the second one makes a pure geometric interpolation using the information of the original particle distribution.

The implicit iterative procedures have been tested in different cases; in the Taylor-Green vortex, these methodologies are able to reduce the error associated with the discretization approximation by more than one order of magnitude and to increase the convergence rate to a higher than second-order, compared with the explicit shifting formulation. These results have been obtained without adding significant extra computational time to the global cost. Additionally, in this project, the implicit method has been applied to the simulation of a bounded case, the moving square in a rectangular box, and a free surface case, the jet impinging on a flat surface. These tests have con-

firmed that the method has the capability to be used in SPH schemes in a wide range of problems, moreover, even in these applications the IIPS prevents the formation of areas with high particle distribution disorder and its performances are superior with respect to explicit shifting procedures.

11.2 Perspectives

The innovative implicit iterative particle shifting methodology has been proposed and reported in one and two dimensions but its extension to 3D does not present any theoretical issues and it can be derived straightforward.

The cases used to validate the methodology are representative of real engineering problems, further investigations in applications with multi-fluid or multi-phase cases are left for future investigations.

Some specific adjustments on the algorithm are required in simulations with fragmented domains, typically in highly distorted and violent flows.

The work on the implicit iterative methodology is seen as an initial step, the view regarding the particle shifting technique has been moved from a local to a global approach, while future investigations are required to better understand the methodology both under mathematical and physical perspectives, some relevant goals have been already achieved as presented in the thesis.

Additionally, efforts must be done to parallelize the technique, even if there are already several methods to solve large linear system, the procedure need to be speeded up with multiprocessors or GPU.

Bibliography

- Amicarelli, A., Marongiu, J.-C., Leboeuf, F., Leduc, J. and Caro, J. (2011), ‘Sph truncation error in estimating a 3d function’, *Computers and Fluids* **44**(1), 279–296.
- Antuono, M., Colagrossi, A. and Marrone, S. (2012), ‘Numerical diffusive terms in weakly-compressible sph schemes’, *Computer Physics Communications* **183**(12), 2570–2580.
- Antuono, M., Sun, P., Marrone, S. and Colagrossi, A. (2021), ‘The δ -ale-sph model: an arbitrary lagrangian-eulerian framework for the δ -sph model with particle shifting technique’, *Computers and Fluids* **216**, 104806.
- Barcarolo, Touzé, L., Oger and Vuyst, D. (2014), ‘Adaptive particle refinement and derefinement applied to the smoothed particle hydrodynamics method’, *Journal of Computational Physics* **273**, 640–657.
- Børve, S., Omang, M. and Trulsen, J. (2005), ‘Regularized smoothed particle hydrodynamics with improved multi-resolution handling’, *Journal of Computational Physics* **208**(1), 345–367.
- Chaniotis, A. and Poulidakos, D. (2004), ‘High order interpolation and differentiation using b-splines’, *Journal of Computational Physics* **197**(1), 253–274.
- Chaniotis, A., Poulidakos, D. and Koumoutsakos, P. (2002), ‘Remeshed smoothed particle hydrodynamics for the simulation of viscous and heat conducting flows’, *Journal of Computational Physics* **182**(1), 67–90.
- Chiron, L., Oger, G., De Leffe, M. and Le Touzé, D. (2018), ‘Analysis and improvements of adaptive particle refinement (apr) through cpu time, accuracy and robustness considerations’, *Journal of Computational Physics* **354**, 552–575.
- Chow, A. D., Rogers, B. D., Lind, S. J. and Stansby, P. K. (2018), ‘Incompressible sph (isph) with fast poisson solver on a gpu’, *Computer Physics Communications* **226**, 81–103.
- Colagrossi, A., Bouscasse, B., Antuono, M. and Marrone, S. (2012), ‘Particle packing algorithm for sph schemes’, *Computer Physics Communications* **183**(8), 1641–1653.

- Colicchio, G., Greco, M. and Faltinsen, O. (2006), 'Fluid-body interaction on a cartesian grid: dedicated studies for a cfd validation', *Proc. IWWFEB* **6**.
- Dehnen, W. and Aly, H. (2012), 'Improving convergence in smoothed particle hydrodynamics simulations without pairing instability', *Monthly Notices of the Royal Astronomical Society* **425**(2), 1068–1082.
- Delorme, L., Colagrossi, A., Souto-Iglesias, A., Zamora-Rodriguez, R. and Botia-Vera, E. (2009), 'A set of canonical problems in sloshing, part i: Pressure field in forced roll—comparison between experimental results and sph', *Ocean Engineering* **36**(2), 168–178.
- Evans, M. W., Harlow, F. H. and Bromberg, E. (1957), The particle-in-cell method for hydrodynamic calculations, Technical report, LOS ALAMOS NATIONAL LAB NM.
- Feldman, J. and Bonet, J. (2007), 'Dynamic refinement and boundary contact forces in SPH with applications in fluid flow problems', *International Journal for Numerical Methods in Engineering* **72**(3), 295–324.
- Fourey, G., Hermange, C., Le Touzé, D. and Oger, G. (2017), 'An efficient fsi coupling strategy between smoothed particle hydrodynamics and finite element methods', *Computer Physics Communications* **217**, 66–81.
- Fourtakas, G. and Rogers, B. (2016), 'Modelling multi-phase liquid-sediment scour and resuspension induced by rapid flows using smoothed particle hydrodynamics (sph) accelerated with a graphics processing unit (gpu)', *Advances in Water Resources* **92**, 186–199.
- Franke, C. and Schaback, R. (1998), 'Solving partial differential equations by collocation using radial basis functions', *Applied Mathematics and Computation* **93**(1), 73–82.
- Gingold, R. A. and Monaghan, J. J. (1977), 'Smoothed particle hydrodynamics: theory and application to non-spherical stars', *Monthly notices of the royal astronomical society* **181**(3), 375–389.
- Gomez-Gesteira, M., Rogers, B. D., Dalrymple, R. A. and Crespo, A. J. (2010), 'State-of-the-art of classical sph for free-surface flows', *Journal of Hydraulic Research* **48**(sup1), 6–27.
- Gotoh, H. and Khayyer, A. (2016), 'Current achievements and future perspectives for projection-based particle methods with applications in ocean engineering', *Journal of Ocean Engineering and Marine Energy* **2**(3), 251–278.

Gray, J. P. and Monaghan, J. J. (2004), ‘Numerical modelling of stress fields and fracture around magma chambers’, *Journal of Volcanology and Geothermal Research* **135**(3), 259–283.

Green, M. D., Vacondio, R. and Peiró, J. (2019), ‘A smoothed particle hydrodynamics numerical scheme with a consistent diffusion term for the continuity equation’, *Computers and Fluids* **179**, 632–644.

Harten, A. and Hyman, J. M. (1983), ‘Self adjusting grid methods for one-dimensional hyperbolic conservation laws’, *Journal of computational physics* **50**(2), 235–269.

Hietel, D., Steiner, K. and Struckmeier, J. (2000), ‘A finite-volume particle method for compressible flows’, *Mathematical Models and Methods in Applied Sciences* **10**(09), 1363–1382.

Hirsch, C. (2007), *Numerical computation of internal and external flows: The fundamentals of computational fluid dynamics*, Elsevier.

Hirt, C. W., Amsden, A. A. and Cook, J. (1974), ‘An arbitrary lagrangian-eulerian computing method for all flow speeds’, *Journal of computational physics* **14**(3), 227–253.

Kansa, E. J. (1990), ‘Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics—i surface approximations and partial derivative estimates’, *Computers and Mathematics with applications* **19**(8-9), 127–145.

Khayyer, A., Gotoh, H. and Shimizu, Y. (2017), ‘Comparative study on accuracy and conservation properties of two particle regularization schemes and proposal of an optimized particle shifting scheme in isph context’, *Journal of Computational Physics* **332**, 236–256.

Khayyer, A., Gotoh, H. and Shimizu, Y. (2019), ‘A projection-based particle method with optimized particle shifting for multiphase flows with large density ratios and discontinuous density fields’, *Computers and Fluids* **179**, 356–371.

Koshizuka, S. and Oka, Y. (1996), ‘Moving-particle semi-implicit method for fragmentation of incompressible fluid’, *Nuclear science and engineering* **123**(3), 421–434.

Koumoutsakos, P. (2005), ‘Multiscale flow simulations using particles’, *Annu. Rev. Fluid Mech.* **37**, 457–487.

Lastiwka, M., Quinlan, N. J. and Basa, M. (2005), ‘Adaptive particle distribution for smoothed particle hydrodynamics’, **46**(10), 1403–1409.

URL: <http://dx.doi.org/10.1002/fld.891>

- Leduc, J. (2010), Etude physique et numérique de l'écoulement dans un système d'injection de turbine Pelton, PhD thesis, Thèse de doctorat, Ecole Centrale de Lyon.
- Lee, E.-S. (2007), *Truly incompressible approach for computing incompressible flow in SPH and comparisons with the traditional weakly compressible approach*, The University of Manchester (United Kingdom).
- Lee, E. S., Moulinec, C., Xu, R., Violeau, D., Laurence, D. and Stansby, P. (2008), 'Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method', *Journal of Computational Physics* **227**(10), 8417–8436.
- LeVeque, R. J. (1992), *Numerical methods for conservation laws*, Vol. 132, Springer.
- Li, Z. (2013), Développement d'une méthode de simulation de couplage fluide-structure à l'aide de la méthode SPH, PhD thesis, Ecully, Ecole centrale de Lyon.
- Li, Z., Leduc, J., Nunez-Ramirez, J., Combescure, A. and Marongiu, J.-C. (2015), 'A non-intrusive partitioned approach to couple smoothed particle hydrodynamics and finite element methods for transient fluid-structure interaction problems with large interface motion', *Computational Mechanics* **55**(4), 697–718.
- Libersky, L. D. and Petschek, A. G. (1991), Smooth particle hydrodynamics with strength of materials, in 'Advances in the free-Lagrange method including contributions on adaptive gridding and the smooth particle hydrodynamics method', Springer, pp. 248–257.
- Lind, S. J. and Stansby, P. (2016), 'High-order eulerian incompressible smoothed particle hydrodynamics with transition to lagrangian free-surface motion', *Journal of Computational Physics* **326**, 290–311.
- Lind, S., Xu, R., Stansby, P. and Rogers, B. D. (2012), '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**(4), 1499–1523.
- Liu, G.-R. and Liu, M. B. (2003), *Smoothed particle hydrodynamics: a meshfree particle method*, World scientific.
- Liu, M. and Liu, G.-R. (2006), 'Restoring particle consistency in smoothed particle hydrodynamics', *Applied numerical mathematics* **56**(1), 19–36.
- Long, T., Hu, D., Yang, G. and Wan, D. (2016), 'A particle-element contact algorithm incorporated into the coupling methods of fem-isph and fem-wcsph for fsi problems', *Ocean Engineering* **123**, 154–163.

- Lucy, L. B. (1977), ‘A numerical approach to the testing of the fission hypothesis’, *The astronomical journal* **82**, 1013–1024.
- Marongiu, J.-C. (2007), Méthode numérique lagrangienne pour la simulation d’écoulements à surface libre: application aux turbines Pelton, PhD thesis, Ecullly, Ecole centrale de Lyon.
- Marongiu, J.-C., Leboeuf, F., Caro, J. and Parkinson, E. (2010), ‘Free surface flows simulations in pelton turbines using an hybrid sph-ale method’, *Journal of Hydraulic Research* **48**(S1), 40–49.
- Marrone, S., Antuono, M., Colagrossi, A., Colicchio, G., Le Touzé, D. and Graziani, G. (2011), ‘ δ -sph model for simulating violent impact flows’, *Computer Methods in Applied Mechanics and Engineering* **200**(13-16), 1526–1542.
- Masciopinto, C. and Palmiotta, D. (2013), ‘Relevance of solutions to the navier-stokes equations for explaining groundwater flow in fractured karst aquifers’, *Water Resources Research* **49**(6), 3148–3164.
- Michel, J. (2020), Développements numériques de la méthode SPH couplée aux Eléments Finis appliqués au phénomène de l’hydroplanage., PhD thesis, École centrale de Nantes.
- Michel, J., Vergnaud, A., Oger, G. and LeTouzé, D. (2021), ‘Considerations on particle shifting technique for sph schemes’.
- Michell, J. H. (1890), ‘V. on the theory of free stream lines’, *Philosophical Transactions of the Royal Society of London.(A.)* pp. 389–431.
- Mokos, A., Rogers, B. D. and Stansby, P. K. (2017), ‘A multi-phase particle shifting algorithm for sph simulations of violent hydrodynamics with a large number of particles’, *Journal of Hydraulic Research* **55**(2), 143–162.
- Molteni, D. and Colagrossi, A. (2009), ‘A simple procedure to improve the pressure evaluation in hydrodynamic context using the sph’, *Computer Physics Communications* **180**(6), 861–872.
- Monaghan, J. (1985), ‘Extrapolating b splines for interpolation’, *Journal of Computational Physics* **60**(2), 253–262.
- Monaghan, J. (1989), ‘On the problem of penetration in particle methods’, *Journal of Computational physics* **82**, 1–15.
- Monaghan, J. J. (1992), ‘Smoothed particle hydrodynamics’, *Annual review of astronomy and astrophysics* **30**(1), 543–574.

- Monaghan, J. J. (1994), ‘Simulating free surface flows with sph’, *Journal of computational physics* **110**(2), 399–406.
- Monaghan, J. J. (2000), ‘Sph without a tensile instability’, *Journal of computational physics* **159**(2), 290–311.
- Monaghan, J. J. (2005), ‘Smoothed particle hydrodynamics’, *Reports on progress in physics* **68**(8), 1703.
- Monaghan, J. J. (2012), ‘Smoothed particle hydrodynamics and its diverse applications’, *Annual Review of Fluid Mechanics* **44**, 323–346.
- Monaghan, J. J. and Lattanzio, J. C. (1985), ‘A refined particle method for astrophysical problems’, *Astronomy and astrophysics* **149**, 135–143.
- Morris, J. (1996), ‘Analysis of sph with applications’, *Monash University, Melbourne, Australia*.
- Morris, J. P., Fox, P. J. and Zhu, Y. (1997), ‘Modeling low reynolds number incompressible flows using sph’, *Journal of computational physics* **136**(1), 214–226.
- Nasar, A. M., Fourtakas, G., Lind, S. J., Rogers, B. D., Stansby, P. and King, J. (2021), ‘High-order velocity and pressure wall boundary conditions in eulerian incompressible sph’, *Journal of Computational Physics* **434**, 109793.
- Nestor, R. M., Basa, M., Lastiwka, M. and Quinlan, N. J. (2009), ‘Extension of the finite volume particle method to viscous flow’, *Journal of Computational Physics* **228**(5), 1733–1749.
- Nestor, R. and Quinlan, N. (2007), ‘Extension of the finite volume particle method to higher order accuracy and viscous flow.’, ” *SPHERIC-Smoothed Particle Hydrodynamics European Research Interest Community*”. p. 95.
- Nestor, R. and Quinlan, N. (2013), ‘Application of the meshless finite volume particle method to flow-induced motion of a rigid body’, *Computers Fluids* **88**, 386–399.
- Neuhauser, M. (2014), Development of a coupled SPH-ALE/Finite Volume method for the simulation of transient flows in hydraulic machines, PhD thesis, Ecully, Ecole centrale de Lyon.
- Oger, G., Doring, M., Alessandrini, B. and Ferrant, P. (2006), ‘Two-dimensional sph simulations of wedge water entries’, *Journal of computational physics* **213**(2), 803–822.

- Oger, G., Marrone, S., Le Touzé, D. and De Leffe, M. (2016), ‘Sph accuracy improvement through the combination of a quasi-lagrangian shifting transport velocity and consistent ale formalisms’, *Journal of Computational Physics* **313**, 76–98.
- Oñate, E., Idelsohn, S., Zienkiewicz, O. and Taylor, R. (1996), ‘A finite point method in computational mechanics. applications to convective transport and fluid flow’, *International journal for numerical methods in engineering* **39**(22), 3839–3866.
- Pineda Rondon, S. F. (2017), Numerical prediction of cavitation erosion, PhD thesis, Lyon.
- Pineda, S., Marongiu, J.-C., Aubert, S. and Lance, M. (2019), ‘Simulation of a gas bubble compression in water near a wall using the sph-ale method’, *Computers and Fluids* **179**, 459–475.
- Price, D. J. (2012), ‘Smoothed particle hydrodynamics and magnetohydrodynamics’, *Journal of Computational Physics* **231**(3), 759–794.
- Price, D. J. and Monaghan, J. J. (2005), ‘Smoothed particle magnetohydrodynamics—iii. multidimensional tests and the $b=0$ constraint’, *Monthly Notices of the Royal Astronomical Society* **364**(2), 384–406.
- Quinlan, N. J. (2018), ‘Extensions of the meshless finite volume particle method (fvpm) for static and dynamic free-surface flows’, *Computers and Fluids* **177**, 33–45.
- Quinlan, N. J., Basa, M. and Lastiwka, M. (2006), ‘Truncation error in mesh-free particle methods’, *International Journal for Numerical Methods in Engineering* **66**(13), 2064–2085.
- Raviart, P.-A. (1985), An analysis of particle methods, in ‘Numerical methods in fluid dynamics’, Springer, pp. 243–324.
- Renaut, G.-A. (2015), Schémas d’ordre élevé pour la méthode SPH-ALE appliquée à des simulations sur machines hydrauliques, PhD thesis, Ecully, Ecole centrale de Lyon.
- Rentschler, M., Marongiu, J., Neuhauser, M. and Parkinson, E. (2018), ‘Overview of sph-ale applications for hydraulic turbines in andritz hydro’, *Journal of Hydrodynamics* **30**(1), 114–121.
- Schaback, R. and Wendland, H. (2006), ‘Kernel techniques: from machine learning to meshless methods’, *Acta numerica* **15**, 543–639.
- Shadloo, M. S., Oger, G. and Le Touzé, D. (2016), ‘Smoothed particle hydrodynamics method for fluid flows, towards industrial applications: Motivations, current state, and challenges’, *Computers and Fluids* **136**, 11–34.

- Shadloo, M. S., Zainali, A., Yildiz, M. and Suleman, A. (2012), 'A robust weakly compressible sph method and its comparison with an incompressible sph', *International Journal for Numerical Methods in Engineering* **89**(8), 939–956.
- Skillen, A., Lind, S., Stansby, P. K. and Rogers, B. D. (2013), 'Incompressible smoothed particle hydrodynamics (sph) with reduced temporal noise and generalised fickian smoothing applied to body–water slam and efficient wave–body interaction', *Computer Methods in Applied Mechanics and Engineering* **265**, 163–173.
- Sulsky, D., Chen, Z. and Schreyer, H. L. (1994), 'A particle method for history-dependent materials', *Computer methods in applied mechanics and engineering* **118**(1-2), 179–196.
- Sun, P., Colagrossi, A., Marrone, S., Antuono, M. and Zhang, A.-M. (2019), 'A consistent approach to particle shifting in the δ -plus-sph model', *Computer Methods in Applied Mechanics and Engineering* **348**, 912–934.
- Sun, P., Colagrossi, A., Marrone, S. and Zhang, A. (2017), 'The δ plus-sph model: Simple procedures for a further improvement of the sph scheme', *Computer Methods in Applied Mechanics and Engineering* **315**, 25–49.
- Swegle, J. W., Hicks, D. L. and Attaway, S. W. (1995), 'Smoothed particle hydrodynamics stability analysis', *Journal of computational physics* **116**(1), 123–134.
- Taylor, G. (1966), 'Oblique impact of a jet on a plane surface', *Phil Trans Roy Soc* **260**(1110), 96–100. cited By 47.
- Tsuruta, N., Khayyer, A. and Gotoh, H. (2013), 'A short note on dynamic stabilization of moving particle semi-implicit method', *Computers and Fluids* **82**, 158–164.
- Vacondio, R., Altomare, C., De Leffe, M., Hu, X., Le Touzé, D., Lind, S., Marongiu, J.-C., Marrone, S., Rogers, B. D. and Souto-Iglesias, A. (2020), 'Grand challenges for smoothed particle hydrodynamics numerical schemes', *Computational Particle Mechanics* pp. 1–14.
- Vacondio, R. and Rogers, B. (2017), 'Consistent iterative shifting for sph methods'.
- Vacondio, R., Rogers, B., Stansby, P. K. and Mignosa, P. (2016), 'Variable resolution for sph in three dimensions: Towards optimal splitting and coalescing for dynamic adaptivity', *Computer Methods in Applied Mechanics and Engineering* **300**, 442–460.
- Vacondio, R., Rogers, B., Stansby, P. K., Mignosa, P. and Feldman, J. (2013), 'Variable resolution for sph: a dynamic particle coalescing and splitting scheme', *Computer Methods in Applied Mechanics and Engineering* **256**, 132–148.

- Van Leer, B. (1979), 'Towards the ultimate conservative difference scheme. v. a second-order sequel to godunov's method', *Journal of computational Physics* **32**(1), 101–136.
- Veen, D. J. (2010), A smoothed particle hydrodynamics study of ship bow slamming in ocean waves, PhD thesis, Curtin University.
- Vergnaud, A. (2020), Améliorations de la précision et de la modélisation de la tension de surface au sein de la méthode SPH, et simulations de cas d'amerrissage d'urgence d'hélicoptères., PhD thesis, École centrale de Nantes.
- Vieira, A., García Feal, O., Domínguez, J., Crespo, A., Gesteira, M. and English, A. (2017), 'Graphical user interface for sph codes: Designsphysics'.
- Vila, J. (1999), 'On particle weighted methods and smooth particle hydrodynamics', *Mathematical models and methods in applied sciences* **9**(02), 161–209.
- Violeau, D. (2012), *Fluid mechanics and the SPH method: theory and applications*, Oxford University Press.
- Violeau, D. and Rogers, B. D. (2016), 'Smoothed particle hydrodynamics (sph) for free-surface flows: past, present and future', *Journal of Hydraulic Research* **54**(1), 1–26.
- Wendland, H. (1995), 'Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree', *Advances in computational Mathematics* **4**(1), 389–396.
- Wendland, H. (2004), *Scattered Data Approximation*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press.
- Xu, R., Stansby, P. and Laurence, D. (2009), 'Accuracy and stability in incompressible sph (isph) based on the projection method and a new approach', *Journal of Computational Physics* **228**(18), 6703–6725.
- Yang, X., Liu, M., Peng, S. and Huang, C. (2016), 'Numerical modeling of dam-break flow impacting on flexible structures using an improved sph-ebg method', *Coastal Engineering* **108**, 56–64.
- Zhu, Y., Fox, P. J. and Morris, J. P. (1999), 'A pore-scale numerical model for flow through porous media', *International journal for numerical and analytical methods in geomechanics* **23**(9), 881–904.